



S.I.BI.M

Total

GSA

ROL	APELLIDO	NOMBRE	CI	E-mail	TEL/CEL
Coordinador	Gularte	Gonzalo	5420070-7	gonzalogulartemvd@gmail.com	094742318
Sub-Coordinador	Iriart	Santiago	5576229-3	santiagoiriart9@gmail.com	093982489
Integrante 1	Acosta	Agustín	5213710-8	agustin14acosta@gmail.com	091027326
Integrante 2	Sunes	Bruno	5258299-1	brunosunes@gmail.com	092111308
Integrante 3	Rosales	Zendhel	6265313-4	zendhelros@gmail.com	096242708

Fecha de culminación:

09/11/2020

TERCERA ENTREGA



Índice

Carátula	1
Índice	2
Introducción	8
Requerimientos de la asignatura Inglés.....	9
Identificación de los integrantes	9
Informe sobre lo que se realizó	10
Actualización del informe.....	11
Abstract	13
Presentation script	14
S.W.O.T Analysis	15
Requerimientos de la asignatura Formación Empresarial.....	16
Resumen ejecutivo	16
Analisis del mercado	19
F.O.D.A	24
Entrevista	26
Plan Estratégico	28
Organigrama	29
Cargos	32
Plan de inversiones	37
Recibo de sueldos	39
Características del producto	44
Plan financiero	45
Plan de comercialización	46
Requerimientos de la asignatura Sociología	47
Pautas de la entrevista a Médico	47
Transcripción entrevista realizada a Médico	49
Análisis entrevista	54
Requerimientos de la asignatura Proyecto	55
Nombre del grupo, integrantes y sus roles	55
Organización de la estructura de desarrollo	55
Formato de actas de reuniones formales e informales	56
Reglas del grupo	56
Ciclo de vida del proyecto	57



Actas de reuniones	60
Diagramas GANTT Primer Entrega.....	61
Análisis y Diseño de Aplicaciones	61
Base de Datos	62
Formación Empresarial	63
Inglés	64
Programación	65
Proyecto	66
Sistemas Operativos	68
Taller	69
PERT Primer Entrega	71
Análisis y Diseño de Aplicaciones	71
Base de Datos	71
Formación Empresarial	71
Inglés	72
Programación	72
Proyecto	72
Sistemas Operativos	73
Taller	73
Varianza Primera Entrega	74
Diagramas GANTT Segunda Entrega.....	76
Análisis y Diseño de Aplicaciones	76
Base de Datos	77
Formación Empresarial	78
Inglés	80
Programación	81
Proyecto	82
Sistemas Operativos	83
Sociología	85
Taller	86
PERT Segunda Entrega	87
Análisis y Diseño de Aplicaciones	87
Base de Datos	87
Formación Empresarial	87

Inglés	88
Programación	88
Proyecto	88
Sistemas Operativos	88
Sociología	89
Taller	89
Varianza Segunda Entrega	90
Diagrama GANTT Total	92
Formación Empresarial	92
Proyecto	94
Ingles	96
Taller.....	97
Sistemas Operativos	99
Base de Datos	101
Programación	103
Sociología	105
Análisis y Diseño de Aplicaciones	106
PERT Tercera Entrega	108
Análisis y Diseño de Aplicaciones	108
Base de Datos	108
Formación Empresarial	108
Ingles	108
Programación	109
Proyecto	109
Sociología	109
Sistemas Operativos	109
Taller	109
Total	110
Varianza Tercera Entrega	112
Plan de contingencias	118
Métricas	119
Requerimientos de la asignatura Análisis y Diseño de Aplicaciones	120
Relevamiento	120
Preguntas Letra Proyecto	120

Preguntas ADA.....	120
Preguntas Formación Empresarial	121
Preguntas Sociología	121
Preguntas Sistemas Operativos	122
Preguntas Ingles	122
Preguntas Proyecto	122
Preguntas Taller.....	123
Estudio de factibilidades	125
Cálculo de métricas por punto de función y Mc Cabe	127
Especificación de requerimientos	128
Glosario	134
F.O.D.A	135
Análisis Costo-Beneficio	137
Diagrama de Clases UML	139
Modelo Esencial	140
Diccionario de datos.....	144
Especificación de procesos	145
Diagrama Warnier-Orr.....	154
Casos de uso.....	155
Diagrama de estados	171
Carta jerárquica	172
Plan de testing.....	180
Agenda de reuniones	218
Requerimientos de la asignatura Base de Datos	219
DER (Primer Entrega).....	219
RNE	219
Diccionario de Datos.....	220
DER (Segunda Entrega).....	222
RNE	223
Diccionario de Datos.....	224
Permisos de usuarios.....	229



DER Final (Tercera Entrega)	233
RNE	233
Diccionario de Datos.....	234
Código de la Base de Datos	239
Política de respaldos	272
Requerimientos de la asignatura Sistemas Operativos	273
SO a utilizar en terminales de usuario	273
SO a utilizar en los servidores.....	275
Manual de instalación del SO en el servidor.....	277
Manual de instalación de MySQL junto a requisitos	288
Configuración del servicio SSH en el cliente y el servidor.....	291
Configuración de red	295
Scripts.....	296
Rutinas Backup (Respaldo)	299
Descripción de Logs (Scripts)	301
Configuración Firewall y SELinux	302
Requerimientos de la asignatura Programación.....	304
Identificación de interfaces del sistema	304
Código para la comprobación de datos ingresados	304
Código que no permite el ingreso de números o símbolos	304
Código que no permite el ingreso de espacios	305
Código que no permite el ingreso de letras.....	305
Código fuente	306
Usuarios y claves del sistema	593
Requerimientos de la asignatura Taller	594
Servicio de Servidores	594
Puesto de trabajo genérico	595
Servidor	597
Servidor de respaldo (NAS)	598
Servidor WEB/MAIL.....	598



Router Firewall	600
Router	600
Switches	601
Racks	602
Rack de Pared	603
Pachera	603
KVM.....	604
UPSS	604
Cable de red	604
Ducto Pvc	606
Bandeja Portacables.....	606
Bandeja Portacables Union.....	607
Conector Ficha RJ45	607
Botitas	607
Caja Red	608
Keystone.....	608
Sistema operativo para puesto de trabajo genérico.....	609
Sistema operativo para el servidor.....	610
Esquema lógico	611
Planos.....	613
Servicios montados en el Router	616
Manual de procedimientos ante fallas del sistema	619
Software de monitoreo	621
Anexos	625
Análisis y Diseño de Aplicaciones	625
Métricas.....	625
Cálculo de métricas por McCabe	627
Agenda de reuniones.....	634
Manual de administrador	691
Manual de usuario.....	705

S.I.BI.M
I.T.I-C.E.T.P
3° BB



Formación Empresarial.....	789
Formularios DGI 0351/0352	789
Contrato de forma jurídica	793
Sistemas Operativos	794
Datos necesarios para usar el servidor	794
Bibliografía.....	795

Introducción

Se desarrollará y documentará un sistema informático destinado a facilitar la gestión de registro de la actividad médica, dicho sistema será elaborado como un producto de predicción basado en la experiencia y la bibliografía académica.

Requisitos de la asignatura Ingles

Identificación de los integrantes de cada equipo de trabajo, los criterios que se siguieron para la conformación del grupo y explicación de los roles que se asignaron.

Criteria for the formation of the group:

We formed this group based on opinions, similar thoughts, a few friendships and the experience obtained the previous year of the groups for the final project. Its name reference to the initial letters of our names (Gonzalo Santiago Agustín).

Criteria for the roles:

Member Agustín Luciano Acosta Bernardo: Agustín thinks that it's so much responsibility being a Coordinator or Sub-Coordinator, so he prefers to be just a member.

Member Bruno Daniel Sunes Pianavilla: Bruno started working with us because of the separation of his group. He didn't have the opportunity to choose a role.

Sub-Coordinator Santiago Miguel Iriart Guibernau: Santiago chose this role because he likes the responsibility that it entails.

Coordinator Gonzalo Lisandro Gularte Artola: He chose this role because he knows how to organize the documentation and the activities in order to achieve all the objectives in each phase of the project.

Informe sobre lo que se realizó para las asignaturas del área tecnológica con énfasis en “Formación Empresarial”.

Project management: The objective is to keep record about the rules, meetings and planify the time it should take to finish the project. To estimate that we use GANTT & PERT that are tools which help to calculate it.

Software analysis and design: The objective is to make an analysis of the program that we were asked to make and the main requirements that it must have. It has to be complemented with some “tools” like S.W.O.T analysis to know the strengths and weaknesses of the product and also of the business.

Operating Systems III: We have to choose an operative system with its version and a guide book to install it. We have to do a program which allows the user to create, modify and delete groups or users.

Programming III: The main objective is that we have to develop the graphic interface in every window without neglecting the code part.

Database Management Systems II: We have to do the ER Model (Entity-Relationship Model) which is a diagram that gives a complete view of the database in order to not make mistakes when making it.

Computer Maintenance & Computer Networks: Deberemos de seleccionar y fundamentar los equipos que se van a usar para los puestos de trabajo y para la sala de servidores, también se deben de ubicar en un plano de la institución médica que cuenta con planta baja y primer piso.

Business training: We have to create a business with general information about it such as the name (GSA) the partners etc. We also have to fill out a form of DGI-BPS to create the business and a contract to say that the business will be a public limited company.

Sugerencias de aportes específicos desde la asignatura inglés (tanto a partir de las inquietudes que los colegas hayan presentado a los alumnos como de los propios estudiantes) para optimizar la elaboración del software, así como su eventual comercialización.

We would like to work more on the vocabulary related to the software development industry.



Actualización del informe sobre lo que se está trabajando en base a los requerimientos de las asignaturas del área tecnológica.

Project management: For this part, as the same as in the first one, the minutes of meetings and the GANTT Diagram, one for each subject and a total one, should be made only of the second part.

Then, the management documentation such as the planification and project control documentation will be made too, along with a contingency plan.

Software analysis and design: In this second part of the project we need to do the cost-benefit analysis. It's a way to know if the IT System is a good inversion to the organization. Some diagrams have to be made, such as, "UML Diagrams" "Nassi-Schneiderman Diagrams" "State Diagrams" "Warnier-Orr Diagrams" "Use Case Diagrams" It helps in the future of the development in our system.

Operating Systems III: For this part, the internet configuration in terminals and in the server is going to be made, along with the configuration of the users creation with his corresponding permits. To end with, some scripts about the automation of different processes such as backups will be made too.

Programming III: In this second part of the project, we need to coding the connection between the app and the database. The app programmed will be used to make all CRUD modifications (Create, Read, Update, Delete) for the data entered by the users of the system.

Database Management Systems II: In the second part of the project, the database physical model will be implemented, here all the entity and relation tables that were made with SQL sentences is where they will be , in turn, a study about the database users permissions will be made, so then, implement them in the physical model that was created before.



Business training: In this second part of the project we will design two plans: Strategic Plan and Operative Part.

The Strategic plan will detail the vision and the mission of our organization.

The vision gives information about our ethical values and our mission details our aims as a business. On the other hand, the operative plan details which human resources we have and the inversions that we will have to assigned to the develop of the entrepreneurship

Computer Maintenance & Computer Networks: In this second part of the project we will do an analysis about the service which will be provided by the business that bought the software, the documentation and structured cabling that will be implemented.

Later, we will need to know the material needed to implement the net and the Computer Equipment.

Sociology: For this part, we have to interview a doctor with questions about diagnostics, diseases and the software and then transcribe the oral interview so we can use the information collected to support us and help the development of the software to be according to the requirements of a health professional.



Abstract

This product is aimed at public and private health institutions of Uruguay and the region, also, it was designed from the study of functional and nonfunctional requirements together with other analysis and design tools.

The developed system offers different functionalities to patients, doctors and medical auxiliaries, such as admission and consultation of a interview between a patient and a doctor, also, it gives the opportunity to enter, modify, delete and consult clinic analyses, treatments and diagnoses made by a patient in the medical institution. It also provides the option to manage forms by allowing medical staff to add and delete forms as well as, enter, modify and delete questions and then add answers to them.

Once the interview between the patient and the doctor has finished, the system suggests possible illnesses.

If the doctor thinks that the suggestion is not good, he can do a diagnoses of the patient and if it is necessary enter an clinical analysis to verify that matches the diagnoses.

The system validates the type and size of data it helps not to enter wrong information.

To make possible this functionalities a connection between the program and the database is needed.

To put the program in a work area is necessary to have the IT devices infrastructure previously installed. That will allow the user to use the system and will allow others activities in the medical institution.

The price of the product has been established using the Function Point Analysis, also the cost-benefit relation and the time we had to develop the software.



Presentation script

To implement the software, it's necessary to implement the infrastructure to be used and that will allow other activities in the medical institution. Workstations and access points will be installed for the patients and their companions, one server for the database, another one for the backups and other for the web and mail used by the institution. To keep the network and workstations secure, measures will be taken, for example, a antivirus for the workstations and servers, the creation of users with the permissions only to do what they're allowed to, the configuration of the firewall in the servers, the creation of virtual networks to prevent the access of unwanted parts of the network, the amount of workstations that can be connected to the network were limited and a router firewall was implemented which can allow or deny the access to the medical institution network.

To achieve the objective, it is necessary to carry out a metrics study to determine the costs and price of the product and then carry out a cost-benefit analysis of the project to decide whether it is worth or not to invest in it. Also, to achieve the objective of developing the system, we need to use planning, analysis and design tools, such as C.A.S.E (Computer-Aided Software Engineering) that contribute to the development of the product.

Finally, the results of our project will be reflected in the functionality, reliability, robustness and performance provided by our computer system.



S.W.O.T Analysis

SWOT ANALYSIS GSA

INTERNAL FACTORS	
STRENGTHS (+)	WEAKNESSES (-)
<p>Great working environment: There is a pleasant bond between the members of the company.</p> <p>Optimal computer equipment for work: We have equipment for carrying out the tasks of developing the software.</p> <p>Extensive IT training: Every member of the company is academically trained in IT.</p> <p>Knowledge of communications tools: We know how to handle tools such as Zoom, Discord, Skype and others to achieve optimal communication between the members or member-costumer.</p>	<p>Few workers: We do too many workers so that it could take more time to develop the software.</p> <p>Limited knowledge about how to manage a Company: We do not have enough experience how to organize administratively a Company.</p> <p>Few economic resources: We are a new Company so that we do not have too much money.</p> <p>Difficulties to get a bank loan: As we are a small Company, we have few opportunities to get a bank loan.</p>
EXTERNAL FACTORS	
OPPORTUNITIES (+)	THREATS (-)
<p>Situation in the world now: The Company can develop software related to the health.</p> <p>Constantly growing IT Sector: The IT sector is growing really fast and when this happens, it usually needs more people to work.</p> <p>There are clients that are waiting to find a company that changes the market: There are clients that are waiting a change in the products.</p>	<p>A lot of competition: There are many consolidated companies recognized by their market experience with the potential to attract customers.</p> <p>Different clients who are going through an economic crisis: Due to the pandemic, they are affected by an economic crisis and don't have enough money either to buy our product or service.</p> <p>The appearance of new technologies, products or services: Not only companies developing a similar product to ours, but also developers amateurs who offer new products or services.</p> <p>Possible non-acceptance of some change or innovation in the market by the customers, which causes the product or service to not be successful: Just as you find clients who are waiting for a change, some clients prefer the classic versions and don't want a change.</p>



Requerimientos de la asignatura Formación Empresarial

Resumen ejecutivo

1.1 Nombre de la empresa: **GSA**

Pensamos en un nombre corto ya que es fácil para nombrar, además la creación del mismo se debe a las iniciales de nuestros nombres.

Logo:



Como toda empresa, aspiramos en algún momento llegar a tener nuestras instalaciones en un edificio como este.

La justificación de los colores del diseño del logo se deben a que logran una gran combinación monocromática entre tonos azules.

1.2 Actividad Comercial

Actividad principal: Desarrollo de Software

Actividad secundaria: Mantenimiento de equipos

Mantenimiento de servidores

Testing

Diseño e instalación de Cableado Estructurado

1.3 Ubicación geográfica

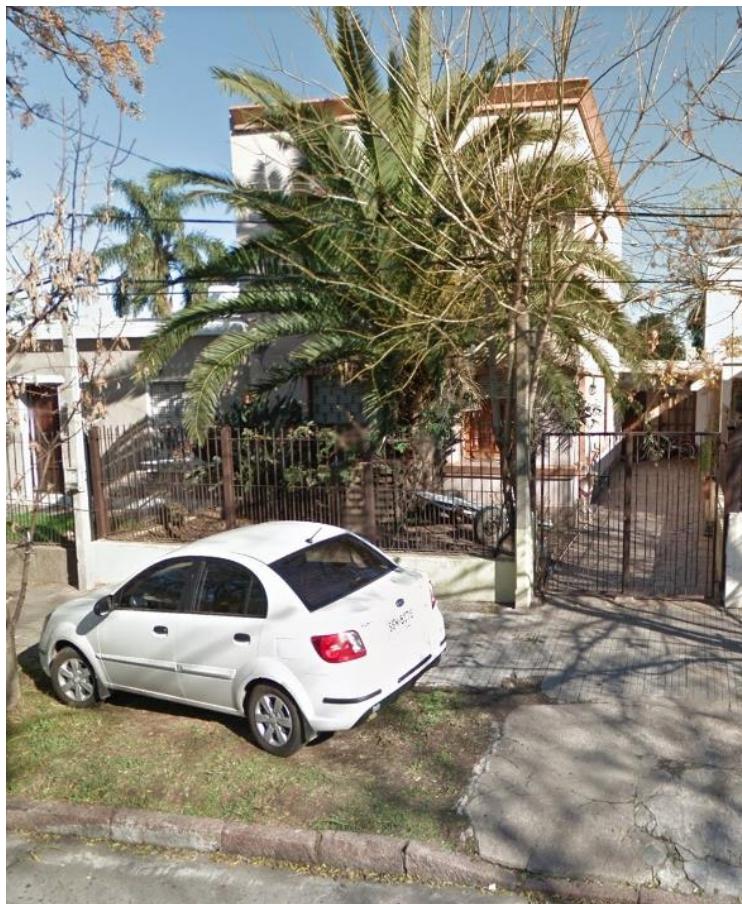
-Mapa

Quijote 2885 Esquina Gustavo Gallinal

Barrio La Blanqueada

Código Postal: 11600

Departamento de Montevideo, Uruguay



Fotos extraídas de Google Maps

Nuestra empresa será ubicada en la casa de uno de los integrantes de la misma, esto nos hará dedicar menos recursos económicos al lugar de trabajo, en el caso que un cliente quiera organizar una reunión podemos optar vía aplicaciones de comunicación como Zoom o si el cliente desea una reunión presencial, se alquilará una Smart Office para realizar el encuentro.

Smart Office

Se alquilará una Smart Office para encuentros con clientes ubicada en el Palacio Salvo, Plaza Independencia 848 a un precio de 20 dólares la hora. La misma cuenta con los siguientes componentes:

- Recepción multilingüe (Español/Inglés/Portugués).
- Pantallas LED de alta definición, pizarras y chromecast.
- Sala de videoconferencia con equipamiento Polycom.
- Acceso a Internet inalámbrico. Frigobar y café con autoservicio. Acceso a Printing Center.



- Croquis / Plano



Análisis de mercado

Resumen del informe CUTI 2015

Ventas

Las tecnologías de la información, representado por las empresas asociadas al cuti creció en 2015. Los datos presentan que en las ventas han alcanzado US\$1068 mill el año anterior, lo que se confirma un crecimiento de 6,5% respecto al año anterior. Dejando de lado a ANTEL las ventas disminuyen en US\$861 mill, lo que supera en un 5,7% al año anterior

Mercado Externo

En 2015, las exportaciones de empresas uruguayas(tanto desde Uruguay y socios en el exterior). Alcanzaron US\$277 millones. Las exportaciones se situaron en US\$ 211 mill el año anterior, lo que dio en una caída que rozó el 7% al valor esperado en 2014. A pesar de esto, se debe recalcar que han tenido un crecimiento muy significativo mirándolo desde la perspectiva del plazo medio. Se demuestra mirando la tasa de crecimiento de los pasados 5 años la cual se sitúa en 3,5%

Por otro lado las ventas realizadas en el exterior del país alcanzaron los US\$66 mill. Aumentando en un 30% con respecto al año anterior.

Las empresas asociadas a CUTI demuestran que tienen una alta tasa de exportación teniendo en cuenta la cantidad de empresas que venden al mundo en 2015(rozando el 80%), A pesar de esto, bajan la intensidad considerando la participación que estas tienen en el total de la facturación.

ORIENTACIÓN EXPORTADORA POR SEGMENTO DE NEGOCIO

Fuente : Encuesta Anual Cuti 2015

Segmento	Ventas al Resto del Mundo (mill. de US\$)	% de empresas exportadoras	% de ventas al resto del mundo
Servicios TI	106	77%	33%
Horizontal	85	89%	43%
Vertical	81	84%	60%
Infraestructura TI	6	40%	1%
Total	277	79%	26%

³ Las ventas desde sucursales y socios de 2014 fue revisada desde US\$ 58 millones hasta US\$ 51 millones.

Destino de las exportaciones desde Uruguay

Durante 2015 Uruguay exporta a más de 40 países en América, Europa, Asia y Oceanía, entre ellos se destaca Estados Unidos como principal destino de las exportaciones de productos y/o desde Uruguay con una participación de 40,7% en ventas totales.

RANKING DESTINOS DE EXPORTACIÓN

Fuente: Encuesta Anual Cuti 2015

Ranking	País	Participación en ventas totales
1	 Estados Unidos	40,7%
2	 Argentina	6,4%
3	 España	6,1%
4	 Colombia	5,6%
5	 Chile	5,4%
6	 México	5,2%
7	 Japón	4,3%
8	 Paraguay	3,4%
9	 Perú	3,2%
10	 Brasil	2,6%

Esta tabla proveniente de la encuesta anual CUTI 2015 muestra qué países están dentro de los principales destinos de exportación, España y Japón destacan su ingreso al TOP 10 en el año 2014, España pasó a tener un 6,1% de participación en ventas totales y Japón un 4,3%

Mientras que Brasil en el año 2014 se posicionaba en el segundo puesto del ranking, en 2015 paso a estar en el décimo puesto del ranking con un 2,6% de participación en ventas totales.

Mercado Local

En el 2015 hubo un crecimiento de la facturación del sector TI debido a que las ventas en el mercado interno aumentaron.

Las empresas socias de CUTI realizaron ventas por US\$791 millones el año pasado, un 9% más respecto a 2014. Esta cifra incluye la facturación de la empresa nacional de telecomunicaciones ANTEL. Dejando de lado a esta empresa, las ventas del mercado interno crecieron 10%, alcanzando la suma de US\$583 millones.

Según segmentos, “Infraestructura TI” se posiciona como el de mayor facturación (US\$411 millones), seguido por “Servicio TI” (US\$212 millones), en tercer puesto le sigue “Horizontal” y por último en el cuarto puesto “Vertical”. Si excluimos a la empresa nacional de telecomunicaciones ANTEL, “Servicios TI” asciende a la primera posición e “Infraestructura TI” queda en segundo lugar con una facturación que se reduce a US\$203 millones.

VENTAS AL MERCADO INTERNO

Fuente : Encuesta Anual Cuti 2015

	Millones de US\$		% de ventas al Mercado Interno
	Con ANTEL	Sin ANTEL	
Infraestructura TI	411	203	99%
Servicios TI	212	212	67%
Horizontal	113	113	57%
Vertical	55	55	40%
Total	791	583	

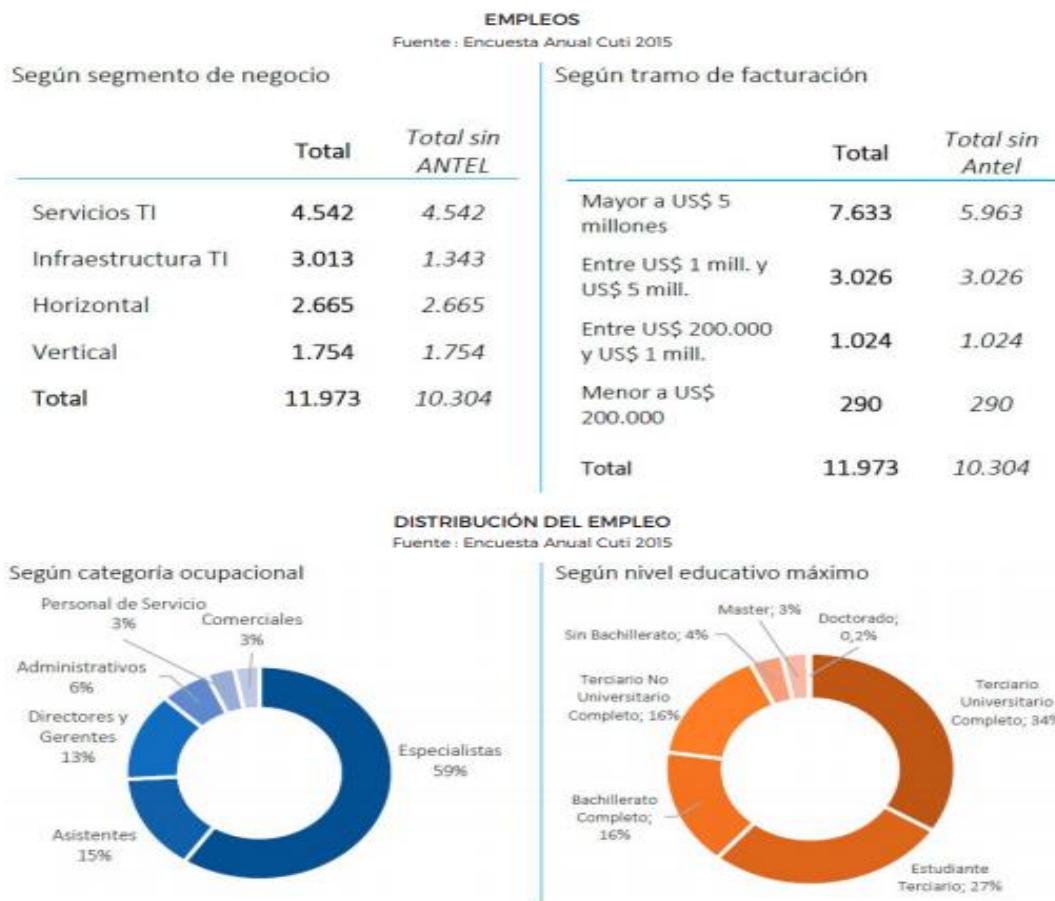
- **Infraestructura TI:** Hace referencia a un conjunto de servicios, dispositivos físicos y aplicaciones del software que sirven como sustento para los sistemas de cualquier compañía.
- **Servicio TI:** Es un conjunto de actividades las cuales buscan satisfacer las necesidades de los clientes por medio de un cambio de condición de los bienes informáticos. Por ejemplo, la instalación y mantenimiento de equipos o la gestión de redes y sistemas.
- **Horizontal:** Una aplicación horizontal es aquella que podría venderse a miles y miles de clientes. Ejemplos: contabilidad, facturación, sueldos, institutos de enseñanza, abogados, médicos, laboratorios de análisis clínicos, etc.
- **Vertical:** Una aplicación vertical es la que solamente le sirve a una empresa o a una organización.

Empleo

Las empresas socias de Cuti, las cuales representan al sector TI, generaron en 2015 casi 12.000 puestos de trabajo. Excluyendo los empleos relacionados con servicios de datos de Antel, alcanzarían 10.304.

Considerando los datos recogidos sobre admisiones y desvinculaciones de personal, el empleo estimador para 2015 representó un aumento en relación a la estimación realizada el año previo. De hecho, el índice de rotación que nos permite extraer conclusiones de la movilidad de personal, se situó en 3,9%, lo que indica una suba de los puestos de trabajo del sector y un nivel de rotación que se encuentra en línea con el observado en encuestas de años anteriores.

Si consideramos los resultados de la Encuesta Continua de Hogares de 2015, el empleo generado por las actividades relacionadas a las tecnologías de la información alcanza 19.716 puestos de trabajo.



Informe Anual TI 2015

Desde hace más de 15 años la Cámara Uruguaya de Tecnologías de la Información (Cuti) realiza cada año una encuesta a sus empresas socias con el objetivo de conocer el desempeño del sector TIC del Uruguay.

El informe recoge los resultados de la encuesta realizada a las empresas entre marzo y julio de 2016. Donde se relevaron indicadores 2015 de las empresas cuya actividad principal es Tecnologías de la Información.

El informe incluyó un reporte personalizado comparativo de los principales indicadores de la empresa respecto a los del promedio de su segmento de actividad.

F.O.D.A

Internas a la empresa

- Fortalezas: Gran ambiente de trabajo:

Existe un vínculo agradable entre los integrantes de la empresa.

Equipos informáticos óptimos para trabajo:

Contamos con equipos aptos para realizar las tareas de desarrollo de Software.

Extensa formación en informática:

Todos los integrantes de la empresa cuentan con una formación académica sobre Informática.

Conocimientos sobre herramientas de comunicación:

Sabemos cómo manejar herramientas como Zoom, Discord, Skype y otras más para lograr una comunicación optima entre integrantes o integrantes-clientes.

- Debilidades: Escaso personal de trabajo:

Contamos con poco personal de trabajo lo cual nos generaría dificultades a la hora de desarrollar un Software ya que nos llevaría más tiempo de lo debido.

Limitado conocimiento acerca de la administración de una empresa:

Carecemos de poca experiencia para organizar administrativamente una empresa.

Pocos recursos económicos:

Debido a que somos una empresa que recién se inicia en el mercado contamos con escasos recursos económicos iniciales.

Pocas oportunidades para obtener un préstamo de un banco:

Como somos una empresa chica tenemos pocas oportunidades de que un banco nos ceda un préstamo.

Externas a la empresa

- Oportunidades: Contexto actual (Pandemia):

La empresa puede desarrollar Software relacionado con la salud que facilite la situación actual.

Sector de comercio informático en constante crecimiento:

El sector informático se encuentra en constante crecimiento y cada vez se solicita a más gente para trabajar, de hecho, este sector cuenta con desempleo negativo.

Ciertos clientes que están a la espera de una empresa que innove en el desarrollo de Software:

Hay clientes que esperan un cambio en los productos del mercado, es decir un Software que innove y que sea distinto a los demás.

- Amenazas: Mucha competencia:

Numerosas empresas consolidadas y reconocidas por su trayectoria en el mercado con potencial de atraer clientes.

Diferentes clientes que están pasando por una crisis económica:

Debido a la pandemia se ven afectados por una crisis económica y no tienen dinero para adquirir nuestro producto y/o servicio.

Aparición de nuevas tecnologías, productos y/o servicios:

No solamente posibles empresas que estén desarrollando un producto similar al nuestro, sino también desarrolladores amateurs que ofrecen nuevos productos y/o servicios.

Possible no aceptación a algún cambio u innovación en el mercado por parte de los clientes, lo que provoca que el producto y/o servicio no tenga éxito:

Así como se encuentran clientes que están a la espera de un cambio, también hay clientes que prefieren lo clásico y no apuestan por el cambio.

Entrevista

Pregunta 1: ¿Cómo empezaste a trabajar en el sector informático?

Fui a la Universidad por una licenciatura en ingeniería nuclear y cuando me gradué no había trabajos disponibles desde que la industria nuclear estaba en decadencia debido a la opinión pública. Entonces necesité cambiar de Carrera al mismo tiempo en que la industria de Informática estaba en pleno crecimiento debido a todo el trabajo de Y2K**. Lo hice muy bien como programador y la compañía decidió a pesar de estar despidiendo a la mayoría del personal el cual participó significativamente en el problema de Y2K. Convertí esa oportunidad en una Carrera exitosa la cual trabajo hasta hoy en día.

Pregunta 2: ¿Cómo funciona el desarrollo de software en los Estados Unidos?

Básicamente hay 3 maneras en los que las compañías (también llamadas consumidores) que necesitan un sistema o software lo aborden:

1. Ellos tienen lo que se le llama como una empresa desarrolladora de software en casa. En este caso, un consumidor decide que la mejor manera para obtener un software es crearlo por ellos mismos. Para muchas compañías esta opción puede ahorrar dinero, pero ellos necesitan tener las personas correctas para hacerlo. Ya que los principios de gestión de la empresa están impulsados por su negocio principal y a menudo no funcionan con tecnología de la información
2. Otro caso es cuando un consumidor contrata a una empresa que se encargará de hacer software personalizado para ellos. La ventaja aquí es tener a alguien más que sea un experto en el campo, crear lo que necesites y siempre estar ahí para adaptarlo y expandirlo si la compañía crece y cambia. El proveedor es generalmente mejor diseñando y creando el software necesario que una empresa desarrolladora de software en casa. Esta opción puede ser más barata o más cara que la opción 1 dependiendo en como el contrato está escrito para proveer el desarrollo de software. Por ejemplo, algunos proveedores de software hacen mucho dinero en tarifas de licenciamiento y mantenimiento que deben ser pagadas en un determinado tiempo (usualmente un año) lo cual puede salirse de control si tu comprador no se da cuenta de esto cuando firma el contrato.
3. Este tercer caso es para el consumidor que compra software que ya está hecho y lo instalan en su red. Esto es usualmente referido como “off the shelf”. Los que producen este software lo hacen de manera muy genérica para que sea fácil adaptarlo a diferentes industrias. Cuando los consumidores compran este tipo de productos tienen que tener en cuenta la cantidad de cambios que le deben de hacer al código para que

se adapte a su negocio. Los cambios pueden ser cambios en el código, testeos, cambios a la base de datos etc. Licencias anuales y tarifas de mantenimiento pueden aplicar en este caso. Dependiendo en como el contrato esté escrito. En varios casos esto es la opción más barata, el problema es que los consumidores pueden no llegar a encontrar lo que están buscando como en las opciones 1 y 2.

Pregunta 3: ¿Qué tipo de Software es el más demandado por las compañías?

Para esta pregunta, solo puedo especular basándome en mi experiencia, la cual es limitada. Diría que el software para las operaciones administrativas es el más demandado. Por Google: Las operaciones administrativas incluyen procesos usados por empleados que ayudan que el negocio funcione sin problemas. Contabilidad, financiamiento, inventario, cumplimiento de orden, distribución y envíos son ejemplos de operaciones administrativas.

Pregunta 4: ¿Qué elementos deberías de tener en cuenta para fijar el precio de un Software?

Hay muchos, pero probablemente los elementos más importantes son:

1. ¿Cuántas líneas de código tendrás que escribir?
2. ¿Cuántas conexiones va a tener el software con otros softwares?
3. ¿Cuánta información va a tener que almacenar y procesar el software?
4. ¿Cuántas transacciones en un periodo de tiempo dado va a tener que procesar el software?
5. ¿Cuántos usuarios habrá en el sistema a la misma vez?
6. ¿Qué pila tecnológica será usada? La pila tecnológica es el conjunto de tecnológicas, programas informáticos y herramientas que se utilizan en el desarrollo y despliegue de una aplicación o sistema informático.
7. ¿Cuánto mantenimiento y/o personalización se le tendrá que hacer al software para cada consumidor?
8. ¿El software será alojado en el sistema del consumidor o el proveedor del software lo alojará en sus servidores?

¿Cuál es el requisito de disponibilidad del software? Las preguntas sobre disponibilidad incluyen: ¿Es aceptable que el software no esté disponible si hay errores? ¿Si es así, por cuánto tiempo? La respuesta a esta pregunta tiene un gran impacto a la hora de saber el precio. Por ejemplo, si la respuesta es que el software tendrá que estar disponible 24 horas al día, los 365 días del año, tendría que haber un sistema de respaldo duplicado e idéntico que automáticamente se haga cargo si el sistema falla.

Plan Estratégico

Visión: Apuntamos a ser reconocidos mundialmente como una empresa prestadora de servicios de desarrollo de Software, la cual destaque por ofrecer servicios confiables e innovadores con un equipo humano altamente calificado así como lograr un gran nivel de satisfacción por parte de nuestros clientes, empleados y socios a través de nuestros valores.

Misión: Nuestra empresa ofrece un servicio de desarrollo de software con el objetivo de facilitar la gestión de otras organizaciones, otorgando soluciones de alta tecnología en continua optimización; con un equipo de trabajo altamente calificado. Nos interesa contribuir con la sostenibilidad del medio ambiente, así como también con aspectos sociales, tecnológicos, culturales y éticos.

Valores:

Competitividad: Aspiramos a poseer una ventaja competitiva única y sostenible en respecto a nuestros competidores.

Colaboración: Nos complementamos con nuestros clientes para mejorar día a día la calidad de las soluciones ofrecidas y así satisfacer mayormente las necesidades de nuestros clientes.

Innovación: Aspiramos a estar anticipados al cambio para así ofrecer lo último del mercado y brindar apoyo y servicio óptimo para nuestros clientes.

Transparencia: Esto implica confianza y las relaciones humanas, incluidas las comerciales, se forjan con confianza.

Excelencia: Aspiramos a llevar la calidad de nuestros productos, servicios y soluciones al máximo para así lograr ofrecer un producto excelente e impulsar a nuestros clientes a permanecer con nosotros.

Claridad: Al transmitir una idea, al cerrar un acuerdo, al explicar una serie de ventajas es vital ser claro, ya que a nadie le gusta no entender algo y tener la sensación de que queden puntos sin tratar y sin aclarar.

Ética: Impulsamos a proveer un compromiso social y cumplir con nuestra normativa interna.

Formación: Promovemos la formación continua en todos los ámbitos dentro de la empresa.

Agilidad: Aportamos velocidad y adaptabilidad en cada proyecto.

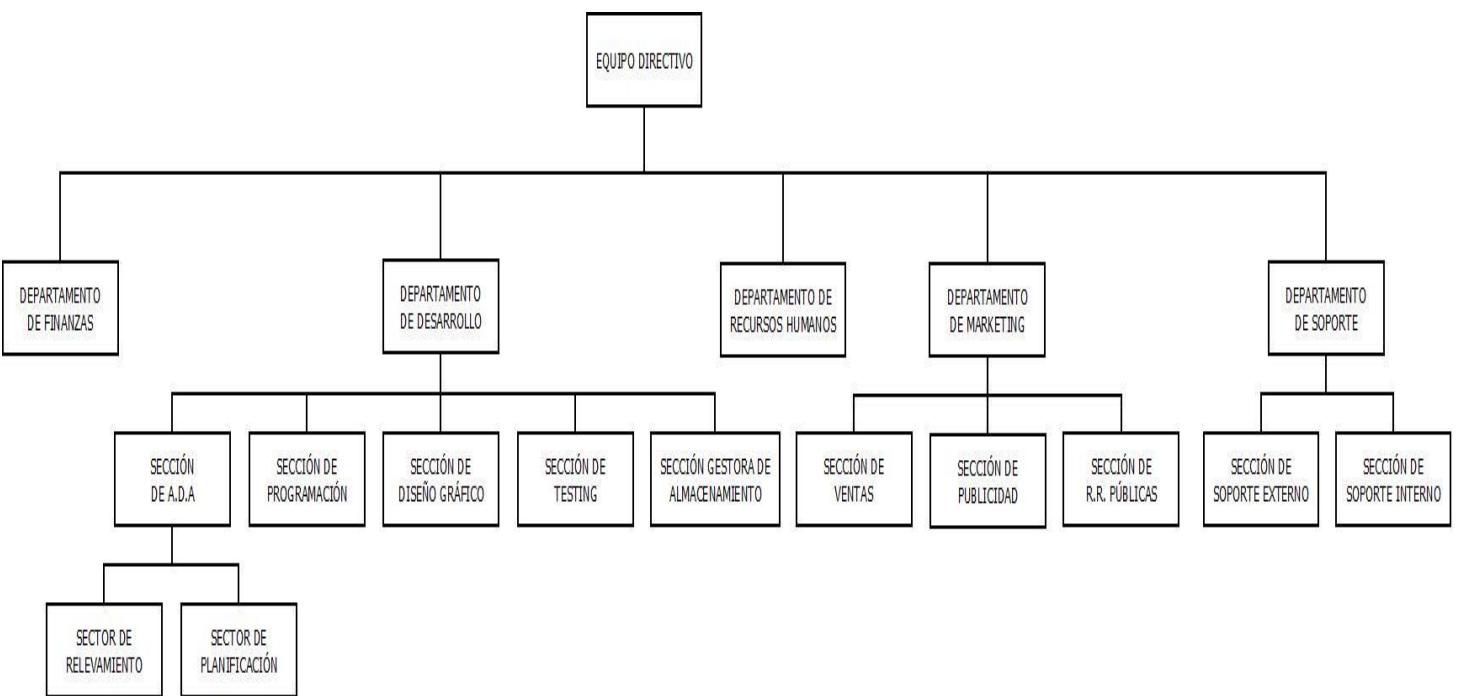
Compromiso: Aspiramos a lograr un compromiso compartiendo conocimientos, experiencias y esfuerzo, para poder ofrecer el mejor servicio. Supone trabajar con liderazgo bajo un método, en un ambiente que promueva el enriquecimiento, en el cual se valoran y se respetan todas las opiniones e ideas.

Trabajo en equipo: Promovemos y apoyamos un trabajo de equipo homogéneo.
Responsabilidad social empresaria: Contribuimos activamente al mejoramiento social, económico y ambiental, para mejorar nuestra situación competitiva y valorativa.

Objetivos:

- Aumentar la calidad de nuestros productos.
- Ser una empresa reconocida y elegida como “socio tecnológico” de sus clientes por la calidad de las propuestas y el estricto cumplimiento de los compromisos asumidos para así lograr una alta reputación entre nuestros clientes.
- Incrementar el margen anual de ingresos de manera responsable y proactiva
- Diseñar aplicaciones informáticas que se ajusten a las necesidades de otras empresas.
- Realizar campañas publicitarias de forma permanente para visibilizar nuestros productos.

Organigrama:



Descripción Organigrama:

EQUIPO DIRECTIVO: Equipo encargado de coordinar en conjunto todas las tareas y decisiones del proyecto y a su vez delegar otro tipo de responsabilidades sobre distintos departamentos.

DEPARTAMENTO DE A.D.A: (Análisis y diseño de aplicaciones): Análisis y Diseño de Aplicaciones es el departamento responsable de administrar la división de sectores como el de relevamiento y planificación.

SECTOR DE RELEVAMIENTO: Este sector se dedica a realizar, a través de técnicas (entrevistas, formularios, cuestionarios, etc.) el relevamiento de los requerimientos funcionales tanto como no funcionales que el software debe tener.

SECTOR DE PLANIFICACIÓN: Este sector se encarga del estudio de factibilidades, en donde se determina si el proyecto es factible o no en términos de recursos humanos, técnicos, operativos y legales.

A su vez en este sector se realizan varios diagramas como DFD (Diagrama de Flujo de Datos), N-S (Diagramas Nassi-Schneiderman), CDU (Casos de uso), Warnier-Orr, UML (Diagrama de Clases), etc., que posteriormente facilitarán la programación en el desarrollo de software.

DEPARTAMENTO DE FINANZAS: Responsable de la administración de recursos económicos de la organización.

DEPARTAMENTO DE DESARROLLO: Se encarga del diseño, desarrollo, implementación de software y hardware, como también del control de calidad de los productos y servicios proporcionados por la empresa. A su vez, se encarga tanto de la coordinación como la supervisión de los proyectos de la empresa.

SECTOR DE PROGRAMACIÓN: Se encarga de proveer soluciones informáticas, mediante desarrollo, programación, implantación, formación y mantenimiento de software a su cliente o clientes internos, es decir, la propia empresa.

SECTOR DE DISEÑO GRÁFICO: Es el sector encargado de diseñar la parte visual del programa que el usuario visualiza al momento de usarlo. La misma, debería ser intuitiva y amigable para hacer de su uso lo menos difícil posible.

SECTOR DE TESTING: Es el sector que se encarga de probar la aplicación cuando esta esté finalizada por el programador para así corroborar que no hayan errores y poder corregirlos a tiempo antes de entregar el producto final al consumidor

SECTOR GESTOR DE ALMACENAMIENTO: Este sector se encarga de la realización y la gestión del almacenamiento en la base de datos.

DEPARTAMENTO DE RECURSOS HUMANOS: Este departamento se encarga de la gestión del personal, es decir, se ocupa de seleccionar, contratar, emplear y retener al personal que la organización necesita para lograr sus objetivos.

DEPARTAMENTO DE MARKETING: Este departamento se encarga de gestionar todos sus sectores adyacentes que son el sector de ventas, sector de publicidad y el sector de relaciones públicas.

SECTOR DE VENTAS: Este sector se encarga de distribuir nuestro producto a los consumidores y hacer posible el alcance al mismo.

SECTOR DE PUBLICIDAD:

Es el sector que se encarga de hacer que los clientes sepan la existencia de nuestra empresa, y de sus productos y servicios, como a su vez de convencerlos de comprar nuestros productos.

SECTOR DE R.R. PÚBLICAS:

Este sector se encarga de aplicar un proceso de comunicación estratégica que construye relaciones mutuamente beneficiosas entre las organizaciones y sus audiencias.

DEPARTAMENTO DE SOPORTE:

Este departamento se encarga de administrar sus divisiones como el sector de soporte interno y externo.

SECTOR DE SOPORTE EXTERNO:

Se encarga de dar soporte a terceros que hayan comprado un software producido por nuestra empresa.

SECTOR DE SOPORTE INTERNO:

Se encarga de dar soporte informático dentro de la empresa.



CARGOS

Nombre del cargo: Programador

Supervisa a: Nadie

Depende de: Departamento de Desarrollo

Objeto del cargo: Es el encargado de la parte del desarrollo, programación, implantación, formación y mantenimiento del software a sus clientes o a la propia empresa.

Tareas:

- Desarrolla
- Ofrece mantenimiento del software
- Programa el software
- Implementa el software
- Optimiza el Software

Requisitos:

Intelectuales: Licenciado en Informática - 2 años de experiencia - Conocimiento en lenguajes de programación como: VB.NET, Java, JavaScript, C++ - Saber Inglés técnico.

Físicos: Hasta 50 años - Carné de salud vigente.

Responsabilidades: Se encarga de desarrollar nuevos programas y/o aplicaciones a través de programación en bloques de código.

Condiciones de trabajo: Lunes a viernes 8 horas diarias.

Sábado 4 horas diarias.

Ubicación: A elección, puede ser trabajar desde la casa o en la empresa.

Servicio de internet: Solo se brinda en la empresa.

Remuneración nominal: UYU\$35.000 x 44 horas semanales.



Nombre del cargo: Analista.

Supervisa a: Nadie.

Depende de: Departamento de desarrollo.

Objeto del cargo: Analiza e intenta mediante herramientas calcular el costo-beneficio del proyecto, cuánto tiempo tardaríamos en hacerlo, etc.

Tareas:

- Realizar análisis costo-beneficio del proyecto.
- Graficar GANTT Y PERT (Tiempo que llevará hacer el proyecto).
- Realizar diagramas (Nassi-Schneiderman, DFD, Warnier-Orr, UML, CDU).
- Analizar los riesgos.
- Realizar métricas.

Requisitos:

Intelectuales: Título de Ingeniería Informática - Inglés Técnico.

Físicos: Tener hasta 50 años - Carné de salud.

Responsabilidades: Comprender las necesidades de los usuarios y a partir de las mismas, definir las especificaciones técnicas de los nuevos programas o aplicaciones.

Condiciones de trabajo: Lunes a viernes 8 horas diarias.

Sábado 4 horas diarias.

Ubicación: En la empresa.

Servicio de internet: Solo se brinda en la empresa.

Remuneración nominal: UYU\$40.000 x 44 horas semanales.



Nombre del cargo: Técnico de soporte.

Supervisa a: Nadie.

Depende de: Soporte.

Objeto del cargo: Se encarga de diagnosticar y solucionar problemas o errores de hardware o software de la empresa desarrolladora y de sus clientes.

Tareas:

- Realiza respaldos de los servidores y base de datos.
- Solucionar problemas o errores existentes en el hardware o software que tengan los puestos de trabajo, la red o los servidores.
- Instruir a los usuarios del programa.
- Solucionar dudas de los clientes de la empresa.
- Realizar actualizaciones de software y hardware a los equipos del cliente.

Requisitos:

Intelectuales: Bachiller de informática - Conocimientos básicos VB.net - Conocimientos básicos de consola UNIX.

Físicos: Tener hasta 50 años - Carné de salud.

Responsabilidades: Brindar soporte a través de soluciones tecnológicas para que los clientes puedan hacer un correcto uso de sus productos o servicios, además tiene la tarea de capacitar a los usuarios del sistema.

Condiciones de trabajo: Lunes a viernes 8 horas diarias.

Sábado 4 horas diarias.

Ubicación: En la empresa.

Servicio de internet: Solo se brinda en la empresa.

Remuneración nominal: UYU\$30.000 x 44 horas semanales

Nombre del cargo: Gerente de proyecto.

Supervisa a: A.D.A, Finanzas, Desarrollo, Recursos Humanos, Marketing, Soporte.

Depende de: Nadie.

Objeto del cargo: El gerente de proyecto es el encargado de coordinar, asignar tareas y realizar la toma de decisiones que afecten directamente al proyecto. Su función principal es lograr que el equipo de trabajo logre con creces alcanzar las metas y/o objetivos planteados.

Tareas:

- Definir objetivos.
- Planificar.
- Gestionar los plazos para lograr terminar el proyecto a tiempo.
- Administrar de manera eficaz y eficiente los recursos.
- Orientar y motivar al equipo de trabajo.

Requisitos:

Intelectuales: Ingeniería en Software - 5 años de experiencia laboral en el área de Tecnologías de la información - Capacidad de liderar - Inglés avanzado.

Físicos: Tener hasta 50 años - Carné de salud.

Responsabilidades: Un gerente de proyecto tiene la responsabilidad de definir objetivos que sean claros y alcanzables según las capacidades de la empresa y sus trabajadores, a través de estrategias empresariales, administrar correctamente los recursos de la empresa y a su vez capacitar, supervisar, liderar y motivar al equipo de proyecto.

Condiciones de trabajo: Lunes a viernes 8 horas diarias.

Sábado 4 horas diarias.

Ubicación: En la empresa.

Servicio de internet: Solo se brinda en la empresa.

Remuneración nominal: UYU\$45.000 x 44 horas semanales

Nombre del cargo: Tester

Supervisa a: Nadie.

Depende de: Programación.

Objeto del cargo: Planificar y llevar a cabo pruebas para corroborar y detectar errores antes de entregar el producto final y comunicarles los mismos a los programadores.

Tareas:

Identificar los riesgos de errores tales como:

- Si el software coincide con la especificación original (o qué hay que hacer para que coincida).
- Si cuando un usuario hace clic en un botón o enlace para que suceda algo, por ejemplo, para ir a una nueva pantalla, se abre la pantalla correcta.
- Si el usuario tiene que introducir algunos datos, el software actúa de forma correcta y los datos se almacenan de forma segura.
- Si está diseñado para que varios usuarios puedan abrirlo al mismo tiempo sin que el programa reduzca su velocidad de procesamiento.

Requisitos:

Intelectuales: Terciario no universitario en tecnologías de información - Experiencia laboral de 2 años en el área de Tecnologías de la información - Inglés intermedio.

Físicos: Tener hasta 50 años - Carné de salud.

Responsabilidades: Supervisar sistemas y llevar a cabo pruebas de estrés, rendimiento, escalabilidad y funcionales, luego a partir de estas detectar potenciales fallos o errores.

Condiciones de trabajo: Lunes a viernes 8 horas diarias.

Sábado 4 horas diarias.

Ubicación: A elección, puede ser trabajar desde la casa o la empresa

Servicio de internet: Solo se brinda en la empresa.

Remuneración nominal: UYU\$30.000 x 44 horas semanales.

Plan de inversiones

PLAN INVERSIONES			
Artículo	Cantidad	Precio unitario (\$UYU)	Total (\$UYU)
Escritorio moderno H: 74.5cm A:155cm P:60cm	5	\$ 3.800,00	\$ 19.000,00
Monitor Acer B246 "24 VGA 60 Hz	10	\$ 5.195,00	\$ 51.950,00
Mouse Logitech M90 USB	5	\$ 306,00	\$ 1.530,00
Teclado Xenex Mecánico USB Retroiluminado	5	\$ 1.944,00	\$ 9.720,00
Silla Lumax ZGS-956	5	\$ 4.145,00	\$ 20.725,00
Laptop Lenovo IdeaPad 330-15IKB	1	\$ 29.639,00	\$ 29.639,00
Impresora Canon Pixma G1100	1	\$ 6.722,00	\$ 6.722,00
Cafetera Nescafe Dolce Gusto Mini Me	1	\$ 6.503,00	\$ 6.503,00
Mesa Ping Pong con ruedas	1	\$ 9.990,00	\$ 9.990,00
Proyector Led Gp80	1	\$ 8.512,00	\$ 8.512,00
Rack TV 136cmx69,5cmx37,5cm	1	\$ 2.290,00	\$ 2.290,00
TV Panavox 43UHD5658	1	\$ 14.499,00	\$ 14.499,00

Mousepad Weibo K8 70cmx30cm	5	\$ 379,00	\$ 1.895,00
Microondas Samsung MS23K3513AK	1	\$ 7.421,00	\$ 7.421,00
Frigobar TEM TCRDF	1	\$ 6.690,00	\$ 6.690,00
Puff	2	\$ 1.390,00	\$ 2.780,00

Aire Acondicionado Everest 12.000 BTU	1	\$ 11.990,00	\$ 11.990,00
Playstation 4 1 TB	1	\$ 25.930,00	\$ 25.930,00
Procesador AMD RYZEN 3 3200G	5	\$ 6.373,00	\$ 31.865,00
Placa madre Gigabyte AB350M-DS3H V2	5	\$ 3.492,00	\$ 17.460,00
Disco HDD Seagate Barracuda 1 TB Sata III	5	\$ 2.510,00	\$ 12.550,00
Disco SSD M.2 WD Green 120GB	5	\$ 1.658,00	\$ 8.290,00
Fuente TC ARMY 450w	5	\$ 1.092,00	\$ 5.460,00
Gabinete TC ARMY Contender Z10	5	\$ 2.182,00	\$ 10.910,00
Memoria RAM Team Elite 4GB DDR4 2666Mhz	10	\$ 1.200,00	\$ 12.000,00
Total:			\$ 336.321,00



Recibos de sueldo:

EMPRESA		
Razón social: GSA SRL		
Domicilio: Quijote 2885		
Grupo 22	Subgrupo 19	
B.P.S. N° -	B.S.E -	
R.U.T. / C.I. 212734833894	Nº planilla MTSS -	
EMPLEADO		
Apellidos, nombres: Santiago Miguel Iriart Guibernau		
Cargo Laboral: Programador	Categoría Laboral: Junior	
Fecha de ingreso 20/05/2020	C.I. 55762293	
Hora Jornal	Mensual X	
Liquidación de pago mes Septiembre	de 20 20	
del 1º al 30º		
NOMINAL		
Sueldo		\$ 35.000
Horas extras		"
Horas nocturnas		"
Feriados pagos		"
Aguinaldo		"
Licencia		"
Salario vacacional		"
Indemnización		"
Antigüedad		"
TOTAL GANADO NOMINAL		\$ 35.000
DESCUENTOS		
Aporte jubilatorio		\$ 5.250
Seguro de enfermedad		" 1.050
I.R.P.F.		" 337
F.R.L.		" 35
Adelantos		"
TOTAL DESCUENTOS		\$ 6.672
Declaro que devuelvo los boletos correspondientes a los pasajes realizados en el mes anterior.	<input checked="" type="checkbox"/>	CANTIDAD A COBRAR \$ 28.320
Declaro que devuelvo los aportes correspondientes a los períodos vencidos en el año año de aporte de resarcimiento.	<input type="checkbox"/>	Recibo el importe y una copia de esta liquidación no teniendo nada que reclamar por ningún concepto.

Razón social: GSA SRL
 Domicilio: Quijote 2885
 Grupo: 22 Subgrupo: 19
 B.P.S. N°: B.S.E.
 R.U.T. / C.I.: 212734833894 N° planilla MTSS: -

EMPLEADO

Apellidos, nombres: Agustín Luciano Acosta Bernardo
 Cargo Laboral: Analista Categoría Laboral: Junior
 Fecha de ingreso 20/05/2020 C.I. 52137108
 Hora Jornal Mensual X
 Liquidación de pago mes Septiembre de 20.20
 del 1º al 30º

NOMINAL

Sueldo		\$ 40.000
Horas extras		"
Horas nocturnas		"
Feriados pagos		"
Aguinaldo		"
Licencia		"
Salario vacacional		"
Indemnización		"
Antigüedad		"

TOTAL GANADO NOMINAL \$ 40.000

DESCUENTOS

Aporte jubilatorio	\$ 6.000
Seguro de enfermedad	" 1.200
I.R.P.F.	" 837
F.R.L.	" 40
Adelantos	"

TOTAL DESCUENTOS \$ 8.077

(Según Decreto N° 168/2007 del 22 de marzo de 2007).
 Declaro que los descuentos corresponden a los
 impuestos relativos al trabajador en el mes anterior.



CANTIDAD A COBRAR \$ 31.923

Declaro que no presto los servicios mencionados dentro
 de la presente vereda ni he sido sujeto de aporte de retención.



Recibo el importe y una copia de esta liquidación no
 teniendo nada que reclamar por ningún concepto.



EMPLEADO

Razón social: GSA SRL
 Domicilio: Quijote 2885
 Grupo 22 Subgrupo 19
 B.P.S. N° - B.S.E -
 R.U.T / C.I. 212734833894 N° planilla MTSS -

EMPLEADO

Apellidos, nombres: Rosales Matos, Zendhel
 Cargo Laboral: Tester Categoria Laboral: Junior
 Fecha de ingreso C.I. 6.265.313-4
 Hora Jornal Mensual
 Liquidación de pago mes Septiembre de 2020
 del 1 al 30

NOMINAL

Sueldo		\$ 30.000
Horas extras		"
Horas nocturnas		"
Feriados pagos		"
Aguinaldo		"
Licencia		"
Salario vacacional		"
Indemnización		"
Antigüedad		"

TOTAL GANADO NOMINAL \$ 30.000

DESCUENTOS

Aporte jubilatorio	\$ 4.500
Seguro de enfermedad	" 1.350
I.R.P.F.	" -
F.R.L.	" 30
Adelantos	"

TOTAL DESCUENTOS \$ 5.880

(Según decreto N° 1682/07 del 22 de marzo de 2007).

Deberá hacer descuento los aportes correspondientes a los siguientes seguros al establecido en el monto anterior:

CANTIDAD A COBRAR

\$ 24.120

(Recibo su haber nominal los aportes correspondientes
y no poseo validez en la liquidación de aportes de retención.)

Recibí el informe y una copia de esta liquidación no teniendo nada que reclamar por ningún concepto.

EMPLEADO

Razón social: GSA SRL	Domicilio: Quijote 2885
Grupo: 22	Subgrupo: 19
B.P.S. N° -	B.S.E. -
R.U.T. / C.I. 212734833894	Nº planilla MTSS -

EMPLEADO

Apellidos, nombres: Bruno Daniel Sunes Pianavilla	
Cargo Laboral: Técnico de soporte	Categoría Laboral: Junior
Fecha de ingreso 20/05/2020	C.I. 52582911
Hora Jornal	Mensual X
Liquidación de pago mes Septiembre	de 20 20
del 1º al 30º	

NOMINAL

Sueldo		\$ 30.000
Horas extras		"
Horas nocturnas		"
Feriados pagos		"
Aguinaldo		"
Licencia		"
Salario vacacional		"
Indemnización		"
Antigüedad		"

TOTAL GANADO NOMINAL \$ 30.000

DESCUENTOS

Aporte jubilatorio	\$ 4.500
Seguro de enfermedad	" 900
I.R.P.F.	" -
F.R.L.	" 30
Adelantos	"

TOTAL DESCUENTOS \$ 5.490

(Según decreto N° 168/2007 del 22 de marzo de 2007).
Porcentaje deducido de los sueldos correspondientes a los
servicios realizados al establecer en el mes anterior:



CANTIDAD A COBRAR \$ 24.510

Declaro que he recibido las apudas correspondientes
y que estoy satisfecho con la liquidación de aportes y descuentos.



Recibí el importe y una copia de esta liquidación no
teniendo nada que reclamar por ningún concepto.



EMPLEADO

Razón social: GSA SRL	Domicilio: Quijote 2885
Grupo: 22	Subgrupo: 19
B.P.S. N° -	B.S.E: -
R.U.T./C.I. 212734833894	Nº planilla MTSS -

EMPLEADO

Apellidos, nombres: Gonzalo Lisandro Gularde Artola	
Cargo Laboral: Gerente de proyecto	Categoría Laboral: Senior
Fecha de ingreso 20/05/2020	C.I. 54200707
Hora Jornal	Mensual X
Liquidación de pago mes Septiembre	de 20 20
del 1º al 30º	

NOMINAL

Sueldo		\$ 45.000
Horas extras		"
Horas nocturnas		"
Feriados pagos		"
Aguinaldo		"
Licencia		"
Salario vacacional		"
Indemnización		"
Antigüedad		"

TOTAL GANADO NOMINAL \$ 45.000

DESCUENTOS

Aporte jubilatorio	\$ 6.750
Seguro de enfermedad	" 1.350
I.R.P.F.	" 1.337
F.R.L.	" 45
Adelantos	"

TOTAL DESCUENTOS \$ 9.482

(Según decreto N° 106/2017 del 22 de marzo de 2017). CANTIDAD A COBRAR \$ 35.548

Declaro que declaro que no poseo conceptos debidos
y/o adeudos pendientes al Bettaplat en el mes actual.

Recibí el importe y una copia de esta liquidación no
teniendo nada que reclamar por ningún concepto.

Características del producto:

Además de la implementación de un software se implementa la infraestructura necesaria para utilizarlo y permitir otras actividades de la institución médica. Se instalarán equipos de trabajo, access points para los pacientes y sus acompañantes, un servidor que aloja la base de datos, otro para los respaldos, y uno más para la página web y el correo electrónico de la institución. Para mantener la seguridad de la red y equipos en la institución se implementaron medidas como la instalación de un antivirus para los puestos de trabajo y servidores, la creación de usuarios con solo los permisos necesarios para realizar su trabajo, la configuración del firewall de los servidores, la creación de redes virtuales para impedir el acceso o egreso de partes no deseadas de la red, se limitaron la cantidad de equipos que es posible conectar a una red, y se implementó un router firewall el cual puede permitir o denegar el acceso a la red de la institución médica.

Estrategias para determinar el precio del producto:

Se determinará el precio del producto a través de las métricas por punto de función realizadas. Teniendo en cuenta la relación costo-beneficio, el tiempo transcurrido hasta la finalización del proyecto

Distribución producto:

Nivel de distribución 0 (Marketing directo): Nuestra distribución será mediante la comunicación directa con el consumidor, mediante llamadas telefónicas, correos electrónicos y demostraciones técnicas sin intervención intermediaria.

Estrategias para la distribución del producto:

Se implementará la distribución exclusiva, ya que el único distribuidor será la misma empresa, y así controlar la distribución del producto solo a zonas y clientes, a los que este se adecue.

Promoción:

Público objetivo: Instituciones prestadoras de salud del Uruguay y la región.

Herramientas a utilizar:

Serán las ventas personales debido a que primero nos comunicaremos con los consumidores a través de los medios anteriormente nombrados, realizaremos una reunión para presentar el producto y luego llevaremos a cabo la venta del mismo.

Plan Financiero

INGRESOS (\$USD)	1	2	3	4	5	6	7	8	9	10
Desarrollo de Software	\$ 10.000,00	\$ 0,00	\$ 10.000,00	\$ 0,00	\$ 35.000,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00
Mantenimiento de equipos	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00
Mantenimiento de servidores	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00
Diseño e instalación de Cableado Estructurado	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00
TOTAL MENSUAL	\$ 13.500,00	\$ 8.500,00	\$ 13.500,00	\$ 8.500,00	\$ 38.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00
EGRESOS (\$USD)	1	2	3	4	5	6	7	8	9	10
Alquiler	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00
Internet	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00
Luz	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00
Agua	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00
Teléfono	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00
Gastos comunes	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00
Impuesto puerta	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00
Comida	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00
Transporte	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00
Papelería	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00
Publicidad	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00
TOTAL MENSUAL	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00
(\$USD)	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8	MES 9	MES 10
SALDO INICIAL	\$ 1.580,00	\$ 14.696,00	\$ 22.812,00	\$ 35.928,00	\$ 44.044,00	\$ 82.160,00	\$ 90.276,00	\$ 93.392,00	\$ 101.508,00	\$ 104.624,00
INGRESOS	\$ 13.500,00	\$ 8.500,00	\$ 13.500,00	\$ 8.500,00	\$ 38.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00
EGRESOS	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00
SALDO FINAL	\$ 14.696,00	\$ 22.812,00	\$ 35.928,00	\$ 44.044,00	\$ 82.160,00	\$ 90.276,00	\$ 93.392,00	\$ 101.508,00	\$ 104.624,00	\$ 112.740,00

11	12	13	14	15	16	17	18	19	20	21	22	23	24	TOTAL
\$ 0,00	\$ 0,00	\$ 55.000,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 55.000,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 165.000,00
\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 1.000,00	\$ 24.000,00
\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 2.500,00	\$ 60.000,00
\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 0,00	\$ 5.000,00	\$ 60.000,00
\$ 3.500,00	\$ 8.500,00	\$ 58.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 8.500,00	\$ 309.000,00
11	12	13	14	15	16	17	18	19	20	21	22	23	24	TOTAL
\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00
\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 78,00	\$ 1.872,00
\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 84,00	\$ 2.016,00
\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 12,00	\$ 288,00
\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 29,00	\$ 696,00
\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00	\$ 0,00
\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 19,00	\$ 456,00
\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 24,00	\$ 576,00
\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 10,00	\$ 240,00
\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 68,00	\$ 1.632,00
\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 60,00	\$ 1.440,00
\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 9.216,00
MES 11	MES 12	MES 13	MES 14	MES 15	MES 16	MES 17	MES 18	MES 19	MES 20	MES 21	MES 22	MES 23	MES 24	
\$ 112.740,00	\$ 115.856,00	\$ 123.972,00	\$ 181.588,00	\$ 189.704,00	\$ 192.820,00	\$ 200.936,00	\$ 204.052,00	\$ 212.168,00	\$ 215.284,00	\$ 223.400,00	\$ 281.516,00	\$ 289.632,00	\$ 292.748,00	
\$ 3.500,00	\$ 8.500,00	\$ 58.000,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	\$ 8.500,00	\$ 8.500,00	\$ 3.500,00	\$ 8.500,00	
\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	\$ 384,00	
\$ 115.856,00	\$ 123.972,00	\$ 181.588,00	\$ 189.704,00	\$ 192.820,00	\$ 200.936,00	\$ 204.052,00	\$ 212.168,00	\$ 215.284,00	\$ 223.400,00	\$ 281.516,00	\$ 289.632,00	\$ 292.748,00	\$ 300.864,00	

- GSA S.R.L -

REFERENCIAS

SALDO INICIAL = INV.INICIAL - COSTO INVERSIÓN

SALDO FINAL = SALDO INICIAL + INGRESOS - EGRESOS

PUNTO EQUILIBRIO (PE)

$$\text{PUNTO EQUILIBRIO (PE)} = \frac{\text{COSTOS FIJOS TOTALES} (\$1.920,00)}{\text{PRECIO VENTA} (\$55.000,00) - \text{COSTO VARIABLE UNITARIO} (\$50.000,00)} = 0,4$$

VALOR ACTUAL NETO (VAN)

$$\text{VALOR ACTUAL NETO (VAN)} = -\text{INVERSIÓN INICIAL} + \frac{(\$127.000,00 - \$4.608,00)}{(1 + 0,3)} + \frac{(\$182.000,00 - \$4.608,00)}{(1 + 0,3)} + \text{VALOR RESIDUAL DE LOS ACTIVOS} (\$55.160,77)$$



Plan de Comercialización

BASIC	STANDARD	PROFESSIONAL
PRECIO BASE		
Soporte presencial de Lunes a Viernes entre 08:00 y 12:00 horas durante 3 años.	Soporte presencial de Luens a Viernes entre 08:00 y 16:00 durante 5 años.	Soporte presencial de Lunes a Viernes entre 8:00 y 20:00 durante 10 años.
Soporte remoto de Lunes a Viernes entre 08:00 y 20:00 horas durante 3 años.	Soporte remoto las 24 horas del día los 7 días de la semana durante 5 años.	Soporte remoto las 24 horas del día los 7 días de la semana durante 10 años.
Capacitación al personal para 30 empleados de la institución.	Capacitación al personal para 70 empleados de la institución.	Capacitación al personal para 150 empleados de la institución.
Licencias de uso para 40 terminales.	Licencias de uso para 60 terminales.	Licencias de uso para 120 terminales.
Actualizacion de Software gratuita cada 3 meses.	Actualizacion de Software gratuita cada 3 meses.	Actualizacion de Software gratuita cada 3 meses.
USD\$ 55,000.00	USD\$ 60,000.00	USD\$ 65,000.00
* Para un plan personalizado o más información contactar a través de: gsasrlcorp@gmail.com * NOTA: Los precios no incluyen el equipamiento informático ni la instalación de los mismos, de ser así, el precio varía.		

Requisitos de la asignatura Sociología

Pautas de la entrevista a Médico

Datos identificatorios:

Nombre: (o anónimo)

Profesión: (ocupación)

Sexo:

Edad:

Institución donde trabaja: pública / privada/ambas

Años de experiencia profesional:

Sobre diagnóstico

¿Cuáles son las **preguntas** que más utiliza usted para diagnosticar enfermedades?

¿Cuáles son los **datos** más importantes a la hora de detectar una enfermedad? ¿Cuáles de ellos son imprescindibles registrar y conservar?

¿Cuál es la mejor forma de realizar un **diagnóstico**?

¿En qué medida influye la **información brindada por el paciente** a la hora de diagnosticar una enfermedad?

Sobre las enfermedades

¿Cuáles son las **enfermedades** o causales más frecuentes por las cuales los pacientes van a consultar?

¿Qué tipo de enfermedades son en las que más **fallan los diagnósticos**?

¿En qué se basan los médicos para solicitar la realización de **análisis clínicos** para un paciente? ¿Qué tan frecuente es?

¿Qué **opinión** tiene sobre la posibilidad que una aplicación pueda ayudar al médico a diagnosticar a un paciente?

Sobre software y aplicación

¿Está de acuerdo con la **implementación de tecnologías informáticas** que le permitan mejorar su diagnóstico médico?

¿Cree que un **software** que es capaz de realizar diagnósticos sería de utilidad para su profesión? ¿Y de confianza?

¿Qué **ventajas y/o desventajas** traería la implementación de un software a la hora de diagnosticar enfermedades?

¿Le parece más eficiente disponer de los **datos de los pacientes** en formato físico o digital? ¿Por qué?

¿Tiene conocimiento de algún software en Uruguay que cumpla estas funciones?

¿Qué **funciones** ve imprescindibles en una aplicación en la cual se guardarán los historiales clínicos?

Transcripción entrevista realizada a Médico

Entrevistador: Juan Pablo Gutierrez (**JP**)

Entrevistado/a: Lucia Margarita (**LM**)

INICIO DE LA ENTREVISTA

JP: ¿Hola?

LM: ¿Cómo andas Juan Pablo?

JP: Bien de bien, ¿y vos?

LM: Bien, bien.

LM: Estamos de viaje hacia un centro de salud de La Costa, por eso estoy con el cinto.

JP: Dale dale no pasa nada.

JP: Bueno, empezando con la entrevista queríamos comentarte, somos estudiantes de Sexto año de Bachiller Informático, de la UTU...

LM: Ahí está, sí.

JP: Y estamos haciendo un proyecto con el que queremos anotar bitácoras médicas y ayudar a los médicos a tener un acceso mucho más sencillo a las mismas.

LM: Ah mirá que bueno, buenísimo.

JP: Ta, quería saber si... primero que nada, si vos trabajabas en una médica pública o privada. Más o menos para saber cómo es.

LM: Si, en pública, en lo que se llama ASSE, Administración de Servicios de Salud del Estado.

JP: ¿Durante cuántos años has trabajado ahí?

LM: Eh... desde 2007, eh... 13 años.

JP: Bastante.

JP: Bueno, queríamos consultarte, primero que nada, ¿cuáles son las preguntas más usuales que solés hacer para diagnosticar cuando llega un paciente?

LM: Ahí está, como no. Bueno, en realidad le preguntas en primer lugar por qué venía, ¿no? En que lo podés ayudar, por qué que venía hoy.

JP: ¿Y hay alguna pregunta que suelas hacer más usual después de que te comenta por qué viene, no sé, si tiene algún dolor, fiebre...?

LM: Ahí está, bueno, por ejemplo, si es por un dolor tenes todo una... una lista de preguntas que se hacen sobre el dolor que les llamamos el ATILIEF porque es, A de aparición, cuando apareció, el tipo de dolor, la localización, la intensidad, la evolución, es decir, de cuando apareció hasta ahora como ha sido, si ha ido aumentando, si ha ido disminuyendo o si está igual, la F que es de, viste que te dije ATILIEF, la F es de fenómenos acompañantes, es decir, si el dolor se acompaña de por ejemplo, si se hincha el lugar donde le duele, si se acompaña de qué está rojo o con calor en el lugar, en la zona, entonces siempre preguntas todas esas cosas que te dije, la A de aparición, tipo, intensidad, localización, evolución y fenómenos acompañantes, por ejemplo en el caso de dolor. En la fiebre también, bueno, cuando apareció, hasta que temperatura, si se la controló, si baja cuando toma un medicamento o si no...

JP: Son varias preguntas.

LM: *Risa*.

JP: Con eso nos ayudaste un montón.

LM: *Risa*. Ya tienen para entretenerte.

JP: No no, es que es buenísimo porque con eso nos ayudas un montón, con las preguntas que tenemos que hacer. Ya que justamente eso es una parte importante que tenemos.

LM: Ahí está, bueno, me alegro.

JP: ¿En qué medida suele ayudar la información que te da el paciente a la hora de hacer un diagnóstico?

LM: ¿Cómo?

JP: ¿Qué tanto suele ayudar la información que te dan los pacientes cuando haces el diagnóstico?

LM: Todo, todo porque todo lo que cuente la persona es lo que te va a orientar hacia el diagnóstico, eso, lo que cuente y después lo que vos encuentres en el examen físico. Es decir, siguiendo con el ejemplo del dolor, suponete que el dolor sea abdominal, le preguntaste todo eso que dijimos, el ATILIEF del dolor, y después vas a examinar el abdomen. Y ahí, con lo que él contó más lo que examinas, generalmente te orientas hacia un diagnóstico, y si no, no estás seguro, mandás a hacer exámenes, yo que se, puede ser una ecografía en el caso del abdomen u otro examen dependiendo de lo que te haya contado.

JP: Bien, genial.

JP: Después, ¿cuáles son las enfermedades más usuales por lo que los pacientes van? Lo que más se les... {ininteligible 0:04:25}

LM: Eh... bueno, lo más usual, porque se repite mucho, porque tienen que ir cada pocos meses para continuar su tratamiento, son las, eh... las enfermedades crónicas, por ejemplo, las diabetes, la hipertensión, artrosis, eh... entonces todo eso, digamos que, aunque no es un... eh... que la persona no esté con dolor o con un síntoma por eso, cada poco tiempo tiene que hacerse controles entonces eso hace que sume mucho en las consultas. Es decir, por ejemplo, si vos un día tenes diez pacientes y capaz que de esos diez, cinco vienen por ese tipo de enfermedades, por artrosis, por hipertensión, por diabetes, colesterol alto... y esos pacientes que fueron hoy, capaz que en tres meses tienen que volver porque se hicieron un examen y tienen que mostrarte el resultado, porque se les acabó el medicamento porque generalmente se los damos por tres meses... y así, entonces ese tipo de consultas de enfermedades en la población nuestra del Uruguay que es bastante envejecida en comparación a otros países, es como muy frecuente, lo que llamamos enfermedades crónicas no transmisibles, la obesidad, la hipertensión, la diabetes, la artrosis, los problemas respiratorios, colesterol... y después tipo, en el caso de las mujeres, bueno, yo que sé, problemas de control del embarazo, por la anticoncepción, que van también, para hacerse los controles por Papanicolaou, entonces, como que a eso se le suman cuestiones preventivas, propias del género femenino que también son frecuentes, por ejemplo, las mujeres que usan algún tipo de anticoncepción ya sea, bueno, pastillas, van cada tanto a repetir a que le volvamos a dar la receta para las pastillas, las que usan otros métodos como que se colocan y después en

realidad {ininteligible 0:06:49} un control más esporádico o menos frecuentes, como por ejemplo, el DIU o los implantes subdérmicos van menos seguido...

LM: *Conversación con otra persona*.

LM: Perdón Juan Pablo que acaba de llegar mi colega.

JP: No pasa nada.

LM: Seguime preguntando.

JP: Dale.

JP: Eh... ¿cuáles son las enfermedades... eh... no... si, ¿cuáles son los diagnósticos que más suelen fallar?

LM: ¿Diagnóstico como...?, ¿qué más...?

JP: Suelen fallar, o sea, que enfermedades generan diagnósticos que fallan.

LM: ¿En qué más nos equivocamos? Ah... que pregunta.

LM: Eh... *bocina*. ¿Más difícil de diagnosticar quizás?

JP: O sea, entre lo que dice el paciente y el resultado.

LM: Y capaz que los... ¿eh?...

JP: En lo que dice el paciente y el resultado.

LM: ¿Cómo, cómo?

JP: Como que haya un fallo ahí.

LM: Y a veces, claro, a veces hay como síntomas muy específicos, como por ejemplo, el dolor de cabeza, los dolores del cuerpo, este... o dolor al abdomen, que muchas veces corresponden a problemas no de carácter biológico digamos, sino que a situaciones de tensión, de estrés, emocionales. Por ejemplo, personas que consultan mucho... que tienen dolor de cabeza seguido y que en realidad puede ser que tengan, no sé, problemas familiares, problemas económicos, problemas laborales, que hacen que estén tensas y que por ejemplo se les produzca un dolor de cabeza. Entonces a veces uno trata, ante un síntoma uno trata de buscar las causas del cuerpo como podría ser, no sé, en el dolor de cabeza que tenga algo más grave como podría ser, un caso extremo, un tumor adentro de la cabeza y... este... y en realidad muchas veces esos síntomas que son como inespecíficos obedecen a problemas más del orden psicológico o emocionales que... que a otra cosa. El dolor abdominal también, los dolores en general del cuerpo, a veces que dicen, "me duele todo"...

JP: Esa como que no ayuda mucho a veces.

LM: Claro, pero bueno, es en realidad un sentimiento real, a la persona le duele todo, le duele la cabeza o le duele... o tiene... la comida le cae mal siempre... este... y en realidad... o también por ejemplo es el estilo de vida, yo que sé, que siempre pasa muchas horas sin comer, cuando come come mucho {ininteligible 0:09:42} y le cae mal y le duele el abdomen, por ejemplo. Entonces a veces es difícil eh... llegar a ese diagnóstico porque a veces no tenés tiempo de preguntar todo... como es su rutina cotidiana y en realidad estás buscando algo... una causa dentro del órgano y en realidad la causa es que la persona se levanta, y toma mate y come a las... pasa todo el día trabajando y come a las cinco de la tarde y a las cinco de la tarde cuando come, come todo junto todo apurado y... muerto de hambre y le cae mal ponele, ¿no? Despues queda como, con el abdomen distendido y/o dolorido y qué sé yo.

JP: Haciendo una pregunta similar, eh... cuál... ¿en qué casos o en qué te basas para mandar a hacer un diagnóstico?... análisis, un análisis perdón.

LM: Sí bueno, eh... en lo que justamente... en lo que te diga la persona... estoy tratando de mandar un mensaje mientras hablo contigo y no sé cómo {ininteligible 0:10:53} pará un poquito... {ininteligible 0:10:56} como puede ser un examen de sangre, una ecografía cuando este... lo que cuenta la persona y lo que encontraste en el examen físico no es concluyente. Entonces por ejemplo, vamos a suponer. La persona... no sé... seguimos con el dolor abdominal, este... cuenta todo un sufrimiento como si fuera de la comida como de mala digestión, entonces vos le mandas un tratamiento en las comidas, en la forma de comer, este... en las cantidades, en la calidad de la comida, en la forma de cocinarla, eh... y en sus rutinas... en su hábito de vida y un medicamento; y si al mes con eso no mejora, sigue con los problemas... ahí le mandas un estudio, por ejemplo en el caso que estoy contando, un estudio que se llama Fibroscopia, que es que, te miran con una camarita dentro del esófago y del estómago {ininteligible 0:12:02} o sea vos, con lo que ya te contó, con lo que lo examinaste, dedujiste un diagnóstico, probas el tratamiento hacia ese diagnóstico y si no mejora con eso que vos hiciste y si la persona lo cumplió, ahí, bueno buscas explicaciones con estudios... con una Fibroscopia por ejemplo.

JP: Bien... ¿qué opinión tenes sobre utilizar una aplicación que te apoye con los diagnósticos? *Voz de fondo*

LM: Eh... en realidad alguna alguien, este... alguna vez... este... alguien alguna vez me contó o he leído sobre eso que tiene un nombre... como ayuda clínica o algo así, que en realidad cuando uno {ininteligible 0:12:53} historia clínica electrónica vas poniendo datos que van como... organizándose de tal forma que orient... que el software digamos te sugiere distintas cosas. Por ejemplo, te sugiere, este... porque ese sistema informático ha acumulado información de muchísimos casos similares y ha visto que todos esos casos similares, tanto por ciento, el noventa por ciento de esos casos similares fue beneficioso pedir tal estudio, entonces... este... a medida que vos vas completando la historia y el sistema ve que se parece a "X" casos, te sugiere mandar ese estudio que funcionó en los otros, o te sugiere preguntar tal cosa o te sugiere mirar tal otra, está bueno.

JP: ¿Vos estarías dispuesta a utilizarlo... si te lo ofrecen?

LM: *Risa*, bueno no sé, sí, creo que sí, creo que sí, a probarlo cómo no.

JP: ¿Sabes si en Uruguay ya llegaron o hay alguno similar?

LM: Eh... yo creo que los Oncólogos, las personas que trabajan los problemas de cáncer, los médicos especialistas tienen un algo así. Tienen una historia clínica electrónica especial que se llama Oncológica que creo que va llenando... va abasteciendo... va alimentando una base de datos que luego los ayuda a tomar decisiones, me parece.

LM: Creo que está en Uruguay... en el mundo está. Creo que acá los Oncólogos también lo tienen.

JP: ¿Qué función ves imprescindible para alguna aplicación que te ayude con {ininteligible 0:14:55} a guardar los historiales clínicos, algo que deberían de tener?

LM: ¿Cómo? ¿Qué función debería tener que perdón?

JP: Eh, una aplicación que te ayude a guardar los historiales clínicos, algo que deberían de tener.

LM: Ay se entrecortó con... *voces de fondo*.

LM: Eh... algo que debería de tener... ay perdón Juan Pablo, se entrecortó cuando dijiste, ¿algo que debería de tener la historia clínica electrónica?

JP: Sí.

LM: ¿La aplicación?

LM: Ah... yo creo que es super importante que se conecte con la farmacia, es decir que vos... este... eh... lo que vos veas en la consulta {ininteligible 0:15:38} tratamiento que ya puedas indicar el tratamiento, el fármaco y que la farmacia con la cédula de la persona, no necesite llevar la receta sino que ya con la cédula de la persona ya se sabe que el médico le indicó tal cosa, por ejemplo.

JP: Dale...

LM: Y bueno...

JP: Te agradecemos muchísimo la entrevista ya que nos ayudaste un montón con el proyecto, no te imaginas cuánto.

LM: *Risa* Bueno me alegro mucho {ininteligible 0:16:08}.

FIN DE LA ENTREVISTA

Análisis entrevista

El objetivo de la entrevista era preguntar qué cosas importantes se tienen que agregar en diagnósticos, que posibles preguntas se pueden hacer en un caso de enfermedad y sobre la utilización de un software como método de ayuda clínica.

El entrevistador, en este caso Juan Pablo, tiene una relación amistosa con la persona entrevistada, lo que hace que la entrevistada al iniciar la entrevista le pregunte, “¿Como andas Juan Pablo?”.

La entrevistada habla y responde correctamente las preguntas, lo que hace que la entrevista mantenga una fluidez no perfecta, pero si aceptable.

Tanto las preguntas como la entrevista fueron preparadas previamente, pero en este caso, la entrevista no fue cometida totalmente de forma formal debido al contexto, el cual era en un salón del ITI y mediante una llamada de Whatsapp mientras que la entrevistada iba en un vehículo. Fuera de eso, la entrevista estuvo correcta y se pudo lograr su cometido.

Para finalizar, el entrevistador pregunta su ultima pregunta, la cual da paso a la culminación de la entrevista.

Luego, Juan Pablo le agradece por la ayuda otorgada mediante las respuestas dadas a lo que la entrevistada reconoce la ayuda.

Durante la entrevista se utiliza un lenguaje informal en varias partes, aunque la mayor cantidad de las preguntas son formales. Esto es motivo de la relación entre el entrevistador y la entrevistada.

Como se utiliza un lenguaje informal y dado el contexto, en partes la entrevistada se sale del tema (habla con otros colegas de trabajo) y a su vez se utilizan varios modismos en lo que dura la entrevista.

Requisitos de la asignatura Proyecto

1) Nombre del grupo, integrantes y sus roles.

Nombre del grupo: GSA

Integrantes: Agustin Luciano Acosta Bernardo, Integrante.

Bruno Daniel Sunes Pianavilla, Integrante.

Santiago Miguel Iriart Guibernau, Sub-Coordinador.

Gonzalo Lisandro Gularte Artola, Coordinador.

2) Organización de la estructura de desarrollo.

Para la estructura de desarrollo se realizará la asignación de roles (Los cuales todos tendrán responsabilidades) así como de normas, entre los roles se encuentran: “Coordinador” el cual tendrá como tarea definir objetivos y ser el encargado de la toma de decisiones para lograr un buen desempeño a la hora de realizar distintas tareas, “Sub-Coordinador” cumple las mismas funciones en conjunto con el coordinador, “Integrante” se entiende como todo otro miembro del grupo que realiza las tareas asignadas.

Además, el coordinador y subcoordinador no solamente realiza la tarea de dirección del equipo sino también la de ejecución.

Se llevará a cabo el estilo de dirección y método “Consultivo” el cual se basa en discutir los diferentes temas planteados con los integrantes del grupo y tomar una decisión en conjunto, es decir, primero se comunica con los integrantes, conoce su opinión y posteriormente se pasa a la decisión.

Cada objetivo o tarea asignada debe de contar con los siguientes criterios:

-Ser específicos y claros. (Que los miembros del grupo los entiendan tanto como los que no pertenecen al mismo).

-Estar enmarcados en el tiempo, es decir, el objetivo deberá de tener una fecha límite para su realización.

También en el equipo se contará con:

-Cohesión: Se buscará que el integrante sienta que pertenece a un grupo y no que solamente está en el mismo, para lograr esto se buscará motivar y dar la oportunidad de participar y aportar en cada tarea.

-Comunicación: Se intentará mantener la máxima comunicación posible en donde el objetivo será compartir diferentes ideas, así como diferentes puntos de vista para mejorar el funcionamiento.

-Interdependencia positiva: Se buscará que cada integrante no sólo sea responsable de su propio aprendizaje sino también del aprendizaje del equipo en general, en donde cada integrante aprenderá del otro y así viceversa.

Sí surgen etapas de conflicto se tratará de sacar los aspectos positivos (Encontrar una solución (Sacar nuevas ideas), generar nuevas normas, posible motivación a un cambio y se conocen los límites, necesidades y derechos individuales) para reducir el impacto de los aspectos negativos.

3) Desarrollar un formato de actas de reuniones formales e informales.

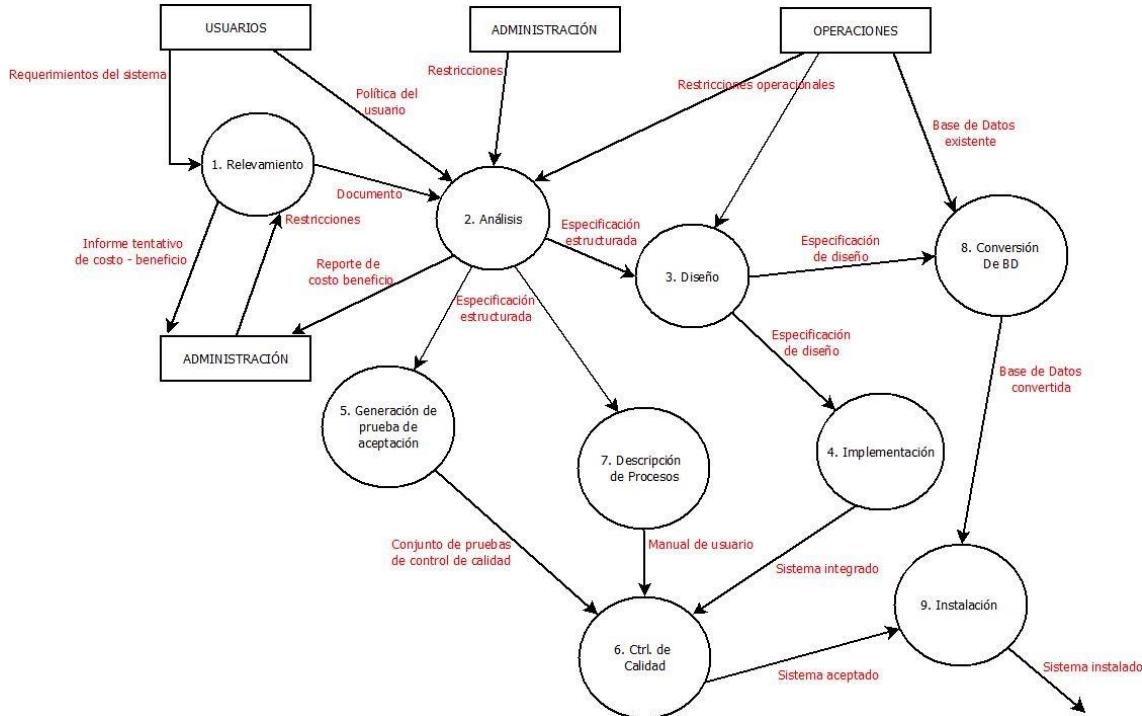
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: X, XX de X de 2020		HORA INICIO: XX:XX	HORA CIERRE: XX:XX	
ASISTENTES				
1	Agustín Acosta - Integrante		Asistencia	
2	Santiago Iriart - Sub-Coordinador		Asistencia	
3	Gonzalo Gularte - Coordinador		Asistencia	
4	Bruno Sunes - Integrante		Asistencia	
TEMAS TRATADOS				
1				
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1				

4) Reglas del grupo.

- Reglas del grupo
 - Se realizarán 2 conferencias formales a la semana para tratar aspectos sobre el proyecto en general, si algún integrante no asiste al 25% de las clases dictadas, se conllevará a la expulsión del grupo.
 - Si se le asigna una tarea a un integrante del grupo, este tendrá un plazo máximo para cumplirla, de lo contrario, si no la cumple recibirá un aviso, si este llega a los 5 avisos se le expulsará del grupo.
 - Se realizarán tareas en conjunto, pero cada uno puede trabajar individualmente en un punto de una entrega y explicarlo formalmente en la conferencia.
 - Se puede tener inasistencias si esta es justificada o avisada anteriormente.

5) Ciclo de vida del proyecto (Formulación e implementación).

Para la primera etapa del proyecto se utilizará el método de Yourdon con el ciclo de vida Estructurado:



En donde en la primera actividad (Relevamiento) se realizará tareas como el estudio de factibilidades, todo empleo de técnicas de Relevamiento (Encuestas, Cuestionarios y entrevistas).

A su vez en la primera actividad también se establecen los objetivos como: Identificar usuarios responsables, identificar deficiencias en el entorno del usuario, definir el alcance inicial del sistema, establecer los objetivos del sistema y preparar un esquema para guiar el resto del proyecto.

Luego para la segunda actividad (Análisis) se hará la parte de documentación, por ejemplo, los DFD, los casos de uso y sus especificaciones, la especificación de requerimientos tanto funcionales como los requerimientos no funcionales.

También se hará un análisis acerca de la rentabilidad de la inversión a la hora de desarrollar el software, cómo serán los costos del mismo y el presupuesto para instalar los equipos y para los recursos humanos que se ocupan del mantenimiento del software.

Para la tercera actividad (Diseño) es en donde se transforma el MER en un diseño de Base de Datos y también se definen las fronteras del sistema y de la interfaz.

Para la cuarta actividad (Implementación) se realizará una programación estructurada en donde se aplicará una implantación descendente.

Para la quinta actividad (Generación de pruebas se aceptación) definir un programa aceptable para el usuario, se comprueba que se cumpla con lo establecido en los casos de uso, puede realizara al mismo tiempo que la implementación y el diseño.

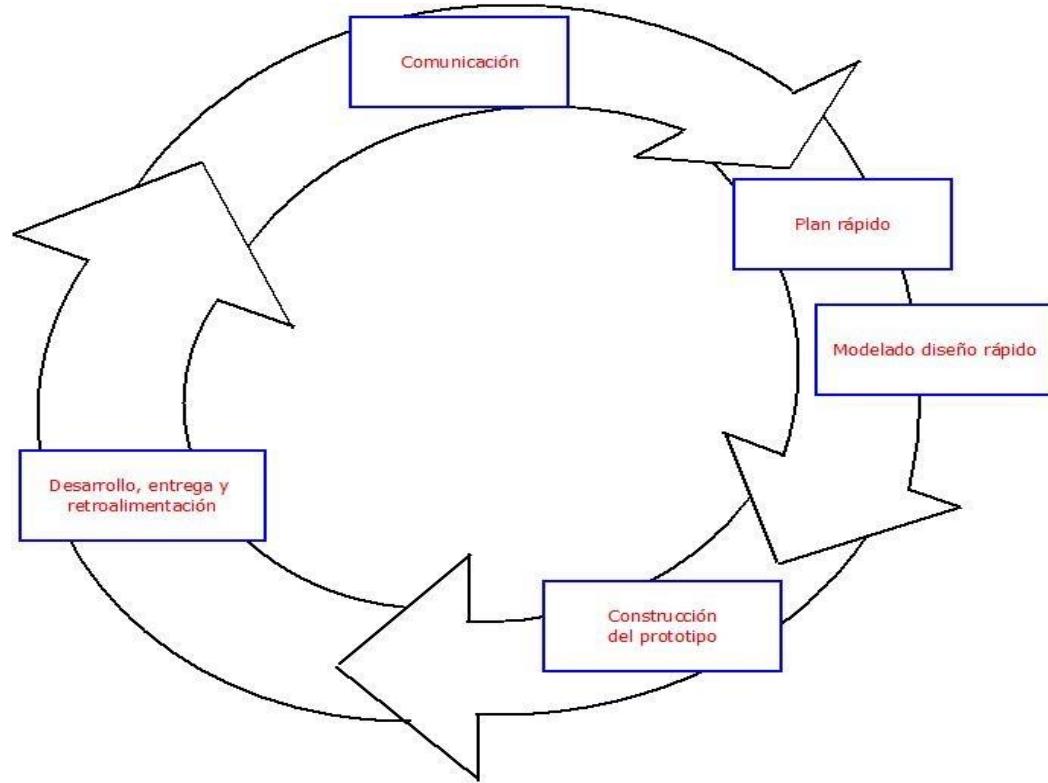
En la sexta actividad (Garantía de calidad) se realizará una prueba final la cual será de aceptación, también se verificará si el sistema posee un nivel de calidad apropiado.

Para la séptima actividad (Descripción de Procedimientos) se generará una descripción acerca del sistema en forma de manual para que los clientes (en este caso) puedan saber cómo va a ser la interacción con la parte automática del sistema.

En la octava actividad (Conversión de base de datos) en caso de que existiese una base de datos actual del usuario se realizaría la migración de la misma, pero este no es el caso.

Finalmente, en la novena actividad (Instalación) se entregará el sistema aceptado junto con la base de datos finalizada y el manual de usuario, a su vez se implementará y se pondrá en marcha.

Por otro lado, para la segunda etapa del proyecto se utilizará un prototipado evolutivo:



Esto debido a que en el caso que exista un plan rápido y surjan nuevos requerimientos se pueda realizar un modelado de diseño rápido y realizar una construcción de una implementación parcial que cubre los requisitos conocidos, para ir aprendiendo a medida que se desarrolla el sistema e incorporarlos al sistema, para luego entregarlo y lograr una retroalimentación mediante la comunicación y en el caso que existan nuevos requerimientos repetir el ciclo.



Actas de reuniones

Ver anexo (ANÁLISIS Y DISEÑO DE APLICACIONES - Anexo IV:
Métricas por punto de función)

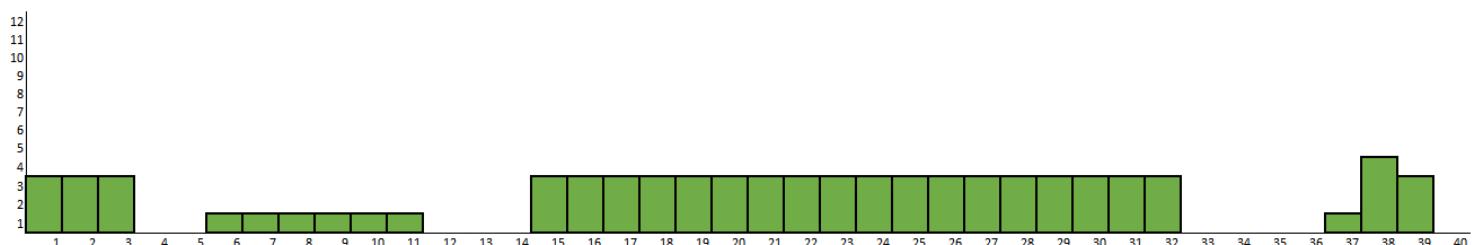
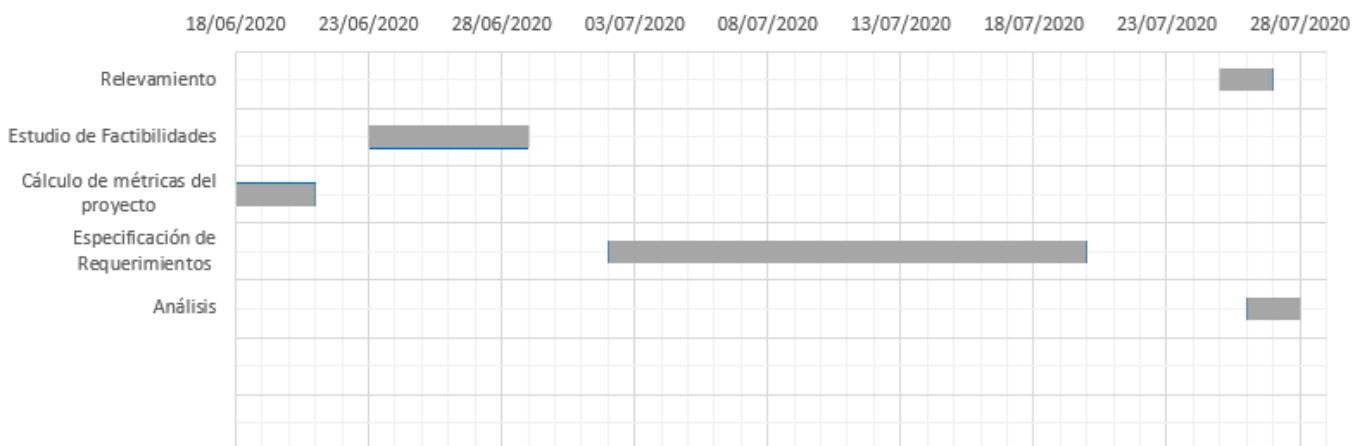
Diagrama GANTT Primer Entrega: Análisis y Diseño de Aplicaciones:

Diagrama Gantt con % completado

Inicio proyecto	18/06/2020
Fin proyecto	28/07/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Relevamiento	25/07/2020	2	28/07/2020	100%	2,00
Estudio de Factibilidades	23/06/2020	6	30/06/2020	100%	6,00
Cálculo de métricas del proyecto	18/06/2020	3	22/06/2020	100%	3,00
Especificación de Requerimientos	02/07/2020	18	27/07/2020	100%	18,00
Análisis	26/07/2020	2	28/07/2020	100%	2,00

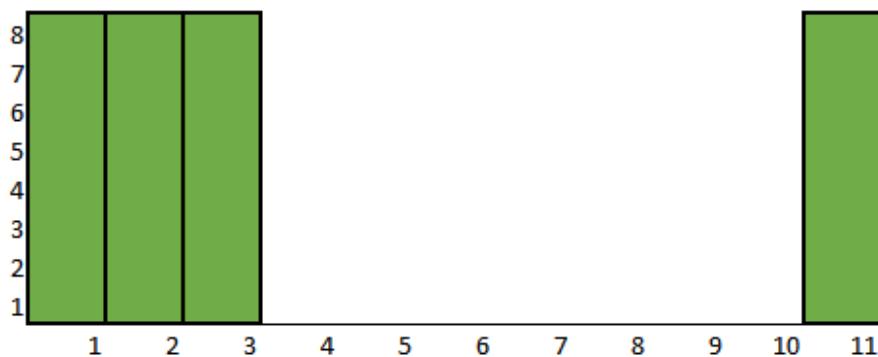
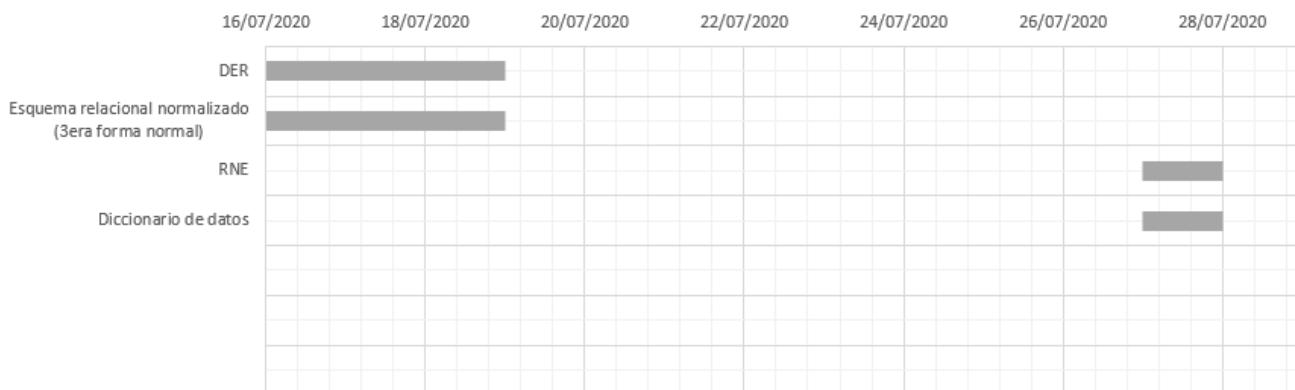
Diagrama Gantt con % completado



Base de Datos:

Diagrama Gantt con % completado					
Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
DER	16/07/2020	3	20/07/2020	100%	3,00
Esquema relacional normalizado (3era forma normal)	16/07/2020	3	20/07/2020	100%	3,00
RNE	27/07/2020	1	27/07/2020	100%	1,00
Diccionario de datos	27/07/2020	1	27/07/2020	100%	1,00

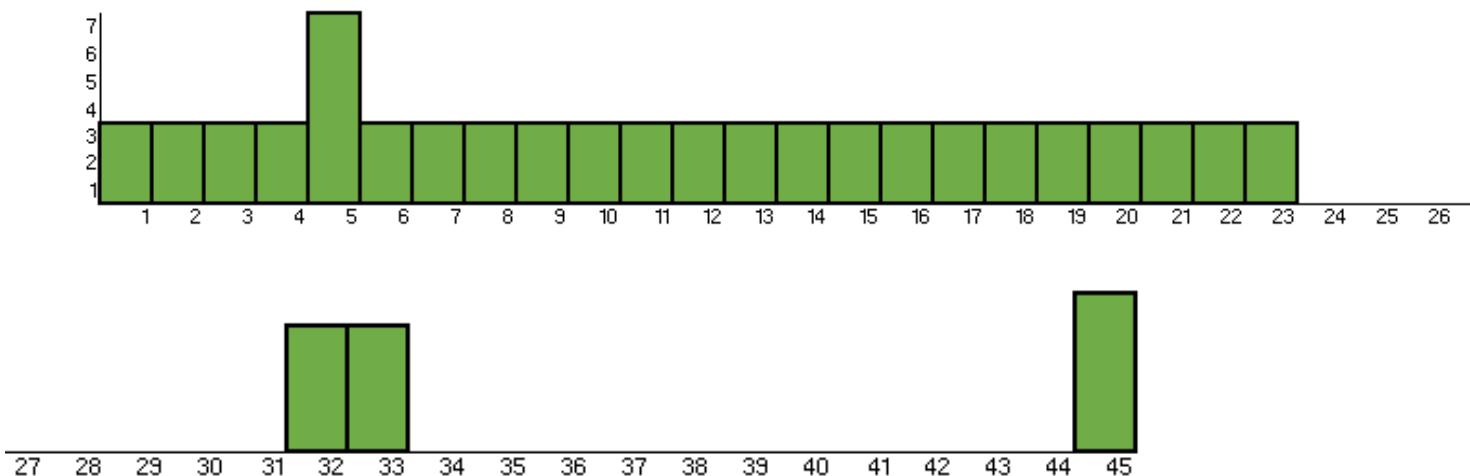
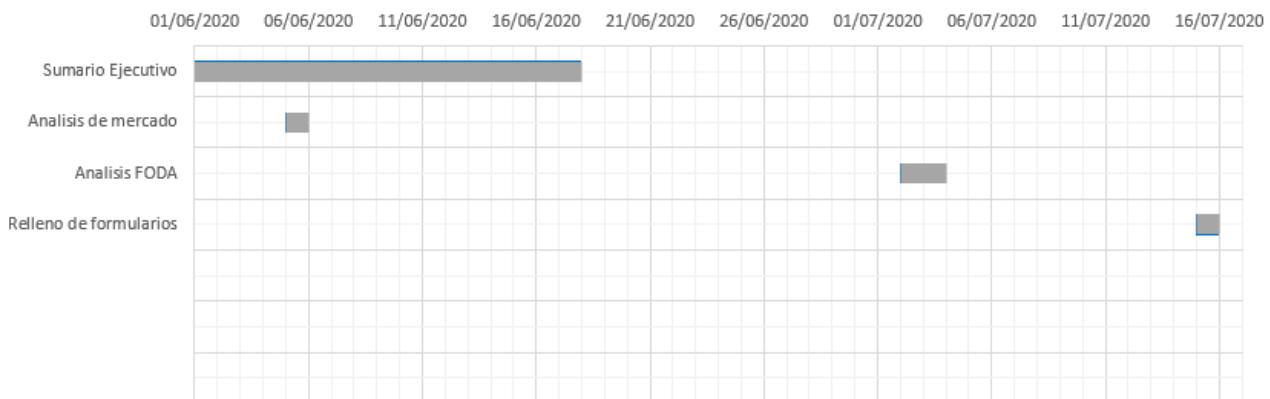
Diagrama Gantt con % completado



Formación Empresarial:

Diagrama Gantt con % completado					
Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Sumario Ejecutivo	01/06/2020	17	23/06/2020	100%	17,00
Analisis de mercado	05/06/2020	1	05/06/2020	100%	1,00
Analisis FODA	02/07/2020	2	03/07/2020	100%	2,00
Relleno de formularios	15/07/2020	1	15/07/2020	100%	1,00

Diagrama Gantt con % completado





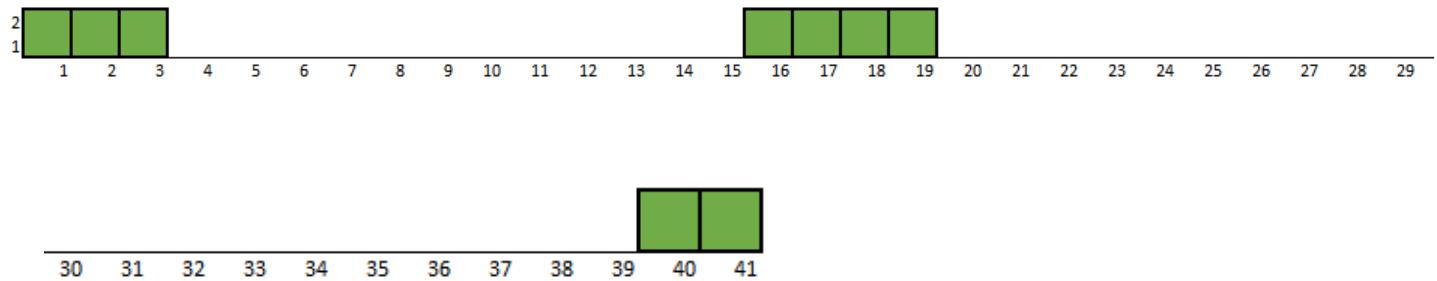
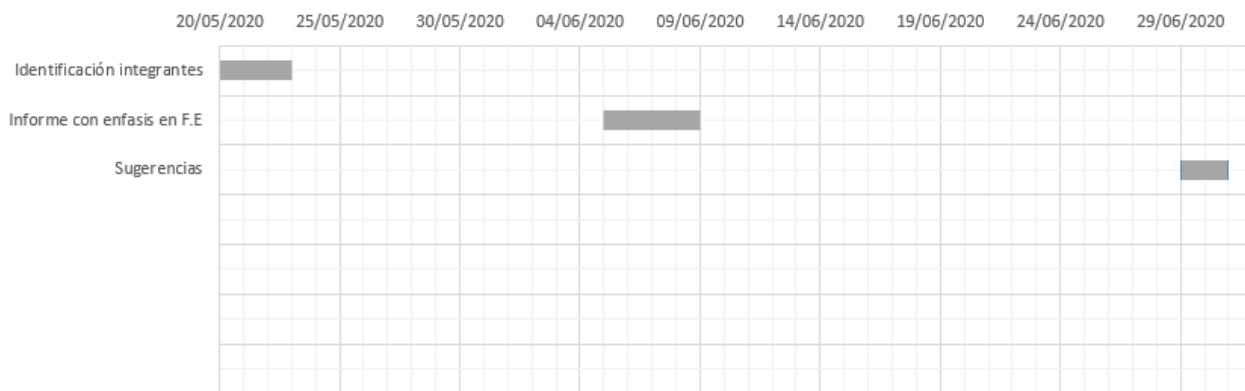
Ingles:

Diagrama Gantt con % completado

Inicio proyecto	20/05/2020
Fin proyecto	30/06/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Identificación integrantes	20/05/2020	3	22/05/2020	100%	3,00
Informe con enfasis en F.E	05/06/2020	4	10/06/2020	100%	4,00
Sugerencias	29/06/2020	2	30/06/2020	100%	2,00

Diagrama Gantt con % completado



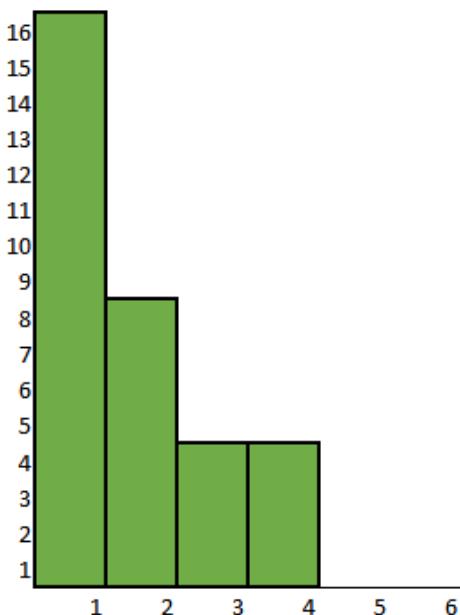
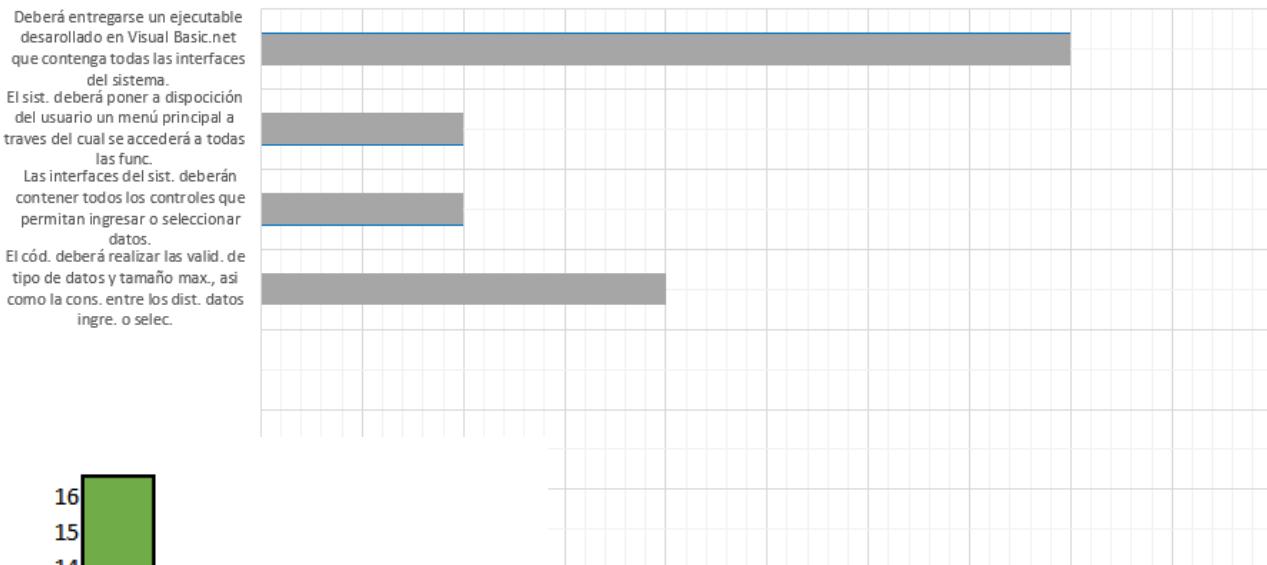
Programación:

Diagrama Gantt con % completado

Inicio proyecto	19/07/2020
Fin proyecto	23/07/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Deberá entregarse un ejecutable desarrollado en Visual Basic.net que contenga todas las interfaces del sistema.	19/07/2020	4	23/07/2020	100%	4,00
El sist. deberá poner a disposición del usuario un menú principal a través del cual se accederá a todas las func.	19/07/2020	1	20/07/2020	100%	1,00
Las interfaces del sist. deberán contener todos los controles que permitan ingresar o seleccionar datos.	19/07/2020	1	20/07/2020	100%	1,00
El cód. deberá realizar las valid. de tipo de datos y tamaño max., así como la cons. entre los dist. datos ingre. o selec.	19/07/2020	2	21/07/2020	100%	2,00

19/07/2020 19/07/2020 20/07/2020 20/07/2020 21/07/2020 21/07/2020 22/07/2020 22/07/2020 23/07/2020 23/07/2020 23/07/2020 24/07/2020





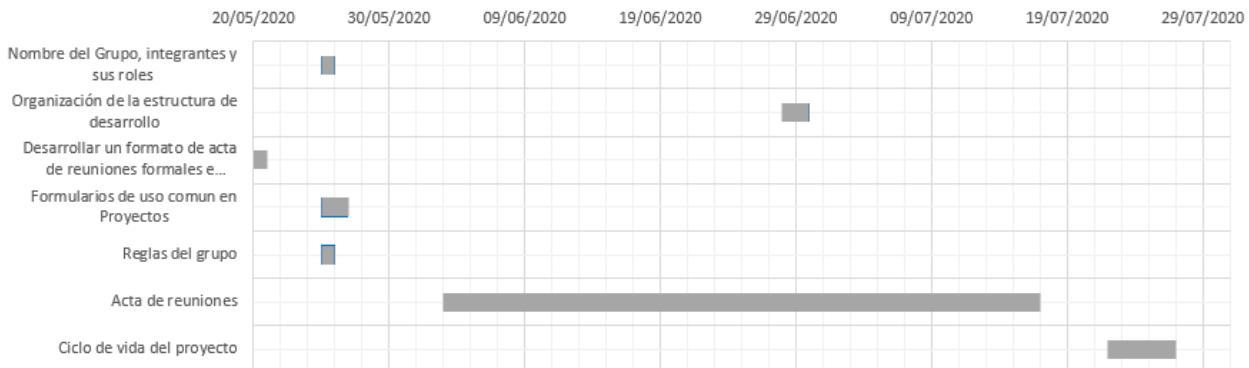
Proyecto:

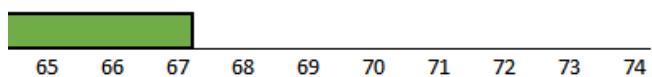
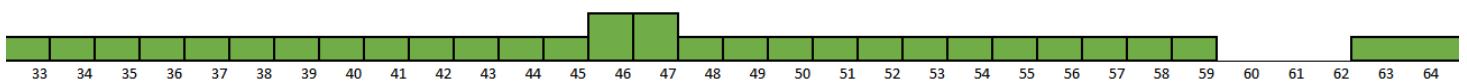
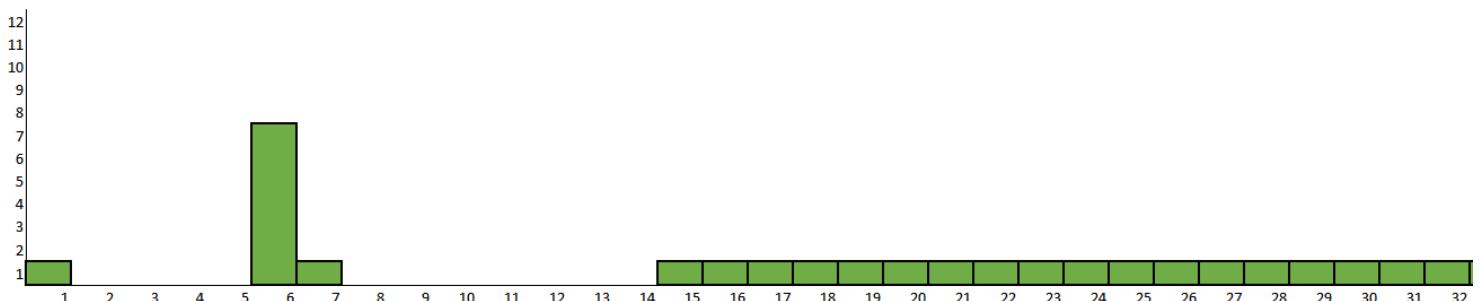
Diagrama Gantt con % completado

Inicio proyecto	20/05/2020
Fin proyecto	03/08/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Nombre del Grupo, integrantes y sus roles	25/05/2020	1	25/05/2020	100%	1,00
Organización de la estructura de desarrollo	28/06/2020	2	30/06/2020	100%	2,00
Desarrollar un formato de acta de reuniones formales e informales	20/05/2020	1	20/05/2020	100%	1,00
Formularios de uso comun en Proyectos	25/05/2020	2	26/05/2020	100%	2,00
Reglas del grupo	25/05/2020	1	25/05/2020	100%	1,00
Acta de reuniones	03/06/2020	44	03/08/2020	100%	44,00
Ciclo de vida del proyecto	22/07/2020	5	28/07/2020	100%	5,00
Documentacion de Inicio y Planificación	26/07/2020	2	28/07/2020	100%	2,00
Diagrama de Gantt	26/07/2020	1	27/07/2020	100%	1,00

Diagrama Gantt con % completado







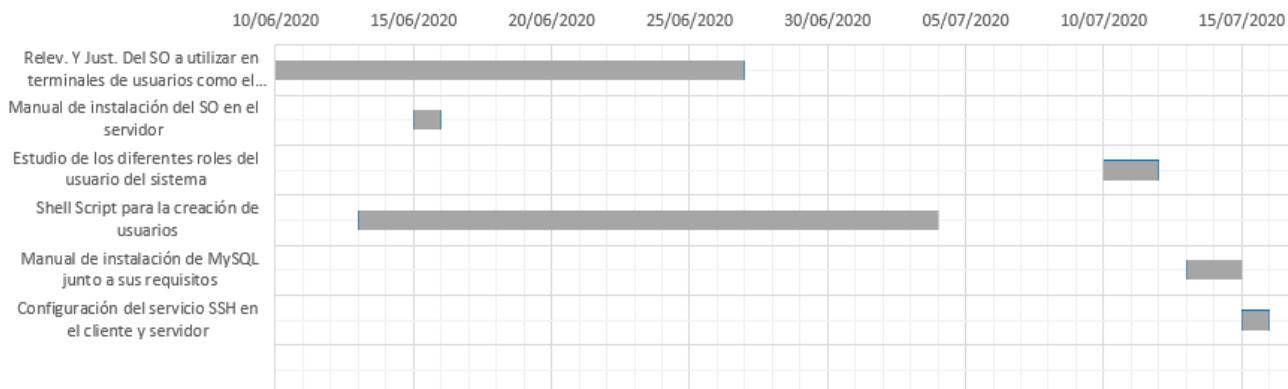
Sistemas Operativos:

Diagrama Gantt con % completado

Inicio proyecto	10/06/2020
Fin proyecto	15/07/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Relev. Y Just. Del SO a utilizar en terminales de usuarios como el servidor	10/06/2020	17	02/07/2020	100%	17,00
Manual de instalación del SO en el servidor	15/06/2020	1	15/06/2020	100%	1,00
Estudio de los diferentes roles del usuario del sistema	10/07/2020	2	13/07/2020	100%	2,00
Shell Script para la creación de usuarios	13/06/2020	21	13/07/2020	100%	21,00
Manual de instalación de MySQL junto a sus requisitos	13/07/2020	2	14/07/2020	100%	2,00
Configuración del servicio SSH en el cliente y servidor	15/07/2020	1	15/07/2020	100%	1,00

Diagrama Gantt con % completado



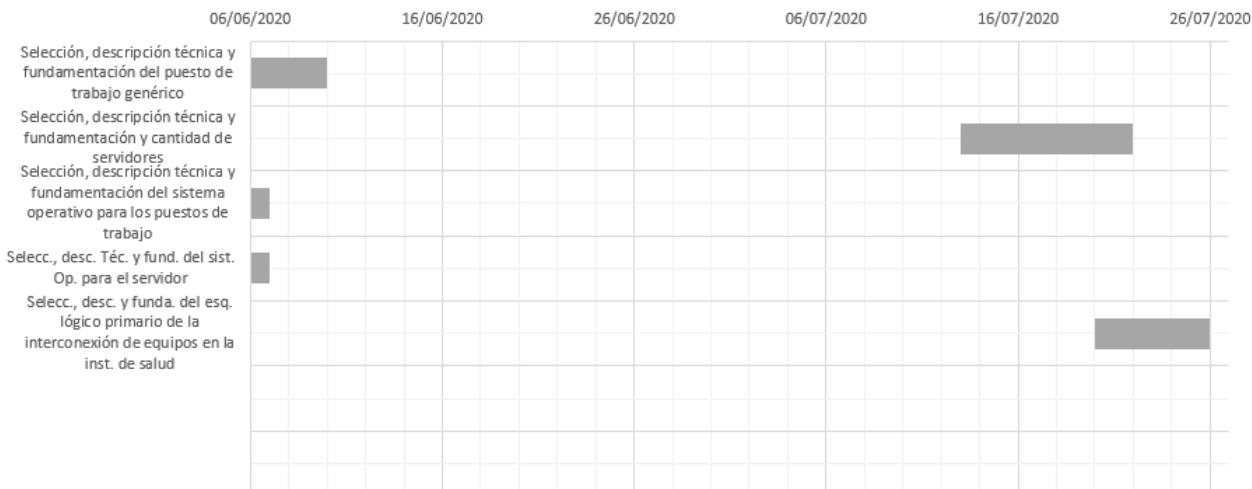
Taller:

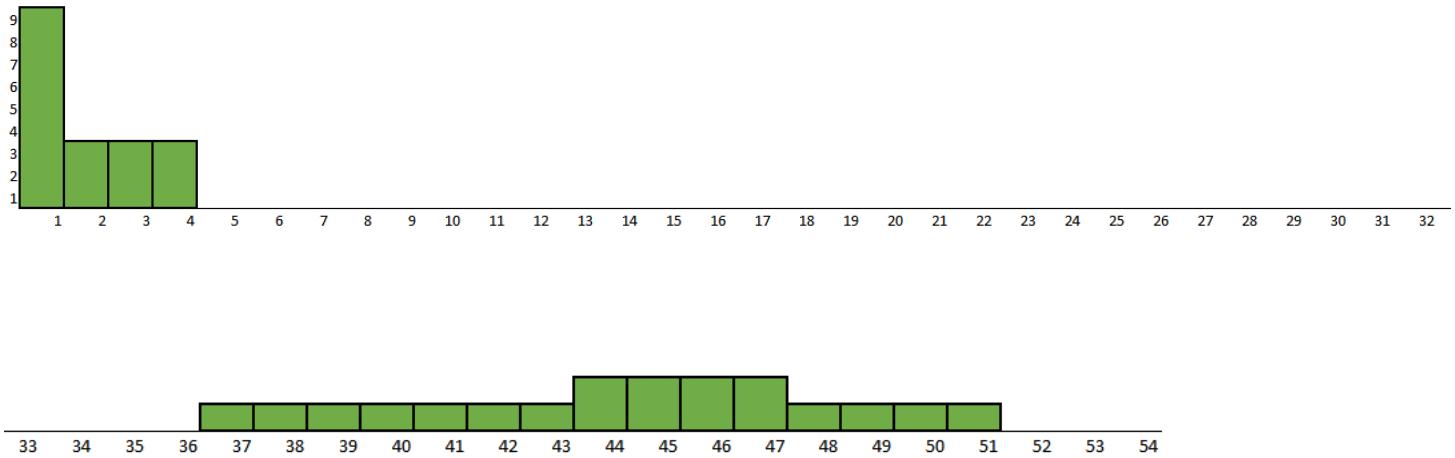
Diagrama Gantt con % completado

Inicio proyecto	06/06/2020
Fin proyecto	27/07/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Selección, descripción técnica y fundamentación del puesto de trabajo genérico	06/06/2020	4	11/06/2020	100%	4,00
Selección, descripción técnica y fundamentación y cantidad de servidores	13/07/2020	9	23/07/2020	100%	9,00
Selección, descripción técnica y fundamentación del sistema operativo para los puestos de trabajo	06/06/2020	1	08/06/2020	100%	1,00
Selec., desc. Téc. y fund. del sist. Op. para el servidor	06/06/2020	1	08/06/2020	100%	1,00
Selec., desc. y funda. del esq. lógico primario de la interconexión de equipos en la inst. de salud	20/07/2020	6	27/07/2020	100%	6,00

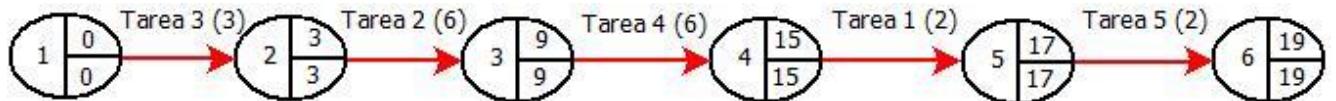
Diagrama Gantt con % completado



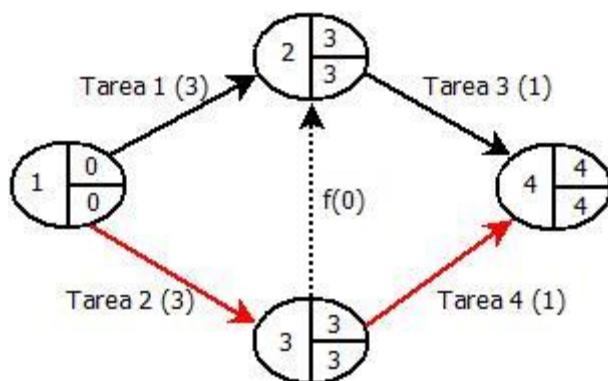


PERT Primer Entrega:

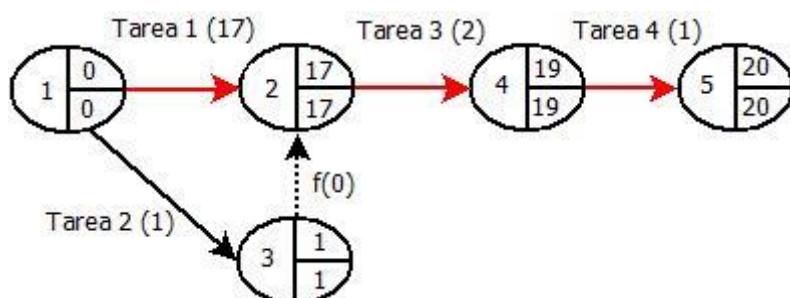
Análisis y Diseño de Aplicaciones



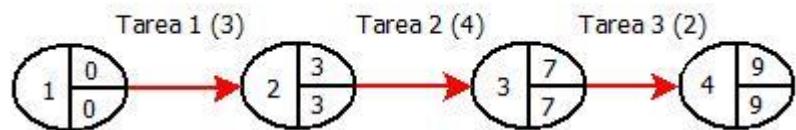
Base de Datos:



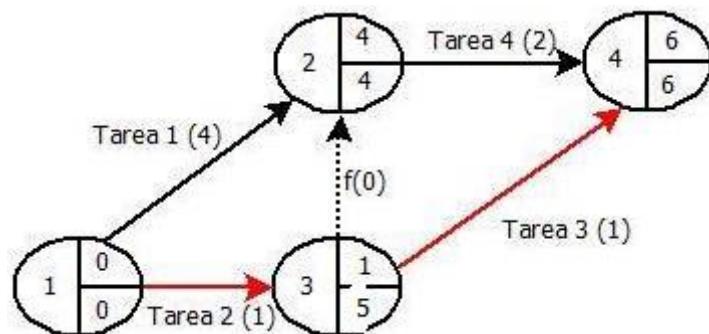
Formación Empresarial:



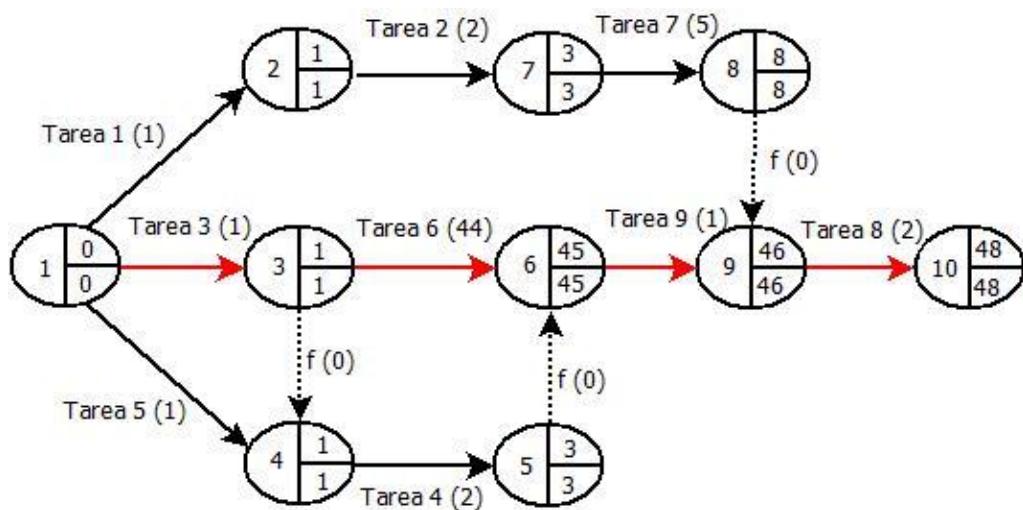
Ingles:



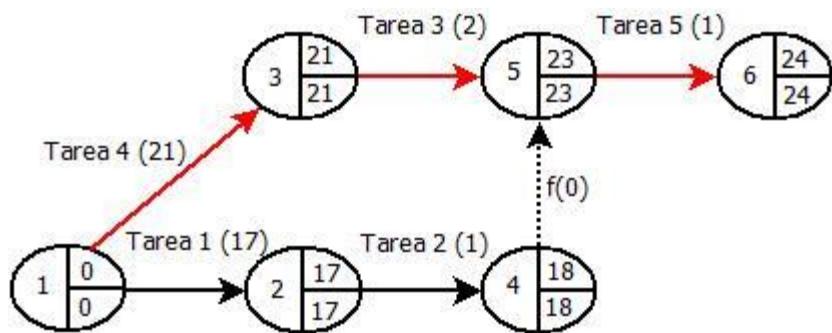
Programación:



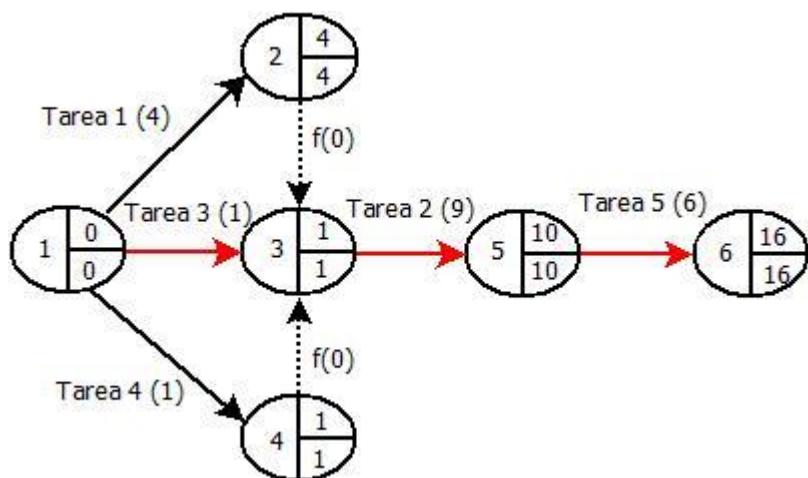
Proyecto:



Sistemas Operativos:



Taller:



Varianza Primera Entrega:

Tareas	Fecha inicio	Duración	Tiempo optimista	Tiempo pesimista	Varianza
Identificación integrantes	20/05/2020	3	1	3	0,11111111
Informe con enfasis en F.E	05/06/2020	4	2	4	0,11111111
Sugerencias	29/06/2020	2	1	2	0,02777778
Selección, descripción técnica y fundamentación del puesto de trabajo genérico	06/06/2020	4	3	6	0,25
Selección, descripción técnica y fundamentación y cantidad de servidores	13/07/2020	9	5	10	0,69444444
Selección, descripción técnica y fundamentación del sistema operativo para los puestos de trabajo	06/06/2020	1	1	3	0,11111111
Selecc., desc. Téc. y fund. del sist. Op. para el servidor	06/06/2020	1	1	3	0,11111111
Selecc., desc. y funda. del esq. lógico primario de la interconexión de equipos en la Inst. de salud	20/07/2020	6	4	8	0,44444444
Deberá entregarse un ejecutable desarrollado en Visual Basic.net que contenga todas las interfaces del sistema.	19/07/2020	4	2	6	0,44444444
El sist. deberá poner a disposición del usuario un menú principal a través del cual se accederá a todas las func.	19/07/2020	1	1	3	0,11111111
Las interfaces del sist. deberán contener todos los controles que permitan ingresar o seleccionar datos.	19/07/2020	1	1	3	0,11111111
El cód. deberá realizar las valid. de tipo de datos y tamaño max., así como la cons. entre los dist. datos ingre. o selec.	19/07/2020	2	1	3	0,11111111
Nombre del Grupo, integrantes y sus roles	25/05/2020	1	1	2	0,02777778
Organización de la estructura de desarrollo	28/06/2020	2	2	4	0,11111111
Desarrollar un formato de acta de reuniones formales e informales	25/05/2020	1	1	2	0,02777778
Formularios de uso comun en Proyectos	25/05/2020	2	2	3	0,02777778
Reglas del grupo	25/05/2020	1	1	2	0,02777778
Acta de reuniones	20/05/2020	54	54	54	0
Ciclo de vida del proyecto	22/07/2020	5	3	6	0,25
Documentacion de Inicio y Planificación	26/07/2020	2	2	5	0,25
Diagrama de Gantt	26/07/2020	1	1	2	0,02777778
Relevamiento	25/07/2020	2	1	3	0,11111111
Estudio de Factibilidades	23/06/2020	6	3	6	0,25
Cálculo de métricas del proyecto	18/06/2020	3	2	5	0,25
Especificación de Requerimientos	02/07/2020	18	10	20	2,77777778
Análisis	26/07/2020	2	2	4	0,11111111

DER	16/07/2020	3	2	6	0,44444444
Esquema relacional normalizado (3era forma normal)	16/07/2020	3	2	6	0,44444444
RNE	27/07/2020	1	1	3	0,11111111
Diccionario de datos	27/07/2020	1	1	2	0,02777778
Relev. Y Just. Del SO a utilizar en terminales de usuarios como el servidor	10/06/2020	17	10	17	1,36111111
Manual de instalación del SO en el servidor	15/06/2020	1	1	3	0,11111111
Estudio de los diferentes roles del usuario del sistema	10/07/2020	2	2	4	0,11111111
Shell Script para la creación de usuarios	13/06/2020	21	15	30	6,25
Manual de instalación de MySQL junto a sus requisitos	13/07/2020	2	1	4	0,25
Configuración del servicio SSH en el cliente y servidor	15/07/2020	1	1	3	0,11111111
Sumario Ejecutivo	01/06/2020	17	13	20	1,36111111
Analisis de mercado	05/06/2020	1	1	3	0,11111111
Analisis FODA	02/07/2020	2	2	4	0,11111111
Relleno de formularios	15/07/2020	1	1	2	0,02777778

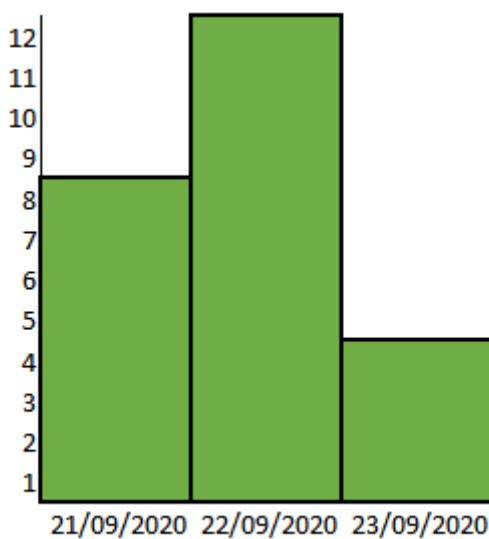
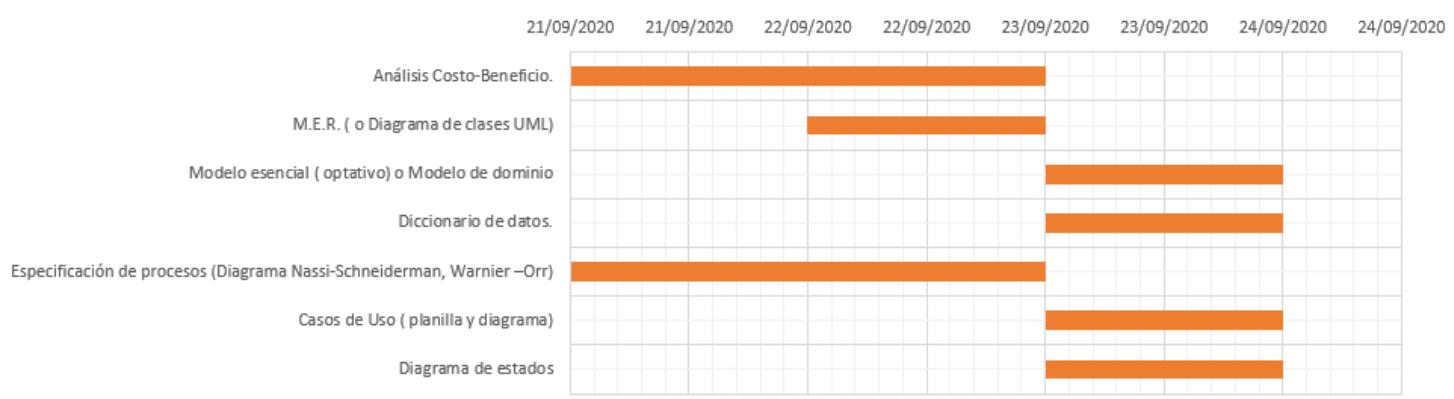
Diagramas GANTT Segunda Entrega:

Análisis y Diseño de Aplicaciones:

Diagrama Gantt con % completado

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Inicio proyecto	21/09/2020				
Fin proyecto	23/09/2020				
Análisis Costo-Beneficio.	21/09/2020	2	22/09/2020	100%	2,00
M.E.R. (o Diagrama de clases UML)	22/09/2020	1	22/09/2020	100%	1,00
Modelo esencial (optativo) o Modelo de dominio	23/09/2020	1	23/09/2020	100%	1,00
Diccionario de datos.	23/09/2020	1	23/09/2020	100%	1,00
Especificación de procesos (Diagrama Nassi-Schneiderman, Warnier –Orr)	21/09/2020	2	22/09/2020	100%	2,00
Casos de Uso (planilla y diagrama)	23/09/2020	1	23/09/2020	100%	1,00
Diagrama de estados	23/09/2020	1	23/09/2020	100%	1,00

Diagrama GANTT





Base de Datos:

Diagrama Gantt con % completado

Inicio proyecto	09/08/2020
Fin proyecto	25/08/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Correcciones (Primera entrega)	13/08/2020	3	17/08/2020	100%	3,00
DDL (Sentencias SQL de creación de tablas y estructuras necesarias)	09/08/2020	3	12/08/2020	100%	3,00
Estudio de los permisos sobre BD, tablas y columnas, considerando los diferentes roles	22/08/2020	2	25/08/2020	100%	2,00
Sentencias SQL de permisos a la Base de Datos.	22/08/2020	2	25/08/2020	100%	2,00
Primera implementación física de la Base de Datos	24/08/2020	1	24/08/2020	100%	1,00

Diagrama GANTT

09/08/2020 11/08/2020 13/08/2020 15/08/2020 17/08/2020 19/08/2020 21/08/2020 23/08/2020 25/08/2020

Correcciones (Primera entrega)



DDL (Sentencias SQL de creación de tablas y estructuras necesarias)



Estudio de los permisos sobre BD, tablas y columnas, considerando los diferentes roles
Estudio de los permisos sobre BD, tablas y columnas, considerando los diferentes roles
Estudio de los permisos sobre BD, tablas y columnas, considerando los diferentes roles



Sentencias SQL de permisos a la Base de Datos.



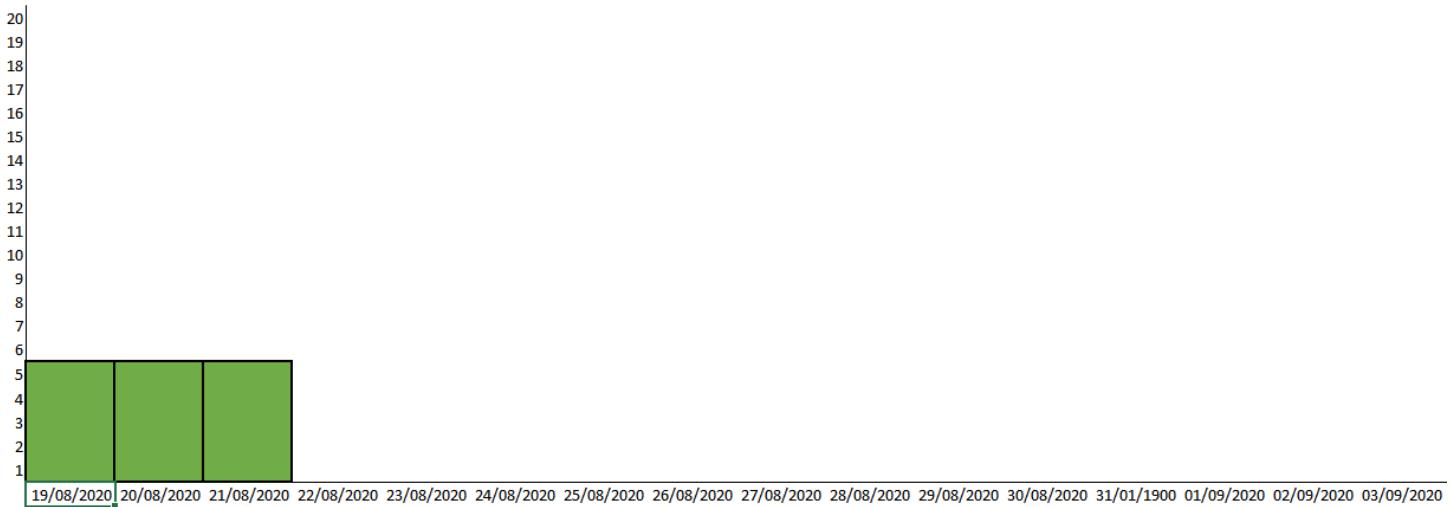
Primera implementación física de la Base de Datos



Formación Empresarial:

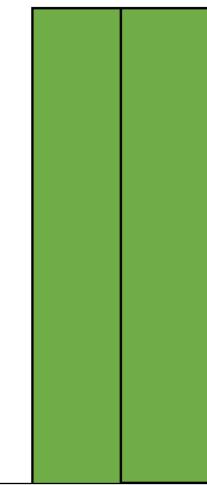
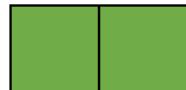
Diagrama Gantt con % completado					
Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Plan estrategico	19/08/2020	3	21/08/2020	100%	3,00
Misión	16/09/2020	2	17/09/2020	100%	2,00
Valores	16/09/2020	2	17/09/2020	100%	2,00
Objetivos	16/09/2020	2	17/09/2020	100%	2,00
Plan operativo	08/09/2020	2	09/09/2020	100%	2,00
Inversiones para la puesta en marcha del emprendimiento	16/09/2020	2	17/09/2020	100%	2,00

Diagrama GANTT





04/09/2020 05/09/2020 06/09/2020 07/09/2020 08/09/2020 09/09/2020 10/09/2020 11/09/2020 12/09/2020 13/09/2020 14/09/2020 15/09/2020 16/09/2020 17/09/2020 18/09/2020 19/09/2020

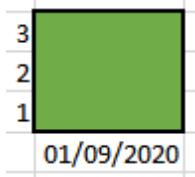
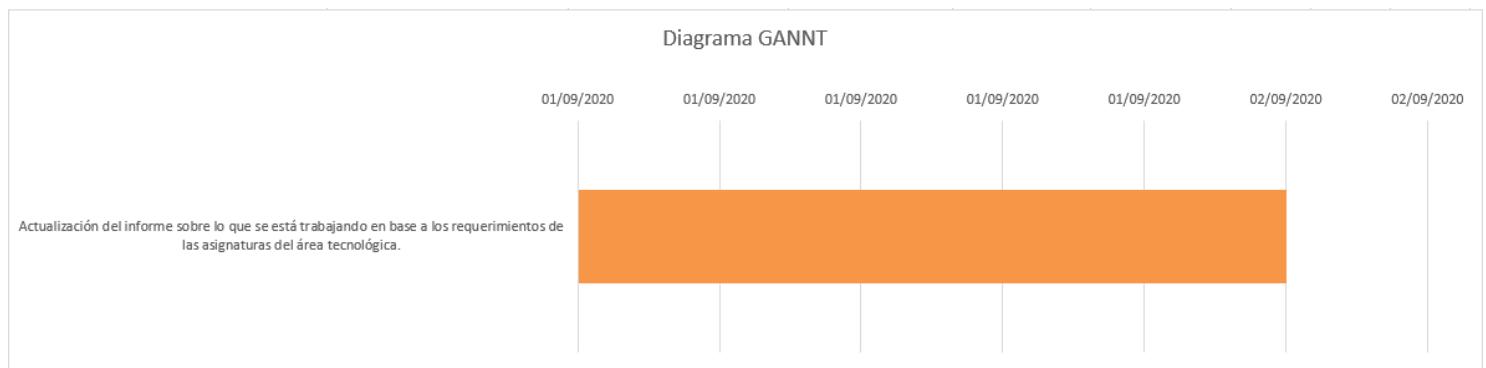




Ingles:

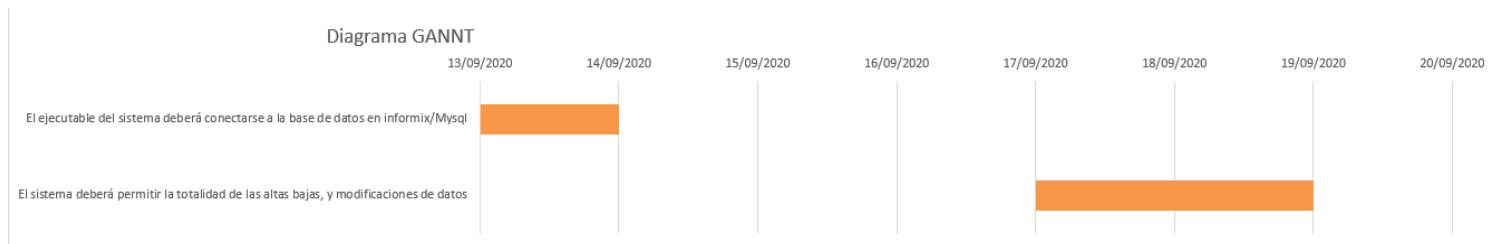
Diagrama Gantt con % completado					
Inicio proyecto	01/09/2020	Finalización		% Completado	Días completados
Tareas	Inicio	Duración en días			
Actualización del informe sobre lo que se está trabajando en base a los requerimientos de las asignaturas del área tecnológica.	01/09/2020	1	01/09/2020	100%	1,00

Diagrama GANNT



Programación:

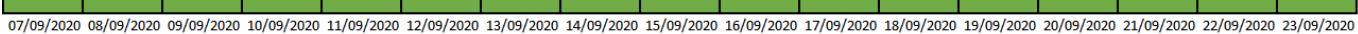
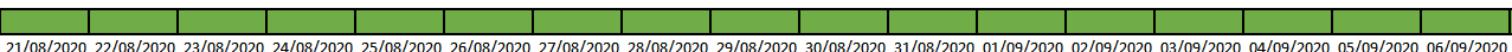
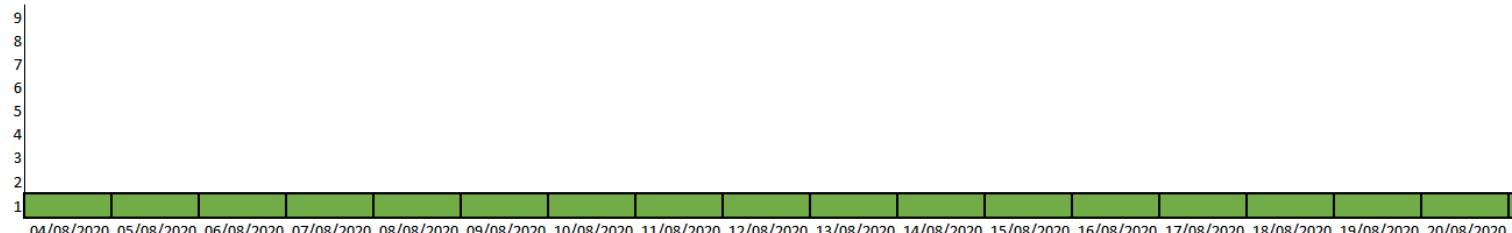
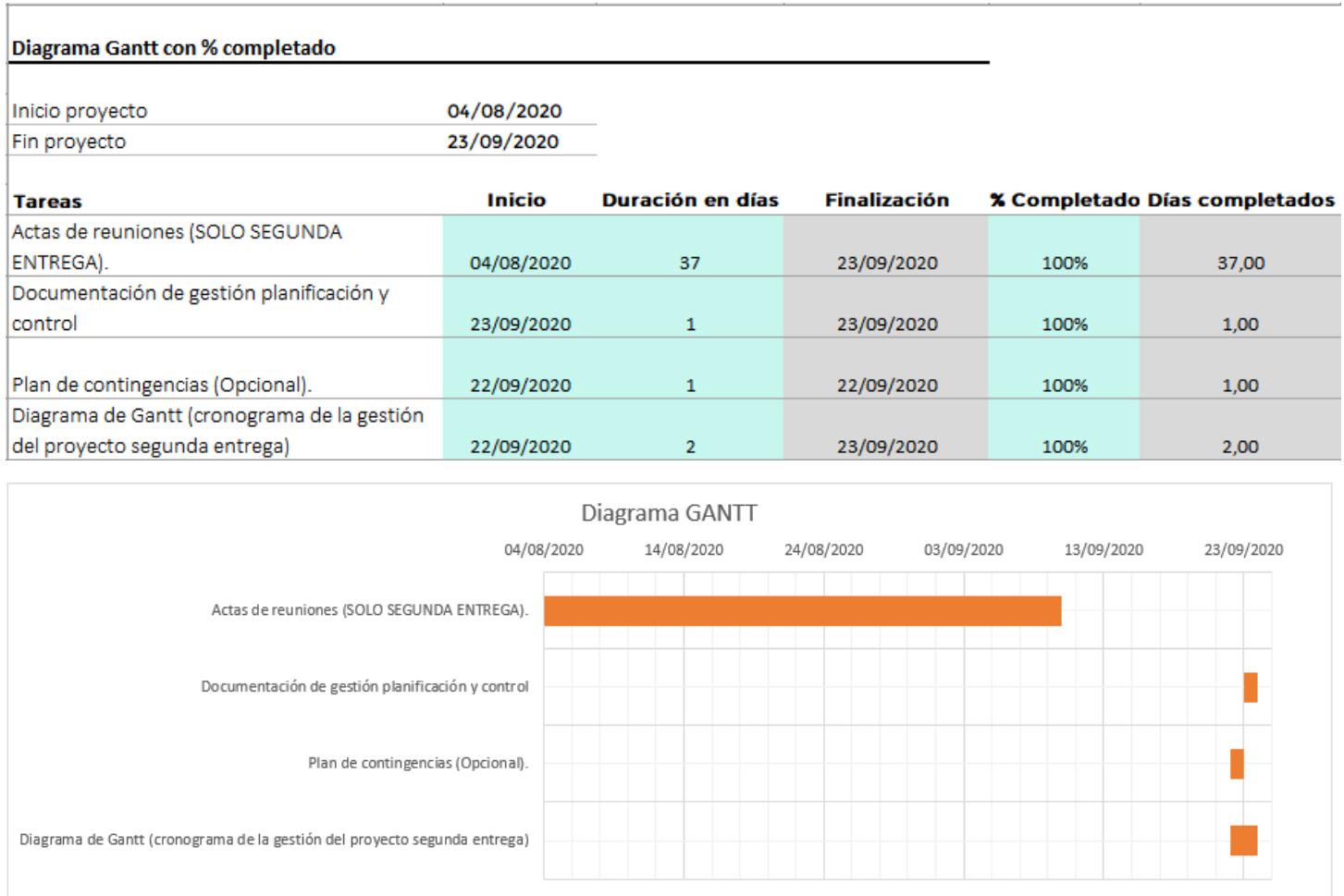
Diagrama Gantt con % completado					
Inicio proyecto	13/09/2020				
Fin proyecto	18/09/2020				
Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
El ejecutable del sistema deberá conectarse a la base de datos en informix/Mysql	13/09/2020	1	14/09/2020	100%	1,00
El sistema deberá permitir la totalidad de las altas bajas, y modificaciones de datos	17/09/2020	2	18/09/2020	100%	2,00



1						
2						
	13/09/2020	14/09/2020	15/09/2020	16/09/2020	17/09/2020	18/09/2020



Proyecto:





Sistemas Operativos:

Diagrama Gantt con % completado

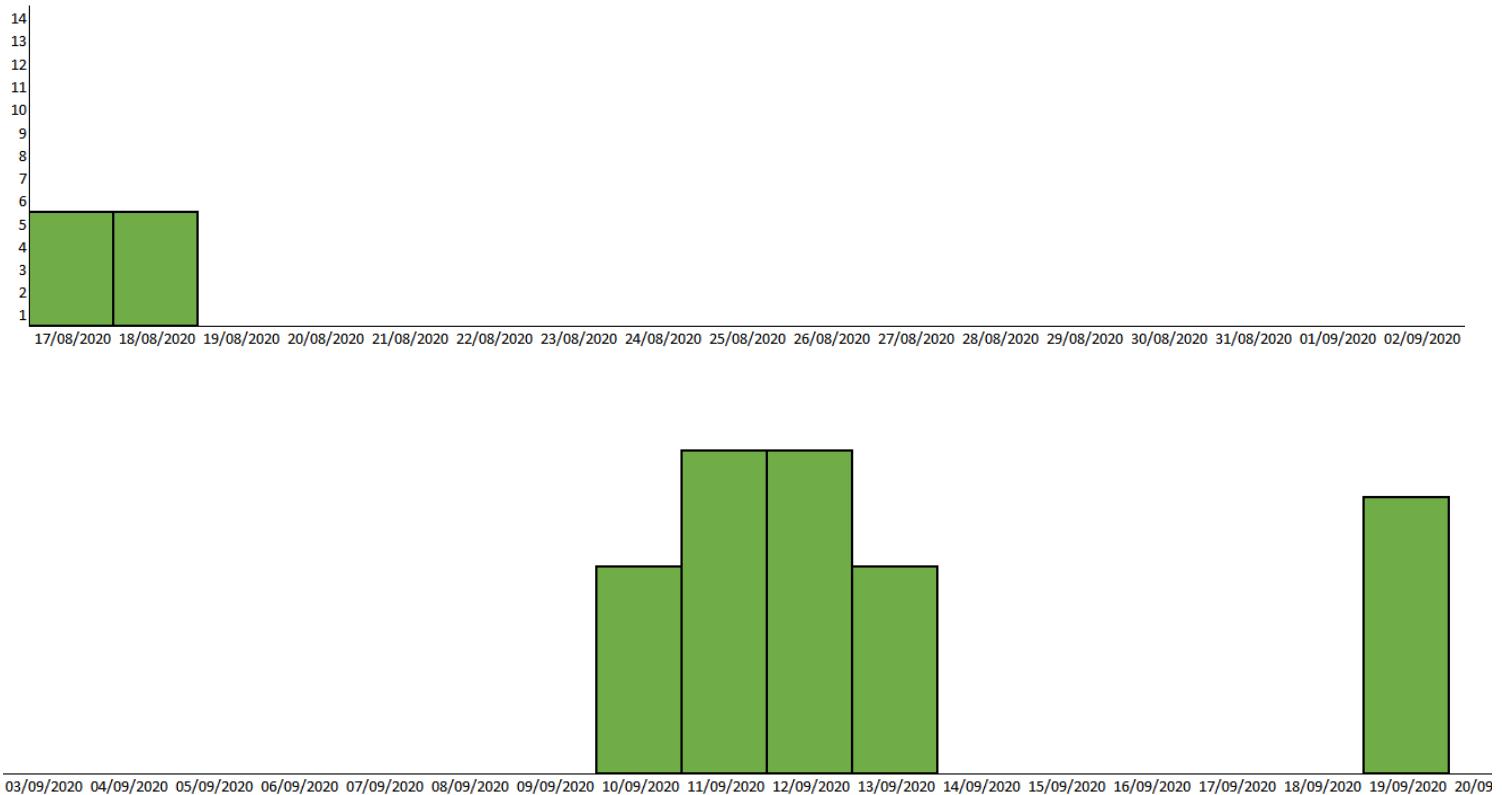
Inicio proyecto	17/08/2020
Fin proyecto	21/09/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Correcciones 1er Entrega	17/08/2020	2	18/08/2020	100%	2,00
Configuraciones de red en las terminales y el servidor.	10/09/2020	4	15/09/2020	100%	4,00
Archivos crontab con rutinas de backup y sus correspondientes scripts para el administrador.	19/09/2020	1	21/09/2020	100%	1,00
Usuarios de proyecto creados de acuerdo al estudio de roles de la primer entrega.	10/09/2020	4	15/09/2020	100%	4,00
Configuración del archivos sudoers con sus respectivos permisos para usuarios y grupos.	11/09/2020	2	14/09/2020	100%	2,00
Shell script con respaldo y restauración de base de datos.	19/09/2020	1	21/09/2020	100%	1,00
Menú para todos los Scripts	19/09/2020	1	21/09/2020	100%	1,00

Diagrama GANTT

17/08/2020 22/08/2020 27/08/2020 01/09/2020 06/09/2020 11/09/2020 16/09/2020 21/09/2020





Sociología:

Diagrama Gantt con % completado

Inicio proyecto	02/09/2020
Fin proyecto	18/09/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Formulación de preguntas	02/09/2020	1	02/09/2020	100%	1,00
Transcripción de entrevista	18/09/2020	1	18/09/2020	100%	1,00

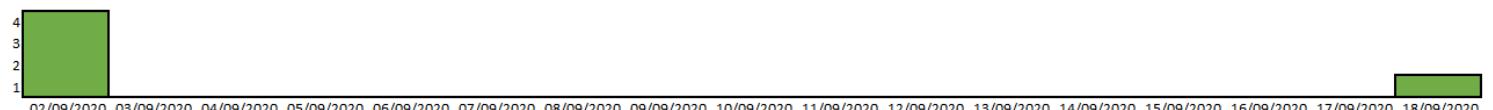
Diagrama Gantt

02/09/2020 04/09/2020 06/09/2020 08/09/2020 10/09/2020 12/09/2020 14/09/2020 16/09/2020 18/09/2020 20/09/2020

Formulación de preguntas



Transcripción de entrevista





Taller:

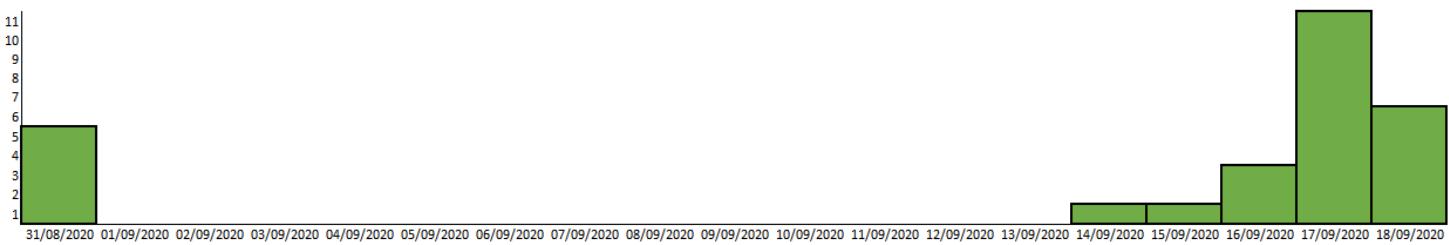
Diagrama Gantt con % completado

Inicio proyecto	31/08/2020
Fin proyecto	18/09/2020

Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Servicios provistos por los servidores	31/08/2020	1	31/08/2020	100%	1,00
Direccionamiento IP para cada VLAN segun los usuarios del sistema	16/09/2020	3	18/09/2020	100%	3,00
Detalle y esquema de la forma en la cual se interconectan los puestos y el armado de los rack con el respectivo	14/09/2020	4	17/09/2020	100%	4,00
Detalle de las UPS para el equipamiento clinico que interconecta las instituciones y un puesto de trabajo por	17/09/2020	1	17/09/2020	100%	1,00
Cableado estructurado	17/09/2020	2	18/09/2020	100%	2,00
calculo de materiales nesesarios para la implementacion de red Y el equipamiento informatico	17/09/2020	1	17/09/2020	100%	1,00

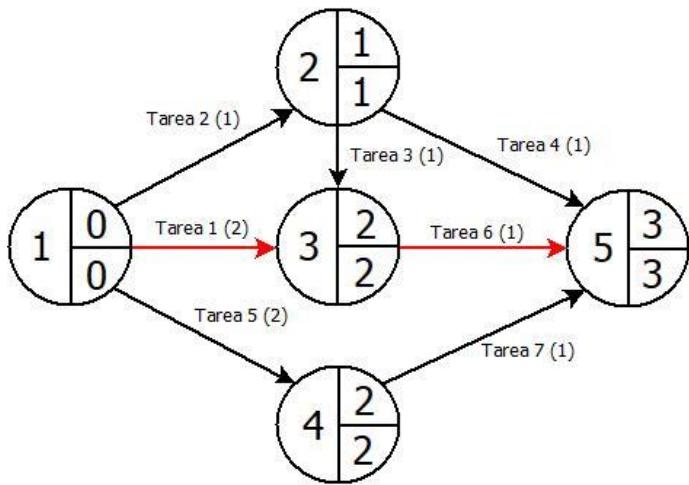
Diagrama GANTT

31/08/2020 02/09/2020 04/09/2020 06/09/2020 08/09/2020 10/09/2020 12/09/2020 14/09/2020 16/09/2020 18/09/2020 20/09/2020

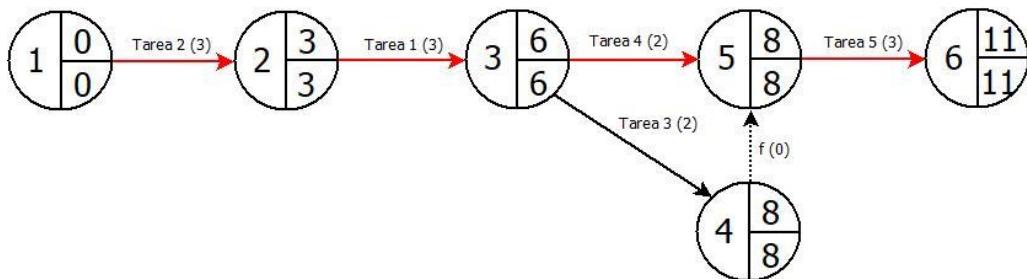


PERT Segunda Entrega:

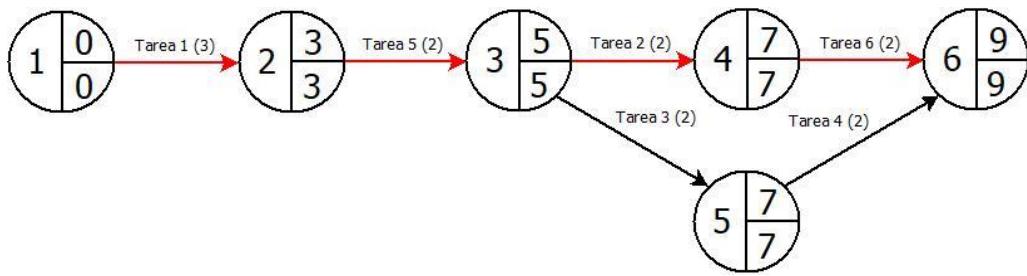
Análisis y Diseño de Aplicaciones.



Base de Datos:



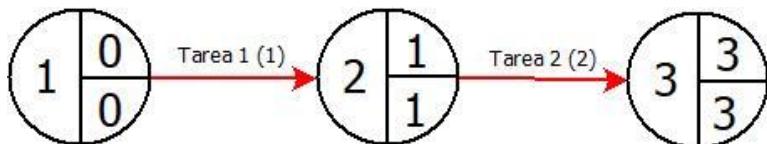
Formación Empresarial:



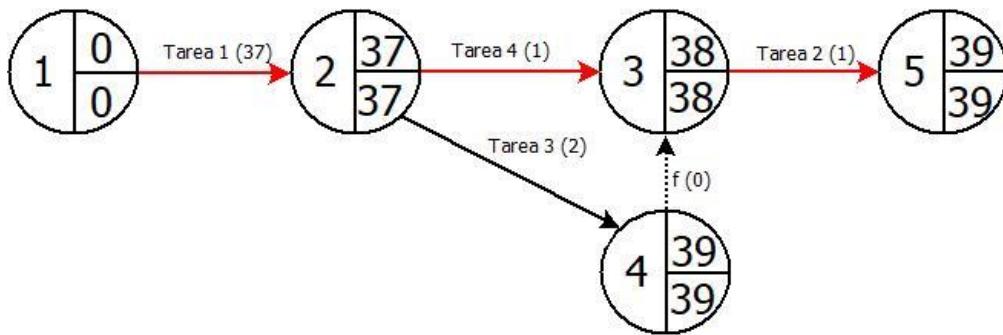
Ingles:



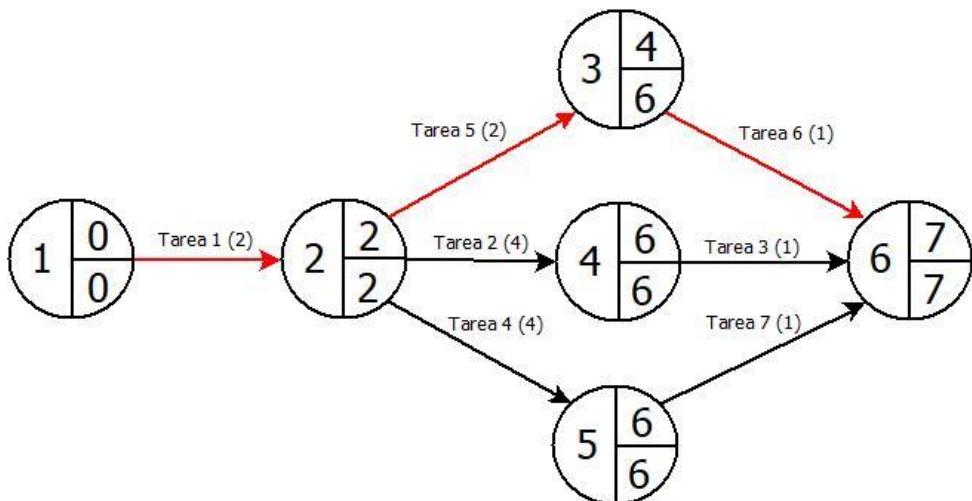
Programación:



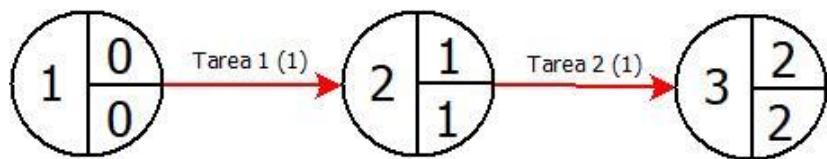
Proyecto:



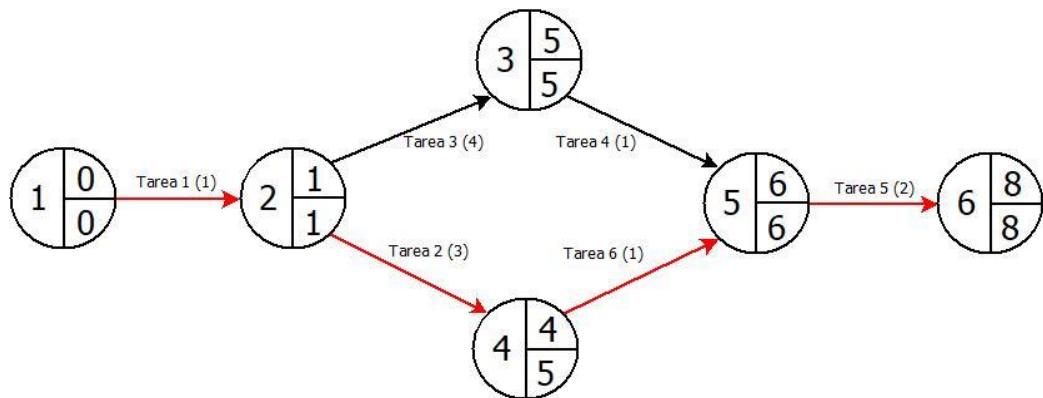
Sistemas Operativos:



Sociología:



Taller:



Varianza Segunda Entrega:

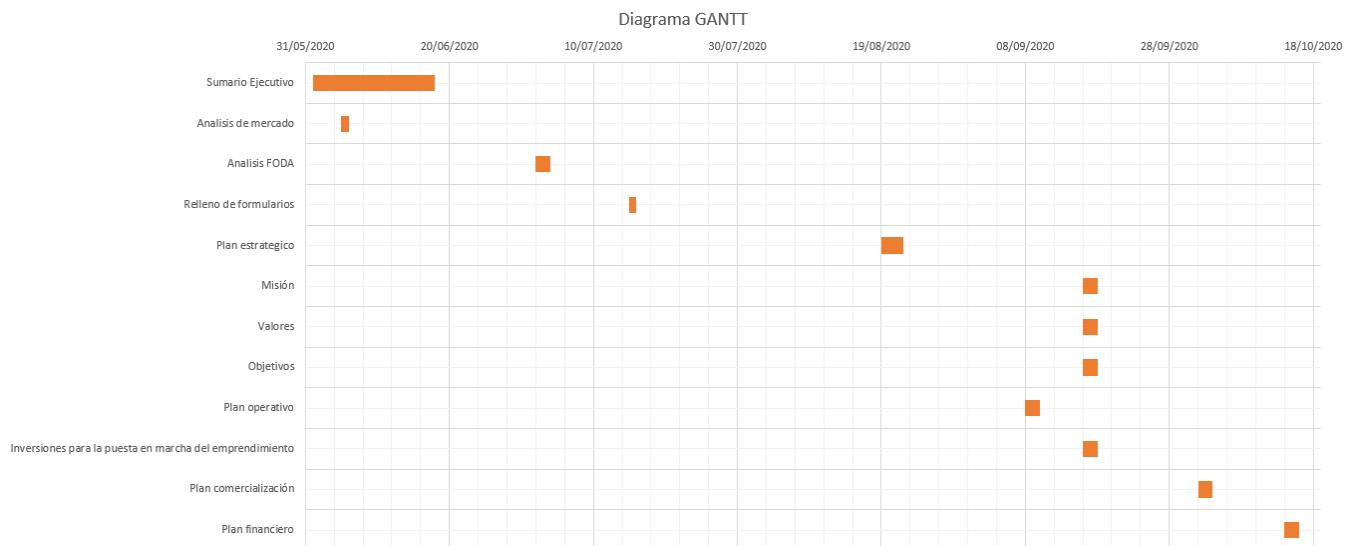
Tareas	Fecha inicio	Duración	Tiempo optimista	Tiempo pesimista	Varianza
El ejecutable del sistema deberá conectarse a la base de datos en informix/Mysql	13/09/2020	1	1	3	0,1111111111
El sistema deberá permitir la totalidad de las altas bajas, y modificaciones de datos	17/09/2020	2	2	10	1,77777778
Formulación de preguntas	02/09/2020	1	1	3	0,1111111111
Transcripción de entrevista	18/09/2020	1	1	3	0,1111111111
Correcciones (Primera entrega)	13/08/2020	3	2	5	0,25
DDL (Sentencias SQL de creación de tablas y estructuras necesarias)	09/08/2020	3	2	5	0,25
Estudio de los permisos sobre BD, tablas y usuarios	22/08/2020	2	1	3	0,1111111111
Sentencias SQL de permisos a la Base de Datos.	22/08/2020	2	1	3	0,1111111111
Primera implementación física de la Base de Datos	24/08/2020	1	1	3	0,1111111111
Actualización del informe sobre lo que se está trabajando en base a los requerimientos de	01/09/2020	1	1	3	0,1111111111
Análisis Costo-Beneficio.	21/09/2020	2	2	4	0,1111111111
M.E.R. (o Diagrama de clases UML)	22/09/2020	1	1	3	0,1111111111
Modelo esencial (optativo) o Modelo de dominio	23/09/2020	1	1	3	0,1111111111
Diccionario de datos.	23/09/2020	1	1	3	0,1111111111
Especificación de procesos (Diagrama Nassi-Schneiderman, Warnier -Orr)	21/09/2020	2	2	5	0,25
Casos de Uso (planilla y diagrama)	23/09/2020	1	1	4	0,25
Diagrama de estados	23/09/2020	1	1	4	0,25
Plan estratégico	19/08/2020	3	2	4	0,1111111111
Misión	16/09/2020	2	1	3	0,1111111111
Valores	16/09/2020	2	1	3	0,1111111111
Objetivos	16/09/2020	2	1	3	0,1111111111
Plan operativo	08/09/2020	2	2	4	0,1111111111
Inversiones para la puesta en marcha del emprendimiento	16/09/2020	2	1	3	0,1111111111

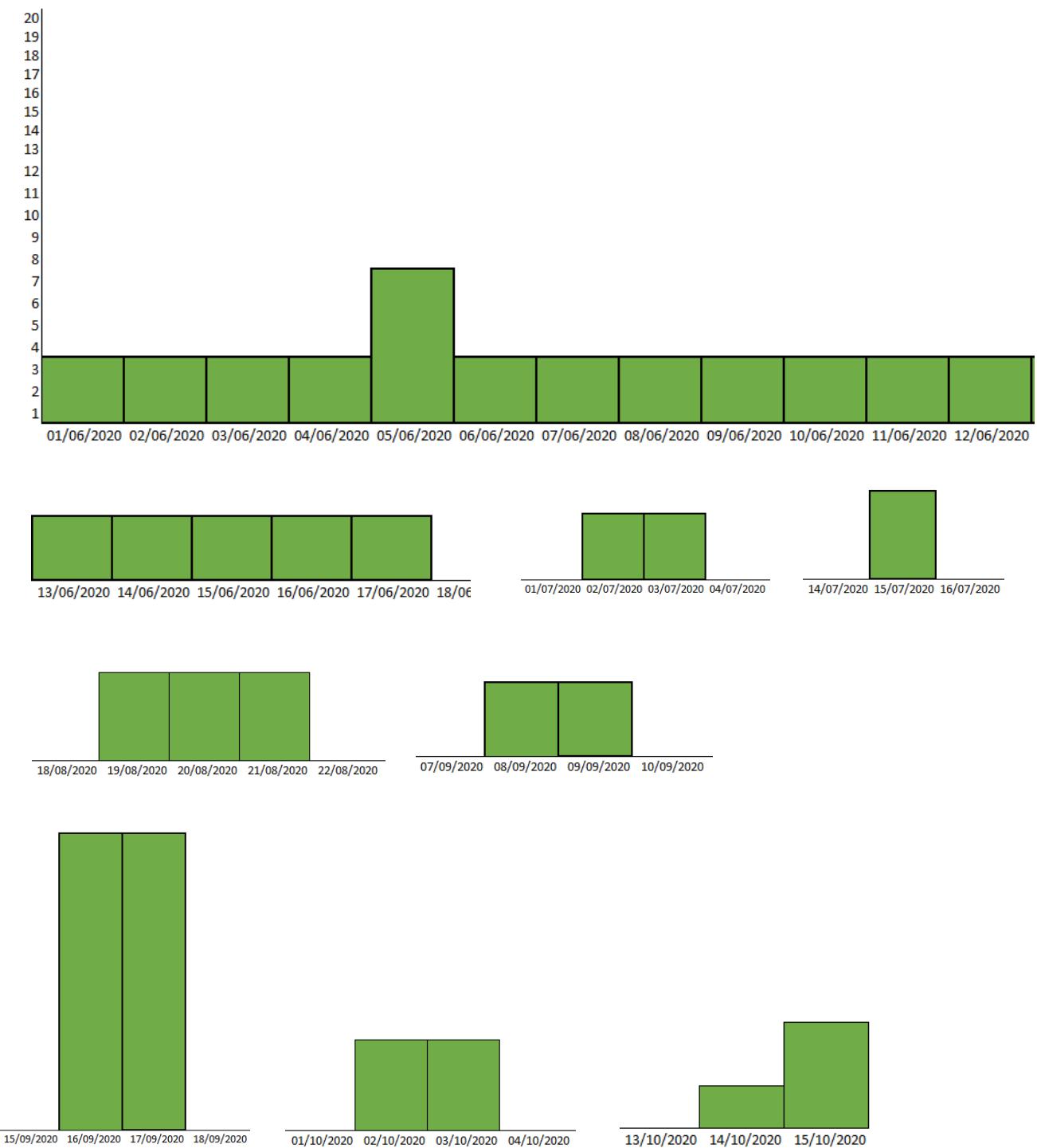
Correcciones 1er Entrega	17/08/2020	2	2	5	0,25
Configuraciones de red en las terminales y el servidor.	10/09/2020	4	2	4	0,1111111111
Archivos crontab con rutinas de backup y sus correspondientes scripts para el administrador.	19/09/2020	1	1	4	0,25
Usuarios de proyecto creados de acuerdo al estudio de roles de la primer entrega.	10/09/2020	4	2	4	0,1111111111
Configuración del archivos sudoers con sus respectivos permisos para usuarios y grupos.	11/09/2020	2	1	3	0,1111111111
Shell script con respaldo y restauración de base de datos.	19/09/2020	1	1	4	0,25
Menú para todos los Scripts	19/09/2020	1	1	3	0,1111111111
Actas de reuniones (SOLO SEGUNDA ENTREGA).	04/08/2020	37	37	37	0
Documentación de gestión planificación y control	23/09/2020	1	1	3	0,1111111111
Plan de contingencias (Opcional).	22/09/2020	1	1	2	0,027777778
Diagrama de Gantt (cronograma de la gestión del proyecto segunda entrega)	22/09/2020	2	1	4	0,25
Servicios provistos por los servidores	31/08/2020	1	1	3	0,1111111111
Direccionamiento IP para cada VLAN según los usuarios del sistema	16/09/2020	3	2	4	0,1111111111
Detalle y esquema de la forma en la cual se interconectan los puestos y el armado de los rack	14/09/2020	4	2	5	0,25
Detalle de las UPS para el equipamiento clínico que interconecta las instituciones y un puesto de trabajo	17/09/2020	1	1	3	0,1111111111
Cableado estructurado	17/09/2020	2	2	4	0,1111111111
Calculo de materiales necesarios para la implementacion de red Y el equipamiento	17/09/2020	1	1	3	0,1111111111

Diagrama GANTT Total:

Formación Empresarial:

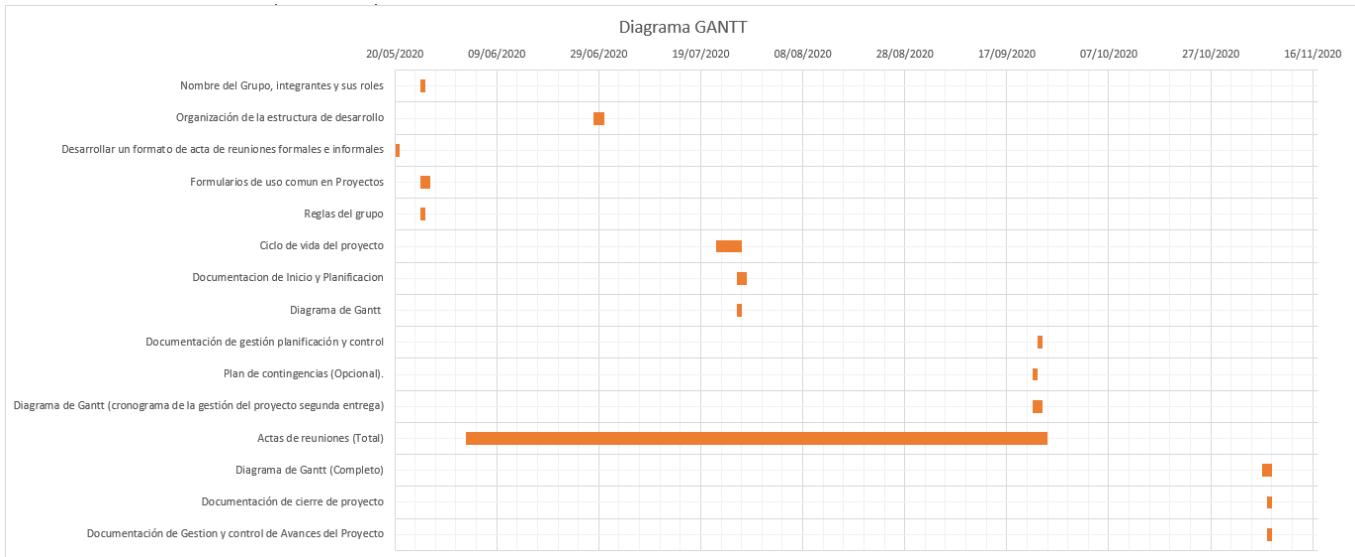
Diagrama Gantt con % completado					
	Inicio proyecto	20/05/2020			
	Fin proyecto	10/11/2020			
Tareas	Inicio	Duración en días	Finalización	% Completado	Días completados
Sumario Ejecutivo	01/06/2020	17	23/06/2020	100%	17,00
Analisis de mercado	05/06/2020	1	05/06/2020	100%	1,00
Analisis FODA	02/07/2020	2	03/07/2020	100%	2,00
Relleno de formularios	15/07/2020	1	15/07/2020	100%	1,00
Plan estrategico	19/08/2020	3	21/08/2020	100%	3,00
Misión	16/09/2020	2	17/09/2020	100%	2,00
Valores	16/09/2020	2	17/09/2020	100%	2,00
Objetivos	16/09/2020	2	17/09/2020	100%	2,00
Plan operativo	08/09/2020	2	09/09/2020	100%	2,00
Inversiones para la puesta en marcha del emprendimiento	16/09/2020	2	17/09/2020	100%	2,00
Plan comercialización	02/10/2020	2	05/10/2020	100%	2,00
Plan financiero	14/10/2020	2	15/10/2020	100%	2,00

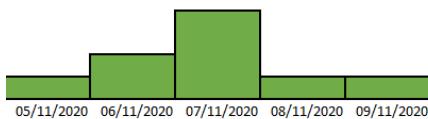
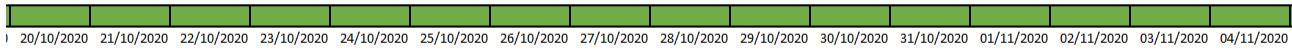
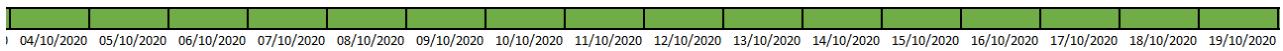
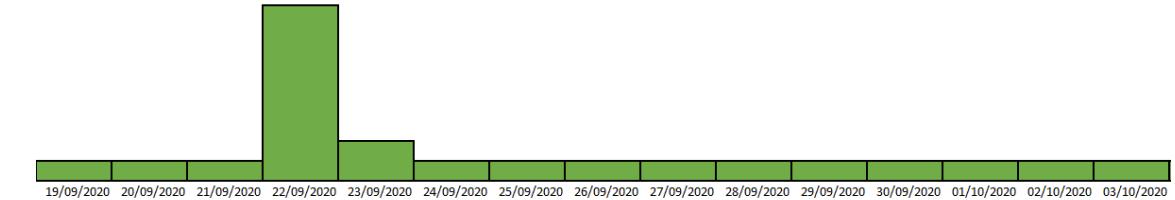
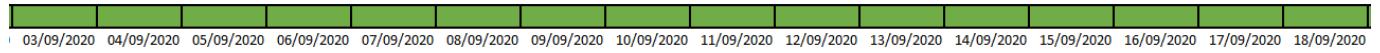
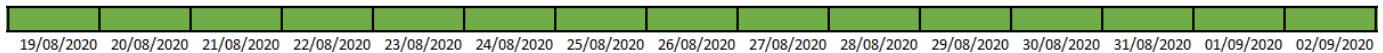
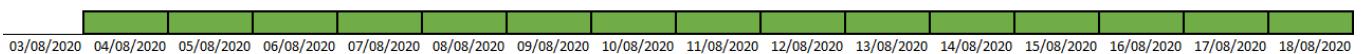
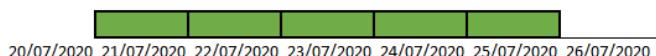
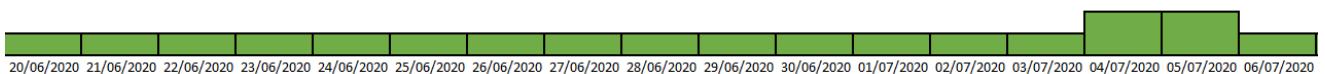
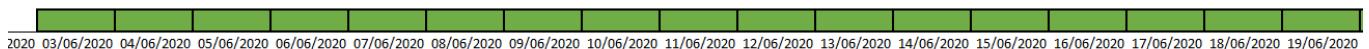
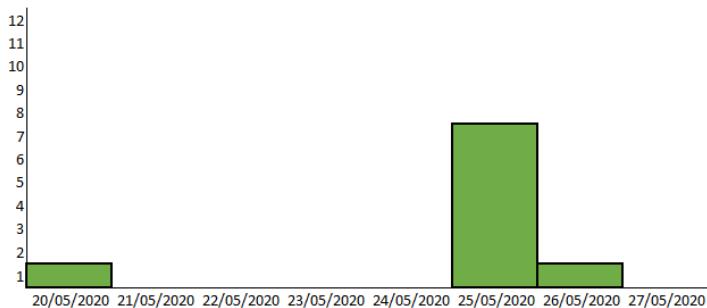




Proyecto:

Nombre del Grupo, integrantes y sus roles	25/05/2020	1	25/05/2020	100%	1,00
Organización de la estructura de desarrollo	28/06/2020	2	30/06/2020	100%	2,00
Desarrollar un formato de acta de reuniones formales e informales	20/05/2020	1	20/05/2020	100%	1,00
Formularios de uso comun en Proyectos	25/05/2020	2	26/05/2020	100%	2,00
Reglas del grupo	25/05/2020	1	25/05/2020	100%	1,00
Ciclo de vida del proyecto	22/07/2020	5	28/07/2020	100%	5,00
Documentacion de Inicio y Planificacion	26/07/2020	2	28/07/2020	100%	2,00
Diagrama de Gantt	26/07/2020	1	27/07/2020	100%	1,00
Documentación de gestión planificación y control	23/09/2020	1	23/09/2020	100%	1,00
Plan de contingencias (Opcional).	22/09/2020	1	22/09/2020	100%	1,00
Diagrama de Gantt (cronograma de la gestión del proyecto segunda entrega)	22/09/2020	2	23/09/2020	100%	2,00
Actas de reuniones (Total)	03/06/2020	114	09/11/2020	100%	114,00
Diagrama de Gantt (Completo)	06/11/2020	2	09/11/2020	100%	2,00
Documentación de cierre de proyecto	07/11/2020	1	09/11/2020	100%	1,00
Documentación de Gestión y control de Avances del Proyecto	07/11/2020	1	09/11/2020	100%	1,00



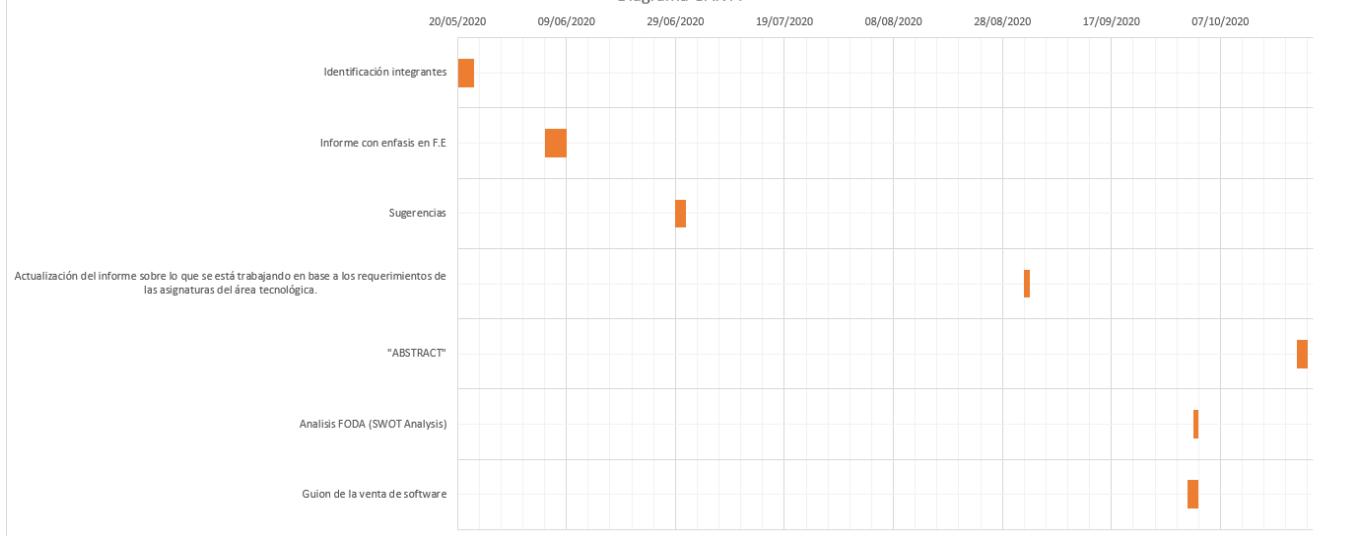




Ingles:

Identificación integrantes	20/05/2020	3	22/05/2020	100%	3,00
Informe con enfasis en F.E	05/06/2020	4	10/06/2020	100%	4,00
Sugerencias	29/06/2020	2	30/06/2020	100%	2,00
Actualización del informe sobre lo que se está trabajando en base a los requerimientos de las asignaturas del área tecnológica.	01/09/2020	1	01/09/2020	100%	1,00
"ABSTRACT"	21/10/2020	2	22/10/2020	100%	2,00
Analisis FODA (SWOT Analysis)	02/10/2020	1	02/10/2020	100%	1,00
Guion de la venta de software	01/10/2020	2	02/10/2020	100%	2,00

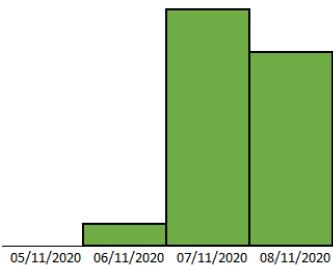
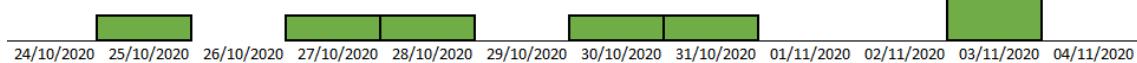
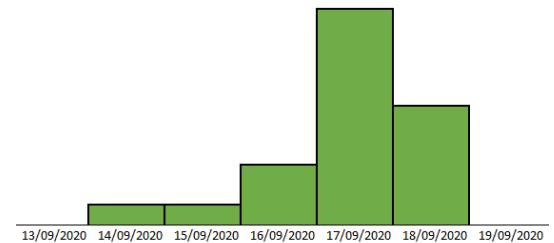
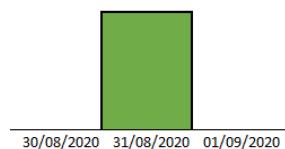
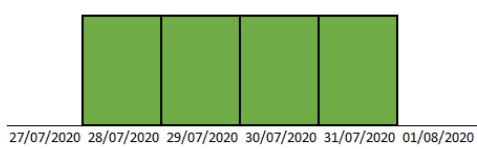
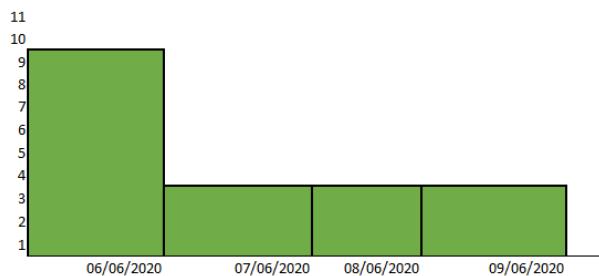
Diagrama GANTT



Taller:

Selección, descripción técnica y fundamentación del puesto de trabajo genérico	06/06/2020	4	11/06/2020	100%	4,00
Selección, descripción técnica y fundamentación y cantidad de servidores	13/07/2020	9	23/07/2020	100%	9,00
Selección, descripción técnica y fundamentación del sistema operativo para los Selecc., desc. Téc. y fund. del sist. Op. para el servidor	06/06/2020	1	08/06/2020	100%	1,00
06/06/2020	1	08/06/2020	100%	1,00	
Selecc., desc. y funda. del esq. lógico primario de la interconexión de equipos en la inst. de	20/07/2020	6	27/07/2020	100%	6,00
Servicios provistos por los servidores	31/08/2020	1	31/08/2020	100%	1,00
Direccionamiento IP para cada VLAN según los usuarios del sistema	16/09/2020	3	18/09/2020	100%	3,00
Detalle y esquema de la forma en la cual se interconectan los puestos y el armado de los	14/09/2020	4	17/09/2020	100%	4,00
Detalle de las UPS para el equipamiento clínico que interconecta las instituciones y un puesto	17/09/2020	1	17/09/2020	100%	1,00
Cableado estructurado	17/09/2020	2	18/09/2020	100%	2,00
Calculo de materiales necesarios para la implementación de red Y el equipamiento	17/09/2020	1	17/09/2020	100%	1,00
Correcciones de la segunda entrega	06/11/2020	3	10/11/2020	100%	3,00
Ruteo intervalan, presentacion de tablas de ruteo	07/11/2020	1	09/11/2020	100%	1,00
Esquema lógico y físico definitivo y funcional en packet tracer, con la config. de los disp. finales	07/11/2020	2	10/11/2020	100%	2,00
Contratacion de servicio de Internet para la institución	28/09/2020	4	01/10/2020	100%	4,00
Descripción de la seguridad informática que implementará	25/10/2020	1	26/10/2020	100%	1,00
Selección, descripción técnica y fundamentación del equipamiento utilizado	27/10/2020	2	28/10/2020	100%	2,00
Descripción de los servicios montados en el servidor con su respectiva configuración	07/11/2020	2	10/11/2020	100%	2,00
Descripción de los servicios montados en el router con su respectiva configuración	07/11/2020	2	10/11/2020	100%	2,00
Solución de interconexión entre casa central y por lo menos 2 sucursales	04/11/2020	1	04/11/2020	100%	1,00
Manual de procedimientos ante fallas del sistema incluido software de monitoreo	03/11/2020	1	03/11/2020	100%	1,00

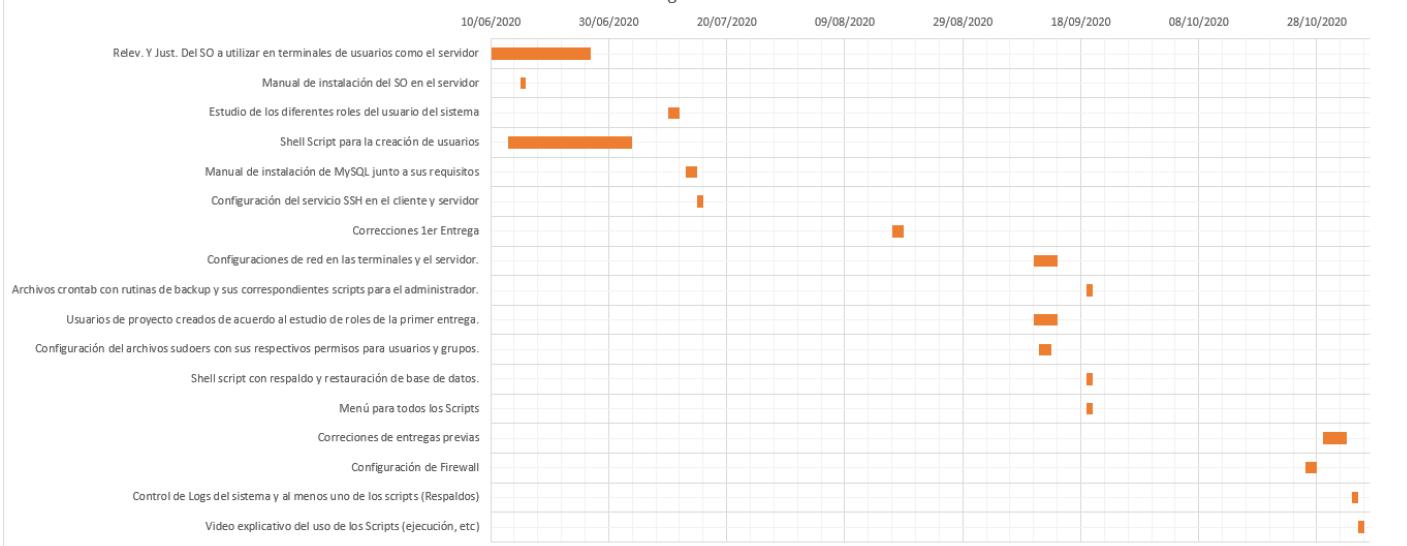


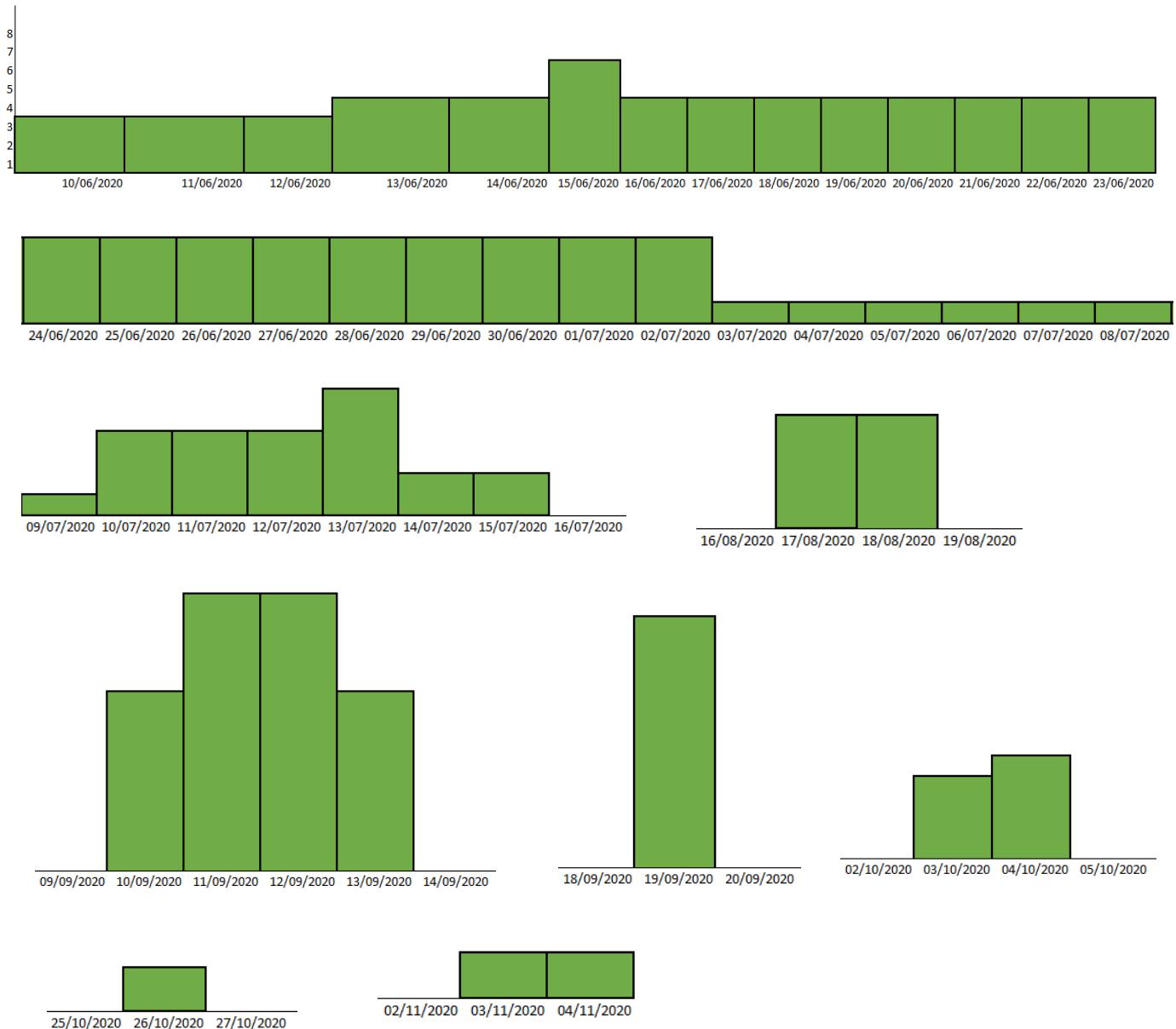


Sistemas Operativos:

Relev. Y Just. Del SO a utilizar en terminales de usuarios como el servidor	10/06/2020	17	02/07/2020	100%	17,00
Manual de instalación del SO en el servidor	15/06/2020	1	15/06/2020	100%	1,00
Estudio de los diferentes roles del usuario del sistema	10/07/2020	2	13/07/2020	100%	2,00
Shell Script para la creación de usuarios	13/06/2020	21	13/07/2020	100%	21,00
Manual de instalación de MySQL junto a sus requisitos	13/07/2020	2	14/07/2020	100%	2,00
Configuración del servicio SSH en el cliente y servidor	15/07/2020	1	15/07/2020	100%	1,00
Correcciones 1er Entrega	17/08/2020	2	18/08/2020	100%	2,00
Configuraciones de red en las terminales y el servidor.	10/09/2020	4	15/09/2020	100%	4,00
Archivos crontab con rutinas de backup y sus correspondientes scripts para el administrador.	19/09/2020	1	21/09/2020	100%	1,00
Usuarios de proyecto creados de acuerdo al estudio de roles de la primer entrega.	10/09/2020	4	15/09/2020	100%	4,00
Configuración del archivos sudoers con sus respectivos permisos para usuarios y grupos.	11/09/2020	2	14/09/2020	100%	2,00
Shell script con respaldo y restauración de base de datos.	19/09/2020	1	21/09/2020	100%	1,00
Menú para todos los Scripts	19/09/2020	1	21/09/2020	100%	1,00
Correcciones de entregas previas	29/10/2020	4	03/11/2020	100%	4,00
Configuración de Firewall	26/10/2020	2	27/10/2020	100%	2,00
Control de Logs del sistema y al menos uno de los scripts (Respaldos)	03/11/2020	1	03/11/2020	100%	1,00
Video explicativo del uso de los Scripts (ejecución, etc)	04/11/2020	1	04/11/2020	100%	1,00

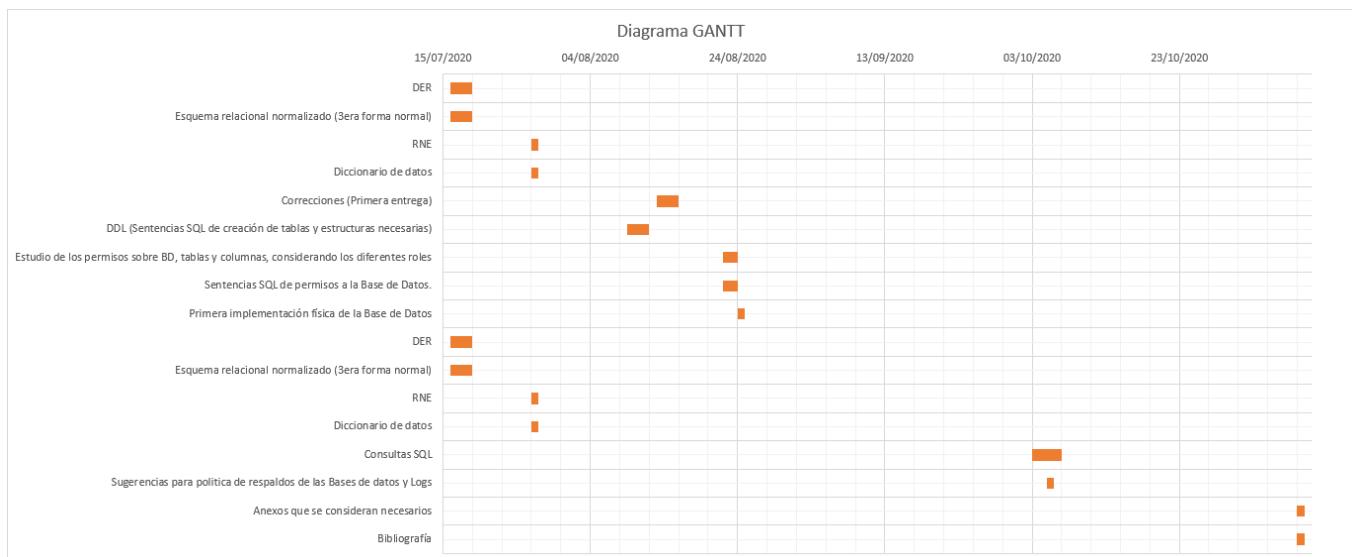
Diagrama GANTT

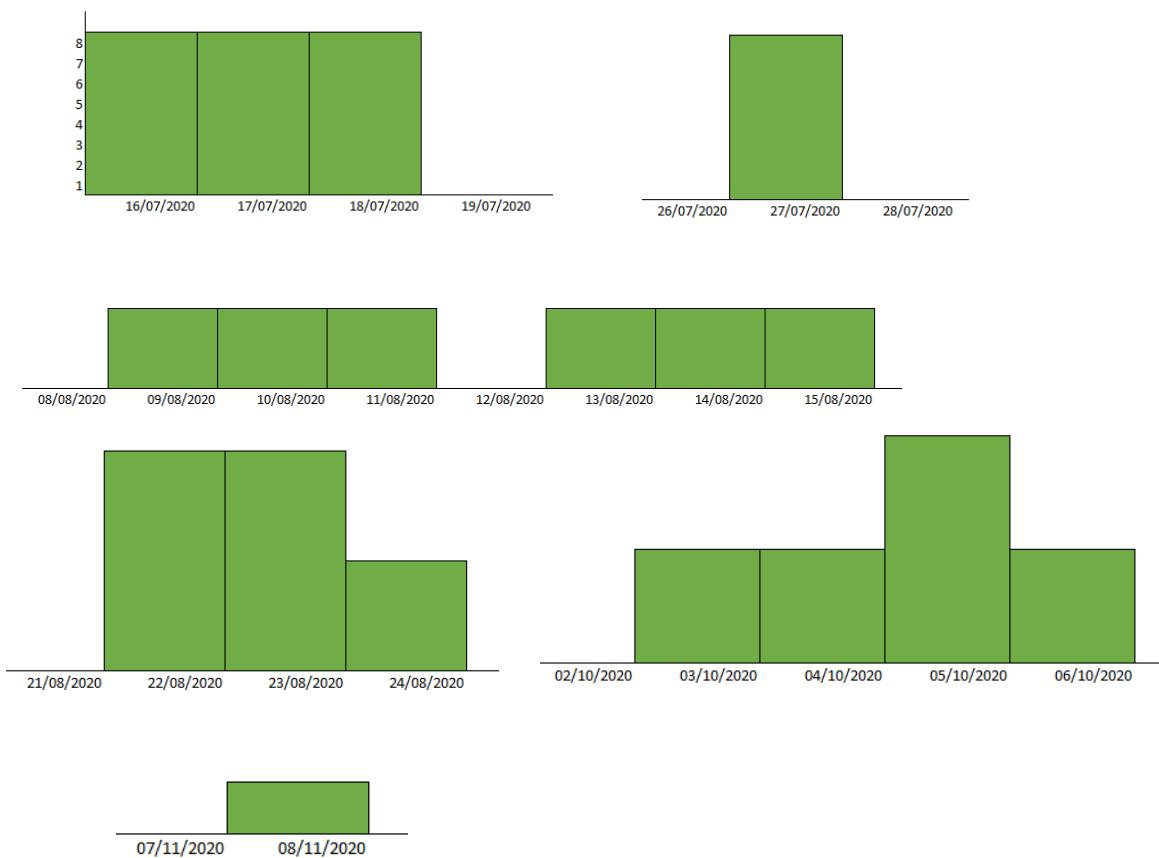




Base de Datos:

DER	16/07/2020	3	20/07/2020	100%	3,00
Esquema relacional normalizado (3era forma normal)	16/07/2020	3	20/07/2020	100%	3,00
RNE	27/07/2020	1	27/07/2020	100%	1,00
Diccionario de datos	27/07/2020	1	27/07/2020	100%	1,00
Correcciones (Primera entrega)	13/08/2020	3	17/08/2020	100%	3,00
DDL (Sentencias SQL de creación de tablas y estructuras necesarias)	09/08/2020	3	12/08/2020	100%	3,00
Estudio de los permisos sobre BD, tablas y columnas, considerando los diferentes roles	22/08/2020	2	25/08/2020	100%	2,00
Sentencias SQL de permisos a la Base de Datos.	22/08/2020	2	25/08/2020	100%	2,00
Primera implementación física de la Base de Datos	24/08/2020	1	24/08/2020	100%	1,00
DER	16/07/2020	3	20/07/2020	100%	3,00
Esquema relacional normalizado (3era forma normal)	16/07/2020	3	20/07/2020	100%	3,00
RNE	27/07/2020	1	27/07/2020	100%	1,00
Diccionario de datos	27/07/2020	1	27/07/2020	100%	1,00
Consultas SQL	03/10/2020	4	08/10/2020	100%	4,00
Sugerencias para política de respaldos de las Bases de datos y Logs	05/10/2020	1	05/10/2020	100%	1,00
Anexos que se consideran necesarios	08/11/2020	1	09/11/2020	100%	1,00
Bibliografía	08/11/2020	1	09/11/2020	100%	1,00

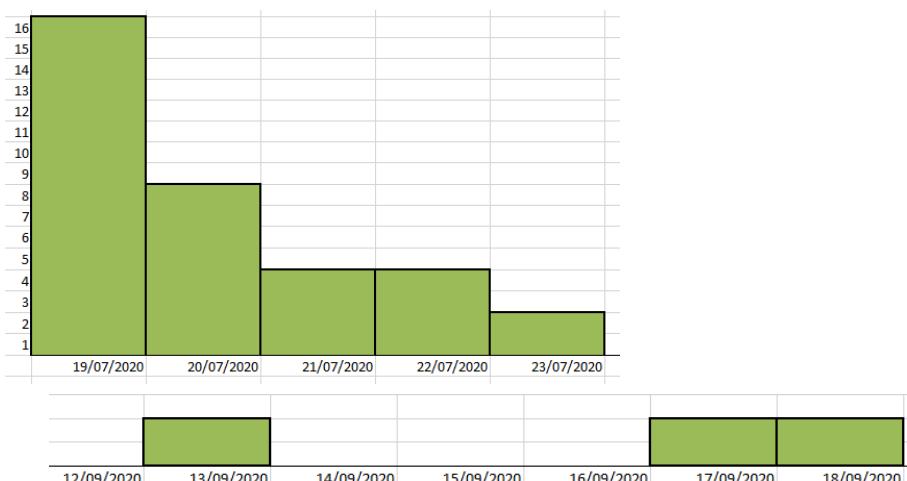
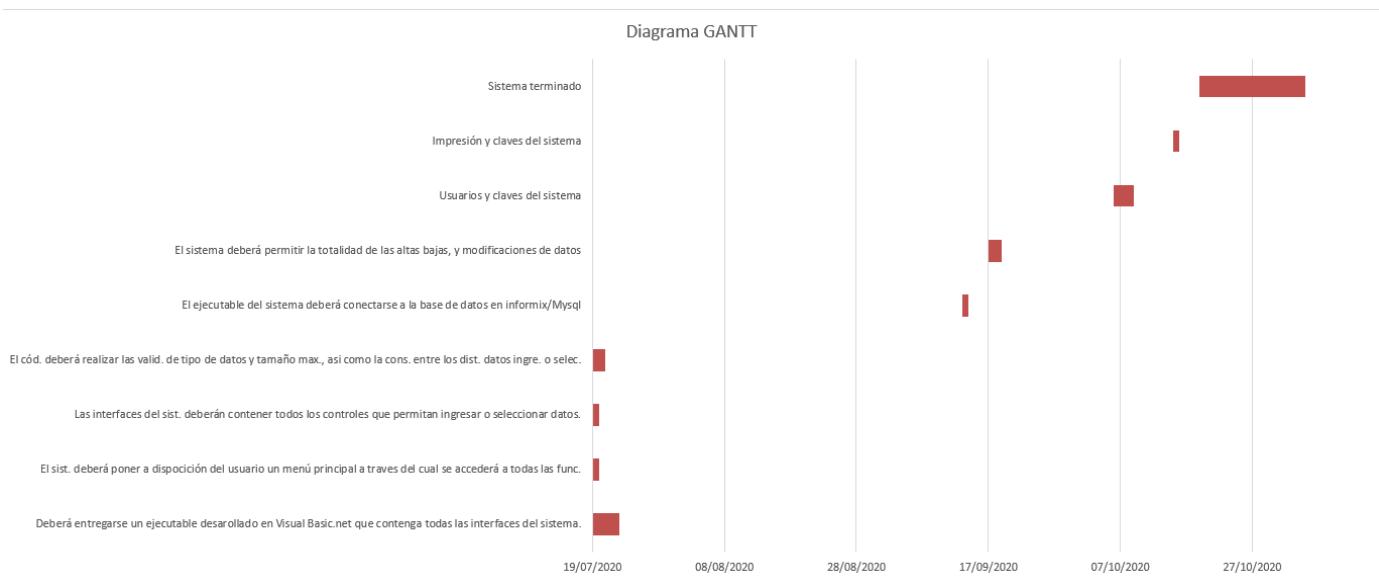




Programación:

Deberá entregarse un ejecutable desarrollado en Visual Basic.net que contenga todas las interfaces del sistema.	19/07/2020	4	23/07/2020	100%	4,00
El sist. deberá poner a disposición del usuario un menú principal a través del cual se accederá a todas las func.	19/07/2020	1	20/07/2020	100%	1,00
Las interfaces del sist. deberán contener todos los controles que permitan ingresar o seleccionar datos.	19/07/2020	1	20/07/2020	100%	1,00
El cód. deberá realizar las valid. de tipo de datos y tamaño max., así como la cons. entre los dist. datos ingre. o selec.	19/07/2020	2	21/07/2020	100%	2,00
El ejecutable del sistema deberá conectarse a la base de datos en informix/Mysql	13/09/2020	1	14/09/2020	100%	1,00
El sistema deberá permitir la totalidad de las altas bajas, y modificaciones de datos	17/09/2020	2	18/09/2020	100%	2,00
Usuarios y claves del sistema	06/10/2020	3	08/10/2020	100%	3,00
Impresión y claves del sistema	15/10/2020	1	15/10/2020	100%	1,00
Sistema terminado	19/10/2020	16	09/11/2020	100%	16,00

Diagrama GANTT





05/10/2020	06/10/2020	07/10/2020	08/10/2020
------------	------------	------------	------------

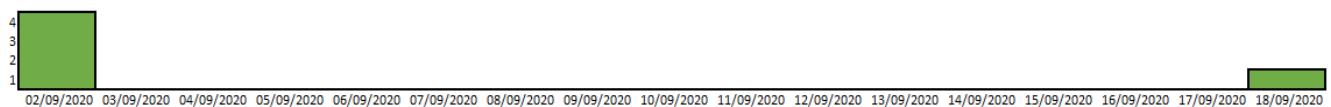
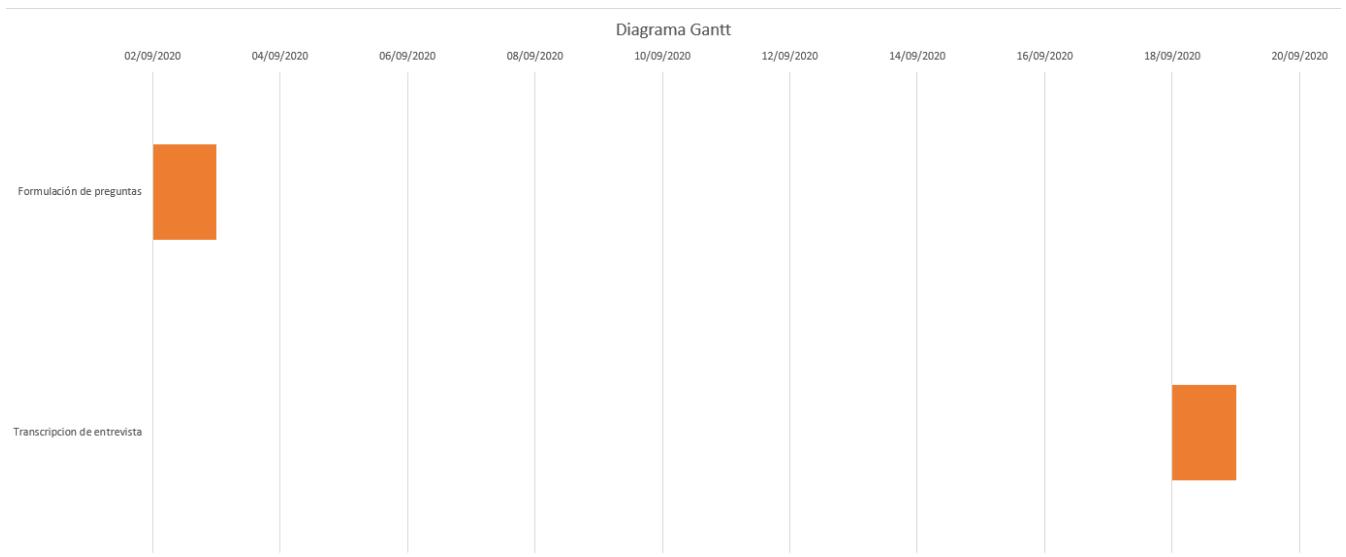
14/10/2020	15/10/2020	16/10/2020
------------	------------	------------

18/10/2020	19/10/2020	20/10/2020	21/10/2020	22/10/2020	23/10/2020	24/10/2020	25/10/2020	26/10/2020	27/10/2020	28/10/2020	29/10/2020
------------	------------	------------	------------	------------	------------	------------	------------	------------	------------	------------	------------

30/10/2020	31/10/2020	01/11/2020	02/11/2020	03/11/2020	04/11/2020
------------	------------	------------	------------	------------	------------

Sociología:

Formulación de preguntas	02/09/2020	1	02/09/2020	100%	1,00
Transcripción de entrevista	18/09/2020	1	18/09/2020	100%	1,00

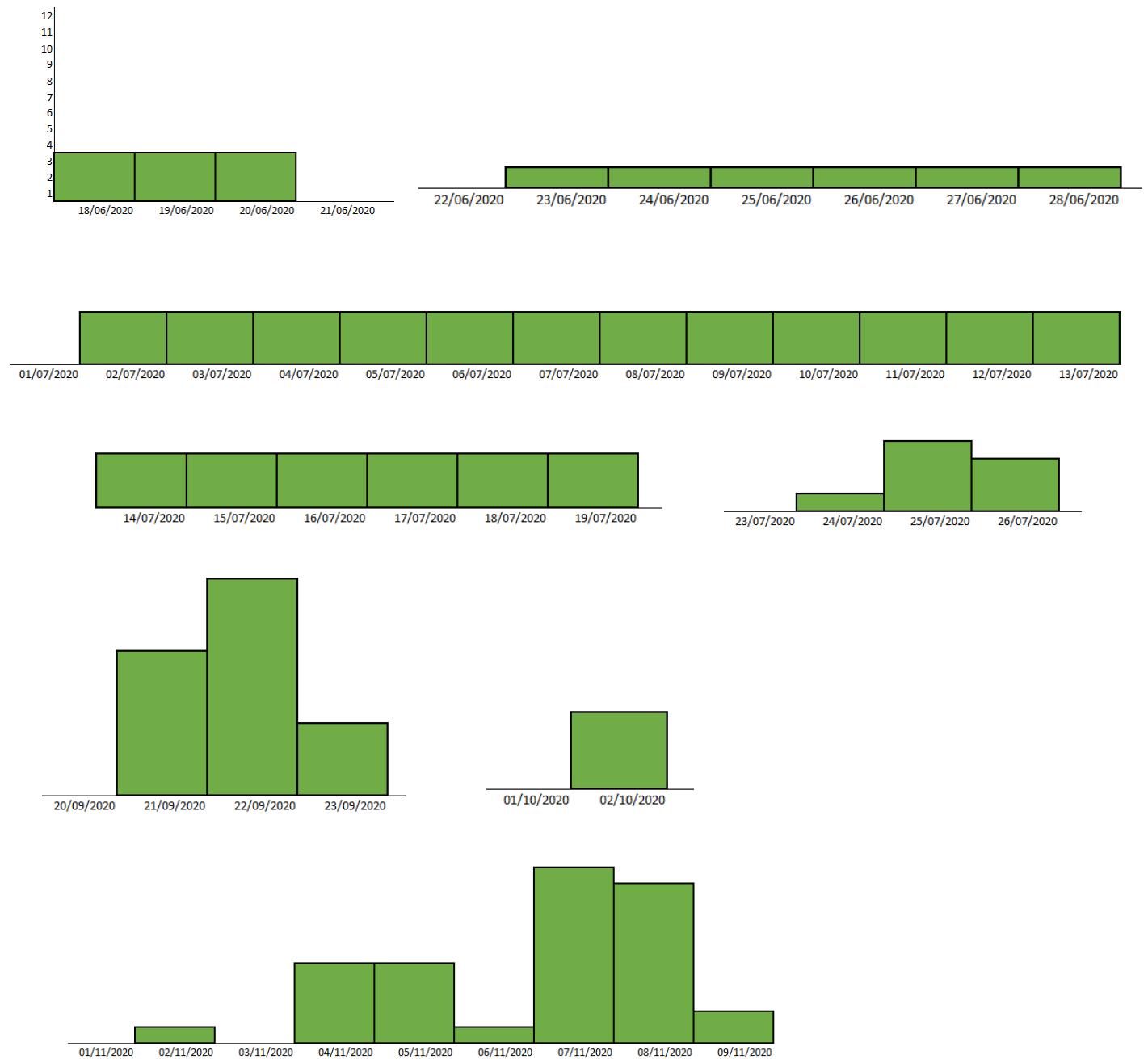


Análisis y Diseño de Aplicaciones:

Relevamiento	25/07/2020	2	28/07/2020	100%	2,00
Estudio de Factibilidades	23/08/2020	6	30/08/2020	100%	6,00
Cálculo de métricas del proyecto	18/08/2020	3	22/08/2020	100%	3,00
Especificación de Requerimientos	02/09/2020	18	27/09/2020	100%	18,00
Análisis	26/07/2020	2	28/07/2020	100%	2,00
Análisis Costo-Beneficio.	21/09/2020	2	22/09/2020	100%	2,00
M.E.R. (o Diagrama de clases UML)	22/09/2020	1	22/09/2020	100%	1,00
Modelo esencial (optativo) o Modelo de dominio	23/09/2020	1	23/09/2020	100%	1,00
Diccionario de datos.	23/09/2020	1	23/09/2020	100%	1,00
Especificación de procesos (Diagrama Nassi-Schneiderman, Warnier -Orr)	21/09/2020	2	22/09/2020	100%	2,00
Casos de Uso (planilla y diagrama)	23/09/2020	1	23/09/2020	100%	1,00
Diagrama de estados	23/09/2020	1	23/09/2020	100%	1,00
Carta jerárquica (Operativo)	07/11/2020	2	10/11/2020	100%	2,00
Diagrama de clases (Operativo)	02/11/2020	1	02/11/2020	100%	1,00
Plan de testing	07/11/2020	1	09/11/2020	100%	1,00
Manual de usuario	04/11/2020	1	04/11/2020	100%	1,00
Manual de administrador	05/11/2020	1	05/11/2020	100%	1,00
Anexos	08/11/2020	1	09/11/2020	100%	1,00
Agenda de reuniones (Operativo)	09/11/2020	1	09/11/2020	100%	1,00
Creditos y bibliografía	09/11/2020	1	09/11/2020	100%	1,00

Diagrama GANTT

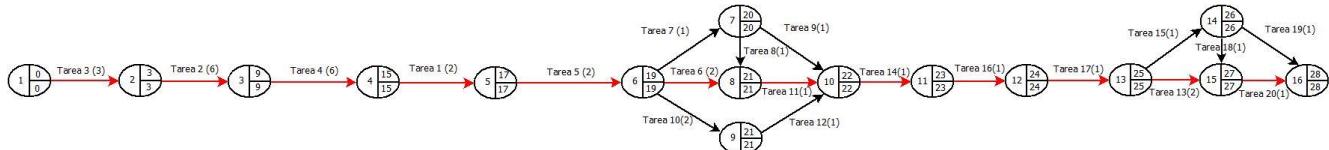




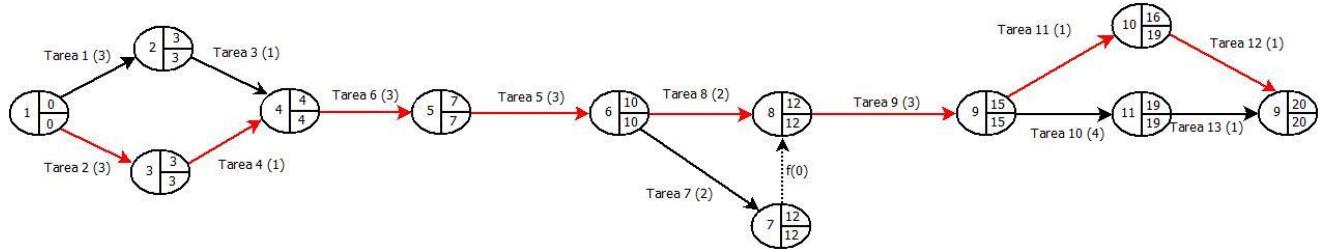
PERT Tercera Entrega:

Diagrama de gestión y control de avances del proyecto

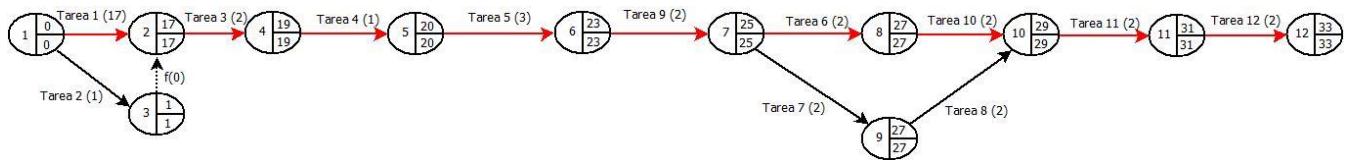
ANALISÍS Y DISEÑO DE APLICACIONES



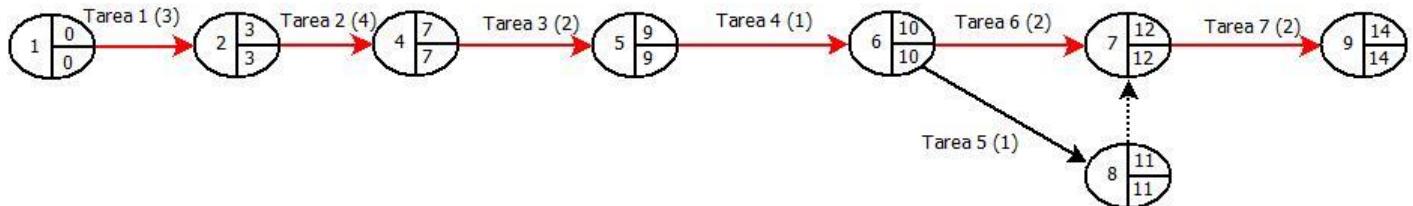
BASE DE DATOS



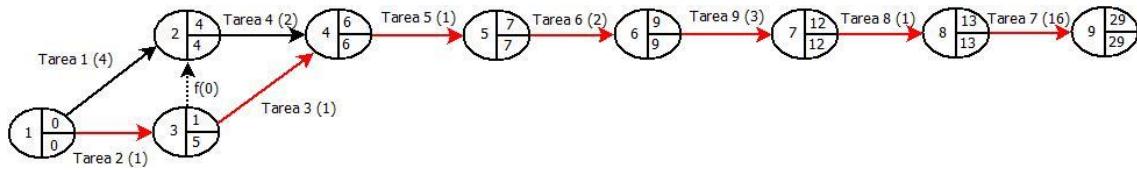
FORMACIÓN EMPRESARIAL



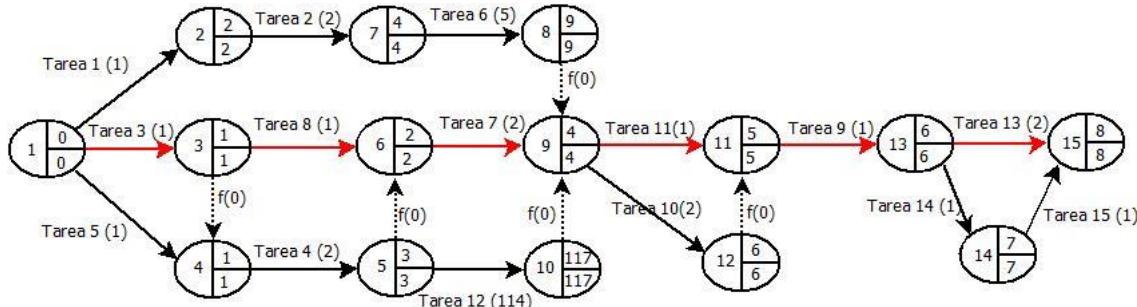
INGLÉS



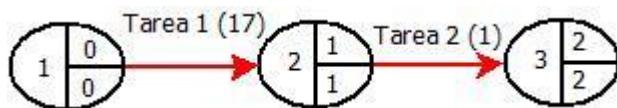
PROGRAMACIÓN



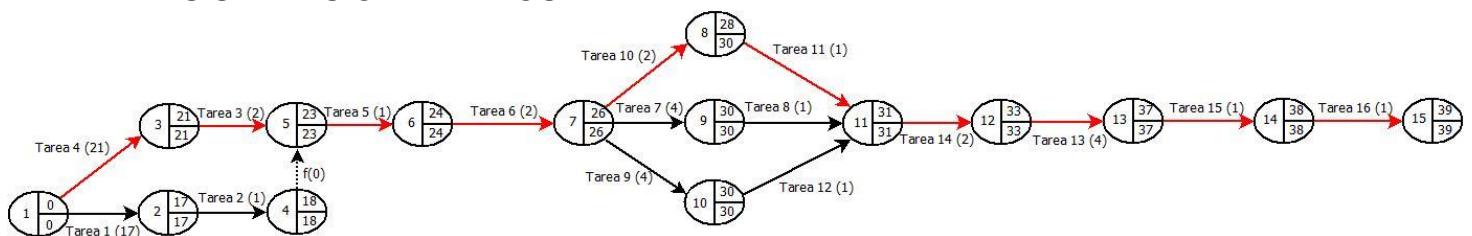
PROYECTO



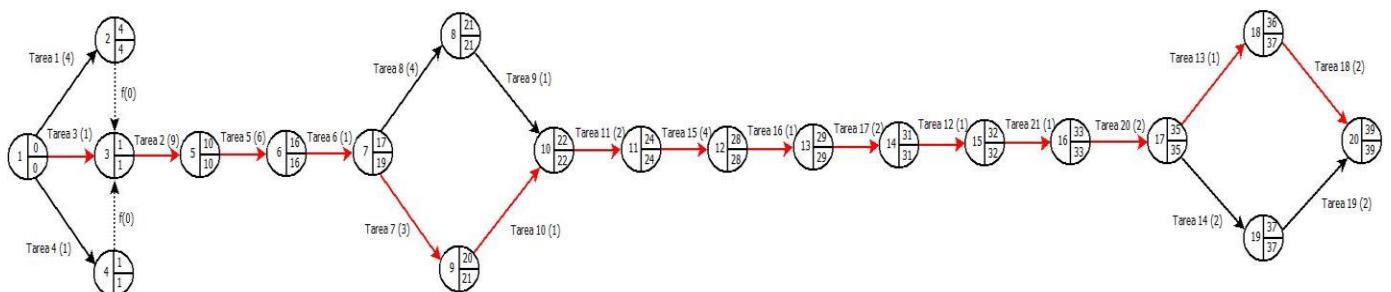
SOCIOLOGÍA



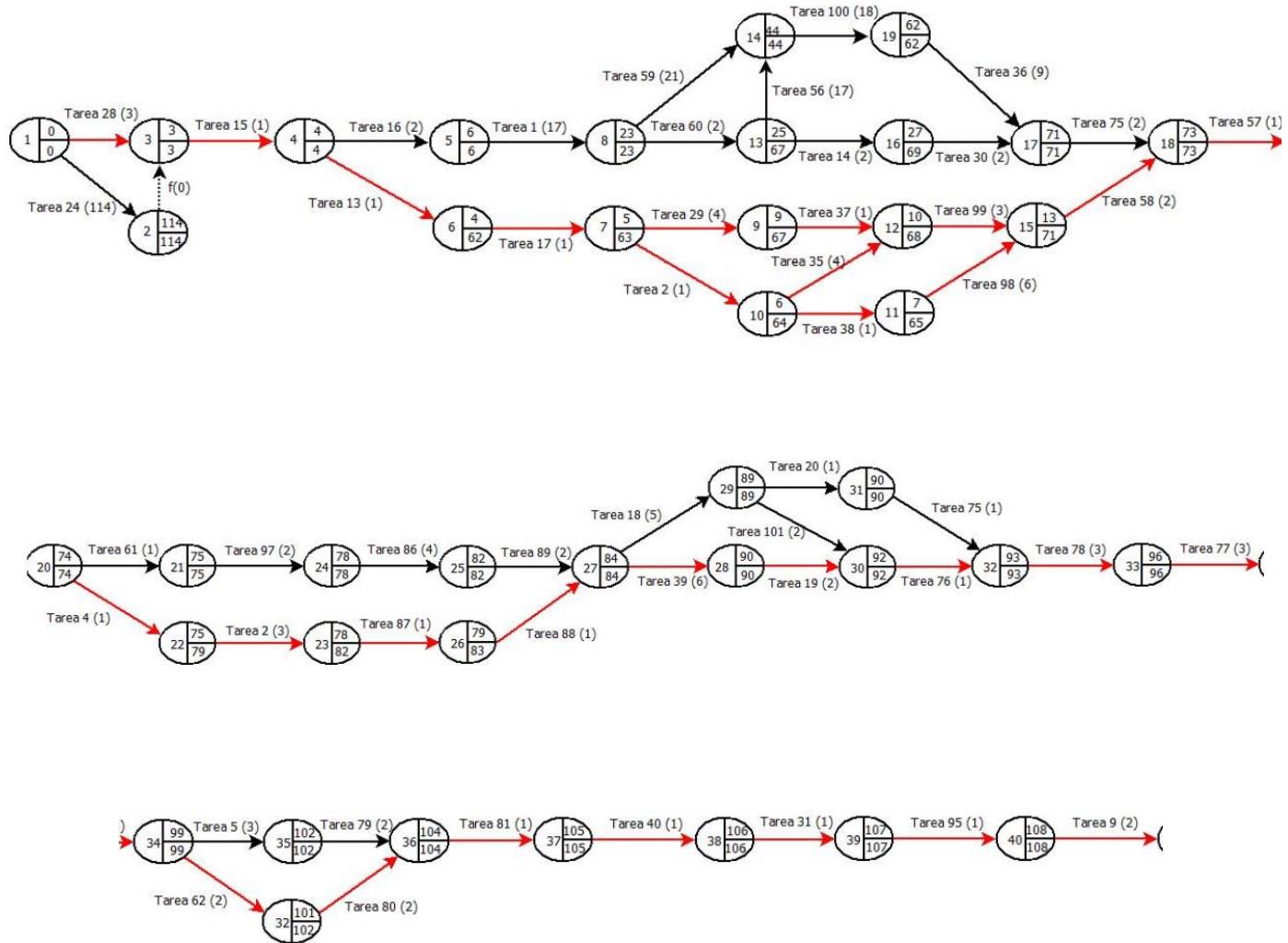
SISTEMAS OPERATIVOS

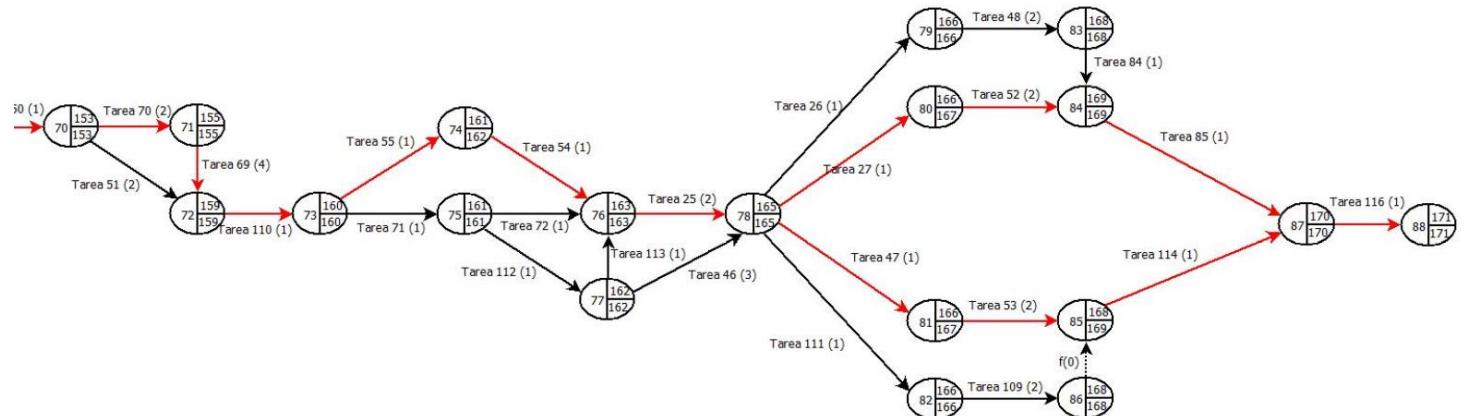
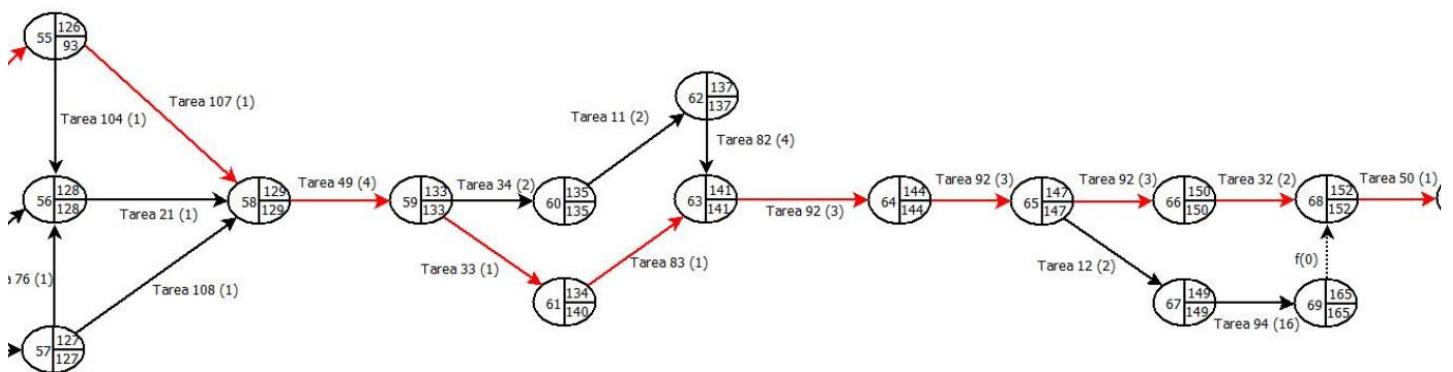
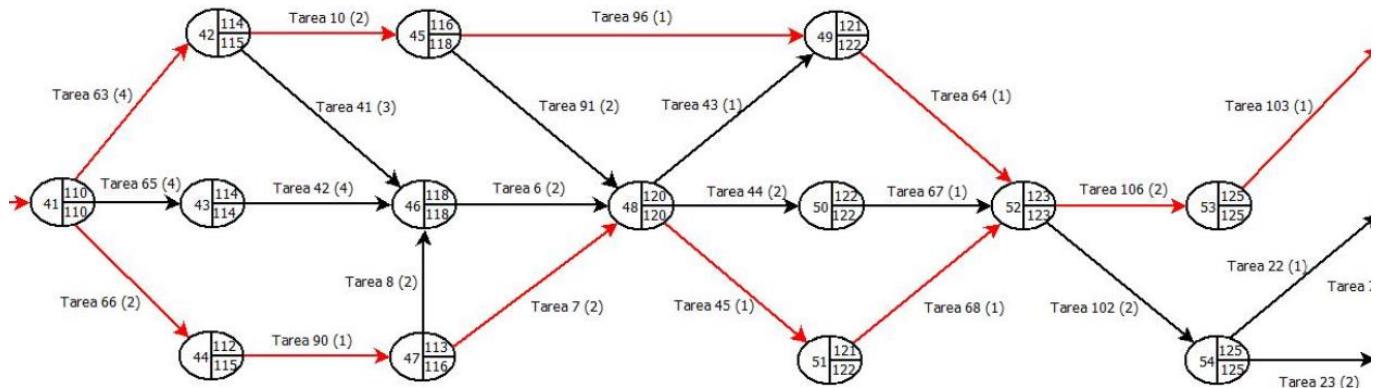


TALLER



TOTAL





Varianza Tercera Entrega:

Tareas	Fecha inicio	Duración	Tiempo optimista	Tiempo pesimista	Varianza
Deberá entregarse un ejecutable desarrollado en Visual Basic.net que contenga todas las interfaces del sistema.	19/7/2020	4	2	4	0,1111111111
El sist. deberá poner a disposición del usuario un menú principal a través del cual se accederá a todas las func.	19/7/2020	1	1	2	0,027777778
Las interfaces del sist. deberán contener todos los controles que permitan ingresar o seleccionar datos.	19/7/2020	1	1	2	0,027777778
El cod. deberá realizar las valid. de tipo de datos y tamaño max., así como la cons. entre los dist. datos ya sea o cole .	19/7/2020	2	1	3	0,1111111111
El ejecutable del sistema deberá conectarse a la base de datos en informix/Mysql	13/9/2020	1	1	3	0,1111111111
El sistema deberá permitir la totalidad de las altas bajas, y modificaciones de datos	17/9/2020	2	2	10	1,777777778
Usuarios y claves del sistema	6/10/2020	3	2	4	0,1111111111
Impresión y claves del sistema	15/10/2020	1	1	3	0,1111111111
Sistema terminado	19/10/2020	2	1	2	0,027777778
Formulación de preguntas	2/9/2020	1	1	3	0,1111111111
Transcripción de entrevista	18/9/2020	1	1	3	0,1111111111
Esquema relacional normalizado (3era forma normal)	16/7/2020	3	2	3	0,027777778
RNE	27/7/2020	1	1	2	0,027777778
Diccionario de datos	27/7/2020	1	1	4	0,25
Correcciones (Primera entrega)	13/8/2020	3	2	5	0,25
DDL (Sentencias SQL de creación de tablas y estructuras necesarias)	9/8/2020	3	2	5	0,25
Estudio de los permisos sobre BD, tablas y usuarios	22/8/2020	2	1	3	0,1111111111
Sentencias SQL de permisos a la Base de Datos.	22/8/2020	2	1	3	0,1111111111
Primera implementación física de la Base de Datos	24/8/2020	1	1	3	0,1111111111
DER	16/7/2020	3	3	4	0,027777778
Esquema relacional normalizado (3era forma normal)	16/7/2020	3	3	5	0,1111111111

RNE	27/7/2020	1	1	2	0,027777778
Diccionario de datos	27/7/2020	1	1	3	0,111111111
Consultas SQL	3/10/2020	4	2	4	0,111111111
Sugerencias para política de respaldos de las Bases de datos y Logs	5/10/2020	1	1	3	0,111111111
Anexos que se consideran necesarios	8/11/2020	1	1	2	0,027777778
Bibliografía	8/11/2020	1	1	2	0,027777778
Identificación integrantes	20/5/2020	3	2	3	0,027777778
Informe con énfasis en F.E	5/6/2020	4	2	4	0,111111111
Sugerencias	29/6/2020	2	2	3	0,027777778
trabajando en base a los requerimientos de las asignaturas del área tecnológica.	1/9/2020	1	1	3	0,111111111
"ABSTRACT"	21/10/2020	2	2	3	0,027777778
Analisis FODA (SWOT Analysis)	2/10/2020	1	1	2	0,027777778
Guion de la venta de software	1/10/2020	2	1	3	0,111111111
Relevamiento	25/7/2020	2	1	3	0,111111111
Estudio de Factibilidades	23/6/2020	6	3	6	0,25
Cálculo de métricas del proyecto	18/6/2020	3	2	5	0,25
Especificación de Requerimientos	2/7/2020	18	10	20	2,777777778
Análisis	26/7/2020	2	2	4	0,111111111
Análisis Costo-Beneficio.	21/9/2020	2	2	4	0,111111111
M.E.R. (o Diagrama de clases UML)	22/9/2020	1	1	3	0,111111111
Modelo esencial (optativo) o Modelo de dominio	23/9/2020	1	1	3	0,111111111

Diccionario de datos.	23/9/2020	1	1	3	0,1111111111
Especificación de procesos (Diagrama Nassi-Schneiderman, Warnier –Orr)	21/9/2020	2	2	5	0,25
Casos de Uso (planilla y diagrama)	23/9/2020	1	1	4	0,25
Diagrama de estados	23/9/2020	1	1	4	0,25
Carta jerárquica(Operativo)	7/11/2020	2	1	3	0,1111111111
Diagrama de clases(Operativo)	2/11/2020	1	1	2	0,027777778
Plan de Testing	7/11/2020	1	1	2	0,027777778
Manual de usuario	4/11/2020	1	1	3	0,1111111111
Manual de administrador	5/11/2020	1	1	3	0,1111111111
Anexos	8/11/2020	1	1	2	0,027777778
Agenda de reuniones (Operativo)	9/11/2020	1	1	2	0,027777778
Créditos y bibliografía	9/11/2020	1	1	2	0,027777778
Sumario Ejecutivo	1/6/2020	17	13	20	1,3611111111
Analisis de mercado	5/6/2020	1	1	3	0,1111111111
Analisis FODA	2/7/2020	2	2	4	0,1111111111
Relleno de formularios	15/7/2020	1	1	2	0,027777778
Plan estrategico	19/8/2020	3	2	4	0,1111111111
Misión	16/9/2020	2	1	3	0,1111111111
Valores	16/9/2020	2	1	3	0,1111111111
Objetivos	16/9/2020	2	1	3	0,1111111111
Plan operativo	8/9/2020	2	2	4	0,1111111111
Inversiones para la puesta en marcha del emprendimiento	16/9/2020	2	1	3	0,1111111111

Plan de comercialización	2/10/2020	2	1	3	0,1111111111
Plan financiero	14/10/2020	2	1	3	0,1111111111
Relev. Y just. Del SO a utilizar en terminales de usuarios como el servidor	10/6/2020	17	10	17	1,3611111111
Manual de instalación del SO en el servidor	15/6/2020	1	1	3	0,1111111111
Estudio de los diferentes roles del usuario del sistema	10/7/2020	2	2	4	0,1111111111
Shell Script para la creación de usuarios	13/6/2020	21	15	30	6,25
Manual de instalación de MySQL junto a sus requisitos	13/7/2020	2	1	4	0,25
Configuración del servicio SSH en el cliente y servidor	15/7/2020	1	1	3	0,1111111111
Correcciones 1er Entrega	17/8/2020	2	2	5	0,25
Configuraciones de red en las terminales y el servidor.	10/9/2020	4	2	4	0,1111111111
Archivos crontab con rutinas de backup y sus correspondientes scripts para el administrador.	19/9/2020	1	1	4	0,25
Usuarios de proyecto creados de acuerdo al estudio de roles de la primera entrega.	10/9/2020	4	2	4	0,1111111111
Configuración del archivos sudoers con sus respectivos permisos para usuarios y grupos.	11/9/2020	2	1	3	0,1111111111
Shell script con respaldo y restauración de base de datos.	19/9/2020	1	1	4	0,25
Menú para todos los Scripts	19/9/2020	1	1	3	0,1111111111
Correcciones de entregas previas	29/10/2020	4	3	4	0,0277777778
Configuración de Firewall	26/10/2020	2	2	3	0,0277777778
Control de Logs del sistema y al menos uno de los scripts (Respaldos)	3/11/2020	1	1	2	0,0277777778
Video explicativo del uso de Scripts (ejecución, etc)	4/11/2020	1	1	2	0,0277777778
Nombre del Grupo, integrantes y sus roles	25/5/2020	1	1	2	0,0277777778
Organización de la estructura de desarrollo	28/6/2020	2	2	4	0,1111111111
Desarrollar un formato de acta de reuniones formales e informales	25/5/2020	1	1	2	0,0277777778

Formularios de uso comun en Proyectos	25/5/2020	2	2	3	0,027777778
Reglas del grupo	25/5/2020	1	1	2	0,027777778
Acta de reuniones	20/5/2020	54	54	54	0
Ciclo de vida del proyecto	22/7/2020	5	3	6	0,25
Documentacion de Inicio y Planificacion	26/7/2020	2	2	5	0,25
Diagrama de Gantt	26/7/2020	1	1	2	0,027777778
Actas de reuniones (SOLO SEGUNDA ENTREGA).	4/8/2020	37	37	37	0
Documentación de gestión planificación y control	23/9/2020	1	1	3	0,111111111
Plan de contingencias (Opcional).	22/9/2020	1	1	2	0,027777778
Diagrama de Gantt (cronograma de la gestión del proyecto segunda entrega)	22/9/2020	2	1	4	0,25
Actas de reuniones (Total)	3/6/2020	114	114	114	0
Diagrama de Gantt (Completo)	6/11/2020	2	1	3	0,111111111
Documentación de cierre de proyecto	7/11/2020	1	1	2	0,027777778
Documentación de Gestión y control de Avances del Proyecto	7/11/2020	1	1	3	0,111111111
Selección, descripción técnica y fundamentación del puesto de trabajo genérico	6/6/2020	4	3	6	0,25
Selección, descripción técnica y fundamentación y cantidad de servidores	13/7/2020	9	5	10	0,694444444
Selección, descripción técnica y fundamentación del sistema operativo para los puestos de trabajo	6/6/2020	1	1	3	0,111111111
Selec., desc. Téc. y fund. del sist. Op. para el servidor	6/6/2020	1	1	3	0,111111111
Selec., desc. y funda. del esq. lógico primario de la interconexión de equipos en la inst. de salud	20/7/2020	6	4	8	0,444444444
Servicios provistos por los servidores	31/8/2020	1	1	3	0,111111111
Direccionamiento IP para cada VLAN según los usuarios del sistema	16/9/2020	3	2	4	0,111111111
interconectan los puestos y el armado de los rack con el respectivo equipamiento	14/9/2020	4	2	5	0,25

Interconecta las instituciones y un puesto de trabajo por cada sector critico	17/9/2020	1	1	3	0,1111111111
Cableado estructurado	17/9/2020	2	2	4	0,1111111111
implementacion de red Y el equipamiento informatico	17/9/2020	1	1	3	0,1111111111
Correcciones de la segunda entrega	30/10/2020	1	1	2	0,027777778
Ruteo intervlan, presentacion de tablas de ruteo	7/11/2020	1	1	2	0,027777778
Esquema logico y fisico definitivo y funcional en packet tracer, con la config. de los disp. finales y red	7/11/2020	2	2	4	0,1111111111
Contratacion de servicio de Internet para la institucion	28/9/2020	4	3	6	0,25
Descripción de la seguridad informática que implementará del equipamiento utilizado para la segur. Informatica	25/10/2020	1	1	3	0,1111111111
Descripción de los servicios montados en el servidor con su respectiva configuración	27/10/2020	2	2	4	0,1111111111
Descripción de los servicios montados en el router con su respectiva configuración	7/11/2020	2	1	3	0,1111111111
Solución de interconexión entre casa central y por lo menos 2 sucursales	7/11/2020	2	2	3	0,027777778
Solución de interconexión entre casa central y por lo menos 2 sucursales	6/11/2020	2	2	5	0,25
Solución de interconexión entre casa central y por lo menos 2 sucursales	3/11/2020	1	1	2	0,027777778

Plan de contingencias

NIVEL	IMPACTO	OCURRENCIA	SITUACIÓN	RIESGO	PRIORIDAD
1	Muy bajo	Muy baja	Rotura de la máquina	4	6
2	Bajo	Baja	Pérdida por robo	12	1
3	Medio	Media	Pérdida de respaldo y archivos	8	4
4	Alto	Alta	Problemas eléctricos	6	5
5	Catastrófico	Muy alta	Incendio	10	2
Riesgo:	Impacto * Ocurrencia	Horas perdidas		9	3
ACCIÓN				REQUIERE CONTROLES	
REPONER COMPONENTES DEFECTUOSOS				Sí (Mantenimiento)	
REPONER BIENES ROBADOS				Sí (Control de Stock)	
REALIZAR HORAS EXTRAS PARA LA RECOMPOSICIÓN DE LOS ARCHIVOS				Sí (Hacer más de un respaldo)	
INVERTIR EN UPS Y EN UNA MEJOR INSTALACIÓN ELÉCTRICA				NO	
REPONER BIENES PERDIDOS Y REALIZAR HORAS EXTRAS PARA RECUPERAR TODO EL TRABAJO PERDIDO				NO	
REALIZAR HORAS EXTRAS PARA RECUPERAR LAS HORAS PERDIDAS				Sí(Verificar tiempo de trabajo)	



Métricas

Ver Anexo (ANÁLISIS Y DISEÑO DE APLICACIONES - Anexo I: Métricas por punto de función)

Requisitos de la asignatura Análisis y Diseño de Aplicaciones: Relevamiento.

- a) Empleo de técnicas vistas durante el curso.

Preguntas Letra Proyecto

Nº1: Mapa conceptual del razonamiento clínico.

¿Qué significa las banderas rojas y amarillas?

R: Amarillas: Puede no haber un real peligro, hay una actitud negativa o catastrófica (algo exagerado).

Roja: Pueden indicar un cierto riesgo.

Preguntas ADA

Nº1: 1. Relevamiento.

a. Empleo de técnicas vistas durante el curso.

b. Elaboración de formularios apropiados para la tarea anterior.

¿Este punto se refiere a los cuestionarios, entrevistas, etc. a realizar al cliente?

R: Preguntas al que nos contrata para desarrollar el software.

Preguntas Formación empresarial

Nº1: 3. Actividad y Giro.

R: Hay que explicar a qué se dedica la empresa de ustedes o sea cual es el negocio que van a desarrollar.

7. Contrato Social.

R: Tiene que ver con la forma jurídica de la empresa (si es SA o SRL).

8. Formularios de Registros frente al Estado (formularios completos: DGI, RUT, PLANILLA,

DE TRABAJO, BPS, y BSE).

R: Tema a dar en clase.

9. Documentos Comerciales.

R: Tiene que ver con la documentación comercial de la empresa (boleta, factura, etc.)

Preguntas Sociología

Nº1: 2. Planteo de preguntas de investigación. formular una pregunta general y dos específicas.

¿A qué se refiere este punto? ¿Se podría detallar?

3. D fundamentación de la importancia del proyecto. relevancia de acuerdo al contexto actual y a la orientación elegida.

¿Al decir "De acuerdo al contexto" se refiere al contexto académico o al contexto por el que estamos pasando (Pandemia)?

R: Las preguntas de investigación son una forma de "encaminar" el trabajo de proyectos en la medida en que ayuda a aclarar los objetivos para los cuales se quiere realizar el proyecto. Cuando se menciona "general" o "específica" se refiere a que se parte de una idea general y luego se describe alguna más detallada.

"Contexto actual" se refiere a la situación sanitaria y como ha afectado los procesos de enseñanza y aprendizaje.



Preguntas SO

Nº1: A la hora de crear el shell script solamente hay que ingresar un usuario o estaría bueno tener varias opciones como, por ejemplo: 1-Crear un usuario nuevo, 2-Crear un usuario con un directorio home distinto, 3- Crear un usuario con un específico User ID, 4-Crear un usuario con un específico Group ID, 5- Añadir usuario sin directorio HOME y así muchas más.

R: Debería tener menú y opciones para agregar usuarios, grupo y asignación de usuarios a grupos.

Nº2: ¿A la hora de agregar usuarios en el script debería estar solamente la opción de agregar un usuario por ejemplo (useradd nombre) o varias opciones como agregar un usuario con un directorio principal distinto y otras más?

R: Con opciones es lo esperado.

Preguntas Ingles

Nº1: ¿Los nombres de las materias los dejamos en español o los cambiamos a inglés?

R: En inglés

Preguntas Proyecto

Nº1: Se supone que el Gantt es sobre las tres entregas de todas las materias, al utilizar la plantilla otorgada por Excel, la misma nos solicita un porcentaje de la tarea cumplido, así como el inicio real y la duración real. ¿Esos datos los dejamos en blanco o cómo hacemos? Ya que son datos desconocidos.

Con respecto a los requerimientos no funcionales, el apartado de Ayuda on-line. ¿La ayuda on-line sería una página o una ventana que emerge del mismo programa?

R: La ayuda online puede ser ambas cosas. Lo ideal es que sea una ventana emergente.

Con respecto al Gantt, el porcentaje es con respecto a la entrega o al HITO que hayan definido



Nº2: 2. Organización de la estructura de desarrollo (Opcional).

8. Documentación de Inicio y Planificación.

¿A qué se refieren estos puntos? ¿Se podrían detallar?

R: Elección del ciclo de vida y generar una estructura de trabajo dentro del grupo.

Se comenzó a hacer desde que presentamos el grupo y hacemos el acta de reuniones y el reglamento del grupo (reglas). Gantt, Pert, Perfil de recursos, Métricas.

Preguntas Taller

Nº1: Tenemos ciertas dudas acerca de la elección de servidores, en base a que requisitos nos basamos para la elección del mismo y la cantidad.

R: Respecto a los servidores les recomiendo que busquen marcas como HP, Lenovo, etc ya armados para su implementación. Luego argumenten su elección y establezcan un mantenimiento del mismo; servicio de fábrica, soporte, repuestos, etc.

b) Preguntas provenientes de Formación empresarial.

Pregunta 1: ¿Cómo empezaste a trabajar en el sector informático?

Pregunta 2: ¿Cómo funciona el desarrollo de software en los Estados Unidos?

Pregunta 3: ¿Qué tipo de Software es el más demandado por las compañías?

Pregunta 4: ¿Qué elementos deberías de tener en cuenta para fijar el precio de un Software?

Preguntas destinadas a la U.D.E.L.A.R, Facultad de Medicina e Instituciones Médicas.

¿Qué preguntas le serían de utilidad a un médico en formularios orientados al dolor, fiebre, malestar u otro síntoma para determinar un diagnóstico o tratamiento para el paciente?

Estudio de factibilidades.

Técnico: Se utilizarán sistemas operativos que ofrecen estabilidad, seguridad, velocidad y escalabilidad tales como: Windows 10 Enterprise 2019 LTSC para las terminales de trabajo y CentOS 8 para servidores.

A su vez el software se desarrollará en lenguaje .NET el cual tiene características como: Programación orientada a objetos, fácil administración y desarrollo.

Como Sistema Gestor de Base de Datos seleccionamos MySQL v8.0 que cuenta con características tales como: Seguridad, soporta gran cantidad de datos, estable, escalable, gran disponibilidad en diferentes sistemas operativos y conexión vía ODBC.

Características del Hardware Disponible para el desarrollo:

EQUIPO	ELEMENTO	CAPACIDAD
PC 1	Memoria RAM:	16 GB
	Disco duro:	HDD 1 TB / SSD 225 GB
	Procesador:	i7 7700k 4.2GHz 4 núcleos x64 bits
	CD ROM/DVD ROM:	No
PC 2	Memoria RAM:	8 GB
	Disco duro:	SSD 500 GB
	Procesador:	i5 8400 2.80GHz 6 núcleos x64 bits
	CD ROM/DVD ROM:	Sí
PC 3	Memoria RAM:	16 GB
	Disco duro:	HDD 1 TB
	Procesador:	i5 8400 2.8GHz 6 núcleos x64 bits
	CD ROM/DVD ROM:	No
PC 4	Memoria RAM:	16 GB
	Disco duro:	HDD 250 GB / HDD 320 GB
	Procesador:	A8-3870 3.3GHz 4 núcleos x64 bits
	CD ROM/DVD ROM:	Sí
LAPTOP 1	Memoria RAM:	3 GB
	Disco duro:	HDD 320 GB
	Procesador:	Core 2 Duo t7200 2.0 GHz 2 núcleos x64 bits
	CD ROM/DVD ROM:	Sí
LAPTOP 2	Memoria RAM:	4 GB
	Disco duro:	HDD 250GB
	Procesador:	i5-2410M 2.9GHz 2 núcleos x64 bits
	CD ROM/DVD ROM:	Sí

Experiencia y Conocimiento del Equipo de Desarrollo:

Somos 4 integrantes en formación académica que estamos adquiriendo conocimientos informáticos y a su vez contamos con el apoyo y observación de distintos docentes.

**Legal:**

Estudio de licenciamiento:

Microsoft Visual Studio: Dicho Software de desarrollo tiene licencia propietaria de Microsoft pero la licencia será suministrada directamente por el cliente.

MySQL: El cliente ya posee licencia, pero nosotros para poder desarrollar, comercializar e implementar MySQL en nuestro sistema deberemos comprar una licencia de tipo Standard que el costo sería de USD 2,000.

Linux CentOS 8: Este será el sistema operativo a utilizar en los servidores de la institución médica, el mismo sistema operativo es de uso libre y gratuito.

Windows 10 Enterprise 2019 LTSC: Windows 10 Enterprise será el sistema operativo a utilizar en las terminales de trabajo, se va a obtener licencias tipo VLK (Volume Licensing Key) debido al gran número de equipos.

Operativo: Contamos con los suficientes recursos humanos, es decir personal capacitados para realizar el testeo, instalación, mantenimiento y operación del sistema informático a desarrollar.

Económico: El sistema será una buena inversión para la empresa, pero los costos de inversión serán bastante elevados ya que hay que destinar presupuesto a las licencias de desarrollo y también de los sistemas operativos, así como destinar presupuesto a la instalación de equipos en la institución y a los recursos humanos involucrados para desarrollar, operar y mantener el sistema. De igual manera los beneficios serán mayores que los costos a destinar y el retorno de la inversión será total.

Cálculo de métricas por punto de función y Mc Cabe:

Ver Anexos (ANÁLISIS Y DISEÑO DE
APLICACIONES - Anexo I: Métricas)



Especificación de requerimientos

1. Introducción

1.1 Propósito del Documento

El propósito del documento será documentar los requerimientos funcionales que el cliente entiende que el sistema debe poseer y los requerimientos no funcionales que son aquellos requerimientos necesarios para que el sistema funcione adecuadamente.

1.2 Propósito del Sistema

El Propósito del Sistema es proporcionar un Software de Bitácora Médica para así poder ayudar a los Médicos de la Institución a controlar la salud de sus pacientes más fácilmente.

1.3 Situación actual

Actualmente no hay en el mercado un sistema que permita predecir mediante porcentajes enfermedades basándose en los síntomas que el paciente presenta.

1.4 Usuarios del Sistema

Médico, podrá acceder a:

- Ingresar datos
- Crear formularios
- Consultar información

Auxiliar, podrá acceder a:

- Ingresar datos
- Crear formularios

Paciente, podrá acceder a:

- Consultar información de diagnóstico
- Consultar resultado de análisis

Administrador, podrá acceder a:

- Agregar Médicos
- Agregar Auxiliares
- Ingresar información a la Base de Datos
 - Modificar permisos de acceso

1.5 Alcance y Limitaciones

1.5.1 Alcance

- El software no estará relacionado con la ficha médica del paciente, sino con una consulta en particular.
- El software está diseñado particularmente para mutualistas de salud.

1.5.2 Limitaciones

- Si bien aún no es un predictor, en algún momento podría llegar a serlo en el caso de que el sistema se auto alimente (feed-back).
- No se utilizará en consultorios privados.



2. Requerimientos

2.1 Requerimientos Funcionales

2.1.1 Ingreso al sistema

El sistema desplegará una ventana llamada “LOGIN” la cual se accede mediante una barra de menú y posibilitará ingresar al sistema mediante un usuario y una contraseña. La misma tendrá una etiqueta de texto en el centro a la izquierda con el texto “Usuario”. A la derecha de la etiqueta “Usuario” habrá un cuadro de texto para llenar el usuario, abajo de la etiqueta “Usuario”, habrá otra etiqueta con el texto “Contraseña”, justamente a la derecha de la misma se ubicará un cuadro de texto para llenar la contraseña.

Para el usuario solo se permitirá ingresar números y letras, además la longitud del mismo debe de ser desde 4 caracteres hasta 16 caracteres.

La contraseña solo permitirá ingresar números y letras, además la longitud debe de ser desde 8 caracteres hasta 24 caracteres.

2.1.2 Administración médicos y auxiliares

El sistema desplegará una ventana llamada “Administrar médicos y auxiliares” la cual solo podrá acceder el administrador y se accede mediante una barra de menú y posibilitará elegir un tipo *ComboBox* (médico/auxiliar) seguidos de datos genéricos como:

Primer nombre *OBLIGATORIO, String* TextBox

Segundo nombre *OPCIONAL, String* TextBox

Primer apellido *OBLIGATORIO, String* TextBox

Segundo apellido *OBLIGATORIO, String* TextBox

Cédula *OBLIGATORIO, Integer* TextBox

Edad *OBLIGATORIO, Integer* TextBox

Dirección *OBLIGATORIO, String* TextBox

Teléfono *Opcional, Integer* TextBox

Contraseña *OBLIGATORIO, String* TextBox ‘Se debe cumplir la misma validación que la contraseña al ingresar al sistema.’

Y si es médico permite ingresar la

Especialidad *OBLIGATORIO, String*.

Debajo de estos datos, se encuentran ubicados 3 botones de manera horizontal los cuales cumplen la función de alta, baja y modificación de los médicos y auxiliares.

2.1.3 Ingreso información del paciente

El sistema desplegará una ventana llamada

“Mantenimiento” la cual solo podrá acceder el auxiliar y/o médico y se accede mediante una barra de menú.

La misma posibilitará al auxiliar/médico ingresar información del paciente consultante. Como, por ejemplo:

- Primer nombre *OBLIGATORIO, String* TextBox
- Segundo nombre *OPCIONAL, String* TextBox
- Primer apellido *OBLIGATORIO, String* TextBox
- Segundo apellido *OBLIGATORIO, String* TextBox
- Cédula *OBLIGATORIO, Integer* TextBox
- Edad *OBLIGATORIO, Integer* TextBox
- Sexo (Masculino (M), Femenino (F)) *OBLIGATORIO, char* ComboBox
- Estado Civil (Soltero/a, Casado/a, Divorciado/a, Viudo/a) *OBLIGATORIO, String* ComboBox
- Dirección *OBLIGATORIO, String* TextBox
- Nombre del acompañante *OPCIONAL, String* TextBox

2.1.4 Ingreso información entrevista inicial

El sistema desplegará una ventana llamada “Entrevista inicial”, en la cual solo podrán ingresar información los médicos y/o auxiliares y se accede mediante una barra de menú.

En la misma ventana se ingresarán datos de inspección general como, por ejemplo:

- Estado de conciencia *OBLIGATORIO, String* TextBox
- Grado de colaboración *OBLIGATORIO, String* TextBox
- Movilidad *OBLIGATORIO, String* TextBox
- Actitud (Apatía, coma, excitación) *OBLIGATORIO, String* ComboBox

Estado de nutrición (Obesidad, anorexia, caquexia)
OBLIGATORIO, String ComboBox

Grado de hidratación *OBLIGATORIO, String* TextBox
Y también permitirá ingresar constantes vitales, como, por ejemplo:

- Temperatura *OBLIGATORIO, Double* TextBox
- Pulso (Fuerte, débil, filiforme) *OBLIGATORIO, String* ComboBox

Frecuencia cardíaca *OBLIGATORIO, String* TextBox
Frecuencia respiratoria *OBLIGATORIO, String* TextBox
Además el sistema también registrará antecedentes personales del paciente, así como enfermedades familiares.

El sistema registrará la Fiebre (El sistema sugerirá la utilización de otro formulario orientado a la fiebre), el Dolor (el sistema sugerirá la utilización de otro formulario orientado al dolor y se utilizará una imagen de órganos y zonas reflejas), el Malestar (el sistema sugerirá la utilización de otro formulario orientado a malestar) y Otro.

2.1.5 Prediagnóstico/diagnóstico

El sistema desplegará una ventana llamada “Prediagnóstico/diagnóstico” en la cual el sistema podrá sugerir posibles diagnósticos a realizar a partir de la información ingresada en la “Entrevista inicial”.

Posibles diagnósticos:

- Diagnóstico clínico o individual
- Diagnóstico de certeza
- Diagnóstico diferencia
- Diagnóstico etiológico
- Diagnóstico genérico
- Diagnóstico heroico
- Diagnóstico lesional, anatómico o topográfico
- Diagnóstico nosológico
- Diagnóstico patogénico
- Diagnóstico presuntivo
- Diagnóstico sindrómico y funcional
- Diagnóstico sintomático

Además, el médico podrá registrar un diagnóstico a partir de su conocimiento.

También, el médico y/o auxiliar podrá realizar la solicitud de un análisis clínico y a su vez el ingreso del mismo.

2.1.6 Ingreso tratamientos

El sistema desplegará una ventana llamada “Plan” en la cual se podrán ingresar los tratamientos a seguir una vez obtenido el diagnóstico, así como tener un registro de controles

2.1.7 ABM Formularios de entrevista

El sistema desplegará una ventana llamada “Creación, eliminación y modificación de formularios” la misma posibilitará al médico y/o auxiliar realizar la alta, baja y modificación de formularios, ya sea a solicitud del médico o de elaboración propia.

2.1.8 Consultar información paciente

El sistema desplegará una ventana llamada “Información paciente” la misma posibilitará solamente al médico consultar sobre la información del paciente (datos personales e historial clínico en la institución)

2.1.9 Consultar información de diagnóstico

El sistema desplegará una ventana llamada “Información de diagnóstico”, la misma posibilitará solamente al paciente consultar información de los diagnósticos que se realizó en la institución médica.

2.1.10 Consultar resultado del análisis clínico



El sistema desplegará una ventana llamada “Resultado de análisis”, la misma posibilitará solamente al paciente consultar sobre el resultado de los análisis realizados en la institución médica.

2.1.11 Salida del sistema

El sistema tendrá una opción llamada “Salir” la cual se accede mediante una barra de menú, la misma posibilitará a todos los usuarios cerrar y salir del sistema.

2.2 Requerimientos No Funcionales

2.2.1 Performance

Para el software a desarrollar la performance no será significativa, pero si medianamente importante ya que el sistema deberá tener una respuesta de pequeñas fracciones de segundo.

2.2.2 Seguridad y Control de Acceso

En el servidor se realizarán los siguientes cambios en la configuración del archivo de configuración ubicado en /etc/ssh/sshd_config

Cambio de puerto

```
#  
Port 3899  
#AddressFamily any  
#ListenAddress 0.0.0.0  
#ListenAddress ::
```

Por defecto, viene con el puerto 22 establecido, al cambiar este dato estaremos mejorando significativamente la seguridad ya que no será un puerto estándar.

Permitir login de root

```
# Authentication:  
  
#LoginGraceTime 2m  
PermitRootLogin no  
#StrictModes yes  
#MaxAuthTries 6  
#MaxSessions 10
```

Por defecto, el root tiene permitido el login, al cambiar este dato a “no”, se mejorará la seguridad ya que la mayoría de intentos de ataques/ingresos al sistema se realizan con el usuario root.



No se podrá ingresar como usuario root, pero un usuario común podrá tomar la identidad del mismo con el comando su root.

Deshabilitar la autenticación por contraseña

```
# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
PasswordAuthentication yes
```

Cambiando esta simple opción la seguridad en el sistema aumentará ya que solo se ingresará mediante claves asimétricas permitidas en el archivo ~/.ssh/authorized_keys y además permitirá automatizar las tareas en un futuro ya que no pedirá la contraseña para realizar alguna acción en el sistema.

Además, se instalarán paquetes como SELinux que es una característica de seguridad de Linux que provee una variedad de políticas de seguridad, incluyendo controles de acceso.

Y se instalará FirewallID que es un controlador frontend para la tabla iptables que se usa para implementar reglas de tráfico de red persistentes.

También en la base de datos se creará una tabla en la que cada usuario tendrá un rol y solamente podrá acceder a las funciones que ese rol ofrece.

2.2.3 Integración con otros sistemas NO APLICA

2.2.4 Interfaz con el usuario

Fue pensada para ser amigable con el usuario y entendible, así como para que sea atractiva.

Los botones serán blancos con letras negras
El color de fondo será gris

2.2.5 Ayuda on-line

Se desplegará una ventana emergente con las preguntas más frecuentes y un mail de contacto para asistencia.

2.2.6 Requerimientos Internacionales, Legales y otros

- Estándar internacional IEEE 802.3 (Cableado estructurado Ethernet)
- Estándar IEEE 802.11 (WI-FI)
- Estándar ANSI/TIA/EIA-568-B
- Norma IEEE-STD-830-1998



3. Glosario

Feed-back: Retroalimentación.

Performance: Rendimiento del sistema.

Puerto: Se refiere a un punto de conexión entre redes en forma de nodo en una red en la que los paquetes de datos se dirigen hacia la dirección de destino final.

Frontend: Frontend es la parte de un sitio web que interactúa con los usuarios.

String: Un tipo de dato el cual solo permite ingresar una cadena de texto.

Integer: Un tipo de dato el cual solo permite ingresar números enteros.

TextBox: Cuadro de texto.

ComboBox: Lista desplegable de elementos.

Char: Un tipo de dato el cual solo permite valores de un carácter.

Análisis F.O.D.A

Internas a la empresa

- Fortalezas: Gran ambiente de trabajo:

Existe un vínculo agradable entre los integrantes de la empresa.

Equipos informáticos óptimos para trabajo:

Contamos con equipos aptos para realizar las tareas de desarrollo de Software.

Extensa formación en informática:

Todos los integrantes de la empresa cuentan con una formación académica sobre Informática.

Conocimientos sobre herramientas de comunicación:

Sabemos cómo manejar herramientas como Zoom, Discord, Skype y otras más para lograr una comunicación optima entre integrantes o integrantes-clientes.

- Debilidades: Escaso personal de trabajo:

Contamos con poco personal de trabajo lo cual nos generaría dificultades a la hora de desarrollar un Software ya que nos llevaría más tiempo de lo debido.

Limitado conocimiento acerca de la administración de una empresa:

Carecemos de poca experiencia para organizar administrativamente una empresa.

Pocos recursos económicos:

Debido a que somos una empresa que recién se inicia en el mercado contamos con escasos recursos económicos iniciales.

Pocas oportunidades para obtener un préstamo de un banco:

Como somos una empresa chica tenemos pocas oportunidades de que un banco nos ceda un préstamo.

Externas a la empresa

- Oportunidades: Contexto actual (Pandemia):

La empresa puede desarrollar Software relacionado con la salud que facilite la situación actual.

Sector de comercio informático en constante crecimiento:

El sector informático se encuentra en constante crecimiento y cada vez se solicita a más gente para trabajar, de hecho, este sector cuenta con desempleo negativo.

Ciertos clientes que están a la espera de una empresa que innove en el desarrollo de Software:

Hay clientes que esperan un cambio en los productos del mercado, es decir un Software que innove y que sea distinto a los demás.

- Amenazas: Mucha competencia:

Numerosas empresas consolidadas y reconocidas por su trayectoria en el mercado con potencial de atraer clientes.

Diferentes clientes que están pasando por una crisis económica:

Debido a la pandemia se ven afectados por una crisis económica y no tienen dinero para adquirir nuestro producto y/o servicio.

Aparición de nuevas tecnologías, productos y/o servicios:

No solamente posibles empresas que estén desarrollando un producto similar al nuestro, sino también desarrolladores amateurs que ofrecen nuevos productos y/o servicios.

Possible no aceptación a algún cambio u innovación en el mercado por parte de los clientes, lo que provoca que el producto y/o servicio no tenga éxito:

Así como se encuentran clientes que están a la espera de un cambio, también hay clientes que prefieren lo clásico y no apuestan por el cambio.

Análisis Costo-Beneficio

SOLUCIÓN 1 - COSTOS POR ÚNICA VEZ

CANTIDAD	DESCRIPCIÓN	IMPORTE (U\$S)
63	HP Compaq 6200 PRO - Core i5 2400 - 250GB HDD - 4GB RAM (Puestos de trabajo genérico)	\$ 15.561,00
1	Servidor Dell PowerEdge R440 - Intel Xeon Silver 41102 - 4x4TB HDD - 16 GB RAM RDIMMs (Servidor Base de Datos) c/Teclado c/Mouse c/Monitor	\$ 5.667,77
1	Servidor Dell PowerEdge R440 - Intel Xeon Silver 41102 - 4x4TB HDD - 16 GB RAM RDIMMs (Servidor de Respaldo)	\$ 5.649,43
1	Servidor Dell PowerEdge R440 - Intel Xeon Silver 41102 - 4x480GB SSD - 16 GB RAM RDIMMs (Servidor de Web/Mail)	\$ 5.269,43
1	Router Firewall fw10408	\$ 584,65
1	Router TP Link Er5120	\$ 205,57
4	Switch TP Link tl sg1024de	\$ 484,76
2	Switch TP Link t1500-28PST	\$ 780,00
1	Rack de 42U LinkBasic	\$ 470,00
5	Rack de 6U	\$ 495,00
1	Pachera 24 puertos CAT6e	\$ 39,34
1	KVM Level One KVM-0422 4 equipos usb	\$ 112,00
6	UPS Forza Atlas FDC-2012R-I	\$ 5.741,82
5	UPS Forza SI-1012ul-c	\$ 753,15
6	Bobina Cable de Red UTP CAT6 305m	\$ 781,89
1	Bobina Cable de Red Nexxt SFTP CAT6a 305m	\$ 375,00
125	Ducto PVC 2x1x200cm	\$ 143,18
117	Bandeja portacables galvanizada 15x6,5x300cm	\$ 3.510,00
5	Bandeja portacables union tee horizontal 15x6,5cm	\$ 125,00
4	Conector ficha RJ45 Bolsax100 Unidades Nexxt CAT6	\$ 128,00
4	Botitas / Capuchón para RJ45 x 100 Unidades	\$ 32,00
65	Cupla RJ45 Hembra Hembra Nexxt CAT5e	\$ 331,50
63	Licencia VLK Windows 10 Enterprise 2019 LTSC	\$ 21.287,70
1	Licencia MySQL Standard Edition	\$ 2.000,00
TOTAL:		\$ 70.528,19

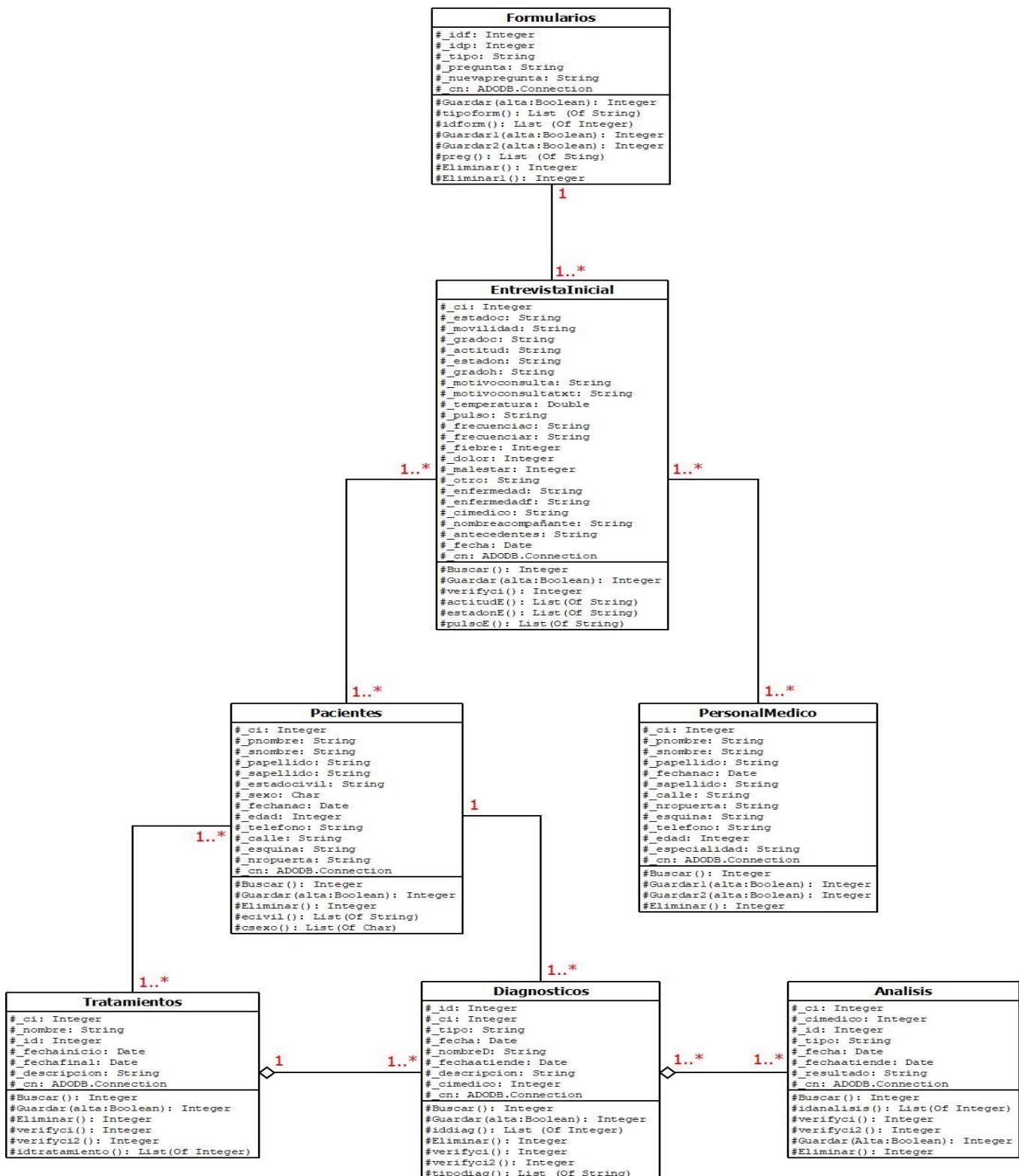
SOLUCIÓN 1 - COSTOS RECURRENTES

CANTIDAD	DESCRIPCIÓN	IMPORTE (U\$S)
1	Plan fibra Ultra 500mb/50mb Antel	\$ 78,00
2	Salario empleados	\$ 9.842,00
TOTAL:		\$ 9.920,00

FLUJO DE EFECTIVO

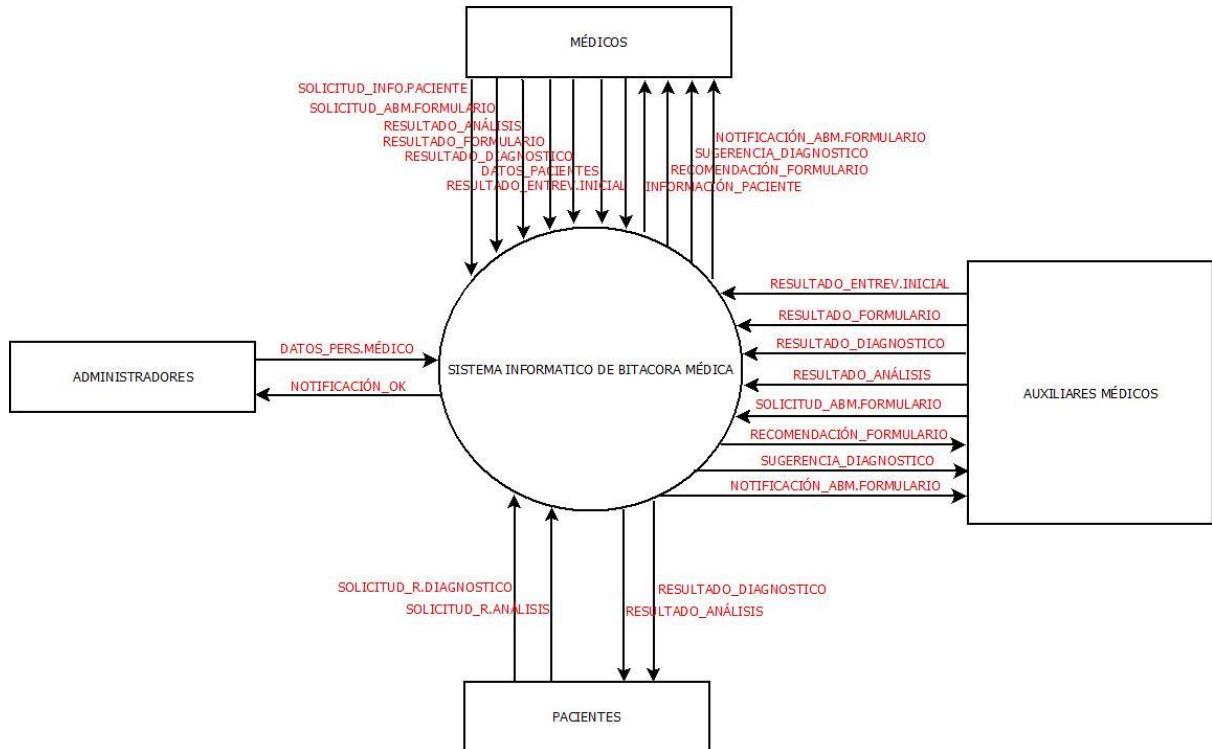
	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	TOTAL
Beneficios	\$ 10.200,00	\$ 10.200,00	\$ 10.200,00	\$ 10.200,00	\$ 91.307,39	\$ 132.107,39
Costos	\$ 9.920,00	\$ 9.920,00	\$ 9.920,00	\$ 9.920,00	\$ 80.448,19	\$ 120.128,19
Efectivo Neto	\$ 280,00	\$ 280,00	\$ 280,00	\$ 280,00	\$ 10.859,20	\$ 11.979,20
Ganancias	\$ 280,00	\$ 560,00	\$ 840,00	\$ 1.120,00	\$ 12.057,20	\$ 14.857,20

Diagrama de Clases UML

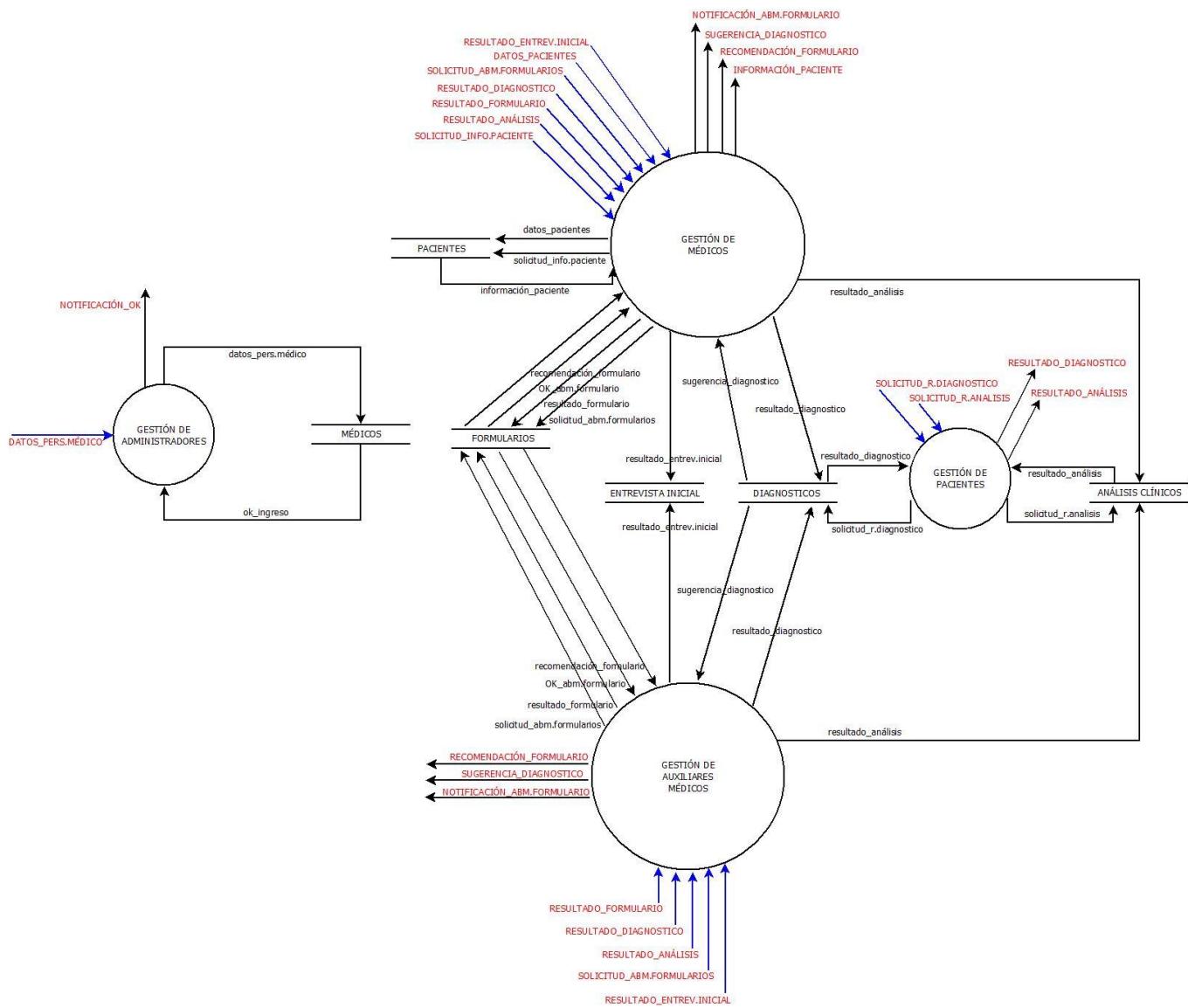


Modelo Esencial

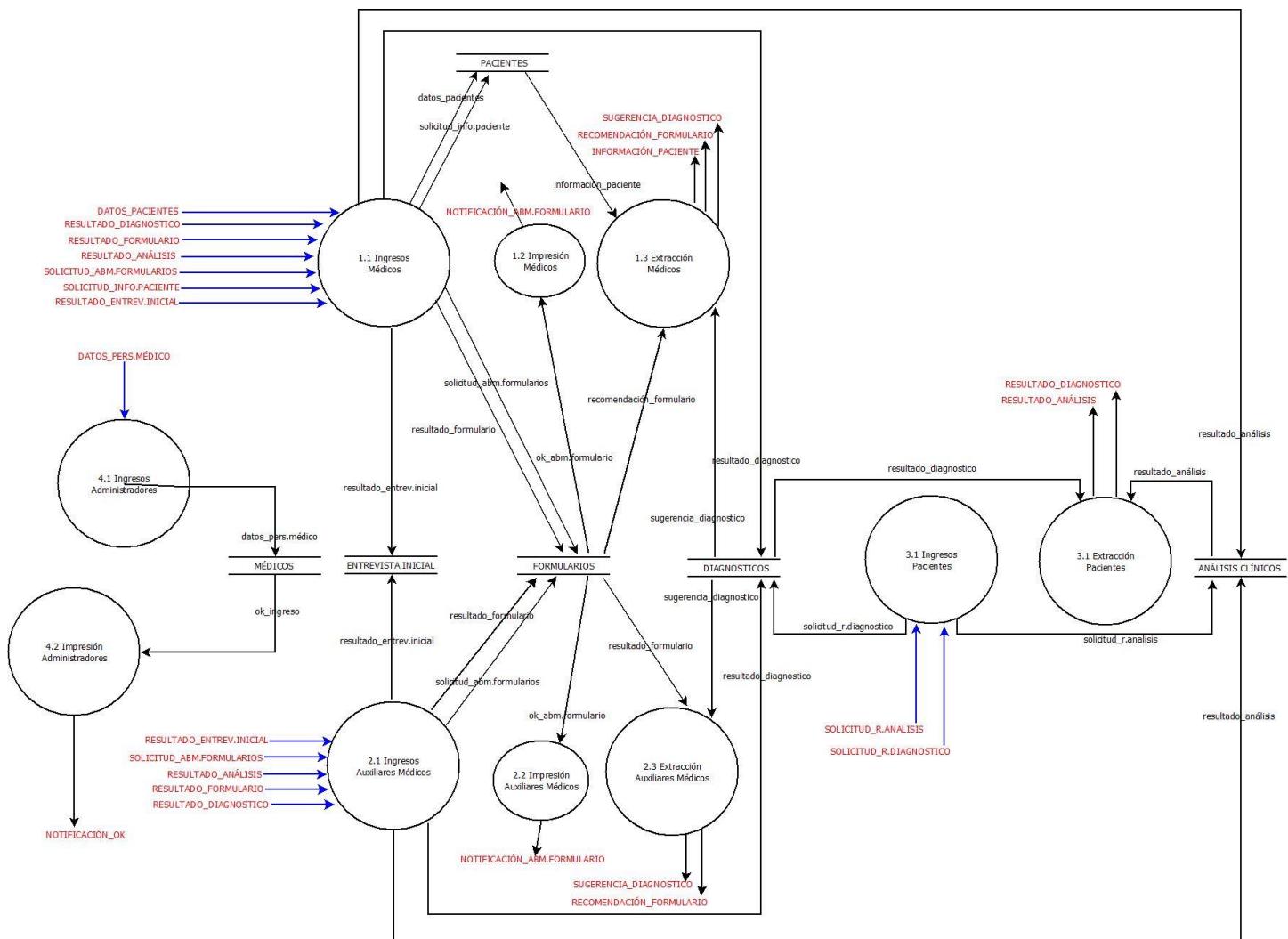
Nivel 0



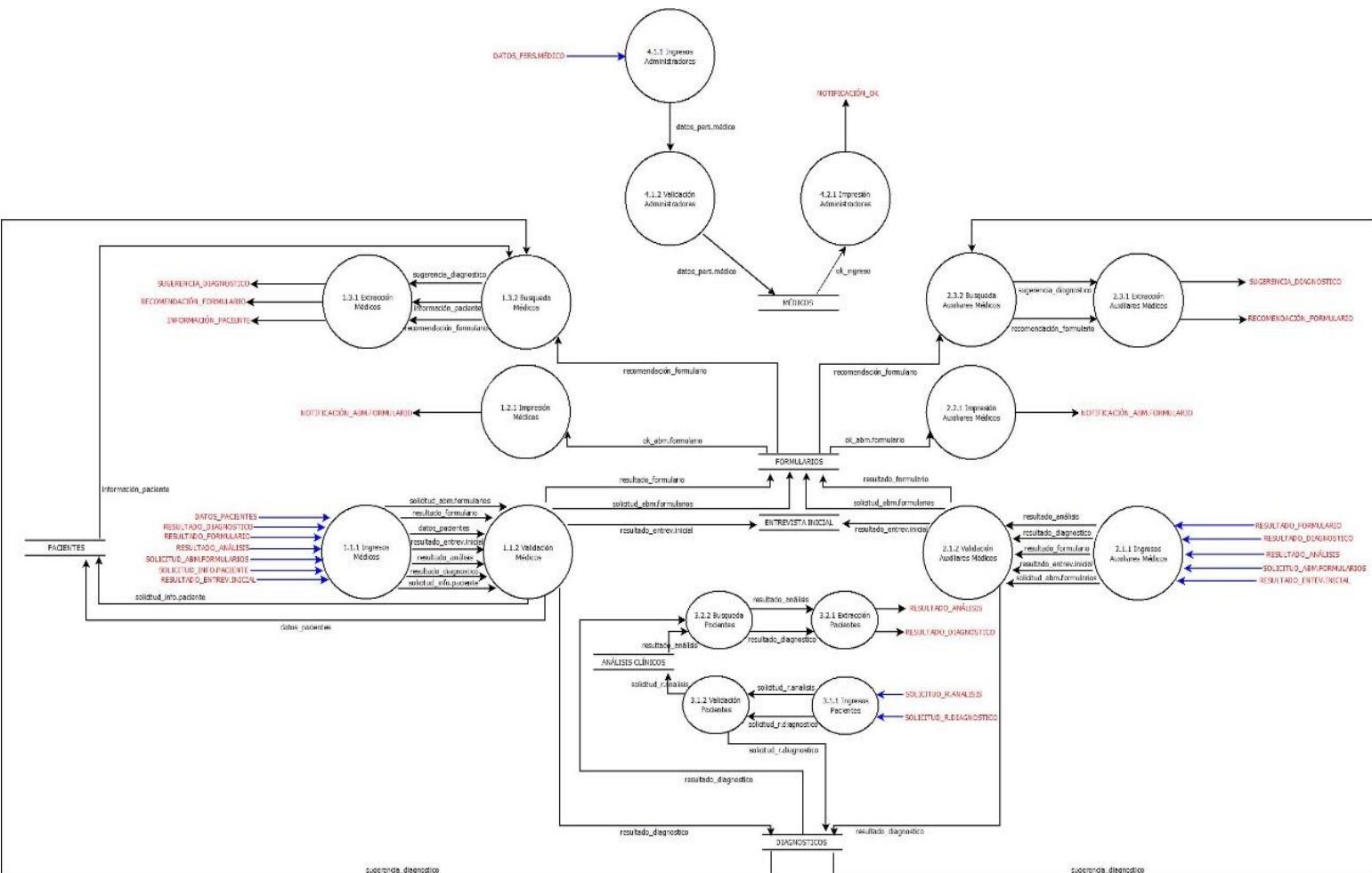
Nivel 1



Nivel 2



Nivel 3



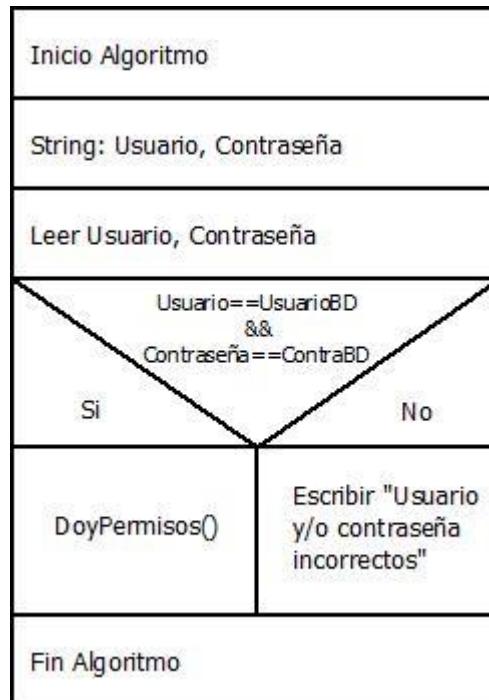
Diccionario de datos

Campo	Descripción
DATOS_PACIENTE	Datos personales del paciente que son ingresados al sistema
DATOS_PERS.MÉDICO	Datos del personal médico ingresados por el administrador al sistema
INFORMACIÓN_PACIENTE	Información del paciente en la institución médica que es consultada por el médico
NOTIFICACIÓN_ABM.FORMULARIOS	Notificación que avisa si una alta, baja o modificación de un formulario se hizo con éxito
NOTIFICACIÓN_OK	Notificación que avisa que los datos del personal médico se ingresaron correctamente
RECOMENDACIÓN_FORMULARIO	Recomendación de un formulario, proveniente de los datos ingresados en el formulario
RESULTADO_ANÁLISIS	Resultado con todos los datos de un análisis clínico realizado por un paciente en la institución
RESULTADO_DIAGNOSTICO	Resultado con todos los datos de un diagnóstico clínico realizado por un paciente en la institución
RESULTADO_ENTREV.INICIAL	Resultado con todos los datos de una entrevista realizada entre un paciente y el personal médico en la institución médica
RESULTADO_FORMULARIO	Resultado con todos los datos de un formulario llenado por el personal médico con las respuestas del paciente
SOLICITUD_ABM.FORMULARIO	Solicitud que realiza el personal médico para realizar una alta, baja o modificación de un formulario
SOLICITUD_INFO.PACIENTE	Solicitud que realiza el médico para obtener la información del paciente en la institución médica
SOLICITUD_R.ANALISIS	Solicitud que realiza el paciente para obtener el resultado de sus análisis realizados en la institución médica
SOLICITUD_R.DIAGNOSTICO	Solicitud que realiza el paciente para obtener el resultado de sus diagnósticos realizados en la institución médica
SUGERENCIA_DIAGNOSTICO	Sugerencia de un diagnóstico, proveniente de los datos ingresados en el formulario

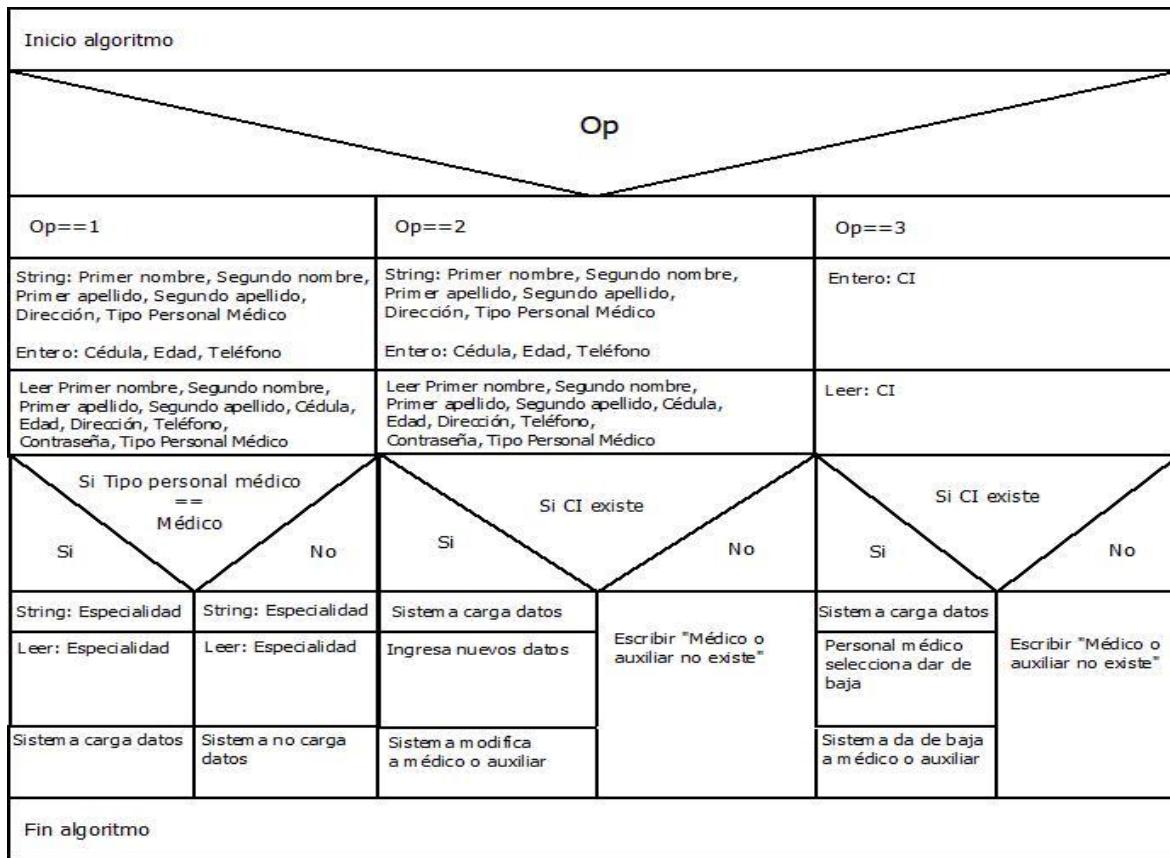
Especificación de procesos

Diagrama Nassi-Schneiderman

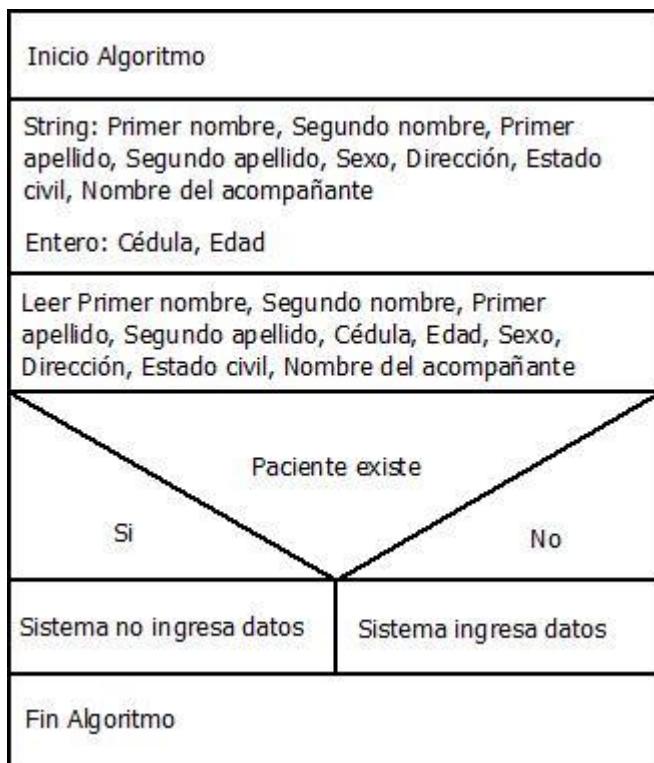
Nº1



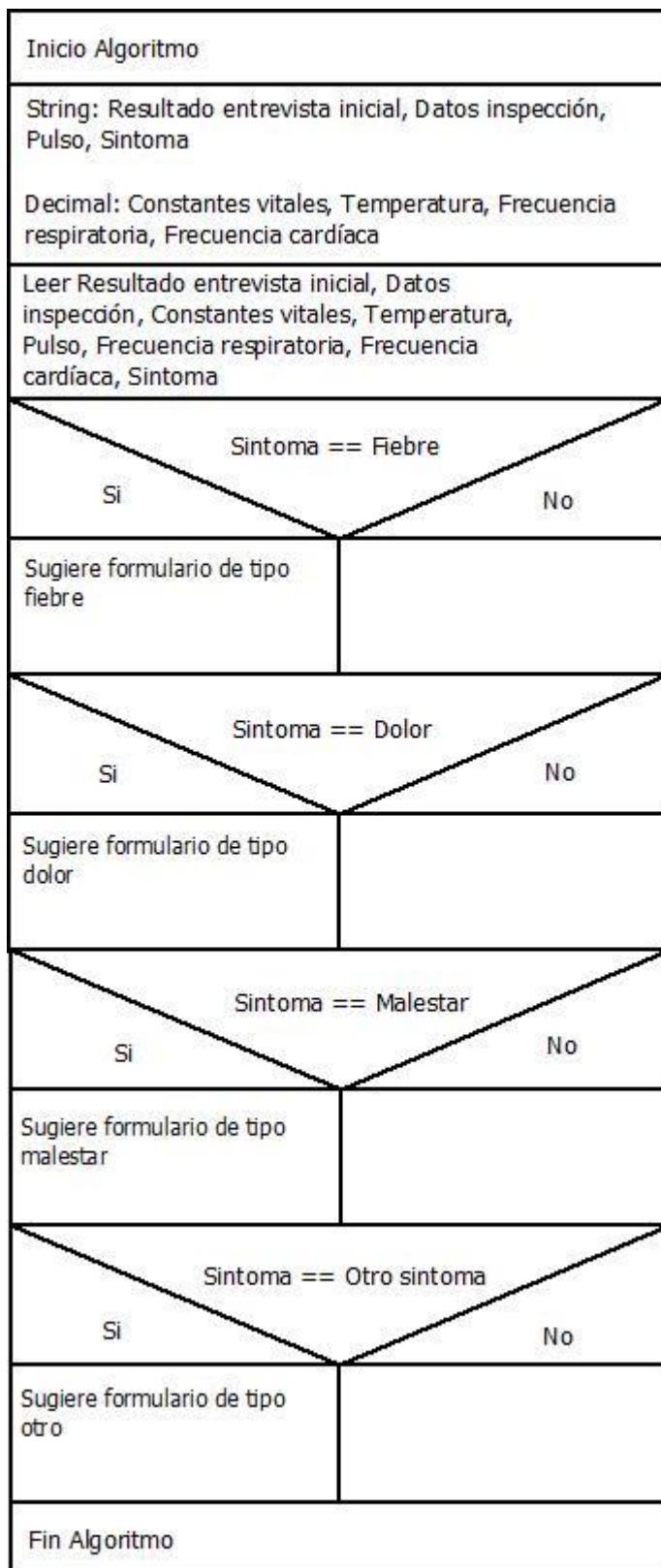
Nº2



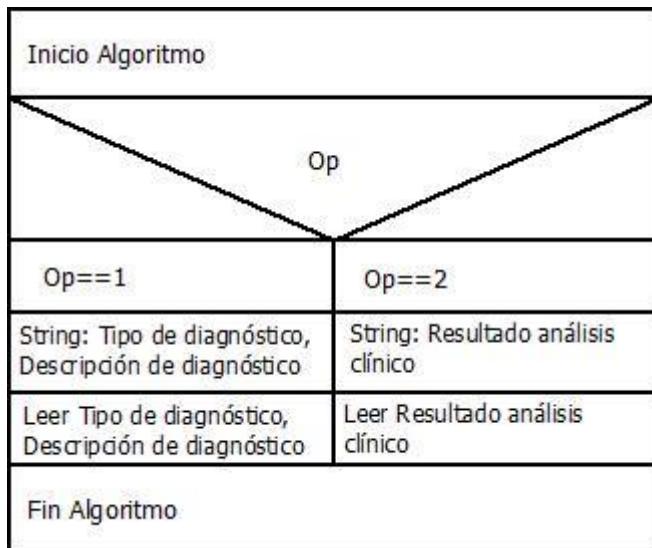
Nº3



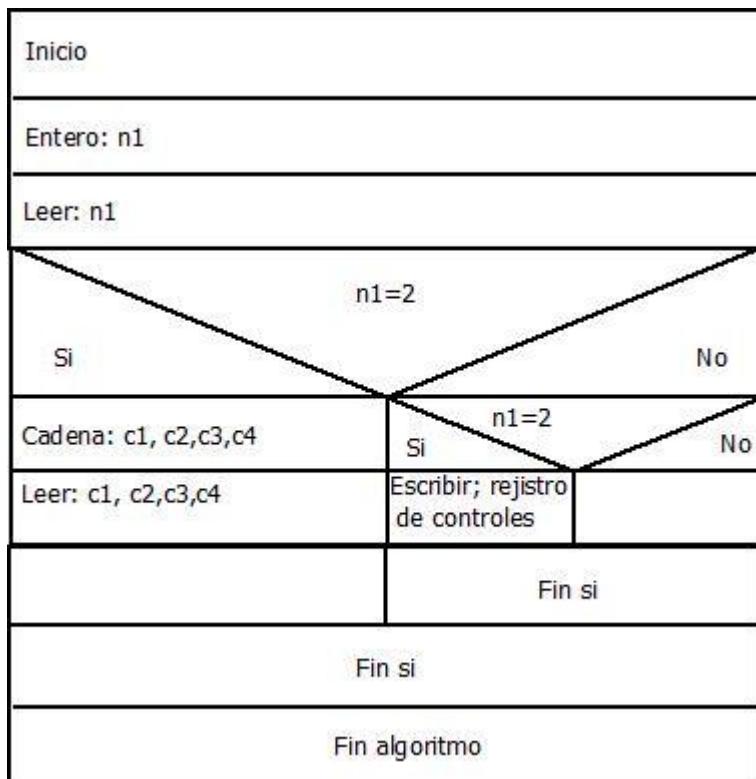
Nº4



Nº5



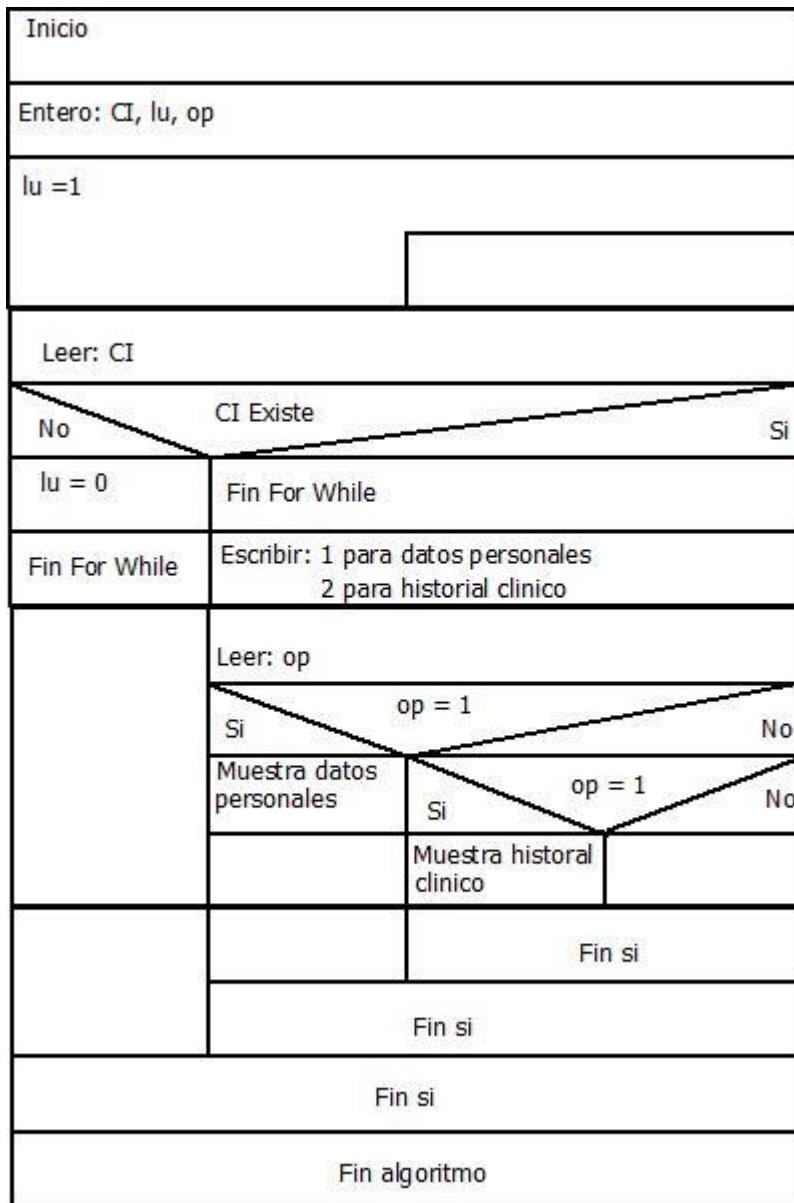
Nº6



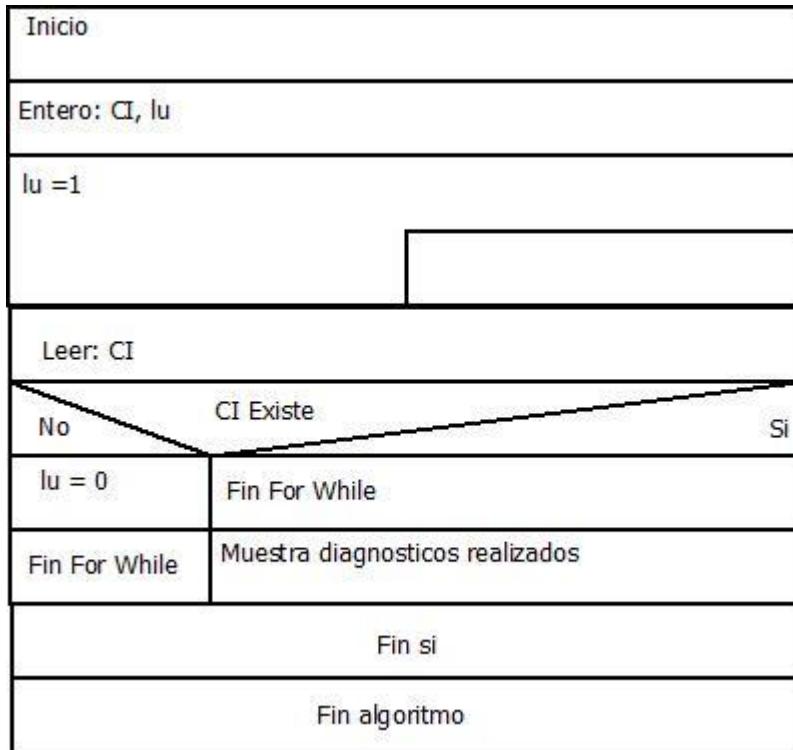
Nº7

Inicio					
Entero: bandera, op					
Leer: op					
No	op = 1	Si			
No	op = 2	Si	Cadena: Tipo		
No	op = 3	Si	Cadena: Tpo		
	Cadena: Tpo	Leer: Tipo	Leer: Tipo		
	Leer: Tipo	Carga formularios por tipo	Bandera = 0		
	Carga formularios por tipo	Cadena: Pregunta	Cadena: Pregunta		
	Entero: ID Formulario	leer: Pregunta	leer: Pregunta		
	Leer: ID Formulario		Escribir: "?Desea ingresar otra pregunta?(1)o 2 cerrar formulario?"		
	Elimina formulario		Leer: op		
Fin si		No	op = 1		
Fin si		Si			
		Bandera=1	Bandera=0		
		Crea formulario			
		Fin si			
		Fin For While			
	Fin si				
	Fin algoritmo				

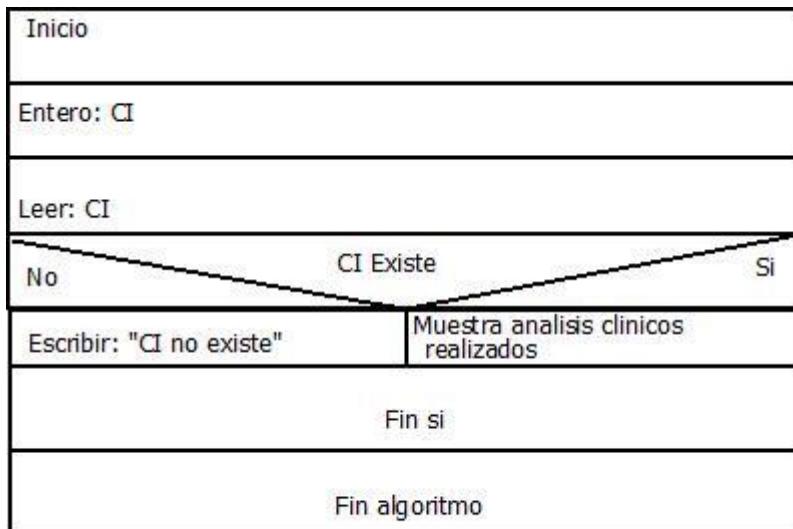
Nº8



Nº9



Nº10

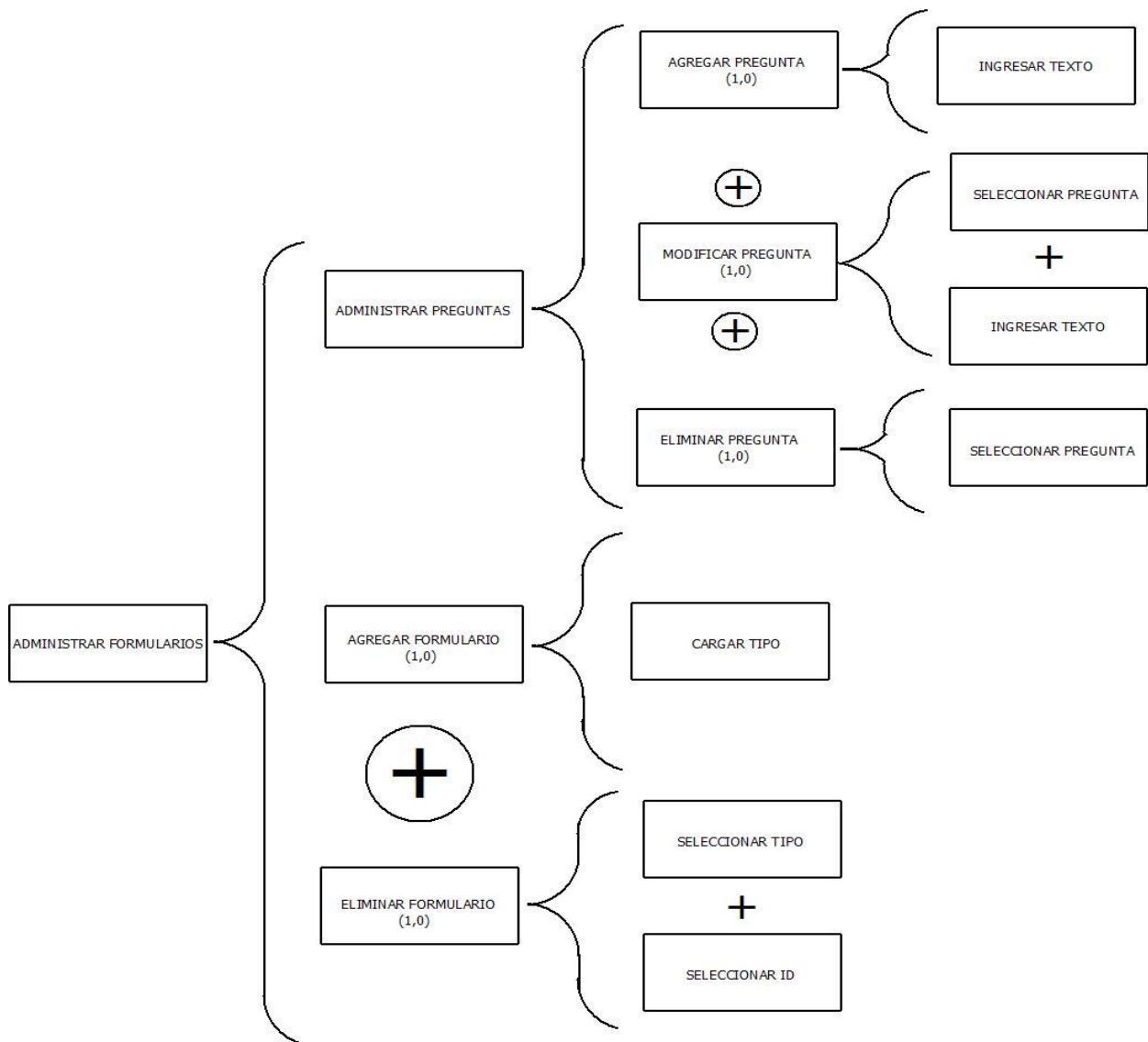




Nº11

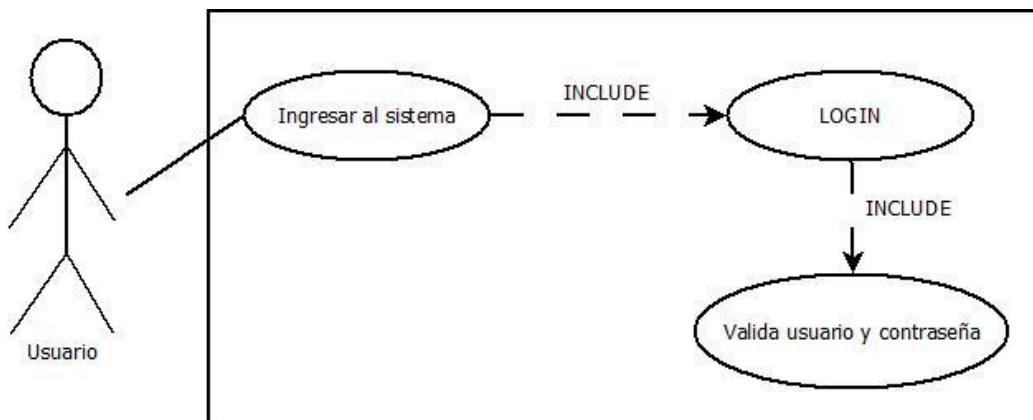
Inicio Algoritmo
Seleccionar "salir del sistema"
Fin Algoritmo

Diagrama Warnier-Orr

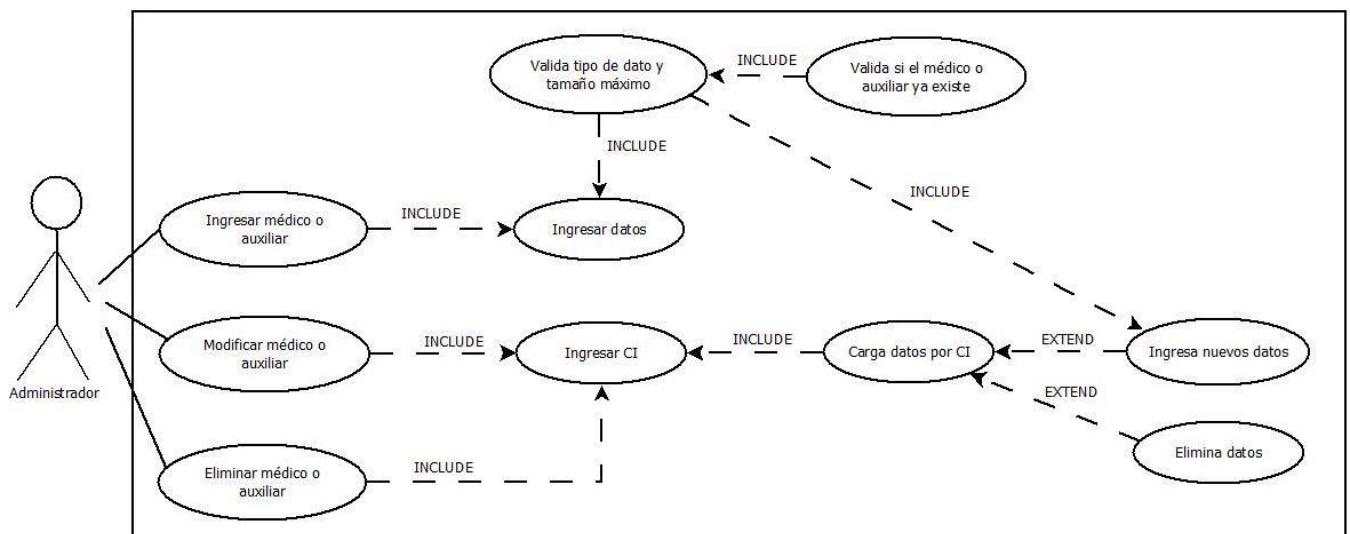


Casos de uso (Planilla y diagrama)

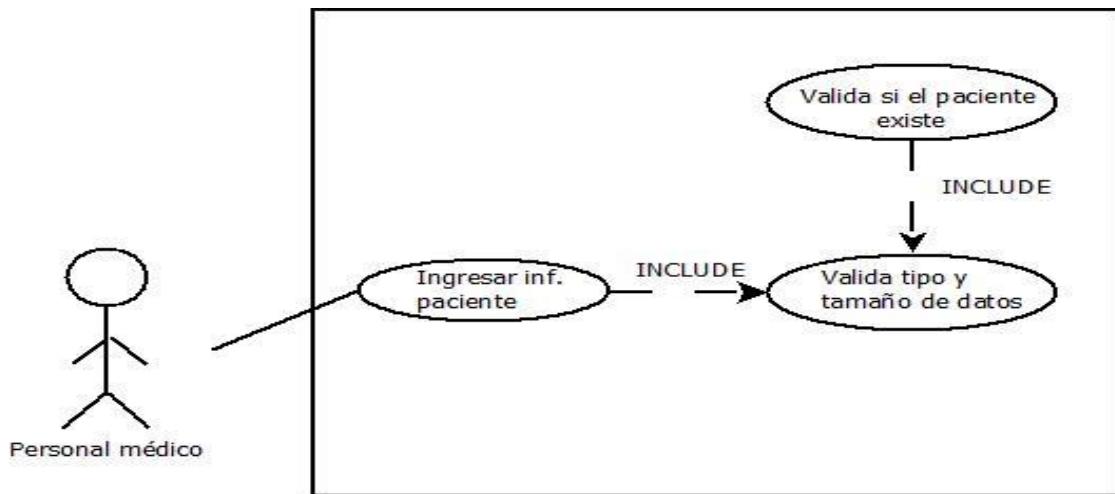
Caso N°1



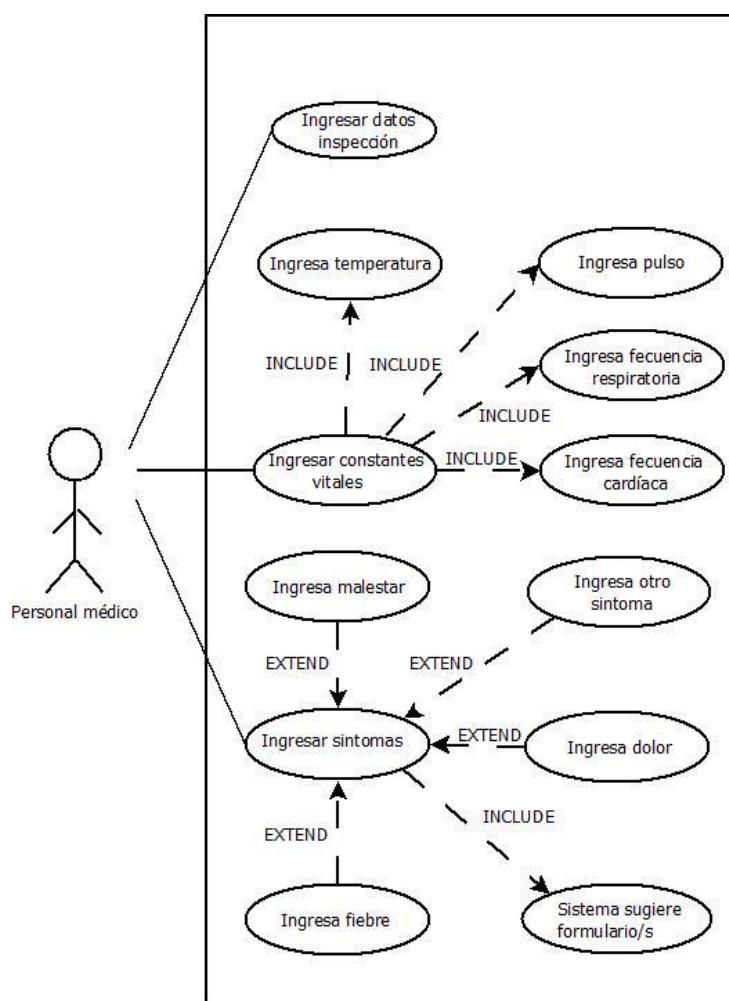
Caso N°2



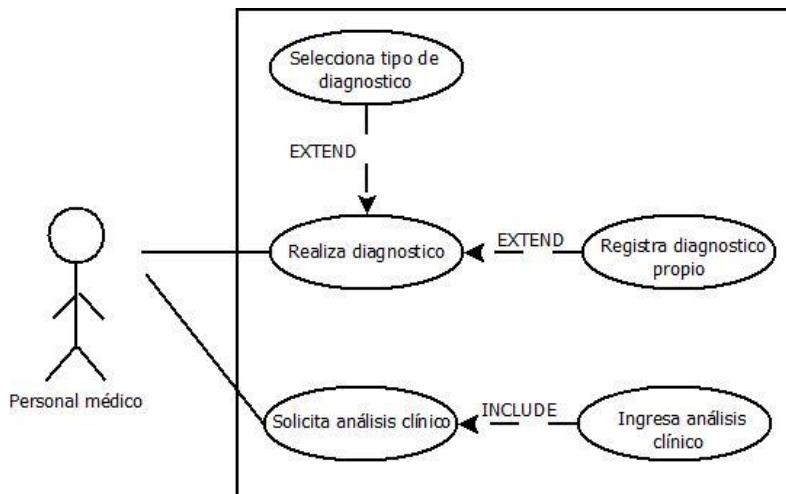
Caso N°3



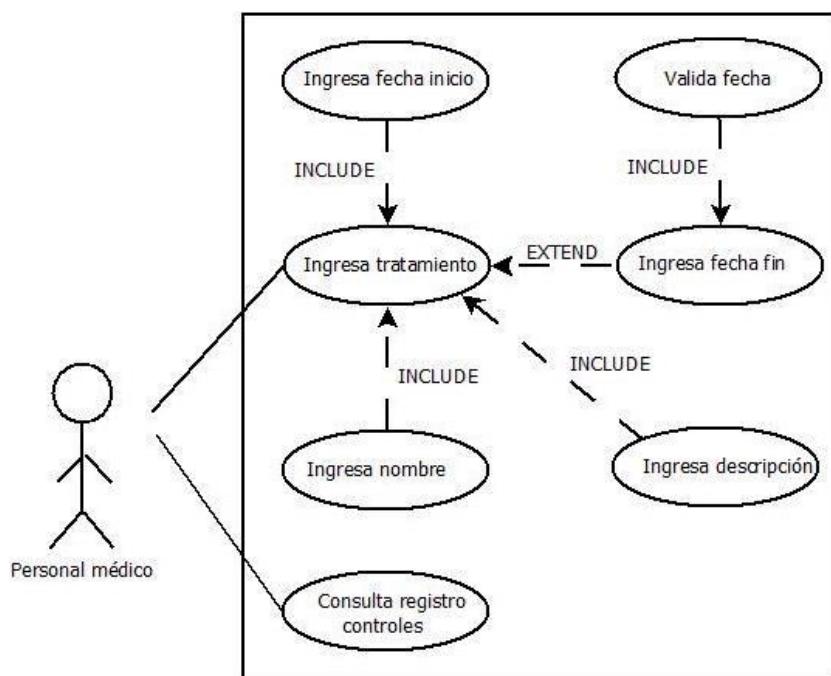
Caso N°4



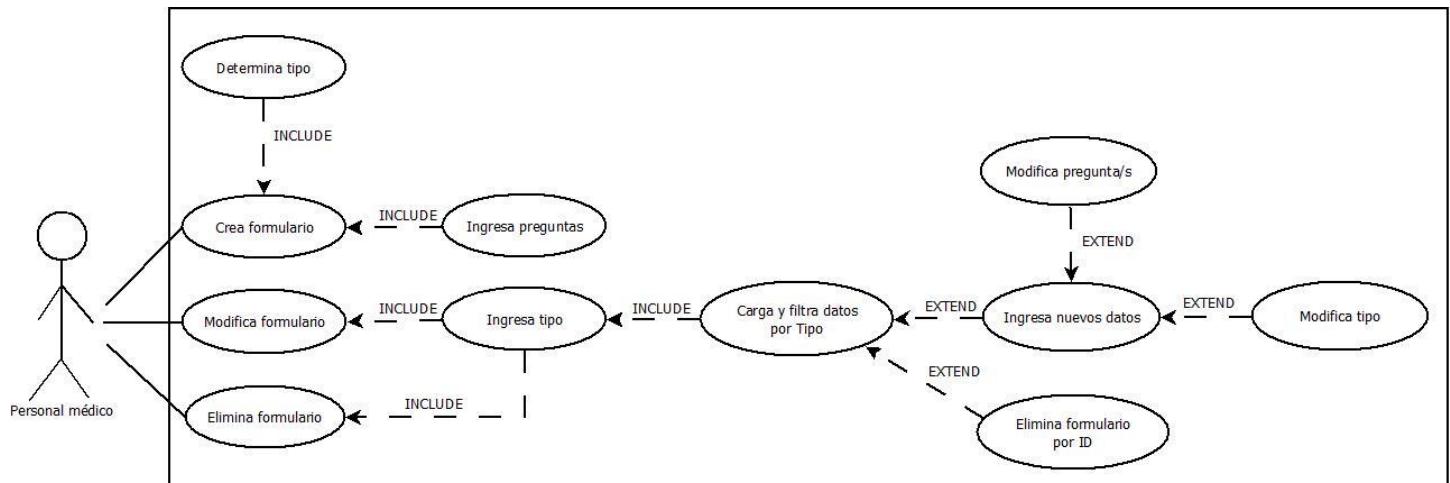
Caso N°5



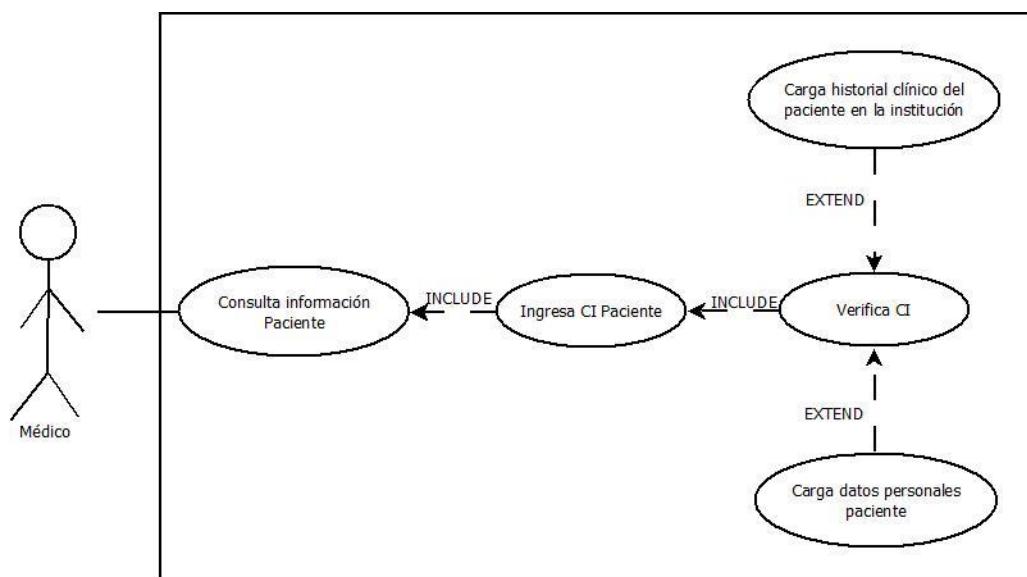
Caso N°6



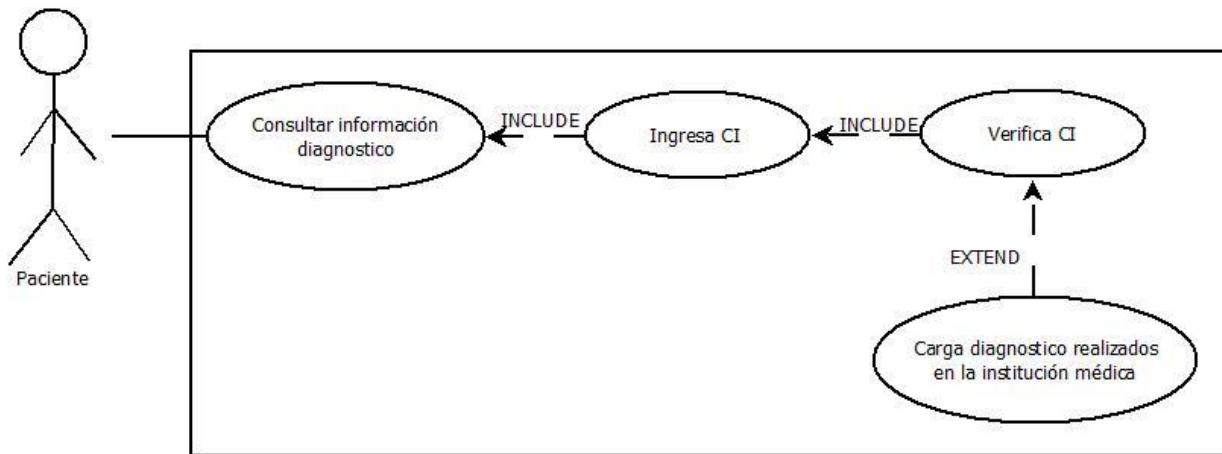
Caso N° 7



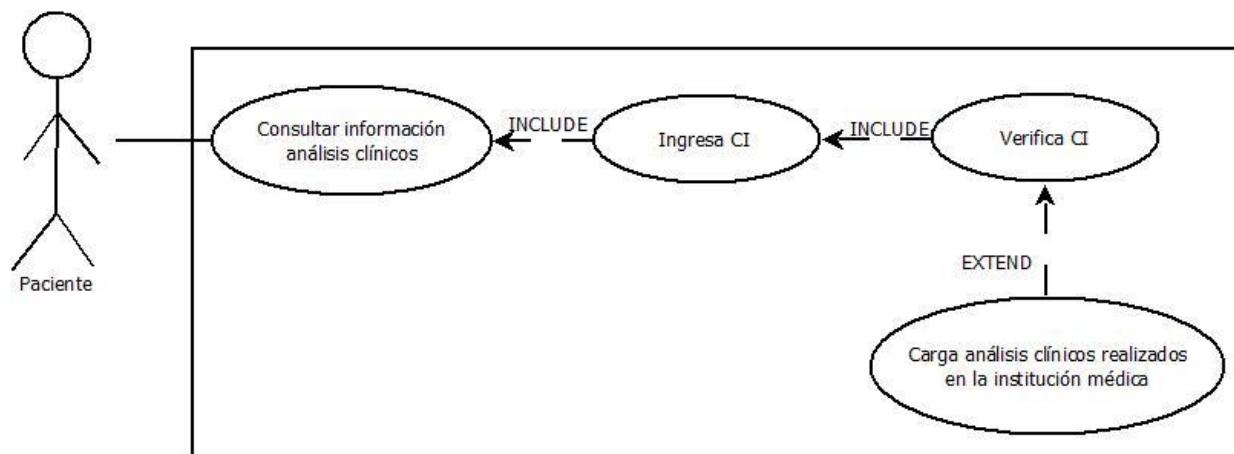
Caso N°8



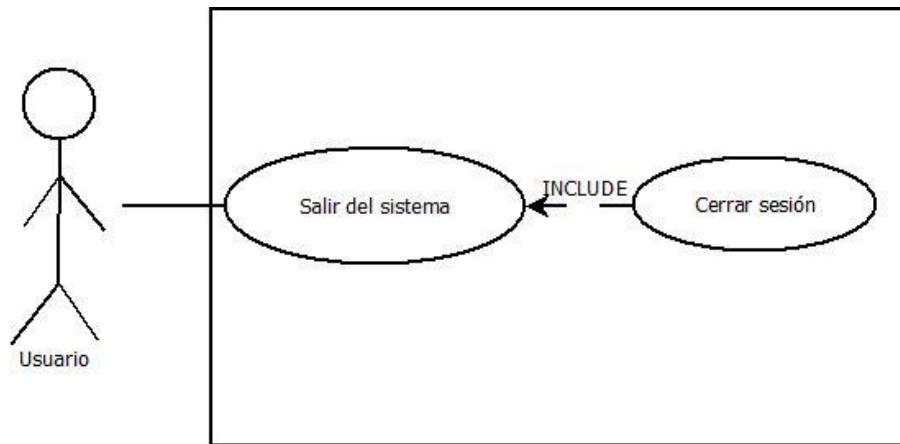
Caso N°9



Caso N°10



Caso N°11



Caso N°1

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Ingreso al sistema
1	R.F. 2.1
ACTOR:	Usuario del sistema
Estado inicial del Sistema:	Funcionando, con funciones bloqueadas.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Usuario ingresa nombre usuario	RA1. Retorna a RP1
2- Usuario ingresa contraseña	
3-Sistema valida usuario y contraseña (RA1)	
4-Usuario ingresa al sistema	
Estado final del Sistema:	Funcionando, con funciones desbloqueadas.
Comentarios:	

Caso N°2

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Ingresar médico o auxiliar
2	R.F. 2.2
ACTOR:	Administrador del sistema
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Administrador ingresa datos del médico o auxiliar	RA1. Retorna a RP1
2-Sistema valida datos personales (RA1)	
3-Sistema valida si el médico o auxiliar ya existe (RA1)	
4- El médico o auxiliar se da de alta en la base de datos	
Estado final del Sistema:	Funcionando.
Comentarios:	Esta ruta principal es solamente para el ingreso del médico o auxiliar.

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Modificar médico o auxiliar
2	R.F. 2.2
ACTOR:	Administrador del sistema
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Administrador ingresa CI del médico o auxiliar.	RA1. Retorna a RP1 "NO SE ENCONTRO A LA PERSONA."
2-Sistema carga datos en pantalla y busca por CI (RA1)	RA2. Retorna a RP3.
3-Administrador ingresa nuevos datos del médico o auxiliar (RA2)	
4- Los datos ingresados se modifican por los anteriores.	
Estado final del Sistema:	Funcionando.
Comentarios:	Esta ruta principal es solamente para la modificación del médico o auxiliar.

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Baja médico o auxiliar
2	R.F. 2.2
ACTOR:	Administrador del sistema
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Administrador ingresa CI del médico o auxiliar.	RA1. Retorna a RP1 "NO SE ENCONTRO A LA PERSONA."
2-Sistema carga datos en pantalla y busca por CI (RA1)	
3-Administrador confirma eliminación de datos.	
4- Los datos del médico o auxiliar se eliminan.	
Estado final del Sistema:	Funcionando.
Comentarios:	Esta ruta principal es solamente para la baja del médico o auxiliar.

Caso N°3

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Ingresar información del paciente
3	R.F. 2.3
ACTOR:	Personal médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Personal médico ingresa la información del paciente consultante.	RA1. Retorna a RP1 RA2. Carga datos del paciente
2- Sistema valida tipo y tamaño de datos. (RA1)	
3-Sistema valida si el paciente existe (RA2)	
4-Se guarda la información del paciente.	
Estado final del Sistema:	Funcionando.
Comentarios:	

Caso N°4

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Ingresar resultado de la entrevista inicial
4	R.F. 2.4
ACTOR:	Personal médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Personal médico ingresa datos de inspección general.	RA1. Si el sistema no sugiere formularios relacionados con el síntoma, el personal médico puede elaborar uno a criterio propio
2- Personal médico ingresa temperatura.	
3-Personal médico ingresa pulso.	
4-Personal médico ingresa frecuencia respiratoria	
5-Personal médico ingresa frecuencia cardíaca	
6-Personal médico ingresa síntomas existentes del paciente.	
7- Sistema sugiere formulario relacionado con el síntoma (Dolor, Malestar, Fiebre, Otro) RA1	
Estado final del Sistema:	Funcionando.
Comentarios:	Se entiende como datos de inspección general a: Estado de conciencia, Grado de colaboración, Movilidad, Actitud, Estado de nutrición y Grado de hidratación.

Caso N°5

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Prediagnóstico/Diagnóstico
5	R.F. 2.5
ACTOR:	Personal médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Personal médico selecciona elaborar un diagnóstico.	RA1. Si desea puede elaborar un diagnóstico propio
2- Personal médico selecciona tipo de diagnóstico. (RA1)	RA2. El personal médico puede cancelar el diagnóstico si no es el correcto.
3-Personal médico confirma diagnóstico. (RA2)	
4-Personal médico solicita un análisis clínico	
5-Personal médico ingresa resultado del análisis clínico.	
Estado final del Sistema:	Funcionando.
Comentarios:	

Caso N°6

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Ingreso tratamientos
6	R.F. 2.6
ACTOR:	Personal médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Personal médico ingresa nombre del tratamiento.	RA1. Puede no ingresar fecha final, si ingresa el sistema valida que la fecha final sea mayor que la fecha de inicio.
2- Personal ingresa descripción del tratamiento.	
3-Personal médico ingresa fecha inicio del tratamiento.	
4-Personal médico ingresa fecha final. RA1	
5- Personal médico confirma tratamiento.	
6-Personal médico consulta registro y controles del paciente en la institución médica.	
Estado final del Sistema:	Funcionando.
Comentarios:	

Caso N°7

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Creación formularios
7	R.F. 2.7
ACTOR:	Personal médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Personal médico determina tipo de formulario.	
2- Personal ingresa preguntas.	
3-Personal médico confirma formulario.	
Estado final del Sistema:	Funcionando.
Comentarios:	Esta ruta principal es solamente para la creación de un formulario.

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Modificación formularios
7	R.F. 2.7
ACTOR:	Personal médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1-Personal médico ingresa tipo de formulario.	
2-Sistema carga y filtra datos por tipo de formulario	
3-Personal médico modifica tipo de formulario	
4- Personal médico modifica la/s pregunta/s.	
Estado final del Sistema:	Funcionando.
Comentarios:	Esta ruta principal es solamente para la modificación de un formulario.

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Baja formularios
7	R.F. 2.7
ACTOR:	Personal médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1-Personal médico ingresa tipo de formulario.	RA1. ID no existe, Retorna a RP3.
2-Sistema carga y filtra datos por tipo de formulario	
3-Personal médico ingresa ID de formulario.	
4-Sistema da de baja el formulario. RA1	
Estado final del Sistema:	Funcionando.
Comentarios:	Esta ruta principal es solamente para la baja de un formulario.

Caso N°8

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Consultar información de paciente
8	R.F. 2.8
ACTOR:	Médico
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1-Médico ingresa CI del paciente	RA1. Retorna a RP1
2-Sistema valida CI. RA1	
3-Sistema carga datos personales del paciente.	
4-Sistema carga historial clínico en la institución.	
Estado final del Sistema:	Funcionando.
Comentarios:	

Caso N°9

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Consultar información de diagnósticos
9	R.F. 2.9
ACTOR:	Paciente
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1-Paciente ingresa CI	RA1. Retorna a RP1
2-Sistema valida CI. RA1	3-Sistema carga lista de diagnósticos del paciente realizados en la institución médica.
3-Sistema carga lista de diagnósticos del paciente realizados en la institución médica.	
Estado final del Sistema:	Funcionando.
Comentarios:	

Caso N°10

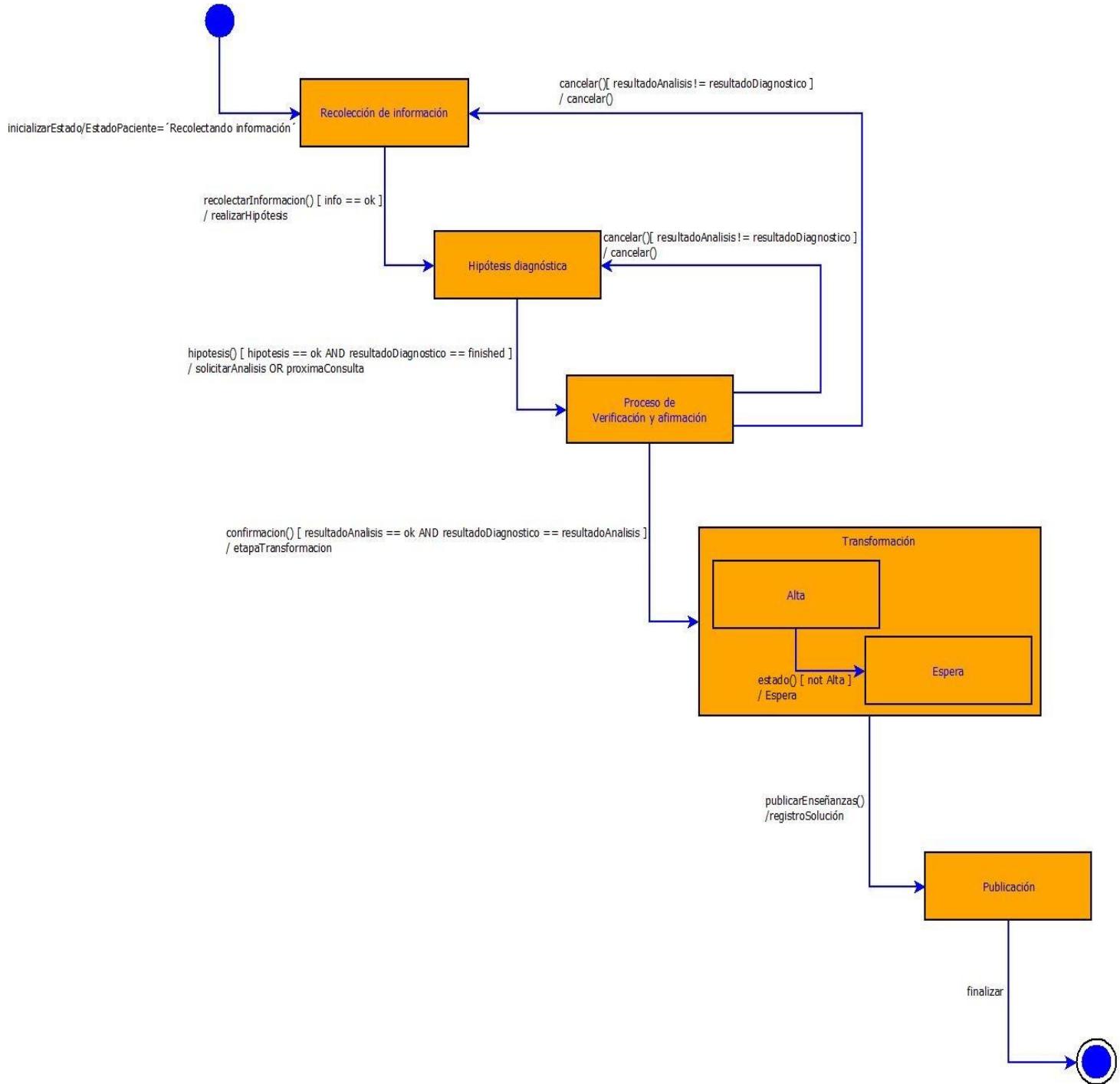
ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Consultar información de análisis clínicos
10	R.F. 2.10
ACTOR:	Paciente
Estado inicial del Sistema:	Funcionando.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1-Paciente ingresa CI	RA1. Retorna a RP1
2-Sistema valida CI. RA1	3-Sistema carga lista de análisis del paciente realizados en la institución médica.
3-Sistema carga lista de análisis del paciente realizados en la institución médica.	
Estado final del Sistema:	Funcionando.
Comentarios:	



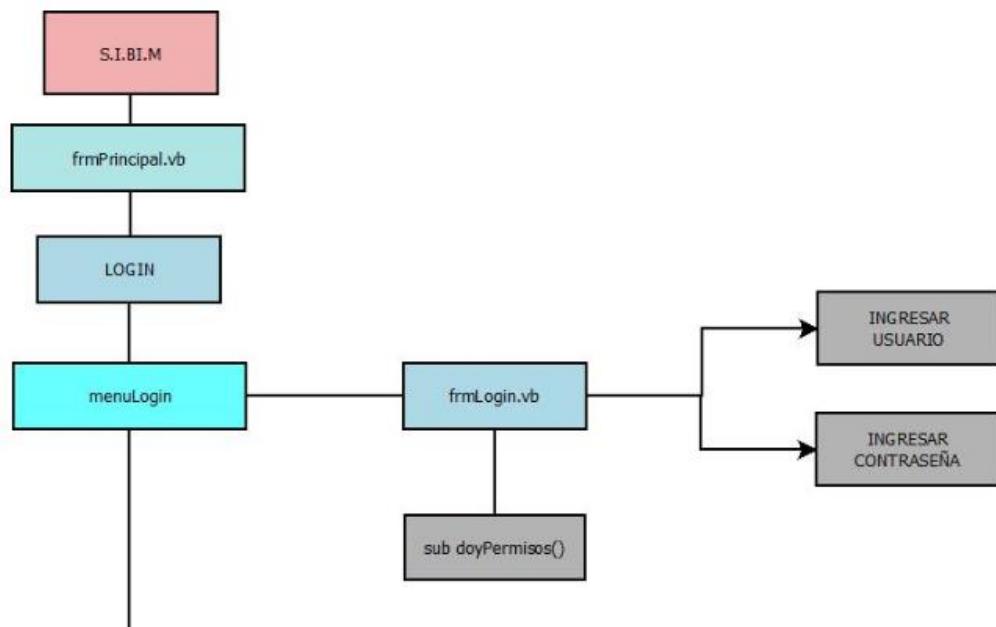
Caso N°11

ESPECIFICACIÓN DE CASOS DE USO	
Nº CASO	Salir del sistema.
11	R.F. 2.11
ACTOR:	Usuario del sistema
Estado inicial del Sistema:	Funcionando, con funciones desbloqueadas.
RUTAS DEL SISTEMA	
RUTA PRINCIPAL	RUTA ALTERNATIVA
1- Usuario selecciona cerrar sesión	
2-Sistema cierra sesión del usuario y bloquea funcionalidades.	
Estado final del Sistema:	Funcionando, con funciones bloqueadas.
Comentarios:	

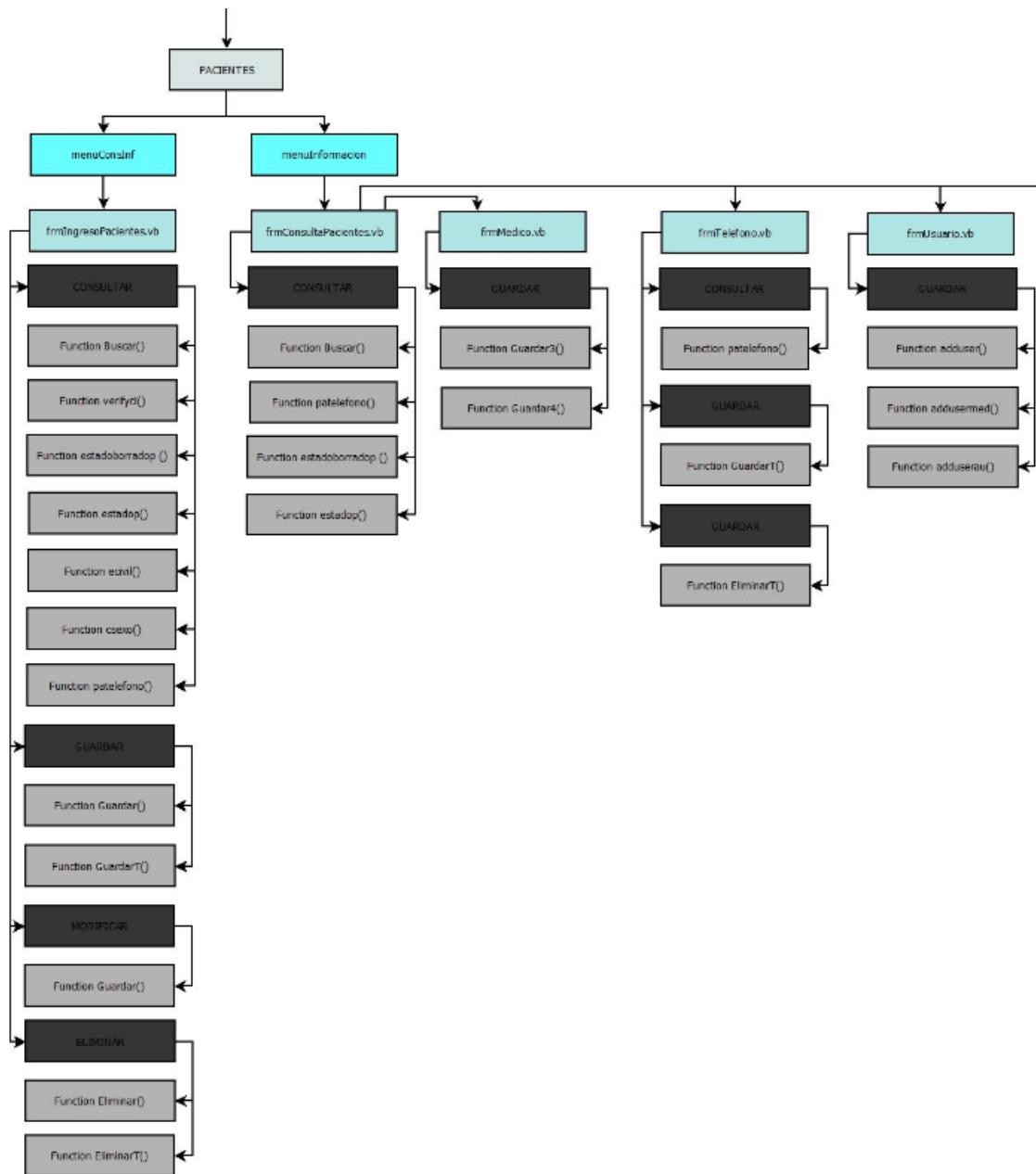
Diagrama de estados



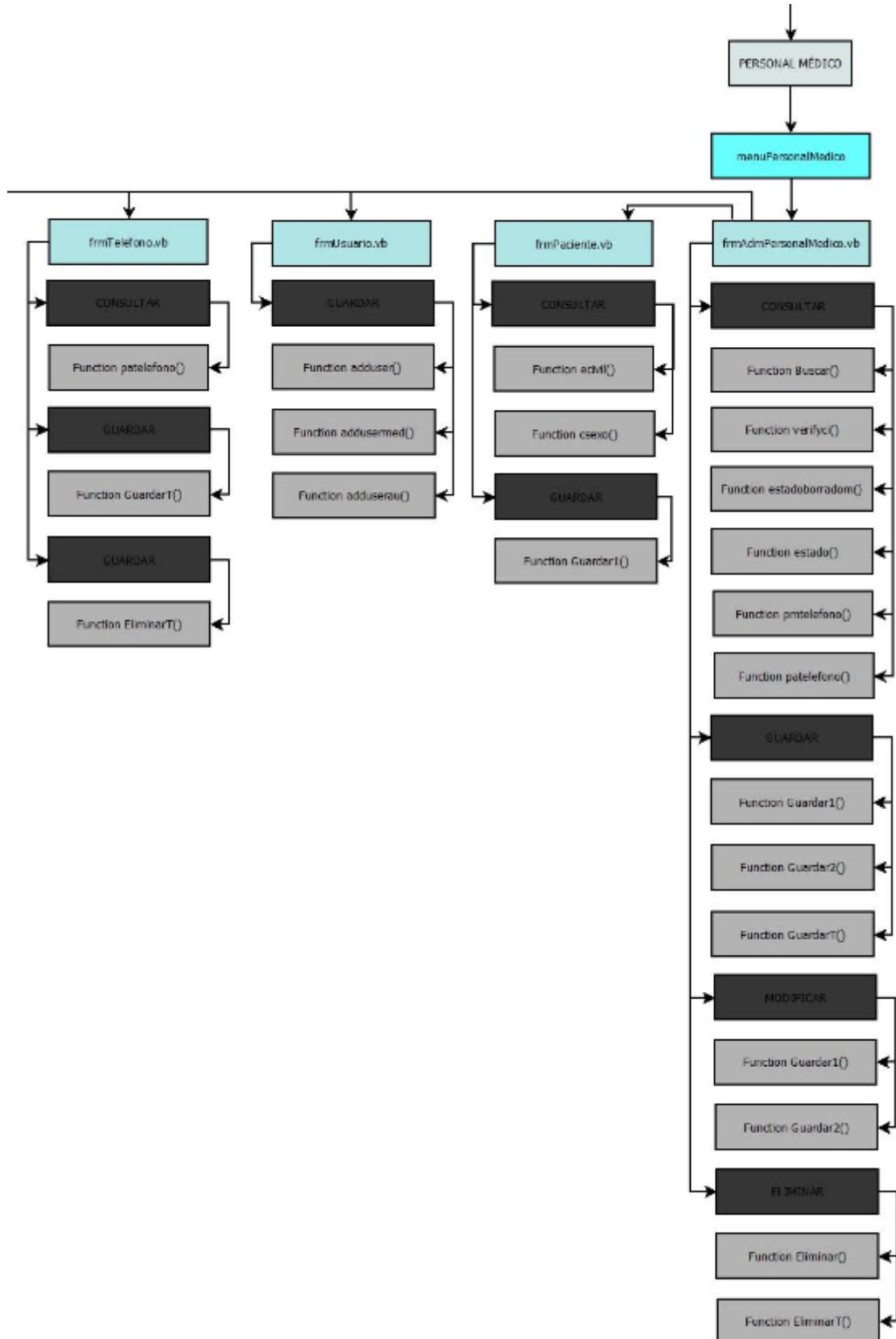
Carta jerárquica LOGIN



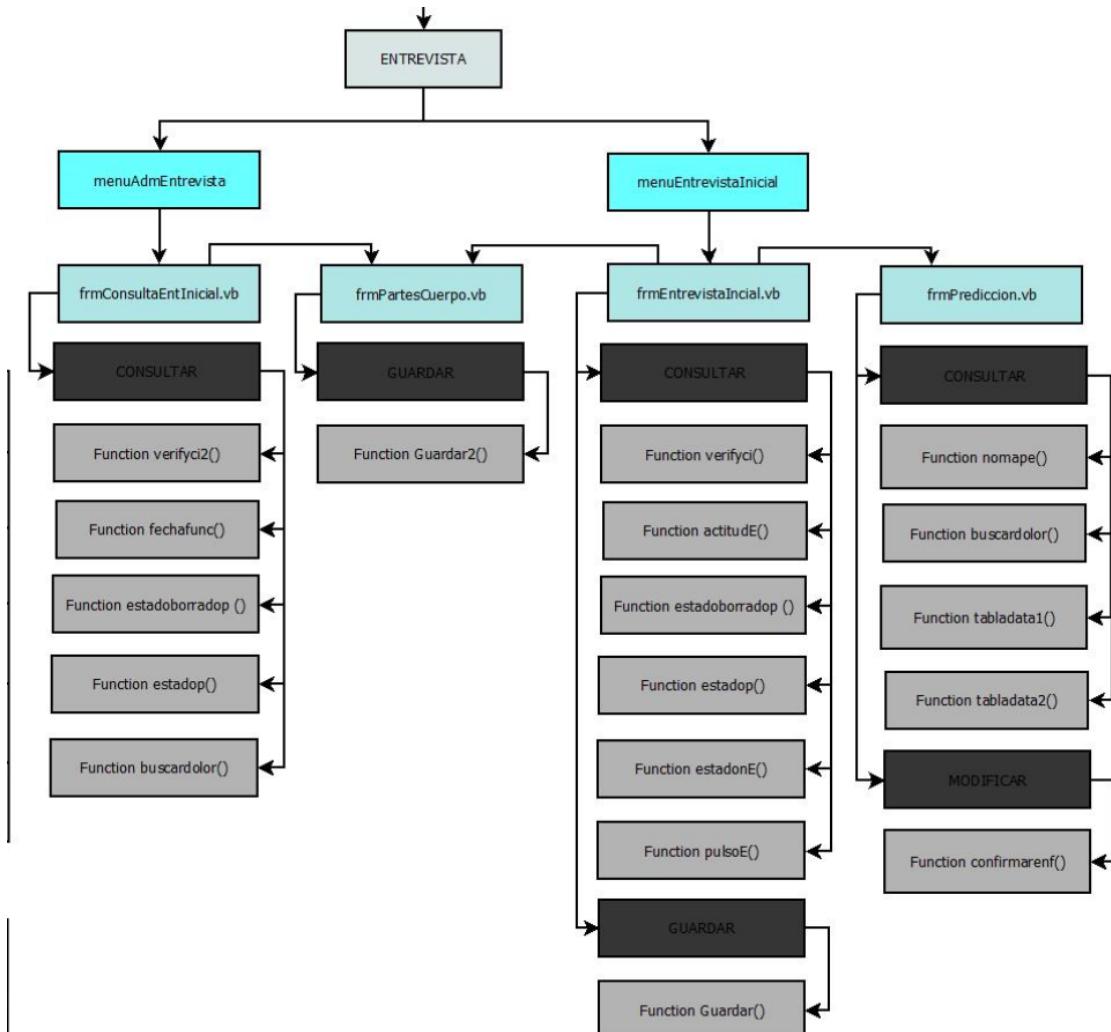
PACIENTES



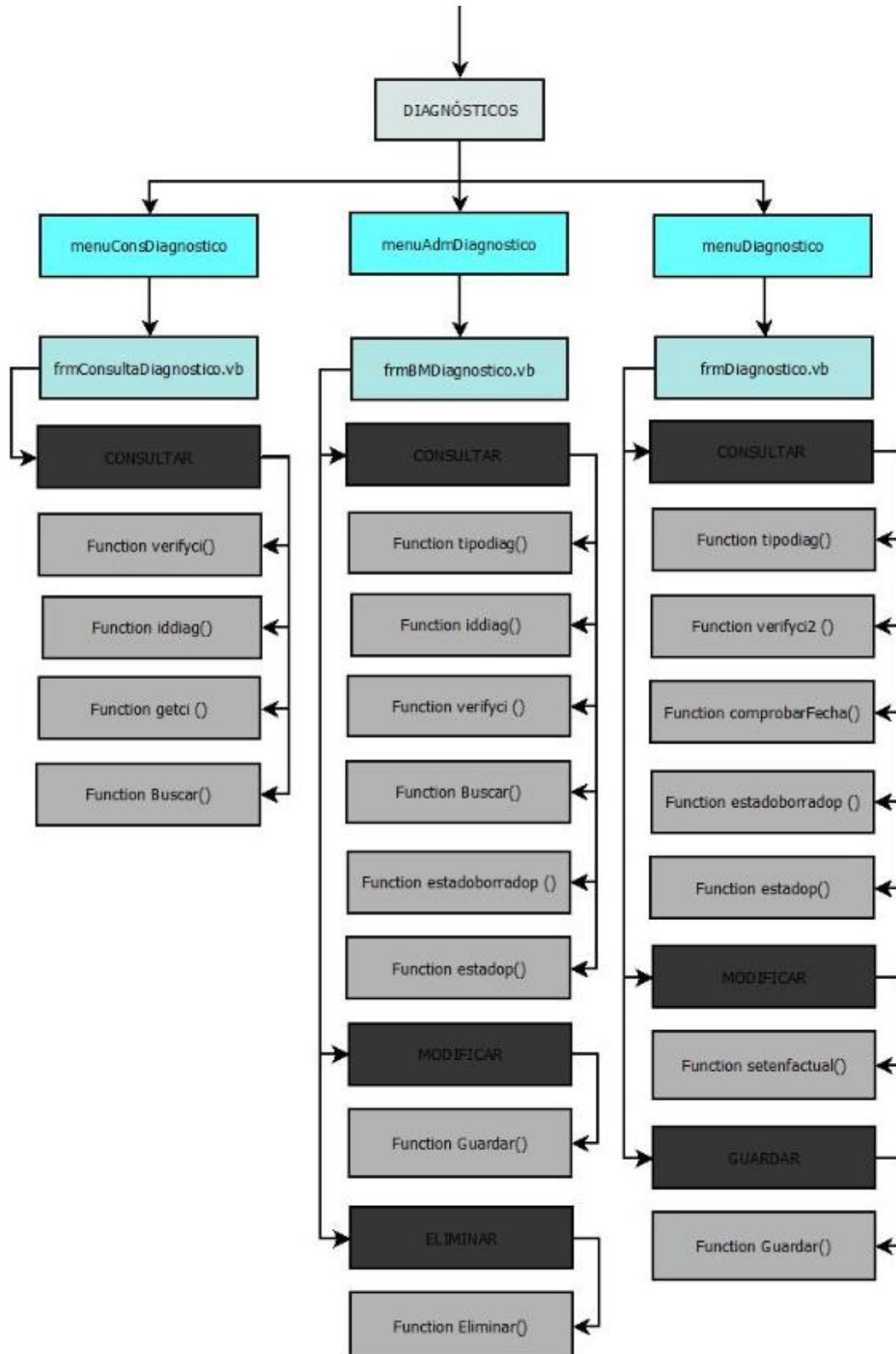
PERSONAL MÉDICO



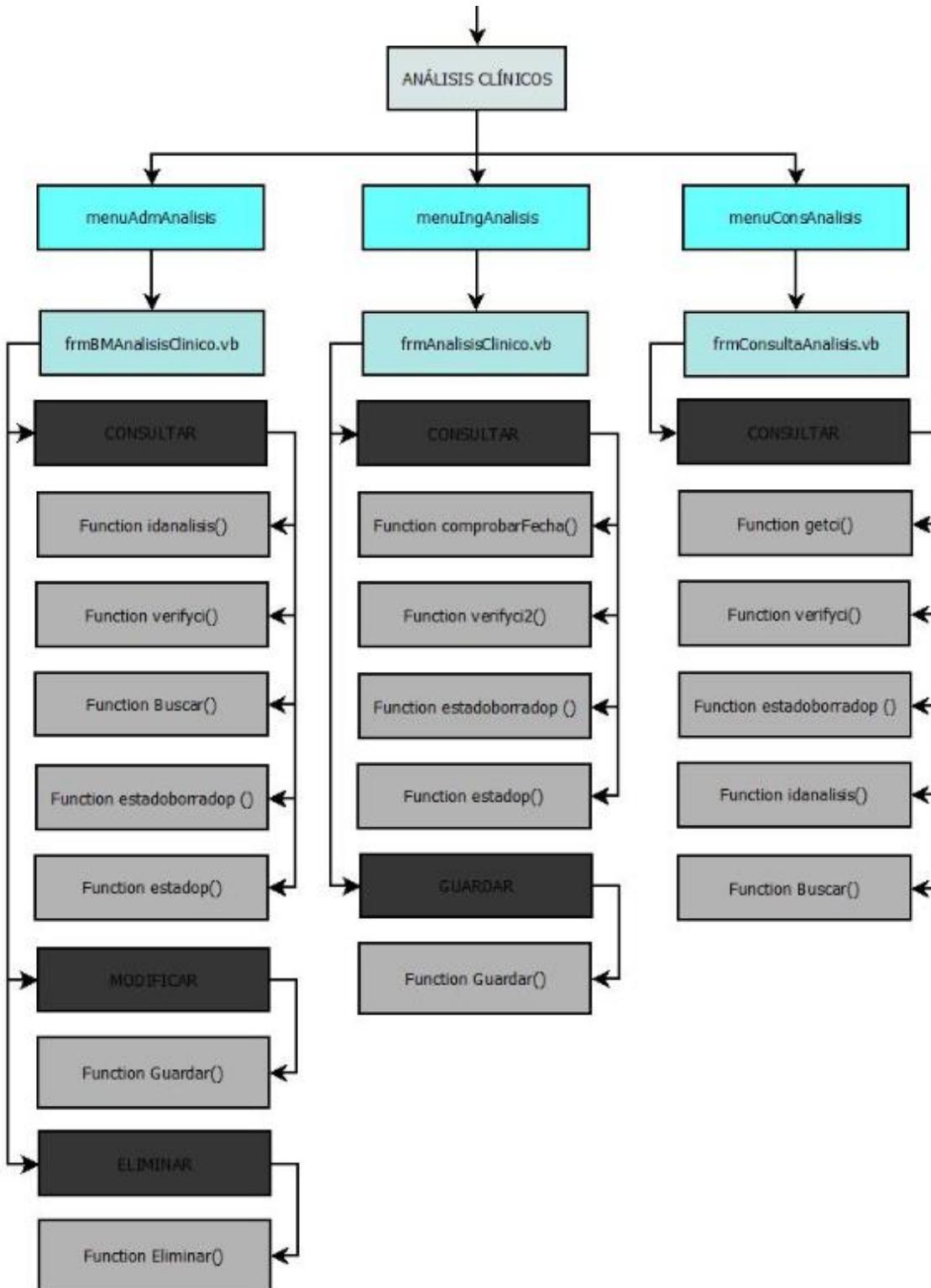
ENTREVISTA



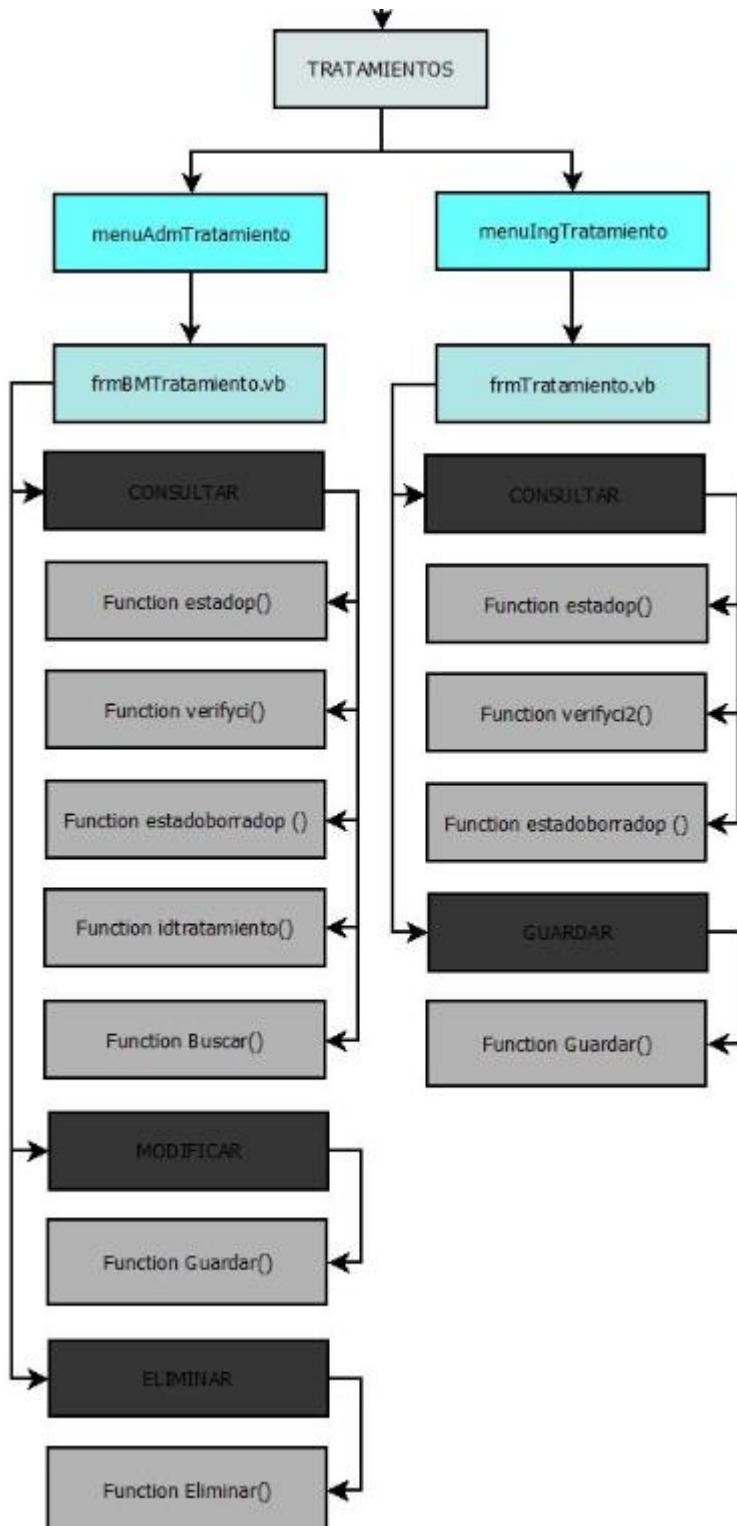
DIAGNÓSTICOS



ANÁLISIS CLÍNICOS



TRATAMIENTOS



FORMULARIOS

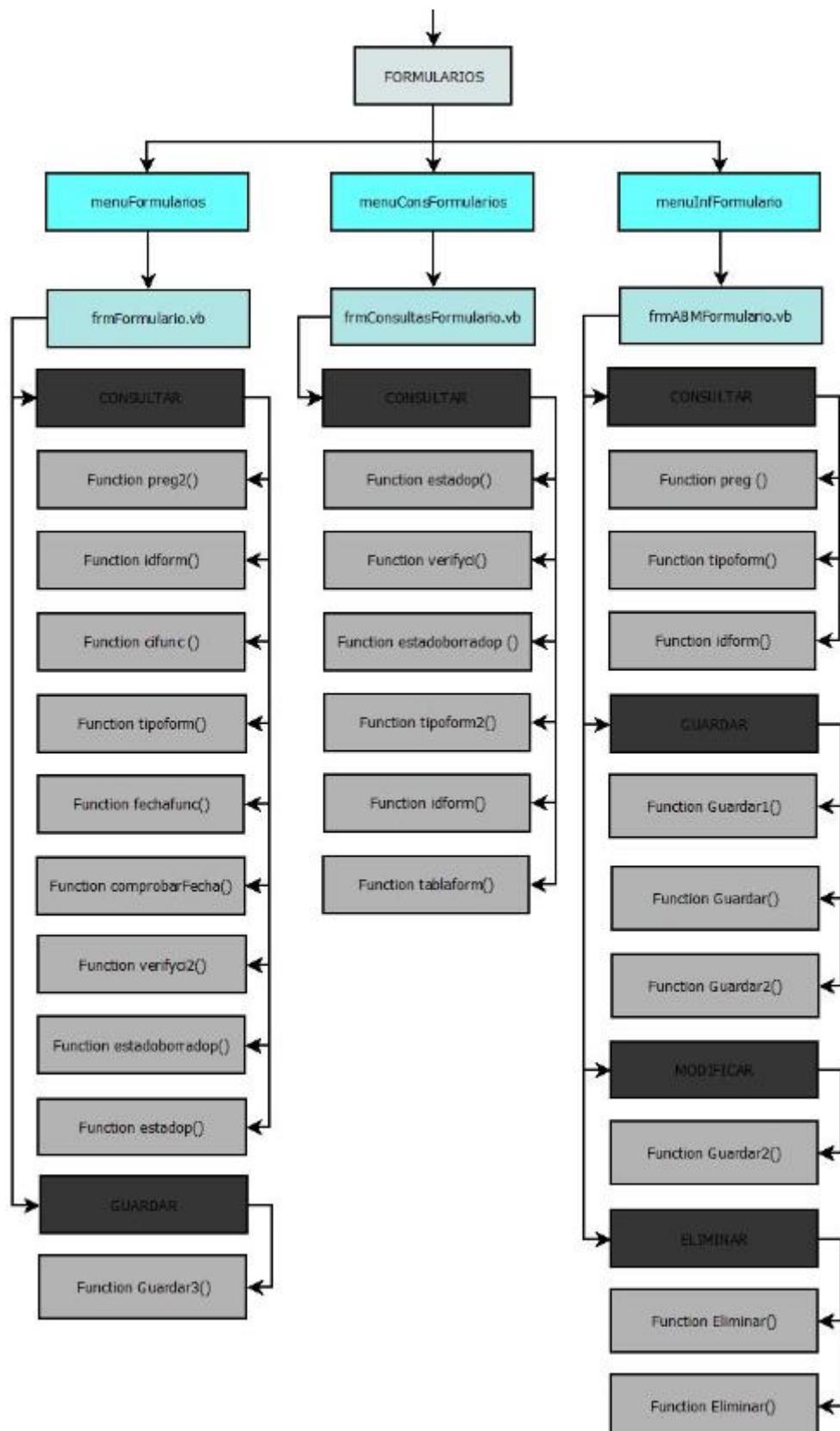
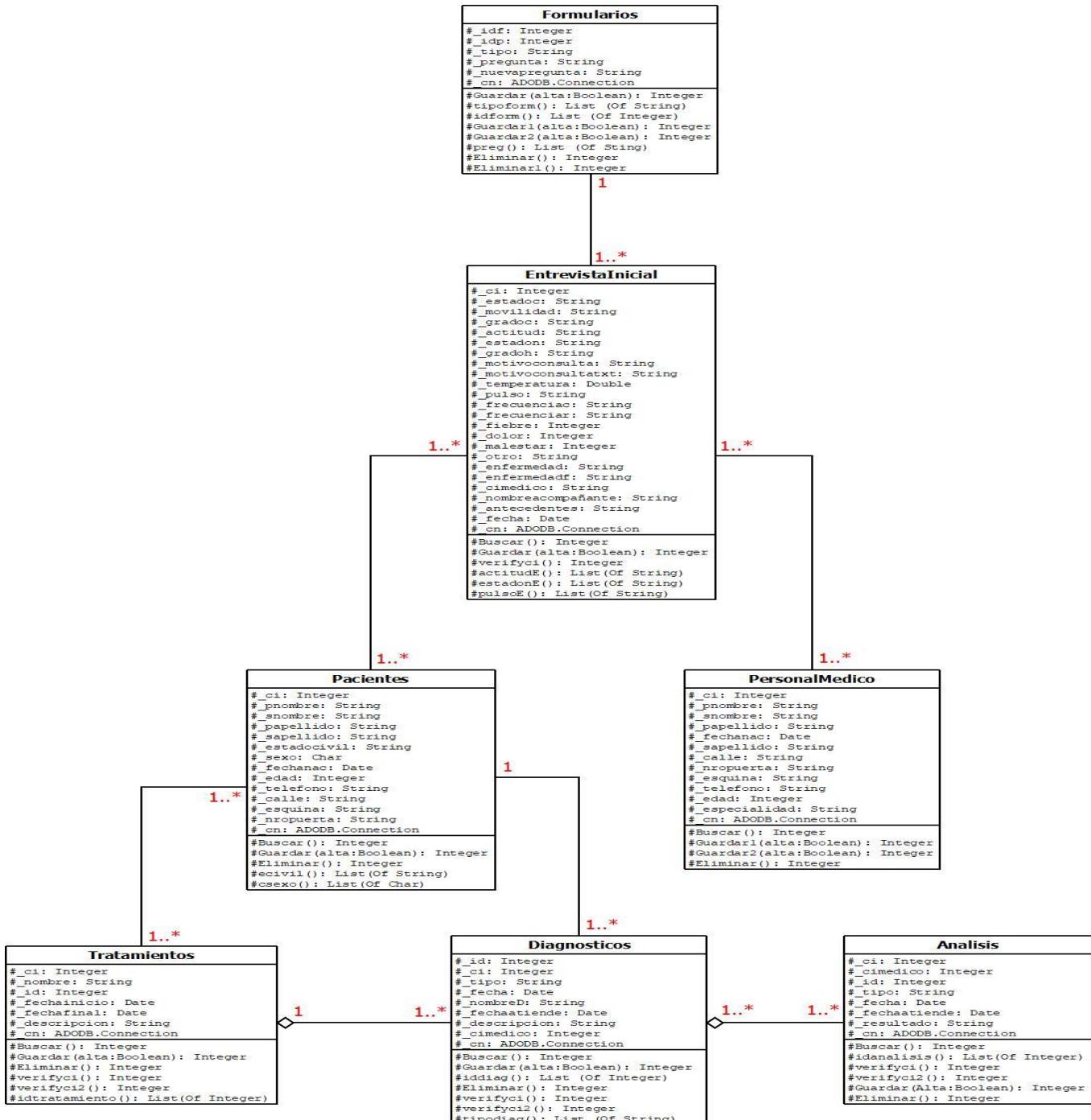




DIAGRAMA DE CLASES



PLAN DE TESTING

CP – Ingresar al sistema

Casos de prueba						
Funcionalidad: INGRESO AL SISTEMA						
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	7/11/2020	ggularde	Ingresar al sistema(Usuario existente) .	Realizar el ingreso al sistema	1. Ingresar usuario (Obligatorio) 2. Ingresar contraseña (Obligatorio) 3. Puslar el botón "Ingresar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder al "Login". Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	7/11/2020	ggularde	Ingresar al sistema (Usuario no existente)	Realizar el ingreso al sistema	1. Ingresar usuario (Obligatorio) 2. Ingresar contraseña (Obligatorio) 3. Puslar el botón "Ingresar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder al "Login". Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp003	7/11/2020	ggularde	Cerrar sesión	Realizar la finalización de la sesión	1. Seleccionar la opción "SI" al preguntarnos si queremos cerrar sesión	Equipo encendido y activo. Permisos para acceder al "Login" . Batería suficiente para realizar la acción.
cp003	7/11/2020	ggularde	Salir del sistema	Realizar la salida del sistema	1. Pulsar en el menú "Salir"	Equipo encendido y activo. Permisos para acceder a "Salir" . Batería suficiente para realizar la acción.
Resultado esperado		Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución
El sistema habilita el menú del programa y le otorga los permisos correspondientes al usuario.		El esperado	-	v1.0	-	7/11/2020 ggularde
El sistema despliega una ventana emergente notificando que el usuario no tiene un rol asignado.		El esperado	-	v1.0	-	7/11/2020 ggularde
El sistema cierra sesión y quita los permisos de acceso.		El esperado	-	v1.0	-	7/11/2020 ggularde
El sistema se cierra.		El esperado	-	v1.0	-	7/11/2020 ggularde

CP – Adm. – Pacientes

Casos de prueba								
Funcionalidad:				ADMINISTRAR PACIENTES				
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución		
cp001	07/11/2020	ggularde	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración - Pacientes". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.		
cp002	07/11/2020	ggularde	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración - Pacientes". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.		
Resultado esperado			Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema carga en pantalla los datos del paciente y habilita los botones para modificar y eliminar al mismo.			El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.			El esperado	-	v1.0	-	07/11/2020	ggularde

cp003	07/11/2020	ggularte	Buscar paciente(Paciente no existente pero si registrado como personal médico) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Administración - Pacientes". Permiso para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado. Batería suficiente para realizar la acción.
cp004	07/11/2020	siriart	Registrar al paciente en el sistema	Realizar el ingreso de datos pertenecientes al ingreso del sistema del paciente.	1. Ingresar usuario(Obligatorio). 2. Ingresar contraseña(Obligatorio). 3. Confirmar contraseña(Obligatorio). 4. Pulsar el botón ingresar.	Equipo encendido y activo. Permiso para acceder a "Administración - Pacientes". Permiso para ejecutar el comando INSERT y GRANT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.

cp005	07/11/2020	siriart	Ingresar paciente(Siendo que este no pertenece al personal médico ni ya está ingresado).	Realizar el ingreso de datos pertenecientes al paciente.	<ol style="list-style-type: none"> 1. Ingresar Primer nombre(Obligatorio). 2. Ingresar Segundo nombre(Opcional). 3. Ingresar primer apellido(Obligatorio). 4. Ingresar segundo apellido (Obligatorio). 5. Seleccionar fecha de nacimiento(Obligatorio). 6. Seleccionar sexo (obligatorio) 7. Ingresar correo(Opcional) 8. Ingresar calle (Obligatorio) 9. Ingresar esquina(Obligatorio) 10. Ingresar nº puerta (Obligatorio) 11. Seleccionar estado civil(Obligatorio) 12. Pulsar botón ingresar 	Equipo encendido y activo. Permisos para acceder a "Administración - Pacientes". Permisos para ejecutar el comando INSERT sobre la base de datos. Botón "Ingresar" habilitado. Batería suficiente para realizar la acción.
cp006	07/11/2020	siriart	Ingresar paciente(Siendo que este no existe pero está ingresado como personal médico).	Realizar el ingreso de datos pertenecientes al paciente.	<ol style="list-style-type: none"> 1. Seleccionar sexo (obligatorio) 2. Seleccionar estado civil(Obligatorio) 3. Pulsar botón ingresar 	Equipo encendido y activo. Permisos para acceder a "Administración - Pacientes". Permisos para ejecutar el comando INSERT sobre la base de datos. Botón "Ingresar" habilitado. Batería suficiente para realizar la acción.
			El sistema despliega una ventana emergente notificando que el paciente se ingresó exitosamente. Aparecerá en pantalla una ventana para ingresar los teléfonos del paciente.	El esperado	v1.0	07/11/2020 ggularde
			El sistema despliega una ventana emergente notificando que el paciente se ingresó exitosamente.	-	v1.0	07/11/2020 ggularde

cp007	07/11/2020	siriart	Agregar teléfono.	Realizar el ingreso del teléfono perteneciente al paciente.	1. Ingresar el teléfono (obligatorio) 2. Pulsar botón agregar	Equipo encendido y activo. Permiso para acceder a "Administración - Pacientes". Permiso para ejecuta el comando INSERT sobre la base de datos. Boton "Agregar" habilitado . Batería suficiente para realizar la acción.
cp008	07/11/2020	siriart	Quitar teléfono.	Realizar la eliminación del teléfono perteneciente al paciente.	1. Seleccionar el teléfono que se desea quitar en la lista desplegable (obligatorio) 2. Pulsar botón quitar	Equipo encendido y activo. Permiso para acceder a "Administración - Pacientes". Permiso para ejecuta el comando DELETE sobre la base de datos. Boton "Quitar" habilitado . Batería suficiente para realizar la acción.
cp009	07/11/2020	siriart	Modificar paciente.	Realizar la modificación de los datos pertenecientes al paciente.	1. Modificar los datos que se desean cambiar (obligatorio) 2. Pulsar botón modificar	Equipo encendido y activo. Permiso para acceder a "Administración - Pacientes". Permiso para ejecuta el comando UPDATE sobre la base de datos. Boton "Modificar" habilitado . Batería suficiente para realizar la acción.

El sistema despliega una ventana emergente notificando que el teléfono se ingresó exitosamente.	El esperado	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el teléfono se quitó exitosamente.	El esperado	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que los datos se modificaron exitosamente.	El esperado	v1.0	-	07/11/2020	ggularde

cp010	07/11/2020	siriart	Eliminar paciente.	Realizar la baja del paciente.	1. Pulsar botón eliminar	Equipo encendido y activo. Permiso para acceder a "Administración - Pacientes". Permiso para ejecuta el comando UPDATE sobre la base de datos. Boton "Eliminar" habilitado . Bateria suficiente para realizar la acción.
cp011	07/11/2020	siriart	Buscar paciente(Que esta dado de baja).	Realizar la busqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Puslar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permiso para acceder a "Administración - Pacientes". Permiso para ejecuta el comando UPDATE, SELECT sobre la base de datos . Bateria suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el paciente se dio de baja exitosamente.			El esperado	v1.0	-	07/11/2020 ggularde
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.			El esperado	v1.0	-	07/11/2020 ggularde

CP – Adm. – Pers. Médico

Casos de prueba						
Funcionalidad:				ADMINISTRAR PACIENTES		
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	7/11/2020	ggularte	Buscar médico o auxiliar(médico o auxiliar existente) .	Realizar la búsqueda del médico o auxiliar.	1. Ingresar cédula de identidad del médico o auxiliar (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración - Personal médico". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	7/11/2020	ggularte	Buscar médico o auxiliar(médico o auxiliar no existente) .	Realizar la búsqueda del médico o auxiliar.	1. Ingresar cédula de identidad del médico o auxiliar (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración - Personal médico". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp003	7/11/2020	ggularte	Buscar médico o auxiliar(médico o auxiliar no existente pero si registrado como paciente) .	Realizar la búsqueda del médico o auxiliar	1. Ingresar cédula de identidad del médico o auxiliar (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración - Personal médico". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema carga en pantalla los datos del médico o auxiliar y habilita los botones para modificar y eliminar al mismo.	El esperado	-	v1.0	-	7/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el médico o auxiliar no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	7/11/2020	ggularte
El sistema despliega una ventana emergente notificando que la persona con la cédula de identidad que se ingresó anteriormente está ingresada como paciente y para realizar el alta como médico o auxiliar será necesario ingresar determinados datos.	El esperado	-	v1.0	-	7/11/2020	ggularte

cp004	7/11/2020	ggularte	Registrar al médico en el sistema	Realizar el ingreso de datos pertenecientes al ingreso del sistema del médico.	1. Ingresar usuario(Obligatorio). 2. Ingresar contraseña(Obligatorio). 3. Confirmar contraseña(Obligatorio). 4. Pulsar el botón ingresar.	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando INSERT y GRANT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp005	7/11/2020	ggularte	Registrar al auxiliar en el sistema	Realizar el ingreso de datos pertenecientes al ingreso del sistema del auxiliar.	1. Ingresar usuario(Obligatorio). 2. Ingresar contraseña(Obligatorio). 3. Confirmar contraseña(Obligatorio). 4. Pulsar el botón ingresar.	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando INSERT y GRANT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp006	7/11/2020	ggularte	Ingresar médico o auxiliar(Siendo que este no es un paciente ni ya está ingresado).	Realizar el ingreso de datos pertenecientes al médico o auxiliar.	1. Ingresar Primer nombre(Obligatorio). 2. Ingresar Segundo nombre(Opcional). 3. Ingresar primer apellido(Obligatorio). 4. Ingresar segundo apellido (Obligatorio). 5. Seleccionar fecha de nacimiento(Obligatorio), 6. Seleccionar sexo (obligatorio) 7. Ingresar correo(Opcional) 8. Ingresar calle (Obligatorio) 9. Ingresar esquina(Obligatorio) 10. Ingresar nº puerta (Obligatorio) 11. Seleccionar estado civil(Obligatorio) 12. Ingresar especialidad (Opcional) 13. Pulsar botón ingresar	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando INSERT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el médico o auxiliar se registró en el sistema correctamente. Se habilita el ingreso de datos para paciente			El esperado	v1.0	-	7/11/2020 ggularte
El sistema despliega una ventana emergente notificando que el médico o auxiliar se registró en el sistema correctamente. Se habilita el ingreso de datos para paciente			El esperado	v1.0	-	7/11/2020 ggularte
El sistema despliega una ventana emergente notificando que el médico o auxiliar se ingresó exitosamente. Aparecerá en pantalla una ventana para ingresar los teléfonos del paciente.			El esperado	v1.0	-	7/11/2020 ggularte

cp007	7/11/2020	ggularde	Ingresar médico o auxiliar(Siendo que este no está ingresado pero es un paciente).	Realizar el ingreso de datos pertenecientes al médico o auxiliar.	1. Tipo de personal (obligatorio) 2. Seleccionar especialidad(Opcional) 3. Pulsar botón ingresar	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando INSERT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp008	7/11/2020	ggularde	Agregar teléfono.	Realizar el ingreso del teléfono perteneciente al médico o auxiliar.	1. Ingresar el teléfono (obligatorio) 2. Pulsar botón agregar	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando INSERT sobre la base de datos. Boton "Agregar" habilitado . Batería suficiente para realizar la acción.
cp009	7/11/2020	ggularde	Quitar teléfono.	Realizar la eliminación del teléfono perteneciente al médico o auxiliar.	1. Seleccionar el teléfono que se desea quitar en la lista desplegable (obligatorio) 2. Pulsar botón quitar	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando DELETE sobre la base de datos. Boton "Quitar" habilitado . Batería suficiente para realizar la acción.
cp010	7/11/2020	ggularde	Modificar médico o auxiliar.	Realizar la modificación de los datos pertenecientes al médico o auxiliar.	1. Modificar los datos que se desean cambiar (obligatorio) 2. Pulsar botón modificar	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando UPDATE sobre la base de datos. Boton "Modificar" habilitado . Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el médico o auxiliar se ingresó exitosamente.			El esperado	v1.0	-	7/11/2020 ggularde
El sistema despliega una ventana emergente notificando que el teléfono se ingresó exitosamente.			El esperado	v1.0	-	7/11/2020 ggularde
El sistema despliega una ventana emergente notificando que el teléfono se quitó exitosamente.			El esperado	v1.0	-	7/11/2020 ggularde
El sistema despliega una ventana emergente notificando que los datos se modificaron exitosamente.			El esperado	v1.0	-	7/11/2020 ggularde

cp011	7/11/2020	ggularde	Eliminar médico o auxiliar.	Realizar la baja del médico o auxiliar.	1. Pulsar botón eliminar	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando UPDATE sobre la base de datos. Boton "Eliminar" habilitado . Batería suficiente para realizar la acción.
cp012	7/11/2020	ggularde	Buscar médico o auxiliar(Que esta dado de baja).	Realizar la búsqueda del médico o auxiliar. Realizar el alta del médico o auxiliar.	1. Ingresar cédula de identidad del médico o auxiliar (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del médico o auxiliar.	Equipo encendido y activo. Permiso para acceder a "Administración - Personal médico". Permiso para ejecuta el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el médico o auxiliar se dio de baja exitosamente.	El esperado		v1.0		7/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el médico o auxiliar se dio de alta exitosamente.	El esperado		v1.0		7/11/2020	ggularde

CP – Adm. – Entrevista

Casos de prueba						
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	7/11/2020	siriart	Buscar paciente(Paciente existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Entrevista". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	7/11/2020	siriart	Buscar paciente(Paciente no existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Entrevista". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp003	7/11/2020	siriart	Buscar paciente(Que está dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Administración-Entrevista". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
Resultado esperado		Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución
El sistema habilitará una agrupación de datos para administrar una entrevista inicial.		El esperado	-	v1.0	-	7/11/2020 ggularde
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.		El esperado	-	v1.0	-	7/11/2020 ggularde
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.		El esperado	-	v1.0	-	7/11/2020 ggularde

cp004	7/11/2020	siriart	Buscar Entrevista.	Realizar la búsqueda de la entrevista	1. Seleccionar fecha de la entrevista que se quiere buscar.	Equipo encendido y activo. Permisos para acceder a "Administración-Entrevista". Permisos para ejecutar el comando SELECT sobre la base de datos. Batería suficiente para realizar la acción.
cp005	7/11/2020	siriart	Consultar las partes del cuerpo con dolor	Realizar la búsqueda de las partes del cuerpo con dolor	1. Pulsar botón consultar partes con dolor del cuerpo.	Equipo encendido y activo. Permisos para acceder a "Administración-Entrevista". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón de consultar las partes del cuerpo está habilitado. Batería suficiente para realizar la acción.
El sistema cargará en pantalla los datos correspondientes a la entrevista.						
El sistema despliega una ventana emergente mostrando las partes del cuerpo marcadas en donde el paciente tuvo dolor.						

CP – Adm. – Análisis Clínico.

Casos de prueba						
Funcionalidad:		ADMINISTRAR ANÁLISIS CLÍNICOS				

Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos		Entradas	Condiciones de ejecución
				Pasos			
cp001	07/11/2020	zrosales	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente		1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Análisis Clínico". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	07/11/2020	zrosales	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente		1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Análisis Clínico". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para administrar un análisis clínico	El esperado	-	v1.0	-	07/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularte

cp003	07/11/2020	zrosales	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Administración-Análisis Clínico". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp004	07/11/2020	zrosales	Buscar Análisis clínico.	Realizar la consulta de un análisis clínico.	1. Seleccionar el número identificador (ID) del análisis clínico. (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Análisis Clínico". Permisos para ejecutar el comando SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp005	07/11/2020	zrosales	Modificar análisis clínico.	Realizar la modificación de los datos pertenecientes al análisis clínico.	1. Modificar los datos que se desean cambiar (obligatorio) 2. Pulsar botón modificar	Equipo encendido y activo. Permisos para acceder a "Administración -Análisis Clínico". Permisos para ejecutar el comando UPDATE sobre la base de datos. Botón "Modificar" habilitado . Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.		El esperado	-	v1.0	-	07/11/2020 ggularte
El sistema cargará en pantalla los datos pertenecientes al análisis clínico con el ID seleccionado anteriormente.		El esperado	-	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que los datos se modificaron exitosamente.		El esperado	-	v1.0	-	07/11/2020 ggularte



cp006	07/11/2020	zrosales	Eliminar análisis clínico.	Realizar la baja del análisis clínico..	1. Pulsar botón eliminar	Equipo encendido y activo. Permisos para acceder a "Administración -Análisis Clínico". Permisos para ejecutar el comando UPDATE sobre la base de datos. Botón "Eliminar" habilitado . Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el análisis clínico se dio de baja exitosamente.	El esperado		v1.0	-	07/11/2020	ggularde

CP – Plan – Ingreso tratamiento

Casos de prueba						
Funcionalidad:				INGRESO TRATAMIENTOS		
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	siriart	Buscar paciente(Paciente existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Plan-Ingreso Tratamientos". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado. Batería suficiente para realizar la acción.
cp002	07/11/2020	siriart	Buscar paciente(Paciente no existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Plan-Ingreso Tratamientos". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado. Batería suficiente para realizar la acción.
cp003	07/11/2020	siriart	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Plan-Ingreso Tratamientos". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos. Batería suficiente para realizar la acción.
cp004	07/11/2020	siriart	Ingresar tratamiento.	Realizar el ingreso de un tratamiento	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Ingresar nombre.(Obligatorio) 4. Ingresar fecha inicio(Obligatorio) 5. Ingresar fecha final (Obligatorio) Ingresar descripción. (Obligatorio) Pulsar botón "Ingresar"	Equipo encendido y activo. Permisos para acceder a "Plan-Ingreso Tratamientos". Permisos para ejecutar el comando INSERT, SELECT sobre la base de datos . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para ingresar un tratamiento	El esperado	-	v1.0	-	07/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró.	El esperado	-	v1.0	-	07/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el tratamiento se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularte

CP – Plan – Ingreso análisis clínico

Casos de prueba						
Funcionalidad:				INGRESO ANÁLISIS CLÍNICOS		
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	siriart	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan-Ingreso Análisis Clínico". Permiso para ejecuta el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	07/11/2020	siriart	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan-Ingreso Análisis Clínico". Permiso para ejecuta el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp003	07/11/2020	siriart	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permiso para acceder a "Plan-Ingreso Análisis Clínico". Permiso para ejecuta el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp004	07/11/2020	siriart	Ingresar análisis clínico.	Realizar el ingreso de un análisis clínico	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Ingresar Tipo.(Obligatorio) 4. Ingresar fecha (Obligatorio) 5. Ingresar resultado (Obligatorio) 6. Pulsar botón "Ingresar"	Equipo encendido y activo. Permiso para acceder a "Plan-Ingreso Análisis Clínico". Permiso para ejecuta el comando INSERT, SELECT sobre la base de datos . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para consultar un análisis clínico	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el análisis clínico se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularde

CP – Prediagnóstico/Diag. – Diagnóstico

Casos de prueba						
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	siriart	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Diagnóstico". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	07/11/2020	siriart	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Diagnóstico". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp003	07/11/2020	siriart	Buscar paciente(Que está dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Diagnóstico". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp004	07/11/2020	siriart	Ingresar datos de diagnóstico	Realizar el ingreso de datos de diagnóstico	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Ingresar Tipo (Obligatorio) 4. Ingresar fecha (Obligatorio) 5. Ingresar descripción (Obligatorio) 6. Pulsar botón "Ingresar"	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Diagnóstico". Permisos para ejecutar el comando INSERT, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
Resultado esperado		Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución
El sistema habilitará una agrupación de datos para consultar un diagnóstico		El esperado	-	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.		El esperado	-	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.		El esperado	-	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que el diagnóstico se dio de alta exitosamente.		El esperado	-	v1.0	-	07/11/2020 ggularte

CP – Prediagnóstico/Diag. – Formulario

Casos de prueba						
Funcionalidad:		Formulario				
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	siriart	Buscar paciente(Paciente existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Formularios". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado. Batería suficiente para realizar la acción.
cp002	07/11/2020	siriart	Buscar paciente(Paciente no existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Formularios". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado. Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para consultar un formulario	El esperado	-	v1.0	-	07/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularte

cp003	07/11/2020	siriart	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Formularios". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos. Batería suficiente para realizar la acción.
cp004	07/11/2020	siriart	Ingresar respuestas a un formulario	Realizar el ingreso de respuestas a un formulario	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Seleccionar fecha atiende (Obligatorio) 4. Seleccionar CI_Medico (Obligatorio) 5. Seleccionar Tipo.(Obligatorio) 6. Seleccionar ID (Obligatorio) 7. Seleccionar pregunta (Obligatorio) 8. Ingresar respuesta(Obligatorio) 9. Pulsar botón "Ingresar"	Equipo encendido y activo. Permisos para acceder a "Prediagnóstico/Diagnóstico-Formularios". Permisos para ejecutar el comando INSERT, SELECT sobre la base de datos. Batería suficiente para realizar la acción.

El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularate
El sistema despliega una ventana emergente notificando que las respuestas han sido ingresadas al formulario exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularate

CP – Anamnesis – Información Pac.

Casos de prueba

Funcionalidad:	Información Paciente

Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	siriart	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Anamnesis-Información paciente". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	07/11/2020	siriart	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Anamnesis-Información paciente". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp003	07/11/2020	siriart	Buscar paciente(Que está dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Anamnesis-Información paciente". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para consultar la información del paciente	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularde

CP – Adm – Tratamientos

Casos de prueba				
Funcionalidad: ADMINISTRAR TRATAMIENTOS				
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos
cp001	07/11/2020	ggularte	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente 1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)

Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	ggularte	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente 1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Tratamiento". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	07/11/2020	ggularte	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Tratamiento". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para administrar un tratamiento	El esperado	-	v1.0	-	07/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularte

cp003	07/11/2020	ggularte	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Administración-Tratamiento". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp004	07/11/2020	ggularte	Buscar tratamiento.	Realizar la consulta de un tratamiento.	1. Seleccionar el número identificador (ID) del tratamiento. (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Tratamiento". Permisos para ejecutar el comando SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp005	07/11/2020	ggularte	Modificar tratamiento.	Realizar la modificación de los datos pertenecientes al tratamiento.	1. Modificar los datos que se desean cambiar (obligatorio) 2. Pulsar botón modificar	Equipo encendido y activo. Permisos para acceder a "Administración - Tratamiento". Permisos para ejecutar el comando UPDATE sobre la base de datos. Botón "Modificar" habilitado . Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.		El esperado	-	v1.0	-	07/11/2020 ggularte
El sistema cargará en pantalla los datos pertenecientes al tratamiento con el ID seleccionado anteriormente.		El esperado	-	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que los datos se modificaron exitosamente.		El esperado	-	v1.0	-	07/11/2020 ggularte



cp006	07/11/2020	ggularte	Eliminar tratamiento.	Realizar la baja del tratamiento.	1. Pulsar botón eliminar	Equipo encendido y activo. Permisos para acceder a "Administración - Tratamiento". Permisos para ejecutar el comando UPDATE sobre la base de datos. Botón "Eliminar" habilitado. Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el tratamiento se dio de baja exitosamente.		El esperado	-	v1.0	-	07/11/2020 ggularte

CP – Adm. – Cons. Formularios

Casos de prueba						
Funcionalidad:				CONSULTAR FORMULARIOS		
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	zrosales	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Consultar formularios". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	07/11/2020	zrosales	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Consultar formularios". Permisos para ejecutar el comando SELECT sobre la base de datos. Botón "Ingresar" habilitado . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para consultar un formulario	El esperado	-	v1.0	-	07/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularte

cp003	07/11/2020	zrosales	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Administración-Consultar formularios". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp004	07/11/2020	zrosales	Consultar formulario con preguntas y respuestas.	Realizar la consulta de un formulario.	1. Seleccionar tipo de formulario. (Obligatorio) 2. Seleccionar número identificador (id) del formulario.(Obligatorio) -	Equipo encendido y activo. Permisos para acceder a "Administración-Consultar formularios". Permisos para ejecutar el comando SELECT sobre la base de datos . Batería suficiente para realizar la acción.

El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggular
El sistema carga en un DataGridView las preguntas con las respuestas correspondientes a las mismas del formulario seleccionado anteriormente.	El esperado	-	v1.0	-	07/11/2020	ggular

CP – Plan – Cons. Análisis Clin.

Casos de prueba						
Funcionalidad:				CONSULTAR ANÁLISIS CLÍNICOS		
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	siriart	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan- consultar análisis clínico". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	07/11/2020	siriart	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan- consultar análisis clínico". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp003	07/11/2020	siriart	Buscar paciente(Paciente existente pero con una cédula distinta a la cédula del paciente que está consultando) .	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan- consultar análisis clínico". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
El sistema habilitará una agrupación de datos para consultar un análisis clínico			El esperado	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.			El esperado	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que solo puede buscar análisis propios y no de otra persona			El esperado	v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que solo puede buscar análisis propios y no de otra persona			El esperado	v1.0	-	07/11/2020 ggularte

cp004	07/11/2020	siriart	Buscar paciente(Que esta dado de baja).	Realizar la busqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Puslar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Plan-consultar análisis clínico". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp005	07/11/2020	zrosales	Buscar análisis.	Realizar la busqueda del análisis.	1. Seleccionar número identificador (id) del análisis.	Equipo encendido y activo. Permisos para acceder a "Plan-consultar análisis clínico". Permisos para ejecutar el comando SELECT sobre la base de datos . Batería suficiente para realizar la acción.

El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	-	v1.0	-	07/11/2020	ggularite
El sistema carga en pantalla los datos pertenecientes al análisis clínico.	El esperado	-	v1.0	-	07/11/2020	ggularite

CP – Plan – Cons. Diagnóstico

Casos de prueba						
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	07/11/2020	zrosales	Buscar paciente(Paciente existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan- consultar diagnóstico". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Bateria suficiente para realizar la acción.
cp002	07/11/2020	zrosales	Buscar paciente(Paciente no existente).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan- Consultar diagnóstico". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Bateria suficiente para realizar la acción.
cp003	07/11/2020	zrosales	Buscar paciente(Paciente existente pero con una cédula distinta a la cédula del paciente que está consultando).	Realizar la búsqueda del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Plan- consultar diagnóstico". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Bateria suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para consultar un diagnóstico	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que solo puede buscar análisis propios y no de otra persona	El esperado	-	v1.0	-	07/11/2020	ggularde

cp004	07/11/2020	zrosales	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permiso para acceder a "Plan-Consultar Diagnóstico". Permiso para ejecuta el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp005	07/11/2020	zrosales	Buscar diagnóstico.	Realizar la búsqueda del diagnóstico.	1. Seleccionar número identificador (id) del diagnóstico.	Equipo encendido y activo. Permiso para acceder a "Plan-Consultar diagnóstico". Permiso para ejecuta el comando SELECT sobre la base de datos . Batería suficiente para realizar la acción.
			El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	v1.0	07/11/2020 ggularde
			El sistema cargará en pantalla los datos pertenecientes al diagnóstico.	El esperado	v1.0	07/11/2020 ggularde

CP – Adm. – Formularios

Casos de prueba			
Funcionalidad: ADMINISTRAR FORMULARIOS			

Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	Pasos	Entradas	Condiciones de ejecución
cp001	7/11/2020	ggularte	Crear formulario.	Realizar el alta de un formulario.	1. Ingresar el tipo de formulario (Obligatorio) 2. Pulsar el botón "Crear" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Administración-Formularios". Permiso para ejecutar el comando INSERT sobre la base de datos. Botón "Crear" habilitado. Batería suficiente para realizar la acción.
cp002	7/11/2020	ggularte	Crear pregunta .	Realizar el alta de una pregunta.	1. Ingresar el texto de la pregunta (Obligatorio) 2. Pulsar el botón "Crear" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Administración-Formularios". Permiso para ejecutar el comando INSERT sobre la base de datos. Botón "Crear" habilitado. Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema despliega una ventana emergente notificando que el formulario se creó exitosamente.	El esperado	-	v1.0	-	7/11/2020	ggularte
El sistema despliega una ventana emergente notificando que la pregunta se creó exitosamente.	El esperado	-	v1.0	-	7/11/2020	ggularte

cp003	7/11/2020	ggularte	Eliminar formulario	Realizar la baja de un formulario.	1. Seleccionar el tipo de formulario (Obligatorio) 2. Seleccionar el número identificador (ID) del formulario(Obligatorio) 3. Pulsar el botón "Eliminar formulario" (Obligatorio)	Equipo encendido y activo. Permiso para acceder a "Administración-Formularios". Permiso para ejecuta el comando UPDATE sobre la base de datos. Boton "Eliminar formulario" habilitado . Batería suficiente para realizar la acción.
cp004	7/11/2020	ggularte	Ingresar pregunta al formulario.	Realizar el alta de una pregunta a un formulario.	1. Seleccionar el tipo de formulario (Obligatorio) 2. Seleccionar el número identificador (ID) del formulario(Obligatorio) 3. Seleccionar la pregunta que se desea ingresar (Obligatorio) 4. Pulsar el botón "Ingresar pregunta al formulario"	Equipo encendido y activo. Permiso para acceder a "Administración-Formularios". Permiso para ejecuta el comando INSERT, SELECT sobre la base de datos . Boton "Ingresar pregunta al formulario" habilitado . Batería suficiente para realizar la acción.
				v1.0		7/11/2020 ggularte
				v1.0		7/11/2020 ggularte

cp005	7/11/2020	ggularde	Modificar pregunta	Realizar la modificación de la pregunta	1. Seleccionar la pregunta que se desea modificar (Obligatorio) 2. Ingresar el texto de la nueva pregunta (Obligatorio) 3. Pulsar el botón "Modificar pregunta"	Equipo encendido y activo. Permiso para acceder a "Administración-Formularios". Permiso para ejecutar el comando SELECT, UPDATE sobre la base de datos. Botón "Modificar pregunta" habilitado. Batería suficiente para realizar la acción.
cp006	7/11/2020	ggularde	Eliminar pregunta	Realizar la baja de una pregunta.	1. Seleccionar la pregunta que se desea modificar (Obligatorio) 2. Pulsar el botón "Eliminar pregunta"	Equipo encendido y activo. Permiso para acceder a "Administración-Formularios". Permiso para ejecutar el comando SELECT, DELETE sobre la base de datos. Botón "Eliminar pregunta" habilitado. Batería suficiente para realizar la acción.

El sistema despliega una ventana emergente notificando que la pregunta se modificó exitosamente.	El esperado	-	v1.0	-	7/11/2020	ggularde
El sistema despliega una ventana emergente notificando que la pregunta se eliminó exitosamente.	El esperado	-	v1.0	-	7/11/2020	ggularde

CP – Anamnesis – Entrevista inicial

Casos de prueba								
Funcionalidad: INGRESAR ENTREVISTA INICIAL								
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba					
cp001	7/11/2020	zrosales	Buscar paciente(Paciente existente) .					
				Pasos		Entradas		Condiciones de ejecución
				Realizar la búsqueda del paciente		1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)		Equipo encendido y activo. Permiso para acceder a "Anamnesis-Entrevista inicial". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.
cp002	7/11/2020	zrosales	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente		1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)		Equipo encendido y activo. Permiso para acceder a "Anamnesis-Entrevista inicial". Permiso para ejecuta el comando SELECT sobre la base de datos. Boton "Ingresar" habilitado . Batería suficiente para realizar la acción.

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para ingresar una entrevista inicial	El esperado	-	v1.0	-	7/11/2020	ggularte
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	7/11/2020	ggularte

cp003	7/11/2020	zrosales	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar". 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Anamnesis-Entrevista inicial". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp004	7/11/2020	zrosales	Ingresar entrevista inicial.	Realizar el ingreso de una entrevista inicial.	1. Seleccionar la fecha de la entrevista(Obligatorio). 2. Ingresar grado de colaboración (Obligatorio). 3. Ingresar movilidad (Obligatorio). 4. Seleccionar actitud (Obligatorio). 5. Seleccionar estado de nutrición (Obligatorio). 6. Ingresar estado de conciencia(Obligatorio). 7. Ingresar enfermedad actual (Opcional). 8. Ingresar antecedentes personales (Opcional). 9. Ingresar nombre de acompañante (Opcional). 10. Ingresar motivo de consulta (Obligatorio). 11. Ingresar temperatura (Obligatorio). 12. Seleccionar pulso (Obligatorio). 13. Ingresar frecuencia cardiaca (Obligatorio). 14. Ingresar frecuencia respiratoria(Obligatorio). 15. Ingresar grado de hidratación (Obligatorio). 16. Ingresar enfermedad familiar (Opcional). 17. Ingresar tipo de enfermedad actual (Opcional). 18. Ingresar tipo de enfermedad anterior (Opcional). 19. Ingresar C I Medico (Obligatorio). 20. Seleccionar síntomas que padece el paciente. 21. Pulsar botón ingresar.	Equipo encendido y activo. Permisos para acceder a "Anamnesis-Entrevista inicial". Permisos para ejecutar el comando SELECT sobre la base de datos . Batería suficiente para realizar la acción.
			El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.	El esperado	v1.0	7/11/2020 ggularde
			El sistema desplegará una ventana emergente que notificará que la entrevista se ingresó exitosamente.	El esperado	v1.0	7/11/2020 ggularde

CP – Adm. – Diagnóstico

Casos de prueba				
Funcionalidad: ADMINISTRAR DIAGNÓSTICOS				
Id	Fecha de creación	Nombre del autor	Resumen del Caso de prueba	
cp001	07/11/2020	ggularde	Buscar paciente(Paciente existente) .	Realizar la búsqueda del paciente 1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)
cp002	07/11/2020	ggularde	Buscar paciente(Paciente no existente) .	Realizar la búsqueda del paciente 1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" (Obligatorio)

Resultado esperado	Resultado obtenido	Ambiente	Versión	Identificación incidente	Fecha de ejecución	Nombre del ejecutante
El sistema habilitará una agrupación de datos para administrar un diagnóstico	El esperado	-	v1.0	-	07/11/2020	ggularde
El sistema despliega una ventana emergente notificando que el paciente no se encontró y si se desea realizar el alta del mismo.	El esperado	-	v1.0	-	07/11/2020	ggularde

cp003	07/11/2020	ggularte	Buscar paciente(Que esta dado de baja).	Realizar la búsqueda del paciente. Realizar el alta del paciente	1. Ingresar cédula de identidad del paciente (Obligatorio) 2. Pulsar el botón "Buscar" 3. Confirmar que se desea realizar el alta del paciente.	Equipo encendido y activo. Permisos para acceder a "Administración-Diagnosticos". Permisos para ejecutar el comando UPDATE, SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp004	07/11/2020	ggularte	Buscar diagnóstico.	Realizar la consulta de un diagnóstico.	1. Seleccionar el número identificador (ID) del diagnóstico. (Obligatorio)	Equipo encendido y activo. Permisos para acceder a "Administración-Diagnosticos". Permisos para ejecutar el comando SELECT sobre la base de datos . Batería suficiente para realizar la acción.
cp005	07/11/2020	ggularte	Modificar diagnóstico.	Realizar la modificación de los datos pertenecientes al diagnóstico.	1. Modificar los datos que se desean cambiar (obligatorio) 2. Pulsar botón modificar	Equipo encendido y activo. Permisos para acceder a "Administración - Diagnosticos". Permisos para ejecutar el comando UPDATE sobre la base de datos. Botón "Modificar" habilitado . Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el paciente se dio de alta exitosamente.		El esperado		v1.0	-	07/11/2020 ggularte
El sistema cargará en pantalla los datos pertenecientes al diagnóstico con el ID seleccionado anteriormente.		El esperado		v1.0	-	07/11/2020 ggularte
El sistema despliega una ventana emergente notificando que los datos se modificaron exitosamente.		El esperado		v1.0	-	07/11/2020 ggularte

cp006	07/11/2020	ggularde	Eliminar diagnóstico.	Realizar la baja del diagnóstico.	1. Pulsar botón eliminar	Equipo encendido y activo. Permisos para acceder a "Administración - Diagnósticos". Permisos para ejecutar el comando UPDATE sobre la base de datos. Botón "Eliminar" habilitado. Batería suficiente para realizar la acción.
El sistema despliega una ventana emergente notificando que el diagnóstico se dio de baja exitosamente.		El esperado		v1.0	-	07/11/2020 ggularde

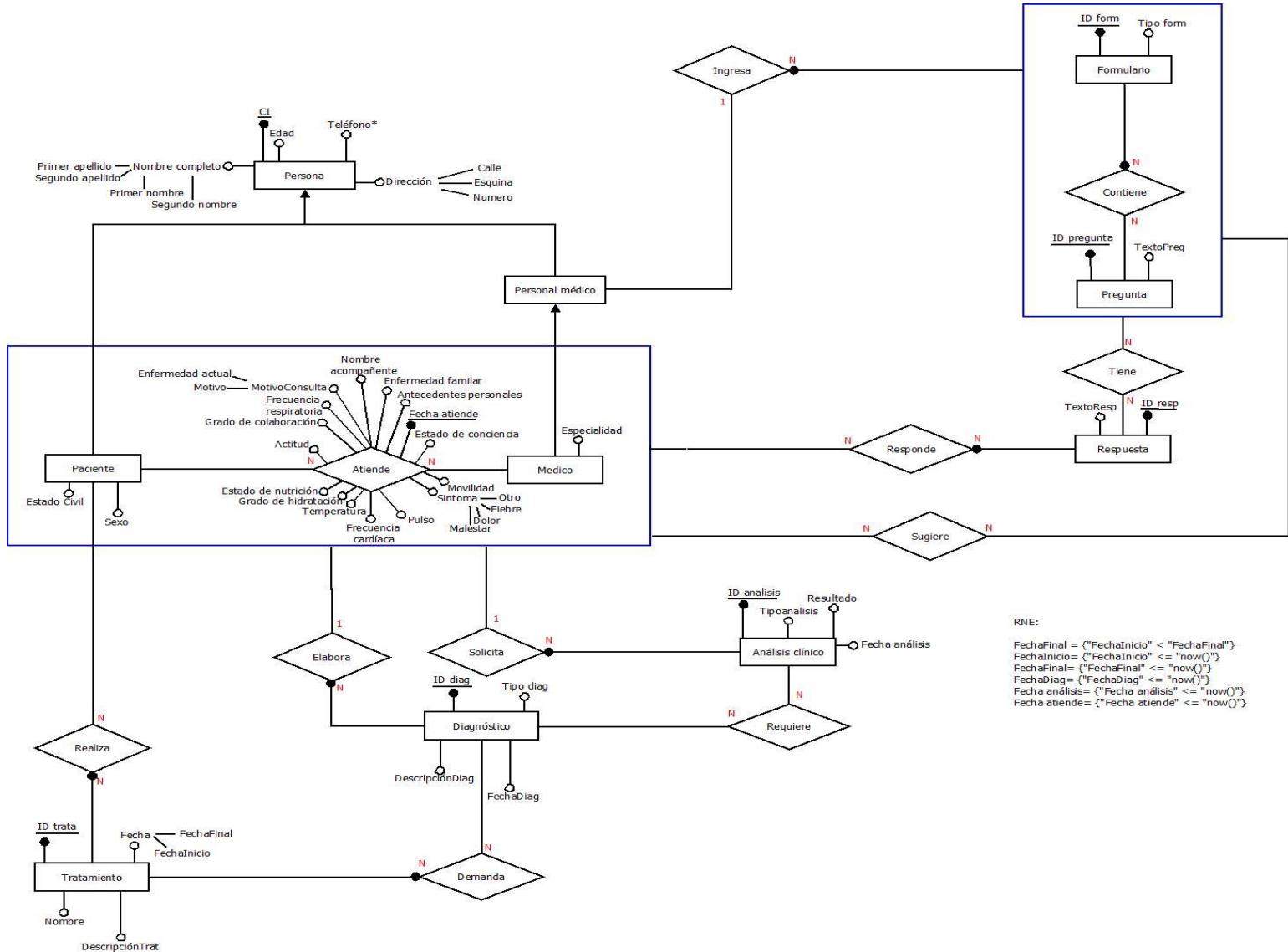


Agenda de reuniones

Ver anexos (ANÁLISIS Y DISEÑO DE APLICACIONES: Anexo II
Agenda de reuniones)

Requisitos de la asignatura Base de Datos

DER (Primera entrega)



RNE

FechaFinal= ("FechaInicio" < "FechaFinal")

FechaInicio=(“FechaInicio” <= “now()”)

FechaFinal=(“FechaFinal” <= “now()”)

FechaDiag= (“FechaDiag” <= “now()”)

Fecha análisis= (“Fecha análisis” <= “now()”)

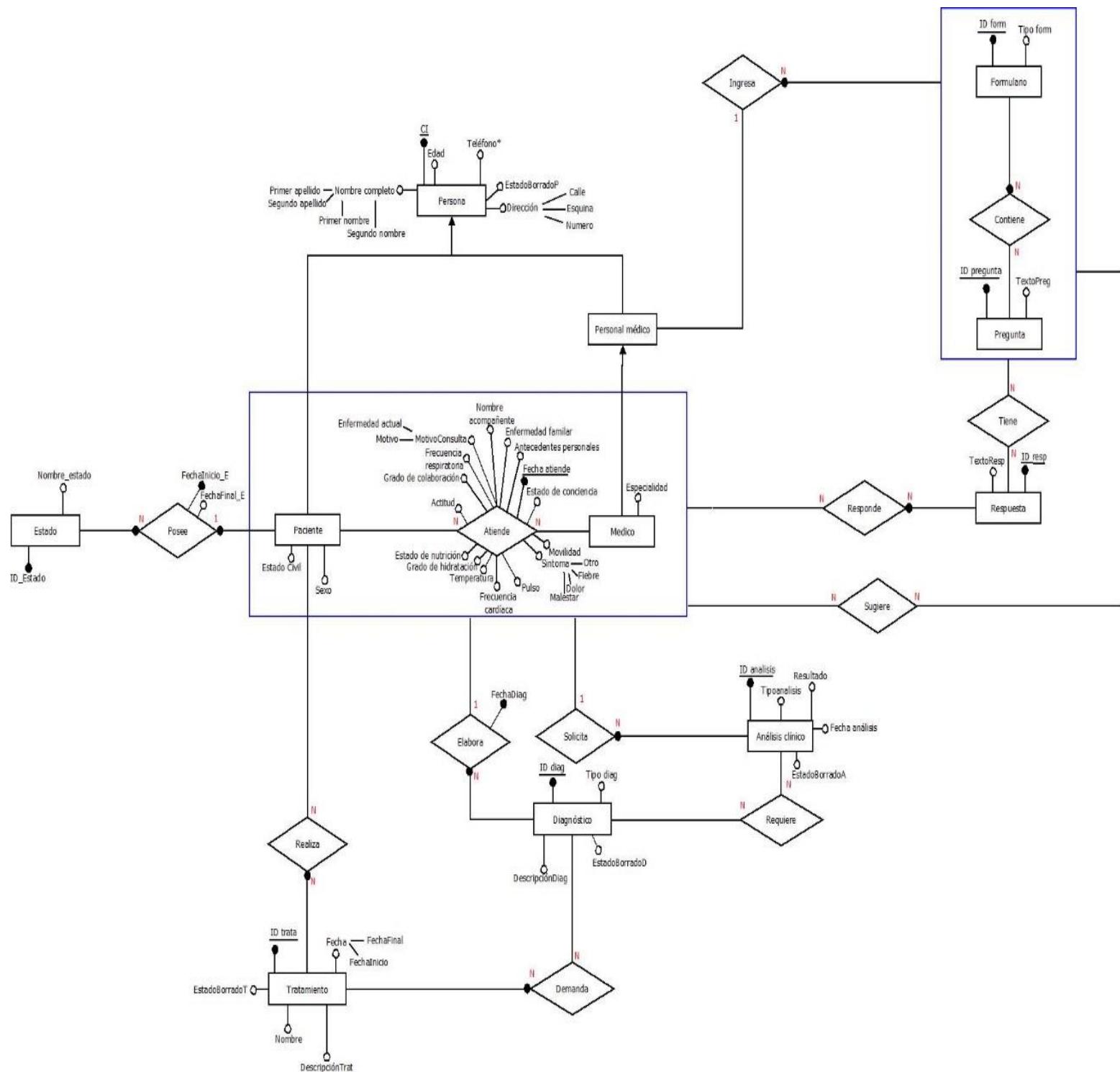
Fecha atiende= (“Fecha atiende” <= “now()”)

Diccionario de Datos

Atributo	Descripción	Tipo	PK	NULL
Actitud	Actitud del paciente a la hora de atenderse con un médico.	VARCHAR(10)		NOT NULL
Antecedentes personales	Antecedentes personales que el paciente haya sufrido.	VARCHAR(55)		
Calle	Calle de la dirección perteneciente a la persona.	VARCHAR(25)		NOT NULL
CI	Cédula de identidad perteneciente a la persona.	INTEGER	PK	NOT NULL
DescripciónDiag	Descripción perteneciente al diagnóstico realizado por un médico sobre un paciente.	VARCHAR(55)		NOT NULL
DescripciónTrat	Descripción perteneciente al tratamiento realizado por un paciente.	VARCHAR(55)		NOT NULL
Dolor	Síntoma del paciente que registra el sistema	BOOLEAN		NOT NULL
Edad	Edad perteneciente a la persona.	INTEGER		NOT NULL
Enfermedad actual	Enfermedad actual del paciente consultante	VARCHAR(55)		
Enfermedad familiar	Enfermedad familiar del paciente consultante.	VARCHAR(55)		
Especialidad	Especialidad del médico.	VARCHAR(25)		NOT NULL
Esquina	Esquina de la dirección perteneciente a la persona.	VARCHAR(25)		NOT NULL
Estado Civil	Estado Civil del paciente consultante.	VARCHAR(12)		NOT NULL
Estado de conciencia	Estado de conciencia del paciente consultante.	VARCHAR(20)		NOT NULL
Estado de nutrición	Estado de nutrición del paciente consultante	VARCHAR(8)		NOT NULL
Fecha análisis	Fecha en la que el paciente se realiza un análisis.	DATETIME		NOT NULL
Fecha atiende	Fecha en la que el paciente realiza una consulta con el médico.	DATETIME	PK	NOT NULL
FechaDiag	Fecha en la que el médico elabora un diagnóstico sobre el paciente.	DATETIME		NOT NULL
FechaFinal	Fecha en la que el tratamiento finaliza.	DATETIME		
FechaInicio	Fecha en la que el tratamiento comienza.	DATETIME		NOT NULL
Fiebre	Síntoma del paciente que registra el sistema	BOOLEAN		NOT NULL
Frecuencia cardíaca	Frecuencia cardíaca del paciente a la hora de consultar con el médico.	VARCHAR(20)		NOT NULL
Frecuencia respiratoria	Frecuencia respiratoria del paciente a la hora de consultar con el médico.	VARCHAR(20)		NOT NULL
Grado de colaboración	Grado de colaboración del paciente a la hora de consultar con el médico.	VARCHAR(20)		NOT NULL
Grado de hidratación	Grado de hidratación del paciente a la hora de consultar con el médico.	VARCHAR(20)		NOT NULL
ID análisis	ID perteneciente al análisis solicitado por el médico sobre un paciente.	INTEGER	PK	NOT NULL
ID diag	ID perteneciente al diagnóstico elaborado por el médico sobre un paciente.	INTEGER	PK	NOT NULL

ID form	ID perteneciente al formulario elaborado por el médico sobre un paciente.	INTEGER	PK	NOT NULL
ID pregunta	ID perteneciente a la pregunta perteneciente a un formulario.	INTEGER	PK	NOT NULL
ID resp	ID perteneciente a la respuesta de la pregunta del formulario.	INTEGER	PK	NOT NULL
ID trata	ID perteneciente al tratamiento realizado por el paciente.	INTEGER	PK	NOT NULL
Malestar	Síntoma del paciente que registra el sistema	BOOLEAN		NOT NULL
Motivo	Motivo por el cual el paciente se atiende con el médico.	VARCHAR(20)		NOT NULL
Movilidad	Movilidad del paciente a la hora de consultar con el médico.	VARCHAR(20)		NOT NULL
Nombre acompañante	Nombre del acompañante del paciente a la hora de consultar con el médico.	VARCHAR(20)		
Número	Número de la calle perteneciente a la persona.	VARCHAR(10)		NOT NULL
Otro	Otro síntoma que tenga el paciente a la hora de consultar con el médico.	VARCHAR(20)		
Primer apellido	Primer apellido perteneciente a la persona.	VARCHAR(20)		NOT NULL
Primer nombre	Primer nombre perteneciente a la persona.	VARCHAR(20)		NOT NULL
Pulso	Pulso del paciente a la hora de consultar con el médico.	VARCHAR(10)		NOT NULL
Resultado	Resultado del análisis que solicita un médico sobre un paciente.	VARCHAR(55)		NOT NULL
Segundo apellido	Segundo apellido perteneciente a la persona.	VARCHAR(20)		NOT NULL
Segundo nombre	Segundo nombre perteneciente a la persona.	VARCHAR(20)		
Sexo	Sexo perteneciente a la persona.	CHAR(1)		NOT NULL
Teléfono	Teléfono perteneciente a la persona.	VARCHAR(9)		NOT NULL
Temperatura	Temperatura del paciente a la hora de consultar con el médico.	DOUBLE		NOT NULL
TextoPreg	Texto de la pregunta de un formulario.	VARCHAR(55)		NOT NULL
TextoResp	Texto de la respuesta perteneciente a la pregunta de un formulario.	VARCHAR(55)		NOT NULL
Tipo análisis	Tipo de análisis que solicita un médico sobre un paciente.	VARCHAR(30)		NOT NULL
Tipo diag	Tipo de diagnóstico que elabora un médico sobre un paciente.	VARCHAR(25)		NOT NULL
Tipo form	Tipo de formulario que ingresa el personal médico.	VARCHAR(25)		NOT NULL

DER (Segunda entrega):



RNE:

```
FechaFinal = {"Fechalnicio" < "FechaFinal"}  
Fechalnicio = {"Fechalnicio" <= "now()"}  
FechaFinal = {"Fechalnicio" <= "now()"}  
FechaDiag = {"FechaDiag" <= "now()"}  
Fecha análisis = {"Fecha análisis" <= "now()"}  
Fecha atiende = {"Fecha atiende <= "now()"}  
FechaFinal_E = {"Fechalnicio_E" < "FechaFinal_E"}  
FechaFinal = {"FechaFinal_E" <= "now()"}
```

Diccionario de datos:

DICCCIONARIO DE DATOS DE LA BASE DE DATOS		ProyectoGSA													
TABLAS CORRESPONDIENTES A ENTIDADES y/o RELACIONES															
Nombre de entidad y tipo	Respuesta, Entidad.														
Objetivo	Es una entidad en la cual se registran las respuestas con el texto de las mismas y su ID correspondiente que responde a una pregunta que está dentro de un formulario.														
Metadatos:															
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido							
ID Resp	INTEGER		SI	NO	NO		hacia tabla hacia atributo	Numero de identificación (ID) perteneciente a la respuesta de la pregunta del formulario.							
TextoResp	VARCHAR	55	NO	NO	NO			Texto de la respuesta perteneciente a la pregunta de un formulario.							
Nombre de entidad y tipo	Analisis_clinico, Entidad.														
Objetivo	Es una entidad en la cual se registran análisis clínicos con sus datos como ID análisis, Tipo Analysis, Resultado y Fecha análisis.														
Metadatos:															
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido							
Cl_Medico	INTEGER		NO	NO	NO		Atiende Cl_Medico	Número de identidad del documento (Cl) perteneciente al médico.							
Cl_Paciente	INTEGER		NO	NO	NO		Atiende Cl_Paciente	Número de identidad del documento (Cl) perteneciente al paciente.							
Fecha_Analisis	DATETIME		NO	NO	NO			Fecha con formato (DD-MM-AAAA) y hora en la que el paciente se realiza un análisis.							
Fecha_Atiende	DATETIME		NO	NO	NO		Atiende Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.							
ID_Analisis	INTEGER		SI	NO	NO			Numero de identificación (ID) perteneciente al análisis solicitado por el médico sobre un paciente.							
Resultado	VARCHAR	55	NO	NO	NO			Resultado del análisis que solicita un médico sobre un paciente.							
TipoAnalisis	VARCHAR	30	NO	NO	NO			Tipo de análisis que solicita un médico sobre un paciente.							
EstadoBorradoA	BOOLEAN		NO	NO	NO			Estado que determina si un análisis clínico está habilitado o no (Baja lógica).							

Nombre de relación y tipo	Realiza, Relación.							
Objetivo	Es una relación en la cual se registran datos como la Cl del Paciente y la ID de un Tratamiento, que corresponde al tratamiento que se realiza un paciente.							
Metadatos:								
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido
Cl_Paciente	INTEGER		SI	NO	NO		Paciente Cl_Paciente	Número de identidad del documento (Cl) perteneciente al paciente.
ID_Trata	INTEGER		SI	NO	NO		Tratamiento ID_Trata	Numero de identificación (ID) perteneciente al tratamiento realizado por el paciente.
Nombre de relación y tipo	Demanda, Relación.							
Objetivo	Es una relación en la cual se registran datos como el ID de un Diagnóstico junto con el ID de un Tratamiento, que corresponde al tratamiento demandado por un diagnóstico.							
Metadatos:								
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido
ID_Diag	INTEGER		SI	NO	NO		Diagnostico ID_Diag	Numero de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.
ID_Trata	INTEGER		SI	NO	NO		Tratamiento ID_Trata	Numero de identificación (ID) perteneciente al tratamiento realizado por el paciente
Nombre de relación y tipo	Atiende, Relación (Agregación).							
Objetivo	Es una relación en la cual se registran datos de una entidad inicial que un médico le realiza a un paciente.							
Metadatos:								
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido
Actitud	CHECK		NO	NO	NO			Actitud del paciente a la hora de atenderse con un médico. (Coma, Apatía, Normal y Excitación)
Antecedentes_Personales	VARCHAR	55	NO	NO	SI			Antecedentes personales que el paciente haya sufrido.
Cl_Medico	INTEGER		SI	NO	NO		Medico Cl_Medico	Número de identidad del documento (Cl) perteneciente al médico.
Cl_Paciente	INTEGER		SI	NO	NO		Paciente Cl_Paciente	Número de identidad del documento (Cl) perteneciente al paciente.
Dolor	BOOLEAN		NO	NO	NO			Síntoma del paciente que registra el sistema.
Enfermedad_Actual	VARCHAR	55	NO	NO	SI			Enfermedad actual del paciente consultante.
Enfermedad_Familiar	VARCHAR	55	NO	NO	SI			Enfermedad familiar del paciente consultante.
Estado_De_Conciencia	VARCHAR	20	NO	NO	NO			Estado de conciencia del paciente consultante.
Estado_De_Nutricion	CHECK		NO	NO	NO			Estado de nutrición del paciente consultante. (Obesidad, Común, Anorexia, Cauquexia)
Fecha_Atiende	DATETIME		SI	NO	NO			Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.
Fiebre	BOOLEAN		NO	NO	NO			Síntoma del paciente que registra el sistema.
Frecuencia_Cardiaca	VARCHAR	20	NO	NO	NO			Frecuencia cardiaca del paciente a la hora de consultar con el médico.
Frecuencia_Respiratoria	VARCHAR	20	NO	NO	NO			Frecuencia respiratoria del paciente a la hora de consultar con el médico.
Grado_De_Colaboracion	VARCHAR	20	NO	NO	NO			Grado de colaboración del paciente a la hora de consultar con el médico.
Grado_De_Hidratacion	VARCHAR	20	NO	NO	NO			Grado de hidratación del paciente a la hora de consultar con el médico.
Malestar	BOOLEAN		NO	NO	NO			Síntoma del paciente que registra el sistema.
Motivo	VARCHAR	20	NO	NO	NO			Motivo por el cual el paciente se atiende con el médico.
Movilidad	VARCHAR	20	NO	NO	NO			Motivo por el cual el paciente se atiende con el médico.
Nombre_Acompañante	VARCHAR	20	NO	NO	SI			Nombre del acompañante del paciente a la hora de consultar con el médico.
Otro	VARCHAR	20	NO	NO	SI			Otro síntoma que tenga el paciente a la hora de consultar con el médico.
Pulso	CHECK		NO	NO	NO			Pulso del paciente a la hora de consultar con el médico. (Fuerte, Normal, Débil, Filiforme)
Temperatura	DOUBLE		NO	NO	NO			Temperatura del paciente a la hora de consultar con el médico.

Nombre de entidad y tipo	Persona, Entidad.											
Objetivo	Es una entidad la cual registra los datos de las personas tanto de paciente como de funcionarios médicos de la institución.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
Calle	VARCHAR	25	NO	NO	NO				Calle de la dirección perteneciente a la persona.			
CI	INTEGER		SI	NO	NO				Número de identidad del documento (CI) perteneciente a la persona.			
Fecha_Nac	DATE		NO	NO	NO				Fecha de nacimiento perteneciente a la persona.			
Esquina	VARCHAR	25	NO	NO	NO				Esquina de la dirección perteneciente a la persona.			
Número	INTEGER		NO	NO	NO				Número de la calle perteneciente a la persona.			
Primer_Apellido	VARCHAR	20	NO	NO	NO				Primer apellido perteneciente a la persona.			
Primer_Nombre	VARCHAR	20	NO	NO	NO				Primer nombre perteneciente a la persona.			
Segundo_Apellido	VARCHAR	20	NO	NO	NO				Segundo apellido perteneciente a la persona.			
Segundo_Nombre	VARCHAR	20	NO	NO	SI				Segundo nombre perteneciente a la persona.			
EstadoBorradoP	BOOLEAN		NO	NO	NO				Estado que determina si una persona sigue perteneciendo a la institución médica (Baja lógica).			
Nombre de entidad y tipo	Personal_Medico, Entidad.											
Objetivo	Es una entidad que hereda de Persona y representa a todo el personal médico de la institución (Médicos y Auxiliares Médicos).											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
CI	INTEGER		SI	NO	NO		Personas CI		Número de identidad del documento (CI) perteneciente a la persona.			
Nombre de entidad y tipo	Paciente, Entidad.											
Objetivo	Es una entidad que hereda de Persona y registra datos propios de los pacientes como el Estado Civil o el Sexo.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
CI_Paciente	INTEGER		SI	NO	NO		Personas CI		Número de identidad del documento (CI) perteneciente al paciente.			
Estado_Civil	CHECK		NO	NO	NO				Estado Civil del paciente consultante. (Soltero/a, Casado/a, Divorciado/a, Viudo/a)			
Sexo	CHECK		NO	NO	NO				Sexo perteneciente a la persona. (F (Femenino), M(Masculino))			
Nombre de entidad y tipo	Persona_Teléfono, Entidad.											
Objetivo	Es una entidad generada por el atributo multivaluado "Teléfono" de Persona la cual registrá todas los teléfonos de las personas junto con la CI correspondiente.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
CI	INTEGER		SI	NO	NO		Personas CI		Número de identidad del documento (CI) perteneciente a la persona.			
Teléfono	VARCHAR	9	SI	NO	NO				Teléfono perteneciente a la persona.			
Nombre de entidad y tipo	Formulario, Entidad.											
Objetivo	Es una entidad en la cual se registran datos de un formulario como su ID y el Tipo de Formulario que es.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Form	INTEGER		SI	NO	NO				Número de identificación (ID) perteneciente al formulario elaborado por el médico sobre un paciente.			
Tipo_Form	VARCHAR	25	NO	NO	NO				Tipo de formulario que ingresa el personal médico.			
Nombre de relación y tipo	Tiene, Relación.											
Objetivo	Es una relación la cual enlaza las preguntas que contiene un formulario con sus respuestas.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Form	INTEGER		SI	NO	NO		Formularios ID_Form		Número de identificación (ID) perteneciente al formulario elaborado por el médico sobre un paciente.			
ID_Preg	INTEGER		SI	NO	NO		Preguntas ID_Preg		Número de identificación (ID) perteneciente a la pregunta perteneciente a un formulario.			
ID_Resp	INTEGER		SI	NO	NO		Respuestas ID_Resp		Número de identificación (ID) perteneciente a la respuesta de la pregunta del formulario.			

Nombre de relación y tipo	Elabora, Relación.								
Objetivo	Es la relación que registra el diagnóstico elaborado por un médico sobre un paciente.								
Metadatos:									
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key hacia tabla	Foreign Key hacia atributo	Descripción del contenido
ID_Diag	INTEGER		SI	NO	NO		Diagnostico	ID_Diag	Numero de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.
Fecha_Diag	DATETIME		SI	NO	NO				Fecha con formato (DD-MM-AAAA) en la que el médico realiza un diagnóstico sobre un paciente.
Fecha_Atiende	DATETIME		NO	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.
CI_Medico	INTEGER		NO	NO	NO		Atiende	CI_Medico	Número de identidad del documento (CI) perteneciente al médico.
CI_Paciente	INTEGER		NO	NO	NO		Atiende	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.
Nombre de relación y tipo	Posee, Relación.								
Objetivo	Es la relación que registra los datos del estado en el que se encuentra un paciente en la institución médica.								
Metadatos:									
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key hacia tabla	Foreign Key hacia atributo	Descripción del contenido
ID_Estado	INTEGER		SI	NO	NO		Estado	ID_Estado	Numero de identificación (ID) perteneciente al estado existente de un paciente en la institución médica.
CI_Paciente	INTEGER		SI	NO	NO		Paciente	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.
FechaInicio_E	DATETIME		SI	NO	NO				Fecha con formato (DD-MM-AAAA) en la que el paciente entra en un estado.
FechaFinal_E	DATETIME		NO	NO	SI				Fecha con formato (DD-MM-AAAA) en la que el paciente finaliza (cambia) un estado.
Nombre de entidad y tipo	Medico, Entidad.								
Objetivo	Es una entidad que hereda de Personal Médico y registra datos como la Especialidad perteneciente a un médico junto con su CI.								
Metadatos:									
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key hacia tabla	Foreign Key hacia atributo	Descripción del contenido
CI_Medico	INTEGER		SI	NO	NO		Personal_Medico	CI	Número de identidad del documento (CI) perteneciente al médico.
Especialidad	VARCHAR	25	NO	NO	NO				Especialidad del médico.
Nombre de entidad y tipo	Diagnostico, Entidad.								
Objetivo	Es una entidad la cual registra todos los datos correspondientes a los diagnóstico elaborados por un médico sobre varios pacientes.								
Metadatos:									
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key hacia tabla	Foreign Key hacia atributo	Descripción del contenido
Descripcion_Diag	VARCHAR	55	NO	NO	NO				Descripción perteneciente al diagnóstico realizado por un médico sobre un paciente.
ID_Diag	INTEGER		SI	NO	NO				Numero de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.
Tipo_Diag	VARCHAR	25	NO	NO	NO				Tipo de diagnóstico que elabora un médico sobre un paciente.
EstadoBorradoD	BOOLEAN		NO	NO	NO				Estado que determina si un diagnóstico está habilitado o no (Baja lógica).
Nombre de entidad y tipo	Tratamiento, Entidad.								
Objetivo	Es una entidad la cual registra datos correspondientes de un tratamiento que se le realiza a un paciente.								
Metadatos:									
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key hacia tabla	Foreign Key hacia atributo	Descripción del contenido
Descripcion_Trat	VARCHAR	55	NO	NO	NO				Descripción perteneciente al tratamiento realizado por un paciente.
FechaFinal	DATETIME		NO	NO	SI				Fecha con formato (DD-MM-AAAA) en la que el tratamiento finaliza.
FechaInicio	DATETIME		NO	NO	NO				Fecha con formato (DD-MM-AAAA) en la que el tratamiento comienza.
ID_Tray	INTEGER		SI	NO	NO				Numero de identificación (ID) perteneciente al tratamiento realizado por el paciente.
Nombre	VARCHAR	30	NO	NO	NO				Nombre perteneciente al diagnóstico elaborado por el médico sobre un paciente.
EstadoBorradoT	BOOLEAN		NO	NO	NO				Estado que determina si un tratamiento está habilitado o no (Baja lógica).

Nombre de entidad y tipo	<i>Pregunta, Entidad.</i>											
Objetivo	<i>Es una entidad la cual registra datos como la ID de una Pregunta y su texto correspondiente.</i>											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Preg	INTEGER		SI	NO	NO		hacia tabla	hacia atributo				
TextoPreg	VARCHAR	55	NO	NO	NO				Número de identificación (ID) perteneciente a la pregunta perteneciente a un formulario.			
<i>Texto de la pregunta de un formulario.</i>												
Nombre de relación y tipo	<i>Sugiere, Relación.</i>											
Objetivo	<i>Es la relación la cual enlaza los formularios que contienen preguntas y se sugieren a partir de la entrevista inicial realizada entre un paciente y un médico.</i>											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
CI_Medico	INTEGER		SI	NO	NO		Atiende	CI_Medico				
CI_Paciente	INTEGER		SI	NO	NO		Atiende	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.			
Fecha_Atiende	DATETIME		SI	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.			
ID_Form	INTEGER		SI	NO	NO		Contiene	ID_Form	Número de identificación (ID) perteneciente al formulario elaborado por el médico sobre un paciente.			
ID_Preg	INTEGER		SI	NO	NO		Contiene	ID_Preg	Número de identificación (ID) perteneciente a la pregunta perteneciente a un formulario.			
Nombre de relación y tipo	<i>Responde, Relación.</i>											
Objetivo	<i>Es la relación la cual enlaza las respuestas dadas por un paciente de un formulario llenado por un médico.</i>											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
CI_Medico	INTEGER		SI	NO	NO		Atiende	CI_Medico				
CI_Paciente	INTEGER		SI	NO	NO		Atiende	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.			
Fecha_Atiende	DATETIME		SI	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.			
ID_Resp	INTEGER		SI	NO	NO		Respuesta	ID_Resp	Número de identificación (ID) perteneciente a la respuesta de la pregunta del formulario.			
Nombre de relación y tipo	<i>Requiere, Relación.</i>											
Objetivo	<i>Es una relación la cual enlaza el diagnóstico que requiere de un análisis clínico para culminarse.</i>											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Analisis	INTEGER		SI	NO	NO		Analisis_Clinico	ID_Analisis				
ID_Diag	INTEGER		SI	NO	NO		Diagnostico	ID_Diag	Número de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.			
Nombre de relación y tipo	<i>Contiene, Relación (Agregación).</i>											
Objetivo	<i>Es la relación que representa las preguntas que contiene un formulario.</i>											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Form	INTEGER		SI	NO	NO		Formulario	ID_Form				
ID_Preg	INTEGER		SI	NO	NO		Pregunta	ID_Preg	Número de identificación (ID) perteneciente a la pregunta perteneciente a un formulario.			
Nombre de entidad y tipo	<i>Estado, Entidad.</i>											
Objetivo	<i>Es la entidad que registra todos los posibles estados junto con su ID en los que se puede encontrar un paciente en la institución médica.</i>											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Estado	INTEGER		SI	NO	NO							
Nombre_Estado	VARCHAR	25	NO	NO	NO				Número de identificación (ID) perteneciente al estado existente de un paciente en la institución médica.			
									Nombre perteneciente al estado en el que se encuentra un paciente.			

Permisos de usuarios:

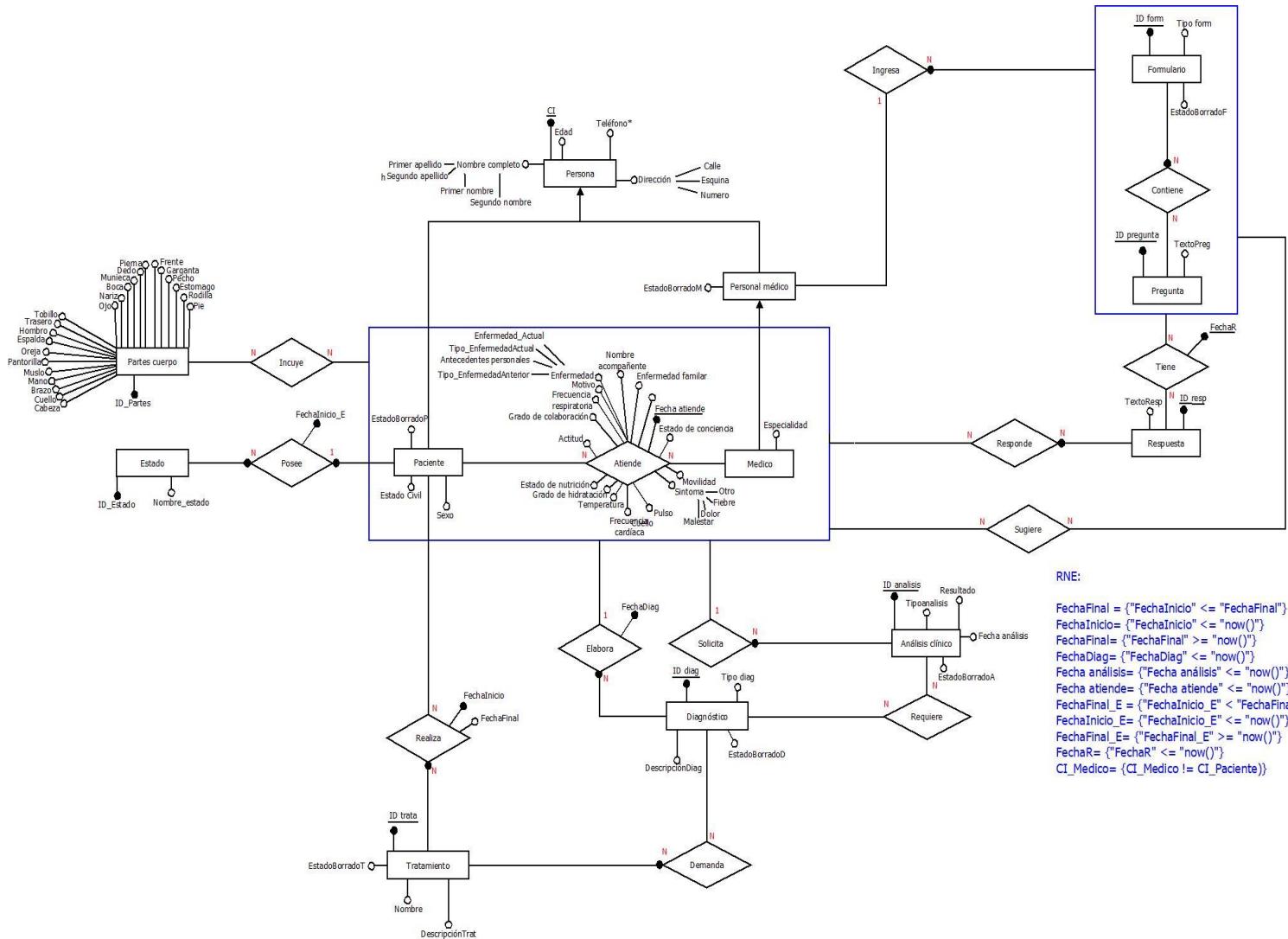
Respuesta				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X		
GSA_Aux		X		
GSA_Pac				
Analisis_clinico				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux	X	X		
GSA_Pac	X			
Realiza				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux	X	X		
GSA_Pac	X			
Demanda				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux				
GSA_Pac				
Atiende				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux	X	X	X	
GSA_Pac	X			
Persona				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X			
GSA_Aux	X			
GSA_Pac	X			

Personal_Medico				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med				
GSA_Aux				
GSA_Pac				
Paciente				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X			
GSA_Aux	X			
GSA_Pac	X			
Persona_Teléfono				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X			
GSA_Aux	X			
GSA_Pac				
Formulario				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux	X	X	X	
GSA_Pac	X			
Tiene				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	X
GSA_Aux	X	X	X	X
GSA_Pac				
Medico				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med				
GSA_Aux				
GSA_Pac				

Diagnóstico				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux		X	X	
GSA_Pac	X			
Tratamiento				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux	X	X	X	
GSA_Pac	X			
Pregunta				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	X
GSA_Aux	X	X	X	X
GSA_Pac				
Sugiere				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux	X	X	X	
GSA_Pac				
Responde				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux				
GSA_Pac				
Requiere				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux				
GSA_Pac				

Contiene				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	X
GSA_Aux	X	X	X	X
GSA_Pac				
Estado				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux				
GSA_Pac				
Elabora				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux				
GSA_Pac	X			
Posee				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X	X	
GSA_Aux	X	X	X	
GSA_Pac				
PartesCuerpo				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X		
GSA_Aux				
GSA_Pac				
Incluye				
Usuario	SELECT	INSERT	UPDATE	DELETE
GSA_Adm	X	X	X	X
GSA_Med	X	X		
GSA_Aux				
GSA_Pac				

DER Final(Tercera entrega):



RNE:

```

FechaFinal = {"FechaInicio" < "FechaFinal"}
FechaInicio = {"FechaInicio" <= "now()"}
FechaFinal = {"now()" <= "FechaFinal_E"}
FechaDiag = {"FechaDiag" <= "now()"}
Fecha análisis = {"Fecha análisis" <= "now()"}
Fecha atiende = {"Fecha atiende" <= "now()"}
FechaFinal_E = {"FechaInicio_E" < "FechaFinal_E"}
FechaInicio_E = {"FechaInicio_E" <= "now()"}
FechaFinal_E = {"now()" <= "FechaFinal_E"}
FechaR = {"FechaR" <= "now()"}
CI_Medico = {"CI_Medico != CI_Paciente"}
    
```

Diccionario de datos:

DICIONARIO DE DATOS DE LA BASE DE DATOS							ProyectoGSA				
TABLAS CORRESPONDIENTES A ENTIDADES y/o RELACIONES											
Nombre de entidad y tipo	Respuesta, Entidad.										
Objetivo	Es una entidad en la cual se registran las respuestas con el texto de las mismas y su ID correspondiente que responde a una pregunta que está dentro de un formulario.										
Metadatos:											
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido			
ID Resp	INTEGER		SI	NO	NO		hacia tabla				
TextoResp	VARCHAR	160	NO	NO	NO		hacia atributo	Número de identificación (ID) perteneciente a la respuesta de la pregunta del formulario. Texto de la respuesta perteneciente a la pregunta de un formulario.			
Nombre de entidad y tipo	Analisis_clinico, Entidad.										
Objetivo	Es una entidad en la cual se registran análisis clínicos con sus datos como ID análisis, Tipo Analisis, Resultado y Fecha analisis.										
Metadatos:											
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido			
CI Medico	INTEGER		NO	NO	NO		Atiende	CI Medico	Número de identidad del documento (CI) perteneciente al médico.		
CI Paciente	INTEGER		NO	NO	NO		Atiende	CI Paciente	Número de identidad del documento (CI) perteneciente al paciente.		
Fecha_Analisis	DATETIME		NO	NO	NO			Fecha con formato (DD-MM-AAAA) y hora en la que el paciente se realiza un análisis.			
Fecha_Atiende	DATETIME		NO	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.		
ID Analisis	INTEGER		SI	NO	NO			Número de identificación (ID) perteneciente al análisis solicitado por el médico sobre un paciente.			
Resultado	VARCHAR	240	NO	NO	NO			Resultado del análisis que solicita un médico sobre un paciente.			
TipoAnalisis	VARCHAR	50	NO	NO	NO			Tipo de análisis que solicita un médico sobre un paciente.			
EstadoBorradoA	BOOLEAN		NO	NO	NO			Estado que determina si un análisis clínico está habilitado o no (Baja lógica).			
Nombre de relación y tipo	Realiza, Relación.										
Objetivo	Es una relación en la cual se registran datos como la CI del Paciente y la ID de un Tratamiento, que corresponde al tratamiento que se realiza un paciente.										
Metadatos:											
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido			
CI Paciente	INTEGER		SI	NO	NO		Paciente	CI Paciente	Número de identidad del documento (CI) perteneciente al paciente.		
FechaFinal	DATETIME		NO	NO	SI			Fecha con formato (DD-MM-AAAA) en la que el tratamiento finaliza.			
FechaInicio	DATETIME		SI	NO	NO			Fecha con formato (DD-MM-AAAA) en la que el tratamiento comienza.			
ID Trata	INTEGER		SI	NO	NO		Tratamiento	ID Trata	Número de identificación (ID) perteneciente al tratamiento realizado por el paciente.		
Nombre de relación y tipo	Demanda, Relación.										
Objetivo	Es una relación en la cual se registran datos como el ID de un Diagnóstico junto con el ID de un Tratamiento, que corresponde al tratamiento demandado por un diagnóstico.										
Metadatos:											
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido			
ID Diaq	INTEGER		SI	NO	NO		Diagnóstico	ID Diaq	Número de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.		
ID Trata	INTEGER		SI	NO	NO		Tratamiento	ID Trata	Número de identificación (ID) perteneciente al tratamiento realizado por el paciente		
Nombre de relación y tipo	Atiende, Relación (Agregación).										
Objetivo	Es una relación en la cual se registran datos de una entidad inicial que un médico le realiza a un paciente.										
Metadatos:											
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key	Descripción del contenido			
Actitud	VARCHAR	10	NO	NO	NO			Actitud del paciente a la hora de atenderse con un médico. (Coma, Apatía, Normal y Excitación)			
Antecedentes_Personales	VARCHAR	55	NO	NO	SI			Antecedentes personales que el paciente haya sufrido.			
CI Medico	INTEGER		SI	NO	NO		Medico	CI Medico	Número de identidad del documento (CI) perteneciente al médico.		
CI Paciente	INTEGER		SI	NO	NO		Paciente	CI Paciente	Número de identidad del documento (CI) perteneciente al paciente.		
Dolor	BOOLEAN		NO	NO	NO			Síntoma del paciente que registra el sistema.			
Tipo_ErfermedadAnterior	VARCHAR	20	NO	NO	SI			Tipo de enfermedad que padeció anteriormente el paciente consultante.			
Tipo_ErfermedadActual	VARCHAR	20	NO	NO	SI			Tipo de enfermedad que padece el paciente consultante.			
Enfermedad_Actual	VARCHAR	55	NO	NO	SI			Enfermedad actual del paciente consultante.			
Enfermedad_Familiar	VARCHAR	55	NO	NO	SI			Enfermedad familiar del paciente consultante.			
Estado_De_Conciencia	VARCHAR	20	NO	NO	NO			Estado de conciencia del paciente consultante.			
Estado_De_Nutricion	CHECK		NO	NO	NO			Estado de nutrición del paciente consultante. (Obesidad, Común, Anorexia, Caquezia)			
Fecha_Atiende	DATETIME		SI	NO	NO			Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.			
Fiebre	BOOLEAN		NO	NO	NO			Síntoma del paciente que registra el sistema.			
Frecuencia_Cardiaca	VARCHAR	20	NO	NO	NO			Frecuencia cardíaca del paciente a la hora de consultar con el médico.			
Frecuencia_Respiratoria	VARCHAR	20	NO	NO	NO			Frecuencia respiratoria del paciente a la hora de consultar con el médico.			
Grado_De_Colaboracion	VARCHAR	20	NO	NO	NO			Grado de colaboración del paciente a la hora de consultar con el médico.			
Grado_De_Hidratacion	VARCHAR	20	NO	NO	NO			Grado de hidratación del paciente a la hora de consultar con el médico.			
Malestar	BOOLEAN		NO	NO	NO			Síntoma del paciente que registra el sistema.			
Motivo	VARCHAR	240	NO	NO	NO			Motivo por el cual el paciente se atiende con el médico.			
Movilidad	VARCHAR	20	NO	NO	NO			Motivo por el cual el paciente se atiende con el médico.			
Nombre_Acompañante	VARCHAR	20	NO	NO	SI			Nombre del acompañante del paciente a la hora de consultar con el médico.			
Otro	VARCHAR	20	NO	NO	SI			Otro síntoma que tenga el paciente a la hora de consultar con el médico.			
Pulso	CHECK		NO	NO	NO			Pulso del paciente a la hora de consultar con el médico. (Fuerte, Normal, Débil, Filiforme)			
Temperatura	DOUBLE		NO	NO	NO			Temperatura del paciente a la hora de consultar con el médico.			

Nombre de entidad y tipo	<i>Persona, Entidad.</i>													
Objetivo	<i>Es una entidad la cual registra los datos de las personas tanto de paciente como de funcionarios médicos de la institución.</i>													
Metadatos:														
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key							
Calle	VARCHAR	25	NO	NO	NO		hacia tabla	hacia atributo						
CI	INTEGER		SI	NO	NO									
Fecha_Nac	DATE		NO	NO	NO									
Esquina	VARCHAR	25	NO	NO	NO									
Número	VARCHAR	10	NO	NO	NO									
Primer_Apellido	VARCHAR	20	NO	NO	NO									
	VARCHAR	25	NO	NO	NO									
Nombre de entidad y tipo	<i>Personal_Medico, Entidad.</i>													
Objetivo	<i>Es una entidad que hereda de Persona y representa a todo el personal médico de la institución (Médicos y Auxiliares Médicos).</i>													
Metadatos:														
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key							
							hacia tabla	hacia atributo						
CI	INTEGER		SI	NO	NO		Persona	CI						
EstadoBorradoM	BOOLEAN		NO	NO	NO									
Nombre de entidad y tipo	<i>Paciente, Entidad.</i>													
Objetivo	<i>Es una entidad que hereda de Persona y registra datos propios de los pacientes como el Estado Civil o el Sexo.</i>													
Metadatos:														
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key							
							hacia tabla	hacia atributo						
CI_Paciente	INTEGER		SI	NO	NO		Persona	CI						
Estado_Civil	CHECK		NO	NO	NO									
Sexo	CHECK		NO	NO	NO									
EstadoBorradoP	BOOLEAN		NO	NO	NO									
Nombre de entidad y tipo	<i>Persona_Teléfono, Entidad.</i>													
Objetivo	<i>Es una entidad generada por el atributo multivalueado *Teléfono* de Persona la cual registrá todas los teléfonos de las personas junto con la CI correspondiente.</i>													
Metadatos:														
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key							
							hacia tabla	hacia atributo						
CI	INTEGER		SI	NO	NO		Persona	CI						
Teléfono	VARCHAR	9	SI	NO	NO									
Nombre de entidad y tipo	<i>Formulario, Entidad.</i>													
Objetivo	<i>Es una entidad en la cual se registran datos de un formulario como su ID y el Tipo de Formulario que es.</i>													
Metadatos:														
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key							
							hacia tabla	hacia atributo						
ID_Form	INTEGER		SI	NO	NO									
Tipo_Form	VARCHAR	25	NO	NO	NO									
Nombre de relación y tipo	<i>Tiene, Relación.</i>													
Objetivo	<i>Es una relación la cual enlaza las preguntas que contiene un formulario con sus respuestas.</i>													
Metadatos:														
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key							
							hacia tabla	hacia atributo						
ID_Form	INTEGER		SI	NO	NO		Formulario	ID_Form						
ID_Preg	INTEGER		SI	NO	NO		Pregunta	ID_Preg						
FechaR	DATETIME		SI	NO	NO									
ID_Resp	INTEGER		SI	NO	NO		Respuesta	ID_Resp						
Nombre de entidad y tipo	<i>Medico, Entidad.</i>													
Objetivo	<i>Es una entidad que hereda de Personal Médico y registra datos como la Especialidad perteneciente a un médico junto con su CI.</i>													
Metadatos:														
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key							
							hacia tabla	hacia atributo						
CI_Medico	INTEGER		SI	NO	NO		Personal_Medico	CI						
Especialidad	VARCHAR	25	NO	NO	NO									

Nombre de entidad y tipo	Diagnóstico, Entidad.											
Objetivo	Es una entidad la cual registra todos los datos correspondientes a los diagnósticos elaborados por un médico sobre varios pacientes.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
Descripción_Diag	VARCHAR	240	NO	NO	NO		hacia tabla	hacia atributo	Descripción perteneciente al diagnóstico realizado por un médico sobre un paciente.			
ID_Diag	INTEGER		SI	NO	NO				Número de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.			
Tipo_Diag	VARCHAR	50	NO	NO	NO				Tipo de diagnóstico que elabora un médico sobre un paciente.			
EstadoBorradoD	BOOLEAN		NO	NO	NO				Estado que determina si un diagnóstico está habilitado o no (Baja lógica).			
Nombre de entidad y tipo	Tratamiento, Entidad.											
Objetivo	Es una entidad la cual registra datos correspondientes de un tratamiento que se le realiza a un paciente.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
Descripción_Trat	VARCHAR	240	NO	NO	NO		hacia tabla	hacia atributo	Descripción perteneciente al tratamiento realizado por un paciente.			
ID_Trat	INTEGER		SI	NO	NO				Número de identificación (ID) perteneciente al tratamiento realizado por el paciente.			
Nombre	VARCHAR	50	NO	NO	NO				Nombre perteneciente al diagnóstico elaborado por el médico sobre un paciente.			
EstadoBorradoT	BOOLEAN		NO	NO	NO				Estado que determina si un tratamiento está habilitado o no (Baja lógica).			
Nombre de entidad y tipo	Pregunta, Entidad.											
Objetivo	Es una entidad la cual registra datos como la ID de una Pregunta y su texto correspondiente.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Preg	INTEGER		SI	NO	NO		hacia tabla	hacia atributo	Número de identificación (ID) perteneciente a la pregunta perteneciente a un formulario.			
TextoPreg	VARCHAR	160	NO	NO	NO				Texto de la pregunta de un formulario.			
Nombre de relación y tipo	Sugiere, Relación.											
Objetivo	Es la relación la cual enlaza los formularios que contienen preguntas y se sugieren a partir de la entrevista inicial realizada entre un paciente y un médico.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
CI_Medico	INTEGER		SI	NO	NO		Atiende	CI_Medico	Número de identidad del documento (CI) perteneciente al médico.			
CI_Paciente	INTEGER		SI	NO	NO		Atiende	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.			
Fecha_Atiende	DATETIME		SI	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.			
ID_Form	INTEGER		SI	NO	NO		Contiene	ID_Form	Número de identificación (ID) perteneciente al formulario elaborado por el médico sobre un paciente.			
ID_Preg	INTEGER		SI	NO	NO		Contiene	ID_Preg	Número de identificación (ID) perteneciente a la pregunta perteneciente a un formulario.			
Nombre de relación y tipo	Responde, Relación.											
Objetivo	Es la relación la cual enlaza las respuestas dadas por un paciente de un formulario llenado por un médico.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
CI_Medico	INTEGER		SI	NO	NO		Atiende	CI_Medico	Número de identidad del documento (CI) perteneciente al médico.			
CI_Paciente	INTEGER		SI	NO	NO		Atiende	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.			
Fecha_Atiende	DATETIME		SI	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.			
ID_Resp	INTFGFR		SI	NO	NO		Responde	ID_Resp	Número de identificación (ID) perteneciente a la respuesta de la pregunta del formulario.			
Nombre de relación y tipo	Requiere, Relación.											
Objetivo	Es una relación la cual enlaza el diagnóstico que requiere de un análisis clínico para culminarse.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Analisis	INTEGER		SI	NO	NO		Analisis_Clinico	ID_Analisis	Número de identificación (ID) perteneciente al análisis solicitado por el médico sobre un paciente.			
ID_Diag	INTEGER		SI	NO	NO		Diagnostico	ID_Diag	Número de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.			
Nombre de relación y tipo	Contiene, Relación (Agregación).											
Objetivo	Es la relación que representa las preguntas que contiene un formulario.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Form	INTEGER		SI	NO	NO		Formulario	ID_Form	Número de identificación (ID) perteneciente al formulario elaborado por el médico sobre un paciente.			
ID_Preg	INTEGER		SI	NO	NO		Pregunta	ID_Preg	Número de identificación (ID) perteneciente a la pregunta perteneciente a un formulario.			
Nombre de entidad y tipo	Estado, Entidad.											
Objetivo	Es la entidad que registra todos los posibles estados junto con su ID en los que se puede encontrar un paciente en la institución médica.											
Metadatos:												
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido			
ID_Estado	INTEGER		SI	NO	NO		hacia tabla	hacia atributo	Número de identificación (ID) perteneciente al estado existente de un paciente en la institución médica.			
Nombre_Estado	VARCHAR	40	NO	NO	NO				Nombre perteneciente al estado en el que se encuentra un paciente.			

Nombre de relación y tipo	<i>Elabora, Relación.</i>														
Objetivo	<i>Es la relación que registra el diagnóstico elaborado por un médico sobre un paciente.</i>														
Metadatos:															
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key								
ID_Diag	INTEGER		SI	NO	NO		Diagnóstico	ID_Diag	Número de identificación (ID) perteneciente al diagnóstico elaborado por el médico sobre un paciente.						
Fecha_Diag	DATETIME		SI	NO	NO			Fecha con formato (DD-MM-AAAA) en la que el médico realiza un diagnóstico sobre un paciente.							
Fecha_Atiende	DATETIME		NO	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.						
CI_Medico	INTEGER		NO	NO	NO		Atiende	CI_Medico	Número de identidad del documento (CI) perteneciente al médico.						
CI_Paciente	INTEGER		NO	NO	NO		Atiende	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.						
Nombre de relación y tipo	<i>Posee, Relación.</i>														
Objetivo	<i>Es la relación que registra los datos del estado en el que se encuentra un paciente en la institución médica.</i>														
Metadatos:															
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key								
ID_Estado	INTEGER		SI	NO	NO		Estado	ID_Estado	Número de identificación (ID) perteneciente al estado existente de un paciente en la institución médica.						
CI_Paciente	INTEGER		SI	NO	NO		Paciente	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.						
FechaInicio_E	DATETIME		SI	NO	NO			Fecha con formato (DD-MM-AAAA) en la que el paciente entra en un estado.							
Nombre de entidad y tipo	<i>vista_NombreMedico, Entidad - Vista.</i>														
Objetivo	<i>Es la una entidad de tipo vista generada a partir de la cédula, primer nombre y primer apellido de todos los médicos</i>														
Metadatos:															
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key								
CI	INTEGER		SI	NO	NO		Estado	ID_Estado	Número de identidad del documento (CI) perteneciente al médico.						
Primer_Nombre	INTEGER		SI	NO	NO		Paciente	CI_Paciente	Primer nombre perteneciente al médico.						
Primer_Apellido	DATETIME		SI	NO	NO			Primer apellido perteneciente al médico.							
Nombre de relación y tipo	<i>Incluye, Relación.</i>														
Objetivo	<i>Es la relación que representa las partes del cuerpo pertenecientes al paciente.</i>														
Metadatos:															
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key								
ID_Partes	INTEGER		SI	NO	NO		PartesCuerpo	ID_Partes	Número de identificación (ID) perteneciente a las partes del cuerpo donde un paciente presenta dolor.						
CI_Paciente	INTEGER		SI	NO	NO		Atiende	CI_Paciente	Número de identidad del documento (CI) perteneciente al paciente.						
Fecha_Atiende	DATETIME		SI	NO	NO		Atiende	Fecha_Atiende	Fecha con formato (DD-MM-AAAA) en la que el paciente realiza una consulta con el médico.						
CI_Medico	INTEGER		SI	NO	NO		Atiende	CI_Medico	Número de identidad del documento (CI) perteneciente al médico.						
Nombre de entidad y tipo	<i>PartesCuerpo, Entidad.</i>														
Objetivo	<i>Es la entidad que representa las partes del cuerpo pertenecientes al paciente.</i>														
Metadatos:															
Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key								
ID_Partes	INTEGER		SI	NO	NO			Número de identificación (ID) perteneciente a las partes del cuerpo donde un paciente presenta dolor.							
Cabeza	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de la cabeza del paciente.							
Cuello	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona del cuello del paciente.							
Brazo	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de los brazos del paciente.							
Mano	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de las manos del paciente.							
Muslo	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de los muslos del paciente.							
Pantorrilla	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de las pantorrillas del paciente.							
Oreja	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de las orejas del paciente.							
Hombro	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de los hombros del paciente.							
Espalda	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de la espalda del paciente.							
Trasero	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona del trasero del paciente.							
Tobillo	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de los tobillos del paciente.							
Ojo	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de los ojos del paciente.							
Nariz	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de la nariz del paciente.							
Boca	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de la boca del paciente.							
Muñeca	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de las muñecas del paciente.							
Dedo	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de los dedos del paciente.							
Pierna	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de las piernas del paciente.							
Frente	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de la frente del paciente.							
Garganta	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de la garganta del paciente.							
Pecho	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona del pecho del paciente.							
Estomago	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona del estómago del paciente.							
Rodilla	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de las rodillas del paciente.							
Pie	BOOLEAN		NO	NO	NO			Atributo que hace referencia a la zona de los pies del paciente.							

Nombre de entidad y tipo		vista SintomasAtiendeCD, Entidad - Vista.									
Objetivo		Es la una entidad de tipo vista generada a partir de la cédula, primer nombre y primer apellido del paciente en conjunto con su enfermedad actual y síntomas que padece, acompañado de la zona en la que tiene dolor.									
Metadatos:		Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido
									hacia tabla	hacia atributo	
CI	INTEGER	SI	NO	NO					Estado	ID Estado	Número de identidad del documento (CI) perteneciente al paciente.
Primer Nombre	INTEGER	SI	NO	NO					Paciente	CI Paciente	Primer nombre perteneciente al paciente.
Primer Apellido	DATETIME	SI	NO	NO							Primer apellido perteneciente al paciente.
Enfermedad_Actual	VARCHAR	55	NO	NO	SI						Enfermedad actual del paciente consultante.
Dolor	BOOLEAN		NO	NO	NO						Síntoma del paciente que registra el sistema.
Malestar	BOOLEAN		NO	NO	NO						Síntoma del paciente que registra el sistema.
Fiebre	BOOLEAN		NO	NO	NO						Síntoma del paciente que registra el sistema.
Otro	VARCHAR	20	NO	NO	SI						Otro síntoma que tenga el paciente a la hora de consultar con el médico.
Cabeza	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de la cabeza del paciente.
Cuello	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona del cuello del paciente.
Brazo	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de los brazos del paciente.
Mano	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de las manos del paciente.
Muslo	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de los muslos del paciente.
Pantorrilla	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de las pantorrillas del paciente.
Oreja	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de las orejas del paciente.
Hombro	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de los hombros del paciente.
Espalda	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de la espalda del paciente.
Trasero	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona del trasero del paciente.
Tobillo	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de los tobillos del paciente.
Ojo	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de los ojos del paciente.
Nariz	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de la nariz del paciente.
Boca	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de la boca del paciente.
Muñeca	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de las muñecas del paciente.
Dedo	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de los dedos del paciente.
Pierna	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de las piernas del paciente.
Frente	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de la frente del paciente.
Garganta	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de la garganta del paciente.
Pecho	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona del pecho del paciente.
Estomago	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona del estómago del paciente.
Rodilla	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de las rodillas del paciente.
Pie	BOOLEAN		NO	NO	NO						Atributo que hace referencia a la zona de los pies del paciente.
Nombre de entidad y tipo		vista SintomasAtiendeSD, Entidad - Vista.									
Objetivo		Es la una entidad de tipo vista generada a partir de la cédula, primer nombre y primer apellido del paciente en conjunto con su enfermedad actual y síntomas que padece.									
Metadatos:		Nombre atributo	Tipo dato	Largo (size)	Clave Primaria	Clave Alternativa	Nulo	Valor default	Foreign Key		Descripción del contenido
									hacia tabla	hacia atributo	
CI	INTEGER	SI	NO	NO							Número de identidad del documento (CI) perteneciente al paciente.
Primer Nombre	INTEGER	SI	NO	NO							Primer nombre perteneciente al paciente.
Primer Apellido	DATETIME	SI	NO	NO							Primer apellido perteneciente al paciente.
Enfermedad_Actual	VARCHAR	55	NO	NO	SI						Enfermedad actual del paciente consultante.
Dolor	BOOLEAN		NO	NO	NO						Síntoma del paciente que registra el sistema.
Malestar	BOOLEAN		NO	NO	NO						Síntoma del paciente que registra el sistema.
Fiebre	BOOLEAN		NO	NO	NO						Síntoma del paciente que registra el sistema.
Otro	VARCHAR	20	NO	NO	SI						Otro síntoma que tenga el paciente a la hora de consultar con el médico.

Código de la Base de Datos:

/*Creación BD*/

CREATE DATABASE ProyectoGSA;

USE ProyectoGSA;

/*Entidades y relaciones*/

CREATE TABLE Persona(CI integer, Fecha_Nac date NOT NULL, Calle varchar(25) NOT NULL, Esquina varchar(25) NOT NULL, Numero varchar(10) NOT NULL, Primer_Apellido varchar(20) NOT NULL, Segundo_Apellido varchar(20) NOT NULL, Primer_Nombre varchar(20) NOT NULL, Segundo_Nombre varchar(20), Correo varchar(40),

PRIMARY KEY (CI));

CREATE TABLE Personal_Medico(CI integer, EstadoBorradoM boolean NOT NULL DEFAULT FALSE,

PRIMARY KEY (CI),

FOREIGN KEY (CI) REFERENCES Persona(CI));

CREATE TABLE Paciente(CI_Paciente integer, Estado_Civil VARCHAR(12) NOT NULL, Sexo CHAR(1) NOT NULL, EstadoBorradoP boolean NOT NULL DEFAULT FALSE,

PRIMARY KEY (CI_Paciente),

FOREIGN KEY (CI_Paciente) REFERENCES Persona(CI),

CONSTRAINT CHK_Paciente CHECK (Estado_Civil IN ('Soltero/a','Viudo/a','Casado/a','Divorciado/a') AND Sexo IN ('F','M'));

CREATE TABLE Medico (CI_Medico integer, Especialidad varchar(25) NOT NULL,

PRIMARY KEY(CI_Medico),

FOREIGN KEY (CI_Medico) REFERENCES Personal_Medico(CI));

CREATE TABLE Persona_Teléfono(CI integer, Teléfono varchar(9),

PRIMARY KEY(CI, Teléfono),



FOREIGN KEY (CI) REFERENCES Persona(CI),
UNIQUE(CI, Telefono));

CREATE TABLE Respuesta(ID_Resp integer AUTO_INCREMENT, TextoResp
varchar(160) NOT NULL,
PRIMARY KEY(ID_Resp));

CREATE TABLE Pregunta (ID_Preg integer AUTO_INCREMENT, TextoPreg
varchar(160) NOT NULL,
PRIMARY KEY(ID_Preg));

CREATE TABLE Atiende(Fecha_Atiende datetime, CI_Medico integer,
CI_Paciente integer, Estado_De_Conciencia varchar(20) NOT NULL, Movilidad
varchar(20) NOT NULL, Otro varchar(20), Fiebre boolean, Dolor boolean,
Malestar boolean, Pulso VARCHAR(10) NOT NULL , Frecuencia_Cardiaca
varchar(20), Temperatura double NOT NULL, Grado_De_Hidratacion
varchar(20) NOT NULL, Estado_De_Nutricion VARCHAR(8) NOT NULL,
Actitud VARCHAR(10) NOT NULL, Grado_De_Colaboracion varchar(20) NOT
NULL, Frecuencia_Respiratoria varchar(20) NOT NULL, Motivo varchar(240)
NOT NULL, Enfermedad_Actual varchar(55), Nombre_Acompaniante
varchar(20), Tipo_EnfermedadActual varchar(20), Tipo_EnfermedadAnterior
varchar(20), Enfermedad_Familiar varchar(55), Antecedentes_Personales
varchar(55),

PRIMARY KEY(Fecha_Atiende, CI_Medico, CI_Paciente),
FOREIGN KEY(CI_Medico) REFERENCES Medico(CI_Medico),
FOREIGN KEY(CI_Paciente) REFERENCES Paciente(CI_Paciente),
CONSTRAINT CHK_Atiende CHECK (Pulso IN
('Fuerte','Débil','Normal','Filiforme') AND Actitud IN
('Coma','Apatía','Normal','Excitación') AND Estado_De_Nutricion IN
('Obesidad','Común','Anorexia','Cauquexia'));

CREATE TABLE Analisis_Clinico(ID_Analisis integer AUTO_INCREMENT,
TipoAnalisis varchar(50) NOT NULL, Resultado varchar(240) NOT NULL,
Fecha_Analisis DATETIME NOT NULL, Fecha_Atiende DATETIME NOT
NULL, CI_Medico integer NOT NULL, CI_Paciente integer NOT NULL,
EstadoBorradoA boolean NOT NULL DEFAULT FALSE,



PRIMARY KEY(ID_Analisis),
FOREIGN KEY (Fecha_Atiende) REFERENCES Atiende(Fecha_Atiende),
FOREIGN KEY (CI_Paciente) REFERENCES Atiende(CI_Paciente),
FOREIGN KEY (CI_Medico) REFERENCES Atiende(CI_Medico));

CREATE TABLE Formulario(ID_Form integer AUTO_INCREMENT, Tipo_Form
varchar(50) NOT NULL, EstadoBorradoF boolean NOT NULL DEFAULT
FALSE,
PRIMARY KEY(ID_Form));

CREATE TABLE Diagnostico(ID_Diag integer AUTO_INCREMENT, Tipo_Diag
varchar(50) NOT NULL, Descripcion_Diag varchar(240) NOT NULL,
EstadoBorradoD boolean NOT NULL DEFAULT FALSE,
PRIMARY KEY(ID_Diag));

CREATE TABLE Tratamiento(ID_Trata integer AUTO_INCREMENT, Nombre
varchar(50) NOT NULL, Descripcion_Trat varchar(240) NOT NULL,
EstadoBorradoT boolean NOT NULL DEFAULT FALSE,
PRIMARY KEY(ID_Trata));

CREATE TABLE Estado(ID_Estado integer AUTO_INCREMENT,
Nombre_Estado varchar(40) NOT NULL,
PRIMARY KEY(ID_Estado));

CREATE TABLE Realiza(CI_Paciente integer, ID_Trata integer, Fechalinicio
datetime NOT NULL, FechaFinal datetime,
PRIMARY KEY(CI_Paciente, ID_Trata, Fechalinicio),
FOREIGN KEY (CI_Paciente) REFERENCES Paciente(CI_Paciente),
FOREIGN KEY (ID_Trata) REFERENCES Tratamiento(ID_Trata));

CREATE TABLE Demanda(ID_Trata integer, ID_Diag integer,
PRIMARY KEY(ID_Trata, ID_Diag),



FOREIGN KEY (ID_Trata) REFERENCES Tratamiento(ID_Trata),
FOREIGN KEY (ID_Diag) REFERENCES Diagnostico(ID_Diag));

CREATE TABLE Contiene (ID_Form integer, ID_Preg integer,
PRIMARY KEY(ID_Form, ID_Preg),
FOREIGN KEY (ID_Form) REFERENCES Formulario(ID_Form),
FOREIGN KEY (ID_Preg) REFERENCES Pregunta(ID_Preg));

CREATE TABLE Sugiere(Fecha_Atiende datetime, CI_Medico integer,
CI_Paciente integer, ID_Form integer, ID_Preg integer,
PRIMARY KEY(Fecha_Atiende, CI_Medico, CI_Paciente, ID_Form, ID_Preg),
FOREIGN KEY (Fecha_Atiende) REFERENCES Atiende(Fecha_Atiende),
FOREIGN KEY (CI_Medico) REFERENCES Atiende(CI_Medico),
FOREIGN KEY (CI_Paciente) REFERENCES Atiende(CI_Paciente),
FOREIGN KEY (ID_Preg) REFERENCES Contiene(ID_Preg),
FOREIGN KEY (ID_Form) REFERENCES Contiene(ID_Form));

CREATE TABLE Responde(CI_Medico integer, CI_Paciente integer,
Fecha_Atiende datetime, ID_Resp integer,
PRIMARY KEY(CI_Medico, CI_Paciente, Fecha_Atiende, ID_Resp),
FOREIGN KEY (CI_Medico) REFERENCES Atiende(CI_Medico),
FOREIGN KEY (CI_Paciente) REFERENCES Atiende(CI_Paciente),
FOREIGN KEY (Fecha_Atiende) REFERENCES Atiende(Fecha_Atiende),
FOREIGN KEY (ID_Resp) REFERENCES Respuesta(ID_Resp));

CREATE TABLE Requiere(ID_Analisis integer, ID_Diag integer,
PRIMARY KEY(ID_Analisis, ID_Diag),
FOREIGN KEY (ID_Analisis) REFERENCES Analisis_Clinico (ID_Analisis),
FOREIGN KEY (ID_Diag) REFERENCES Diagnostico(ID_Diag));



```
CREATE TABLE Tiene(ID_Form integer, ID_Preg integer, ID_Resp integer,  
FechaR datetime,  
PRIMARY KEY(ID_Form, ID_Preg, ID_Resp, FechaR),  
FOREIGN KEY(ID_Resp) REFERENCES Respuesta(ID_Resp),  
CONSTRAINT `tiene_ibfk_11`  
    FOREIGN KEY (`ID_Form`)  
    REFERENCES `proyectogsa`.`contiene` (`ID_Form`)  
    ON DELETE CASCADE,  
CONSTRAINT `tiene_ibfk_22`  
    FOREIGN KEY (`ID_Preg`)  
    REFERENCES `proyectogsa`.`contiene` (`ID_Preg`)  
    ON DELETE CASCADE);
```

```
CREATE TABLE Elabora(ID_Diag integer, Fecha_Diag datetime,  
Fecha_Atiende datetime, CI_Medico integer, CI_Paciente integer,  
PRIMARY KEY(ID_Diag, Fecha_Diag),  
FOREIGN KEY(ID_Diag) REFERENCES Diagnostico(ID_Diag),  
FOREIGN KEY(Fecha_Atiende) REFERENCES Atiende(Fecha_Atiende),  
FOREIGN KEY(CI_Medico) REFERENCES Atiende(CI_Medico),  
FOREIGN KEY(CI_Paciente) REFERENCES Atiende(CI_Paciente));
```

```
CREATE TABLE Posee(ID_Estado integer, CI_Paciente integer, Fechalinicio_E  
datetime,  
PRIMARY KEY(ID_Estado, CI_Paciente, Fechalinicio_E),  
FOREIGN KEY(ID_Estado) REFERENCES Estado(ID_Estado),  
FOREIGN KEY(CI_Paciente) REFERENCES Paciente(CI_Paciente));
```

```
CREATE TABLE PartesCuerpo (ID_Partes integer auto_increment, Cabeza  
boolean, Cuello boolean, Brazo boolean, Mano boolean, Muslo boolean,  
Pantorrilla boolean, Oreja boolean, Hombro boolean, Espalda boolean, Trasero  
boolean, Tobillo boolean, Ojo boolean, Nariz boolean, Boca boolean, Munieca
```



boolean, Dedo boolean, Pierna boolean, Frente boolean, Garganta boolean,
Pecho boolean, Estomago boolean, Rodilla boolean, Pie boolean,
PRIMARY KEY (ID_Partes);

```
CREATE TABLE Incluye(ID_Partes integer, CI_Paciente integer,  
Fecha_Atiende datetime, CI_Medico integer,  
PRIMARY KEY(ID_Partes, CI_Paciente, CI_Medico, Fecha_Atiende),  
FOREIGN KEY(ID_Partes) REFERENCES PartesCuerpo(ID_Partes),  
FOREIGN KEY(Fecha_Atiende) REFERENCES Atiende(Fecha_Atiende),  
FOREIGN KEY(CI_Medico) REFERENCES Atiende(CI_Medico),  
FOREIGN KEY(CI_Paciente) REFERENCES Atiende(CI_Paciente));
```

```
CREATE TABLE Rol(ID_Rol smallint AUTO_INCREMENT, Nombre_Rol  
varchar(20) NOT NULL,  
PRIMARY KEY(ID_Rol));
```

```
CREATE TABLE Usuario_Rol(Usuario varchar(20), Rol smallint NOT NULL, CI  
integer NOT NULL,  
PRIMARY KEY(Usuario),  
FOREIGN KEY(Rol) REFERENCES Rol(ID_Rol));
```

```
CREATE VIEW  
vista_SintomasAtiendeCD AS  
SELECT p.CI, p.Primer_Nombre, p.Primer_Apellido, a.Enfermedad_Actual,  
a.Dolor, a.Fiebre, a.Malestar, a.Otro, pc.Cabeza, pc.Cuello, pc.Brazo, pc.Mano,  
pc.Muslo, pc.Pantorrilla, pc.Oreja, pc.Hombro, pc.Espalda, pc.Trasero,  
pc.Tobillo, pc.Ojo, pc.Nariz, pc.Boca, pc.Munieca, pc.Dedo, pc.Pierna,  
pc.Frente, pc.Garganta, pc.Pecho, pc.Estomago, pc.Rodilla, pc.Pie  
FROM Persona as p  
JOIN Paciente as pa ON p.CI=pa.CI_Paciente  
JOIN Atiende as a ON pa.CI_Paciente=a.CI_Paciente  
JOIN Incluye as i ON a.Fecha_Atiende=i.Fecha_Atiende
```



```
JOIN PartesCuerpo as pc ON i.ID_Partes=pc.ID_Partes;
CREATE VIEW
vista_SintomasAtiendeSD AS
SELECT p.Cl, p.Primer_Nombre, p.Primer_Apellido, a.Enfermedad_Actual,
a.Dolor, a.Fiebre, a.Malestar, a.Otro
FROM Persona as p
JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente
JOIN Atiende as a ON pa.Cl_Paciente=a.Cl_Paciente;
CREATE VIEW
vista_NombreMedico AS
SELECT p.Cl, p.Primer_Nombre, p.Primer_Apellido
FROM Persona as p
JOIN Personal_Medico as pm ON p.Cl=pm.Cl
JOIN Medico as m ON pm.Cl=m.Cl_Medico;

/*PRIVILEGIOS DE USUARIOS*/
INSERT INTO `Rol` (`Nombre_Rol`) VALUES
('Medico'),
('Auxiliar'),
('Paciente'),
('Administrador');
INSERT INTO `Usuario_Rol` (`Usuario`, `Rol`, `Cl`) VALUES
('GSA_Med',1,54200707),
('GSA_Aux',2,55762293),
('GSA_Pac',3,52137108),
('GSA_Adm',4,52582991);

/*Datos principales*/
INSERT INTO `Estado` (`Nombre_Estado`) VALUES
```

('Recolección de información'),
('Hipótesis diagnóstica'),
('Proceso de verificación y afirmación'),
('Transformación');

```
INSERT INTO `Diagnostico` (`Tipo_Diag`, `Descripcion_Diag`) VALUES  
('Clínico','A'),  
('Certeza','B'),  
('Diferencia','C'),  
('Etiológico','D'),  
('Genérico','E'),  
('Heroico','F'),  
('Lesional','G'),  
('Anatómico','H'),  
('Topográfico','I'),  
('Individual','J'),  
('Nosológico','K'),  
('Patogenético','A'),  
('Presuntivo','A'),  
('Sindómico y funcional','A'),  
('Sintomático','A');
```

```
INSERT INTO `Persona`  
(`CI`, `Fecha_Nac`, `Calle`, `Esquina`, `Numero`, `Primer_Apellido`, `Segundo_Ape  
llido`, `Primer_Nombre`, `Segundo_Nombre`, `Correo`) VALUES  
(55762293,'2002-10-  
22','a','b','1111','Iriart','Guibernau','Santiago','Miguel','santiagoiriart9@gmail.com'  
,),  
(54200707,'2002-12-  
2','c','d','1112','Gularte','Artola','Gonzalo','Lisandro','gonzalogulartemvd@gmail.c  
om'),
```



(52137108,'2002-4-
18','e','f','1113','Acosta','Bernardo','Agustin','Luciano','agustin14acosta@gmail.com'),
(52582991,'2002-10-
14','g','h','1114','Sunes','Pianavilla','Bruno','Daniel','brunosunes@gmai.com'),
(62653134,'2000-11-
11','i','j','1115','Rosales','Matos','Zendhel','','mruci2020@gmail.com'),
(11111111,'2001-12-
11','i','j','1116','Rodriguez','Cabrera','Kevin','','krodriguez2020@gmail.com'),
(11111112,'1990-9-
10','i','j','1117','Olivera','Trinidad','Nahuel','','nahuoliveraa@gmail.com'),
(11111113,'1998-12-28','i','j','1118','Plada','Rodriguez','Patricia','Betiana','patriciaplada@gmail.com'),
(11111114,'2018-11-
10','i','j','1119','Maldonado','Correa','Francesca','','francorrea25@gmail.com'),
(11111115,'1991-8-
4','i','j','1120','Perez','Disalvo','Martin','','martinpd@hotmail.com'),
(11111116,'1990-3-
11','i','j','1120','Santos','Pedrosa','Fabian','','fabiansantospe@hotmail.com'),
(11111117,'1980-4-
12','i','j','1120','Olivera','Fernandez','Silvia','','silviolivera18@gmail.com'),
(11111118,'1970-5-
13','i','j','1120','Berta','García','Agustina','','agustinagarcia@hotmail.com'),
(11111119,'1967-6-
14','i','j','1120','Suarez','Villasante','Carolina','','carosuarez8@gmail.com'),
(11111120,'2012-7-
15','i','j','1120','Cordoba','Traverso','Valentin','','ValentinTraverso@gmail.com'),
(11111121,'1999-8-
16','i','j','1120','Nievas','Troncoso','Micaela','','MicaelaTroncoso@gmail.com'),
(11111122,'1978-9-
17','i','j','1120','Medina','Telechea','Sandra','','SandraTelechea@hotmail.com'),
(11111123,'1945-10-
18','i','j','1120','Mayol','Garay','Maria','','MariaGaray@gmail.com'),
(11111124,'2016-11-
19','i','j','1120','Silvera','Hernandez','Emma','','EmmaHernandez@hotmail.com'),



(11111125,'2013-12-20','i','j','1120','Rodriguez','Gonzalez','Maxiemiliano','Pedro','MaxiemilianoGonzalez@hotmail.com'),

(11111126,'1950-10-22','a','b','1129','Ferreira','Godoy','Alberto','Pedro','AlbertoGodoy@gmail.com');

INSERT INTO `Persona_Teléfono` (`CI`, `Teléfono`) VALUES

(55762293,'093982489'),

(54200707,'094742318'),

(52137108,'091027326'),

(52582991,'092111308'),

(62653134,'096242702'),

(11111111,'095762985'),

(11111112,'098876243'),

(11111113,'091876435'),

(11111114,'099339902'),

(11111115,'099199199'),

(11111116,'099122393'),

(11111117,'095567456'),

(11111118,'094444444'),

(11111119,'095554554'),

(11111120,'094223232'),

(11111121,'094556766'),

(11111122,'096666666'),

(11111123,'097777777'),

(11111124,'093345678'),

(11111125,'091023495');

INSERT INTO `Paciente` (`CI_Paciente`, `Estado_Civil`, `Sexo`) VALUES

(55762293,'Soltero/a','M'),

(54200707,'Soltero/a','M'),
(52137108,'Soltero/a','F'),
(52582991,'Soltero/a','F'),
(62653134,'Soltero/a','F'),
(11111111,'Soltero/a','F'),
(11111112,'Casado/a','F'),
(11111113,'Soltero/a','F'),
(11111114,'Soltero/a','F'),
(11111115,'Viudo/a','F'),
(11111116,'Soltero/a','F'),
(11111117,'Casado/a','F'),
(11111126,'Viudo/a','M');

INSERT INTO `Personal_Medico` (`CI`) VALUES

(11111118),
(11111119),
(11111120),
(11111121),
(11111122),
(11111123),
(11111124),
(11111125);

INSERT INTO `Medico` (`CI_Medico`,`Especialidad`) VALUES

(11111118,'Traumatología'),
(11111119,'Dermatología'),
(11111120,'Ginecología'),
(11111121,'Oftalmología'),
(11111122,'Otorrinolaringología'),

(11111124,'Urología'),
(11111125,'Neumología');

INSERT INTO `Atiende`
(`Fecha_Atiende`, `CI_Paciente`, `CI_Medico`, `Estado_De_Conciencia`, `Movilidad`, `Otro`, `Fiebre`, `Dolor`, `Malestar`, `Pulso`, `Frecuencia_Cardiaca`, `Temperatura`, `Grado_De_Hidratacion`, `Estado_De_Nutricion`, `Actitud`, `Grado_De_Colaboracion`, `Frecuencia_Respiratoria`, `Motivo`, `Enfermedad_Actual`, `Nombre_Acompanante`, `Tipo_EnfermedadActual`, `Tipo_EnfermedadAnterior`, `Enfermedad_Familiar`, `Antecedentes_Personales`) VALUES
(
('2018-06-01
13:00:20',11111111,11111118,'Normal','Buena',",TRUE, FALSE, FALSE, 'Fuerte',
'125',37.9,'Normal','Caquexia','Normal','Normal','Baja','Fatiga','Anemia','Alberto','
Sanginea','Lecion','Artrosis', 'Desgarre muscular'),
(
('2014-04-14
15:00:40',11111126,11111125,'Anormal','Normal','Mareo',TRUE, FALSE, FALSE
, 'Filiforme', '80', 36.5, 'Sobrehidratado ', 'Anorexia', 'Apatía', 'Normal', 'Baja', 'Dolor
muscular', 'Desgarre muscular', 'Rodolfo', 'Lecion', ", ",),
(
('2018-06-01
13:00:20',11111116,11111125,'Normal','Normal',",TRUE, FALSE, FALSE, 'Débil',
'114',38.6,'Normal','Obesidad','Normal','Normal','Baja','visión
borrosa','Miopia','Ramón','Ocular', ", ",),
(
('2020-06-22
09:30:23',62653134,11111121,'Anormal','Buena','Mareo',TRUE, FALSE, TRUE,
'Normal','104',35.2,'Sobrehidratado','Común','Excitación','Normal','Baja','Dificultad al respirar','Asma','Ramón','Respiratorio','Ocular', ", 'Astigmatismo'),
(
('2020-08-18
16:00:23',11111114,11111125,'Normal','Baja',",TRUE, FALSE, FALSE, 'Fuerte',
'98',38.9,'Normal','Común','Normal','Normal','Baja','Dificultad al respirar','Covid-19','Ramón','Respiratorio','Respiratorio', ", 'Asma'),
(
('2015-11-02
16:00:34',62653134,11111125,'Normal','Normal','Desorientacion',FALSE, TRUE,
FALSE, 'Normal', '90', 39.2, 'Normal', 'Anorexia', 'Apatía', 'Normal', 'Baja', 'Dolor', 'Hipercoagibilidad', 'Agus', 'Sanginea', 'Vista', ", 'Miopia'),
(
('2019-01-01
18:15:12',11111126,11111118,'Anormal','Normal',",FALSE, TRUE, TRUE, 'Débil',
'108',36.8,'Normal','Anorexia','Normal','Normal','Baja','Malestar','Cancer','Rodolfo
, 'Tumor', 'Lecion', 'Diabetes', 'Fractura'),

('2020-07-28
13:50:23',11111117,11111120,'Normal','Baja',",FALSE,TRUE,FALSE,'Filiforme',
'100',36.9,'Deshidratado','Común','Excitación','Normal','Baja','Dolor de
cabeza','Aneurismas de los vasos del cerebro','José','Sanginea','Respiratorio', ",
'Gripe'),

('2010-08-06
19:30:43',11111115,11111125,'Anormal','Buena',",TRUE,TRUE,FALSE,'Normal
'112',39.8,'Normal','Anorexia','Normal','Normal','Baja','Dolor
Estomscal','Tumor','Lázaro','Tumor','Lecion', ", 'Ematoma'),

('2020-11-01
13:00:20',11111111,11111118,'Normal','Buena',",TRUE,FALSE,FALSE,'Fuerte',
'125',37.9,'Normal','Caquexia','Normal','Normal','Baja','Fatiga','Anemia','Alberto',
'Sanginea','Lecion','Artrosis','Desgarre muscular'),

('2020-11-14
15:00:40',11111126,11111125,'Anormal','Normal','Mareo',TRUE,FALSE,FALSE
, 'Filiforme','80',36.5,'Sobrehidratado ','Anorexia','Apatía','Normal','Baja','Dolor
muscular','Desgarre muscular','Rodolfo','Lecion', ", ", ")

('2020-11-01
13:00:20',11111116,11111125,'Normal','Normal',",TRUE,FALSE,FALSE,'Débil',
'114',38.6,'Normal','Obesidad','Normal','Normal','Baja','visión
borrosa','Miopia','Ramón','Ocular', ", ", ")

('2020-11-22
09:30:23',62653134,11111121,'Anormal','Buena','Mareo',TRUE,FALSE,TRUE,
'Normal','104',35.2,'Sobrehidratado','Común','Excitación','Normal','Baja','Dificulta
d al respirar','Asma','Ramón','Respiratorio','Ocular', ", 'Astigmatismo'),

('2020-10-01
13:00:20',11111111,11111118,'Normal','Buena',",TRUE,FALSE,FALSE,'Fuerte',
'125',37.9,'Normal','Caquexia','Normal','Normal','Baja','Fatiga','Anemia','Alberto',
'Sanginea','Lecion','Artrosis','Desgarre muscular'),

('2020-10-02
15:00:40',11111126,11111118,'Anormal','Normal','Mareo',TRUE,FALSE,FALSE
, 'Filiforme','80',36.5,'Sobrehidratado ','Anorexia','Apatía','Normal','Baja','Dolor
muscular','Desgarre muscular','Rodolfo','Lecion', ", ", ")

('2020-10-03
13:00:20',11111116,11111118,'Normal','Normal',",TRUE,FALSE,FALSE,'Débil',
'114',38.6,'Normal','Obesidad','Normal','Normal','Baja','Caída','Fractura en el
fémul','Ramón','Lecion', ", ", ")

('2020-10-14
09:30:23',62653134,11111118,'Anormal','Buena','Mareo',TRUE,FALSE,TRUE,
,

Normal','104',35.2,'Sobrehidratado','Común','Excitación','Normal','Baja','Dificultad al respirar','Asma','Ramón','Respiratorio','Lecion', ", "),

('2020-10-05

16:00:23',11111114,11111125,'Normal','Baja',"",TRUE, FALSE, FALSE, 'Fuerte', '98',38.9,'Normal','Común','Normal','Normal','Baja','Dificultad al respirar','Covid-19','Ramón','Respiratorio','Respiratorio', ", 'Asma'),

('2020-10-16

16:00:34',62653134,11111125,'Normal','Normal','Desorientacion',FALSE,TRUE, FALSE, 'Normal', '90',39.2,'Normal','Anorexia','Apatía','Normal','Baja','Dolor','Hipercoagibilidad','Agus','Sanginea','Vista', ", 'Miopia'),

('2020-10-17

18:15:12',11111126,11111120,'Anormal','Normal',"",FALSE, TRUE, TRUE, 'Débil', '108',36.8,'Normal','Anorexia','Normal','Normal','Baja','Malestar','Cancer','Rodolfo ','Tumor','Lecion', 'Diabetes', 'Fractura'),

('2020-10-28

13:50:23',11111117,11111120,'Normal','Baja',"",FALSE, TRUE, FALSE, 'Filiforme', '100',36.9,'Desidratado','Común','Excitación','Normal','Baja','Dolor de cabeza','Aneurismas de los vasos del cerebro','José','Sanginea','Respiratorio', ", 'Gripe'),

('2020-10-09

19:30:43',11111115,11111120,'Anormal','Buena',"",TRUE, TRUE, FALSE, 'Normal ', '112',39.8,'Normal','Anorexia','Normal','Normal','Baja','Dolor Estomscal','Tumor','Lázaro','Tumor','Lecion', ", 'Ematoma'),

('2020-06-10

13:00:20',11111111,11111118,'Normal','Buena',"",TRUE, FALSE, FALSE, 'Fuerte', '125',37.9,'Normal','Caquexia','Normal','Normal','Baja','Fatiga','Anemia','Alberto','Sanginea','Lecion', 'Artrosis', 'Desgarre muscular'),

('2020-04-11

15:00:40',11111126,11111125,'Anormal','Normal','Mareo',TRUE, FALSE, FALSE, 'Filiforme', '80',36.5,'Sobrehidratado ','Anorexia','Apatía','Normal','Baja','Dolor muscular','Desgarre muscular','Rodolfo','Lecion', ", ", "),

('2020-02-12

13:00:20',62653134,11111125,'Normal','Normal',"",TRUE, FALSE, FALSE, 'Débil', '114',38.6,'Normal','Obesidad','Normal','Normal','Baja','visión borrosa','Miopia','Ramón','Ocular', ", ", "),

('2020-06-13

09:30:23',62653134,11111121,'Anormal','Buena','Mareo',TRUE, FALSE, TRUE, 'Normal','104',35.2,'Sobrehidratado','Común','Excitación','Normal','Baja','Dificultad al respirar','Asma','Ramón','Respiratorio','Ocular', ", 'Astigmatismo'),

('2020-06-14
13:00:20',11111111,11111118,'Normal','Buena',"",TRUE, FALSE, FALSE, 'Fuerte',
'125',37.9,'Normal','Caquexia','Normal','Normal','Baja','Fatiga','Anemia','Alberto','
Sanginea','Lecion', 'Artrosis', 'Desgarre muscular'),

('2020-02-15
15:00:40',62653134,11111118,'Anormal','Normal','Mareo',TRUE, FALSE, FALSE
, 'Filiforme', '80', 36.5, 'Sobrehidratado ', 'Anorexia', 'Apatía', 'Normal', 'Baja', 'Dolor
muscular', 'Desgarre muscular', 'Rodolfo', 'Lecion', " ", " ", ")

('2020-02-16
13:00:20',62653134,11111125,'Normal','Normal',"",TRUE, FALSE, FALSE, 'Débil',
'114',38.6,'Normal','Obesidad','Normal','Normal','Baja','visión
borrosa','Miopia','Ramón','Ocular', " ", " ", ")

('2020-06-02
09:30:23',62653134,11111121,'Anormal','Buena','Mareo',TRUE, FALSE, TRUE,
'Normal','104',35.2,'Sobrehidratado','Común','Excitación','Normal','Baja','Dificulta
d al respirar','Asma','Ramón','Respiratorio','Ocular', " ", 'Astigmatismo'),

('2020-02-19
13:00:20',11111111,11111118,'Normal','Buena',"",TRUE, FALSE, FALSE, 'Fuerte',
'125',37.9,'Normal','Caquexia','Normal','Normal','Baja','Fatiga','Anemia','Alberto','
Sanginea','Lecion', 'Artrosis', 'Desgarre muscular'),

('2020-04-20
15:00:40',11111111,11111125,'Anormal','Normal','Mareo',TRUE, FALSE, FALSE
, 'Filiforme', '80', 36.5, 'Sobrehidratado ', 'Anorexia', 'Apatía', 'Normal', 'Baja', 'Dolor
muscular', 'Desgarre muscular', 'Rodolfo', 'Lecion', " ", " ", ")

('2020-02-21
13:00:20',11111116,11111125,'Normal','Normal',"",TRUE, FALSE, FALSE, 'Débil',
'114',38.6,'Normal','Obesidad','Normal','Normal','Baja','visión
borrosa','Miopia','Ramón','Ocular', " ", " ", ")

('2020-02-23
09:30:23',62653134,11111121,'Anormal','Buena','Mareo',TRUE, FALSE, TRUE,
'Normal','104',35.2,'Sobrehidratado','Común','Excitación','Normal','Baja','Dificulta
d al respirar','Asma','Ramón','Respiratorio','Ocular', " ", 'Astigmatismo'),

('2020-11-01
15:00:40',11111126,11111125,'Anormal','Normal','Mareo',TRUE, FALSE, FALSE
, 'Filiforme', '80', 36.5, 'Sobrehidratado ', 'Anorexia', 'Apatía', 'Normal', 'Baja', 'Dolor
muscular', 'Desgarre muscular', 'Rodolfo', 'Lecion', NULL, NULL, NULL),

('2020-11-01
09:30:23',62653134,11111121,'Anormal','Buena','Mareo',TRUE, FALSE, TRUE,
,

Normal','104',35.2,'Sobrehidratado','Común','Excitación','Normal','Baja','Dificulta
d al respirar','Asma','Ramón','Respiratorio','Ocular', NULL, 'Astigmatismo');

INSERT INTO `PartesCuerpo` (`Cabeza`, `Cuello`, `Brazo`, `Mano`, `Muslo`,
`Pantorrilla`, `Oreja`, `Hombro`, `Espalda`, `Trasero`, `Tobillo`, `Ojo`, `Nariz`,
`Boca`, `Munieca`, `Dedo`, `Pierna`, `Frente`, `Garganta`, `Pecho`, `Estomago`,
`Rodilla`, `Pie`) VALUES

(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
FALSE, TRUE, FALSE, FALSE, FALSE),

(FALSE, FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, FALSE, FALSE,
FALSE, FALSE, FALSE, FALSE, FALSE, TRUE, FALSE, FALSE, FALSE,
FALSE, FALSE, FALSE, FALSE, FALSE),

(TRUE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
FALSE, FALSE, FALSE, FALSE, FALSE),

(FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
FALSE, TRUE, FALSE, FALSE, FALSE);

INSERT INTO `Incluye` (`ID_Partes`, `CI_Paciente`, `Fecha_Atiende`,
`CI_Medico`) VALUES

(1, 62653134, '2015-11-02 16:00:34', 11111125),

(2, 11111126, '2019-01-01 18:15:12', 11111118),

(3, 11111117, '2020-07-28 13:50:23', 11111120),

(4 ,11111115, '2010-08-06 19:30:43', 11111125);

INSERT INTO `Analisis_Clinico`
(`TipoAnalisis`, `Resultado`, `Fecha_Analisis`, `Fecha_Atiende`, `CI_Medico`, `CI_
Paciente`) VALUES

('Química sanguíne','Anemia','2018-06-04 13:30:44','2018-06-01
13:00:20',11111118,11111111),

('Radiografia','Sin resultados','2014-04-14 14:30:22','2014-04-14
15:00:40',11111125,11111126),

('Ocular','Miopia','2018-06-04 13:30:44','2018-06-01
13:00:20',11111125,11111116),
('HIS','Covid-19 negativo','2020-06-22 10:15:34','2020-06-22
09:30:23',11111121, 62653134),
('HIS','Covid-19 positivo','2020-08-18 16:40:34','2020-08-18
16:00:23',11111125,11111114),
('Coagulación','hipercoagibilidad','2015-11-12 17:50:32','2015-11-02
16:00:34',11111125,62653134),
('Marcadores tumorales','Cancer','2019-01-30 09:30:43','2019-01-01
18:15:12',11111118,11111126),
('Resonancia magnética de la cabeza','Aneurismas de los vasos del
cerebro','2020-08-04 14:30:21','2020-07-28 13:50:23',11111120, 11111117),
('Resonancia magnética del tórax','Tumor','2010-08-23 20:00:00','2010-08-06
19:30:43',11111125, 11111115),
('Química sanguíne','Anemia','2020-11-04 13:00:20','2020-11-01
13:00:20',11111118,11111111),
('Radiografia','Sin resultados','2020-11-01 16:30:40','2020-11-01
15:00:40',11111125,11111126),
('Ocular','Miopia','2020-11-02 13:00:20','2020-11-01
13:00:20',11111125,11111116),
('HIS','Covid-19 negativo','2020-11-06 09:30:23','2020-11-01
09:30:23',11111121, 62653134);

INSERT INTO `Diagnostico` (`Tipo_Diag`, `Descripcion_Diag`) VALUES
(('Certeza','Anemia'),
(('Lecional','Desgarre muscular'),
(('Presuntivo','Miopia'),
(('Presuntivo','Asma'),
(('Nosológico','Covid-19'),
(('Presuntivo','Hipercoagibilidad'),
(('Anatomico','Cancer'),
(('Presuntivo','Aneurismas de los vasos del cerebro'),
(('Presuntivo','Tumor');

```
INSERT INTO `Tratamiento` (`Nombre`, `Descripcion_Trat`) VALUES
('Medicamento para anemia','Tomar suplementos de hierro.'),
('Tratamiento por desgarre muscular','0 horas: aplicación de frío, 48 horas: reposo, vendaje compresivo de la zona, 72 horas: dejar de aplicar frío, iniciar reabilitacion, 3 semanas finaliza el tratamiento.'),
('Tratamiento para la miopia','Uso de lentes'),
('Tratamiento para el asma','tomar Corticoesteroides inhalados'),
('Tratamiento para el Covid-19','Ingerir ibuprofeno, jarabe para la tos, reposo, ingerir líquidos'),
('Tratamiento para la hipercoagibilidad','Ingerir anticoagulantes durante un periodo de 3 a 6 meses.'),
('Tratamiento para cáncer','Cirujía en la que se extirpa el cáncer'),
('Desbiacion de flujo','Desviar el flujo sanguíneo para que no llegue al saco del aneurisma.'),
('Tratamiento para tumores','Se prosede a extirparse mediante cirugía el tumor');
```

```
INSERT INTO `Realiza` (`CI_Paciente`, `ID_Trata`, `Fechalnicio`, `FechaFinal`)
VALUES
(55762293,1,'2018-06-14 13:30:44','2019-06-15 12:00:30'),
(55762293,2,'2014-05-06 14:00:59','2014-05-27 14:1:50'),
(54200707,3,'2018-06-06 12:30:49','9998-12-31 00:00:00'),
(52137108,4,'2020-06-28 10:15:34','9998-12-31 00:00:00'),
(11111114,5,'2020-08-18 20:20:49','9998-12-31 00:00:00'),
(52137108,6,'2015-11-22 16:30:34','2016-11-24 17:15:39'),
(52582991,7,'2019-02-11 12:30:59','2020-01-25 13:50:54'),
(62653134,8,'2020-08-15 15:00:48','9998-12-31 00:00:00'),
(11111111,9,'2010-08-28 21:15:54','2011-11-08 19:30:58');
```

```
INSERT INTO `Elabora`
(`ID_Diag`, `Fecha_Diag`, `Fecha_Atiende`, `CI_Medico`, `CI_Paciente`)
VALUES
S.I.BI.M
I.T.I-C.E.T.P
3° BB
257
```



(1,'2018-06-12 13:20:49','2018-06-01 13:00:20',11111118, 11111111),
(2,'2014-04-12 14:51:32','2014-04-14 15:00:40',11111125, 11111126),
(3,'2018-06-04 13:30:44','2018-06-01 13:00:20',11111125, 11111116),
(4,'2020-06-26 10:59:21','2020-06-22 09:30:23',11111121, 62653134),
(5,'2020-08-18 18:40:34','2020-08-18 16:00:23',11111125, 11111114),
(6,'2015-11-20 18:10:30','2015-11-02 16:00:34',11111125, 62653134),
(7,'2019-02-06 10:40:47','2019-01-01 18:15:12',11111118, 11111126),
(8,'2020-08-10 16:13:13','2020-07-28 13:50:23',11111120, 11111117),
(9,'2010-08-26 19:24:53','2010-08-06 19:30:43',11111125, 11111115);

INSERT INTO `Demanda` (`ID_Trata`, `ID_Diag`) VALUES

(1,1),
(2,2),
(3,3),
(4,4),
(5,5),
(6,6),
(7,7),
(8,8),
(9,9);

INSERT INTO `Requiere` (`ID_Analisis`, `ID_Diag`) VALUES

(1,1),
(2,2),
(3,3),
(4,4),
(5,5),
(6,6),
(7,7),

(8,8),

(9,9);

INSERT INTO `Respuesta` (`TextoResp`) VALUES

('Me duele el pié'),

('Hace 5 dias'),

('Es un dolor agudo'),

('Siento que tengo mala digestión'),

('Desde ayer'),

('No'),

('Desde ayer'),

('No'),

('Si'),

('No');

INSERT INTO `Pregunta` (`TextoPreg`) VALUES

('¿Donde le duele?'),

('¿Hace cuanto le empezo el dolor?'),

('¿Qué tipo de dolor?'),

('¿Qué se siente?'),

('¿Hace cuando se siente ese malestar?'),

('¿Tiene toz o fiebre?'),

('¿Hace cuanto le empezó?'),

('¿Se hizo la prueba del covid?'),

('¿Estás resfriado?'),

('¿Se hizo alguna herida o cirugia recientemente?');

INSERT INTO `Formulario` (`Tipo_Form`) VALUES

('Dolor'),



('Dolor'),
('Malestar'),
('Fiebre');

```
INSERT INTO `Responde`  
(`CI_Medico`, `CI_Paciente`, `Fecha_Atiende`, `ID_Resp`) VALUES  
(11111118, 11111111, '2018-06-01 13:00:20', 1),  
(11111118, 11111111, '2018-06-01 13:00:20', 2),  
(11111118, 11111111, '2018-06-01 13:00:20', 3),  
(11111125, 11111114, '2020-08-18 16:00:23', 1),  
(11111125, 11111114, '2020-08-18 16:00:23', 2),  
(11111125, 11111114, '2020-08-18 16:00:23', 3),  
(11111125, 11111115, '2010-08-06 19:30:43', 4),  
(11111125, 11111115, '2010-08-06 19:30:43', 5),  
(11111125, 11111115, '2010-08-06 19:30:43', 6),  
(11111125, 11111116, '2018-06-01 13:00:20', 4),  
(11111125, 11111116, '2018-06-01 13:00:20', 5),  
(11111125, 11111116, '2018-06-01 13:00:20', 6),  
(11111120, 11111117, '2020-07-28 13:50:23', 4),  
(11111120, 11111117, '2020-07-28 13:50:23', 5),  
(11111120, 11111117, '2020-07-28 13:50:23', 6),  
(11111125, 11111126, '2014-04-14 15:00:40', 4),  
(11111125, 11111126, '2014-04-14 15:00:40', 5),  
(11111125, 11111126, '2014-04-14 15:00:40', 6),  
(11111118, 11111126, '2019-01-01 18:15:12', 7),  
(11111118, 11111126, '2019-01-01 18:15:12', 8),  
(11111118, 11111126, '2019-01-01 18:15:12', 9),  
(11111118, 11111126, '2019-01-01 18:15:12', 10),  
(11111125, 62653134, '2015-11-02 16:00:34', 7),
```



```
(11111125, 62653134, '2015-11-02 16:00:34', 8),  
(11111125, 62653134, '2015-11-02 16:00:34', 9),  
(11111125, 62653134, '2015-11-02 16:00:34', 10),  
(11111121, 62653134, '2020-06-22 09:30:23', 7),  
(11111121, 62653134, '2020-06-22 09:30:23', 8),  
(11111121, 62653134, '2020-06-22 09:30:23', 9),  
(11111121, 62653134, '2020-06-22 09:30:23', 10);
```

```
INSERT INTO `Contiene` (`ID_Form`, `ID_Preg`) VALUES
```

```
(1, 1),  
(1, 2),  
(1, 3),  
(2, 1),  
(2, 2),  
(2, 3),  
(3, 4),  
(3, 5),  
(3, 6),  
(4, 7),  
(4, 8),  
(4, 9),  
(4, 10);
```

```
INSERT INTO `Tiene` (`ID_Form`, `ID_Preg`, `ID_Resp`, `FechaR`) VALUES
```

```
(1, 1, 1, '2018-06-01 13:00:20'),  
(1, 2, 2, '2020-08-18 16:00:23'),  
(1, 3, 3, '2010-08-06 19:30:43'),  
(2, 1, 1, '2018-06-01 13:00:20'),  
(2, 2, 2, '2020-08-18 16:00:23'),
```

(2, 3, 3, '2010-08-06 19:30:43'),
(3, 4, 4, '2018-06-01 13:00:20'),
(3, 5, 5, '2020-08-18 16:00:23'),
(3, 6, 6, '2010-08-06 19:30:43'),
(4, 7, 7, '2018-06-01 13:00:20'),
(4, 8, 8, '2020-08-18 16:00:23'),
(4, 9, 9, '2010-08-06 19:30:43'),
(4, 10, 10, '2010-08-06 18:15:12');

INSERT INTO `Posee` (`ID_Estado`, `CI_Paciente`, `Fechainicio_E`) VALUES
(1, 55762293, '2020-10-06'),
(2, 54200707, '2020-10-06'),
(3, 52137108, '2020-10-06'),
(4, 52582991, '2020-10-06'),
(1, 62653134, '2020-10-06'),
(2, 11111111, '2020-10-06'),
(3, 11111112, '2020-10-06'),
(4, 11111113, '2020-10-06'),
(1, 11111114, '2020-10-06'),
(2, 11111115, '2020-10-06'),
(3, 11111116, '2020-10-06'),
(4, 11111117, '2020-10-06'),
(1, 11111126, '2020-10-06');
/**/

/*PERMISOS USUARIOS*/

CREATE ROLE 'GSAAadministradores';
CREATE ROLE 'GSAMedicos';
CREATE ROLE 'GSAAuxiliares';



```
CREATE ROLE 'GSAPacientes';

/*Todas las tablas*/

GRANT ALL PRIVILEGES ON ProyectoGSA.* TO 'GSAAAdministradores';

GRANT CREATE USER ON *.* TO 'GSAAAdministradores';

/*Persona*/

GRANT SELECT ON ProyectoGSA.Persona TO
'GSAMedicos','GSAAuxiliares','GSAPacientes';

/*Persona_Teléfono*/

GRANT SELECT ON ProyectoGSA.Persona_Teléfono TO
'GSAMedicos','GSAAuxiliares';

/*Paciente*/

GRANT SELECT ON ProyectoGSA.Paciente TO
'GSAMedicos','GSAAuxiliares','GSAPacientes';

/*Respuesta*/

GRANT INSERT, SELECT ON ProyectoGSA.Respuesta TO 'GSAMedicos';

GRANT INSERT ON ProyectoGSA.Respuesta TO 'GSAAuxiliares';

/*Analisis_clinico*/

GRANT INSERT, SELECT, UPDATE ON ProyectoGSA.Analisis_clinico TO
'GSAMedicos';

GRANT INSERT, SELECT ON ProyectoGSA.Analisis_clinico TO
'GSAAuxiliares';

GRANT SELECT ON ProyectoGSA.Analisis_clinico TO 'GSAPacientes';

/*Realiza*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Realiza TO
'GSAMedicos';

GRANT SELECT, INSERT ON ProyectoGSA.Realiza TO 'GSAAuxiliares';

GRANT SELECT ON ProyectoGSA.Realiza TO 'GSAPacientes';

/*Demanda*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Demanda TO
'GSAMedicos';

/*Atiende*/
```



```
GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Atiende TO
'GSAMedicos';

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Atiende TO
'GSAAuxiliares';

GRANT SELECT ON ProyectoGSA.Atiende TO 'GSAPacientes';

/*Formulario*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Formulario TO
'GSAMedicos','GSAAuxiliares';

/*Tiene*/

GRANT SELECT, INSERT, UPDATE, DELETE ON ProyectoGSA.Tiene TO
'GSAMedicos','GSAAuxiliares';

/*Diagnostico*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Diagnostico TO
'GSAMedicos';

GRANT INSERT, UPDATE ON ProyectoGSA.Diagnostico TO 'GSAAuxiliares';

GRANT SELECT ON ProyectoGSA.Diagnostico TO 'GSAPacientes';

/*Tratamiento*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Tratamiento TO
'GSAMedicos';

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Tratamiento TO
'GSAAuxiliares';

GRANT SELECT ON ProyectoGSA.Tratamiento TO 'GSAPacientes';

/*Pregunta*/

GRANT ALL PRIVILEGES ON ProyectoGSA.Pregunta TO
'GSAMedicos','GSAAuxiliares';

/*Sugiere */

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Sugiere TO
'GSAMedicos','GSAAuxiliares';

/*Responde*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Responde TO
'GSAMedicos';

/*Requiere*/
```



GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Requiere TO
'GSAMedicos';

/*Contiene*/

GRANT ALL PRIVILEGES ON ProyectoGSA.Contiene TO
'GSAMedicos','GSAAuxiliares';

/*Estado*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Estado TO
'GSAMedicos';

/*Elabora*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Elabora TO
'GSAMedicos';

GRANT SELECT ON ProyectoGSA.Elabora TO 'GSAPacientes';

/*Posee*/

GRANT SELECT, INSERT, UPDATE ON ProyectoGSA.Posee TO
'GSAMedicos','GSAAuxiliares';

/*Partes Cuerpo*/

GRANT SELECT, INSERT ON ProyectoGSA.PartesCuerpo TO 'GSAMedicos';

/*Incluye*/

GRANT SELECT, INSERT ON ProyectoGSA.Posee TO 'GSAMedicos';

/*Rol*/

GRANT SELECT ON ProyectoGSA.Rol TO
'GSAMedicos','GSAAuxiliares','GSAPacientes';

/*Usuario_Rol*/

GRANT SELECT ON ProyectoGSA.Usuario_Rol TO
'GSAMedicos','GSAAuxiliares','GSAPacientes';

FLUSH PRIVILEGES;

/*USUARIOS*/

CREATE USER 'GSA_Med'@'192.168.1.209'

IDENTIFIED BY 'GSAMedico';

GRANT GSAMedicos TO 'GSA_Med'@'192.168.1.209';

CREATE USER 'GSA_Aux'@'192.168.1.209'



IDENTIFIED BY 'GSAAuxiliar';
GRANT GSAAuxiliares TO 'GSA_Aux'@'192.168.1.209';
CREATE USER 'GSA_Pac'@'192.168.1.209'
IDENTIFIED BY 'GSAPaciente';
GRANT GSAPacientes TO 'GSA_Pac'@'192.168.1.209';
CREATE USER 'GSA_Adm'@'192.168.1.209'
IDENTIFIED BY 'GSAAdministrador';
GRANT GSAAdministradores TO 'GSA_Adm'@'192.168.1.209';
GRANT SUPER ON *.* TO GSAAdministradores;
/**/
/*CONSULTAS OBLIGATORIAS*/
/*1.Mostrar nombre de los médicos y cantidad de pacientes que atendieron en
un determinado mes.*/
SELECT p.Primer_Nombre as 'Nombre', p.Primer_Apellido as 'Apellido',
COUNT(a.CI_Paciente) as 'Cantidad de pacientes atendidos'
FROM Persona as p
JOIN Personal_Medico as pm ON p.CI=pm.CI
JOIN Medico as m ON pm.CI=m.CI_Medico
JOIN Atiende as a ON m.CI_Medico=a.CI_Medico
WHERE Month(a.Fecha_Atiende)= 8 AND Year(a.Fecha_Atiende)=2020
GROUP BY p.Primer_Nombre, p.Primer_Apellido;
/*2.Mostrar Historial clínico de un paciente determinado (todos los diagnósticos
que tuvo un paciente con sus correspondientes fechas).*/
SELECT e.CI_Paciente as 'CI Paciente', d.ID_Diag as 'ID Diagnóstico',
d.Descripcion_Diag as 'Descripción diagnóstico', e.Fecha_Diag as 'Fecha de
elaboración'
FROM Diagnostico as d
JOIN Elabora as e ON d.ID_Diag=e.ID_Diag
WHERE CI_Paciente=62653134;
/*3.Nombre y apellido de los auxiliares de todos los médicos.*/
SELECT p.Primer_Nombre as 'Nombre', p.Primer_Apellido as 'Apellido'



```
FROM Persona as p
JOIN Personal_Medico as pm ON p.Cl=pm.Cl
ORDER BY p.Primer_Apellido;

/*4.Nombre de los pacientes y tipo de tratamiento que se encuentran realizando
al día de hoy.*/
SELECT p.Primer_Nombre as 'Nombre', p.Primer_Apellido as 'Apellido',
t.Nombre as 'Nombre de tratamiento'

FROM Persona as p
JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente
JOIN Realiza as r ON pa.Cl_Paciente=r.Cl_Paciente
JOIN Tratamiento as t ON r.ID_Trata=t.ID_Trata
WHERE r.FechaFinal = '9998-12-31 00:00:00' /*EN NUESTRO SISTEMA LA
FECHA FINAL SE REPRESENTA COMO 9998-12-31 00:00:00*/
ORDER BY p.Primer_Apellido;

/*5.De todas las consultas realizadas este mes, mostrar nombre del paciente,
síntoma que presenta y nombre del médico que lo atendió.*/
SELECT p.Primer_Nombre as 'Nombre paciente', p.Primer_Apellido as
'Apellido paciente', vsa.Dolor as 'Síntoma dolor', vsa.Fiebre as 'Síntoma fiebre',
vsa.Malestar as 'Síntoma malestar', vsa.Otro as 'Otro síntoma',
vnm.Primer_Nombre as 'Nombre médico', vnm.Primer_Apellido as 'Apellido
médico'

FROM Persona as p
JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente
JOIN Atiende as a ON pa.Cl_Paciente=a.Cl_Paciente
JOIN vista_SintomasAtiendeSD as vsa ON vsa.Cl=a.Cl_Paciente
JOIN vista_NombreMedico as vnm ON vnm.Cl=a.Cl_Medico
JOIN Elabora as e ON a.Fecha_Atiende=e.Fecha_Atiende
JOIN Diagnostico as d ON e.ID_Diag=d.ID_Diag
WHERE Month(e.Fecha_Atiende)= Month(now());

/*6.Listado de médicos por tipo de especialidad.*/
SELECT m.Cl_Medico as 'Cl Medico', p.Primer_Nombre as 'Nombre',
p.Primer_Apellido as 'Apellido', m.Especialidad
```

```
FROM Medico as m
JOIN Personal_Medico as pm ON pm.Cl= m.Cl_Medico
JOIN Persona as p ON p.Cl=pm.Cl
ORDER BY Especialidad;
/*7.Nombre de los pacientes que nunca consultaron en un mes determinado.*/
SELECT DISTINCT p.Primer_Nombre as 'Nombre', p.Primer_Apellido as
'Apellido'
FROM Persona as p
JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente
JOIN Atiende as a ON pa.Cl_Paciente=a.Cl_Paciente
WHERE Month(a.Fecha_Atiende) NOT IN ("3")
ORDER BY p.Primer_Apellido;
/*8.Días transcurridos entre la consulta y la realización de análisis para los
pacientes atendidos este mes.*/
SELECT ac.Cl_Paciente as 'Cl Paciente', a.Fecha_Atiende as 'Fecha de la
consulta', ac.Fecha_Analisis 'Fecha análisis' , datediff(ac.Fecha_Analisis,
a.Fecha_Atiende) as 'Días transcurridos entre la consulta y la realización de
análisis para los pacientes atendidos este mes'
FROM Atiende as a
JOIN Analysis_Clinico as ac ON a.Fecha_Atiende=ac.Fecha_Atiende
WHERE MONTH(a.Fecha_Atiende)=Month(now()) AND
YEAR(a.Fecha_Atiende)=Year(now());
/*CONSULTAS OPCIONALES*/
/*9.Mostrar el estado en el que se encuentran los pacientes al día de hoy, es
decir, nombre del paciente y nombre del estado.*/
SELECT p.Primer_Nombre as 'Nombre', p.Primer_Apellido as 'Apellido',
e.Nombre_Estado as 'Nombre estado'
FROM Persona as p
JOIN Paciente as pa ON pa.Cl_Paciente=p.Cl
JOIN Posee as po ON pa.Cl_Paciente=po.Cl_Paciente
JOIN Estado as e ON e.ID_Estado=po.ID_Estado;
/*10.Análisis y/o estudios realizados a un paciente determinado en este año.*/

```

```
SELECT CI_Paciente as 'CI Paciente',TipoAnalisis as 'Tipo análisis', Resultado
as 'Resultado análisis', Fecha_Analisis as 'Fecha de elaboración del análisis'
FROM Analisis_Clinico
WHERE CI_Paciente=11111117 AND YEAR(Fecha_Analisis)=YEAR(now());
/*11.Listar todas las enfermedades con su correspondiente tipo ordenadas
alfabéticamente por tipo.*/
SELECT DISTINCT Tipo_EnfermedadActual as 'Tipo enfermedad',
Enfermedad_Actual as 'Enfermedad'
FROM Atiende
ORDER BY Tipo_EnfermedadActual ASC;
/*12.Promedio de consultas diarias en el mes.*/
SELECT count(a.Fecha_Atiende) AS 'Numero de consultas en el mes',
COUNT(a.Fecha_Atiende)/30 AS 'Promedio'
From Atiende as a
WHERE MONTH(a.Fecha_Atiende)=Month(now()) AND
YEAR(a.Fecha_Atiende)=Year(now());
/*13.Mostrar la enfermedad por la que han consultado la mayor cantidad de
pacientes.*/
SELECT Motivo, COUNT(Motivo) AS 'Veces consultado/a'
FROM Atiende
GROUP BY Motivo
ORDER BY COUNT(Motivo) DESC
LIMIT 1;
/*14.Mostrar nombre de los pacientes que presentan la enfermedad covid 19 y
son mayores de 60 o presentan una enfermedad anterior de tipo respiratorio.
Poner como encabezado en la salida “Persona de riesgo COVID”.*/
SELECT p.CI, p.Primer_Nombre, p.Primer_Apellido as 'Persona de riesgo
COVID'
FROM Persona as p
JOIN Paciente as pa ON p.CI=pa.CI_Paciente
JOIN Atiende as a ON pa.CI_Paciente=a.CI_Paciente
```



WHERE a.Enfermedad_Actual='COVID 19' AND TIMESTAMPDIFF(YEAR, p.Fecha_Nac, curdate())>60 OR Tipo_EnfermedadAnterior='Respiratorio';

/*15.Mostrar nombre, apellido del paciente y las preguntas y respuestas del formulario aplicado en la consulta de una fecha determinada.*/

SELECT DISTINCT p.Primer_Nombre as 'Nombre', p.Primer_Apellido as 'Apellido', pr.TextoPreg as 'Pregunta', re.TextoResp 'Respuesta'

FROM Persona as p

JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente

JOIN Atiende as a ON a.Cl_Paciente=pa.Cl_Paciente

JOIN Responde as r ON r.Fecha_Atiende=a.Fecha_Atiende

JOIN Respuesta as re ON re.ID_Resp=r.ID_Resp

JOIN Tiene as t ON t.ID_Resp=re.ID_Resp

JOIN Pregunta as pr ON pr.ID_Preg=t.ID_Preg

JOIN Contiene as c ON c.ID_Preg=pr.ID_Preg

WHERE YEAR(r.Fecha_Atiende)=2018 AND MONTH(r.Fecha_Atiende)=06
AND DAY(r.Fecha_Atiende) =1;

/*16.Médico que atendió más consultas en un mes determinado.*/

SELECT m.Cl_Medico as 'Cl Medico', p.Primer_Nombre as 'Nombre',
p.Primer_Apellido as 'Apellido', COUNT(a.Fecha_Atiende) as 'Cantidad de
consultas atendidas'

FROM Persona as p

JOIN Personal_Medico as pm ON p.Cl=pm.Cl

JOIN Medico as m ON pm.Cl=m.Cl_Medico

JOIN Atiende as a ON m.Cl_Medico=a.Cl_Medico

WHERE Month(a.Fecha_Atiende)= 11 AND Year(a.Fecha_Atiende)=2020

GROUP BY a.Fecha_Atiende

ORDER BY COUNT(a.Fecha_Atiende) DESC

LIMIT 1;

/*17.Mostrar nombre de los pacientes que no están realizando ningún
tratamiento.*/

SELECT DISTINCT pa.Cl_Paciente as 'Cl Paciente', p.Primer_Nombre as 'Nombre', p.Primer_Apellido as 'Apellido'

```
FROM Persona as p
JOIN Paciente as pa ON pa.CI_Paciente=p.CI
LEFT JOIN Realiza as re ON pa.CI_Paciente=re.CI_Paciente
LEFT JOIN Tratamiento as tr ON tr.ID_Trata=re.ID_Trata
WHERE pa.CI_Paciente NOT IN (SELECT DISTINCT CI_Paciente FROM
Realiza) OR re.FechaFinal < now();

/*18.Mostrar nombre de los pacientes que consultaron más de 5 veces en el
año.*/
SELECT p.CI as 'CI Paciente', p.Primer_Nombre as 'Nombre',
p.Primer_Apellido as 'Apellido', COUNT(a.Fecha_Atiende) as 'Veces que
consultaron'
FROM Persona as p
JOIN Paciente as pa ON p.CI=pa.CI_Paciente
JOIN Atiende as a ON pa.CI_Paciente=a.CI_Paciente
WHERE YEAR(a.Fecha_Atiende)=Year(now())
GROUP BY p.CI
HAVING COUNT(a.Fecha_Atiende) > 5;

/*19.Mostrar el nombre de los 3 pacientes que consultaron más veces en los
últimos 40 días.*/
SELECT a.CI_Paciente as 'CI', Primer_Nombre AS 'Nombre', Primer_Apellido
as 'Apellido', count(a.Fecha_Atiende) as 'Veces que consultó'
FROM Persona as p
JOIN Atiende as a ON a.CI_Paciente=p.CI
WHERE datediff(NOW(),a.Fecha_Atiende)<=40
GROUP BY a.CI_Paciente
ORDER BY count(a.Fecha_Atiende) DESC
LIMIT 3;

/*20.Patología o enfermedad que padecen más pacientes.*/
SELECT Enfermedad_Actual as 'Patología o enfermedad que padecen más
pacientes'
FROM Atiende
```



```
GROUP BY Enfermedad_Actual  
ORDER BY COUNT(Enfermedad_Actual) DESC  
LIMIT 1;
```

Política de respaldos:

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
21:00 hs	DIF.	INC.	INC.	INC.	INC.	INC.	
00:00 hs							COM.



Requisitos de la asignatura Sistemas Operativos

Requerimientos de la asignatura de Sistemas Operativos III
Relevamiento y justificación del Sistema Operativo a utilizar tanto en las terminales de los usuarios como en el servidor.

Sistema operativo a utilizar en las terminales de usuarios

Windows 10 Enterprise 2019 LTSC (Tipo VLK)

Requisitos mínimos:

- Procesador: 1 GHz o más rápido o sistema en un chip (SoC)
- RAM: 1 GB para 32 bits o 2 GB para 64 bits
- Espacio en disco duro: 16 GB para un SO de 32 bits o 32 GB para un SO de 64 bits
- Tarjeta gráfica: DirectX 9 o posterior con un controlador WDDM 1.0
- Pantalla: 800x600

Soporte:

- Fecha inicial del ciclo de vida: 13/11/2018
- Fecha final del soporte estándar: 9/1/2024
- Fecha final soporte extendido: 9/1/2029

Características de soporte:

Al comprar el licenciamiento de Windows accedemos y contamos con el soporte y apoyo por parte de los técnicos especialistas, la ayuda en línea y la comunidad de Microsoft mientras que la fecha de soporte sea vigente, por lo que garantizamos una solución a cualquier problema existente.

Precio del licenciamiento:

21,287 dólares en total para 63 licencias Microsoft Windows 10 Enterprise Edition

Total Parcial	\$ 19,530.00
Impponible	\$ 19,530.00
Tasa de Impuesto	9.000%
Impuesto Adeudado	\$ 1,757.70
Envío	
Total Debido	\$ 21,287.70



Justificación

Se elige como sistema operativo Windows ya que el Software de Bitácora Médica será desarrollado en Visual Basic.NET que utilizará el framework .NET. Además se comprarán licencias VLK (Volume Licesing Key) ya que habrá un total de 63 puestos de trabajo (equipos de terminales de trabajo) en la institución.

La razón de la elección de esta versión no es por las características que la versión Enterprise ofrece, sino por la versión Enterprise 2019 LTSC (Long Term Servicing Channel) que esta nos brinda un soporte a mayor plazo (**10 años**), además esta versión no incluye Microsoft Edge, Cortana, ni "Modern-Apps", esto hace que el sistema tenga una leve mejora de rendimiento (Mayor desempeño por parte del sistema al procesar distintas actividades ya que no utiliza el consumo excesivo de la memoria y el disco) a diferencia de otras versiones, a su vez la versión LTSC no recibe actualizaciones semi-anuales otorgándole al sistema ser más estable (Tiene menos fallos comparados a otras versiones) por más tiempo y que no reciba cambios bruscos de forma habitual, sino que solo recibirá las actualizaciones imprescindibles. Esto favorece al equipo ya que solo se utilizará para el uso del Software de Bitácora Médica y no tendrá herramientas innecesarias.



Sistema operativo a utilizar en los servidores

CentOS 8 (Linux)

Requisitos mínimos:

- Procesador: 2 GHz o superior
- RAM: 1 GB
- Espacio en disco duro: 20 GB

Soporte:

- Fecha inicial del ciclo de vida: 24/09/2019
- Fecha final del soporte estándar: 05/2024
- Fecha final soporte extendido: 31/05/2029

Características de soporte:

CentOS 8 tiene una amplia comunidad y varios foros en los cuales basarnos e informarnos para ofrecerle y brindarle soluciones, además CentOS 8 y RHEL comparten características similares por lo cual también podemos apoyarnos en el soporte de RHEL para prestar y ofrecer distintas y amplias soluciones.

Precio del licenciamiento:

CentOS 8 es un Software libre distribuido bajo la licencia pública general de GNU/Linux, es decir, es de costo gratuito, además CentOS 8 es un sistema operativo de código abierto.



Justificación

El servidor utilizará como sistema operativo CentOS 8 (Linux) ya que la institución solicita la instalación de infraestructura de Linux en los servidores.

La elección de CentOS se debe a que es un sistema operativo basado en RHEL (Red Hat Enterprise Linux) por lo cual cuenta con la seguridad de RedHat, ya que CentOS toma las funciones de RHEL y las adapta quitando términos y sintaxis utilizados en RHEL, esto facilita en gran parte la administración ya que podemos adquirir información de distintas fuentes de consulta.

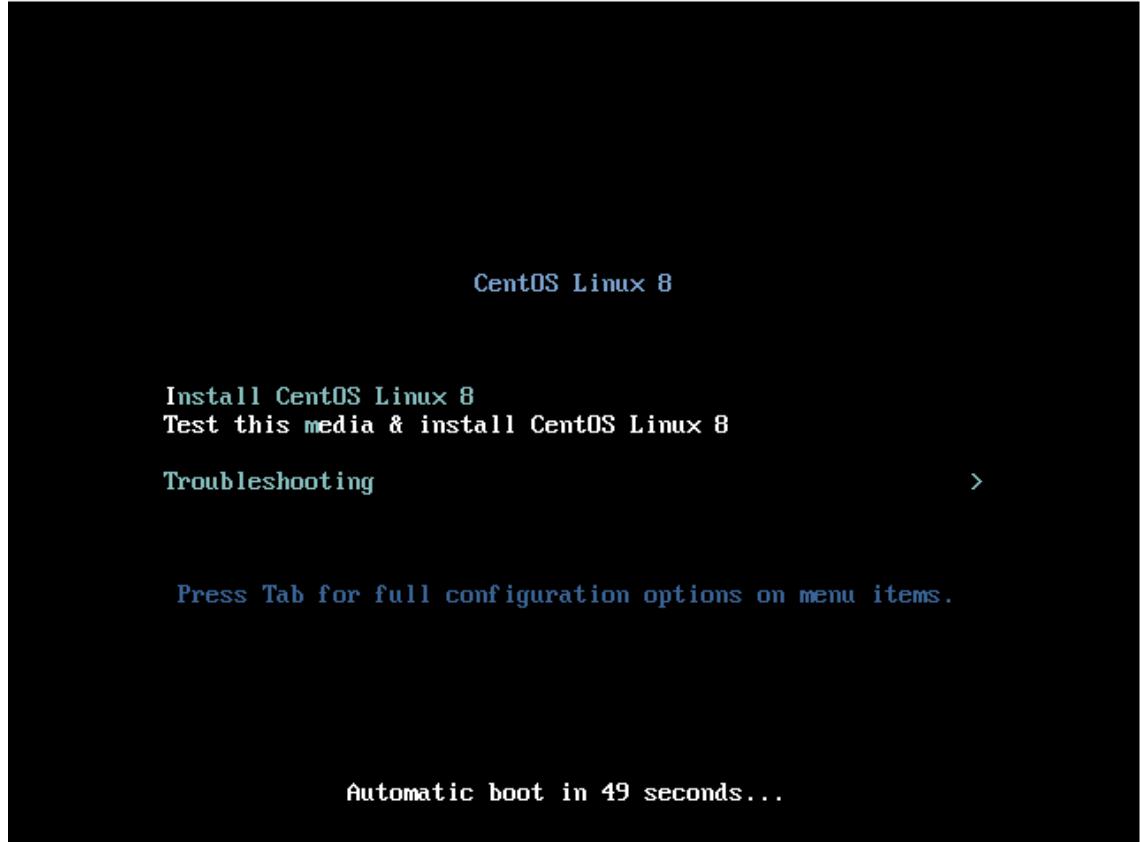
CentOS es más estable (Reduce el riesgo de caídas, errores y pérdida de rendimiento) ya que recibe actualizaciones con poca frecuencia y a su vez es eficaz en el consumo de recursos, a comparación con otras distribuciones de Linux tiene la capacidad de procesar tareas de forma más rápida y eficiente. También debido al vínculo con Red Hat, CentOS ofrece similares funciones de seguridad y firewall, además esta distribución admite la extensión [SELinux](#) (Security Enhanced Linux).

La preferencia de CentOS 8 ante otras versiones como CentOS 7, es que CentOS 8 recibe soporte hasta períodos de 2029 (10 años desde su lanzamiento).



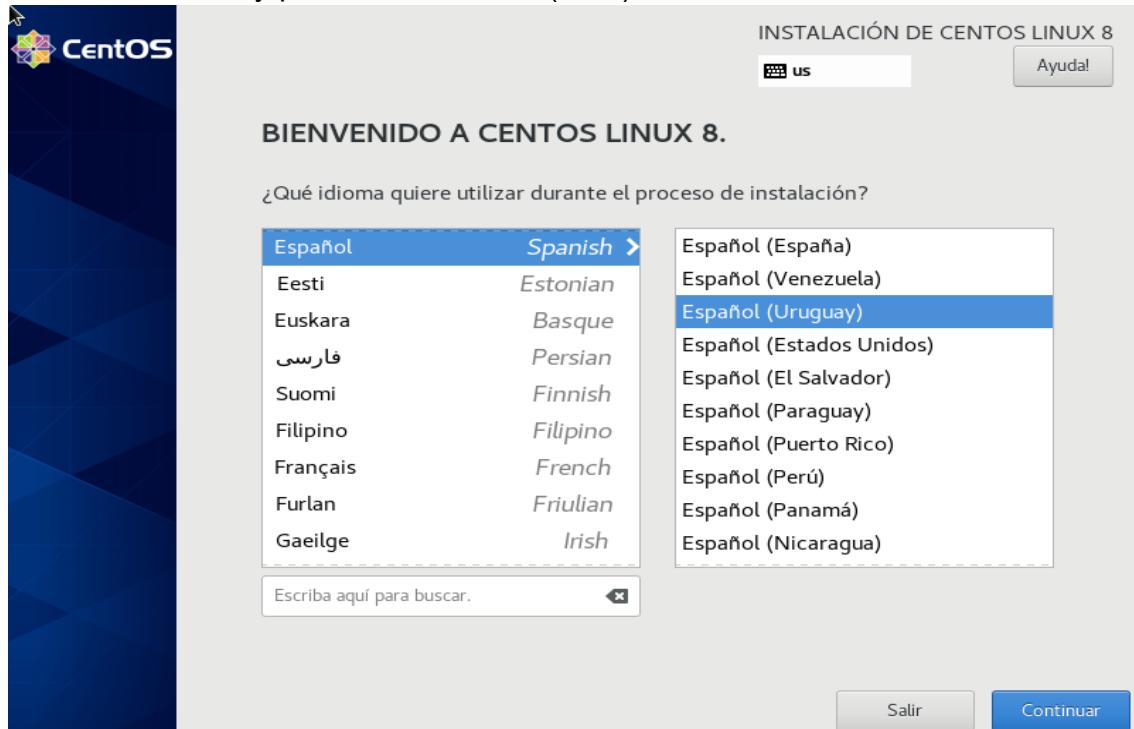
Manual de instalación del Sistema Operativo en el servidor

1. Una vez insertado el ISO o DVD de CentOS 8 se pasa a iniciar el sistema operativo.





2. Cuando se encuentre en esta pantalla seleccionará la opción “Install CentOS Linux 8” y presionara ENTER(Intro).



3. En esta pantalla seleccionará el lenguaje a preferencia, una vez elegido le dará al botón “Continuar”.





4. Acá podrá cambiar distintas configuraciones como “Fecha y hora” o cambiar el idioma nuevamente, una vez finalizadas estas configuraciones entrará a “Selección de software” en la sección “Software”.

5. En este caso seleccionará “Instalación mínima” ya que para las funciones que va a realizar el servidor este tipo de instalación basta, una vez seleccionado le dará a “Hecho”.

DESTINO DE LA INSTALACIÓN

Hecho

INSTALACIÓN DE CENTOS LINUX 8

latam Ayuda!

Selección de dispositivo

Seleccione los dispositivos en que le gustaría instalar. Se mantendrán sin tocar hasta que pulse el botón «Comenzar instalación» del menú principal.

Discos estándares locales

- 20 GiB
 - ATA VBOX HARDDISK
 - sda / 20 GiB libre

Los discos que se dejen aquí sin seleccionar no se tocarán.

Discos especializados y de red

Añadir un disco...

Los discos que se dejen aquí sin seleccionar no se tocarán.

Configuración de almacenamiento

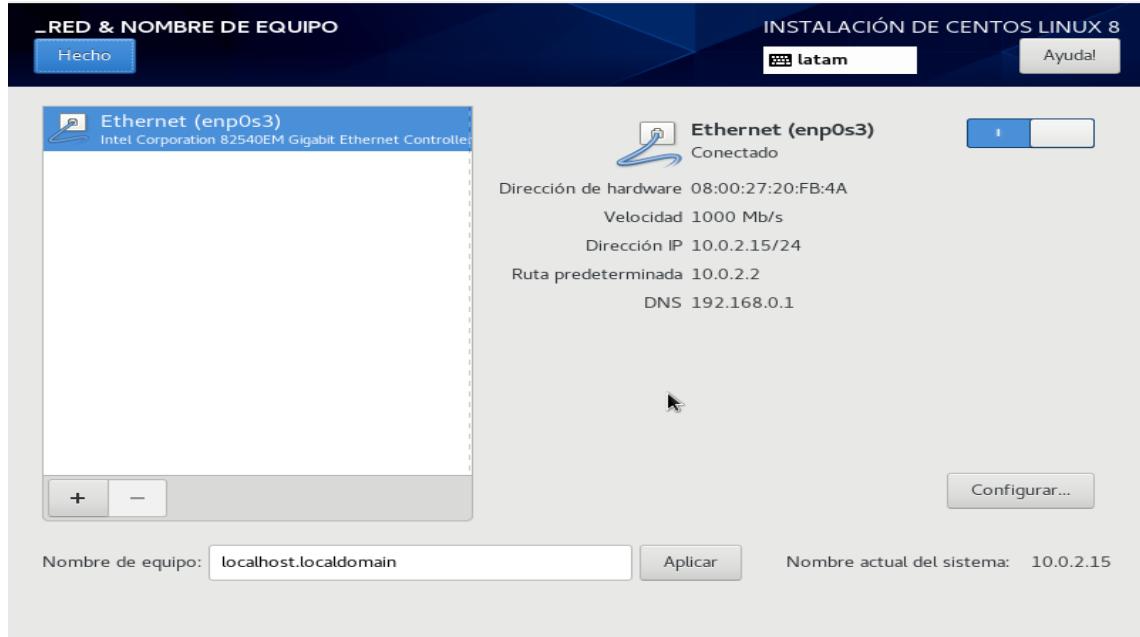
(*) Automática () Personalizada

Quiero dejar más espacio disponible.

[Resumen completo del disco y el cargador de arranque...](#)

1 disco seleccionado; 20 GiB de capacidad; 20 GiB libres [Actualizar...](#)

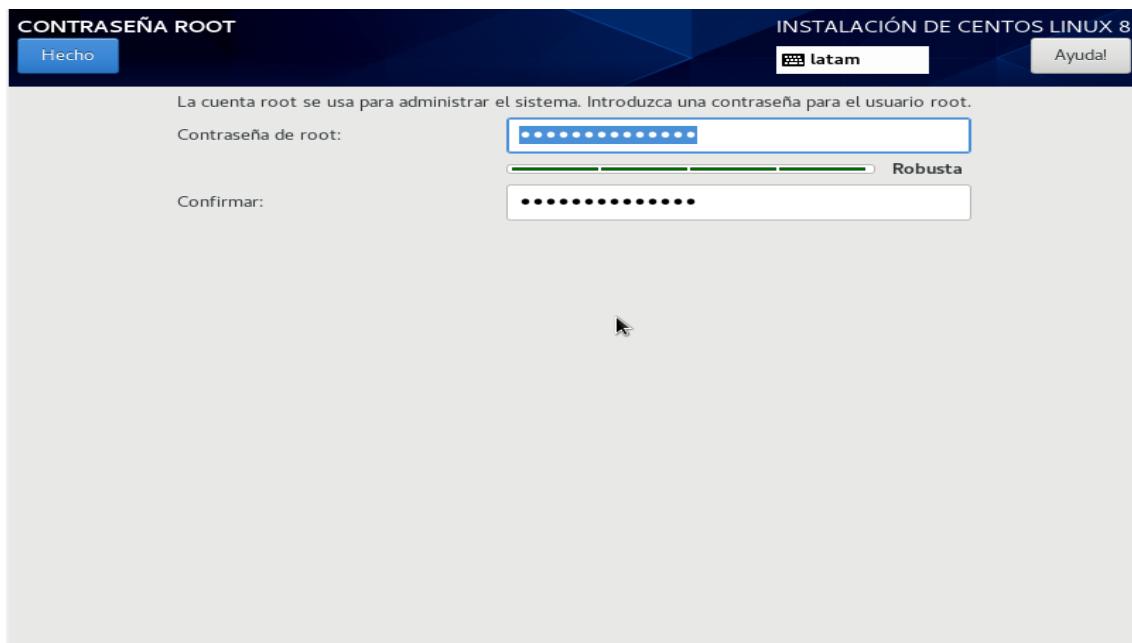
6. Una vez realizado el punto anterior (5.) debe volver a la pantalla de inicio llamada “Resumen de instalación” (3.) y entrará en la opción “Destino de la instalación” en la sección “Sistema”, una vez allí seleccionará el disco a usar y le dará a “Hecho”.



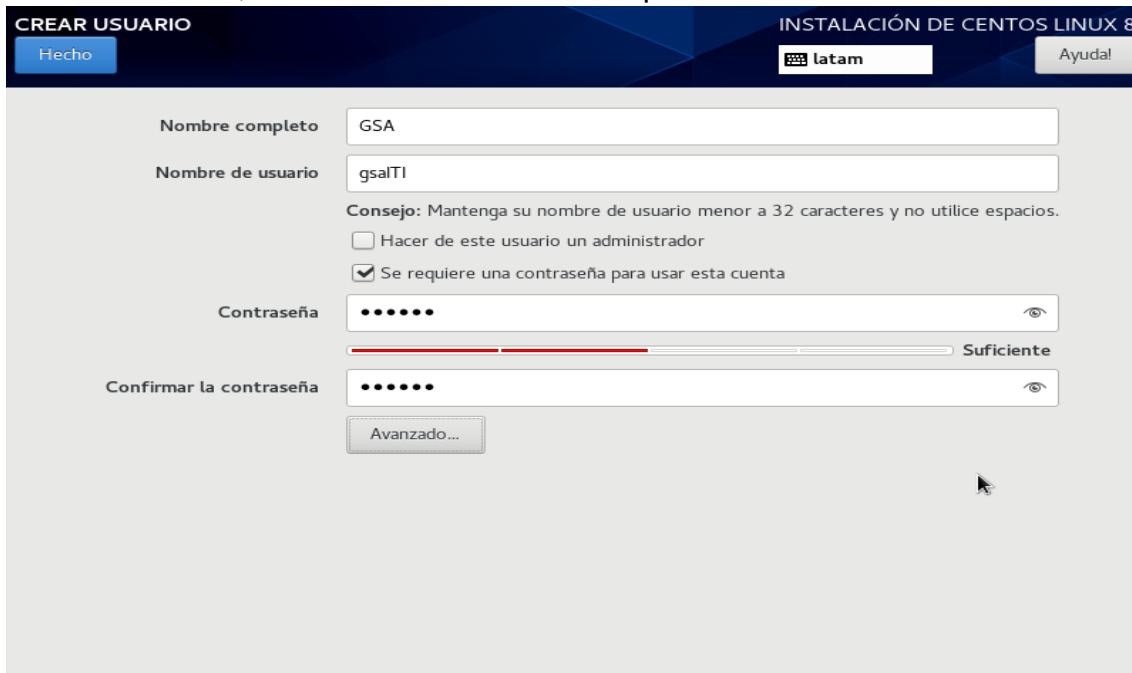
7. Realizado el punto anterior (7.) volverá a “Resumen de la instalación” (3.) e entrará a “Red y nombre de equipo” en la sección “Sistema”, una vez activado el puerto Ethernet le dará a “Hecho” donde volverá a la pantalla de “Resumen de instalación” y deberá darle a “Empezar instalación”.



8. En esta pantalla mostrará el progreso de la instalación del sistema operativo, de mientras deberá de crear una contraseña para el usuario root, una vez creada le dará a “Hecho” y volverá a la pantalla anterior.



9. Una vez configurada la contraseña para el usuario root, se procederá a crear un usuario, una vez rellenados los campos le dará a “Hecho”.



10. Ya realizados el punto 8 y 9 (8., 9.) esperará a que finalicé la instalación.

Una vez finalizada le dará a “Finalizar configuración” y luego a “Reiniciar” (Antes de reiniciar retiré el DVD o ISO correspondiente al S.O.).



Estudio de los diferentes roles de usuarios del sistema

1. Personal total de la institución médica

Gerente: Es el empleado que planea, organiza, coordina y dirige las actividades de una institución.

NO TOCA EL SISTEMA

Auxiliar de Servicio: Es el empleado responsable de mantener, el orden, la higiene y el ambiente de trabajo de la institución limpia.

NO TOCA EL SISTEMA

Seguridad: Ejerce la vigilancia del lugar, así como la protección de las personas que puedan estar por allí.

NO TOCA EL SISTEMA

Atención al cliente: Son los encargados de relacionarse con los clientes para ayudarlos si tienen dudas o para anticiparse a sus necesidades.

NO TOCA EL SISTEMA

Farmacéutico: Persona que se dedica a dar y/o vender los medicamentos recetados por los médicos.

NO TOCA EL SISTEMA

Recepcionista: Persona encargada de atender el teléfono, así como al público para ayudarlos con sus problemas.

NO TOCA EL SISTEMA

Conductor: Es el funcionario que tiene de cargo la conducción de ambulancias y vehículos de urgencia.

NO TOCA EL SISTEMA

Medico: Es el que tiene la responsabilidad de tomar las decisiones que afectan el cuidado del paciente, incluyendo el diagnóstico y el tratamiento.

TOCA EL SISTEMA

Auxiliar Medico: Puede llevar a cabo ciertos procedimientos tales como tomar antecedentes médicos, realizar exámenes físicos tomar muestras tanto de sangre como orina, aplicar vacunas, etc., bajo la orientación de un médico.

TOCA EL SISTEMA

Paciente: Es la persona que solicita asistencia médica debido a un dolor o malestar.

TOCA EL SISTEMA

Administrador sistema: Es el encargado de administrar el sistema, es decir realizar el mantenimiento del sistema, dar de alta y baja a los médicos y auxiliares, a su vez también administra la base de datos.

TOCA EL SISTEMA

Administrador técnico: Son los empleados pertenecientes al soporte IT, cuya función es proporcionar asistencia con el hardware o software de una o algún dispositivo electrónico o mecánico.

NO TOCA EL SISTEMA

2. Personal de la institución médica que utiliza el sistema

Médico: Las funciones que realizan dentro del sistema el personal médico son las siguientes:

- Ingresar datos
- Crear formularios
- Consultar información

Auxiliar Médico: Las funciones que realizan dentro del sistema los Auxiliar médico son las siguientes:

- Ingresar datos
- Crear formularios

Paciente: Las funciones que realizan dentro del sistema los pacientes son las siguientes:

- Consultar información de diagnóstico
- Consultar resultado de análisis

Administrador sistema: Las funciones que realizan dentro del sistema los administradores son las siguientes:

- Agregar Médicos
- Agregar Auxiliares
- Ingresar información a la Base de Datos
- Modificar permisos de acceso

3. Creación de usuarios y grupos en los diferentes sistemas operativos

Terminales de trabajo

Usuarios: Médico, Auxiliar Médico, Paciente, Administrador sistema, Administrador técnico.

Grupos: Usuarios, Médicos, Auxiliares Médicos, Pacientes.

Justificación: Se crearán los usuarios mencionados anteriormente para cada persona que toque el sistema, a su vez también se creará el usuario del administrador técnico para que pueda proporcionar asistencia. Respecto a los grupos, se creará un grupo para todos los usuarios llamado Usuarios que tendrá los permisos básicos, a su vez se creará un grupo para cada rol, es decir Médicos, Auxiliares Médicos y Pacientes ya que ninguno realiza las mismas funciones dentro del sistema y se diferencian.

No se agregó un grupo de Administradores de sistema y/o Técnico porque solamente hay uno de cada uno en el Cliente.

Servidor

Usuarios: Administrador_Sistema, Administrador_Tecnico.

Grupos: Administradores_Sist, Administradores_Tec.

Justificación: Se crearán los usuarios “Administrador_Sistema” ya que serán los encargados que estarán administrando el servidor y la base de datos, a su vez también se crearán los usuarios “Administrador_Tecnico” los cuales serán los encargados de mantener el correcto funcionamiento de los componentes del servidor, además el motivo de creación de usuario es debido a que quizás en algún momento este tenga que utilizar el sistema operativo para ejecutar algún que otro Software de monitoreo o realizar alguna configuración.

Para estos usuarios se creó un grupo perteneciente a cada rol, como “Administradores_Sist” y “Administradores_Tec” ya que son más de un administrador técnico y más de un administrador de sistema en la empresa y comparten las



mismas funciones y permisos en el rol que les corresponde a cada uno.



Manual de Instalación de MySql junto a sus requisitos.

Requisitos mínimos:

CentOS 8 actualizado.
Un usuario con permisos de sudo o usuario root.
Conexión a internet.
CPU de 2 cores
2 GB de RAM
1.3 GB de espacio en el disco

Guía

1. Primero deberá actualizar los repositorios de la distribución así:

```
[root@10 ~]# sudo yum update
```

Cuando se le solicite, presione s y luego ENTER para confirmar que desea continuar.

2. Luego instalará el siguiente paquete de MySQL:

```
[root@10 ~]# sudo yum install -y mysql-server
```

En este caso, no le solicitará confirmar que desea continuar ya que ingreso -y en la línea de comandos.

3. Por defecto, el servicio estará inactivo, por lo que deberá ingresar el siguiente comando:

```
[root@10 ~]# sudo systemctl start mysqld
```

4. Para que el servicio se active cada vez que se inicie el sistema operativo, deberá ingresar el siguiente comando:

```
[root@10 ~]# sudo systemctl enable mysqld
```

5. Para configurar la seguridad de mysql, deberá ingresar el siguiente comando:

```
[root@localhost ~]# mysql_secure_installation
```

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords and improve security. It checks the strength of password and allows the users to set only those passwords which are secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: y_

Debe ingresar y/Y para confirmar el agregar un componente a la contraseña.



A continuación, le aparecerá en pantalla lo siguiente:

```
LOW Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 0
Please set the password for root here.

New password:
Re-enter new password:
Estimated strength of the password: 50
Do you wish to continue with the password provided?(Press y/Y for Yes, any other key for No) :
```

Deberá seleccionar un nivel de seguridad a la contraseña, 0 = bajo, 1 = medio y 2 = fuerte.

Una vez ingresada la contraseña, confirmará ingresando y/Y, luego le aparecerá en pantalla las siguientes 4 preguntas las cuales confirmará ingresando y/Y:

Acá le pregunta si quiere remover los usuarios anónimos.

```
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.
```

```
Remove anonymous users? (Press y/Y for Yes, any other key for No) : y
```

Acá le pregunta si quiere deshabilitar el login de root remoto.

```
Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.
```

```
Disallow root login remotely? (Press y/Y for Yes, any other key for No) : y
```

Acá le pregunta si quiere eliminar la base de datos que viene por defecto llamada "prueba" o "test".

```
By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.
```

```
Remove test database and access to it? (Press y/Y for Yes, any other key for No) : y
```



Acá le pregunta si quiere recargar los privilegios de las tablas.

```
Reloading the privilege tables will ensure that all changes  
made so far will take effect immediately.
```

```
Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
```

Una vez finalizadas estas 4 preguntas, estaría finalizada la configuración de seguridad de MySQL.

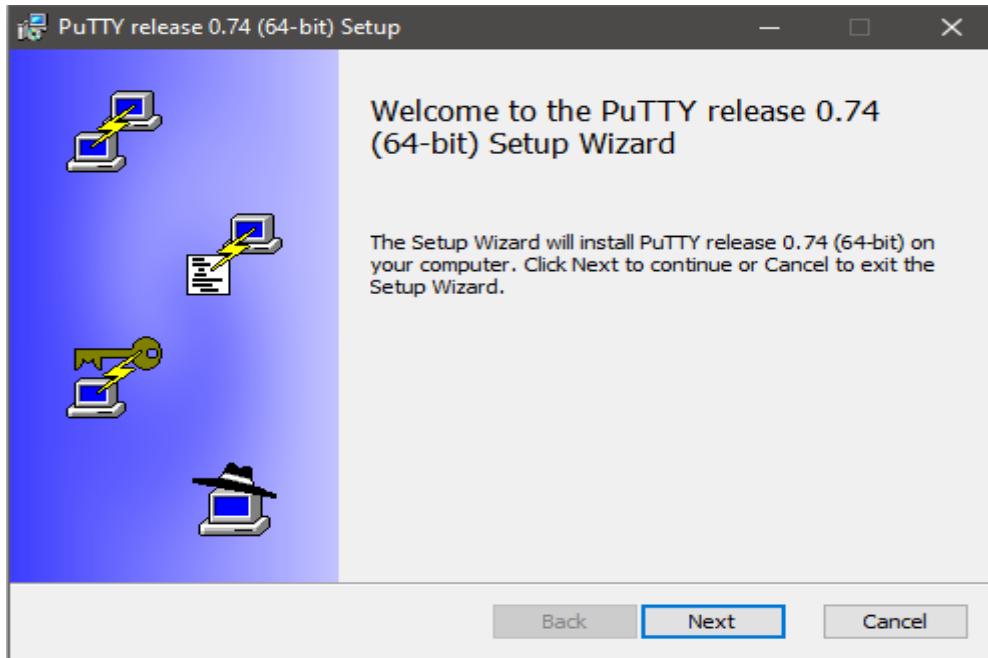


Configuración del servicio SSH en el cliente y el servidor.

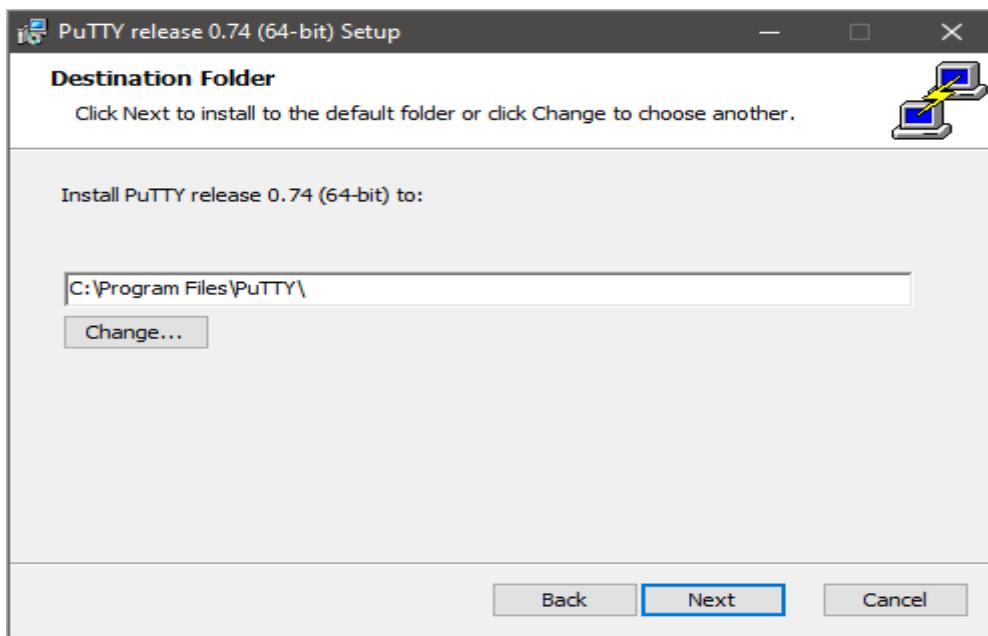
Terminales de trabajo

Se debe realizar la instalación del software PuTTY SSH en el puesto de trabajo de los clientes, a continuación, una breve explicación de la configuración:

Se debe de iniciar la instalación y presionar el botón de Next (Siguiente).

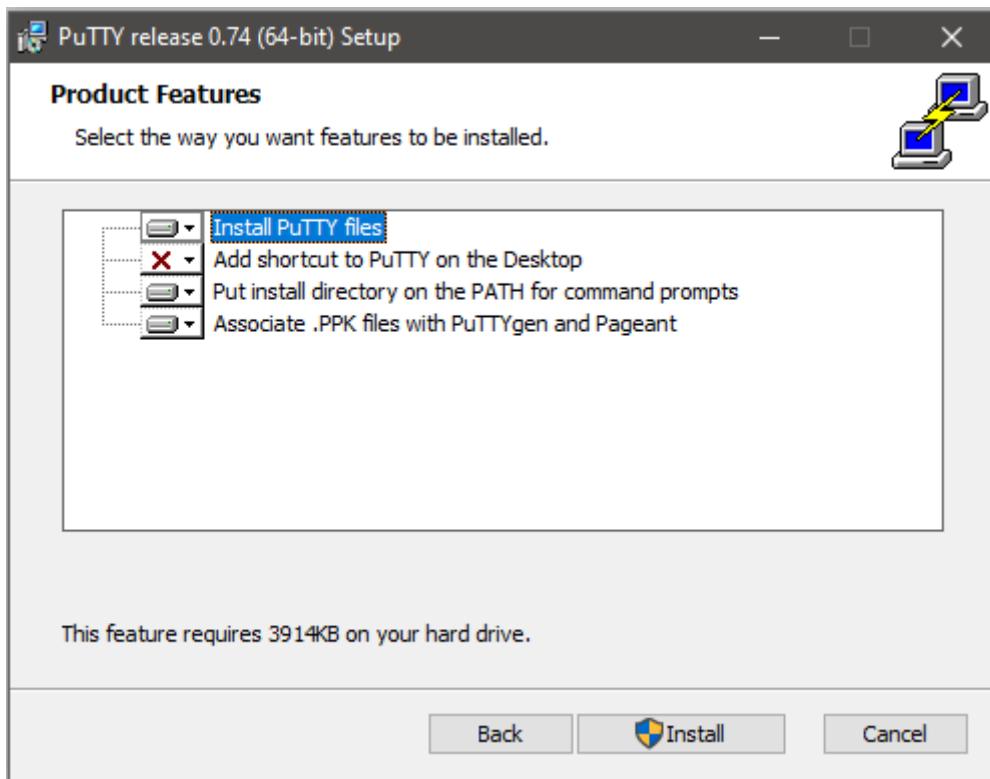


Seleccionará la unidad de almacenamiento en donde se instalará el programa y le dará a Next (Siguiente).

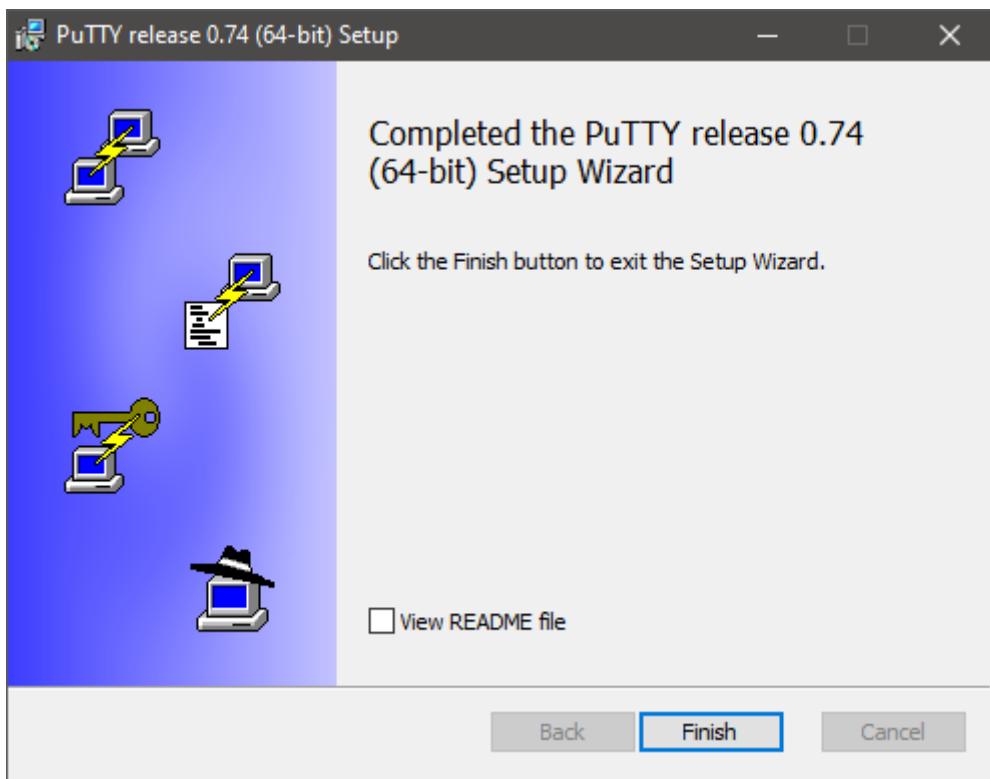




Luego presionará Install (Instalar) y le pedirá permisos de administrador.



Una vez finalizada la instalación le dará a Finish (Finalizar) y la configuración SSH en el cliente estará terminada.





Servidor

Se comprobará antes de todo que el sistema tenga implementado los 3 paquetes openssh, para ello se deberá ingresar el siguiente comando:

```
[root@localhost ~]# yum list installed | grep openssh
openssh.x86_64           8.0p1-4.el8_1          ::BaseOS
openssh-clients.x86_64     8.0p1-4.el8_1          ::BaseOS
openssh-server.x86_64      8.0p1-4.el8_1          ::BaseOS
```

- En CentOS al instalar el sistema operativo ya vienen instalados estos 3 paquetes.

Se realizaron los siguientes cambios en la configuración del archivo de configuración ubicado en /etc/ssh/sshd_config

Cambio de puerto

```
# Port 3899
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
```

Por defecto, viene con el puerto 22 establecido, al cambiar este dato estaremos mejorando significativamente la seguridad ya que no será un puerto estándar.

Permitir login de root

```
# Authentication:
#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

Por defecto, el root tiene permitido el login, al cambiar este dato a “no”, se mejorará la seguridad ya que la mayoría de intentos de ataques/ingresos al sistema se realizan con el usuario root.

No se podrá ingresar como usuario root, pero un usuario común podrá tomar la identidad del mismo con el comando su root.



Deshabilitar la autenticación por contraseña

```
# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
PasswordAuthentication yes
```

Cambiando esta simple opción la seguridad en el sistema aumentará ya que solo se ingresará mediante claves asimétricas permitidas en el archivo `~/.ssh/authorized_keys` y además permitirá automatizar las tareas en un futuro ya que no pedirá la contraseña para realizar alguna acción en el sistema.



Configuración de red

La configuración de red para cada interfaz será la siguiente, variando la dirección IP (IPADDR) según el servidor en el que se aplique la configuración.

```
echo "TYPE="Ethernet"
PROXY_METHOD="none"
BROWSER_ONLY="no"
BOOTPROTO="static"
DEFROUTE="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="yes"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_FAILURE_FATAL="no"
IPV6_ADDR_GEN_MODE="stable-privacy"
NAME="enp0s9"
DEVICE="enp0s9"
ONBOOT="yes"
IPADDR="192.168.10X.X"
PREFIX="24"
GATEWAY="192.168.10X.1"
DNS1="200.40.30.245"
DNS2="200.40.220.245"
```



Scripts

Donde quedan

Los scripts estarán situados en la carpeta denominada “Scripts” dentro del directorio del usuario(~/Scripts).

Como se ejecutan

Antes de ejecutar cualquier Script debe de asegurarse que este cuente con los permisos de ejecución, en el caso que no cuente con permisos de ejecución de los podrá asignar ingresado el siguiente comando en la consola:

chmod u+x [NOMBRE_SCRIPT]

Además para que se realicen correctamente todas las funciones del Script, este mismo se debe ejecutar como usuario Sudo.

A su vez se recomienda ejecutar el Script de la siguiente manera:

./[NOMBRE_SCRIPT]

Que hacen

shellScriptGSA.sh

Función: Shell Script que se utiliza para la agregación, modificación, modificación y consulta (ADMINISTRACIÓN) de los grupos y usuarios del sistema

scriptUyG.sh

Función: Shell Script que se utiliza para la creación de usuarios y grupos de acuerdo al estudio de roles hecho anteriormente y a su vez la agregación de los administradores al grupo Wheel de CentOS.

scriptRedes.sh

Función: Shell Script que se utiliza para la configuración semiautomática (Se debe ingresar una opción en el menú) de las redes en los distintos servidores.

BackupBD

Función: Shell Script que se utiliza para realizar una copia de seguridad de la base de datos y enviarla al servidor de respaldos.

BackupCompletoGSA

Función: Shell Script que se utiliza para realizar la copia de seguridad completa de los archivos detallados en la rutina de respaldos.



BackupIncremental00GSA

Función: Shell Script que se utiliza para realizar la copia de seguridad incremental (00:00hs) de los archivos detallados en la rutina de respaldos.

BackupIncremental13GSA

Función: Shell Script que se utiliza para realizar la copia de seguridad incremental (13:00) de los archivos detallados en la rutina de respaldos.

LogsGSA.sh

Función: Shell Script que se utiliza para visualizar los logs del sistema y filtrar por prioridad y servicio.

scriptBackup.sh

Función: Shell Script que se utiliza para programar las tareas de los respaldos, completos, incrementales y de la base de datos en la agenda crontab.

scriptConfigFWSE.sh

Función: Shell Script que se utiliza para aplicar la configuración en los servicios Firewall y SELinux que esta detallada en la carpeta.

scriptConfigSSH.sh

Función: Shell Script que se utiliza para aplicar la configuración en el archivo sshd_config que esta detallada en la carpeta.

ScriptCTBU.sh

Función: Shell Script que se utiliza para programar las tareas de los respaldos, completos, incrementales y de la base de datos en la agenda crontab a la hora de inicializar el servidor por primera vez.

ScriptInstalacionMysql.sh

Función: Shell Script que se utiliza para la instalación y configuración de MySQL.

ScriptRestauracion.sh

Función: Shell Script que se utiliza para realizar la restauración de los respaldos.



ScriptUpd.sh

Función: Shell Script que se utiliza para actualizar el sistema y programar la misma tarea en la agenda crontab.

ScriptVerLogs.sh

Función: Shell Script que se utiliza para visualizar los logs de los Scripts.

scriptMenu.sh

Función: Shell Script que se utiliza para conectar a través de un menú todos los scripts creados anteriormente.



Rutinas Backup (Respaldo)

Los archivos, directorios u ficheros de los cuales se realizarán copias en cada respaldo, serán los siguientes:

/etc/passwd: En este fichero se encuentran almacenados los usuarios de nuestro sistema operativo.

/etc/shadow: Este fichero almacena las contraseñas cifradas de los usuarios y demás información sobre el mismo.

/etc/sysconfig/network-scripts/: En este directorio se encuentran las configuraciones de todas las interfaces de red.

/etc/group: En este fichero se almacena la información de los distintos grupos existentes en el sistema.

/etc/ssh/sshd_config: En este fichero se encuentra la configuración del servicio SSH.

/etc/sudoers: En este fichero se encuentra la configuración de los usuarios y grupos que tienen habilitado el permiso sudo.

/Scripts: En este directorio se encuentran situados todos los scripts necesarios para la administración del sistema.

Respaldo completo

Tiempo: Cada domingo de la semana, todas las semanas del año, cada año a las 00:00 hs.

Motivo: Debido al alto consumo de almacenamiento y alto tiempo de transferencia, lo recomendable es hacer solamente un respaldo completo a la semana.

A su vez, la elección del día domingo se debe a que en la institución médica hay poco uso de la red y en consecuencia al alto tiempo de transferencia se eligieron las 00:00 hs como momento para realizar el respaldo.

Destino: Las copias se extraerán al servidor de respaldo instalado en la institución médica, para posteriormente ser respaldado en otra sucursal y en la nube.

Respaldo incremental (1)

Tiempo: Lunes a Sábado, de cada semana, de cada año a las 00:00 hs.

Motivo: Debido a que al consumir menos recursos y tener un tiempo de transferencia menor al completo, este respaldo se puede elaborar una vez por día y la elección de la hora es basada en el mismo criterio que en el de respaldo completo.

Destino: Las copias se extraerán al servidor de respaldo instalado en la institución médica, para posteriormente ser respaldado en otra sucursal y en la nube.

Respaldo incremental (2)

Tiempo: Todos los días, de cada semana, de cada año a las 13:00 hs.

Motivo: Se llevará a cabo un respaldo incremental en la mitad del día (13:00 hs) para evitar la pérdida de datos entre cada respaldo incremental (00:00hs) en el horario laboral.

Destino: Las copias se extraerán al servidor de respaldo instalado en la institución médica, para posteriormente ser respaldado en otra sucursal y en la nube.



Descripción de Logs (Scripts)

- Donde queda

Los Logs de nuestros scripts se encuentran almacenados en archivos de texto dentro del directorio ~/Logs, cada Script tiene su archivo de Logs asignado.

- Información contendida

Muestra la fecha y hora en formato ((día semana) (mes) (día mes) HH:MM:SS (zona horaria) (año)) y posteriormente le sigue un mensaje que notifica si la operación tuvo éxito o no.



Configuración Firewall y SELinux

Firewall

Previamente a realizar la configuración del servicio firewall, se estableció el inicio del servicio al levantar el sistema:

```
sudo systemctl enable firewalld
```

Para la configuración de Firewall se realizaron los siguientes cambios en la zona pública:

```
[gsaITI@localhost ~]$ sudo firewall-cmd --zone=public --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3 enp0s9
  sources:
  services: cockpit dhcpcd-client mysql ssh
  ports: 3899/tcp 3306/tcp
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Se añadieron los servicios mysql y ssh a la zona pública a través de los comandos:

```
[gsaITI@localhost ~]$ sudo firewall-cmd --zone=public --add-service=mysql --permanent
[gsaITI@localhost ~]$ sudo firewall-cmd --zone=public --add-service=ssh --permanent
```

A su vez se añadieron los puertos 3899 (Puerto SSH) y 3306 (Puerto MySQL) con protocolo tcp a través de los comandos:

```
[gsaITI@localhost ~]$ sudo firewall-cmd --zone=public --add-port=3899/tcp --permanent
[gsaITI@localhost ~]$ sudo firewall-cmd --zone=public --add-port=3306/tcp --permanent
```

Y también se eliminó el puerto 22 por motivos de seguridad a través del comando:

```
[gsaITI@localhost ~]$ sudo firewall-cmd --zone=public --remove-port=22/tcp --permanent
```



SELinux

Para la configuración de SELinux se añadieron los siguientes puertos SSH y MySQL:

```
[gsaITI@localhost ~]$ sudo semanage port -l | grep ssh  
ssh_port_t                      tcp      3899, 22
```

```
[gsaITI@localhost ~]$ sudo semanage port -l | grep mysqld  
mysqld_port_t                    tcp      1186, 3306, 63132-63164
```

A través de los comandos:

```
[gsaITI@localhost ~]$ sudo semanage port -a -t ssh_port_t -p tcp 3899  
[gsaITI@localhost ~]$ sudo semanage port -a -t mysqld_port_t -p tcp 3306
```

Requisitos de la asignatura Programación

1) Identificación de interfaces del sistema

-  [frmABMFormulario.vb](#)
-  [frmAdmPersonalMedico.vb](#)
-  [frmAnalisisClinico.vb](#)
-  [frmConsultaAnalisis.vb](#)
-  [frmConsultaDiagnostico.vb](#)
-  [frmCreditos.vb](#)
-  [frmDiagnostico.vb](#)
-  [frmEntrevistalnicial.vb](#)
-  [frmIngresoPacientes.vb](#)
-  [frmLogin.vb](#)
-  [frmPrincipal.vb](#)
-  [frmTratamiento.vb](#)
-  [Rutinas.vb](#)

2) Código para la comprobación de datos ingresados

Código que no permite el ingreso de números o símbolos

```
If Char.IsLetter(e.KeyChar) Then  
    e.Handled = False  
  
    ElseIf Char.IsControl(e.KeyChar) Then  
        e.Handled = False  
  
    ElseIf Char.IsSeparator(e.KeyChar) Then  
        e.Handled = False  
  
    Else  
        e.Handled = True  
  
        MessageBox.Show("No se permite ingresar números o  
símbolos.", "ADVERTENCIA")  
  
    End If
```



Código que no permite el ingreso de espacios

```
If Char.IsNumber(e.KeyChar) Then  
    e.Handled = False  
  
ElseIf Char.IsControl(e.KeyChar) Then  
    e.Handled = False  
  
ElseIf Char.IsSeparator(e.KeyChar) Then  
    e.Handled = True  
  
    MessageBox.Show("No se permite ingresar  
espacios.", "ADVERTENCIA")  
  
Else  
    e.Handled = False  
  
End If
```

Código que no permite el ingreso de letras

```
If Char.IsNumber(e.KeyChar) Then  
    e.Handled = False  
  
ElseIf Char.IsControl(e.KeyChar) Then  
    e.Handled = False  
  
ElseIf Char.IsSeparator(e.KeyChar) Then  
    e.Handled = False  
  
Else  
    e.Handled = True  
  
    MessageBox.Show("No se permite ingresar letras.",  
"ADVERTENCIA")  
  
End If
```

Para los casos en los que se utilizó más de un filtro, se utilizaron combinaciones de los códigos previamente mostrados.



Codigo Fuente:

Clases:

Analisis:

Public Class Analisis

```
Protected _ci As Integer  
Protected _cimedico As Integer  
Protected _id As Integer  
Protected _tipo As String  
Protected _fecha As Date  
Protected _fechaatiende As Date  
Protected _resultado As String  
Protected _estadoborrardop As Integer  
Protected _cn As ADODB.Connection  
'Metodos constructores  
Sub New()  
    _ci = 0  
    _cimedico = 0  
    _id = 0  
    _estadoborrardop = 0  
    _tipo = ""  
    _fecha = New Date  
    _fechaatiende = New Date  
    _resultado = ""  
    _cn = New ADODB.Connection  
End Sub  
Sub New(ByVal estadoborrardop As Integer, ByVal ci As Integer, ByVal  
cimedico As Integer, ByVal id As Integer, ByVal tipo As String, ByVal fecha As  
Date, ByVal fechaatiende As Date, ByVal resultado As String, ByVal cn As  
ADODB.Connection)  
    _ci = ci
```



```
_estadoborrado = estadoborrado  
_cimedico = cimedico  
_id = id  
_tipo = tipo  
_fecha = fecha  
_fechaatiende = fechaatiende  
_resultado = resultado  
_cn = cn
```

```
End Sub
```

```
'Get y set de las propiedades
```

```
Property estadoborrado As Integer
```

```
Get
```

```
    Return (_estadoborrado)
```

```
End Get
```

```
Set(ByVal value As Integer)
```

```
    _estadoborrado = value
```

```
End Set
```

```
End Property
```

```
Property ci As Integer
```

```
Get
```

```
    Return (_ci)
```

```
End Get
```

```
Set(ByVal value As Integer)
```

```
    _ci = value
```

```
End Set
```

```
End Property
```

```
Property cimedico As Integer
```

```
Get
```

```
    Return (_cimedico)
```



```
End Get  
Set(ByVal value As Integer)  
    _cimedico = value  
End Set  
End Property  
Property id As Integer  
    Get  
        Return (_id)  
    End Get  
    Set(ByVal value As Integer)  
        _id = value  
    End Set  
End Property  
Property tipo As String  
    Get  
        Return (_tipo)  
    End Get  
    Set(ByVal value As String)  
        _tipo = value  
    End Set  
End Property  
Property fecha As Date  
    Get  
        Return (_fecha)  
    End Get  
    Set(ByVal value As Date)  
        _fecha = value  
    End Set  
End Property
```



Property fechaatiende As Date

Get

 Return (_fechaatiende)

End Get

Set(ByVal value As Date)

 _fechaatiende = value

End Set

End Property

Property resultado As String

Get

 Return (_resultado)

End Get

Set(ByVal value As String)

 _resultado = value

End Set

End Property

Property cn As ADODB.Connection

Get

 Return (_cn)

End Get

Set(ByVal value As ADODB.Connection)

 _cn = value

End Set

End Property

Function Buscar() As Integer

 Dim sql As String

 Dim rs As New ADODB.Recordset

 Dim retorno As Integer = 0

 If cn.State = 0 Then



```
retorno = 1

Else

    sql = "select * from Analisis_Clinico where CI_Paciente=" & _ci & " AND
ID_Analisis=" & _id & ";"

    Try

        rs.Open(sql, _cn)

        Catch ex As Exception

            Return (4)

        End Try

        If rs.RecordCount = 0 Then

            retorno = 3

        Else

            _resultado = rs("Resultado").Value
            _fecha = rs("Fecha_Analisis").Value
            _tipo = rs("TipoAnalisis").Value

        End If

        rs.Close()

    End If

    Return (retorno)

End Function

Function idanalisis() As List(Of Integer)

    Dim retorno As New List(Of Integer)
    Dim rs As New ADODB.Recordset
    Dim sql As String

    sql = "SELECT DISTINCT ID_Analisis FROM Analisis_Clinico WHERE
EstadoBorradoA= FALSE AND CI_Paciente=" & _ci & ","

    Try

        rs = _cn.Execute(sql)

        Catch ex As Exception


```



```
Return (retorno)

End Try

retorno.Clear()

While Not rs.EOF

    retorno.Add(rs(0).Value)

    rs.MoveNext()

End While

rs.Close()

rs = Nothing

Return (retorno)

End Function

Function verifyci() As Integer

    Dim sql As String

    Dim rs As New ADODB.Recordset

    Dim retorno As Integer = 0

    If cn.State = 0 Then

        retorno = 1

        MsgBox("1")

    Else

        sql = "select an.Cl_Paciente, pa.EstadoBorradoP, an.EstadoBorradoA
from Persona as p JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente JOIN Atiende
as a ON pa.Cl_Paciente=a.Cl_Paciente JOIN Analisis_Clinico as an ON
a.Cl_Paciente=an.Cl_Paciente where an.EstadoBorradoA = FALSE AND
an.Cl_Paciente=" & _ci & ":""

        Try

            rs.Open(sql, _cn)

        Catch ex As Exception

            MsgBox(ex.ToString)

        Return (4)

    End Try

End Function
```



```
If rs.RecordCount = 0 Then
    retorno = 3
Else
    _ci = rs("CI_Paciente").Value
    _estadoborradop = rs("EstadoBorradoP").Value
    rs.Close()
End If
End If
Return (retorno)
End Function

Function verifyci2() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select pa.CI_Paciente, pa.EstadoBorradoP from Persona as p
JOIN Paciente as pa ON p.CI=pa.CI_Paciente where pa.CI_Paciente=" & _ci &
";
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            MsgBox(ex.ToString)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _ci = rs("CI_Paciente").Value
        End If
    End If
End Function
```



```
_estadoborradow = rs("EstadoBorradoP").Value
rs.Close()
End If
End If
Return (retorno)
End Function

Function Guardar(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql As String
    Dim sql2 As String
    Dim sql3 As String
    Dim sql4 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql = "SELECT MAX(Fecha_Atiende) AS Fecha_AtiendeC FROM
Atiende WHERE CI_Paciente=" & _ci & ";"
            Try
                rs.Open(sql, _cn)
            Catch ex As Exception
                Return (5)
            End Try
            _fechaatiende = rs("Fecha_AtiendeC").Value
            rs.Close()
            sql2 = "SELECT CI_Medico AS CI_MedicoC FROM Atiende WHERE
Fecha_Atiende="" & Format(_fechaatiende, "yyyy-MM-dd HH:mm:ss") & "" AND
CI_Paciente=" & _ci & ";"
            Try
```



```
rs.Open(sql2, _cn)
Catch ex As Exception
    Return (6)
End Try
_cimedico = rs("CI_MedicoC").Value
rs.Close()

sql3 = "insert into Analisis_Clinico(CI_Paciente, TipoAnalisis,
Fecha_Analisis, Resultado, CI_Medico, Fecha_Atiende) VALUES (" & _ci & ","
& _tipo & "," & Format(_fecha, "yyyy-MM-dd HH:mm:ss") & "," & _resultado &
"," & _cimedico & "," & Format(_fechaatiende, "yyyy-MM-dd HH:mm:ss") & ");"

Try
    _cn.Execute(sql3)
Catch ex As Exception
    Return (7)
End Try
sql4 = "insert into Posee(ID_Estado,CI_Paciente,Fechainicio_E)
VALUES ('3','" & _ci & "','" & Format(Now(), "yyyy-MM-dd HH:mm:ss") & "');"

Try
    _cn.Execute(sql4)
Catch ex As Exception
    Return (3)
End Try
Else
    sql = "update Analisis_Clinico set TipoAnalisis=" & _tipo & ",",
Fecha_Analisis=" & Format(_fecha, "yyyy-MM-dd HH:mm:ss") & ",",
Resultado=" & _resultado & " where ID_Analisis=" & _id & ";"

Try
    _cn.Execute(sql)
Catch ex As Exception
    Return (8)
End Try
```



```
End If
End If
Return (retorno)
End Function

Function Eliminar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Analisis_Clinico set EstadoBorradoA= TRUE where
ID_Analisis =" & _id & ";"
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
    End If
    Return (retorno)
End Function

Function estadop() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Paciente set EstadoBorradoP= False where CI_Paciente
=" & _ci & ","
    End If
End Function
```



```
Try
    _cn.Execute(sql)
Catch ex As Exception
    Return (2)
End Try
End If
Return (retorno)
End Function

Function getci() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select CI from Usuario_Rol WHERE Usuario=" & cipub & ";"
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            MsgBox(ex.ToString)
            Return (2)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _ci = rs("CI").Value
        End If
        rs.Close()
    End If
End Function
```



```
Return (retorno)
End Function

Function comprobarFecha() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select Fecha_Atiende from Atiende WHERE Cl_Paciente=" & _ci
& ""
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            MsgBox(ex.ToString)
            Return (2)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            End If
            rs.Close()
        End If
        Return (retorno)
    End Function

End Class
```

Diagnosticos

Public Class Diagnosticos



```
Protected _id As Integer  
Protected _ci As Integer  
Protected _tipo As String  
Protected _fecha As Date  
Protected _fechaatiende As Date  
Protected _nombreD As String  
Protected _descripcion As String  
Protected _estadoborradoP As Integer  
Protected _enfactual As String  
Protected _cimedico As Integer  
Protected _cn As ADODB.Connection  
'Metodos constructores  
Sub New()  
    _ci = 0  
    _cimedico = 0  
    _tipo = ""  
    _enfactual = ""  
    _fecha = New Date  
    _fechaatiende = New Date  
    _descripcion = ""  
    _estadoborradoP = 0  
    _nombreD = ""  
    _cn = New ADODB.Connection  
    _id = 0  
End Sub
```

```
Sub New(ByVal enfactual As String, ByVal estadoborradoP As Integer, ByVal  
ci As Integer, ByVal nombreD As String, ByVal cimedico As Integer, ByVal tipo  
As String, ByVal fechaatiende As Date, ByVal fecha As Date, ByVal descripcion  
As String, ByVal cn As ADODB.Connection, ByVal id As Integer)
```



```
_ci = ci  
_enfactual = enfactual  
_estadoborradop = estadoborradop  
_tipo = tipo  
_fecha = fecha  
_fechaatiende = fechaatiende  
_descripcion = descripcion  
_cn = cn  
_id = id  
_nombreD = nombreD  
_cimedico = cimedico
```

End Sub

'Get y set de las propiedades

Property enfactual As String

Get

 Return (_enfactual)

End Get

Set(ByVal value As String)

 _enfactual = value

End Set

End Property

Property estadoborradop As Integer

Get

 Return (_estadoborradop)

End Get

Set(ByVal value As Integer)

 _estadoborradop = value

End Set

End Property



```
Property ci As Integer
    Get
        Return (_ci)
    End Get
    Set(ByVal value As Integer)
        _ci = value
    End Set
End Property

Property cimedico As Integer
    Get
        Return (_cimedico)
    End Get
    Set(ByVal value As Integer)
        _cimedico = value
    End Set
End Property

Property nombreD As String
    Get
        Return (_nombreD)
    End Get
    Set(ByVal value As String)
        _nombreD = value
    End Set
End Property

Property tipo As String
    Get
        Return (_tipo)
    End Get
    Set(ByVal value As String)
```



```
_tipo = value
End Set
End Property
Property fecha As Date
Get
    Return (_fecha)
End Get
Set(ByVal value As Date)
    _fecha = value
End Set
End Property
Property descripcion As String
Get
    Return (_descripcion)
End Get
Set(ByVal value As String)
    _descripcion = value
End Set
End Property
Property cn As ADODB.Connection
Get
    Return (_cn)
End Get
Set(ByVal value As ADODB.Connection)
    _cn = value
End Set
End Property
Property id As Integer
Get
```



```
Return (_id)
End Get
Set(ById value As Integer)
    _id = value
End Set
End Property
Property fechaatiende As Date
    Get
        Return (_fechaatiende)
    End Get
    Set(ById value As Date)
        _fechaatiende = value
    End Set
End Property
Function Buscar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select * from Diagnostico as d JOIN Elabora as e ON
d.ID_Diag=e.ID_Diag where e.ID_Diag=" & _id & " AND e.CI_Paciente=" & _ci
& ";"
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (4)
        End Try
        If rs.RecordCount = 0 Then
```



```
retorno = 3

Else
    _ci = rs("CI_Paciente").Value
    _descripcion = rs("Descripcion_Diag").Value
    _fecha = rs("Fecha_Diag").Value
    _nombreD = rs("Tipo_Diag").Value
End If

rs.Close()

End If

Return (retorno)

End Function

Function Guardar(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql As String
    Dim sql2 As String
    Dim sql3 As String
    Dim sql4 As String
    Dim sql5 As String
    Dim sql6 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql = "insert into Diagnostico (Tipo_Diag,Descripcion_Diag) VALUES
            (" & _tipo & "," & _descripcion & ")"
            Try
                _cn.Execute(sql)
            Catch ex As Exception

```



```
        Return (4)

    End Try

    sql2 = "SELECT MAX(ID_Diag) AS ID_DiagC FROM Diagnostico;"

    Try
        rs.Open(sql2, _cn)

    Catch ex As Exception
        Return (4)

    End Try

    _id = rs("ID_DiagC").Value
    rs.Close()

    sql3 = "SELECT MAX(Fecha_Atiende) AS Fecha_AtiendeC FROM
Atiende WHERE CI_Paciente=" & _ci & ";"

    Try
        rs.Open(sql3, _cn)

    Catch ex As Exception
        Return (5)

    End Try

    _fechaatiende = rs("Fecha_AtiendeC").Value
    rs.Close()

    sql4 = "SELECT CI_Medico AS CI_MedicoC FROM Atiende WHERE
Fecha_Atiende="" & Format(_fechaatiende, "yyyy-MM-dd HH:mm:ss") & "" AND
CI_Paciente=" & _ci & ";"

    Try
        rs.Open(sql4, _cn)

    Catch ex As Exception
        Return (6)

    End Try

    _cimedico = rs("CI_MedicoC").Value
    rs.Close()
```



```
sql5 = "insert into  
Elabora(ID_Diag, Fecha_Diag, Fecha_Atiende, CI_Medico, CI_Paciente) VALUES  
(" & _id & "," & Format(_fecha, "yyyy-MM-dd HH:mm:ss") & "," &  
Format(_fechaatiende, "yyyy-MM-dd HH:mm:ss") & "," & _cimedico & "," & _ci  
& ");"
```

Try

```
_cn.Execute(sql5)
```

Catch ex As Exception

```
Return (2)
```

End Try

```
sql6 = "insert into Posee(ID_Estado, CI_Paciente, Fechalinicio_E)  
VALUES ('3','" & _ci & "','" & Format(Now(), "yyyy-MM-dd HH:mm:ss") & ");"
```

Try

```
_cn.Execute(sql6)
```

Catch ex As Exception

```
Return (3)
```

End Try

Else

```
sql = "update Diagnostico set Tipo_Diag=" & _tipo & ",  
Descripcion_Diag=" & _descripcion & " where ID_Diag=" & _id & ";"
```

Try

```
_cn.Execute(sql)
```

Catch ex As Exception

```
Return (7)
```

End Try

```
sql = "update Elabora set Fecha_Diag=" & Format(_fecha, "yyyy-MM-  
dd HH:mm:ss") & " where ID_Diag=" & _id & ";"
```

Try

```
_cn.Execute(sql)
```

Catch ex As Exception

```
Return (8)
```



```
End Try
End If
End If
Return (retorno)
End Function

Function iddiag() As List(Of Integer)
    Dim retorno As New List(Of Integer)
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "SELECT DISTINCT e.ID_Diag FROM Elabora as e JOIN
Diagnostico as d ON e.ID_Diag=d.ID_Diag WHERE d.EstadoBorradoD=False
AND e.CI_Paciente=" & _ci & ;"
    Try
        rs = _cn.Execute(sql)
    Catch ex As Exception
        Return (retorno)
    End Try
    retorno.Clear()
    While Not rs.EOF
        retorno.Add(rs(0).Value)
        rs.MoveNext()
    End While
    rs.Close()
    rs = Nothing
    Return (retorno)
End Function

Function Eliminar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
```



```
If cn.State = 0 Then
    retorno = 1
Else
    sql = "update Diagnostico set EstadoBorradoD= TRUE where ID_Diag
    =" & _id & ","
    Try
        _cn.Execute(sql)
    Catch ex As Exception
        Return (2)
    End Try
End If
Return (retorno)
End Function

Function setenfactual() As Integer
    Dim sql As String
    Dim sql2 As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "SELECT MAX(Fecha_Atiende) AS Fecha_AtiendeC FROM
Atiende WHERE CI_Paciente=" & _ci & ";"
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (5)
        End Try
        _fechaatiende = rs("Fecha_AtiendeC").Value
        rs.Close()
    End If
End Function
```



```
sql2 = "update Atiende set Enfermedad_Actual=" & _enfactual & ""
where Fecha_Atiende =" & Format(_fechaatiende, "yyyy-MM-dd HH:mm:ss") &
" AND CI_Paciente=" & _ci & ";"
```

Try

```
_cn.Execute(sql2)
```

Catch ex As Exception

```
Return (2)
```

End Try

End If

```
Return (retorno)
```

End Function

Function verifyci() As Integer

```
Dim sql As String
```

```
Dim rs As New ADODB.Recordset
```

```
Dim retorno As Integer = 0
```

```
If cn.State = 0 Then
```

```
    retorno = 1
```

Else

```
    sql = "select e.CI_Paciente, pa.EstadoBorradoP from Persona as p
JOIN Paciente as pa ON p.Cl=pa.CI_Paciente JOIN Atiende as a ON
pa.CI_Paciente=a.CI_Paciente JOIN Elabora as e ON
a.CI_Paciente=e.CI_Paciente where e.CI_Paciente=" & _ci & ";"
```

Try

```
rs.Open(sql, _cn)
```

Catch ex As Exception

```
Return (4)
```

End Try

```
If rs.RecordCount = 0 Then
```

```
    retorno = 3
```

Else

```
    _ci = rs("CI_Paciente").Value
```



```
_estadoborradoP = rs("EstadoBorradoP").Value
rs.Close()
End If
End If
Return (retorno)
End Function

Function verifyci2() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select pa.Cl_Paciente, pa.EstadoBorradoP from Persona as p
JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente where pa.Cl_Paciente=" & _ci &
"
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (4)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _ci = rs("Cl_Paciente").Value
            _estadoborradoP = rs("EstadoBorradoP").Value
            rs.Close()
        End If
    End If
    Return (retorno)
```



End Function

Function tipodiag() As List(Of String)

Dim retorno As New List(Of String)

Dim rs As New ADODB.Recordset

Dim sql As String

sql = "SELECT DISTINCT Tipo_Diag FROM Diagnostico;"

Try

 rs = _cn.Execute(sql)

 Catch ex As Exception

 Return (retorno)

 End Try

 retorno.Clear()

 While Not rs.EOF

 retorno.Add(rs(0).Value)

 rs.MoveNext()

 End While

 rs.Close()

 rs = Nothing

 Return (retorno)

End Function

Function estadop() As Integer

Dim sql As String

Dim rs As New ADODB.Recordset

Dim retorno As Integer = 0

If cn.State = 0 Then

 retorno = 1

Else

 sql = "update Paciente set EstadoBorradoP= False where CI_Paciente
 = " & _ci & ","



```
Try
    _cn.Execute(sql)
Catch ex As Exception
    Return (2)
End Try
End If
Return (retorno)
End Function

Function getci() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select CI from Usuario_Rol WHERE Usuario=" & cipub & ";"
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            MsgBox(ex.ToString)
            Return (2)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _ci = rs("CI").Value
        End If
        rs.Close()
    End If
End Function
```



```
Return (retorno)
End Function

Function comprobarFecha() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select Fecha_Atiende from Atiende WHERE Cl_Paciente=" & _ci
& "";;
        Try
            rs.Open(sql, _cn)
            Catch ex As Exception
                MsgBox(ex.ToString)
                Return (2)
            End Try
            If rs.RecordCount = 0 Then
                retorno = 3
            Else
                End If
                rs.Close()
            End If
            Return (retorno)
        End Function
    End Class
```

Entrevista Inicial

```
Imports System.Data.SqlClient
```



Public Class Entrevistainicial

Protected _ci As Integer
Protected _estadoc As String
Protected _gradoc As String
Protected _movilidad As String
Protected _actitud As String
Protected _estadon As String
Protected _gradoh As String
Protected _motivoconsulta As String
Protected _temperatura As Double
Protected _pulso As String
Protected _fechacb As Date
Protected _frecuenciac As String
Protected _frecuenciar As String
Protected _fiebre As Integer
Protected _dolor As Integer
Protected _idp As Integer
Protected _malestar As Integer
Protected _otro As String
Protected _enfermedad As String
Protected _tipoenfactual As String
Protected _tipoenfanterior As String
Protected _enfermedadf As String
Protected _cimedico As String
Protected _nombreacompañante As String
Protected _antecedentes As String
Protected _nombrepaciente As String
Protected _apellidopaciente As String
Protected _estadoborradop As Integer



Protected _fecha As Date
Protected _boca As Integer
Protected _brazo As Integer
Protected _cabeza As Integer
Protected _cuello As Integer
Protected _dedo As Integer
Protected _espalda As Integer
Protected _estomago As Integer
Protected _frente As Integer
Protected _garganta As Integer
Protected _hombro As Integer
Protected _mano As Integer
Protected _muslo As Integer
Protected _munieca As Integer
Protected _nariz As Integer
Protected _ojo As Integer
Protected _oreja As Integer
Protected _pantorrilla As Integer
Protected _pecho As Integer
Protected _pie As Integer
Protected _pierna As Integer
Protected _rodilla As Integer
Protected _tobillo As Integer
Protected _trasero As Integer
Protected _cn As ADODB.Connection

Sub New()

_ci = 0
_estadoborradop = 0

_cimedico = 0
_estadoc = ""
_gradoc = ""
_nombrepaciente = ""
_apellidopaciente = ""
_movilidad = ""
_enfermedad = ""
_enfermedadf = ""
_actitud = ""
_estadon = ""
_gradoh = ""
_motivoconsulta = ""
_temperatura = 0
_idp = 0
_pulso = ""
_frecuenciac = ""
_frecuenciar = ""
_tipoenfactual = ""
_tipoenfanterior = ""
_fiebre = 0
_dolor = 0
_malestar = 0
_nombreacompañante = ""
_otro = ""
_boca = 0
_brazo = 0
_cabeza = 0
_cuello = 0
_dedo = 0

```
_espalda = 0  
_estomago = 0  
_frente = 0  
_garganta = 0  
_hombro = 0  
_mano = 0  
_muslo = 0  
_munieca = 0  
_nariz = 0  
_ojo = 0  
_oreja = 0  
_pantorrilla = 0  
_pecho = 0  
_pie = 0  
_pierna = 0  
_rodilla = 0  
_tobillo = 0  
_trasero = 0  
_fechacb = New Date  
_fecha = New Date  
_cn = New ADODB.Connection
```

End Sub

Sub New(ByVal nombrepaciente As String, ByVal apellidopaciente As String, ByVal idp As Integer, ByVal trasero As Integer, ByVal tobillo As Integer, ByVal rodilla As Integer, ByVal pierna As Integer, ByVal nariz As Integer, ByVal ojo As Integer, ByVal oreja As Integer, ByVal pantorrilla As Integer, ByVal pecho As Integer, ByVal pie As Integer, ByVal hombro As Integer, ByVal mano As Integer, ByVal muslo As Integer, ByVal munieca As Integer, ByVal boca As Integer, ByVal brazo As Integer, ByVal cabeza As Integer, ByVal cuello As Integer, ByVal dedo As Integer, ByVal espalda As Integer, ByVal estomago As Integer, ByVal frente As Integer, ByVal garganta As Integer, ByVal estadoborradop As Integer, ByVal ci As Integer, ByVal fechacb As Date, ByVal

tipoenfactual As String, ByVal tipoenfanterior As String, ByVal cimedico As Integer, ByVal estadoc As String, ByVal antecedentes As String, ByVal nombreacompañante As String, ByVal enfermedad As String, ByVal enfermedadf As String, ByVal gradoc As String, ByVal movilidad As String, ByVal actitud As String, ByVal estadon As String, ByVal gradoh As Double, ByVal temperatura As Double, ByVal pulso As Double, ByVal frecuenciac As String, ByVal frecuenciar As String, ByVal fiebre As Integer, ByVal dolor As Integer, ByVal malestar As Integer, ByVal otro As String, ByVal motivoconsultatxt As String, ByVal fecha As Date, ByVal cn As ADODB.Connection)

_ci = ci
_idp = idp
_nombrepaciente = nombrepaciente
_apellidopaciente = apellidopaciente
_fechacb = fechacb
_tipoenfactual = tipoenfactual
_estadoborradow = estadoborradow
_tipoenfanterior = tipoenfanterior
_cimedico = cimedico
_estadoc = estadoc
_gradoc = gradoc
_movilidad = movilidad
_actitud = actitud
_estadon = estadon
_gradoh = gradoh
_motivoconsulta = motivoconsulta
_temperatura = temperatura
_pulso = pulso
_frecuenciac = frecuenciac
_frecuenciar = frecuenciar
_fiebre = fiebre
_dolor = dolor

_malestar = malestar
_enfermedad = enfermedad
_enfermedadf = enfermedadf
_antecedentes = antecedentes
_otro = otro
_fecha = fecha
_nombreacompañante = nombreacompañante
_boca = boca
_brazo = brazo
_cabeza = cabeza
_cuello = cuello
_dedo = dedo
_espalda = espalda
_estomago = estomago
_frente = frente
_garganta = garganta
_hombro = hombro
_mano = mano
_muslo = muslo
_munieca = munieca
_nariz = nariz
_ojo = ojo
_oreja = oreja
_pantorrilla = pantorrilla
_pecho = pecho
_pie = pie
_pierna = pierna
_rodilla = rodilla
_tobillo = tobillo



```
_trasero = trasero  
_cn = cn  
End Sub  
  
Property nombrepaciente As String  
    Get  
        Return (_nombrepaciente)  
    End Get  
    Set(ByVal value As String)  
        _nombrepaciente = value  
    End Set  
End Property  
  
Property apellidopaciente As String  
    Get  
        Return (_apellidopaciente)  
    End Get  
    Set(ByVal value As String)  
        _apellidopaciente = value  
    End Set  
End Property  
  
Property boca As Integer  
    Get  
        Return (_boca)  
    End Get  
    Set(ByVal value As Integer)  
        _boca = value  
    End Set  
End Property  
  
Property idp As Integer  
    Get
```



```
Return (_idp)
End Get
Set(ByVal value As Integer)
    _idp = value
End Set
End Property
Property brazo As Integer
    Get
        Return (_brazo)
    End Get
    Set(ByVal value As Integer)
        _brazo = value
    End Set
End Property
Property cabeza As Integer
    Get
        Return (_cabeza)
    End Get
    Set(ByVal value As Integer)
        _cabeza = value
    End Set
End Property
Property cuello As Integer
    Get
        Return (_cuello)
    End Get
    Set(ByVal value As Integer)
        _cuello = value
    End Set
End Property
```



End Property

Property dedo As Integer

Get

 Return (_dedo)

End Get

Set(ByVal value As Integer)

 _dedo = value

End Set

End Property

Property espalda As Integer

Get

 Return (_espalda)

End Get

Set(ByVal value As Integer)

 _espalda = value

End Set

End Property

Property estomago As Integer

Get

 Return (_estomago)

End Get

Set(ByVal value As Integer)

 _estomago = value

End Set

End Property

Property frente As Integer

Get

 Return (_frente)

End Get



Set(ByVal value As Integer)

 _frente = value

End Set

End Property

Property garganta As Integer

Get

 Return (_garganta)

End Get

Set(ByVal value As Integer)

 _garganta = value

End Set

End Property

Property hombro As Integer

Get

 Return (_hombro)

End Get

Set(ByVal value As Integer)

 _hombro = value

End Set

End Property

Property mano As Integer

Get

 Return (_mano)

End Get

Set(ByVal value As Integer)

 _mano = value

End Set

End Property

Property muslo As Integer



Get

 Return (_muslo)

End Get

Set(ByVal value As Integer)

 _muslo = value

End Set

End Property

Property munieca As Integer

Get

 Return (_munieca)

End Get

Set(ByVal value As Integer)

 _munieca = value

End Set

End Property

Property nariz As Integer

Get

 Return (_nariz)

End Get

Set(ByVal value As Integer)

 _nariz = value

End Set

End Property

Property ojo As Integer

Get

 Return (_ojo)

End Get

Set(ByVal value As Integer)

 _ojo = value



```
End Set  
End Property  
Property oreja As Integer  
    Get  
        Return (_oreja)  
    End Get  
    Set(ByVal value As Integer)  
        _oreja = value  
    End Set  
End Property  
Property pantorrilla As Integer  
    Get  
        Return (_pantorrilla)  
    End Get  
    Set(ByVal value As Integer)  
        _pantorrilla = value  
    End Set  
End Property  
Property pecho As Integer  
    Get  
        Return (_pecho)  
    End Get  
    Set(ByVal value As Integer)  
        _pecho = value  
    End Set  
End Property  
Property pie As Integer  
    Get  
        Return (_pie)
```



```
End Get  
Set(ByVal value As Integer)  
    _pie = value  
End Set  
End Property  
Property pierna As Integer  
    Get  
        Return (_pierna)  
    End Get  
    Set(ByVal value As Integer)  
        _pierna = value  
    End Set  
End Property  
Property rodilla As Integer  
    Get  
        Return (_rodilla)  
    End Get  
    Set(ByVal value As Integer)  
        _rodilla = value  
    End Set  
End Property  
Property tobillo As Integer  
    Get  
        Return (_tobillo)  
    End Get  
    Set(ByVal value As Integer)  
        _tobillo = value  
    End Set  
End Property
```



Property trasero As Integer

Get

 Return (_trasero)

End Get

Set(ByVal value As Integer)

 _ventas = value

End Set

End Property

Property estadoborradop As Integer

Get

 Return (_estadoborradop)

End Get

Set(ByVal value As Integer)

 _estadoborradop = value

End Set

End Property

Property ci As Integer

Get

 Return (_ci)

End Get

Set(ByVal value As Integer)

 _ci = value

End Set

End Property

Property fechacb As Date

Get

 Return (_fechacb)

End Get

Set(ByVal value As Date)



```
_fechacb = value
End Set
End Property
Property tipoenfactual As String
    Get
        Return (_tipoenfactual)
    End Get
    Set(ByVal value As String)
        _tipoenfactual = value
    End Set
End Property
Property tipoenfanterior As String
    Get
        Return (_tipoenfanterior)
    End Get
    Set(ByVal value As String)
        _tipoenfanterior = value
    End Set
End Property
Property cimedico As Integer
    Get
        Return (_cimedico)
    End Get
    Set(ByVal value As Integer)
        _cimedico = value
    End Set
End Property
Property estadoc As String
    Get
```



```
Return (_estadoc)
End Get
Set(ByVal value As String)
    _estadoc = value
End Set
End Property
Property antecedentes As String
    Get
        Return (_antecedentes)
    End Get
    Set(ByVal value As String)
        _antecedentes = value
    End Set
End Property
Property gradoc As String
    Get
        Return (_gradoc)
    End Get
    Set(ByVal value As String)
        _gradoc = value
    End Set
End Property
Property nombreacompañante As String
    Get
        Return (_nombreacompañante)
    End Get
    Set(ByVal value As String)
        _nombreacompañante = value
    End Set
End Property
```



End Property

Property movilidad As String

Get

 Return (_movilidad)

End Get

Set(ByVal value As String)

 _movilidad = value

End Set

End Property

Property enfermedad As String

Get

 Return (_enfermedad)

End Get

Set(ByVal value As String)

 _enfermedad = value

End Set

End Property

Property enfermedadf As String

Get

 Return (_enfermedadf)

End Get

Set(ByVal value As String)

 _enfermedadf = value

End Set

End Property

Property actitud As String

Get

 Return (_actitud)

End Get



Set(ByVal value As String)

 _actitud = value

End Set

End Property

Property estadon As String

Get

 Return (_estadon)

End Get

Set(ByVal value As String)

 _estadon = value

End Set

End Property

Property gradoh As String

Get

 Return (_gradoh)

End Get

Set(ByVal value As String)

 _gradoh = value

End Set

End Property

Property motivoconsulta As String

Get

 Return (_motivoconsulta)

End Get

Set(ByVal value As String)

 _motivoconsulta = value

End Set

End Property

Property temperatura As Double



Get

 Return (_temperatura)

End Get

Set(ByVal value As Double)

 _temperatura = value

End Set

End Property

Property pulso As String

Get

 Return (_pulso)

End Get

Set(ByVal value As String)

 _pulso = value

End Set

End Property

Property frecuenciac As String

Get

 Return (_frecuenciac)

End Get

Set(ByVal value As String)

 _frecuenciac = value

End Set

End Property

Property frecuenciar As String

Get

 Return (_frecuenciar)

End Get

Set(ByVal value As String)

 _frecuenciar = value



```
End Set  
End Property  
Property otro As String  
    Get  
        Return (_otro)  
    End Get  
    Set(ByVal value As String)  
        _otro = value  
    End Set  
End Property  
Property fiebre As Integer  
    Get  
        Return (_fiebre)  
    End Get  
    Set(ByVal value As Integer)  
        _fiebre = value  
    End Set  
End Property  
Property dolor As Integer  
    Get  
        Return (_dolor)  
    End Get  
    Set(ByVal value As Integer)  
        _dolor = value  
    End Set  
End Property  
Property malestar As Integer  
    Get  
        Return (_malestar)
```



```
End Get

Set(ByName value As Integer)
    _malestar = value

End Set

End Property

Property cn As ADODB.Connection
    Get
        Return (_cn)
    End Get

    Set(ByName value As ADODB.Connection)
        _cn = value

    End Set

End Property

Property fecha As Date
    Get
        Return (_fecha)
    End Get

    Set(ByName value As Date)
        _fecha = value

    End Set

End Property

Function Buscar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0

    If cn.State = 0 Then
        retorno = 1
    Else
```



```
sql = "select * from Atiende WHERE Fecha_Atiende="" &
Format(_fechacb, "yyyy-MM-dd HH:mm:ss") & ":";
```

Try

```
rs.Open(sql, _cn)
```

Catch ex As Exception

```
Return (2)
```

End Try

If rs.RecordCount = 0 Then

```
retorno = 3
```

Else

```
_actitud = rs("Actitud").Value
```

```
_antecedentes = rs("Antecedentes_Personales").Value
```

```
_dolor = rs("Dolor").Value
```

```
_enfermedad = rs("Enfermedad_Actual").Value
```

```
_enfermedadf = rs("Enfermedad_Familiar").Value
```

```
_estadoc = rs("Estado_De_Conciencia").Value
```

```
_estadon = rs("Estado_De_Nutricion").Value
```

```
_fiebre = rs("Fiebre").Value
```

```
_frecuenciac = rs("Frecuencia_Cardiaca").Value
```

```
_frecuenciar = rs("Frecuencia_Respiratoria").Value
```

```
_gradoc = rs("Grado_De_Colaboracion").Value
```

```
_gradoh = rs("Grado_De_Hidratacion").Value
```

```
_malestar = rs("Malestar").Value
```

```
_motivoconsulta = rs("Motivo").Value
```

```
_movilidad = rs("Movilidad").Value
```

```
_nombreacompanante = rs("Nombre_Acompaniante").Value
```

```
_otro = rs("Otro").Value
```

```
_pulso = rs("Pulso").Value
```

```
_temperatura = rs("Temperatura").Value
```



```
_tipoenfactual = rs("Tipo_EnfermedadActual").Value  
_tipoenfanterior = rs("Tipo_EnfermedadAnterior").Value  
_cimedico = rs("CI_Medico").Value  
rs.Close()  
End If  
End If  
Return (retorno)  
End Function  
  
Function Guardar(ByVal alta As Boolean) As Integer  
    Dim retorno As Integer = 0  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    Dim sql2 As String  
    If cn.State = 0 Then  
        retorno = 1  
    Else  
        If alta Then  
            sql = "insert into Atiende (CI_Paciente, Estado_De_Conciencia,  
Grado_De_Colaboracion, Movilidad, Grado_De_Hidratacion, Motivo,  
Temperatura, Frecuencia_Cardiaca, Frecuencia_Respiratoria, Otro, Actitud,  
Estado_de_Nutricion, Pulso, Enfermedad_Actual, Nombre_Acompanante,  
Enfermedad_Familiar, Antecedentes_Personales, Fiebre, Dolor, Malestar,  
Fecha_Atiende, CI_Medico, Tipo_EnfermedadActual,  
Tipo_EnfermedadAnterior) VALUES (" & _ci & "," & _estadoc & "," & _gradoc  
& "," & _movilidad & "," & _gradoh & "," & _motivoconsulta & "," &  
_temperatura & "," & _frecuenciac & "," & _frecuenciar & "," & _otro & "," &  
_actitud & "," & _estadon & "," & _pulso & "," & _enfermedad & "," &  
_nombreacompanante & "," & _enfermedadf & "," & _antecedentes & "," &  
_fiebre & "," & _dolor & "," & _malestar & "," & Format(_fecha, "yyyy-MM-dd  
HH:mm:ss") & "," & _cimedico & "," & _tipoenfactual & "," & _tipoenfanterior &  
");"  
Try  
    cn.Execute(sql)
```



```
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (2)

End Try

sql2 = "insert into Posee(ID_Estado,CI_Paciente,Fechalnicio_E)
VALUES ('2','" & _ci & "','" & Format(Now(), "yyyy-MM-dd HH:mm:ss") & ")"

Try
    _cn.Execute(sql2)
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (3)

End Try

End If

Return (retorno)

End Function

Function verifyci() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0

    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select pa.CI_Paciente, pa.EstadoBorradoP from Persona as p
JOIN Paciente as pa ON p.Cl=pa.CI_Paciente where pa.CI_Paciente=" & _ci &
"
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (4)
        End Try
    End If
End Function
```



```
End Try

If rs.RecordCount = 0 Then
    retorno = 3
Else
    _ci = rs("CI_Paciente").Value
    _estadoborradoP = rs("EstadoBorradoP").Value
    rs.Close()
End If

End If

Return (retorno)

End Function

Function verifyci2() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select pa.CI_Paciente, pa.EstadoBorradoP from Persona as p
JOIN Paciente as pa ON p.CI=pa.CI_Paciente JOIN Atiende as a ON
pa.CI_Paciente=a.CI_Paciente where a.CI_Paciente=" & _ci & ";"

        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (4)
        End Try

        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _ci = rs("CI_Paciente").Value
        End If
    End If
End Function
```



```
_estadoborradow = rs("EstadoBorradoP").Value
rs.Close()
End If
End If
Return (retorno)
End Function

Function buscardolor() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select * from PartesCuerpo as pc JOIN Incluye as i ON
pc.ID_Partes=i.ID_Partes JOIN Atiende as a ON i.CI_Paciente=a.CI_Paciente
where i.Fecha_Atiende="" & Format(_fecha, "yyyy-MM-dd HH:mm:ss") & ","
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (4)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _boca = rs("Boca").Value
            _brazo = rs("Brazo").Value
            _cabeza = rs("Cabeza").Value
            _cuello = rs("Cuello").Value
            _dedo = rs("Dedo").Value
            _espalda = rs("Espalda").Value
        End If
    End If
End Function
```



```
_estomago = rs("Estomago").Value  
_frente = rs("Frente").Value  
_garganta = rs("Garganta").Value  
_hombro = rs("Hombro").Value  
_mano = rs("Mano").Value  
_muslo = rs("Muslo").Value  
_munieca = rs("Munieca").Value  
_nariz = rs("Nariz").Value  
_ojo = rs("Ojo").Value  
_oreja = rs("Oreja").Value  
_pantorrilla = rs("Pantorrilla").Value  
_pecho = rs("Pecho").Value  
_pie = rs("Pie").Value  
_pierna = rs("Pierna").Value  
_rodilla = rs("Rodilla").Value  
_tobillo = rs("Tobillo").Value  
_trasero = rs("Trasero").Value  
rs.Close()  
End If  
End If  
Return (retorno)  
End Function  
Function actitudE() As List(Of String)  
    Dim retorno As New List(Of String)  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    sql = "SELECT DISTINCT Actitud FROM Atiende;"  
    Try  
        rs = _cn.Execute(sql)
```



```
Catch ex As Exception
    Return (retorno)
End Try
retorno.Clear()
While Not rs.EOF
    retorno.Add(rs(0).Value)
    rs.MoveNext()
End While
rs.Close()
rs = Nothing
Return (retorno)
End Function

Function fechafunc() As List(Of String)
    Dim retorno As New List(Of String)
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "SELECT Fecha_Atiende FROM Atiende WHERE CI_Paciente= " &
    _ci & ","
    Try
        rs = _cn.Execute(sql)
    Catch ex As Exception
        Return (retorno)
    End Try
    retorno.Clear()
    While Not rs.EOF
        retorno.Add(rs(0).Value)
        rs.MoveNext()
    End While
    rs.Close()
```



```
rs = Nothing  
Return (retorno)  
End Function  
  
Function estadonE() As List(Of String)  
    Dim retorno As New List(Of String)  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    sql = "SELECT DISTINCT Estado_De_Nutricion FROM Atiende;"  
    Try  
        rs = _cn.Execute(sql)  
        Catch ex As Exception  
            Return (retorno)  
        End Try  
        retorno.Clear()  
        While Not rs.EOF  
            retorno.Add(rs(0).Value)  
            rs.MoveNext()  
        End While  
        rs.Close()  
        rs = Nothing  
        Return (retorno)  
    End Function  
  
Function pulsoE() As List(Of String)  
    Dim retorno As New List(Of String)  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    sql = "SELECT DISTINCT Pulso FROM Atiende;"  
    Try  
        rs = _cn.Execute(sql)
```



```
Catch ex As Exception
    Return (retorno)
End Try
retorno.Clear()
While Not rs.EOF
    retorno.Add(rs(0).Value)
    rs.MoveNext()
End While
rs.Close()
rs = Nothing
Return (retorno)
End Function

Function estadop() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Paciente set EstadoBorradoP= False where CI_Paciente
        =" & _ci & ";"
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
    End If
    Return (retorno)
End Function
```



Function Guardar2(ByVal alta As Boolean) As Integer

```
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql As String
    Dim sql2 As String
    Dim sql3 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql = "insert into PartesCuerpo (Cabeza, Cuello, Brazo, Mano, Muslo,
Pantorrilla, Oreja, Hombro, Espalda, Trasero, Tobillo, Ojo, Nariz, Boca,
Munieca, Dedo, Pierna, Frente, Garganta, Pecho, Estomago, Rodilla, Pie)
VALUES (" & _cabeza & "," & _cuello & "," & _brazo & "," & _mano & "," &
_muslo & "," & _pantorrilla & "," & _oreja & "," & _hombro & "," & _espalda &
"," & _trasero & "," & _tobillo & "," & _oj o & "," & _nariz & "," & _boca & ","
& _munieca & "," & _dedo & "," & _pierna & "," & _frente & "," & _garganta &
"," & _pecho & "," & _estomago & "," & _rodilla & "," & _pie & ")");
        Try
            cn.Execute(sql)
        Catch ex As Exception
            MsgBox(ex.ToString)
        Return (2)
    End Try
    sql2 = "SELECT MAX(ID_Partes) AS ID_PartesC FROM
PartesCuerpo;"
```

Try

```
        rs.Open(sql2, _cn)
    Catch ex As Exception
        MsgBox(ex.ToString)
    Return (3)
End Try
```



```
_idp = rs("ID_PartesC").Value  
sql3 = "insert into Incluye (ID_Partes, CI_Paciente, CI_Medico,  
Fecha_Atiende) VALUES (" & _idp & "," & _ci & "," & _cimedico & "," &  
Format(_fecha, "yyyy-MM-dd HH:mm:ss") & ";"  
  
Try  
    cn.Execute(sql3)  
    Catch ex As Exception  
        MsgBox(ex.ToString)  
        Return (2)  
    End Try  
End If  
End If  
Return (retorno)  
End Function  
  
Function tabladata1() As DataTable  
    Dim retorno As New DataTable  
    Dim rs As New ADODB.Recordset  
    Dim da As New OleDb.OleDbDataAdapter  
    Dim sql As String  
    sql = "select Enfermedad_Actual as 'Enfermedad' from  
vista_SintomasAtiendeSD where Dolor =" & _dolor & " AND Fiebre =" & _fiebre  
& " AND Malestar =" & _malestar & " AND Otro=" & _otro & ";"  
    Try  
        rs.Open(sql, _cn)  
        Catch ex As Exception  
            Return (retorno)  
        End Try  
        da.Fill(retorno, rs)  
        Return (retorno)  
    End Function
```



Function tabladata2() As DataTable

 Dim retorno As New DataTable

 Dim rs As New ADODB.Recordset

 Dim da As New OleDb.OleDbDataAdapter

 Dim sql As String

 sql = "select Enfermedad_Actual as 'Enfermedad' from vista_SintomasAtiendeCD where Dolor =" & _dolor & " AND Fiebre =" & _fiebre & " AND Malestar =" & _malestar & " AND Otro=" & _otro & " AND Cabeza =" & _cabeza & " AND Cuello =" & _cuello & " AND Brazo=" & _brazo & " AND Mano =" & _mano & " AND Muslo =" & _muslo & " AND Pantorrilla=" & _pantorrilla & " AND Oreja =" & _oreja & " AND Hombro =" & _hombro & " AND Espalda=" & _espalda & " AND Trasero =" & _trasero & " AND Tobillo =" & _tobillo & " AND Ojo=" & _oj & " AND Nariz =" & _nariz & " AND Boca =" & _boca & " AND Munieca=" & _munieca & " AND Dedo =" & _dedo & " AND Pierna =" & _pierna & " AND Frente=" & _frente & " AND Garganta =" & _garganta & " AND Pecho =" & _pecho & " AND Estomago=" & _estomago & " AND Rodilla =" & _rodilla & " AND Pie=" & _pie & ";"

 Try

 rs.Open(sql, _cn)

 Catch ex As Exception

 Return (retorno)

 End Try

 da.Fill(retorno, rs)

 Return (retorno)

End Function

Function nomape() As Integer

 Dim sql As String

 Dim rs As New ADODB.Recordset

 Dim retorno As Integer = 0

 If cn.State = 0 Then

 retorno = 1

 Else



```
sql = "select Primer_Nombre, Primer_Apellido from Persona where CI=" & _ci & ";"
```

```
Try
```

```
    rs.Open(sql, _cn)
```

```
    Catch ex As Exception
```

```
        Return (4)
```

```
    End Try
```

```
    If rs.RecordCount = 0 Then
```

```
        retorno = 3
```

```
    Else
```

```
        _nombrepaciente = rs("Primer_Nombre").Value
```

```
        _apellidopaciente = rs("Primer_Apellido").Value
```

```
        rs.Close()
```

```
    End If
```

```
End If
```

```
Return (retorno)
```

```
End Function
```

```
Function confirmarenf() As Integer
```

```
    Dim sql As String
```

```
    Dim rs As New ADODB.Recordset
```

```
    Dim retorno As Integer = 0
```

```
    If cn.State = 0 Then
```

```
        retorno = 1
```

```
    Else
```

```
        sql = "update Atiende set Enfermedad_Actual=" & _enfermedad & "" & "where CI_Paciente=" & _ci & " AND Fecha_Atiende=" & Format(_fecha, "yyyy-MM-dd HH:mm:ss") & ";"
```

```
    Try
```

```
        _cn.Execute(sql)
```

```
    Catch ex As Exception
```



```
    Return (2)  
End Try  
End If  
Return (retorno)  
End Function  
End Class
```



Formularios

```
Imports System.Data.SqlClient
```

```
Public Class Formularios
```

```
    Protected _idf As Integer
```

```
    Protected _idp As Integer
```

```
    Protected _tipo As String
```

```
    Protected _ci As Integer
```

```
    Protected _cimedico As Integer
```

```
    Protected _fecha As Date
```

```
    Protected _idr As Integer
```

```
    Protected _pregunta As String
```

```
    Protected _respuesta As String
```

```
    Protected _nuevapregunta As String
```

```
    Protected _estadoborradop As Integer
```

```
    Protected _cn As ADODB.Connection
```

```
'Metodos constructores
```

```
Sub New()
```

```
    _tipo = ""
```

```
    _respuesta = ""
```

```
    _estadoborradop = 0
```

```
    _pregunta = ""
```

```
    _nuevapregunta = ""
```

```
    _fecha = New Date
```

```
    _idf = 0
```

```
    _idr = 0
```

```
    _ci = 0
```

```
    _cimedico = 0
```



```
_idp = 0
```

```
_cn = New ADODB.Connection
```

```
End Sub
```

```
Sub New(ByVal cimedico As Integer, ByVal idr As Integer, ByVal respuesta  
As String, ByVal fecha As Date, ByVal estadoborradop As Integer, ByVal ci As  
Integer, ByVal tipo As String, ByVal pregunta As String, ByVal nuevapregunta  
As String, ByVal cn As ADODB.Connection, ByVal idf As Integer, ByVal idp As  
Integer)
```

```
    _idf = idf
```

```
    _idr = idr
```

```
    _idp = idp
```

```
    _fecha = fecha
```

```
    _cimedico = cimedico
```

```
    _tipo = tipo
```

```
    _respuesta = respuesta
```

```
    _estadoborradop = estadoborradop
```

```
    _pregunta = pregunta
```

```
    _nuevapregunta = nuevapregunta
```

```
    _cn = cn
```

```
    _ci = ci
```

```
End Sub
```

```
'Get y set de las propiedades
```

```
Property cimedico As Integer
```

```
    Get
```

```
        Return (_cimedico)
```

```
    End Get
```

```
    Set(ByVal value As Integer)
```

```
        _cimedico = value
```

```
    End Set
```



End Property

Property idr As Integer

Get

 Return (_idr)

End Get

Set(ByVal value As Integer)

 _idr = value

End Set

End Property

Property respuesta As String

Get

 Return (_respuesta)

End Get

Set(ByVal value As String)

 _respuesta = value

End Set

End Property

Property fecha As Date

Get

 Return (_fecha)

End Get

Set(ByVal value As Date)

 _fecha = value

End Set

End Property

Property tipo As String

Get

 Return (_tipo)

End Get



```
Set(ByVal value As String)
```

```
    _tipo = value
```

```
End Set
```

```
End Property
```

```
Property pregunta As String
```

```
Get
```

```
    Return (_pregunta)
```

```
End Get
```

```
Set(ByVal value As String)
```

```
    _pregunta = value
```

```
End Set
```

```
End Property
```

```
Property nuevapregunta As String
```

```
Get
```

```
    Return (_nuevapregunta)
```

```
End Get
```

```
Set(ByVal value As String)
```

```
    _nuevapregunta = value
```

```
End Set
```

```
End Property
```

```
Property estadoborradop As Integer
```

```
Get
```

```
    Return (_estadoborradop)
```

```
End Get
```

```
Set(ByVal value As Integer)
```

```
    _estadoborradop = value
```

```
End Set
```

```
End Property
```

```
Property ci As Integer
```



Get

 Return (_ci)

End Get

Set(ByVal value As Integer)

 _ci = value

End Set

End Property

Property idf As Integer

Get

 Return (_idf)

End Get

Set(ByVal value As Integer)

 _idf = value

End Set

End Property

Property idp As Integer

Get

 Return (_idp)

End Get

Set(ByVal value As Integer)

 _idp = value

End Set

End Property

Property cn As ADODB.Connection

Get

 Return (_cn)

End Get

Set(ByVal value As ADODB.Connection)

 _cn = value



```
End Set

End Property

Function Guardar(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql = "INSERT INTO Formulario (Tipo_Form) VALUES (" & _tipo &
        ");"
            Try
                _cn.Execute(sql)
            Catch ex As Exception
                Return (5)
            End Try
        End If
        Return (retorno)
    End Function

Function tipoform() As List(Of String)
    Dim retorno As New List(Of String)
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "SELECT DISTINCT Tipo_Form FROM Formulario WHERE
EstadoBorradoF= FALSE;"
    Try
        rs = _cn.Execute(sql)
    Catch ex As Exception

```



```
Return (retorno)

End Try

retorno.Clear()

While Not rs.EOF

    retorno.Add(rs(0).Value)

    rs.MoveNext()

End While

rs.Close()

rs = Nothing

Return (retorno)

End Function

Function verifyci() As Integer

    Dim sql As String

    Dim rs As New ADODB.Recordset

    Dim retorno As Integer = 0

    If cn.State = 0 Then

        retorno = 1

    Else

        sql = "select r.Cl_Paciente, pa.EstadoBorradoP from Persona as p JOIN
        Paciente as pa ON p.Cl=pa.Cl_Paciente JOIN Atiende as a ON
        pa.Cl_Paciente=a.Cl_Paciente JOIN Responde as r ON
        a.Cl_Paciente=r.Cl_Paciente where r.Cl_Paciente=" & _ci & ","

        Try

            rs.Open(sql, _cn)

        Catch ex As Exception

            Return (4)

        End Try

        If rs.RecordCount = 0 Then

            retorno = 3

        Else


```



```
_ci = rs("CI_Paciente").Value
_estadoborradop = rs("EstadoBorradoP").Value
rs.Close()

End If

End If

Return (retorno)

End Function

Function estadop() As Integer

Dim sql As String
Dim rs As New ADODB.Recordset
Dim retorno As Integer = 0
If cn.State = 0 Then
    retorno = 1
Else
    sql = "update Paciente set EstadoBorradoP= False where CI_Paciente
    =" & _ci & ";"
    Try
        _cn.Execute(sql)
    Catch ex As Exception
        Return (2)
    End Try
End If

Return (retorno)

End Function

Function fechafunc() As List(Of String)

Dim retorno As New List(Of String)
Dim rs As New ADODB.Recordset
Dim sql As String
sql = "SELECT Fecha_Atiende FROM Atiende WHERE CI_Paciente= " &
_ci & ","

```



Try

 rs = _cn.Execute(sql)

 Catch ex As Exception

 Return (retorno)

 End Try

 retorno.Clear()

 While Not rs.EOF

 retorno.Add(rs(0).Value)

 rs.MoveNext()

 End While

 rs.Close()

 rs = Nothing

 Return (retorno)

End Function

Function cifunc() As List(Of String)

 Dim retorno As New List(Of String)

 Dim rs As New ADODB.Recordset

 Dim sql As String

 sql = "SELECT CI_Medico FROM Atiende WHERE Fecha_Atiende= " &
Format(_fecha, "yyyy-MM-dd HH:mm:ss") & ":"

Try

 rs = _cn.Execute(sql)

 Catch ex As Exception

 Return (retorno)

 End Try

 retorno.Clear()

 While Not rs.EOF

 retorno.Add(rs(0).Value)

 rs.MoveNext()



```
End While

rs.Close()
rs = Nothing

Return (retorno)

End Function

Function verifyci2() As Integer

    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select pa.Cl_Paciente, pa.EstadoBorradoP from Persona as p
JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente where pa.Cl_Paciente=" & _ci &
";
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (4)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _ci = rs("Cl_Paciente").Value
            _estadoborradop = rs("EstadoBorradoP").Value
            rs.Close()
        End If
    End If
    Return (retorno)

End Function
```



```
Function tipoform2() As List(Of String)
    Dim retorno As New List(Of String)
    Dim rs As New ADODB.Recordset
    Dim sql As String

    sql = "SELECT DISTINCT f.Tipo_Form FROM Formulario as f JOIN
Contiene as c ON f.ID_Form=c.ID_Form JOIN Tiene as t ON
c.ID_Form=t.ID_Form JOIN Respuesta as r ON t.ID_Resp=r.ID_Resp JOIN
Responde as re ON r.ID_Resp=re.ID_Resp WHERE f.EstadoBorradoF=False
AND re.CI_Paciente=" & _ci & ":";

    Try
        rs = _cn.Execute(sql)
    Catch ex As Exception
        Return (retorno)
    End Try
    retorno.Clear()
    While Not rs.EOF
        retorno.Add(rs(0).Value)
        rs.MoveNext()
    End While
    rs.Close()
    rs = Nothing
    Return (retorno)
End Function

Function idform() As List(Of Integer)
    Dim retorno As New List(Of Integer)
    Dim rs As New ADODB.Recordset
    Dim sql As String

    sql = "SELECT DISTINCT ID_Form FROM Formulario WHERE
EstadoBorradoF=False AND Tipo_Form="" & _tipo & ":";

    Try
        rs = _cn.Execute(sql)
```



Catch ex As Exception

 Return (retorno)

End Try

 retorno.Clear()

 While Not rs.EOF

 retorno.Add(rs(0).Value)

 rs.MoveNext()

 End While

 rs.Close()

 rs = Nothing

 Return (retorno)

End Function

Function Guardar1(ByVal alta As Boolean) As Integer

 Dim retorno As Integer = 0

 Dim rs As New ADODB.Recordset

 Dim sql As String

 If cn.State = 0 Then

 retorno = 1

 Else

 If alta Then

 sql = "INSERT INTO Pregunta (TextoPreg) VALUES ("" & _pregunta
& "");"

 Try

 _cn.Execute(sql)

 Catch ex As Exception

 Return (5)

 End Try

 Else

 End If



```
End If  
Return (retorno)  
End Function  
  
Function Guardar2(ByVal alta As Boolean) As Integer  
    Dim retorno As Integer = 0  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    Dim sql2 As String  
    If cn.State = 0 Then  
        retorno = 1  
    Else  
        If alta Then  
            sql = "SELECT ID_Preg FROM Pregunta where TextoPreg="" &  
            _pregunta & "";"  
            Try  
                rs.Open(sql, _cn)  
            Catch ex As Exception  
                Return (5)  
            End Try  
            _idp = rs("ID_Preg").Value  
            rs.Close()  
            sql2 = "INSERT INTO Contiene (ID_Form,ID_Preg) VALUES (" & _idf  
            & "," & _idp & ")";  
            Try  
                _cn.Execute(sql2)  
            Catch ex As Exception  
                Return (5)  
            End Try  
        Else  
    End If
```



```
sql = "SELECT ID_Preg FROM Pregunta where TextoPreg="" &
_pregunta & "";"  
Try  
    rs.Open(sql, _cn)  
Catch ex As Exception  
    Return (5)  
End Try  
_idp = rs("ID_Preg").Value  
rs.Close()  
sql2 = "update Pregunta set TextoPreg="" & _nuevapregunta & ""
where ID_Preg="" & _idp & "";"  
Try  
    _cn.Execute(sql2)  
Catch ex As Exception  
    Return (5)  
End Try  
End If  
End If  
Return (retorno)  
End Function  
Function preg() As List(Of String)  
    Dim retorno As New List(Of String)  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    sql = "SELECT DISTINCT TextoPreg FROM Pregunta;"  
    Try  
        rs = _cn.Execute(sql)  
    Catch ex As Exception  
        Return (retorno)  
    End Try
```



```
retorno.Clear()
While Not rs.EOF
    retorno.Add(rs(0).Value)
    rs.MoveNext()
End While
rs.Close()
rs = Nothing
Return (retorno)
End Function

Function preg2() As List(Of String)
    Dim retorno As New List(Of String)
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "SELECT DISTINCT p.TextoPreg FROM Pregunta as p JOIN
Contiene as c ON p.ID_Preg=c.ID_Preg WHERE c.ID_Form=" & _idf & ","
    Try
        rs = _cn.Execute(sql)
    Catch ex As Exception
        Return (retorno)
    End Try
    retorno.Clear()
    While Not rs.EOF
        retorno.Add(rs(0).Value)
        rs.MoveNext()
    End While
    rs.Close()
    rs = Nothing
    Return (retorno)
End Function
```



```
Function Eliminar() As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql As String
    Dim sql2 As String
    Dim sql3 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "SELECT ID_Preg FROM Pregunta where TextoPreg="" &
        _pregunta & """;
        Try
            rs.Open(sql, _cn)
            Catch ex As Exception
                MsgBox(ex.ToString)
                Return (5)
            End Try
            _idp = rs("ID_Preg").Value
            rs.Close()
            sql2 = "delete from Tiene where ID_Preg=" & _idp & ";"
            Try
                _cn.Execute(sql2)
            Catch ex As Exception
                MsgBox(ex.ToString)
                Return (2)
            End Try
            sql2 = "delete from Contiene where ID_Preg=" & _idp & ";"
            Try
                _cn.Execute(sql2)
```



```
Catch ex As Exception
    MsgBox("a")
    MsgBox(ex.ToString)
    Return (2)

End Try

sql3 = "delete from Pregunta where ID_Preg=" & _idp & ","
Try
    _cn.Execute(sql3)
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (2)

End Try

End If

Return (retorno)

End Function

Function Eliminar1() As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql As String
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Formulario set EstadoBorradoF= TRUE where ID_Form="
        & _idf & ","
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (3)
        End Try
    End If
End Function
```



```
End If  
Return (retorno)  
End Function  
  
Function tablaform() As DataTable  
    Dim retorno As New DataTable  
    Dim rs As New ADODB.Recordset  
    Dim da As New OleDb.OleDbDataAdapter  
    Dim sql As String  
    sql = "select p.TextoPreg as 'Pregunta', r.TextoResp as 'Respuesta' from Pregunta as p join Tiene as t on p.ID_Preg=t.ID_Preg join Respuesta as r ON t.ID_Resp=r.ID_Resp where t.ID_Form=" & _idf & ";"  
    Try  
        rs.Open(sql, _cn)  
    Catch ex As Exception  
        Return (retorno)  
    End Try  
    da.Fill(retorno, rs)  
    Return (retorno)  
End Function  
  
Function Guardar3(ByVal alta As Boolean) As Integer  
    Dim retorno As Integer = 0  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    Dim sql2 As String  
    Dim sql3 As String  
    Dim sql4 As String  
    Dim sql5 As String  
    If cn.State = 0 Then  
        retorno = 1  
    Else
```



```
If alta Then
    sql = "INSERT INTO Respuesta (TextoResp) VALUES (" &
    _respuesta & ")";
    Try
        _cn.Execute(sql)
    Catch ex As Exception
        Return (2)
    End Try
    sql2 = "SELECT MAX(ID_Resp) AS ID_RespC FROM Respuesta;" 
    Try
        rs.Open(sql2, _cn)
    Catch ex As Exception
        Return (3)
    End Try
    _idr = rs("ID_RespC").Value
    rs.Close()
    sql3 = "INSERT INTO Responde
(CI_Medico,CI_Paciente,Fecha_Atiende,ID_Resp) VALUES (" & _cimedico &
"," & _ci & "," & Format(_fecha, "yyyy-MM-dd HH:mm:ss") & "," & _idr & ");"
    Try
        _cn.Execute(sql3)
    Catch ex As Exception
        Return (4)
    End Try
    sql4 = "SELECT ID_Preg FROM Pregunta where TextoPreg="" &
_pregunta & "";;
    Try
        rs.Open(sql4, _cn)
    Catch ex As Exception
        Return (5)
```



```
End Try

_idp = rs("ID_Preg").Value
rs.Close()

sql5 = "INSERT INTO Tiene (ID_Form,ID_Preg,ID_Resp,FechaR)
VALUES (" & _idf & "," & _idp & "," & _idr & "," & Format(Now, "yyyy-MM-dd
HH:mm:ss") & ")"

Try
    _cn.Execute(sql5)
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (6)
End Try

End If

Return (retorno)
End Function

Function comprobarFecha() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select Fecha_Atiende from Atiende WHERE CI_Paciente=" & _ci
        & "";;
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            MsgBox(ex.ToString)
            Return (2)
        End Try
    End If
End Function
```



```
End Try  
If rs.RecordCount = 0 Then  
    retorno = 3  
Else  
End If  
rs.Close()  
End If  
Return (retorno)  
End Function  
End Class
```

Pacientes

Public Class Pacientes

```
Protected _ci As Integer  
Protected _pnombre As String  
Protected _snombre As String  
Protected _usuario As String  
Protected _papellido As String  
Protected _sapellido As String  
Protected _estadocivil As String  
Protected _sexo As Char  
Protected _fechanac As Date  
Protected _edad As Integer  
Protected _telefono As String  
Protected _calle As String  
Protected _correo As String  
Protected _esquina As String  
Protected _nropuerta As String  
Protected _contrasena As String  
Protected _estadoborradop As Integer  
Protected _cn As ADODB.Connection
```

'Metodos constructores

Sub New()

```
_ci = 0  
_usuario = ""  
_pnombre = ""  
_snombre = ""  
_estadoborradop = 0  
_papellido = ""  
_sapellido = ""
```



```
_estadocivil = ""  
  
_sexo = ""  
  
_contrasena = ""  
  
_edad = 0  
  
_correo = ""  
  
_fechanac = New Date  
  
_telefono = ""  
  
_calle = ""  
  
_esquina = ""  
  
_nropuerta = ""  
  
_cn = New ADODB.Connection
```

End Sub

```
Sub New(ByVal usuario As String, ByVal contrasena As String, ByVal correo  
As String, ByVal estadoborradoP As Integer, ByVal ci As Integer, ByVal  
pnombre As String, ByVal snombre As String, ByVal papellido As String, ByVal  
sapellido As String, ByVal estadocivil As String, ByVal sexo As Char, ByVal  
ocupacion As String, ByVal edad As Integer, ByVal fechanac As Date, ByVal  
telefono As String, ByVal calle As String, ByVal esquina As String, ByVal  
nropuerta As String, ByVal cn As ADODB.Connection)
```

```
    _ci = ci  
  
    _usuario = usuario  
  
    _correo = correo  
  
    _contrasena = contrasena  
  
    _pnombre = pnombre  
  
    _snombre = snombre  
  
    _papellido = papellido  
  
    _sapellido = sapellido  
  
    _estadoborradoP = estadoborradoP  
  
    _estadocivil = estadocivil  
  
    _sexo = sexo  
  
    _edad = edad
```



```
_fechanac = fechanac  
_telefono = telefono  
_calle = calle  
_esquina = esquina  
_nropuerta = nropuerta  
_cn = cn  
End Sub  
'Get y set de las propiedades  
Property usuario As String  
    Get  
        Return (_usuario)  
    End Get  
    Set(ByVal value As String)  
        _usuario = value  
    End Set  
End Property  
Property contrasena As String  
    Get  
        Return (_contrasena)  
    End Get  
    Set(ByVal value As String)  
        _contrasena = value  
    End Set  
End Property  
Property correo As String  
    Get  
        Return (_correo)  
    End Get  
    Set(ByVal value As String)
```



```
_correo = value  
End Set  
End Property  
Property estadoborradoP As Integer  
Get  
    Return (_estadoborradoP)  
End Get  
Set(ByVal value As Integer)  
    _estadoborradoP = value  
End Set  
End Property  
Property ci As Integer  
Get  
    Return (_ci)  
End Get  
Set(ByVal value As Integer)  
    _ci = value  
End Set  
End Property  
Property pnombre As String  
Get  
    Return (_pnombre)  
End Get  
Set(ByVal value As String)  
    _pnombre = value  
End Set  
End Property  
Property snombre As String  
Get
```



```
Return (_snombre)
End Get
Set(ByVal value As String)
    _snombre = value
End Set
End Property
Property papellido As String
    Get
        Return (_papellido)
    End Get
    Set(ByVal value As String)
        _papellido = value
    End Set
End Property
Property sapellido As String
    Get
        Return (_sapellido)
    End Get
    Set(ByVal value As String)
        _sapellido = value
    End Set
End Property
Property estadocivil As String
    Get
        Return (_estadocivil)
    End Get
    Set(ByVal value As String)
        _estadocivil = value
    End Set
End Property
```



End Property

Property sexo As Char

Get

 Return (_sexo)

End Get

Set(ByVal value As Char)

 _sexo = value

End Set

End Property

Property fechanac As Date

Get

 Return (_fechanac)

End Get

Set(ByVal value As Date)

 _fechanac = value

End Set

End Property

Property edad As Integer

Get

 Return (_edad)

End Get

Set(ByVal value As Integer)

 _edad = value

End Set

End Property

Property telefono As String

Get

 Return (_telefono)

End Get



Set(ByVal value As String)

 _telefono = value

End Set

End Property

Property calle As String

Get

 Return (_calle)

End Get

Set(ByVal value As String)

 _calle = value

End Set

End Property

Property esquina As String

Get

 Return (_esquina)

End Get

Set(ByVal value As String)

 _esquina = value

End Set

End Property

Property nropuerta As String

Get

 Return (_nropuerta)

End Get

Set(ByVal value As String)

 _nropuerta = value

End Set

End Property

Property cn As ADODB.Connection



```
Get
    Return (_cn)
End Get
Set(ByVal value As ADODB.Connection)
    _cn = value
End Set
End Property
Function Buscar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select * from Persona as p JOIN Paciente as pa ON
p.Cl=pa.Cl_Paciente WHERE pa.Cl_Paciente=" & _ci & ","
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            MsgBox(ex.ToString)
            Return (2)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _pnombre = rs("Primer_Nombre").Value
            _estadoborradop = rs("EstadoBorradoP").Value
            _snombre = rs("Segundo_Nombre").Value
            _papellido = rs("Primer_Apellido").Value
        End If
    End If
End Function
```



```
_sapellido = rs("Segundo_Apellido").Value  
_estadocivil = rs("Estado_Civil").Value  
_sexo = rs("Sexo").Value  
_fechanac = rs("Fecha_Nac").Value  
_calle = rs("Calle").Value  
_esquina = rs("Esquina").Value  
_nropuerta = rs("Numero").Value  
_correo = rs("Correo").Value  
rs.Close()  
  
sql = "select TIMESTAMPDIFF(YEAR, Fecha_Nac, curdate()) as Edad from Persona where CI=" & _ci & ":"  
  
Try  
    rs.Open(sql, _cn)  
    _edad = rs("Edad").Value  
  
Catch ex As Exception  
    Return (5)  
  
End Try  
  
End If  
rs.Close()  
End If  
  
Return (retorno)  
  
End Function  
  
Function verifyci() As Integer  
    Dim sql As String  
    Dim rs As New ADODB.Recordset  
    Dim retorno As Integer = 0  
    If cn.State = 0 Then  
        retorno = 1  
    Else
```



```
sql = "select CI from Personal_Medico WHERE CI=" & _ci & ":"  
Try  
    rs.Open(sql, _cn)  
Catch ex As Exception  
    MsgBox(ex.ToString)  
    Return (2)  
End Try  
If rs.RecordCount = 0 Then  
    retorno = 3  
Else  
    _ci = rs("CI").Value  
End If  
rs.Close()  
End If  
Return (retorno)  
End Function  
Function patelefono() As List(Of String)  
    Dim retorno As New List(Of String)  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    sql = "select Telefono from Persona_Telefono where CI=" & _ci & ":"  
    Try  
        rs = _cn.Execute(sql)  
    Catch ex As Exception  
        Return (retorno)  
    End Try  
    retorno.Clear()  
    While Not rs.EOF  
        retorno.Add(rs(0).Value)  
    End While
```



```
rs.MoveNext()  
End While  
rs.Close()  
rs = Nothing  
Return (retorno)  
End Function  
  
Function Guardar(ByVal alta As Boolean) As Integer  
    Dim retorno As Integer = 0  
    Dim rs As New ADODB.Recordset  
    Dim sql As String  
    Dim sql3 As String  
    Dim sql4 As String  
    If cn.State = 0 Then  
        retorno = 1  
    Else  
        If alta Then  
            sql = "insert into Persona  
(CI, Fecha_Nac, Calle, Esquina, Numero, Primer_Apellido, Segundo_Apellido, Prim  
er_Nombre, Segundo_Nombre, Correo) VALUES (" & _ci & "," &  
Format(_fechanac, "yyyy-MM-dd HH:mm:ss") & "," & _calle & "," & _esquina &  
"," & _nropuerta & "," & _papellido & "," & _sapellido & "," & _pnombre & ","  
& _snombre & "," & _correo & ");"  
            Try  
                _cn.Execute(sql)  
            Catch ex As Exception  
                Return (2)  
            End Try  
            sql3 = "insert into Paciente(CI_Paciente, Estado_Civil, Sexo) VALUES  
(" & _ci & "," & _estadocivil & "," & _sexo & ");"  
            Try  
                _cn.Execute(sql3)
```



```
Catch ex As Exception
    Return (5)
End Try

sql4 = "insert into Posee(ID_Estado,CI_Paciente,Fechalnicio_E)
VALUES ('1','" & _ci & "','" & Format(Now(), "yyyy-MM-dd HH:mm:ss") & "');"
Try
    _cn.Execute(sql4)
Catch ex As Exception
    Return (5)
End Try

Else
    sql = "update Paciente set Estado_Civil=" & _estadocivil & ", Sexo="
& _sexo & " where CI_Paciente=" & _ci & ";"
    Try
        _cn.Execute(sql)
    Catch ex As Exception
        Return (5)
    End Try

    sql3 = "update Persona set Fecha_Nac=" & Format(_fechanac,
"yyyy-MM-dd HH:mm:ss") & ", Calle=" & _calle & ", Esquina=" & _esquina & ",
Numero=" & _ropuerta & ", Primer_Apellido=" & _papellido & ",
Segundo_Apellido=" & _apellido & ", Primer_Nombre=" & _pnombre & ",
Segundo_Nombre=" & _snombre & ", Correo=" & _correo & " where CI=" &
_ci & ";"
    Try
        _cn.Execute(sql3)
    Catch ex As Exception
        Return (7)
    End Try

End If

End If
```



```
Return (retorno)
End Function

Function GuardarT(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql2 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql2 = "insert into Persona_Teléfono(Cl,Telefono) VALUES (" & _ci &
",'" & _telefono & ");"
            Try
                _cn.Execute(sql2)
            Catch ex As Exception
                Return (3)
            End Try
        End If
        End If
        Return (retorno)
    End Function

    Function EliminarT(ByVal alta As Boolean) As Integer
        Dim retorno As Integer = 0
        Dim rs As New ADODB.Recordset
        Dim sql2 As String
        If cn.State = 0 Then
            retorno = 1
        Else
            If alta Then
```



```
sql2 = "delete from Persona_Teléfono where Telefono= "" & _telefono  
& "";"
```

```
Try
```

```
    _cn.Execute(sql2)
```

```
Catch ex As Exception
```

```
    Return (3)
```

```
End Try
```

```
End If
```

```
End If
```

```
Return (retorno)
```

```
End Function
```

```
Function Guardar1(ByVal alta As Boolean) As Integer
```

```
    Dim retorno As Integer = 0
```

```
    Dim rs As New ADODB.Recordset
```

```
    Dim sql3 As String
```

```
    Dim sql4 As String
```

```
    If cn.State = 0 Then
```

```
        retorno = 1
```

```
    Else
```

```
        If alta Then
```

```
            sql3 = "insert into Paciente(Cl_Paciente,Estado_Civil,Sexo) VALUES  
(" & _ci & "," & _estadocivil & "," & _sexo & ");"
```

```
        Try
```

```
            _cn.Execute(sql3)
```

```
        Catch ex As Exception
```

```
            Return (5)
```

```
        End Try
```

```
        sql4 = "insert into Posee(ID_Estado,Cl_Paciente,Fechainicio_E)  
VALUES ('1'," & _ci & "," & Format(Now(), "yyyy-MM-dd HH:mm:ss") & ");"
```

```
        Try
```



```
_cn.Execute(sql4)
Catch ex As Exception
    Return (5)
End Try
End If
Return (retorno)
End Function

Function adduser(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql3 As String
    Dim sql4 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql3 = "CREATE USER " & _usuario & "'@'192.168.1.209'
IDENTIFIED BY " & _contrasena & ";"
            Try
                _cn.Execute(sql3)
            Catch ex As Exception
                MsgBox(ex.ToString)
                Return (5)
            End Try
            sql3 = "GRANT GSAPacientes TO " & _usuario &
"'@'192.168.1.209';"
            Try
                _cn.Execute(sql3)
            Catch ex As Exception

```



```
    MsgBox(ex.ToString)
    Return (5)
End Try

sql4 = "insert into Usuario_Rol(Usuario,Rol,CI) VALUES (" &
_usuario & ",3'" & _ci & ");"

Try
    _cn.Execute(sql4)
Catch ex As Exception
    Return (6)
End Try

End If

Return (retorno)
End Function

Function Eliminar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0

    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Paciente set EstadoBorradoP= TRUE where CI_Paciente
=" & _ci & ";"

        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
    End If

    Return (retorno)
```



End Function

Function ecivil() As List(Of String)

 Dim retorno As New List(Of String)

 Dim rs As New ADODB.Recordset

 Dim sql As String

 sql = "SELECT DISTINCT Estado_Civil FROM Paciente;"

 Try

 rs = _cn.Execute(sql)

 Catch ex As Exception

 Return (retorno)

 End Try

 retorno.Clear()

 While Not rs.EOF

 retorno.Add(rs(0).Value)

 rs.MoveNext()

 End While

 rs.Close()

 rs = Nothing

 Return (retorno)

End Function

Function csexo() As List(Of Char)

 Dim retorno As New List(Of Char)

 Dim rs As New ADODB.Recordset

 Dim sql As String

 sql = "SELECT DISTINCT Sexo FROM Paciente;"

 Try

 rs = _cn.Execute(sql)

 Catch ex As Exception

 Return (retorno)



```
End Try
retorno.Clear()
While Not rs.EOF
    retorno.Add(rs(0).Value)
    rs.MoveNext()
End While
rs.Close()
rs = Nothing
Return (retorno)
End Function

Function estadop() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Paciente set EstadoBorradoP= False where CI_Paciente
        =" & _ci & ";"
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
    End If
    Return (retorno)
End Function

End Class
```



Personal Médico

Public Class PersonalMedico

```
Protected _ci As Integer  
Protected _pnombre As String  
Protected _snombre As String  
Protected _usuario As String  
Protected _contrasena As String  
Protected _papellido As String  
Protected _fechanac As Date  
Protected _sapellido As String  
Protected _calle As String  
Protected _nropuerta As String  
Protected _esquina As String  
Protected _correo As String  
Protected _telefono As String  
Protected _edad As Integer  
Protected _especialidad As String  
Protected _estadoborradom As Integer  
Protected _cn As ADODB.Connection  
  
Sub New()  
    _ci = 0  
    _edad = 0  
    _usuario = ""  
    _contrasena = ""  
    _estadoborradom = 0  
    _fechanac = New Date  
    _pnombre = ""  
    _correo = ""  
    _snombre = ""
```



```
_papellido = ""  
_sapellido = ""  
_nropuerta = ""  
_esquina = ""  
_calle = ""  
_telefono = ""  
_especialidad = ""  
_cn = New ADODB.Connection
```

End Sub

```
Sub New(ByVal usuario As String, ByVal contrasena As String, ByVal correo  
As String, ByVal estadoborrdom As Integer, ByVal ci As Integer, ByVal edad  
As Integer, ByVal fechanac As Date, ByVal calle As String, ByVal pnombre As  
String, ByVal snombre As String, ByVal papellido As String, ByVal sapellido As  
String, ByVal nropuerta As String, ByVal esquina As String, ByVal telefono As  
String, ByVal especialidad As String, ByVal cn As ADODB.Connection)
```

```
    _ci = ci  
    _usuario = usuario  
    _contrasena = contrasena  
    _pnombre = pnombre  
    _correo = correo  
    _edad = edad  
    _snombre = snombre  
    _fechanac = fechanac  
    _calle = calle  
    _papellido = papellido  
    _sapellido = sapellido  
    _nropuerta = nropuerta  
    _esquina = esquina  
    _telefono = telefono  
    _especialidad = especialidad  
    _cn = cn
```



```
_estadoborradom = estadoborradom  
End Sub  
  
Property usuario As String  
    Get  
        Return (_usuario)  
    End Get  
    Set(ByVal value As String)  
        _usuario = value  
    End Set  
End Property  
  
Property contrasena As String  
    Get  
        Return (_contrasena)  
    End Get  
    Set(ByVal value As String)  
        _contrasena = value  
    End Set  
End Property  
  
Property correo As String  
    Get  
        Return (_correo)  
    End Get  
    Set(ByVal value As String)  
        _correo = value  
    End Set  
End Property  
  
Property estadoborradom As Integer  
    Get  
        Return (_estadoborradom)
```



```
End Get

Set(ByVal value As Integer)
    _estadoborradom = value

End Set

End Property

Property ci As Integer
    Get
        Return (_ci)
    End Get
    Set(ByVal value As Integer)
        _ci = value
    End Set
End Property

Property edad As Integer
    Get
        Return (_edad)
    End Get
    Set(ByVal value As Integer)
        _edad = value
    End Set
End Property

Property pnombre As String
    Get
        Return (_pnombre)
    End Get
    Set(ByVal value As String)
        _pnombre = value
    End Set
End Property
```



Property fechanac As Date

Get

 Return (_fechanac)

End Get

Set(ByVal value As Date)

 _fechanac = value

End Set

End Property

Property calle As String

Get

 Return (_calle)

End Get

Set(ByVal value As String)

 _calle = value

End Set

End Property

Property snombre As String

Get

 Return (_pnombre)

End Get

Set(ByVal value As String)

 _snombre = value

End Set

End Property

Property papellido As String

Get

 Return (_papellido)

End Get

Set(ByVal value As String)



```
_papellido = value
End Set
End Property
Property sapellido As String
    Get
        Return (_sapellido)
    End Get
    Set(ByVal value As String)
        _sapellido = value
    End Set
End Property
Property nropuerta As String
    Get
        Return (_nropuerta)
    End Get
    Set(ByVal value As String)
        _nropuerta = value
    End Set
End Property
Property esquina As String
    Get
        Return (_esquina)
    End Get
    Set(ByVal value As String)
        _esquina = value
    End Set
End Property
Property telefono As String
    Get
```



```
Return (_telefono)
End Get
Set(ByVal value As String)
    _telefono = value
End Set
End Property
Property especialidad As String
    Get
        Return (_especialidad)
    End Get
    Set(ByVal value As String)
        _especialidad = value
    End Set
End Property
Property cn As ADODB.Connection
    Get
        Return (_cn)
    End Get
    Set(ByVal value As ADODB.Connection)
        _cn = value
    End Set
End Property
Function Buscar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
```



```
sql = "select * from Persona as p JOIN Personal_Medico as m ON  
p.Cl=m.Cl where m.Cl=" & _ci & ";"
```

Try

```
rs.Open(sql, _cn)
```

Catch ex As Exception

```
Return (2)
```

End Try

If rs.RecordCount = 0 Then

```
retorno = 3
```

Else

```
_pnombre = rs("Primer_Nombre").Value
```

```
_snombre = rs("Segundo_Nombre").Value
```

```
_papellido = rs("Primer_Apellido").Value
```

```
_sapellido = rs("Segundo_Apellido").Value
```

```
_fechanac = rs("Fecha_Nac").Value
```

```
_calle = rs("Calle").Value
```

```
_esquina = rs("Esquina").Value
```

```
_nropuerta = rs("Numero").Value
```

```
_estadoborradoM = rs("EstadoBorradoM").Value
```

```
rs.Close()
```

```
sql = "select Telefono from Persona_Telefono where Cl=" & _ci & ";"
```

Try

```
rs.Open(sql, _cn)
```

```
_telefono = rs("Telefono").Value
```

Catch ex As Exception

```
Return (4)
```

End Try

```
rs.Close()
```

```
sql = "select Especialidad from Medico where Cl_Medico=" & _ci & ";"
```



Try

```
rs.Open(sql, _cn)
If rs.EOF And rs.BOF Then
    _especialidad = "-"
Else
    _especialidad = rs("Especialidad").Value
End If
```

Catch ex As Exception

```
Return (5)
End Try
```

```
rs.Close()
```

```
sql = "select TIMESTAMPDIFF(YEAR, Fecha_Nac, curdate()) as Edad from Persona where Cl=" & _ci & ";"
Try
```

```
rs.Open(sql, _cn)
_edad = rs("Edad").Value
rs.Close()
```

Catch ex As Exception

```
Return (6)
End Try
```

```
End If
```

End If

```
Return (retorno)
```

End Function

Function estado() As Integer

```
Dim sql As String
Dim rs As New ADODB.Recordset
Dim retorno As Integer = 0
If cn.State = 0 Then
```



```
retorno = 1

Else

    sql = "update Personal_Medico set EstadoBorradoM= FALSE where CI
    =" & _ci & ";"

    Try

        _cn.Execute(sql)

    Catch ex As Exception

        Return (2)

    End Try

End If

Return (retorno)

End Function

Function Guardar1(ByVal alta As Boolean) As Integer

    Dim retorno As Integer = 0

    Dim rs As New ADODB.Recordset

    Dim sql As String

    Dim sql3 As String

    Dim sql4 As String

    If cn.State = 0 Then

        retorno = 1

    Else

        If alta Then

            sql = "insert into Persona
(Cl, Fecha_Nac, Calle, Esquina, Numero, Primer_Apellido, Segundo_Apellido, Prim
er_Nombre, Segundo_Nombre, Correo) VALUES (" & _ci & "," &
Format(_fechanac, "yyyy-MM-dd HH:mm:ss") & "," & _calle & "," & _esquina &
"," & _nropuerta & "," & _papellido & "," & _sapellido & "," & _pnombre & ","
& _snombre & "," & _correo & ");"

            Try

                _cn.Execute(sql)

            Catch ex As Exception


```



```
    MsgBox(ex.ToString)  
    Return (2)  
End Try  
  
sql3 = "insert into Personal_Medico(CI) VALUES (" & _ci & ");"  
Try  
    _cn.Execute(sql3)  
Catch ex As Exception  
    Return (4)  
End Try  
  
sql4 = "insert into Medico(CI_Medico, Especialidad) VALUES (" & _ci  
& "," & _especialidad & ");"  
Try  
    _cn.Execute(sql4)  
Catch ex As Exception  
    Return (4)  
End Try  
  
Else  
    sql = "update Medico set Especialidad=" & _especialidad & " where  
    CI_Medico=" & _ci & ","  
    Try  
        _cn.Execute(sql)  
    Catch ex As Exception  
        MsgBox(ex.ToString)  
        Return (5)  
    End Try  
  
    sql3 = "update Persona set Fecha_Nac=" & Format(_fechanac,  
    "yyyy-MM-dd HH:mm:ss") & ", Calle=" & _calle & ", Esquina=" & _esquina & ",  
    Numero=" & _nropuerta & ", Primer_Apellido=" & _papellido & ",  
    Segundo_Apellido=" & _sapellido & ", Primer_Nombre=" & _pnombre & ",  
    Segundo_Nombre=" & _snombre & ", Correo=" & _correo & " where CI=" &  
    _ci & ";"
```



```
Try
    _cn.Execute(sql3)
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (7)
End Try
End If
End If
Return (retorno)
End Function

Function addusermed(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql3 As String
    Dim sql4 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql3 = "CREATE USER " & _usuario & "'@'192.168.1.209'
IDENTIFIED BY " & _contrasena & ";"
            Try
                _cn.Execute(sql3)
            Catch ex As Exception
                MsgBox(ex.ToString)
                Return (5)
            End Try
            sql3 = "GRANT GSAMedicos TO " & _usuario & "'@'192.168.1.209';"
            Try
```



```
_cn.Execute(sql3)
Catch ex As Exception
    Return (5)
End Try

sql4 = "insert into Usuario_Rol(Usuario,Rol) VALUES (" & _usuario &
",,'1'," & _ci & ")"
Try
    _cn.Execute(sql4)
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (6)
End Try

End If
Return (retorno)
End Function

Function adduseraux(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql3 As String
    Dim sql4 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
            sql3 = "CREATE USER " & _usuario & "'@'192.168.1.209'
IDENTIFIED BY " & _contrasena & "';"
            Try
                _cn.Execute(sql3)
            Catch ex As Exception

```



```
        Return (5)

    End Try

    sql3 = "GRANT GSAAuxiliares TO '" & _usuario &
    "'@'192.168.1.209';"

    Try
        _cn.Execute(sql3)
    Catch ex As Exception
        Return (5)
    End Try

    sql4 = "insert into Usuario_Rol(Usuario,Rol) VALUES ('" & _usuario &
    "','"2','" & _ci & ");"

    Try
        _cn.Execute(sql4)
    Catch ex As Exception
        Return (6)
    End Try

    End If

    End If

    Return (retorno)

End Function

Function Guardar2(ByVal alta As Boolean) As Integer
    Dim retorno As Integer = 0
    Dim rs As New ADODB.Recordset
    Dim sql As String
    Dim sql3 As String
    If cn.State = 0 Then
        retorno = 1
    Else
        If alta Then
```



```
sql = "insert into Persona
(CI, Fecha_Nac, Calle, Esquina, Numero, Primer_Apellido, Segundo_Apellido, Prim
er_Nombre, Segundo_Nombre, Correo) VALUES (" & _ci & "," &
Format(_fechanac, "yyyy-MM-dd HH:mm:ss") & "," & _calle & "," & _esquina &
"," & _nropuerta & "," & _papellido & "," & _sapellido & "," & _pnombre & ","
& _snombre & "," & _correo & ");"

Try
    _cn.Execute(sql)
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (2)
End Try

sql3 = "insert into Personal_Medico(CI) VALUES (" & _ci & ");"

Try
    _cn.Execute(sql3)
Catch ex As Exception
    Return (4)
End Try

Else
    sql3 = "update Persona set Fecha_Nac=" & Format(_fechanac,
"yyyy-MM-dd HH:mm:ss") & ", Calle=" & _calle & ", Esquina=" & _esquina & ",
Numero=" & _nropuerta & ", Primer_Apellido=" & _papellido & ",
Segundo_Apellido=" & _sapellido & ", Primer_Nombre=" & _pnombre & ",
Segundo_Nombre=" & _snombre & ", Correo=" & _correo & " where CI=" &
_ci & ","

Try
    _cn.Execute(sql3)
Catch ex As Exception
    MsgBox(ex.ToString)
    Return (7)
End Try

End If
```



```
End If  
Return (retorno)  
End Function  
  
Function Guardar3(ByVal alta As Boolean) As Integer  
    Dim retorno As Integer = 0  
    Dim rs As New ADODB.Recordset  
    Dim sql3 As String  
    Dim sql4 As String  
    If cn.State = 0 Then  
        retorno = 1  
    Else  
        If alta Then  
            sql3 = "insert into Personal_Medico(Cl) VALUES (" & _ci & ");"  
            Try  
                _cn.Execute(sql3)  
            Catch ex As Exception  
                Return (4)  
            End Try  
            sql4 = "insert into Medico(Cl_Medico, Especialidad) VALUES (" & _ci  
& "," & _especialidad & ");"  
            Try  
                _cn.Execute(sql4)  
            Catch ex As Exception  
                Return (4)  
            End Try  
        End If  
    End If  
    Return (retorno)  
End Function
```



```
Function verifyci() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select CI_Paciente from Paciente WHERE CI_Paciente=" & _ci &
        ","
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            MsgBox(ex.ToString)
            Return (2)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else
            _ci = rs("CI_Paciente").Value
        End If
        rs.Close()
    End If
    Return (retorno)
End Function

Function pmtelefono() As List(Of String)
    Dim retorno As New List(Of String)
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "select Telefono from Persona_Telefono where CI=" & _ci & ":"
```



Try

```
rs = _cn.Execute(sql)
```

Catch ex As Exception

```
    Return (retorno)
```

End Try

```
retorno.Clear()
```

While Not rs.EOF

```
    retorno.Add(rs(0).Value)
```

```
    rs.MoveNext()
```

End While

```
rs.Close()
```

```
rs = Nothing
```

```
Return (retorno)
```

End Function

Function Guardar4(ByVal alta As Boolean) As Integer

```
Dim retorno As Integer = 0
```

```
Dim rs As New ADODB.Recordset
```

```
Dim sql3 As String
```

If cn.State = 0 Then

```
    retorno = 1
```

Else

If alta Then

```
    sql3 = "insert into Personal_Medico(Cl) VALUES (" & _ci & ");"
```

Try

```
    _cn.Execute(sql3)
```

Catch ex As Exception

```
    Return (4)
```

End Try

End If



```
End If
Return (retorno)
End Function

Function Eliminar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Personal_Medico set EstadoBorradoM= TRUE where CI
        =" & _ci & ","
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
    End If
    Return (retorno)
End Function

End Class
```



Rutinas

Module Rutinas

```
Public cn As New ADODB.Connection  
Public cipub As String  
Public Sub doyPermisos(ByVal usuario As String)  
    Dim sql As String  
    Dim rs As New ADODB.Recordset  
    frmPrincipal.menuInformacion.Enabled = False  
    frmPrincipal.InformaciónPacienteToolStripMenuItem.Enabled = False  
    frmPrincipal.menuFormularios.Enabled = False  
    frmPrincipal.FormularioToolStripMenuItem.Enabled = False  
    frmPrincipal.menu_Consulta_Formularios.Enabled = False  
    frmPrincipal.menuPersonalMedico.Enabled = False  
    frmPrincipal.menuEntrevistaInicial.Enabled = False  
    frmPrincipal.menuDiagnstico.Enabled = False  
    frmPrincipal.menuIngTratamiento.Enabled = False  
    frmPrincipal.menuIngAnalisis.Enabled = False  
    frmPrincipal.menuConsAnalisis.Enabled = False  
    frmPrincipal.menuConsDiagnstico.Enabled = False  
    frmPrincipal.DiagnosticosToolStripMenuItem.Enabled = False  
    frmPrincipal.TratamientosToolStripMenuItem.Enabled = False  
    frmPrincipal.AnalisisClínicosToolStripMenuItem.Enabled = False  
    frmPrincipal.EntrevistaToolStripMenuItem.Enabled = False  
    If cn.State = 0 Then  
        MsgBox("Debe abrir su sesión.")  
    Else  
        sql = "select Rol from Usuario_Rol where Usuario=" & usuario & ";"  
        Try  
            rs.Open(sql, cn)
```



Catch ex As Exception

```
    MsgBox("Hubo errores al obtener rol del usuario.")
```

```
    Exit Sub
```

```
End Try
```

```
If rs.RecordCount = 0 Then
```

```
    MsgBox("El usuario no tiene rol asignado, comuníquese con el adm.  
del sistema.")
```

```
Else
```

```
    Select Case rs(0).Value
```

```
        Case 1 'Medico
```

```
            'Formularios
```

```
            frmPrincipal.menuFormularios.Enabled = True
```

```
            'Ingreso datos
```

```
            frmPrincipal.FormularioToolStripMenuItem.Enabled = True
```

```
            frmPrincipal.menuIngTratamiento.Enabled = True
```

```
            frmPrincipal.menuIngAnalisis.Enabled = True
```

```
            frmPrincipal.menuEntrevistaInicial.Enabled = True
```

```
            frmPrincipal.menuDiagnostico.Enabled = True
```

```
            'Consultar info
```

```
            frmPrincipal.InformaciónPacienteToolStripMenuItem.Enabled =
```

```
True
```

```
            frmPrincipal.menu_Consulta_Formularios.Enabled = True
```

```
            frmPrincipal.DiagnosticosToolStripMenuItem.Enabled = True
```

```
            frmPrincipal.TratamientosToolStripMenuItem.Enabled = True
```

```
            frmPrincipal.AnálisisClínicosToolStripMenuItem.Enabled = True
```

```
            frmPrincipal.EntrevistaToolStripMenuItem.Enabled = True
```

```
        Case 2 'Auxiliar
```

```
            'Formularios
```

```
            frmPrincipal.menuFormularios.Enabled = True
```

```
            'Ingreso datos
```



```
frmPrincipal.FormularioToolStripMenuItem.Enabled = True
frmPrincipal.menuIngTratamiento.Enabled = True
frmPrincipal.menuIngAnalisis.Enabled = True
frmPrincipal.menuEntrevistaInicial.Enabled = True
frmPrincipal.menuDiagnostico.Enabled = True

Case 3 'Paciente
    'Consultar diagnostico
    frmPrincipal.menuConsDiagnostico.Enabled = True
    'Consultar analisis
    frmPrincipal.menuConsAnalisis.Enabled = True

Case 4 'Administrador
    frmPrincipal.menuInformacion.Enabled = True
    frmPrincipal.menuPersonalMedico.Enabled = True

End Select

End If

rs.Close()

End If

rs = Nothing

End Sub

'Permitir números con espacios

Public Sub permitirLetrasCE(ByVal e As
System.Windows.Forms.KeyPressEventArgs)

If Char.IsLetter(e.KeyChar) Then
    e.Handled = False
ElseIf Char.IsControl(e.KeyChar) Then
    e.Handled = False
ElseIf Char.IsSeparator(e.KeyChar) Then
    e.Handled = False
Else
```



```
e.Handled = True  
    MessageBox.Show("No se permite ingresar números o símbolos.",  
    "ADVERTENCIA")  
End If  
End Sub  
'Permitir números sin espacios  
Public Sub permitirNumerosSE(ByVal e As  
System.Windows.Forms.KeyPressEventArgs)  
If Char.IsNumber(e.KeyChar) Then  
    e.Handled = False  
ElseIf Char.IsControl(e.KeyChar) Then  
    e.Handled = False  
ElseIf Char.IsSeparator(e.KeyChar) Then  
    e.Handled = True  
    MessageBox.Show("No se permite ingresar espacios.",  
    "ADVERTENCIA")  
Else  
    e.Handled = True  
    MessageBox.Show("No se permite ingresar letras.", "ADVERTENCIA")  
End If  
End Sub  
End Module
```



Tratamientos

Public Class Tratamientos

 Protected _ci As Integer

 Protected _nombre As String

 Protected _id As Integer

 Protected _fechainicio As Date

 Protected _fechafinal As Date

 Protected _descripcion As String

 Protected _estadoborradoP As Integer

 Protected _cn As ADODB.Connection

'Metodos constructores

Sub New()

 _ci = 0

 _estadoborradoP = 0

 _nombre = ""

 _id = 0

 _fechainicio = New Date

 _fechafinal = New Date

 _descripcion = ""

 _cn = New ADODB.Connection

End Sub

Sub New(ByVal estadoborradoP As Integer, ByVal ci As Integer, ByVal nombre As String, ByVal id As Integer, ByVal fechainicio As Date, ByVal fechafinal As Date, ByVal descripcion As String, ByVal cn As ADODB.Connection)

 _ci = ci

 _nombre = nombre

 _estadoborradoP = estadoborradoP

 _id = id

 _fechainicio = fechainicio



```
_fechafinal = fechafinal  
_descripcion = descripcion  
_cn = cn  
End Sub  
'Get y set de las propiedades  
Property estadoborradop As Integer  
    Get  
        Return (_estadoborradop)  
    End Get  
    Set(ByVal value As Integer)  
        _estadoborradop = value  
    End Set  
End Property  
Property ci As Integer  
    Get  
        Return (_ci)  
    End Get  
    Set(ByVal value As Integer)  
        _ci = value  
    End Set  
End Property  
Property nombre As String  
    Get  
        Return (_nombre)  
    End Get  
    Set(ByVal value As String)  
        _nombre = value  
    End Set  
End Property
```



```
Property id As Integer
    Get
        Return (_id)
    End Get
    Set(ByVal value As Integer)
        _id = value
    End Set
End Property

Property fechainicio As Date
    Get
        Return (_fechainicio)
    End Get
    Set(ByVal value As Date)
        _fechainicio = value
    End Set
End Property

Property fechafinal As Date
    Get
        Return (_fechafinal)
    End Get
    Set(ByVal value As Date)
        _fechafinal = value
    End Set
End Property

Property descripcion As String
    Get
        Return (_descripcion)
    End Get
    Set(ByVal value As String)
```



```
_descripcion = value
End Set
End Property
Property cn As ADODB.Connection
Get
    Return (_cn)
End Get
Set(ByVal value As ADODB.Connection)
    _cn = value
End Set
End Property
Function Buscar() As Integer
    Dim sql As String
    Dim sql2 As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select * from Tratamiento as t JOIN Realiza as r ON
t.ID_Trata=r.ID_Trata where r.ID_Trata=" & _id & " AND r.Cl_Paciente=" & _ci
& ";"
        Try
            rs.Open(sql, _cn)
        Catch ex As Exception
            Return (2)
        End Try
        If rs.RecordCount = 0 Then
            retorno = 3
        Else

```



```
_nombre = rs("Nombre").Value
_ci = rs("CI_Paciente").Value
_id = rs("ID_Trata").Value
_fechainicio = rs("Fechainicio").Value
_descripcion = rs("Descripcion_Trat").Value
rs.Close()

sql2 = "select r.FechaFinal from Tratamiento as t JOIN Realiza as r
ON t.ID_Trata=r.ID_Trata where r.ID_Trata=" & _id & " AND r.CI_Paciente=" &
_ci & ";"

Try
    rs.Open(sql2, _cn)
Catch ex As Exception
    Return (2)
End Try

If rs.RecordCount = 0 Then
    retorno = 3
Else
    If rs.EOF And rs.BOF Then
        _fechainicio = "NO ESTABLECIDA"
    Else
        _fechafinal = rs("FechaFinal").Value
    End If
    rs.Close()
End If

End If

Return (retorno)
```

End Function

```
Function Guardar(ByVal alta As Boolean) As Integer
```



```
Dim retorno As Integer = 0
Dim rs As New ADODB.Recordset
Dim sql As String
Dim sql2 As String
Dim sql3 As String
Dim sql4 As String
If cn.State = 0 Then
    retorno = 1
Else
    If alta Then
        sql = "insert into Tratamiento (Nombre,Descripcion_Trat) VALUES (""
& _nombre & ","" & _descripcion & ")";
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
        sql2 = "SELECT MAX(ID_Trata) AS ID_TrataC FROM Tratamiento;"
        Try
            rs.Open(sql2, _cn)
        Catch ex As Exception
            Return (3)
        End Try
        _id = rs("ID_TrataC").Value
        rs.Close()
        sql3 = "insert into
Realiza(Fechainicio,FechaFinal,CI_Paciente,ID_Trata) VALUES ("" &
Format(_fechainicio, "yyyy-MM-dd HH:mm:ss") & ","" & Format(_fechafinal,
"yyyy-MM-dd HH:mm:ss") & ","" & _ci & "," & _id & ")";
        Try
```



```
_cn.Execute(sql3)
Catch ex As Exception
    Return (4)
End Try

sql4 = "insert into Posee(ID_Estado,CI_Paciente,Fechainicio_E)
VALUES ('4','" & _ci & "','" & Format(Now(), "yyyy-MM-dd HH:mm:ss") & ");"
Try
    _cn.Execute(sql4)
Catch ex As Exception
    Return (3)
End Try

Else
    sql = "update Tratamiento set Nombre="" & _nombre & """",
Descripcion_Trat="" & _descripcion & "" where ID_Trata="" & _id & ";"

    Try
        _cn.Execute(sql)
    Catch ex As Exception
        MsgBox(ex.ToString)
        Return (4)
    End Try

    sql2 = "update Realiza set Fechainicio="" & Format(_fechainicio,
"yyyy-MM-dd HH:mm:ss") & "", FechaFinal="" & Format(_fechafinal, "yyyy-MM-
dd HH:mm:ss") & "" where ID_Trata="" & _id & ";"

    Try
        _cn.Execute(sql2)
    Catch ex As Exception
        MsgBox(ex.ToString)
        Return (4)
    End Try
```



```
End If
End If
Return (retorno)
End Function

Function Eliminar() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Tratamiento set EstadoBorradoT= TRUE where ID_Trata
        =" & _id & ";"
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
    End If
    Return (retorno)
End Function

Function verifyci() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "select pa.Cl_Paciente, pa.EstadoBorradoP from Persona as p
JOIN Paciente as pa ON p.Cl=pa.Cl_Paciente JOIN Realiza as r ON
```



```
pa.CI_Paciente=r.CI_Paciente JOIN Tratamiento as t ON r.ID_Trata=t.ID_Trata  
where EstadoBorradoT = FALSE AND pa.CI_Paciente=" & _ci & ";"
```

Try

```
rs.Open(sql, _cn)
```

Catch ex As Exception

```
MsgBox(ex.ToString)
```

Return (4)

End Try

If rs.RecordCount = 0 Then

```
retorno = 3
```

Else

```
_ci = rs("CI_Paciente").Value
```

```
_estadoborradoP = rs("EstadoBorradoP").Value
```

```
rs.Close()
```

End If

End If

Return (retorno)

End Function

Function verifyci2() As Integer

```
Dim sql As String
```

```
Dim rs As New ADODB.Recordset
```

```
Dim retorno As Integer = 0
```

If cn.State = 0 Then

```
retorno = 1
```

Else

```
sql = "select CI_Paciente, EstadoBorradoP from Paciente where  
CI_Paciente=" & _ci & ";"
```

Try

```
rs.Open(sql, _cn)
```

Catch ex As Exception



```
Return (4)

End Try

If rs.RecordCount = 0 Then
    retorno = 3
Else
    _ci = rs("CI_Paciente").Value
    _estadoborradoP = rs("EstadoBorradoP").Value
    rs.Close()
End If

End If

Return (retorno)

End Function

Function idtratamiento() As List(Of Integer)
    Dim retorno As New List(Of Integer)
    Dim rs As New ADODB.Recordset
    Dim sql As String
    sql = "SELECT DISTINCT r.ID_Trata FROM Realiza as r JOIN
        Tratamiento as t ON r.ID_Trata=t.ID_Trata WHERE t.EstadoBorradoT=False
        AND r.CI_Paciente=" & _ci & ";"
    Try
        rs = _cn.Execute(sql)
    Catch ex As Exception
        Return (retorno)
    End Try
    retorno.Clear()
    While Not rs.EOF
        retorno.Add(rs(0).Value)
        rs.MoveNext()
    End While
    rs.Close()
```



```
rs = Nothing
Return (retorno)
End Function

Function estadop() As Integer
    Dim sql As String
    Dim rs As New ADODB.Recordset
    Dim retorno As Integer = 0
    If cn.State = 0 Then
        retorno = 1
    Else
        sql = "update Paciente set EstadoBorradoP= False where CI_Paciente
        =" & _ci & ","
        Try
            _cn.Execute(sql)
        Catch ex As Exception
            Return (2)
        End Try
    End If
    Return (retorno)
End Function

End Class
```



Formularios

frmABMFormulario

Public Class frmABMFormulario

```
Private Sub cmdCrear_Click(sender As Object, e As EventArgs) Handles cmdCrear.Click
```

```
    Dim f As New Formularios
```

```
    f.tipo = txtCrearTipoForm.Text
```

```
    f.cn = cn
```

```
    Select Case f.Guardar(1)
```

```
        Case 0
```

```
            cboTipo.Items.Clear()
```

```
            f.tipo = txtCrearTipoForm.Text
```

```
            MsgBox("Formulario creado éxitosamente.",
```

MessageBoxIcon.Information, "S.I.BI.M - GSA")

```
            f.cn = cn
```

```
            Dim lista As List(Of String)
```

```
            lista = f.tipoform()
```

```
            For Each Diagnostico In lista
```

```
                cboTipo.Items.Add(Diagnostico)
```

```
            Next
```

```
            Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
```

nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

```
            Case Else : MsgBox("Hubo errores al agregar un formulario. En caso de
```

persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -

GSA")

```
        End Select
```

```
        f = Nothing
```

```
    End Sub
```



```
Private Sub frmABMFormulario_Leave(sender As Object, e As EventArgs)  
Handles Me.Leave
```

```
    Me.Close()  
End Sub
```

```
Private Sub frmABMFormulario_Load(sender As Object, e As EventArgs)  
Handles MyBase.Load
```

```
    Dim f As New Formularios  
    f.cn = cn  
    cboTipo.Items.Clear()  
    Dim lista As List(Of String)  
    lista = f.tipoform()  
    For Each Diagnostico In lista  
        cboTipo.Items.Add(Diagnostico)  
    Next  
    cboPregunta.Items.Clear()  
    Dim lista1 As List(Of String)  
    lista1 = f.preg()  
    For Each Pregunta In lista1  
        cboPregunta.Items.Add(Pregunta)  
    Next  
End Sub
```

```
Private Sub cboTipo_SelectedIndexChanged(sender As Object, e As EventArgs) Handles cboTipo.SelectedIndexChanged
```

```
    Dim f As New Formularios  
    f.tipo = cboTipo.SelectedItem()  
    f.cn = cn  
    Dim lista As List(Of Integer)  
    lista = f.idform()
```



```
    cboID.Items.Clear()
    cboID.ResetText()
    For Each id In lista
        cboID.Items.Add(id)
    Next
    cmdEliminarForm.Enabled = True
End Sub
```

```
Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles cmdIngresar.Click
```

```
    Dim f As New Formularios
    f.pregunta = cboPregunta.SelectedItem()
    f.idf = cboID.SelectedItem()
    f.cn = cn
    Select Case f.Guardar2(1)
        Case 0
            MsgBox("Pregunta ingresada éxitosamente.",
                    MessageBoxIcon.Information, "S.I.BI.M - GSA")
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
        Case Else : MsgBox("Hubo errores al ingresar la pregunta. En caso de
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -
GSA")
    End Select
    f = Nothing
End Sub
```

```
Private Sub cmdModPregunta_Click(sender As Object, e As EventArgs) Handles cmdModPregunta.Click
```

```
    Dim f As New Formularios
```



```
f.pregunta = cboPregunta.SelectedItem()
f.nuevapregunta = txtNuevaPregunta.Text()
f.idf = cbolD.SelectedItem()
f.cn = cn
Select Case f.Guardar2(0)
    Case 0
        cboPregunta.Items.Clear()
        cboPregunta.ResetText()
        f.cn = cn
        Dim lista1 As List(Of String)
        lista1 = f.preg()
        For Each Pregunta In lista1
            cboPregunta.Items.Add(Pregunta)
        Next
        MsgBox("Pregunta modificada con éxito.",_
        MessageBoxIcon.Information, "S.I.BI.M - GSA")
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
    Case Else : MsgBox("Hubo errores al modificar la pregunta. En caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
End Select
f = Nothing
End Sub
```

```
Private Sub cmdEliminarPregunta_Click(sender As Object, e As EventArgs)
Handles cmdEliminarPregunta.Click
    Dim f As New Formularios
    f.pregunta = cboPregunta.SelectedItem()
    f.nuevapregunta = txtNuevaPregunta.Text()
    f.idf = cbolD.SelectedItem()
```



```
f.cn = cn
```

```
Select Case f.Eliminar()
```

```
Case 0
```

```
    cboPregunta.Items.Clear()
```

```
    f.cn = cn
```

```
    cboPregunta.ResetText()
```

```
    Dim lista1 As List(Of String)
```

```
    lista1 = f.preg()
```

```
    For Each Pregunta In lista1
```

```
        cboPregunta.Items.Add(Pregunta)
```

```
    Next
```

```
    MsgBox("Pregunta eliminada con éxito.",  
MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
    Case Else : MsgBox("Hubo errores al eliminar la pregunta. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
f = Nothing
```

```
End Sub
```

```
Private Sub cmdEliminarForm_Click(sender As Object, e As EventArgs)  
Handles cmdEliminarForm.Click
```

```
    Dim f As New Formularios
```

```
    f.idf = cbold.SelectedItem()
```

```
    f.cn = cn
```

```
    Select Case f.Eliminar1()
```

```
    Case 0
```

```
        cbold.Items.Clear()
```

```
        f.cn = cn
```



```
cbold.ResetText()
Dim lista As List(Of Integer)
lista = f.idform()
For Each id In lista
    cbold.Items.Add(id)
Next
cboTipo.Items.Clear()
cboTipo.ResetText()
Dim lista1 As List(Of String)
lista1 = f.tipoform()
For Each Diagnostico In lista1
    cboTipo.Items.Add(Diagnostico)
Next
MsgBox("Formulario eliminado con éxito.",
MessageBoxIcon.Information, "S.I.BI.M - GSA")
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
Case Else : MsgBox("Hubo errores al eliminar el formulario. En caso de
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -
GSA")
End Select
f = Nothing
End Sub

Private Sub Button1_Click_1(sender As Object, e As EventArgs) Handles
Button1.Click
    Dim f As New Formularios
    f.pregunta = txtCrearPregunta.Text
    f.cn = cn
    Select Case f.Guardar1(1)
        Case 0
```



```
cboPregunta.Items.Clear()

MsgBox("Pregunta creada éxitosamente.",
MessageBoxIcon.Information, "S.I.BI.M - GSA")

f.cn = cn

Dim lista1 As List(Of String)
lista1 = f.preg()
For Each Pregunta In lista1
    cboPregunta.Items.Add(Pregunta)
Next

Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al crear la pregunta. En caso de
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -
GSA")

End Select

f = Nothing

End Sub

Private Sub cboPregunta_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles cboPregunta.SelectedIndexChanged

If cboPregunta.SelectedItem Is Nothing Then
    cmdEliminarPregunta.Enabled = False
    cmdModPregunta.Enabled = False
    cmdIngresar.Enabled = False
Else
    cmdEliminarPregunta.Enabled = True
    cmdModPregunta.Enabled = True
    cmdIngresar.Enabled = True
End If

End Sub

End Class
```



frmAdmPersonalMedico

Public Class frmAdmPersonalMedico

```
Private Sub txtEdad_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtEdad.KeyPress  
    permitirNumerosSE(e)  
End Sub
```

```
Private Sub txtBCI_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtCl.KeyPress  
    permitirNumerosSE(e)  
End Sub
```

```
Private Sub txtPnombre_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtPnombre.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtPapellido_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtPapellido.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtSnombre_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtSnombre.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtSapellido_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtSapellido.KeyPress  
    permitirLetrasCE(e)
```



End Sub

```
Private Sub rbMedico_CheckedChanged(sender As Object, e As EventArgs)
Handles rbMedico.CheckedChanged
```

```
    lblEspecialidad.Visible = True
```

```
    txtEspecialidad.Visible = True
```

End Sub

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim m As New PersonalMedico
```

```
    m.ci = Val(txtCI.Text)
```

```
    m.telefono = cbTelefono.Text
```

```
    m.cn = cn
```

```
    Select Case m.verifyci()
```

```
        Case 0
```

```
            Select Case m.Buscar()
```

```
                Case 0 'Encontré
```

```
                    If m.estadoborradom = 1 Then
```

```
                        If MsgBox("El médico o auxiliar se dió de baja, ¿Desaea darle
de alta?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                            Select Case m.estado()
```

```
                                Case 0
```

```
                                    MsgBox("El médico o auxiliar se dio de alta
éxitosamente.", MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                                Case 1
```

```
                                    MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                                Case Else
```



```
        MsgBox("Hubo errores al efectuar el alta del médico o  
auxiliar. En caso de persistir, consulte al adm. del sistema.",  
MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
    End Select
```

```
End If
```

```
Else
```

```
    txtPnombre.Text = m.pnombre
```

```
    txtSnombre.Text = m.snombre
```

```
    txtPapellido.Text = m.papellido
```

```
    txtSapellido.Text = m.sapellido
```

```
    dtpFechaNac.Text = m.fechanac
```

```
    txtEdad.Text = m.edad
```

```
    txtCalle.Text = m.calle
```

```
    txtTelefono.Text = m.telefono
```

```
    txtNpuerta.Text = m.nropuerta
```

```
    txtEsquina.Text = m.esquina
```

```
    txtEspecialidad.Text = m.especialidad
```

```
    gbBuscar.Enabled = False
```

```
    gdatos.Visible = True
```

```
    cmdIngresar.Enabled = False
```

```
    cmdModificar.Enabled = True
```

```
    cmdEliminar.Enabled = True
```

```
    Dim lista2 As List(Of String)
```

```
    lista2 = m.pmtelefono()
```

```
    For Each Telefono In lista2
```

```
        cbTelefono.Items.Add(Telefono)
```

```
    Next
```

```
    gbTelefono.Visible = True
```

```
    If m.especialidad.Equals("-") Then
```



```
rbAuxiliarMedico.Checked = True
rbMedico.Checked = False
txtEspecialidad.Visible = False
txtEspecialidad.Clear()

Else
    rbAuxiliarMedico.Checked = False
    rbMedico.Checked = True
    txtEspecialidad.Visible = True
End If

End If

Case 3 'No encontré
    MsgBox("Este paciente ya tiene los datos necesarios
ingresados, para ingresar como personal médico solamente hace falta ingresar
los siguientes datos.", MessageBoxButtons.Information, "S.I.BI.M - GSA")
    frmUsuario.Show()
    frmUsuario.cmdIngresar.Visible = True
    frmMedico.Enabled = False
    frmMedico.Show()

Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al buscar el médico o paciente.
En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error,
"S.I.BI.M - GSA")

End Select

Case 3 'Encontré
    Select Case m.Buscar
        Case 0 'Encontré
            If m.estadoborradom = 1 Then
                If MsgBox("El médico o auxiliar se dió de baja, ¿Desea darle
de alta?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
                    Select Case m.estado()

```



Case 0

 MsgBox("El médico o auxiliar se dio de alta
éxitosamente.", MessageBoxButtons.Information, "S.I.BI.M - GSA")

Case 1

 MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else

 MsgBox("Hubo errores al efectuar el alta del médico o
auxiliar. En caso de persistir, consulte al adm. del sistema.",
MessageBoxIcon.Error, "S.I.BI.M - GSA")

End Select

End If

Else

 txtPnombre.Text = m.pnombre
 txtSnombre.Text = m.snombre
 txtPapellido.Text = m.papellido
 txtSapellido.Text = m.sapellido
 dtpFechaNac.Text = m.fechanac
 txtEdad.Text = m.edad
 txtCalle.Text = m.calle
 txtTelefono.Text = m.telefono
 txtNpuerta.Text = m.nropuerta
 txtEsquina.Text = m.esquina
 txtEspecialidad.Text = m.especialidad
 gbBuscar.Enabled = False
 gbDatos.Visible = True
 cmdIngresar.Enabled = False
 cmdModificar.Enabled = True
 cmdEliminar.Enabled = True
 Dim lista2 As List(Of String)



```
lista2 = m.pmtelefono()
For Each Telefono In lista2
    cbTelefono.Items.Add(Telefono)
Next
gbTelefono.Visible = True
If m.especialidad.Equals("-") Then
    rbAuxiliarMedico.Checked = True
    rbMedico.Checked = False
    txtEspecialidad.Visible = False
    txtEspecialidad.Clear()
Else
    rbAuxiliarMedico.Checked = False
    rbMedico.Checked = True
    txtEspecialidad.Visible = True
End If
End If

Case 3
If MsgBox("El médico o auxiliar médico no se encontró,
¿Desearía ingresarlo?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") =
MsgBoxResult.Yes Then
    frmUsuario.Show()
    frmUsuario.cmdIngresarM.Visible = True
    frmUsuario.gbRadio.Visible = True
    Me.Enabled = False
    gbTelefono.Visible = False
    gbBuscar.Enabled = False
    gbDatos.Visible = True
    cmdIngresar.Enabled = True
    cmdEliminar.Enabled = False
    cmdModificar.Enabled = False
```



```
        End If  
    End Select  
  
    Case Else : MsgBox("Hubo errores al buscar el médico o paciente. En  
caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error,  
"S.I.BI.M - GSA")  
  
    End Select  
  
    m = Nothing  
  
End Sub
```

```
Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles  
cmdIngresar.Click  
  
    Dim m As New PersonalMedico  
  
    m.ci = Val(txtCI.Text)  
  
    m.pnombre = txtPnombre.Text  
  
    m.snombre = txtSnombre.Text  
  
    m.papellido = txtPapellido.Text  
  
    m.sapellido = txtSapellido.Text  
  
    m.fechanac = dtpFechaNac.Text  
  
    m.calle = txtCalle.Text  
  
    m.nropuerta = txtNpuerta.Text  
  
    m.esquina = txtEsquina.Text  
  
    m.correo = txtTelefono.Text  
  
    m.especialidad = txtEspecialidad.Text  
  
    m.cn = cn  
  
    If rbAuxiliarMedico.Checked = False And rbMedico.Checked = False Then  
  
        MsgBox("Debe seleccionar el tipo de personal. (Médico o Auxiliar)",  
MessageBoxIcon.Warning, "S.I.BI.M - GSA")  
  
    Else  
  
        If rbMedico.Checked = True Then  
  
            Select Case m.Guardar1(1)
```



Case 0

 MsgBox("Se ha ingresado éxitosamente al médico.",
 MessageBoxIcon.Information, "S.I.BI.M - GSA")

 frmTelefono.Show()

 frmTelefono.cmdAgregar2.Visible = True

 frmTelefono.cmdQuitar2.Visible = True

 Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.")

 Case Else : MsgBox("Hubo errores al ingresar el médico. En caso
de persistir, consulte al adm. del sistema.")

End Select

ElseIf rbAuxiliarMedico.Checked = True Then

 Select Case m.Guardar2(1)

 Case 0

 MsgBox("Se ha ingresado éxitosamente al auxiliar médico.",
 MessageBoxIcon.Information, "S.I.BI.M - GSA")

 Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

 Case Else : MsgBox("Hubo errores al ingresar el auxiliar médico.
En caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error,
 "S.I.BI.M - GSA")

 End Select

End If

End If

m = Nothing

End Sub

Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles
cmdCancelar.Click

 txtPnombre.Clear()

 txtSnombre.Clear()

 txtPapellido.Clear()



```
txtSApellido.Clear()
dtpFechaNac.ResetText()
txtEdad.Clear()
txtCalle.Clear()
txtTelefono.Clear()
txtNpuerta.Clear()
txtEsquina.Clear()
cbTelefono.ResetText()
cbTelefono.Items.Clear()
txtEspecialidad.Clear()
rbAuxiliarMedico.Checked = False
rbMedico.Checked = False
gbBuscar.Enabled = True
gbDatos.Visible = False

End Sub
```

```
Private Sub cmdEliminar_Click(sender As Object, e As EventArgs) Handles cmdEliminar.Click
```

```
    Dim m As New PersonalMedico
    m.ci = Val(txtCI.Text)
    m.cn = cn
    Select Case m.Eliminar()
        Case 0
            MsgBox("Médico o auxiliar eliminado éxitosamente.",_
                MessageBoxButtons.Information, "S.I.BI.M - GSA")
            txtPnombre.Clear()
            txtSnombre.Clear()
            txtPapellido.Clear()
            txtSapellido.Clear()
            dtpFechaNac.ResetText()
```



```
txtEdad.Clear()  
txtCalle.Clear()  
txtTelefono.Clear()  
txtNpuerta.Clear()  
txtEsquina.Clear()  
cbTelefono.ResetText()  
cbTelefono.Items.Clear()  
txtEspecialidad.Clear()  
rbAuxiliarMedico.Checked = False  
rbMedico.Checked = False  
gbBuscar.Enabled = True  
gbDatos.Visible = False
```

Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al eliminar el médico o auxiliar. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

m = Nothing

End Sub

```
Private Sub cmdModificar_Click(sender As Object, e As EventArgs) Handles cmdModificar.Click  
    Dim m As New PersonalMedico  
    m.ci = Val(txtCl.Text)  
    m.pnombre = txtPnombre.Text  
    m.snombre = txtSnombre.Text  
    m.papellido = txtPapellido.Text  
    m.sapellido = txtSapellido.Text  
    m.fechanac = dtpFechaNac.Text
```



```
m.calle = txtCalle.Text  
m.telefono = txtTelefono.Text  
m.nropuerta = txtNpuerta.Text  
m.esquina = txtEsquina.Text  
m.especialidad = txtEspecialidad.Text  
m.cn = cn  
If rbAuxiliarMedico.Checked = False And rbMedico.Checked = False Then  
    MsgBox("Debe seleccionar el tipo de personal. (Médico o Auxiliar)",  
    MessageBoxIcon.Warning, "S.I.BI.M - GSA")  
Else  
    If rbMedico.Checked = True Then  
        Select Case m.Guardar1(0)  
            Case 0  
                MsgBox("Médico modificado con éxito.",  
                MessageBoxIcon.Information, "S.I.BI.M - GSA")  
                Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
                Case Else : MsgBox("Hubo errores al efectuar la modificación del  
médico. En caso de persistir, consulte al adm. del sistema.",  
                MessageBoxIcon.Error, "S.I.BI.M - GSA")  
        End Select  
    Elseif rbAuxiliarMedico.Checked = True Then  
        Select Case m.Guardar2(0)  
            Case 0  
                MsgBox("Auxiliar modificado con éxito.",  
                MessageBoxIcon.Information, "S.I.BI.M - GSA")  
                Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
                Case Else : MsgBox("Hubo errores al efectuar la modificación del  
auxiliar. En caso de persistir, consulte al adm. del sistema.",  
                MessageBoxIcon.Error, "S.I.BI.M - GSA")  
        End Select
```



End If

End If

m = Nothing

End Sub

```
Private Sub rbAuxiliarMedico_CheckedChanged(sender As Object, e As EventArgs) Handles rbAuxiliarMedico.CheckedChanged
```

lblEspecialidad.Visible = False

txtEspecialidad.Visible = False

txtEspecialidad.Clear()

End Sub

```
Private Sub frmAdmPersonalMedico_Leave(sender As Object, e As EventArgs) Handles Me.Leave
```

Me.Close()

End Sub

```
Private Sub frmAdmPersonalMedico_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

dtpFechaNac.Format = DateTimePickerFormat.Custom

dtpFechaNac.CustomFormat = "yyyy/MM/dd HH:mm:ss"

End Sub

```
Private Sub dtpFechaNac_ValueChanged(sender As Object, e As EventArgs) Handles dtpFechaNac.ValueChanged
```

Dim Anios As Integer

Dim Meses As Integer

Dim Dias As Integer

Dim FechaCalculo As Date

FechaCalculo = dtpFechaNac.Value



```
If.dtpFechaNac.Value > Now Then  
    MsgBox("Error, no se permite ingresar una fecha mayor a la fecha  
actual.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
   .dtpFechaNac.ResetText()  
End If  
  
Do While DateAdd("YYYY", 1, FechaCalculo) <= Now  
    Anios = Anios + 1  
    FechaCalculo = DateAdd("YYYY", 1, FechaCalculo)  
Loop  
  
Do While DateAdd("m", 1, FechaCalculo) <= Now  
    Meses = Meses + 1  
    FechaCalculo = DateAdd("m", 1, FechaCalculo)  
Loop  
  
Do While DateAdd("d", 1, FechaCalculo) <= Now  
    Dias = Dias + 1  
    FechaCalculo = DateAdd("d", 1, FechaCalculo)  
Loop  
  
Dim Edad As String = Anios & " años, " & Meses & " meses y " & Dias &  
días"  
txtEdad.Text = Edad  
End Sub  
  
Private Sub cmdAgregar_Click(sender As Object, e As EventArgs) Handles  
cmdAgregar.Click  
    Dim p As New Pacientes
```



```
p.ci = Val(txtCl.Text)  
p.telefono = cbTelefono.Text  
p.cn = cn  
Select Case p.GuardarT(1)
```

Case 0

```
    MsgBox("Teléfono ingresado éxitosamente.",  
    MessageBoxButtons.Information, "S.I.BI.M - GSA")
```

```
    cbTelefono.Items.Clear()  
    Dim lista2 As List(Of String)  
    lista2 = p.patelefono()  
    For Each Telefono In lista2  
        cbTelefono.Items.Add(Telefono)  
    Next
```

Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al ingresar el teléfono. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

p = Nothing

End Sub

```
Private Sub cmdQuitar_Click(sender As Object, e As EventArgs) Handles  
cmdQuitar.Click  
  
    Dim p As New Pacientes  
    p.ci = Val(txtCl.Text)  
    p.telefono = cbTelefono.Text  
    p.cn = cn  
    Select Case p.EliminarT(1)  
        Case 0
```



```
MsgBox("Teléfono eliminado éxitosamente.",  
MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
cbTelefono.Items.Clear()  
  
Dim lista2 As List(Of String)  
  
lista2 = p.patelefono()  
  
For Each Telefono In lista2  
  
    cbTelefono.Items.Add(Telefono)
```

```
Next
```

```
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
Case Else : MsgBox("Hubo errores al eliminar el teléfono. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
p = Nothing
```

```
End Sub
```

```
Private Sub cbTelefono_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles cbTelefono.KeyPress
```

```
    permitirNumerosSE(e)
```

```
End Sub
```

```
End Class
```



frmAnalisisClinico

Public Class frmAnalisisClinico

Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles cmdIngresar.Click

 Dim a As New Analisis

 a.ci = Val(txtBCI.Text)

 a.resultado = txtResultado.Text

 a.tipo = txtTipo.Text

 a.fecha = dtpFecha.Value

 a.cn = cn

 Select Case a.Guardar(1)

 Case 0

 MsgBox("Análisis ingresado con éxito.",
 MessageBoxIcon.Information, "S.I.BI.M - GSA")

 Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
 nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

 Case Else : MsgBox("Hubo errores al ingresar el análisis. En caso de
 persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -
 GSA")

 End Select

 a = Nothing

End Sub

Private Sub frmAnalisisClinico_Leave(sender As Object, e As EventArgs)
Handles Me.Leave

 Me.Close()

End Sub

Private Sub frmAnalisisClinico_Load(sender As Object, e As EventArgs)
Handles MyBase.Load

 dtpFecha.Format = DateTimePickerFormat.Custom



```
dtpFecha.CustomFormat = "yyyy/MM/dd HH:mm:ss"  
End Sub  
  
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click  
    Dim a As New Analisis  
    a.ci = Val(txtBCI.Text)  
    a.cn = cn  
    Select Case a.verifyci2()  
        Case 0 'Encontré  
            If a.estadoborradop = 1 Then  
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",  
                    MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then  
                    Select Case a.estadop()  
                        Case 0  
                            MsgBox("Paciente dado de alta éxitosamente.",  
                                MessageBoxIcon.Information, "S.I.BI.M - GSA")  
                        Case 1  
                            MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
                        Case Else  
                            MsgBox("Hubo errores al realizar el alta del paciente. En  
caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error,  
"S.I.BI.M - GSA")  
                    End Select  
                End If  
            Else  
                Select Case a.comprobarFecha()  
                    Case 0  
                        If MsgBox("¿Desea ingresar los datos de un análisis clínico  
perteneciente a un paciente con C.I " & (txtBCI.Text) & "?",  
                            MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```



```
gbBuscar.Enabled = False  
gbDatos.Enabled = True  
gbDatos.Visible = True  
End If  
  
Case Else  
  
    MsgBox("El paciente no tiene consultas con un médico, para  
    ingresar un análisis clínico es necesario previamente que se atienda con un  
    médico.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
End Select  
  
End If  
  
Case 3 'No encontré  
  
    MsgBox("El paciente no existe, asegurese de que ingreso  
    correctamente la CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
    txtTipo.Clear()  
    txtResultado.Clear()  
    txtTipo.Clear()  
  
Case Else  
  
    MsgBox("Hubo errores al buscar el paciente. En caso de persistir,  
    consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
End Select  
  
a = Nothing  
  
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles  
cmdCancelar.Click  
  
    txtBCI.Clear()  
    dtpFecha.ResetText()  
    txtResultado.Clear()  
    txtTipo.Clear()  
    gbBuscar.Enabled = True  
    gbDatos.Visible = False
```



End Sub

```
Private Sub txtBCI_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles txtBCI.KeyPress
```

```
    permitirNumerosSE(e)
```

End Sub

```
Private Sub dtpFecha_ValueChanged(sender As Object, e As EventArgs)
Handles dtpFecha.ValueChanged
```

```
    If dtpFecha.Value > Now Then
```

```
        MsgBox("Error, no se permite ingresar una fecha mayor a la fecha
actual.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
        dtpFecha.ResetText()
```

```
    End If
```

End Sub

End Class



frmBMAAnalisisClinico

Public Class frmBMAAnalisisClinico

```
Private Sub cboid_SelectedIndexChanged(sender As Object, e  
As EventArgs) Handles cboid.SelectedIndexChanged
```

```
    Dim a As New Analisis
```

```
    a.ci = Val(txtCl.Text)
```

```
    a.tipo = Val(txtTipo.Text)
```

```
    a.resultado = txtResultado.Text
```

```
    a.fecha = dtpFecha.Value
```

```
    a.id = cboid.SelectedItem()
```

```
    a.cn = cn
```

```
    Select Case a.Buscar()
```

```
        Case 0 'Encontré
```

```
            txtCl.Text = a.ci
```

```
            txtTipo.Text = a.tipo
```

```
            dtpFecha.Value = a.fecha
```

```
            txtResultado.Text = a.resultado
```

```
            cmdEliminar.Enabled = True
```

```
            cmdModificar.Enabled = True
```

```
        Case 3 'No encontré
```

```
            txtCl.Clear()
```

```
            txtResultado.Clear()
```

```
            txtTipo.Clear()
```

```
        Case Else : MsgBox("Hubo errores al buscar el análisis  
clínico. En caso de persistir, consulte al adm. del sistema.",  
MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
    End Select
```



```
a = Nothing
```

```
End Sub
```

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs)  
Handles cmdBuscar.Click
```

```
    Dim a As New Analisis
```

```
    a.ci = Val(txtCl.Text)
```

```
    a.cn = cn
```

```
    Dim lista As List(Of Integer)
```

```
    lista = a.idanalisis()
```

```
    Select Case a.verifyci()
```

```
        Case 0 'Encontré
```

```
            If a.estadoborradop = 1 Then
```

```
                If MsgBox("El paciente se dió de baja, ¿Desaea darle  
de alta?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") =  
                    MsgBoxResult.Yes Then
```

```
                    Select Case a.estadop()
```

```
                        Case 0
```

```
                            MsgBox("El alta del paciente se realizó  
éxitosamente.", MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                        Case 1
```

```
                            MsgBox("Se ha perdido la sesión. Debe  
loguearse nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                        Case Else
```

```
                            MsgBox("Hubo errores al realizar el alta del  
paciente. En caso de persistir, consulte al adm. del sistema.",  
                    MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                End Select
```



```
End If  
Else  
    For Each ida In lista  
        cbolD.Items.Add(ida)  
    Next  
    gbBuscar.Enabled = False  
    gbDatos.Enabled = True  
    gbDatos.Visible = True  
End If  
  
Case 3 'No encontré  
    MsgBox("El paciente no existe, o no tiene análisis clínicos existentes, asegurese de que ingreso correctamente la CI.",  
    MessageBoxButtons.Error, "S.I.BI.M - GSA")  
    cbolD.Items.Clear()  
    txtCI.Clear()  
    txtResultado.Clear()  
    txtTipo.Clear()  
  
    Case Else : MsgBox("Hubo errores al encontrar al paciente.  
    En caso de persistir, consulte al adm. del sistema.",  
    MessageBoxButtons.Error, "S.I.BI.M - GSA")  
End Select  
a = Nothing  
End Sub
```

```
Private Sub cmdModificar_Click(sender As Object, e As  
EventArgs) Handles cmdModificar.Click
```

```
    Dim a As New Analisis  
    a.id = cbolD.SelectedItem()
```



```
a.fecha = dtpFecha.Text  
a.resultado = txtResultado.Text  
a.tipo = txtTipo.Text  
a.cn = cn  
Select Case a.Guardar(0)
```

```
Case 0
```

```
    MsgBox("Análisis clínico modificado con éxito.",  
    MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
Case Else : MsgBox("Hubo errores al realizar la  
modificación. En caso de persistir, consulte al adm. del sistema.",  
MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
End Select
```

```
a = Nothing
```

```
End Sub
```

```
Private Sub cmdCancelar_Click_1(sender As Object, e As  
EventArgs) Handles cmdCancelar.Click
```

```
    cbolD.Items.Clear()  
    cbolD.ResetText()  
    txtCl.Clear()  
    txtTipo.Clear()  
    dtpFecha.ResetText()  
    txtResultado.Clear()  
    cmdEliminar.Enabled = False  
    cmdModificar.Enabled = False  
    gbBuscar.Enabled = True
```



```
    gbDatos.Visible = False  
End Sub
```

```
Private Sub cmdEliminar_Click(sender As Object, e As  
EventArgs) Handles cmdEliminar.Click
```

```
    Dim a As New Analisis  
    a.id = Val(cbID.SelectedItem)  
    a.ci = Val(txtCI.Text)  
    a.cn = cn  
    Select Case a.Eliminar()
```

```
        Case 0  
            cbID.Items.Clear()  
            cbID.ResetText()  
            txtTipo.Clear()  
            dtpFecha.ResetText()  
            txtResultado.Clear()  
            Dim lista As List(Of Integer)  
            lista = a.idanalisis()  
            For Each ida In lista  
                cbID.Items.Add(ida)  
            Next
```

```
        MsgBox("Analisis clínico eliminado exitosamente.",  
        MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
        Case Else : MsgBox("Hubo errores al eliminar el análisis  
clínico. En caso de persistir, consulte al adm. del sistema.",  
        MessageBoxIcon.Error, "S.I.BI.M - GSA")
```



End Select

a = Nothing

End Sub

Private Sub frmBMAnalisisClinico_Leave(sender As Object, e As EventArgs) Handles Me.Leave

 Me.Close()

End Sub

Private Sub frmBMAnalisisClinico_Load(sender As Object, e As EventArgs) Handles MyBase.Load

 dtpFecha.Format = DateTimePickerFormat.Custom

 dtpFecha.CustomFormat = "yyyy/MM/dd HH:mm:ss"

End Sub

Private Sub txtCI_KeyPress(sender As Object, e As KeyPressEventArgs) Handles txtCI.KeyPress

 permitirNumerosSE(e)

End Sub

Private Sub dtpFecha_ValueChanged(sender As Object, e As EventArgs) Handles dtpFecha.ValueChanged

 If dtpFecha.Value > Now Then

 MsgBox("Error, no se permite ingresar una fecha mayor a la fecha actual.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

 dtpFecha.ResetText()

 End If

End Sub

End Class



FrmBMDiagnostico

Public Class frmBMDiagnostico

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim d As New Diagnosticos
```

```
    d.ci = Val(txtCI.Text)
```

```
    d.cn = cn
```

```
    Dim lista As List(Of Integer)
```

```
    lista = d.iddiag()
```

```
    Select Case d.verifyci()
```

```
        Case 0 'Encontré
```

```
            If d.estadoborradop = 1 Then
```

```
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",  
                MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                    Select Case d.estadop()
```

```
                        Case 0
```

```
                            MsgBox("El alta del paciente se realizó éxitosamente.",  
                            MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                        Case 1
```

```
                            MsgBox("Se ha perdido la sesión. Debe loguearse  
                            nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                        Case Else
```

```
                            MsgBox("Hubo errores al realizar el alta del paciente. En  
                            caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error,  
                            "S.I.BI.M - GSA")
```

```
                    End Select
```

```
                End If
```

```
            Else
```

```
                For Each ida In lista
```

```
                    cbolD.Items.Add(ida)
```



Next

gbBuscar.Enabled = False

gbDatos.Enabled = True

gbDatos.Visible = True

End If

Case 3 'No encontré

MsgBox("El paciente no existe, o no tiene diagnosticos existentes, asegurese de que ingreso correctamente la CI.",
MessageBoxIcon.Error, "S.I.BI.M - GSA")

cboID.Items.Clear()

txtCI.Clear()

txtDescripcion.Clear()

cboTipo.Items.Clear()

End Select

d = Nothing

End Sub

Private Sub cmdModificar_Click(sender As Object, e As EventArgs) Handles cmdModificar.Click

Dim d As New Diagnosticos

d.ci = Val(txtCI.Text)

d.id = Val(cboID.Text)

d.fecha =.dtpFecha.Text

d.tipo = cboTipo.SelectedItem()

d.descripcion = txtDescripcion.Text

d.cn = cn

Select Case d.Guardar(0)

Case 0

MsgBox("Diagnóstico modificado éxitosamente.",
MessageBoxIcon.Information, "S.I.BI.M - GSA")



Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al modificar el diagnóstico. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

d = Nothing

End Sub

```
Private Sub frmBMDiagnostico_Leave(sender As Object, e As EventArgs)
Handles Me.Leave
```

Me.Close()

End Sub

```
Private Sub frmBMDiagnostico_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
```

dtpFecha.Format = DateTimePickerFormat.Custom

dtpFecha.CustomFormat = "yyyy/MM/dd HH:mm:ss"

Dim d As New Diagnosticos

d.cn = cn

Dim lista As List(Of String)

lista = d.tipodiag()

For Each Diagnostico In lista

cboTipo.Items.Add(Diagnostico)

Next

End Sub

```
Private Sub cboID_SelectedIndexChanged_1(sender As Object, e As EventArgs)
Handles cboID.SelectedIndexChanged
```

Dim d As New Diagnosticos

d.ci = Val(txtCI.Text)



```
d.id = Val(cboid.Text)  
d.cn = cn  
Select Case d.Buscar()  
    Case 0 'Encontré  
        txtCl.Text = d.ci  
        cboTipo.SelectedItem = d.tipo  
        cboTipo.Text = d.nombreD  
        dtpFecha.Value = d.fecha  
        txtDescripcion.Text = d.descripcion  
        cmdEliminar.Enabled = True  
        cmdModificar.Enabled = True  
  
    Case 3 'No encontré  
        txtCl.Clear()  
        cboid.ResetText()  
        txtDescripcion.Clear()  
  
    Case Else : MsgBox("Hubo errores al buscar el diagnóstico. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
End Select  
d = Nothing  
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click  
    cboid.Items.Clear()  
    cboid.ResetText()  
    cboTipo.ResetText()  
    txtCl.Clear()  
    dtpFecha.ResetText()  
    txtDescripcion.Clear()
```



```
cmdEliminar.Enabled = False  
cmdModificar.Enabled = False  
gbBuscar.Enabled = True  
gbDatos.Visible = False  
End Sub
```

```
Private Sub cmdEliminar_Click(sender As Object, e As EventArgs) Handles cmdEliminar.Click  
    Dim d As New Diagnosticos  
    d.id = Val(cboid.SelectedItem)  
    d.ci = Val(txtCl.Text)  
    d.cn = cn  
    Select Case d.Eliminar()  
        Case 0  
            cboid.Items.Clear()  
            cboid.ResetText()  
            cboTipo.ResetText()  
            txtDescripcion.Clear()  
            dtpFecha.ResetText()  
            Dim lista As List(Of Integer)  
            lista = d.iddiag()  
            For Each ida In lista  
                cboid.Items.Add(ida)  
            Next  
            MsgBox("Diagnóstico eliminado exitosamente.",  
                MessageBoxButtons.Information, "S.I.BI.M - GSA")  
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
        Case Else : MsgBox("Hubo errores al eliminar el diagnóstico. En caso  
de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -  
GSA")
```



End Select

d = Nothing

End Sub

```
Private Sub txtCI_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles txtCI.KeyPress
```

permitirNumerosSE(e)

End Sub

```
Private Sub.dtpFecha_ValueChanged(sender As Object, e As EventArgs)
Handles.dtpFecha.ValueChanged
```

If dtpFecha.Value > Now Then

MsgBox("Error, no se permite ingresar una fecha mayor a la fecha actual.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

dtpFecha.ResetText()

End If

End Sub

End Class



frmBMTratamiento

Public Class frmBMTratamiento

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim t As New Tratamientos
```

```
    t.ci = Val(txtCI.Text)
```

```
    t.cn = cn
```

```
    Dim lista As List(Of Integer)
```

```
    lista = t.idtratamiento()
```

```
    Select Case t.verifyci()
```

```
        Case 0 'Encontré
```

```
            If t.estadoborradop = 1 Then
```

```
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",  
                MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                    Select Case t.estadop()
```

```
                        Case 0
```

```
                            MsgBox("El alta del paciente se realizó éxitosamente.",  
                            MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                        Case 1
```

```
                            MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                        Case Else
```

```
                            MsgBox("Hubo errores al realizar el alta. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
                    End Select
```

```
                End If
```

```
            Else
```

```
                For Each ida In lista
```

```
                    cbolD.Items.Add(ida)
```



Next

gbBuscar.Enabled = False

gbDatos.Enabled = True

gbDatos.Visible = True

End If

Case 3 'No encontré

MsgBox("El paciente no existe, o no tiene tratamientos existentes, asegurese de que ingreso correctamente la CI.",
MessageBoxIcon.Error, "S.I.BI.M - GSA")

txtCI.Clear()

txtDescripcion.Clear()

txtNombre.Clear()

dtpFinal.ResetText()

dtpInicio.ResetText()

End Select

t = Nothing

End Sub

Private Sub cboID_SelectedIndexChanged(sender As Object, e As EventArgs) Handles cboID.SelectedIndexChanged

Dim t As New Tratamientos

t.ci = Val(txtCI.Text)

t.descripcion = txtDescripcion.Text

t.nombre = txtNombre.Text

t.fechafinal = dtpFinal.Value

t.fechainicio = dtpInicio.Value

t.id = cboID.SelectedItem()

t.cn = cn

Select Case t.Buscar()

Case 0 'Encontré



```
txtCl.Text = t.ci  
txtNombre.Text = t.nombre  
txtDescripcion.Text = t.descripcion  
dtpFinal.Value = t.fechafinal  
dtpInicio.Value = t.fechainicio  
cmdEliminar.Enabled = True  
cmdModificar.Enabled = True
```

Case 3 'No encontré

```
txtCl.Clear()  
txtNombre.Clear()  
txtDescripcion.Clear()  
dtpFinal.ResetText()  
dtpInicio.ResetText()
```

End Select

t = Nothing

End Sub

```
Private Sub cmdEliminar_Click(sender As Object, e As EventArgs) Handles cmdEliminar.Click  
    Dim t As New Tratamientos  
    t.id = Val(cboid.SelectedItem)  
    t.ci = Val(txtCl.Text)  
    t.cn = cn  
    Select Case t.Eliminar()  
        Case 0  
            cboid.Items.Clear()  
            cboid.ResetText()  
            txtNombre.Clear()  
            txtDescripcion.Clear()
```



```
dtpFinal.ResetText()
dtpInicio.ResetText()
Dim lista As List(Of Integer)
lista = t.idtratamiento()
For Each ida In lista
    cboid.Items.Add(ida)
Next
MsgBox("Tratamiento eliminado éxitosamente.",
MessageBoxIcon.Information, "S.I.BI.M - GSA")
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
Case Else : MsgBox("Hubo errores al eliminar el tratamiento. En caso
de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -
GSA")
End Select

t = Nothing
End Sub

Private Sub cmdModificar_Click(sender As Object, e As EventArgs) Handles
cmdModificar.Click
    Dim t As New Tratamientos
    t.id = cboid.SelectedItem()
    t.descripcion = txtDescripcion.Text
    t.nombre = txtNombre.Text
    t.fechafinal = dtpFinal.Value
    t.fechainicio = dtpInicio.Value
    t.cn = cn
    Select Case t.Guardar(0)
        Case 0
            MsgBox("Tratamiento modificado éxitosamente.",
MessageBoxIcon.Information, "S.I.BI.M - GSA")
```



Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al modificar el tratamiento. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

t = Nothing

End Sub

Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click

cbolD.Items.Clear()

cbolD.ResetText()

txtCl.Clear()

txtNombre.Clear()

txtDescripcion.Clear()

dtpFinal.ResetText()

dtpInicio.ResetText()

cmdEliminar.Enabled = False

cmdModificar.Enabled = False

gbBuscar.Enabled = True

gbDatos.Visible = False

End Sub

Private Sub frmBMTratamiento_Leave(sender As Object, e As EventArgs) Handles Me.Leave

Me.Close()

End Sub

Private Sub frmBMTratamiento_Load(sender As Object, e As EventArgs) Handles MyBase.Load



```
dtpFinal.Format = DateTimePickerFormat.Custom  
dtpFinal.CustomFormat = "yyyy/MM/dd HH:mm:ss"  
dtpInicio.Format = DateTimePickerFormat.Custom  
dtpInicio.CustomFormat = "yyyy/MM/dd HH:mm:ss"  
End Sub
```

```
Private Sub txtCI_KeyPress(sender As Object, e As KeyPressEventArgs)  
Handles txtCI.KeyPress  
    permitirNumerosSE(e)  
End Sub
```

```
Private Sub dtpInicio_ValueChanged(sender As Object, e As EventArgs)  
Handles dtpInicio.ValueChanged  
    If dtpInicio.Value > Now Then  
        MsgBox("Error, no se permite ingresar una fecha mayor a la fecha  
actual.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
        dtpInicio.ResetText()  
    End If  
End Sub
```

```
Private Sub dtpFinal_ValueChanged(sender As Object, e As EventArgs)  
Handles dtpFinal.ValueChanged  
    If dtpFinal.Value < dtpInicio.Value Then  
        MsgBox("Error, no se permite ingresar una fecha final que sea menor a  
la fecha inicial.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
        dtpFinal.ResetText()  
    End If  
    If dtpFinal.Value = ("31/12/9998") Then  
        cbFechaFinal.Checked = True  
    Else  
        cbFechaFinal.Checked = False  
    End If
```



```
End If  
End Sub  
  
Private Sub cbFechaFinal_CheckedChanged(sender As Object, e As  
EventArgs) Handles cbFechaFinal.CheckedChanged  
    If cbFechaFinal.Checked = True Then  
        dtpFinal.Value = "9998-12-31 00:00:00"  
        dtpFinal.Enabled = False  
        dtpFinal.Visible = False  
        txtNula.Visible = True  
    Else  
        txtNula.Visible = False  
        dtpFinal.ResetText()  
        dtpFinal.Visible = True  
        dtpFinal.Enabled = True  
    End If  
End Sub  
End Class
```



frmConsultaAnalisis

Public Class frmConsultaAnalisis

```
Private Sub txtCI_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtCI.KeyPress
    permitirNumerosSE(e)
End Sub
```

```
Private Sub frmConsultaAnalisis_Leave(sender As Object, e As EventArgs)
Handles Me.Leave
    Me.Close()
End Sub
```

```
Private Sub frmConsultaAnalisis_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
    dtpFecha.Format = DateTimePickerFormat.Custom
    dtpFecha.CustomFormat = "yyyy/MM/dd HH:mm:ss"
End Sub
```

```
Private Sub cboid_SelectedIndexChanged(sender As Object, e As EventArgs) Handles cboid.SelectedIndexChanged
    Dim a As New Analisis
    a.ci = Val(txtCI.Text)
    a.tipo = txtTipo.Text
    a.resultado = txtResultado.Text
    a.fecha = dtpFecha.Value
    a.id = cboid.SelectedItem()
    a.cn = cn
    Select Case a.Buscar()
        Case 0 'Encontré
```



```
txtCl.Text = a.ci  
txtTipo.Text = a.tipo  
dtpFecha.Value = a.fecha  
txtResultado.Text = a.resultado  
  
Case 3 'No encontré  
    txtCl.Clear()  
    txtResultado.Clear()  
    txtTipo.Clear()  
  
End Select  
a = Nothing  
End Sub
```

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click  
    Dim a As New Analisis  
    a.ci = Val(txtCl.Text)  
    a.cn = cn  
    Dim lista As List(Of Integer)  
    lista = a.idanalisis()  
    Select Case a.verifyci()  
        Case 0 'Encontré  
            If a.estadoborradop = 1 Then  
                MsgBox("No puede consultar la siguiente información debido a que usted está dado de baja, para que se realice el alta de vuelta debe de contactar con la institución médica.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
            Else  
                Select Case a.getci()  
                    Case 0  
                        If a.ci = Val(txtCl.Text) Then  
                            For Each ida In lista
```



```
    cbID.Items.Add(ida)
    Next
    gbBuscar.Enabled = False
    gbDatos.Enabled = True
    gbDatos.Visible = True
Else
    MsgBox("Solo puedes buscar registros de tu propiedad, por
favor ingresa tu CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
End If
Case Else : MsgBox("Solo puedes buscar registros de tu
propiedad, por favor ingresa tu CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
End Select
End If
Case 3 'No encontré
    MsgBox("El paciente no existe, o no tiene análisis clínicos
existentes, asegúrese de que ingreso correctamente la CI.",
MessageBoxIcon.Error, "S.I.BI.M - GSA")
    cbID.Items.Clear()
    txtCI.Clear()
    txtResultado.Clear()
    txtTipo.Clear()
Case Else
    MsgBox("Hubo error al efectuar la búsqueda. En caso de persistir,
consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
End Select
a = Nothing
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles
cmdCancelar.Click
    cbID.Items.Clear()
```



```
cbolD.ResetText()  
txtCl.Clear()  
txtTipo.Clear()  
dtpFecha.ResetText()  
txtResultado.Clear()  
gbBuscar.Enabled = True  
gbDatos.Visible = False  
End Sub
```

```
Private Sub dtpFecha_ValueChanged(sender As Object, e As EventArgs)  
Handles dtpFecha.ValueChanged  
If dtpFecha.Value > Now Then  
    MsgBox("Error, no se permite ingresar una fecha mayor a la fecha  
actual.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
    dtpFecha.ResetText()  
End If  
End Sub  
End Class
```



frmConsultaDiagnostico

Public Class frmConsultaDiagnostico

```
Private Sub txtCI_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtCI.KeyPress
    permitirNumerosSE(e)
End Sub
```

```
Private Sub cmdBuscarCI_Click(sender As Object, e As EventArgs) Handles cmdBuscarCI.Click
```

```
    Dim d As New Diagnosticos
```

```
    d.ci = Val(txtCI.Text)
```

```
    d.cn = cn
```

```
    Dim lista As List(Of Integer)
```

```
    lista = d.iddiag()
```

```
    Select Case d.verifyci()
```

```
        Case 0 'Encontré
```

```
            If d.estadoborradop = 1 Then
```

```
                MsgBox("No puede consultar la siguiente información debido a que  
usted está dado de baja, para que se realice el alta de vuelta debe de contactar  
con la institución médica.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
        Else
```

```
            Select Case d.getci()
```

```
                Case 0
```

```
                    If d.ci = Val(txtCI.Text) Then
```

```
                        For Each ida In lista
```

```
                            cbold.Items.Add(ida)
```

```
                        Next
```

```
                        gbBuscar.Enabled = False
```

```
                        gbDatos.Enabled = True
```



```
gbDatos.Visible = True  
Else  
    MsgBox("Solo puedes buscar registros de tu propiedad, por favor ingresa tu CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
End If  
  
Case Else : MsgBox("Solo puedes buscar registros de tu propiedad, por favor ingresa tu CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
End Select  
  
End If  
  
Case 3 'No encontré  
  
    MsgBox("El paciente no existe, o no tiene un diagnostico existente, asegurese de que ingreso correctamente la CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
    cbolD.Items.Clear()  
    txtCI.Clear()  
    txtDescripcion.Clear()  
    txtNombreD.Clear()  
  
End Select  
  
d = Nothing  
  
End Sub  
  
Private Sub cbolD_SelectedIndexChanged(sender As Object, e As EventArgs) Handles cbolD.SelectedIndexChanged  
  
    Dim d As New Diagnosticos  
    d.ci = Val(txtCI.Text)  
    d.id = Val(cbolD.Text)  
    d.cn = cn  
  
    Select Case d.Buscar()  
  
        Case 0 'Encontré  
            txtCI.Text = d.ci  
            txtNombreD.Text = d.nombreD  
            dtpFecha.Value = d.fecha  
    End Select  
End Sub
```



```
txtDescripcion.Text = d.descripcion
```

```
Case 3 'No encontré
```

```
    txtCl.Clear()
```

```
    txtNombreD.Clear()
```

```
    txtDescripcion.Clear()
```

```
Case Else : MsgBox("Hubo errores al efectuar operación. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
End Select
```

```
d = Nothing
```

```
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click
```

```
    cboID.Items.Clear()
```

```
    cboID.ResetText()
```

```
    txtCl.Clear()
```

```
    dtpFecha.ResetText()
```

```
    txtDescripcion.Clear()
```

```
    gbBuscar.Enabled = True
```

```
    gbDatos.Visible = False
```

```
End Sub
```

```
Private Sub frmConsultaDiagnostico_Leave(sender As Object, e As EventArgs) Handles Me.Leave
```

```
    Me.Close()
```

```
End Sub
```

```
Private Sub frmConsultaDiagnostico_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    dtpFecha.Format = DateTimePickerFormat.Custom
```



```
dtpFecha.CustomFormat = "yyyy/MM/dd HH:mm:ss"  
End Sub  
  
Private Sub.dtpFecha_ValueChanged(sender As Object, e As EventArgs)  
Handles.dtpFecha.ValueChanged  
If.dtpFecha.Value > Now Then  
    MsgBox("Error, no se permite ingresar una fecha mayor a la fecha  
actual.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
   .dtpFecha.ResetText()  
End If  
End Sub  
  
End Class
```



frmConsultaEntInicial

```
Public Class frmConsultaEntInicial
```

```
    Private Sub txtBCI_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtCI.KeyPress
        permitirNumerosSE(e)
    End Sub
```

```
    Private Sub cbFecha_SelectedIndexChanged(sender As Object, e As EventArgs) Handles cbFecha.SelectedIndexChanged
```

```
        Dim i As New EntrevistaInicial
```

```
        i.ci = Val(txtCI.Text)
```

```
        i.fechacb = cbFecha.SelectedItem
```

```
        i.antecedentes = txtAntecedentes.Text
```

```
        i.enfermedad = txtEnfermedadActual.Text
```

```
        i.enfermedadf = txtEnfermedadFamiliar.Text
```

```
        i.estadoc = txtEstadoConciencia.Text
```

```
        i.frecuenciac = txtFrecuenciaCardiaca.Text
```

```
        i.frecuenciar = txtFrecuenciaResp.Text
```

```
        i.gradoc = txtGradoDeColab.Text
```

```
        i.gradoh = txtGradoDeHidratacion.Text
```

```
        i.motivoconsulta = txtMotivo.Text
```

```
        i.movilidad = txtMovilidad.Text
```

```
        i.nombreacompanante = txtNombreAcompanante.Text
```

```
        i.otro = txtOtro.Text
```

```
        i.temperatura = Val(txtTemperatura.Text)
```

```
        i.cimedico = Val(txtCIMedico.Text)
```

```
        i.pulso = txtPulso.Text
```

```
        i.actitud = txtActitud.Text
```

```
        i.estadon = txtEstadoNutricion.Text
```

```
        i.cn = cn
```



Select Case i.Buscar()

Case 0 'Encontré

txtAntecedentes.Text = i.antecedentes

txtEnfermedadActual.Text = i.enfermedad

txtTipoEnfActual.Text = i.tipoenfactual

txtTipoEnfAnt.Text = i.tipoenfanterior

txtEnfermedadFamiliar.Text = i.enfermedadf

txtEstadoConciencia.Text = i.estadoc

txtFrecuenciaCardiaca.Text = i.frecuenciac

txtFrecuenciaResp.Text = i.frecuenciar

txtGradoDeColab.Text = i.gradoc

txtGradoDeHidratacion.Text = i.gradoh

txtMotivo.Text = i.motivoconsulta

txtMovilidad.Text = i.movilidad

txtNombreAcompanante.Text = i.nombreacompañante

txtTemperatura.Text = i.temperatura

txtCIMedico.Text = i.cimedico

txtPulso.Text = i.pulso

txtActitud.Text = i.actitud

txtEstadoNutricion.Text = i.estadon

If i.fiebre = 1 Then

 CheckBoxFiebre.Checked = True

Else

 CheckBoxFiebre.Checked = False

End If

If i.malestar = 1 Then

 CheckBoxMalestar.Checked = True

Else

 CheckBoxMalestar.Checked = False



```
End If

If i.dolor = 1 Then
    CheckBoxDolor.Checked = True
Else
    CheckBoxDolor.Checked = False
End If

If i.otro = "" Then
    CheckBoxOtroS.Checked = False
    txtOtro.Clear()
Else
    CheckBoxOtroS.Checked = True
    txtOtro.Text = i.otro
End If

gbBuscar.Enabled = False
gbDatos.Enabled = True
gbDatos.Visible = True

Case Else : MsgBox("Hubo errores al efectuar operación. En caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

End Select

i = Nothing

End Sub

Private Sub CheckBoxOtroS_CheckedChanged(sender As Object, e As EventArgs) Handles CheckBoxOtroS.CheckedChanged
    If CheckBoxOtroS.Checked = True Then
        txtOtro.Visible = True
    Else
        txtOtro.Clear()
        txtOtro.Visible = False
    End If

```



End Sub

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim i As New EntrevistaInicial
```

```
    i.ci = Val(txtCI.Text)
```

```
    i.cn = cn
```

```
    Select Case i.verifyci2()
```

```
        Case 0 'Encontré
```

```
            If i.estadoborradoP = 1 Then
```

```
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",  
                MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                    Select Case i.estadop()
```

```
                        Case 0
```

```
                            MsgBox("Operación efectuada con éxito.",  
                            MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                        Case 1
```

```
                            MsgBox("Se ha perdido la sesión. Debe loguearse  
                            nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                        Case Else
```

```
                            MsgBox("Hubo errores al efectuar operación. En caso de  
                            persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
                            GSA")
```

```
                    End Select
```

```
                End If
```

```
            Else
```

```
                Dim lista As List(Of String)
```

```
                lista = i.fechafunc()
```

```
                For Each Fecha In lista
```

```
                    cbFecha.Items.Add(Fecha)
```

```
                Next
```



```
    gbBuscar.Enabled = False  
    gbDatos.Enabled = True  
    gbDatos.Visible = True  
  
End If  
  
Case 3 'No encontré  
  
    MsgBox("El paciente no existe, o no se le ha realizado ninguna entrevista, asegurese de que ingreso correctamente la CI.",  
    MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
    txtCI.Clear()  
  
End Select  
  
i = Nothing  
  
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click  
  
    txtCI.Clear()  
    txtCIMedico.Clear()  
    txtEstadoConciencia.Clear()  
    txtGradoDeColab.Clear()  
    cbFecha.ResetText()  
    txtMovilidad.Clear()  
    txtPulso.Clear()  
    txtActitud.Clear()  
    txtEstadoNutricion.Clear()  
    txtGradoDeHidratacion.Clear()  
    txtEstadoConciencia.Clear()  
    txtMotivo.Clear()  
    txtFrecuenciaResp.Clear()  
    txtTemperatura.Clear()  
    txtFrecuenciaCardiaca.Clear()
```



```
txtEnfermedadActual.Clear()  
txtNombreAcompañante.Clear()  
txtEnfermedadFamiliar.Clear()  
txtAntecedentes.Clear()  
txtTipoEnfActual.Clear()  
txtTipoEnfAnt.Clear()  
txtOtro.Clear()  
CheckBoxDolor.Checked = False  
CheckBoxFiebre.Checked = False  
CheckBoxOtroS.Checked = False  
CheckBoxMalestar.Checked = False  
gbBuscar.Enabled = True  
gbDatos.Visible = False  
cbFecha.Items.Clear()
```

```
End Sub
```

```
Private Sub frmConsultaEntInicial_Leave(sender As Object, e As EventArgs)  
Handles Me.Leave
```

```
    Me.Close()  
End Sub
```

```
Private Sub CheckBoxDolor_CheckedChanged(sender As Object, e As EventArgs) Handles CheckBoxDolor.CheckedChanged
```

```
    If CheckBoxDolor.Checked = True Then
```

```
        cmdConsultar.Visible = True
```

```
    Else
```

```
        cmdConsultar.Visible = False
```

```
    End If
```

```
End Sub
```



```
Private Sub cmdConsultar_Click(sender As Object, e As EventArgs) Handles cmdConsultar.Click
    Dim i As New EntrevistaInicial
    i.ci = Val(txtCI.Text)
    i.fecha = cbFecha.SelectedItem
    i.cn = cn
    Select Case i.buscardolor()
        Case 0 'Encontré
            frmPartesCuerpo.Show()
            frmPartesCuerpo.cmdIngresar.Enabled = False
            If i.boca = 1 Then
                frmPartesCuerpo.cbBoca.Checked = True
            Else
                frmPartesCuerpo.cbBoca.Checked = False
            End If
            If i.brazo = 1 Then
                frmPartesCuerpo.cbBrazo.Checked = True
            Else
                frmPartesCuerpo.cbBrazo.Checked = False
            End If
            If i.cabeza = 1 Then
                frmPartesCuerpo.cbCabeza.Checked = True
            Else
                frmPartesCuerpo.cbCabeza.Checked = False
            End If
            If i.cuello = 1 Then
                frmPartesCuerpo.cbCuello.Checked = True
            Else
                frmPartesCuerpo.cbCuello.Checked = False
            End If
        End Case
    End Select
End Sub
```



```
End If  
If i.dedo = 1 Then  
    frmPartesCuerpo.cbDedo.Checked = True  
Else  
    frmPartesCuerpo.cbDedo.Checked = False  
End If  
If i.espalda = 1 Then  
    frmPartesCuerpo.cbEspalda.Checked = True  
Else  
    frmPartesCuerpo.cbEspalda.Checked = False  
End If  
If i.estomago = 1 Then  
    frmPartesCuerpo.cbEstomago.Checked = True  
Else  
    frmPartesCuerpo.cbEstomago.Checked = False  
End If  
If i.frente = 1 Then  
    frmPartesCuerpo.cbFrente.Checked = True  
Else  
    frmPartesCuerpo.cbFrente.Checked = False  
End If  
If i.garganta = 1 Then  
    frmPartesCuerpo.cbGarganta.Checked = True  
Else  
    frmPartesCuerpo.cbGarganta.Checked = False  
End If  
If i.hombro = 1 Then  
    frmPartesCuerpo.cbHombro.Checked = True  
Else
```



```
frmPartesCuerpo.cbHombro.Checked = False  
End If  
If i.mano = 1 Then  
    frmPartesCuerpo.cbMano.Checked = True  
Else  
    frmPartesCuerpo.cbMano.Checked = False  
End If  
If i.muslo = 1 Then  
    frmPartesCuerpo.cbMuslo.Checked = True  
Else  
    frmPartesCuerpo.cbMuslo.Checked = False  
End If  
If i.munieca = 1 Then  
    frmPartesCuerpo.cbMunieca.Checked = True  
Else  
    frmPartesCuerpo.cbMunieca.Checked = False  
End If  
If i.nariz = 1 Then  
    frmPartesCuerpo.cbNariz.Checked = True  
Else  
    frmPartesCuerpo.cbNariz.Checked = False  
End If  
If i.ojo = 1 Then  
    frmPartesCuerpo.cbOjo.Checked = True  
Else  
    frmPartesCuerpo.cbOjo.Checked = False  
End If  
If i.oreja = 1 Then  
    frmPartesCuerpo.cbOreja.Checked = True
```



```
Else
    frmPartesCuerpo.cbOreja.Checked = False
End If

If i.pantorrilla = 1 Then
    frmPartesCuerpo.cbPantorrilla.Checked = True
Else
    frmPartesCuerpo.cbPantorrilla.Checked = False
End If

If i.pecho = 1 Then
    frmPartesCuerpo.cbPecho.Checked = True
Else
    frmPartesCuerpo.cbPecho.Checked = False
End If

If i.pie = 1 Then
    frmPartesCuerpo.cbPie.Checked = True
Else
    frmPartesCuerpo.cbPie.Checked = False
End If

If i.pierna = 1 Then
    frmPartesCuerpo.cbPierna.Checked = True
Else
    frmPartesCuerpo.cbPierna.Checked = False
End If

If i.rodilla = 1 Then
    frmPartesCuerpo.cbRodilla.Checked = True
Else
    frmPartesCuerpo.cbRodilla.Checked = False
End If

If i.tobillo = 1 Then
```



```
frmPartesCuerpo.cbTobillo.Checked = True
Else
    frmPartesCuerpo.cbTobillo.Checked = False
End If
If i.trasero = 1 Then
    frmPartesCuerpo.cbTrasero.Checked = True
Else
    frmPartesCuerpo.cbTrasero.Checked = False
End If
Case Else 'No encontré
    MsgBox("Hubo errores al consultar las partes del cuerpo. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
End Select
End Sub

End Class
```



FrmConsultar Pacientes

Public Class frmConsultarPacientes

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click
```

```
    txtPnombre.Clear()  
    txtSnombre.Clear()  
    txtPapellido.Clear()  
    txtSapellido.Clear()  
    cbTelefono.ResetText()  
    dtpFechaNac.ResetText()  
    txtEdad.Clear()  
    txtCalle.Clear()  
    txtEcivil.Clear()  
    txtSexo.Clear()  
    cbTelefono.Items.Clear()  
    txtCorreo.Clear()  
    txtNpuerta.Clear()  
    txtEsquina.Clear()  
    gbBuscar.Enabled = True  
    gbDatos.Visible = False
```

```
End Sub
```

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim p As New Pacientes  
    p.ci = Val(txtBCI.Text)  
    p.cn = cn  
    Select Case p.Buscar()
```



Case 0 'Encontré

If p.estadoborradop = 1 Then

If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",
MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then

Select Case p.estadop()

Case 0

 MsgBox("El alta del paciente se realizó éxitosamente",
 MessageBoxIcon.Information, "S.I.BI.M - GSA")

Case 1

 MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

Case Else

 MsgBox("Hubo errores al realizar el alta. En caso de
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -
GSA")

End Select

End If

Else

 txtPnombre.Text = p.pnombre

 txtSnombre.Text = p.snombre

 txtPapellido.Text = p.papellido

 txtSapellido.Text = p.sapellido

 dtpFechaNac.Text = p.fechanac

 txtEdad.Text = p.edad

 txtEcivil.Text = p.estadocivil

 txtSexo.Text = p.sexo

 txtCalle.Text = p.calle

 txtCorreo.Text = p.correo

 txtNpuerta.Text = p.nropuerta

 txtEsquina.Text = p.esquina

Dim lista2 As List(Of String)



```
lista2 = p.patelefono()
For Each Telefono In lista2
    cbTelefono.Items.Add(Telefono)
Next
gbBuscar.Enabled = False
gbDatos.Visible = True
End If

Case 3
    MsgBox("No se encontró al paciente.", MessageBoxButtons.Error,
"S.I.BI.M - GSA")

    Case Else : MsgBox("Hubo errores al buscar el paciente. En caso de
persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -
GSA")

End Select
p = Nothing
End Sub

Private Sub frmConsultarPacientes_Leave(sender As Object, e As
EventArgs) Handles Me.Leave
    Me.Close()
End Sub
End Class
```



FrmConsultasFormulario

```
Public Class frmConsultasFormulario
```

```
    Private Sub txtCI_KeyPress(sender As Object, e As KeyPressEventArgs)  
        Handles txtCI.KeyPress
```

```
        permitirNumerosSE(e)
```

```
    End Sub
```

```
    Private Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs)  
        Handles cmdCancelar.Click
```

```
        txtCI.Clear()
```

```
        cbolD.Items.Clear()
```

```
        cbolD.ResetText()
```

```
        cboTipo.Items.Clear()
```

```
        cboTipo.ResetText()
```

```
        dgv_Form.Columns.Clear()
```

```
        gbFormulario.Enabled = False
```

```
        gbFormulario.Visible = False
```

```
        gbFiltros.Visible = False
```

```
        gbFiltros.Enabled = False
```

```
        gbBuscar.Enabled = True
```

```
    End Sub
```

```
    Private Sub cmdBuscar_Click(ByVal sender As System.Object, ByVal e As  
        System.EventArgs) Handles cmdBuscar.Click
```

```
        Dim f As New Formularios
```

```
        f.tipo = cboTipo.SelectedItem()
```

```
        f.cn = cn
```

```
        f.ci = txtCI.Text
```

```
        cboTipo.Items.Clear()
```

```
        Dim lista As List(Of String)
```



```
lista = f.tipoform2()

Select Case f.verifyci()

    Case 0 'Encontré

        If f.estadoborradop = 1 Then

            If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",_
                MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then

                Select Case f.estadop()

                    Case 0

                        MsgBox("El alta del paciente se realizó éxitosamente.",_
                            MessageBoxButtons.Information, "S.I.BI.M - GSA")

                    Case 1

                        MsgBox("Se ha perdido la sesión. Debe loguearse_
                            nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

                    Case Else

                        MsgBox("Hubo errores al realizar el alta. En caso de_
                            persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -_
                            GSA")

                End Select

            End If

        Else

            For Each Diagnostico In lista

                cboTipo.Items.Add(Diagnostico)

            Next

            gbBuscar.Enabled = False
            gbFiltros.Enabled = True
            gbFiltros.Visible = True
            gbFormulario.Enabled = True
            gbFormulario.Visible = True

        End If

    Case 3 'No encontré
```



```
    MsgBox("El paciente no existe, o no tiene formularios respondidos,  
asegurese de que ingreso correctamente la CI.", MessageBoxButtons.Error,  
"S.I.BI.M - GSA")
```

```
        cbOID.Items.Clear()  
  
        txtCI.Clear()  
  
    End Select  
  
    f = Nothing  
  
End Sub
```

```
Private Sub cboTipo_SelectedIndexChanged(sender As Object, e As  
EventArgs) Handles cboTipo.SelectedIndexChanged
```

```
    Dim f As New Formularios  
  
    f.tipo = cboTipo.SelectedItem()  
  
    f.cn = cn  
  
    Dim lista As List(Of Integer)  
  
    lista = f.idform()  
  
    cbOID.Items.Clear()  
  
    cbOID.ResetText()  
  
    For Each id In lista  
  
        cbOID.Items.Add(id)  
  
    Next  
  
End Sub
```

```
Private Sub cbOID_SelectedIndexChanged(sender As Object, e As  
EventArgs) Handles cbOID.SelectedIndexChanged
```

```
    Dim f As New Formularios  
  
    f.idf = cbOID.SelectedItem()  
  
    f.cn = cn  
  
    dgv_Form.DataSource = f.tablaform()  
  
    f = Nothing
```



End Sub

```
Private Sub frmConsultasFormulario_Leave(sender As Object, e As  
EventArgs) Handles Me.Leave
```

```
    Me.Close()
```

```
End Sub
```

```
End Class
```



frmCreditos

Public Class frmCreditos

```
Private Sub frmCreditos_Leave(sender As Object, e As EventArgs) Handles  
Me.Leave
```

```
    Me.Close()
```

```
End Sub
```

```
End Class
```



frmDiagnostico

Public Class frmDiagnostico

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim d As New Diagnosticos
```

```
    d.ci = Val(txtCI.Text)
```

```
    d.cn = cn
```

```
    Select Case d.verifyci2()
```

```
        Case 0 'Encontré
```

```
            If d.estadoborradop = 1 Then
```

```
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",  
                MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                    Select Case d.estadop()
```

```
                        Case 0
```

```
                            MsgBox("El alta del paciente se realizó éxitosamente.",  
                            MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                        Case 1
```

```
                            MsgBox("Se ha perdido la sesión. Debe loguearse  
                            nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                        Case Else
```

```
                            MsgBox("Hubo errores al realizar el alta. En caso de  
                            persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
                            GSA")
```

```
                    End Select
```

```
                End If
```

```
            Else
```

```
                Select Case d.comprobarFecha()
```

```
                    Case 0
```

```
                        If MsgBox("¿Desea ingresar los datos de un diagnóstico  
                        perteneciente a un paciente con C.I " & (txtCI.Text) & "?", MsgBoxStyle.YesNo,  
                        "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                            gbBuscar.Enabled = False
```



```
        gbDatos.Enabled = True  
        gbDatos.Visible = True  
    End If  
  
    Case Else  
  
        MsgBox("El paciente no tiene consultas con un médico, para  
        ingresar un diagnóstico es necesario previamente que se atienda con un  
        médico.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
    End Select  
  
End If  
  
Case 3 'No encontré  
  
    MsgBox("El paciente no existe, asegurese de que ingreso  
    correctamente la CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
    nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
    Case Else : MsgBox("Hubo errores al buscar el paciente. En caso de  
    persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -  
    GSA")  
  
End Select  
  
d = Nothing  
  
End Sub
```

```
Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles  
cmdIngresar.Click  
  
    Dim d As New Diagnosticos  
  
    d.ci = txtCI.Text  
    d.tipo = cboTipo.SelectedItem()  
    d.fecha = dtpFecha.Value  
    d.descripcion = txtDescripcion.Text  
    d.cn = cn  
  
    Select Case d.Guardar(1)  
        Case 0
```



```
    MsgBox("Diagnóstico ingresado éxitosamente.",  
    MessageBoxButtons.Information, "S.I.BI.M - GSA")
```

```
    Dim myValue As String = InputBox("En el caso de que el diagnóstico  
realizado sea correcto, ingrese la enfermedad actual que detecto en el  
paciente.", "ESTABLECER ENFERMEDAD ACTUAL - S.I.BI.M - GSA",  
"Ingrese enfermedad")
```

```
    If myValue.Equals("") Then
```

```
        Else
```

```
            d.enfactual = myValue
```

```
            Select Case d.setenfactual
```

```
                Case 0
```

```
                    MsgBox("Enfermedad actual actualizada éxitosamente.",  
                    MessageBoxButtons.Information, "S.I.BI.M - GSA")
```

```
                Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
                Case Else : MsgBox("Hubo errores al actualizar la enfermedad  
actual. En caso de persistir, consulte al adm. del sistema.",  
                MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
            End Select
```

```
        End If
```

```
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
        Case Else : MsgBox("Hubo errores al ingresar el diagnóstico. En caso  
de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -  
GSA")
```

```
    End Select
```

```
    d = Nothing
```

```
End Sub
```

```
Private Sub frmDiagnosticos_Leave(sender As Object, e As EventArgs)  
Handles Me.Leave
```

```
    Me.Close()
```

```
End Sub
```



```
Private Sub frmDiagnostico_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
```

```
    dtpFecha.Format = DateTimePickerFormat.Custom
```

```
    dtpFecha.CustomFormat = "yyyy/MM/dd HH:mm:ss"
```

```
    Dim d As New Diagnosticos
```

```
    d.cn = cn
```

```
    Dim lista As List(Of String)
```

```
    lista = d.tipodiag()
```

```
    For Each Diagnostico In lista
```

```
        cboTipo.Items.Add(Diagnostico)
```

```
    Next
```

```
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles
cmdCancelar.Click
```

```
    cboTipo.ResetText()
```

```
    txtCl.Clear()
```

```
    dtpFecha.ResetText()
```

```
    txtDescripcion.Clear()
```

```
    gbBuscar.Enabled = True
```

```
    gbDatos.Visible = False
```

```
End Sub
```

```
Private Sub txtCl_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles txtCl.KeyPress
```

```
    permitirNumerosSE(e)
```

```
End Sub
```

```
Private Sub dtpFecha_ValueChanged(sender As Object, e As EventArgs)
Handles dtpFecha.ValueChanged
```



```
If dtpFecha.Value > Now Then  
    MsgBox("Error, no se permite ingresar una fecha mayor a la fecha  
actual.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
    dtpFecha.ResetText()  
End If  
End Sub  
End Class
```



frmEntrevistalInicial

Public Class frmEntrevistalInicial

```
    Private Sub txtBCI_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtCl.KeyPress
        permitirNumerosSE(e)
    End Sub
```

```
    Private Sub txtTemperatura_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtTemperatura.KeyPress
```

```
        If Char.IsNumber(e.KeyChar) Then
            e.Handled = False
        ElseIf Char.IsControl(e.KeyChar) Then
            e.Handled = False
        ElseIf Char.IsSeparator(e.KeyChar) Then
            e.Handled = True
            MessageBox.Show("No se permite ingresar espacios.", "ADVERTENCIA")
        ElseIf Char.IsPunctuation(e.KeyChar) Then
            e.Handled = False
        Else
            MessageBox.Show("No se permite ingresar letras a excepción de puntos o números.", "ADVERTENCIA")
            e.Handled = True
        End If
    End Sub
```

```
    Private Sub cmdIngresar_Click(ByVal sender As Object, e As EventArgs) Handles cmdIngresar.Click
```

```
        Dim i As New EntrevistalInicial
```



```
i.ci = Val(txtCl.Text)
i.cimedico = Val(txtCIMedico.Text)
i.estadoc = txtEstadoConciencia.Text
i.gradoc = txtGradoDeColab.Text
i.fecha = dtpFecha.Value
i.movilidad = txtMovilidad.Text
i.actitud = cbActitud.SelectedItem()
i.pulso = cbPulso.SelectedItem()
i.estadon = cbEstadoNutricion.SelectedItem()
i.gradoh = txtGradoDeHidratacion.Text
i.estadoc = txtEstadoConciencia.Text
i.motivoconsulta = txtMotivo.Text
i.temperatura = txtTemperatura.Text
i.frecuenciac = txtFrecuenciaCardiaca.Text
i.frecuenciar = txtFrecuenciaResp.Text
i.enfermedad = txtEnfermedadActual.Text
i.nombreacompanante = txtNombreAcompanante.Text
i.enfermedadf = txtEnfermedadFamiliar.Text
i.antecedentes = txtAntecedentes.Text
i.tipoenfanterior = txtTipoEnfAnt.Text
i.tipoenfactual = txtTipoEnfActual.Text
i.otro = txtOtro.Text
i.cn = cn

If CheckBoxFiebre.Checked = True Then
    i.fiebre = 1
Else
    i.fiebre = 0
End If

If CheckBoxDolor.Checked = True Then
```



```
i.dolor = 1
Else
    i.dolor = 0
End If
If CheckBoxMalestar.Checked = True Then
    i.malestar = 1
Else
    i.malestar = 0
End If
Select Case i.Guardar(1)
    Case 0
        MsgBox("Entrevista inicial ingresada éxitosamente.",
        MessageBoxIcon.Information, "S.I.BI.M - GSA")
        If CheckBoxDolor.Checked = True Then
            frmPartesCuerpo.Show()
        Else
            frmPrediccion.Show()
        End If
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
    Case Else : MsgBox("Hubo errores al ingresar la entrevista inicial. En
caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error,
"S.I.BI.M - GSA")
End Select
i = Nothing
End Sub

Private Sub CheckBoxOtroS_CheckedChanged(sender As Object, e As
EventArgs) Handles CheckBoxOtroS.CheckedChanged
If CheckBoxOtroS.Checked = True Then
    txtOtro.Visible = True

```



```
Else
    txtOtro.Clear()
    txtOtro.Visible = False
End If
End Sub
```

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles
cmdBuscar.Click
```

```
    Dim i As New EntrevistaInicial
    i.ci = Val(txtCI.Text)
    i.cn = cn
    Select Case i.verifyci()
        Case 0 'Encontré
            If i.estadoborradop = 1 Then
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",_
                    MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
                    Select Case i.estadop()
                        Case 0
                            MsgBox("El alta del paciente se realizó éxitosamente.",_
                                MessageBoxIcon.Information, "S.I.BI.M - GSA")
                        Case 1
                            MsgBox("Se ha perdido la sesión. Debe loguearse_
                                nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
                        Case Else
                            MsgBox("Hubo errores al realizar el alta. En caso de_
                                persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -_
                                GSA")
                    End Select
                End If
            Else
                For Each tipo In i.actitudE
```



```
    cbActitud.Items.Add(tipo)
Next
For Each tipo In i.estadonE
    cbEstadoNutricion.Items.Add(tipo)
Next
For Each tipo In i.pulsoE
    cbPulso.Items.Add(tipo)
Next
gbBuscar.Enabled = False
gbDatos.Enabled = True
gbDatos.Visible = True
End If
Case 3 'No encontré
    MsgBox("El paciente no existe, asegurese de que ingreso correctamente la CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
    txtCI.Clear()
End Select
i = Nothing
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click
    txtCI.Clear()
    txtCIMedico.Clear()
    txtEstadoConciencia.Clear()
    txtGradoDeColab.Clear()
    dtpFecha.ResetText()
    txtMovilidad.Clear()
    cbActitud.ResetText()
    cbPulso.ResetText()
```



```
cbEstadoNutricion.ResetText()
txtGradoDeHidratacion.Clear()
txtEstadoConciencia.Clear()
txtMotivo.Clear()
txtFrecuenciaResp.Clear()
txtTemperatura.Clear()
txtFrecuenciaCardiaca.Clear()
txtEnfermedadActual.Clear()
txtNombreAcompanante.Clear()
txtEnfermedadFamiliar.Clear()
txtAntecedentes.Clear()
txtTipoEnfActual.Clear()
txtTipoEnfAnt.Clear()
txtOtro.Clear()
CheckBoxDolor.Checked = False
CheckBoxFiebre.Checked = False
CheckBoxOtroS.Checked = False
CheckBoxMalestar.Checked = False
gbBuscar.Enabled = True
gbDatos.Visible = False
End Sub
```

```
Private Sub frmEntrevistalInicial_Leave(sender As Object, e As EventArgs)
Handles Me.Leave
    Me.Close()
End Sub
```

```
Private Sub frmEntrevistalInicial_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
```

```
dtpFecha.Format = DateTimePickerFormat.Custom
```



```
dtpFecha.CustomFormat = "yyyy/MM/dd HH:mm:ss"
```

```
End Sub
```

```
Private Sub txtGradoDeColab_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtGradoDeColab.KeyPress
```

```
    permitirLetrasCE(e)
```

```
End Sub
```

```
Private Sub txtMovilidad_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtMovilidad.KeyPress
```

```
    permitirLetrasCE(e)
```

```
End Sub
```

```
Private Sub txtEstadoConciencia_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtEstadoConciencia.KeyPress
```

```
    permitirLetrasCE(e)
```

```
End Sub
```

```
Private Sub txtNombreAcompañante_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtNombreAcompañante.KeyPress
```

```
    permitirLetrasCE(e)
```

```
End Sub
```

```
Private Sub txtCIMedico_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtCIMedico.KeyPress
```

```
    permitirNumerosSE(e)
```

```
    If txtCIMedico.Text.Equals(txtCI.Text) Then
```

```
        MsgBox("Error, no se permite ingresar un médico que se atienda a sí  
mismo.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
        txtCIMedico.Clear()
```

```
    End If
```



End Sub

```
Private Sub txtTipoEnfActual_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtTipoEnfActual.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtTipoEnfAnt_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtTipoEnfAnt.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtGradoDeHidratacion_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtGradoDeHidratacion.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub dtpFecha_ValueChanged(sender As Object, e As EventArgs)  
Handles dtpFecha.ValueChanged
```

```
If dtpFecha.Value > Now Then  
    MsgBox("Error, no se permite ingresar una fecha mayor a la fecha  
actual.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
    dtpFecha.ResetText()  
End If  
End Sub
```

```
Private Sub txtTemperatura_TextChanged(sender As Object, e As  
EventArgs) Handles txtTemperatura.TextChanged
```

```
If Val(txtTemperatura.Text) >= 37.5 Then  
    CheckBoxFiebre.Checked = True  
Else
```



```
CheckBoxFiebre.Checked = False  
End If  
End Sub  
End Class
```



frmFormulario

Public Class frmFormulario

```
Private Sub cmdBuscar_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim f As New Formularios
```

```
    f.ci = Val(txtCI.Text)
```

```
    f.cn = cn
```

```
    Select Case f.verifyci2()
```

```
        Case 0 'Encontré
```

```
            If f.estadoborradop = 1 Then
```

```
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",  
                MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                    Select Case f.estadop()
```

```
                        Case 0
```

```
                            MsgBox("El alta del paciente se realizó éxitosamente.",  
                            MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                        Case 1
```

```
                            MsgBox("Se ha perdido la sesión. Debe loguearse  
                            nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                        Case Else
```

```
                            MsgBox("Hubo errores al realizar el alta. En caso de  
                            persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
                            GSA")
```

```
                    End Select
```

```
                End If
```

```
            Else
```

```
                Select Case f.comprobarFecha()
```

```
                    Case 0
```

```
                        Dim lista As List(Of String)
```

```
                        lista = f.fechafunc()
```



```
For Each Fecha In lista
    cbFecha.Items.Add(Fecha)
Next
Dim lista2 As List(Of String)
lista2 = f.tipoform()
For Each Diagnostico In lista2
    cboTipo.Items.Add(Diagnostico)
Next
gbBuscar.Enabled = False
gbDatos.Enabled = True
gbDatos.Visible = True
gbIngPreg.Enabled = True
gbIngPreg.Visible = True
gbSelecPreg.Enabled = True
gbSelecPreg.Visible = True
gbIngResp.Enabled = True
gbIngResp.Visible = True
Case Else
    MsgBox("El paciente no tiene consultas con un médico, para
    contestar un formulario es necesario previamente que se atienda con un
    médico.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
End Select
End If
Case 3 'No encontré
    MsgBox("El paciente no existe, asegurese de que ingreso
    correctamente la CI.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
Case Else : MsgBox("Hubo errores al buscar el paciente. En
caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error,
"S.I.BI.M - GSA")
```



```
End Select  
f = Nothing  
End Sub
```

```
Private Sub cbFecha_SelectedIndexChanged(sender As Object, e As  
EventArgs) Handles cbFecha.SelectedIndexChanged
```

```
    Dim f As New Formularios  
    f.fecha = cbFecha.SelectedItem()  
    f.cn = cn  
    Dim lista As List(Of String)  
    lista = f.cifunc()  
    For Each CI In lista  
        cbCIMedico.Items.Add(CI)  
    Next  
End Sub
```

```
Private Sub cboTipo_SelectedIndexChanged(sender As Object, e As  
EventArgs) Handles cboTipo.SelectedIndexChanged
```

```
    Dim f As New Formularios  
    f.tipo = cboTipo.SelectedItem()  
    f.cn = cn  
    Dim lista As List(Of Integer)  
    lista = f.idform()  
    cboPregunta.Items.Clear()  
    cboPregunta.ResetText()  
    cbolD.Items.Clear()  
    cbolD.ResetText()  
    For Each id In lista  
        cbolD.Items.Add(id)  
    Next
```



End Sub

```
Private Sub cboid_SelectedIndexChanged(sender As Object, e As EventArgs) Handles cboid.SelectedIndexChanged
```

```
    Dim f As New Formularios
```

```
    f.idf = cboid.SelectedItem()
```

```
    f.cn = cn
```

```
    cboPregunta.Items.Clear()
```

```
    cboPregunta.ResetText()
```

```
    Dim lista1 As List(Of String)
```

```
    lista1 = f.preg2()
```

```
    For Each Pregunta In lista1
```

```
        cboPregunta.Items.Add(Pregunta)
```

```
    Next
```

End Sub

```
Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles cmdIngresar.Click
```

```
    Dim f As New Formularios
```

```
    f.idf = cboid.SelectedItem()
```

```
    f.ci = txtCI.Text
```

```
    f.respuesta = txtRespuesta.Text
```

```
    f.pregunta = cboPregunta.SelectedItem()
```

```
    f.fecha = cbFecha.SelectedItem()
```

```
    f.cimedico = cbCIMedico.SelectedItem()
```

```
    f.cn = cn
```

```
    Select Case f.Guardar3(1)
```

```
        Case 0
```

```
            MsgBox("Respuesta ingresada éxitosamente.",  
                MessageBoxButtons.Information, "S.I.BI.M - GSA")
```



Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al ingresar la respuesta. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

f = Nothing

End Sub

```
Private Sub frmFormulario_Leave(sender As Object, e As EventArgs)  
Handles Me.Leave
```

Me.Close()

End Sub

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles  
cmdCancelar.Click
```

txtCl.Clear()

txtRespuesta.Clear()

cbCIMedico.ResetText()

cbFecha.ResetText()

cboID.ResetText()

cboPregunta.ResetText()

cboTipo.ResetText()

gbBuscar.Enabled = True

gbDatos.Enabled = False

gbDatos.Visible = False

gbIngPreg.Enabled = False

gbIngPreg.Visible = False

gbSelecPreg.Enabled = False

gbSelecPreg.Visible = False

gbIngResp.Enabled = False



```
gblngResp.Visible = False  
End Sub  
End Class
```



FrmlngresoPacientes

Public Class frmIngresoPacientes

```
Private Sub txtBCI_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtBCI.KeyPress  
    permitirNumerosSE(e)  
End Sub
```

```
Private Sub txtPnombre_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtPnombre.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtPapellido_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtPapellido.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtSnombre_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtSnombre.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtSapellido_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtSapellido.KeyPress  
    permitirLetrasCE(e)  
End Sub
```

```
Private Sub txtEdad_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles txtEdad.KeyPress  
    permitirNumerosSE(e)
```



End Sub

```
Private Sub cmdBuscar_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles cmdBuscar.Click
```

```
    Dim p As New Pacientes
```

```
    p.ci = Val(txtBCI.Text)
```

```
    p.cn = cn
```

```
    Select Case p.verifyci()
```

```
        Case 0
```

```
            Select Case p.Buscar()
```

```
                Case 0 'Encontré
```

```
                    If p.estadoborradop = 1 Then
```

```
                        If MsgBox("El paciente se dió de baja, ¿Desaea darle de  
alta?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                            Select Case p.estadop()
```

```
                                Case 0
```

```
                                    MsgBox("El alta del paciente se realizó éxitosamente",  
MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                                Case 1
```

```
                                    MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                                Case Else
```

```
                                    MsgBox("Hubo errores al realizar el alta. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
                            End Select
```

```
                        End If
```

```
                    Else
```

```
                        txtPnombre.Text = p.pnombre
```

```
                        txtSnombre.Text = p.snombre
```

```
                        txtPapellido.Text = p.papellido
```



```
txtSapellido.Text = p.sapellido  
dtpFechaNac.Text = p.fechanac  
txtEdad.Text = p.edad  
cbEcivil.Text = p.estadocivil  
cbSexo.Text = p.sexo  
txtCalle.Text = p.calle  
txtCorreo.Text = p.correo  
txtNpuerta.Text = p.nropuerta  
txtEsquina.Text = p.esquina  
Dim lista As List(Of String)  
lista = p.ecivil()  
For Each EstadoCivil In lista  
    cbEcivil.Items.Add(EstadoCivil)  
Next  
Dim lista1 As List(Of Char)  
lista1 = p.csexo()  
For Each Sexo In lista1  
    cbSexo.Items.Add(Sexo)  
Next  
Dim lista2 As List(Of String)  
lista2 = p.patelefono()  
For Each Telefono In lista2  
    cbTelefono.Items.Add(Telefono)  
Next  
gbTelefono.Visible = True  
gbBuscar.Enabled = False  
gbDatos.Visible = True  
cmdIngresar.Enabled = False  
cmdEliminar.Enabled = True
```



```
cmdModificar.Enabled = True  
End If  
  
Case 3 'No encontré  
  
    MsgBox("Esta persona perteneciente al personal médico ya  
tiene los datos necesarios ingresados, para ingresar como paciente  
solamente hace falta ingresar los siguientes datos.",  
MessageBoxIcon.Information, "S.I.BI.M - GSA")  
  
    frmUsuario.Show()  
  
    frmUsuario.cmdIngresar.Visible = True  
  
    frmPaciente.Enabled = False  
  
    frmPaciente.Show()  
  
    frmPaciente.cmdIngresar2.Visible = True  
  
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
  
    Case Else : MsgBox("Hubo errores al buscar el paciente. En caso  
de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")  
  
End Select  
  
Case 3  
  
Select Case p.Buscar()  
  
Case 0 'Encontré  
  
If p.estadoborradop = 1 Then  
  
    If MsgBox("El paciente se dió de baja, ¿Desaea darle de  
alta?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then  
  
        Select Case p.estadop()  
  
        Case 0  
  
            MsgBox("El alta del paciente se realizó éxitosamente",  
MessageBoxIcon.Information, "S.I.BI.M - GSA")  
  
        Case 1  
  
            MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
  
        Case Else
```



MsgBox("Hubo errores al realizar el alta. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

End If

Else

txtPnombre.Text = p.pnombre

txtSnombre.Text = p.snombre

txtPapellido.Text = p.papellido

txtSapellido.Text = p.sapellido

dtpFechaNac.Text = p.fechanac

txtEdad.Text = p.edad

cbEcivil.Text = p.estadocivil

cbSexo.Text = p.sexo

txtCalle.Text = p.calle

txtCorreo.Text = p.correo

txtNpuerta.Text = p.nropuerta

txtEsquina.Text = p.esquina

Dim lista As List(Of String)

lista = p.ecivil()

For Each EstadoCivil In lista

 cbEcivil.Items.Add(EstadoCivil)

Next

Dim lista1 As List(Of Char)

lista1 = p.csexo()

For Each Sexo In lista1

 cbSexo.Items.Add(Sexo)

Next

Dim lista2 As List(Of String)

lista2 = p.patelefono()



For Each Telefono In lista2

 cbTelefono.Items.Add(Telefono)

 Next

 gbTelefono.Visible = True

 gbBuscar.Enabled = False

 gbDatos.Visible = True

 cmdIngresar.Enabled = False

 cmdEliminar.Enabled = True

 cmdModificar.Enabled = True

 End If

Case 3

 If MsgBox("El paciente no se encontró, ¿Desea ingresarlo?",
 MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then

 frmUsuario.Show()

 frmUsuario.cmdIngresar.Visible = True

 Me.Enabled = False

 Dim lista As List(Of String)

 lista = p.ecivil()

 For Each EstadoCivil In lista

 cbEcivil.Items.Add(EstadoCivil)

 Next

 Dim lista1 As List(Of Char)

 lista1 = p.csexo()

 For Each Sexo In lista1

 cbSexo.Items.Add(Sexo)

 Next

 gbTelefono.Visible = False

 gbBuscar.Enabled = False

 gbDatos.Visible = True



```
    cmdIngresar.Enabled = True  
    cmdEliminar.Enabled = False  
    cmdModificar.Enabled = False  
  
End If  
  
End Select  
  
Case Else : MsgBox("Hubo errores al buscar el paciente. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")  
  
End Select  
  
p = Nothing  
End Sub
```

```
Private Sub cmdEliminar_Click(sender As Object, e As EventArgs) Handles cmdEliminar.Click  
  
Dim p As New Pacientes  
p.ci = Val(txtBCI.Text)  
p.cn = cn  
  
Select Case p.Eliminar()  
  
Case 0  
  
    MsgBox("Paciente eliminado éxitosamente.",  
    MessageBoxButtons.Information, "S.I.BI.M - GSA")  
  
    txtPnombre.Clear()  
    txtSnombre.Clear()  
    txtPapellido.Clear()  
    txtSapellido.Clear()  
    cbEcivil.ResetText()  
    cbTelefono.ResetText()  
    dtpFechaNac.ResetText()  
    txtEdad.Clear()  
    txtCalle.Clear()
```



```
cbEcivil.Items.Clear()  
cbSexo.Items.Clear()  
cbTelefono.Items.Clear()  
txtCorreo.Clear()  
txtNpuerta.Clear()  
txtEsquina.Clear()  
gbTelefono.Visible = False  
gbBuscar.Enabled = True  
gbDatos.Visible = False
```

Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al eliminar el paciente. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

p = Nothing

End Sub

```
Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles cmdIngresar.Click  
    Dim p As New Pacientes  
    p.ci = Val(txtBCI.Text)  
    p.pnombre = txtPnombre.Text  
    p.snombre = txtSnombre.Text  
    p.papellido = txtPapellido.Text  
    p.sapellido = txtSapellido.Text  
    p.fechanac = dtpFechaNac.Text  
    p.estadocivil = cbEcivil.Text  
    p.sex = cbSexo.Text  
    p.calle = txtCalle.Text
```



```
p.correo = txtCorreo.Text  
p.nropuerta = txtNpuerta.Text  
p.esquina = txtEsquina.Text  
p.cn = cn  
Select Case p.Guardar(1)  
    Case 0  
        MsgBox("Paciente ingresado éxitosamente.",  
        MessageBoxIcon.Information, "S.I.BI.M - GSA")  
        frmTelefono.Show()  
        frmTelefono.cmdAgregar.Visible = True  
        frmTelefono.cmdQuitar.Visible = True  
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
    Case Else : MsgBox("Hubo errores al ingresar el paciente. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")  
End Select  
p = Nothing  
End Sub  
  
Private Sub cmdModificar_Click(sender As Object, e As EventArgs) Handles  
cmdModificar.Click  
    Dim p As New Pacientes  
    p.ci = Val(txtBCI.Text)  
    p.pnombre = txtPnombre.Text  
    p.snombre = txtSnombre.Text  
    p.papellido = txtPapellido.Text  
    p.sapellido = txtSapellido.Text  
    p.fechanac = dtpFechaNac.Text  
    p.estadocivil = cbEcivil.Text  
    p.sexo = cbSexo.Text  
    p.calle = txtCalle.Text
```



```
p.telefono = txtCorreo.Text  
p.nropuerta = txtNpuerta.Text  
p.esquina = txtEsquina.Text  
p.cn = cn  
Select Case p.Guardar(0)  
    Case 0  
        MsgBox("El paciente se modificó éxitosamente.",  
        MessageBoxIcon.Information, "S.I.BI.M - GSA")  
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
        nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
        Case Else : MsgBox("Hubo errores al modificar el paciente. En caso de  
        persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
        GSA")  
    End Select  
    p = Nothing  
End Sub  
  
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles  
cmdCancelar.Click  
    txtPnombre.Clear()  
    txtSnombre.Clear()  
    txtPapellido.Clear()  
    txtSapellido.Clear()  
    cbEcivil.ResetText()  
    cbTelefono.ResetText()  
    dtpFechaNac.ResetText()  
    txtEdad.Clear()  
    txtCalle.Clear()  
    cbEcivil.Items.Clear()  
    cbSexo.Items.Clear()  
    cbTelefono.Items.Clear()
```



```
txtCorreo.Clear()
txtNpuerta.Clear()
txtEsquina.Clear()
gbTelefono.Visible = False
gbBuscar.Enabled = True
gbDatos.Visible = False
End Sub
```

```
Private Sub.dtpFechaNac_ValueChanged(sender As Object, e As EventArgs) Handles dtpFechaNac.ValueChanged
```

```
    Dim Anios As Integer
    Dim Meses As Integer
    Dim Dias As Integer
    Dim FechaCalculo As Date
    FechaCalculo = dtpFechaNac.Value
```

```
If dtpFechaNac.Value > Now Then
    MsgBox("Error, no se permite ingresar una fecha de nacimiento mayor a la fecha actual.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
End If
```

```
Do While DateAdd("YYYY", 1, FechaCalculo) <= Now
    Anios = Anios + 1
    FechaCalculo = DateAdd("YYYY", 1, FechaCalculo)
Loop
```

```
Do While DateAdd("m", 1, FechaCalculo) <= Now
    Meses = Meses + 1
    FechaCalculo = DateAdd("m", 1, FechaCalculo)
Loop
```



```
Do While DateAdd("d", 1, FechaCalculo) <= Now
```

```
    Dias = Dias + 1
```

```
    FechaCalculo = DateAdd("d", 1, FechaCalculo)
```

```
Loop
```

```
Dim Edad As String = Anios & " años, " & Meses & " meses y " & Dias & "  
días"
```

```
txtEdad.Text = Edad
```

```
End Sub
```

```
Private Sub frmIngresoPacientes_Leave(sender As Object, e As EventArgs)  
Handles Me.Leave
```

```
    Me.Close()
```

```
End Sub
```

```
Private Sub frmIngresoPacientes_Load(sender As Object, e As EventArgs)  
Handles MyBase.Load
```

```
    dtpFechaNac.Format = DateTimePickerFormat.Custom
```

```
    dtpFechaNac.CustomFormat = "yyyy/MM/dd HH:mm:ss"
```

```
End Sub
```

```
Private Sub cmdAddMed_Click(sender As Object, e As EventArgs)
```

```
    Dim m As New PersonalMedico
```

```
    m.ci = Val(txtBCI.Text)
```

```
    m.cn = cn
```

```
    Select Case m.Buscar()
```

```
        Case 0 'Encontré
```

```
            If m.estadoborradom = 1 Then
```



If MsgBox("El médico o auxiliar se dió de baja, ¿Desearía darle de alta?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then

Select Case m.estado()

Case 0

 MsgBox("El médico o auxiliar se dio de alta exitosamente.",
 MessageBoxIcon.Information, "S.I.BI.M - GSA")

Case 1

 MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

Case Else

 MsgBox("Hubo errores al efectuar el alta del médico o
auxiliar. En caso de persistir, consulte al adm. del sistema.",
 MessageBoxIcon.Error, "S.I.BI.M - GSA")

End Select

End If

End If

 MsgBox("Esta persona ya pertenece al personal médico.",
 MessageBoxIcon.Error, "S.I.BI.M - GSA")

Case 3 'No encontré

 frmMedico.Show()

Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al buscar el personal médico. En
caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error,
"S.I.BI.M - GSA")

End Select

m = Nothing

End Sub

Private Sub cmdAregar_Click(sender As Object, e As EventArgs) Handles
cmdAregar.Click

 Dim p As New Pacientes



```
p.ci = Val(txtBCI.Text)  
p.telefono = cbTelefono.Text  
p.cn = cn
```

```
Select Case p.GuardarT(1)
```

```
Case 0
```

```
    MsgBox("Teléfono ingresado éxitosamente.",  
    MessageBoxButtons.Information, "S.I.BI.M - GSA")
```

```
    cbTelefono.Items.Clear()  
    Dim lista2 As List(Of String)  
    lista2 = p.patelefono()  
    For Each Telefono In lista2  
        cbTelefono.Items.Add(Telefono)  
    Next
```

```
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
Case Else : MsgBox("Hubo errores al ingresar el teléfono. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
p = Nothing
```

```
End Sub
```

```
Private Sub cmdQuitar_Click(sender As Object, e As EventArgs) Handles  
cmdQuitar.Click
```

```
    Dim p As New Pacientes  
    p.ci = Val(txtBCI.Text)  
    p.telefono = cbTelefono.Text  
    p.cn = cn
```

```
Select Case p.EliminarT(1)
```

```
Case 0
```



```
    MsgBox("Teléfono eliminado éxitosamente.",  
    MessageBoxButtons.Information, "S.I.BI.M - GSA")
```

```
    cbTelefono.Items.Clear()  
  
    Dim lista2 As List(Of String)  
  
    lista2 = p.patelefono()  
  
    For Each Telefono In lista2  
  
        cbTelefono.Items.Add(Telefono)  
  
    Next
```

```
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
    Case Else : MsgBox("Hubo errores al eliminar el teléfono. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
p = Nothing
```

```
End Sub
```

```
Private Sub cbEcivil_KeyPress(sender As Object, e As KeyPressEventArgs)  
Handles cbEcivil.KeyPress
```

```
e.Handled = True
```

```
    MsgBox("Solo puede seleccionar los estados civiles existentes.",  
MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
End Sub
```

```
Private Sub cbSexo_KeyPress(sender As Object, e As KeyPressEventArgs)  
Handles cbSexo.KeyPress
```

```
e.Handled = True
```

```
    MsgBox("Solo puede seleccionar los sexos existentes.",  
MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
End Sub
```



```
Private Sub cbTelefono_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles cbTelefono.KeyPress
```

```
    permitirNumerosSE(e)
```

```
End Sub
```

```
End Class
```



frmLogin

Public Class frmLogin

```
Private Sub txtUsuario_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtUsuario.KeyPress
```

```
If Char.IsSeparator(e.KeyChar) Then
```

```
    e.Handled = True
```

```
    MessageBox.Show("No se permite ingresar un usuario con espacios.",  
    "ADVERTENCIA - S.I.BI.M - GSA")
```

```
Else
```

```
    e.Handled = False
```

```
End If
```

```
End Sub
```

```
Private Sub cmdIngresar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdIngresar.Click
```

```
Try
```

```
    cn.Open("MIODBC", txtUsuario.Text, txtContrasenia.Text)
```

```
    Catch ex As Exception
```

```
        MsgBox("Usuario y/o contraseña incorrectos.", MessageBoxIcon.Error,  
        "S.I.BI.M - GSA")
```

```
    Exit Sub
```

```
End Try
```

```
cn.CursorLocation = ADODB.CursorLocationEnum.adUseClient
```

```
doyPermisos(txtUsuario.Text)
```

```
cipub = txtUsuario.Text
```

```
Me.Close()
```

```
End Sub
```



```
Private Sub txtContrasenia_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtContrasenia.KeyPress
```

```
    If Char.IsSeparator(e.KeyChar) Then
```

```
        e.Handled = True
```

```
        MessageBox.Show("No se permite ingresar una contraseña con espacios.", "ADVERTENCIA - S.I.BI.M - GSA")
```

```
    Else
```

```
        e.Handled = False
```

```
    End If
```

```
End Sub
```

```
Private Sub txtContrasenia_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtContrasenia.TextChanged
```

```
    If txtContrasenia.Text.Length < 8 Then
```

```
        lblContraseniaMSG.Visible = True
```

```
        cmdIngresar.Enabled = False
```

```
    ElseIf txtContrasenia.Text.Length > 24 Then
```

```
        lblContraseniaMSG.Visible = True
```

```
        cmdIngresar.Enabled = False
```

```
    Else
```

```
        lblContraseniaMSG.Visible = False
```

```
        If lblUsuarioMSG.Visible = False And lblUsuarioMSG.Visible = False
```

```
    Then
```

```
        cmdIngresar.Enabled = True
```

```
    Else
```

```
        cmdIngresar.Enabled = False
```

```
    End If
```

```
End If
```

```
End Sub
```



```
Private Sub txtUsuario_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtUsuario.TextChanged
    If txtUsuario.Text.Length < 4 Then
        lblUsuarioMSG.Visible = True
        cmdIngresar.Enabled = False
    ElseIf txtUsuario.Text.Length > 16 Then
        lblUsuarioMSG.Visible = True
        cmdIngresar.Enabled = False
    Else
        lblUsuarioMSG.Visible = False
        If lblContraseniaMSG.Visible = False And lblUsuarioMSG.Visible = False Then
            cmdIngresar.Enabled = True
        Else
            cmdIngresar.Enabled = False
        End If
    End If
End Sub
```

```
Private Sub frmLogin_Leave(sender As Object, e As EventArgs) Handles Me.Leave
    Me.Close()
End Sub
End Class
```



frmMedico

Public Class frmMedico

```
Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles cmdIngresar.Click
```

```
    Dim m As New PersonalMedico
```

```
    m.ci = Val(frmAdmPersonalMedico.txtCl.Text)
```

```
    m.especialidad = txtEspecialidad.Text
```

```
    m.cn = cn
```

```
    If rbAuxiliarMedico.Checked = False And rbMedico.Checked = False Then
```

```
        MsgBox("Debe seleccionar el tipo de personal. (Médico o Auxiliar)",  
        MessageBoxIcon.Warning, "S.I.BI.M - GSA")
```

```
    Else
```

```
        If rbMedico.Checked = True Then
```

```
            Select Case m.Guardar3(1)
```

```
                Case 0
```

```
                    MsgBox("Se ha ingresado éxitosamente al médico.",  
                    MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                    Me.Close()
```

```
                Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
                nuevamente.")
```

```
                Case Else : MsgBox("Hubo errores al ingresar el médico. En caso  
                de persistir, consulte al adm. del sistema.")
```

```
            End Select
```

```
        ElseIf rbAuxiliarMedico.Checked = True Then
```

```
            Select Case m.Guardar4(1)
```

```
                Case 0
```

```
                    MsgBox("Se ha ingresado éxitosamente al auxiliar médico.",  
                    MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                    Me.Close()
```



Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al ingresar el auxiliar médico. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

End If

End If

m = Nothing

End Sub

Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click

txtEspecialidad.Clear()

rbAuxiliarMedico.Checked = False

rbMedico.Checked = False

Me.Close()

End Sub

Private Sub rbMedico_CheckedChanged(sender As Object, e As EventArgs) Handles rbMedico.CheckedChanged

lblEspecialidad.Visible = True

txtEspecialidad.Visible = True

End Sub

Private Sub rbAuxiliarMedico_CheckedChanged(sender As Object, e As EventArgs) Handles rbAuxiliarMedico.CheckedChanged

lblEspecialidad.Visible = False

txtEspecialidad.Visible = False

txtEspecialidad.Clear()

End Sub



frmPaciente

Public Class frmPaciente

```
Private Sub frmPaciente_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    Dim p As New Pacientes
```

```
    p.cn = cn
```

```
    Dim lista As List(Of String)
```

```
    lista = p.ecivil()
```

```
    For Each EstadoCivil In lista
```

```
        cbECivil.Items.Add(EstadoCivil)
```

```
    Next
```

```
    Dim lista1 As List(Of Char)
```

```
    lista1 = p.csexo()
```

```
    For Each Sexo In lista1
```

```
        cbSexo.Items.Add(Sexo)
```

```
    Next
```

```
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles cmdCancelar.Click
```

```
    Me.Close()
```

```
End Sub
```

```
Private Sub cmdIngresar2_Click(sender As Object, e As EventArgs) Handles cmdIngresar2.Click
```

```
    Dim p As New Pacientes
```

```
    p.ci = Val(frmIngresoPacientes.txtBCI.Text)
```

```
    p.estadocivil = cbECivil.SelectedItem
```

```
    p.sexo = cbSexo.SelectedItem
```



```
p.cn = cn  
Select Case p.Guardar1(1)  
Case 0  
    MsgBox("Paciente ingresado éxitosamente.",  
    MessageBoxIcon.Information, "S.I.BI.M - GSA")  
    Me.Close()  
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
Case Else : MsgBox("Hubo errores al ingresar el paciente. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")  
End Select  
p = Nothing  
End Sub  
End Class
```



frmPartesCuerpo

Public Class frmPartesCuerpo

```
Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles cmdIngresar.Click
```

```
    Dim i As New Entrevistalnicial
```

```
    i.ci = Val(frmEntrevistalnicial.txtCI.Text)
```

```
    i.cimedico = frmEntrevistalnicial.txtCIMedico.Text
```

```
    i.fecha = frmEntrevistalnicial.dtpFecha.Value
```

```
    i.cn = cn
```

```
    If cbBoca.Checked = True Then
```

```
        i.boca = 1
```

```
    Else
```

```
        i.boca = 0
```

```
    End If
```

```
    If cbBrazo.Checked = True Then
```

```
        i.brazo = 1
```

```
    Else
```

```
        i.brazo = 0
```

```
    End If
```

```
    If cbCabeza.Checked = True Then
```

```
        i.cabeza = 1
```

```
    Else
```

```
        i.cabeza = 0
```

```
    End If
```

```
    If cbCuello.Checked = True Then
```

```
        i.cuello = 1
```

```
    Else
```

```
        i.cuello = 0
```

```
    End If
```



```
If cbDedo.Checked = True Then  
    i.dedo = 1  
Else  
    i.dedo = 0  
End If  
  
If cbEspalda.Checked = True Then  
    i.espalda = 1  
Else  
    i.espalda = 0  
End If  
  
If cbEstomago.Checked = True Then  
    i.estomago = 1  
Else  
    i.estomago = 0  
End If  
  
If cbFrente.Checked = True Then  
    i.frente = 1  
Else  
    i.frente = 0  
End If  
  
If cbGarganta.Checked = True Then  
    i.garganta = 1  
Else  
    i.garganta = 0  
End If  
  
If cbHombro.Checked = True Then  
    i.hombro = 1  
Else  
    i.hombro = 0
```



```
End If  
If cbMano.Checked = True Then  
    i.mano = 1  
Else  
    i.mano = 0  
End If  
If cbMuslo.Checked = True Then  
    i.muslo = 1  
Else  
    i.muslo = 0  
End If  
If cbMunieca.Checked = True Then  
    i.munieca = 1  
Else  
    i.munieca = 0  
End If  
If cbNariz.Checked = True Then  
    i.nariz = 1  
Else  
    i.nariz = 0  
End If  
If cbOjo.Checked = True Then  
    i.ojo = 1  
Else  
    i.ojo = 0  
End If  
If cbOreja.Checked = True Then  
    i.oreja = 1  
Else
```



```
i.oreja = 0  
End If  
If cbPantorrilla.Checked = True Then  
    i.pantorrilla = 1  
Else  
    i.pantorrilla = 0  
End If  
If cbPecho.Checked = True Then  
    i.pecho = 1  
Else  
    i.pecho = 0  
End If  
If cbPie.Checked = True Then  
    i.pie = 1  
Else  
    i.pie = 0  
End If  
If cbPierna.Checked = True Then  
    i.pierna = 1  
Else  
    i.pierna = 0  
End If  
If cbRodilla.Checked = True Then  
    i.rodilla = 1  
Else  
    i.rodilla = 0  
End If  
If cbTobillo.Checked = True Then  
    i.tobillo = 1
```



Else

i.tobillo = 0

End If

If cbTrasero.Checked = True Then

i.trasero = 1

Else

i.trasero = 0

End If

Select Case i.Guardar2(1)

Case 0

 MsgBox("Partes del cuerpo con dolor éxitosamente.",
 MessageBoxIcon.Information, "S.I.BI.M - GSA")

 frmPrediccion.Show()

 Me.Close()

 Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")

 Case Else : MsgBox("Hubo errores al ingresar la el dolor de las partes
del cuerpo. En caso de persistir, consulte al adm. del sistema.",
 MessageBoxIcon.Error, "S.I.BI.M - GSA")

End Select

i = Nothing

End Sub

Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles
cmdCancelar.Click

 cbBoca.Checked = False

 cbBrazo.Checked = False

 cbCabeza.Checked = False

 cbCuello.Checked = False

 cbDedo.Checked = False

 cbEspalda.Checked = False



```
cbEstomago.Checked = False  
cbFrente.Checked = False  
cbGarganta.Checked = False  
cbHombro.Checked = False  
cbMano.Checked = False  
cbMuslo.Checked = False  
cbMunieca.Checked = False  
cbNariz.Checked = False  
cbOjo.Checked = False  
cbOreja.Checked = False  
cbPantorrilla.Checked = False  
cbPecho.Checked = False  
cbPie.Checked = False  
cbPierna.Checked = False  
cbRodilla.Checked = False  
cbTobillo.Checked = False  
cbTrasero.Checked = False  
Me.Close()  
End Sub  
End Class
```



frmPredicción

```
Public Class frmPrediccion
```

```
    Private Sub frmPrediccion_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
        Dim i As New Entrevistalnicial
```

```
        txtCl.Text = frmEntrevistalnicial.txtCl.Text
```

```
        i.ci = Val(txtCl.Text)
```

```
        i.cn = cn
```

```
        Select Case i.nomape()
```

```
            Case 0 'Encontré
```

```
                txtNombre.Text = i.nombrepaciente
```

```
                txtApellido.Text = i.apellidopaciente
```

```
            Case Else
```

```
                MsgBox("Hubo errores al realizar la operación. En caso de persistir,  
                consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
        End Select
```

```
        If frmEntrevistalnicial.CheckBoxFiebre.Checked = True Then
```

```
            txtFiebre.Text = "SI"
```

```
            i.fiebre = 1
```

```
        Else
```

```
            txtFiebre.Text = "NO"
```

```
            i.fiebre = 0
```

```
        End If
```

```
        If frmEntrevistalnicial.CheckBoxDolor.Checked = True Then
```

```
            txtDolor.Text = "SI"
```

```
            cmdConsultar.Enabled = True
```

```
            i.dolor = 1
```

```
        Else
```

```
            txtDolor.Text = "NO"
```



```
cmdConsultar.Enabled = False  
i.dolor = 0  
End If  
If frmEntrevistalnicial.CheckBoxMalestar.Checked = True Then  
    txtMalestar.Text = "SI"  
    i.malestar = 1  
Else  
    txtMalestar.Text = "NO"  
    i.malestar = 0  
End If  
If frmEntrevistalnicial.CheckBoxOtroS.Checked = True Then  
    txtOtro.Text = frmEntrevistalnicial.txtOtro.Text  
    i.otro = txtOtro.Text  
Else  
    txtOtro.Text = "NO"  
    i.otro = ""  
End If  
If txtDolor.Text.Equals("SI") Then  
    DataGridView1.DataSource = i.tabladata2()  
Else  
    DataGridView1.DataSource = i.tabladata1()  
End If  
i = Nothing  
End Sub
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles cmdConfirmar.Click
```

```
    Dim i As New Entrevistalnicial  
    txtCl.Text = frmEntrevistalnicial.txtCl.Text
```



```
i.ci = Val(txtCI.Text)
i.fecha = frmEntrevistalInicial.dtpFecha.Value
i.enfermedad = txtEnfermedadActual.Text
i.cn = cn
Select Case i.confirmarenf()
    Case 0 'Encontré
        MsgBox("Enfermedad actual del paciente modificada con éxito.",_
        MessageBoxIcon.Information, "S.I.BI.M - GSA")
    Case Else
        MsgBox("Hubo errores al realizar la operación. En caso de persistir,_
        consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
End Select
End Sub

Private Sub DataGridView1_CellClick(sender As Object, e As_
DataGridViewCellEventArgs) Handles DataGridView1.CellClick
    txtEnfermedadActual.Text =
Convert.ToString(DataGridView1.Rows(e.RowIndex).Cells(0).Value.ToString())
End Sub

Private Sub cmdConsultar_Click(sender As Object, e As EventArgs) Handles cmdConsultar.Click
    Dim i As New EntrevistalInicial
    i.ci = Val(txtCI.Text)
    i.fecha = frmEntrevistalInicial.dtpFecha.Value
    i.cn = cn
    Select Case i.buscardolor()
        Case 0 'Encontré
            frmPartesCuerpo.Show()
            frmPartesCuerpo.cmdIngresar.Enabled = False
        If i.boca = 1 Then
```



```
frmPartesCuerpo.cbBoca.Checked = True
Else
    frmPartesCuerpo.cbBoca.Checked = False
End If
If i.brazo = 1 Then
    frmPartesCuerpo.cbBrazo.Checked = True
Else
    frmPartesCuerpo.cbBrazo.Checked = False
End If
If i.cabeza = 1 Then
    frmPartesCuerpo.cbCabeza.Checked = True
Else
    frmPartesCuerpo.cbCabeza.Checked = False
End If
If i.cuello = 1 Then
    frmPartesCuerpo.cbCuello.Checked = True
Else
    frmPartesCuerpo.cbCuello.Checked = False
End If
If i.dedo = 1 Then
    frmPartesCuerpo.cbDedo.Checked = True
Else
    frmPartesCuerpo.cbDedo.Checked = False
End If
If i.espalda = 1 Then
    frmPartesCuerpo.cbEspalda.Checked = True
Else
    frmPartesCuerpo.cbEspalda.Checked = False
End If
```



```
If i.estomago = 1 Then
    frmPartesCuerpo.cbEstomago.Checked = True
Else
    frmPartesCuerpo.cbEstomago.Checked = False
End If

If i.frente = 1 Then
    frmPartesCuerpo.cbFrente.Checked = True
Else
    frmPartesCuerpo.cbFrente.Checked = False
End If

If i.garganta = 1 Then
    frmPartesCuerpo.cbGarganta.Checked = True
Else
    frmPartesCuerpo.cbGarganta.Checked = False
End If

If i.hombro = 1 Then
    frmPartesCuerpo.cbHombro.Checked = True
Else
    frmPartesCuerpo.cbHombro.Checked = False
End If

If i.mano = 1 Then
    frmPartesCuerpo.cbMano.Checked = True
Else
    frmPartesCuerpo.cbMano.Checked = False
End If

If i.muslo = 1 Then
    frmPartesCuerpo.cbMuslo.Checked = True
Else
    frmPartesCuerpo.cbMuslo.Checked = False
```



```
End If  
If i.munieca = 1 Then  
    frmPartesCuerpo.cbMunieca.Checked = True  
Else  
    frmPartesCuerpo.cbMunieca.Checked = False  
End If  
If i.nariz = 1 Then  
    frmPartesCuerpo.cbNariz.Checked = True  
Else  
    frmPartesCuerpo.cbNariz.Checked = False  
End If  
If i.ojo = 1 Then  
    frmPartesCuerpo.cbOjo.Checked = True  
Else  
    frmPartesCuerpo.cbOjo.Checked = False  
End If  
If i.oreja = 1 Then  
    frmPartesCuerpo.cbOreja.Checked = True  
Else  
    frmPartesCuerpo.cbOreja.Checked = False  
End If  
If i.pantorrilla = 1 Then  
    frmPartesCuerpo.cbPantorrilla.Checked = True  
Else  
    frmPartesCuerpo.cbPantorrilla.Checked = False  
End If  
If i.pecho = 1 Then  
    frmPartesCuerpo.cbPecho.Checked = True  
Else
```



```
frmPartesCuerpo.cbPecho.Checked = False  
End If  
If i.pie = 1 Then  
    frmPartesCuerpo.cbPie.Checked = True  
Else  
    frmPartesCuerpo.cbPie.Checked = False  
End If  
If i.pierna = 1 Then  
    frmPartesCuerpo.cbPierna.Checked = True  
Else  
    frmPartesCuerpo.cbPierna.Checked = False  
End If  
If i.rodilla = 1 Then  
    frmPartesCuerpo.cbRodilla.Checked = True  
Else  
    frmPartesCuerpo.cbRodilla.Checked = False  
End If  
If i.tobillo = 1 Then  
    frmPartesCuerpo.cbTobillo.Checked = True  
Else  
    frmPartesCuerpo.cbTobillo.Checked = False  
End If  
If i.trasero = 1 Then  
    frmPartesCuerpo.cbTrasero.Checked = True  
Else  
    frmPartesCuerpo.cbTrasero.Checked = False  
End If  
Case Else 'No encontré
```



MsgBox("Hubo errores al consultar las partes del cuerpo. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

End Sub

End Class





FrmPrincipal

Public Class frmPrincipal

```
Private Sub menuLogin_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menuLogin.Click
```

```
If cn.State = 0 Then
```

```
    frmLogin.MdiParent = Me
```

```
    frmLogin.Show()
```

```
    frmLogin.WindowState = FormWindowState.Maximized
```

```
Else
```

```
If MsgBox("¿Desea cerrar sesión?", MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
    cn.Close()
```

```
    frmLogin.MdiParent = Me
```

```
    frmLogin.Show()
```

```
    frmLogin.WindowState = FormWindowState.Maximized
```

```
    menuInformacion.Enabled = False
```

```
    menuFormularios.Enabled = False
```

```
    menuPersonalMedico.Enabled = False
```

```
    menuEntrevistaInicial.Enabled = False
```

```
    menuDiagnostico.Enabled = False
```

```
    menuIngTratamiento.Enabled = False
```

```
    FormularioToolStripMenuItem.Enabled = False
```

```
    InformaciónPacienteToolStripMenuItem.Enabled = False
```

```
    EntrevistaToolStripMenuItem.Enabled = False
```

```
    menuIngAnalisis.Enabled = False
```

```
    menuConsAnalisis.Enabled = False
```

```
    menuConsDiagnostico.Enabled = False
```

```
    menu_Consulta_Formularios.Enabled = False
```



```
    DiagnosticosToolStripMenuItem.Enabled = False  
    TratamientosToolStripMenuItem.Enabled = False  
    AnalisisClinicosToolStripMenuItem.Enabled = False  
End If  
End If  
End Sub
```

```
Private Sub menuFormularios_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menuFormularios.Click  
    frmABMFormulario.MdiParent = Me  
    frmABMFormulario.Show()  
    frmABMFormulario.WindowState = FormWindowState.Maximized  
End Sub
```

```
Private Sub menuPersonalMedico_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menuPersonalMedico.Click  
    frmAdmPersonalMedico.MdiParent = Me  
    frmAdmPersonalMedico.Show()  
    frmAdmPersonalMedico.WindowState = FormWindowState.Maximized  
End Sub
```

```
Private Sub menuEntrevistalInicial_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menuEntrevistalInicial.Click  
    frmEntrevistalInicial.MdiParent = Me  
    frmEntrevistalInicial.Show()  
    frmEntrevistalInicial.WindowState = FormWindowState.Maximized  
End Sub
```



Private Sub menuDiagnóstico_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menuDiagnóstico.Click

```
frmDiagnóstico.MdiParent = Me  
frmDiagnóstico.Show()  
frmDiagnóstico.WindowState = FormWindowState.Maximized  
End Sub
```

Private Sub menulngTratamiento_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menulngTratamiento.Click

```
frmTratamiento.MdiParent = Me  
frmTratamiento.Show()  
frmTratamiento.WindowState = FormWindowState.Maximized  
End Sub
```

Private Sub menulngAnalisis_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menulngAnalisis.Click

```
frmAnalysisClinico.MdiParent = Me  
frmAnalysisClinico.Show()  
frmAnalysisClinico.WindowState = FormWindowState.Maximized  
End Sub
```

Private Sub menulngDiagnóstico_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)

```
frmDiagnóstico.MdiParent = Me  
frmDiagnóstico.Show()  
frmDiagnóstico.WindowState = FormWindowState.Maximized  
End Sub
```

Private Sub menuConsAnalisis_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menuConsAnalisis.Click

```
frmConsultaAnalisis.MdiParent = Me
```



```
frmConsultaAnalisis.Show()
frmConsultaAnalisis.WindowState = FormWindowState.Maximized
End Sub
```

```
Private Sub menuConsDiagnostico_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuConsDiagnostico.Click
    frmConsultaDiagnostico.MdiParent = Me
    frmConsultaDiagnostico.Show()
    frmConsultaDiagnostico.WindowState = FormWindowState.Maximized
End Sub
```

```
Private Sub DiagnosticosToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles DiagnosticosToolStripMenuItem.Click
    frmBMDiagnostico.MdiParent = Me
    frmBMDiagnostico.Show()
    frmBMDiagnostico.WindowState = FormWindowState.Maximized
End Sub
```

```
Private Sub TratamientosToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles TratamientosToolStripMenuItem.Click
    frmBMTratamiento.MdiParent = Me
    frmBMTratamiento.Show()
    frmBMTratamiento.WindowState = FormWindowState.Maximized
End Sub
```

```
Private Sub AnalisisClinicosToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles AnalisisClinicosToolStripMenuItem.Click
    frmBMAnalisisClinico.MdiParent = Me
    frmBMAnalisisClinico.Show()
    frmBMAnalisisClinico.WindowState = FormWindowState.Maximized
```



End Sub

```
Private Sub menuCreditos_Click(sender As Object, e As EventArgs) Handles  
menuCreditos.Click
```

```
    frmCreditos.MdiParent = Me  
    frmCreditos.Show()  
    frmCreditos.WindowState = FormWindowState.Maximized
```

End Sub

```
Private Sub frmPrincipal_Load(sender As Object, e As EventArgs) Handles  
 MyBase.Load
```

```
    frmLogin.MdiParent = Me  
    frmLogin.Show()  
    frmLogin.WindowState = FormWindowState.Maximized  
    menuInformacion.Enabled = False  
    InformaciónPacienteToolStripMenuItem.Enabled = False  
    menuFormularios.Enabled = False  
    FormularioToolStripMenuItem.Enabled = False  
    menu_Consulta_Formularios.Enabled = False  
    menuPersonalMedico.Enabled = False  
    menuEntrevistaInicial.Enabled = False  
    menuDiagnóstico.Enabled = False  
    menuIngTratamiento.Enabled = False  
    menuIngAnálisis.Enabled = False  
    menuConsAnálisis.Enabled = False  
    menuConsDiagnóstico.Enabled = False  
    DiagnósticosToolStripMenuItem.Enabled = False  
    TratamientosToolStripMenuItem.Enabled = False  
    AnálisisClínicosToolStripMenuItem.Enabled = False  
    EntrevistaToolStripMenuItem.Enabled = False
```



End Sub

```
Private Sub menuSalir_Click(sender As Object, e As EventArgs) Handles  
menuSalir.Click
```

```
If cn.State = 0 Then
```

```
    Me.Close()
```

```
Else
```

```
    cn.Close()
```

```
    Me.Close()
```

```
End If
```

End Sub

```
Private Sub EntrevistaToolStripMenuItem_Click(sender As Object, e As  
EventArgs) Handles EntrevistaToolStripMenuItem.Click
```

```
    frmConsultaEntInicial.MdiParent = Me
```

```
    frmConsultaEntInicial.Show()
```

```
    frmConsultaEntInicial.WindowState = FormWindowState.Maximized
```

End Sub

```
Private Sub AToolStripMenuItem_Click(ByVal sender As Object, ByVal e As  
EventArgs) Handles menu_Consulta_Formularios.Click
```

```
    frmConsultasFormulario.MdiParent = Me
```

```
    frmConsultasFormulario.Show()
```

```
    frmConsultasFormulario.WindowState = FormWindowState.Maximized
```

End Sub

```
Private Sub FormularioToolStripMenuItem_Click(sender As Object, e As  
EventArgs) Handles FormularioToolStripMenuItem.Click
```

```
    frmFormulario.MdiParent = Me
```

```
    frmFormulario.Show()
```



```
frmFormulario.WindowState = FormWindowState.Maximized
```

```
End Sub
```

```
Private Sub PacientesToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles menuInformacion.Click
```

```
frmIngresoPacientes.MdiParent = Me
```

```
frmIngresoPacientes.Show()
```

```
frmIngresoPacientes.WindowState = FormWindowState.Maximized
```

```
End Sub
```

```
Private Sub InformaciónPacienteToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles InformaciónPacienteToolStripMenuItem.Click
```

```
frmConsultarPacientes.MdiParent = Me
```

```
frmConsultarPacientes.Show()
```

```
frmConsultarPacientes.WindowState = FormWindowState.Maximized
```

```
End Sub
```

```
End Class
```



frmTelefono

Public Class frmTelefono

```
Private Sub cmdAgregar_Click(sender As Object, e As EventArgs) Handles cmdAgregar.Click
```

```
    Dim p As New Pacientes
```

```
    p.ci = Val(frmIngresoPacientes.txtBCI.Text)
```

```
    p.telefono = cbTelefono.Text
```

```
    p.cn = cn
```

```
    Select Case p.GuardarT(1)
```

```
        Case 0
```

```
            MsgBox("Teléfono ingresado éxitosamente.",  
        MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
            cbTelefono.Items.Clear()
```

```
            Dim lista2 As List(Of String)
```

```
            lista2 = p.patelefono()
```

```
            For Each Telefono In lista2
```

```
                cbTelefono.Items.Add(Telefono)
```

```
            Next
```

```
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
        nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
        Case Else : MsgBox("Hubo errores al ingresar el teléfono. En caso de  
        persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
        GSA")
```

```
    End Select
```

```
    p = Nothing
```

```
End Sub
```

```
Private Sub cmdQuitar_Click(sender As Object, e As EventArgs) Handles cmdQuitar.Click
```

```
    Dim p As New Pacientes
```



```
p.ci = Val(frmIngresoPacientes.txtBCI.Text)
```

```
p.telefono = cbTelefono.Text
```

```
p.cn = cn
```

```
Select Case p.EliminarT(1)
```

```
Case 0
```

```
    MsgBox("Teléfono eliminado éxitosamente.",  
    MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
    cbTelefono.Items.Clear()
```

```
    Dim lista2 As List(Of String)
```

```
    lista2 = p.patelefono()
```

```
    For Each Telefono In lista2
```

```
        cbTelefono.Items.Add(Telefono)
```

```
    Next
```

```
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
    Case Else : MsgBox("Hubo errores al eliminar el teléfono. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
p = Nothing
```

```
End Sub
```

```
Private Sub cmdQuitar2_Click(sender As Object, e As EventArgs) Handles  
cmdQuitar2.Click
```

```
Dim p As New Pacientes
```

```
p.ci = Val(frmAdmPersonalMedico.txtCI.Text)
```

```
p.telefono = cbTelefono.Text
```

```
p.cn = cn
```

```
Select Case p.EliminarT(1)
```

```
Case 0
```



```
MsgBox("Teléfono eliminado éxitosamente.",  
MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
cbTelefono.Items.Clear()  
  
Dim lista2 As List(Of String)  
  
lista2 = p.patelefono()  
  
For Each Telefono In lista2  
  
    cbTelefono.Items.Add(Telefono)  
  
Next
```

```
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
Case Else : MsgBox("Hubo errores al eliminar el teléfono. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
p = Nothing
```

```
End Sub
```

```
Private Sub cmdAgregar2_Click(sender As Object, e As EventArgs) Handles  
cmdAgregar2.Click
```

```
Dim p As New Pacientes
```

```
p.ci = Val(frmAdmPersonalMedico.txtCl.Text)
```

```
p.telefono = cbTelefono.Text
```

```
p.cn = cn
```

```
Select Case p.GuardarT(1)
```

```
Case 0
```

```
MsgBox("Teléfono ingresado éxitosamente.",  
MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
cbTelefono.Items.Clear()  
  
Dim lista2 As List(Of String)  
  
lista2 = p.patelefono()  
  
For Each Telefono In lista2
```



```
cbTelefono.Items.Add(Telefono)
```

```
Next
```

```
Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
Case Else : MsgBox("Hubo errores al ingresar el teléfono. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
p = Nothing
```

```
End Sub
```

```
Private Sub cbTelefono_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles cbTelefono.KeyPress
```

```
    permitirNumerosSE(e)
```

```
End Sub
```

```
End Class
```



frmTratamiento

Public Class frmTratamiento

```
Private Sub cmdBuscarCI_Click(sender As Object, e As EventArgs) Handles cmdBuscar.Click
```

```
    Dim t As New Tratamientos
```

```
    t.ci = Val(txtCI.Text)
```

```
    t.cn = cn
```

```
    Select Case t.verifyci2()
```

```
        Case 0 'Encontré
```

```
            If t.estadoborradop = 1 Then
```

```
                If MsgBox("El paciente se dió de baja, ¿Desaea darle de alta?",  
                MsgBoxStyle.YesNo, "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                    Select Case t.estadop()
```

```
                        Case 0
```

```
                            MsgBox("Paciente dado de alta éxitosamente.",  
                            MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
                        Case 1
```

```
                            MsgBox("Se ha perdido la sesión. Debe loguearse  
                            nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
                        Case Else
```

```
                            MsgBox("Hubo errores al realizar el alta del paciente. En  
                            caso de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error,  
                            "S.I.BI.M - GSA")
```

```
                    End Select
```

```
                End If
```

```
            Else
```

```
                If MsgBox("¿Desea ingresar los datos de un tratamiento  
                perteneciente a un paciente con C.I " & (txtCI.Text) & "?", MsgBoxStyle.YesNo,  
                "S.I.BI.M - GSA") = MsgBoxResult.Yes Then
```

```
                    gbBuscar.Enabled = False
```

```
                    gbDatos.Enabled = True
```

```
                    gbDatos.Visible = True
```



End If

End If

Case 3 'No encontré

MsgBox("El paciente no existe, asegurese de que ingreso correctamente la CI.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

Case Else : MsgBox("Hubo errores al buscar el paciente. En caso de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M - GSA")

End Select

t = Nothing

End Sub

Private Sub cmdIngresar_Click(sender As Object, e As EventArgs) Handles cmdIngresar.Click

Dim t As New Tratamientos

t.ci = Val(txtCI.Text)

t.nombre = txtNombre.Text

t.fechainicio = dtpInicio.Text

t.fecharfinal = dtpFinal.Text

t.descripcion = txtDescripción.Text

t.cn = cn

Select Case t.Guardar(1)

Case 0

t.ci = txtCI.Text

t.nombre = txtNombre.Text

t.fechainicio = dtpInicio.Text

t.fecharfinal = dtpFinal.Text

t.descripcion = txtDescripción.Text



```
    MsgBox("Tratamiento ingresado éxitosamente.",  
    MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
    Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
    Case Else : MsgBox("Hubo errores al ingresar el tratamiento. En caso  
de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
End Select
```

```
t = Nothing
```

```
End Sub
```

```
Private Sub frmTratamiento_Leave(sender As Object, e As EventArgs)  
Handles Me.Leave  
    Me.Close()  
End Sub
```

```
Private Sub frmTratamiento_Load(sender As Object, e As EventArgs)  
Handles MyBase.Load  
    dtpInicio.Format = DateTimePickerFormat.Custom  
    dtpInicio.CustomFormat = "yyyy/MM/dd HH:mm:ss"  
    dtpFinal.Format = DateTimePickerFormat.Custom  
    dtpFinal.CustomFormat = "yyyy/MM/dd HH:mm:ss"  
End Sub
```

```
Private Sub cmdCancelar_Click(sender As Object, e As EventArgs) Handles  
cmdCancelar.Click  
    txtNombre.Clear()  
    dtpFinal.ResetText()  
    dtpInicio.ResetText()  
    txtDescripción.Clear()  
    gbBuscar.Enabled = True
```



```
    gbDatos.Visible = False
```

```
End Sub
```

```
Private Sub txtCI_KeyPress(sender As Object, e As KeyPressEventArgs)  
Handles txtCI.KeyPress
```

```
    permitirNumerosSE(e)
```

```
End Sub
```

```
Private Sub dtpInicio_ValueChanged(sender As Object, e As EventArgs)  
Handles dtpInicio.ValueChanged
```

```
    If dtpInicio.Value > Now Then
```

```
        MsgBox("Error, no se permite ingresar una fecha mayor a la fecha  
actual.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
        dtpInicio.ResetText()
```

```
    End If
```

```
End Sub
```

```
Private Sub dtpFinal_ValueChanged(sender As Object, e As EventArgs)  
Handles dtpFinal.ValueChanged
```

```
    If dtpFinal.Value < dtpInicio.Value Then
```

```
        MsgBox("Error, no se permite ingresar una fecha final que sea menor a  
la fecha inicial.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
```

```
        dtpFinal.ResetText()
```

```
    End If
```

```
    If dtpFinal.Value = ("31/12/9998") Then
```

```
        cbFechaFinal.Checked = True
```

```
    Else
```

```
        cbFechaFinal.Checked = False
```

```
    End If
```

```
End Sub
```



```
Private Sub cbFechaFinal_CheckedChanged(sender As Object, e As  
EventArgs) Handles cbFechaFinal.CheckedChanged  
  
    If cbFechaFinal.Checked = True Then  
  
        dtpFinal.Value = "9998-12-31 00:00:00"  
  
        dtpFinal.Enabled = False  
  
        dtpFinal.Visible = False  
  
        txtNula.Visible = True  
  
    Else  
  
        txtNula.Visible = False  
  
        dtpFinal.ResetText()  
  
        dtpFinal.Visible = True  
  
        dtpFinal.Enabled = True  
  
    End If  
  
End Sub  
  
End Class
```



frmUsuario

Public Class frmUsuario

```
Private Sub txtContrasena_KeyPress(ByVal sender As Object, ByVal e As System.Windows.Forms.KeyPressEventArgs) Handles txtContrasena.KeyPress
```

```
If Char.IsSeparator(e.KeyChar) Then
```

```
    e.Handled = True
```

```
    MessageBox.Show("No se permite ingresar una contraseña con espacios.", "ADVERTENCIA - S.I.BI.M - GSA")
```

```
Else
```

```
    e.Handled = False
```

```
End If
```

```
End Sub
```

```
Private Sub txtContrasena_TextChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles txtContrasena.TextChanged
```

```
If txtContrasena.Text.Length < 8 Then
```

```
    lblContraseniaMSG.Visible = True
```

```
    cmdIngresar.Enabled = False
```

```
    cmdIngresarM.Enabled = False
```

```
ElseIf txtContrasena.Text.Length > 24 Then
```

```
    lblContraseniaMSG.Visible = True
```

```
    cmdIngresar.Enabled = False
```

```
    cmdIngresarM.Enabled = False
```

```
Else
```

```
    lblContraseniaMSG.Visible = False
```

```
If lblCCContrasena.Visible = False And lblUsuarioMSG.Visible = False  
Then
```

```
    cmdIngresar.Enabled = True
```



```
cmdIngresarM.Enabled = True  
End If  
End If  
End Sub
```

```
Private Sub txtCContrasena_KeyPress(ByVal sender As Object, ByVal e As  
System.Windows.Forms.KeyPressEventArgs) Handles  
txtCContrasena.KeyPress
```

```
If Char.IsSeparator(e.KeyChar) Then  
    e.Handled = True  
    MessageBox.Show("No se permite ingresar una contraseña con  
espacios.", "ADVERTENCIA - S.I.BI.M - GSA")  
Else  
    e.Handled = False  
End If  
End Sub
```

```
Private Sub txtCContrasena_TextChanged(ByVal sender As System.Object,  
 ByVal e As System.EventArgs) Handles txtCContrasena.TextChanged
```

```
If txtContrasena.Text.Equals(txtCContrasena.Text) Then  
    lblCContrasena.Visible = False  
    If lblContraseniaMSG.Visible = False And lblUsuarioMSG.Visible =  
False Then  
        cmdIngresar.Enabled = True  
        cmdIngresarM.Enabled = True  
    End If  
    Else  
        lblCContrasena.Visible = True  
        cmdIngresar.Enabled = False  
        cmdIngresarM.Enabled = False  
    End If
```



End Sub

```
Private Sub cmdIngresar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdIngresar.Click
```

```
    Dim p As New Pacientes
```

```
    p.usuario = txtUsuario.Text
```

```
    p.contrasena = txtContrasena.Text
```

```
    p.ci = Val(frmIngresoPacientes.txtBCI.Text)
```

```
    p.cn = cn
```

```
    Select Case p.adduser(1)
```

```
        Case 0
```

```
            MsgBox("Usuario ingresado éxitosamente.",  
        MessageBoxIcon.Information, "S.I.BI.M - GSA")
```

```
            frmIngresoPacientes.Enabled = True
```

```
            frmPaciente.Enabled = True
```

```
            Me.Close()
```

```
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")
```

```
        Case Else : MsgBox("Hubo errores al ingresar el usuario. En caso de  
persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")
```

```
    End Select
```

```
    p = Nothing
```

```
End Sub
```

```
Private Sub txtUsuario_KeyPress(sender As Object, e As  
KeyPressEventArgs) Handles txtUsuario.KeyPress
```

```
    If Char.IsSeparator(e.KeyChar) Then
```

```
        e.Handled = True
```

```
        MessageBox.Show("No se permite ingresar un usuario con espacios.",  
"ADVERTENCIA - S.I.BI.M - GSA")
```



```
Else  
    e.Handled = False  
End If  
End Sub
```

```
Private Sub cmdIngresarM_Click(sender As Object, e As EventArgs) Handles cmdIngresarM.Click  
    Dim m As New PersonalMedico  
    m.usuario = txtUsuario.Text  
    m.contrasena = txtContrasena.Text  
    m.ci = Val(frmAdmPersonalMedico.txtCl.Text)  
    m.cn = cn  
    If rbAuxiliarMedico.Checked = False And rbMedico.Checked = False Then  
        MsgBox("Debe seleccionar el tipo de usuario. (Médico o Auxiliar)",  
        MessageBoxIcon.Warning, "S.I.BI.M - GSA")  
    Else  
        If rbMedico.Checked = True Then  
            Select Case m.addusermed(1)  
                Case 0  
                    MsgBox("Usuario ingresado éxitosamente.",  
                    MessageBoxIcon.Information, "S.I.BI.M - GSA")  
                    frmAdmPersonalMedico.Enabled = True  
                    frmAdmPersonalMedico.rbMedico.Checked = True  
                    frmMedico.rbMedico.Checked = True  
                    frmMedico.Enabled = True  
                    Me.Close()  
                Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse  
nuevamente.", MessageBoxIcon.Error, "S.I.BI.M - GSA")  
                Case Else : MsgBox("Hubo errores al ingresar el usuario. En caso  
de persistir, consulte al adm. del sistema.", MessageBoxIcon.Error, "S.I.BI.M -  
GSA")  
            End Select  
        End If  
    End If  
End Sub
```



```
End Select

ElseIf rbAuxiliarMedico.Checked = True Then
    Select Case m.adduseraux(1)
        Case 0
            MsgBox("Usuario ingresado éxitosamente.",
                MessageBoxButtons.Information, "S.I.BI.M - GSA")
            frmAdmPersonalMedico.Enabled = True
            frmAdmPersonalMedico.rbAuxiliarMedico.Checked = True
            frmMedico.rbMedico.Checked = True
            frmMedico.Enabled = True
            Me.Close()
        Case 1 : MsgBox("Se ha perdido la sesión. Debe loguearse
nuevamente.", MessageBoxButtons.Error, "S.I.BI.M - GSA")
        Case Else : MsgBox("Hubo errores al ingresar el usuario. En caso
de persistir, consulte al adm. del sistema.", MessageBoxButtons.Error, "S.I.BI.M -
GSA")
    End Select
End If

End If

m = Nothing

End Sub
```

```
Private Sub txtUsuario_TextChanged(sender As Object, e As EventArgs)
Handles txtUsuario.TextChanged
    If txtUsuario.Text.Length < 4 Then
        lblUsuarioMSG.Visible = True
        cmdIngresar.Enabled = False
        cmdIngresarM.Enabled = False
    ElseIf txtUsuario.Text.Length > 16 Then
        lblUsuarioMSG.Visible = True
        cmdIngresar.Enabled = False
    End If
End Sub
```



```
cmdIngresarM.Enabled = False  
Else  
    lblUsuarioMSG.Visible = False  
    If lblContraseniaMSG.Visible = False And lblCCContrasena.Visible =  
        False Then  
        cmdIngresar.Enabled = True  
        cmdIngresarM.Enabled = True  
    Else  
        cmdIngresar.Enabled = False  
        cmdIngresarM.Enabled = False  
    End If  
End If  
End Sub  
End Class
```



Usuarios y claves del sistema:

Usuario: GSA_Pac

Clave: GSAPaciente

Usuario: GSA_Adm

Clave: GSAAministrador

Usuario: GSA_Aux

Clave: GSAAuxiliar

Usuario: GSA_Med

Clave: GSAMedico

Requisitos de la asignatura Taller

Servicios de Servidores:

Servidor de Base de Datos

Este servicio se encargará de transferir información almacenada en la base de datos de la institución médica a el software de la bitácora médica instalado en las terminales de trabajo, al mismo tiempo y situado en lugares distintos, se puede acceder a la base de datos y de esta manera, organizar la información mediante el uso de tablas, índices y registros.

Servidor de respaldo (NAS)

Este servicio estará dedicado al almacenamiento local de copias de seguridad de todos los servidores de la institución médica para facilitar así la seguridad de los datos y evitar la pérdida de los mismos.

Servidor de Correo

Este servicio se encargará de gestionar todos los emails que recibe y envía un correo electrónico propio de la institución médica. Además, es el responsable de todas las operaciones de dicho email.

Servidor Web

Este servicio se ocupará de almacenar los archivos HTML de la página web de la institución médica.

Los documentos HTML incluyen textos, imágenes, videos, presentaciones y todo tipo de información. Estos archivos se encuentran en un formato especial que solo los navegadores de los clientes pueden visualizar.

Servidor de acceso remoto

Este servicio permitirá realizar la conexión remota segura y encriptada para que ciertos usuarios puedan administrar el servidor a distancia.

Este servicio será provisto y vendrá integrado dentro del funcionamiento de nuestros servidores Dell mediante el controlador del acceso remoto integrado (iDrac9).



1) Puesto de trabajo genérico

Descripción técnica

PC

HP Compaq 6200 PRO (Desktop SFF)

- Intel Core i5 (2nd Gen) 2400 3.1 GHz
- Disco Duro HDD 250GB
- RAM 4GB DDR3
- Intel® HD Graphics 2000

Teclado

- Genius KM-200

Mouse

- Genius KM-200

Monitor

- Monitor LED AOC E1660SW
- Tamaño 16 pulgadas
- Aspecto de radio 16:9
- Resolución 1366px X 769px

Puertos

- x10 USB 2.0
- x1 Puerto serie
- x1 PS/2 Teclado
- x1 PS/2 Mouse
- x1 VGA
- x1 DVI
- x1 Micrófono/Auriculares Jack
- x1 Line IN
- x1 Line OUT
- x1 Puerto paralelo
- x1 eSATA
- x1 RJ-45

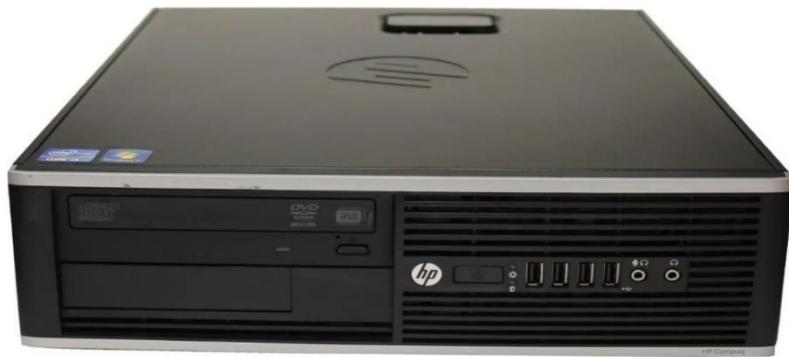
Interfaz de red



- Integrado Intel 82579LM Gigabit Ethernet
- 802.11 b/g/n Wireless

Precio: 247 dólares.

Imagen ilustrativa:



Fundamentación

Todos los puestos de trabajo tendrán un equipo HP Compaq 6200 Pro ya que es un PC ya ensamblado y no tendríamos que invertir tiempo en el ensamblado de equipos, además este equipo utiliza un gabinete “SFF” (Small Form Factor) el cual ocuparía poco espacio físico.



2) Servidor

Descripción Técnica

Dell PowerEdge R340

Procesador: Intel® Xeon® E-2224 3.4GHz, 8M cache, 4C/4T, turbo (71W)

RAM: 8GB 2666MT/s DDR4 ECC UDIMM

Almacenamiento: Dos 1TB 7.2K RPM SATA 3 3.5pulgadas Hot-plug en raid 1

Fuente: Dual, redundante, Hot-plug, 350W

Accesorios:

- ReadyRails™ Sliding Rails With Cable Management Arm
- OpenManage DVD Kit, Poweredge R340
- teclado y mouse óptico
- Dell 19in P190S European Silver Professional (1280X1024) TCO03 DVI-D

Precio: 1,936.15 dólares.

Servidor base de datos

Modelo: PowerEdge R440

Especificaciones Basicas:

Intel® Xeon® Silver 41102, 8C/16T, 11M Cache, Turbo, HT (85W) DDR4-2400

RAM: 16GB 2666MT/s RDIMMs

Almacenamiento: Cuatro 4TB 7.2K RPM SATA 6Gbps 512n 3.5in Hot-plug
HDDs raid 6

Controladora raid: PERC H730P+, 2Gb NV Cache

Fuente: Dual, Hot Plug, Redundant Power Supply (1+1), 550W

Red: integrada Broadcom 5720 doble puerto 1Gb LOM

Accesorios:

- ReadyRails Sliding Rails Without Cable Management Arm
- DVD ROM, SATA interno
- teclado y mouse óptico
- Dell 19in E190S European Black Value (1280X1024) TC003

Costo total: \$5,667.77



Servidor de respaldo (NAS)

Especificaciones Basicas:

Modelo: PowerEdge R440

Intel® Xeon® Silver 41102, 8C/16T, 11M Cache, Turbo, HT (85W) DDR4-2400

RAM: 16GB 2666MT/s RDIMMs

Almacenamiento: Cuatro 4TB 7.2K RPM SATA 6Gbps 512n 3.5in Hot-plug
HDDs raid 6

Controladora raid: PERC H730P+, 2Gb NV Cache

Fuente: Dual, Hot Plug, Redundant Power Supply (1+1), 550W

Red: integrada Broadcom 5720 doble puerto 1Gb LOM

Accesorios:

- ReadyRails Sliding Rails Without Cable Management Arm
- DVD ROM, SATA interno

Costo total: \$5,649.43

Servidor WEB/MAIL

Especificaciones Basicas:

Modelo: PowerEdge R440

Intel® Xeon® Silver 41102, 8C/16T, 11M Cache, Turbo, HT (85W) DDR4-2400

RAM: 16GB 2666MT/s RDIMMs

Almacenamiento: Cuatro 480GB SSD SATA Mix Use 6Gbps 512 2.5in Hot-plug
AG Drive, 3.5in HYB CARR, 3 DWPD, 2628 TBW SSDs raid 6

Controladora raid: PERC H730P+, 2Gb NV Cache

Fuente: Dual, Hot Plug, Redundant Power Supply (1+1), 550W

Red: integrada Broadcom 5720 doble puerto 1Gb LOM

Accesorios:

- ReadyRails Sliding Rails Without Cable Management Arm
- DVD ROM, SATA interno

Costo total: \$5,269.43



Imagen ilustrativa:



Fundamentación

Este servidor fue elegido ya que tiene la potencia de cálculo necesaria para alojar la base de datos, a su vez los componentes son fáciles de remplazar y fiables y esto es una cosa a tener en cuenta debido a que el servidor estará encendido y en funcionamiento las 24 horas del día, los 7 días de la semana, además se pensó en una solución a futuro ya que lo propuesto anteriormente no tendría problemas en futuras versiones del sistema operativo y la ampliación de la base de datos.



Rúter Firewall

Modelo: Cisco Asa5506 K9=

Especificaciones Básicas:

8 puertos gigabit ethernet rj45

1 RJ-45 y Mini USB consola

4GB ram

50 GB mSata



Costo: \$ 25.520, 40

Rúter

Modelo: Tp Link Er5120

Especificaciones Básicas:

Puertos:

1 Fixed 10/100/1000Mbps WAN

3 Free Changeable 10/100/1000Mbps WAN/LAN

1 Fixed 10/100/1000Mbps LAN/DMZ

1 Console (RJ-45 On RS232)

Costo: U\$S 205, 57





Switches

TP-Link tl sg1024de

Cantidad:4

Especificaciones Básicas:

24 puertos RJ45 a 10/100/1000 Mbps con negociación automática

(MDI/MDIX automático)

Costo: \$ 5.290



Modelo: T1500-28PCT

Cantidad:2

Especificaciones Básicas:

Puertos:

24 10/100Mbps RJ45

4 10/100/1000Mbps RJ45

2 combo Gigabit SFP Slots

Costo: U\$S 390





Racks

Rack de 42U de pie

Especificaciones Basicas:

Rack de 42U de pie de 80 x 100 cm LinkBasic Rack

Puerta de Vidrio y laterales desmontable

Incluye ruedas de tránsito y los pies ajustables.

Cumple con los requerimientos de ANSI/EIARS-310-d

Material: placa de acero sin costura.

Espesor de tren flexible vertical, de hasta 2mm.

Soporta de 500 kg Peso.

Incluye: kit de 4 ventiladores PDU 6 tomas schuko 3 bandejas fijas de 4 puntos

Kit de 40 tornillos y casoleta

Costo: U\$S 470





Rack de Pared 6U Fijo

Cantidad: 5

Especificaciones Básicas:

Color: Negro

600 x 450 mm

Costo por unidad: U\$S 99

Costo total: U\$S 495



Pachera

24 puertos Cat 6e

Costo: U\$S 39, 34



KVM

Switch Kvm-0422 Level One De 4 Puertos Usb

Especificaciones Básicas:

Permite conectar hasta 4 equipos

Utiliza la conexión vga para video y tiene usb para mouse y teclado

Costo: U\$S 112



UPPs

Para rack

Forza Atlas FDC-2012R-I

Cantidad 6

Especificaciones Básicas:

Capacidad: 2000VA/2000W

Tensión nominal: 200-240VCA

Costo: U\$S 956, 97



De escritorio

Forza SL-1012UL-C

Cantidad 5

Especificaciones Básicas:

Capacidad: 1000va 600w

Tensión nominal: 220V+/-10%

Costo: \$ 6.575



Cable de red:

Bobina cable Utp Cat6 Ab356nxt02 Nexxt

Cantidad:6

Costo por unidad: U\$S 219

Costo total: U\$S 1.314





Bobina cable sftp cat 6a 305m Nexxt

Costo: U\$S 375



Ducto pvc 2x1x200Cn

Cantidad:125

Costo por unidad: UYU\$50

Costo total: UYU\$6250



Bandeja portacables galvanizada 15x6,5x300Cn

Cantidad:117

Costo por unidad: U\$S30

Costo total: U\$S3510



Bandeja Portacables Union Tee Horizontal 150 X 65 Mm

Cantidad: 5

Costo por unidad: U\$S25

Costo total: U\$S125



Conecotor Ficha Rj45 Bolsa X 100 Unidades Cat6 Nexxt

Cantidad: 4

Costo por unidad: U\$S32

Costo total: U\$S128



Botitas / Capuchón Para Rj45 X100 Unidades Color Azul

Cantidad: 4

Costo por unidad: U\$S8

Costo total: U\$S32





Caja Red Simple Sin Jack Nexxt

Cantidad:65

Costo por unidad: U\$S2,0

Costo total: U\$S130,0



Jack rj45 x50 cat6 Keystone 110

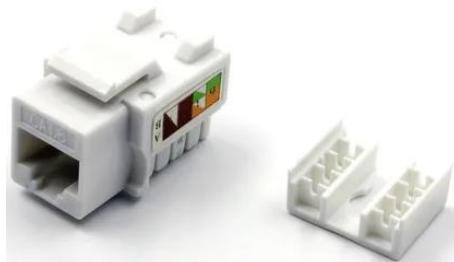
Cantidad:1

Costo: UYU\$4.556

Jack rj45 x20 cat6 Keystone 110

Cantidad:1

Costo: UYU\$2.612





3) Sistema operativo para el puesto de trabajo genérico

Windows 10 Enterprise 2019 LTSC (Tipo VLK)

Requisitos mínimos:

- Procesador: 1 GHz o más rápido o sistema en un chip (SoC)
- RAM: 1 GB para 32 bits o 2 GB para 64 bits
- Espacio en disco duro: 16 GB para un SO de 32 bits o 32 GB para un SO de 64 bits
- Tarjeta gráfica: DirectX 9 o posterior con un controlador WDDM 1.0
- Pantalla: 800x600

Soporte:

- Fecha inicial del ciclo de vida: 13/11/2018
- Fecha final del soporte estándar: 9/1/2024
- Fecha final soporte extendido: 9/1/2029

Características de soporte:

Al comprar el licenciamiento de Windows accedemos y contamos con el soporte y apoyo por parte de los técnicos especialistas, la ayuda en línea y la comunidad de Microsoft mientras que la fecha de soporte sea vigente, por lo que garantizamos una solución a cualquier problema existente.

Justificación

Se elige como sistema operativo Windows ya que el Software de Bitácora Médica será desarrollado en Visual Basic.NET que utilizará el framework .NET. Además, comprarán licencias VLK (Volume Licesing Key) ya que habrá un total de 63 puestos de trabajo en la institución.

La razón de la elección de esta versión no es por las características que la versión Enterprise ofrece, sino por la versión Enterprise 2019 LTSC (Long Term Servicing Channel) que esta nos brinda un soporte a mayor plazo (10 años), además esta versión no incluye Microsoft Edge, Cortana, ni "[Modern-Apps](#)", esto hace que el sistema tenga una leve mejora de rendimiento (Mayor desempeño por parte del sistema al procesar distintas actividades ya que no utiliza el consumo excesivo de la memoria y el disco) a diferencia de otras versiones, a su vez la versión LTSC no recibe actualizaciones semi-anuales otorgándole al sistema ser más estable (Tiene menos fallos comparados a otras versiones) por más tiempo y que no reciba cambios bruscos de forma habitual, sino que solo recibirá las actualizaciones imprescindibles. Esto favorece al equipo ya que solo se utilizará para el uso del Software de Bitácora Médica y no tendrá herramientas innecesarias.



3) Sistema operativo para el servidor

CentOS 8 (Linux)

Requisitos mínimos:

- Procesador: 2 GHz o superior
- RAM: 1 GB
- Espacio en disco duro: 20 GB

Soporte:

- Fecha inicial del ciclo de vida: **24/09/2019**
- **Fecha final del soporte estándar: 05/2024**
- **Fecha final soporte extendido: 31/05/2029**

Características de soporte:

CentOS 8 tiene una amplia comunidad y varios foros en los cuales basarnos e informarnos para ofrecerle y brindarle soluciones, además CentOS 8 y RHEL comparten características similares por lo cual también podemos apoyarnos en el soporte de RHEL para prestar y ofrecer distintas y amplias soluciones.

Justificación

Los servidores utilizarán como sistema operativo CentOS 8 (Linux) ya que la institución solicita la instalación de infraestructura de Linux en los servidores.

La elección de CentOS se debe a que es un sistema operativo basado en RHEL (Red Hat Enterprise Linux) por lo cual cuenta con la seguridad de RedHat, ya que CentOS toma las funciones de RHEL y las adapta quitando términos y sintaxis utilizados en RHEL, esto facilita en gran parte la administración.

CentOS es más estable (Reduce el riesgo de caídas, errores y pérdida de rendimiento) ya que recibe actualizaciones con poca frecuencia y a su vez es eficaz en el consumo de recursos, a comparación con otras distribuciones de Linux tiene la capacidad de procesar tareas de forma más rápida y eficiente. También debido al vínculo con Red Hat, CentOS ofrece similares funciones de seguridad y firewall, además esta distribución admite la extensión [SELinux](#) (Security Enhanced Linux).

La preferencia de CentOS 8 ante otras versiones como CentOS 7, es que CentOS 8 recibe soporte hasta períodos de 2019 (10 años desde su lanzamiento).

5) Esquema lógico

Descripción Técnica

Para la instalación de la red se utilizará topología de estrella para la interconexión de las sucursales y para la red interna se utilizó una de árbol.

Se instalarán:

Un total de 65 Puestos de trabajo genéricos, 6 access points, 6 switches y un router en toda la institución.

Habrá 4 equipos divididos en 2 puestos de triage cercanos a los 15 boxes de emergencia, el motivo de su ubicación es debido a que en salas de emergencias no tendría por qué haber puestos de trabajos ya que son salas de camas y además ciertos médicos deberán acceder al software de la bitácora médica.

Otros 15 equipos divididos en los 15 boxes de análisis clínicos (1 en cada box), además en otros 2 puestos de triage restantes hay 2 equipos, la razón de la ubicación es debido a la alta cantidad de equipos en los boxes de análisis clínicos que utilizarían los médicos para acceder al software mencionado anteriormente, por lo cual no haría falta ubicar más equipos en los puestos triage.

También para cada puesto de atención al cliente hay ubicado 1 equipo por puesto a excepción de un único puesto de atención al cliente que hay 2 equipos ya que damos por entendido que en el mismo puesto trabajan 2 personas.

Luego también se encuentran 2 equipos en las salas de reuniones (1 en cada sala de reuniones) con el motivo de que si algún empleado de la institución quiere utilizar un proyector o realizar la muestra de cualquier tipo de documento.

Después, hay 4 equipos para los puestos de auto consulta (cada puesto utiliza 1 equipo), es necesario la implementación de un equipo allí mismo para que cada puesto de auto consulta funcione correctamente.

Hay otro equipo por cada cafetería ya que se entiende que serían necesarios para el funcionamiento del negocio.

A su vez, hay 2 equipos distribuidos en los 2 puestos de gerencia (1 en cada puesto) para que un gerente si desea pueda utilizar el software de la institución.

Además, hay otros 25 equipos distribuidos en los 25 consultorios para médicos y especialistas de la planta alta (1 en cada consultorio), para que los mismos especialistas y médicos accedan al software.

Alojando la base de datos habrá un servidor.

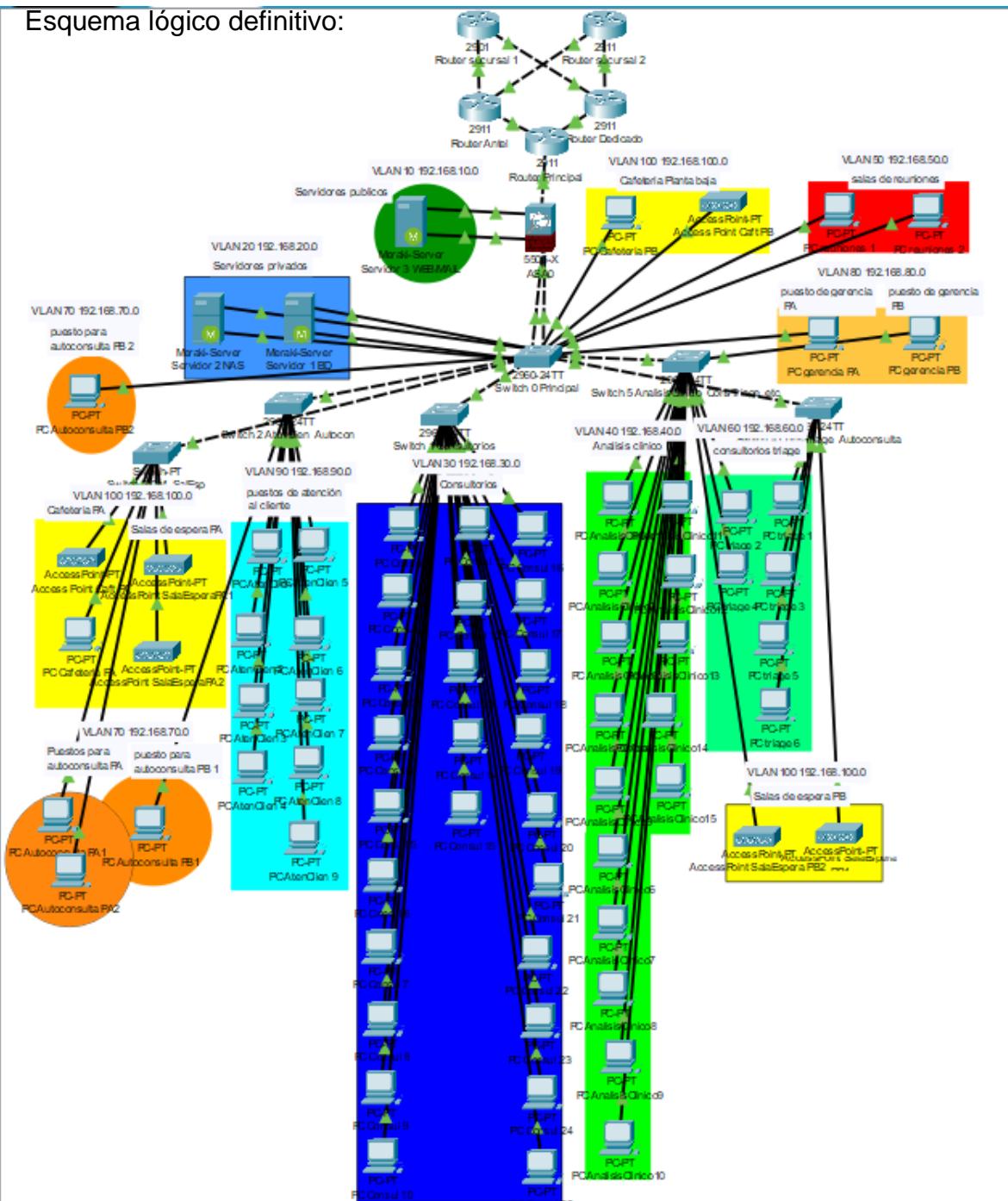
Para brindar acceso a internet a los pacientes y sus acompañantes se instalarán 4 access points para las salas de espera.

También habrá 2 access points para las cafeterías para el uso de sus clientes.

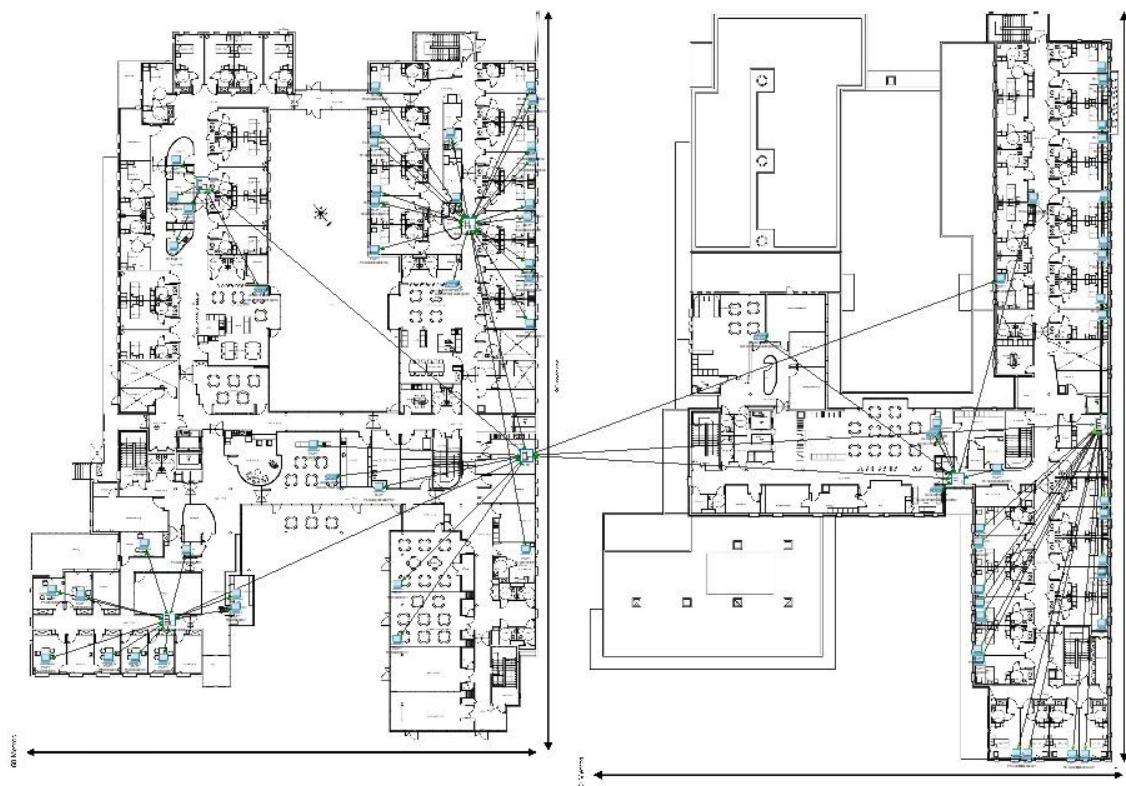
Luego para conectarse con el exterior se utilizará un router.

Finalmente habrá 6 suiches que se encargaran de proveer las conexiones necesarias para la implementación de la red.

Esquema lógico definitivo:



Esquema lógico en planos:



- █ 15 boxes de análisis clínicos
- █ 15 boxes de emergencia
- █ 8 puestos de atención al cliente
- █ 4 salas de espera
- █ 2 cafeterías
- █ 2 salas de reuniones
- █ 25 consultorios para médicos y especialistas
- █ 4 consultorios triage (atención primaria de emergencia)
- █ 2 puestos de gerencia
- █ 4 puestos para autoconsulta
- █ 2 salas de redes
- █ Puesto de trabajo
- █ Rack de redes
- █ Ducto PVC 2x2
- █ Bandeja galvanizada
- █ Access Point
- △ Switch
- █ Backbone

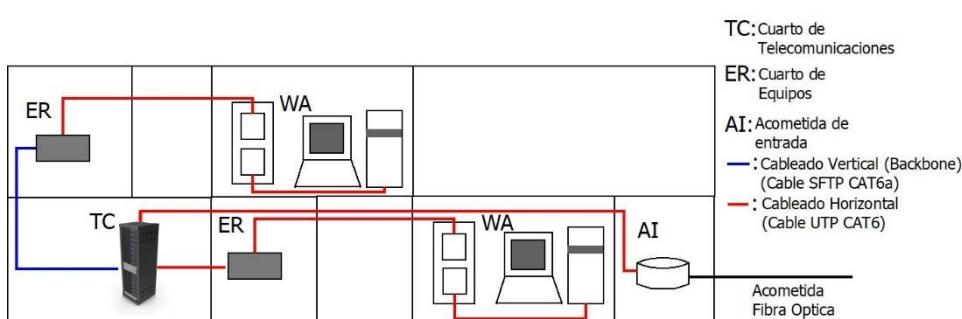
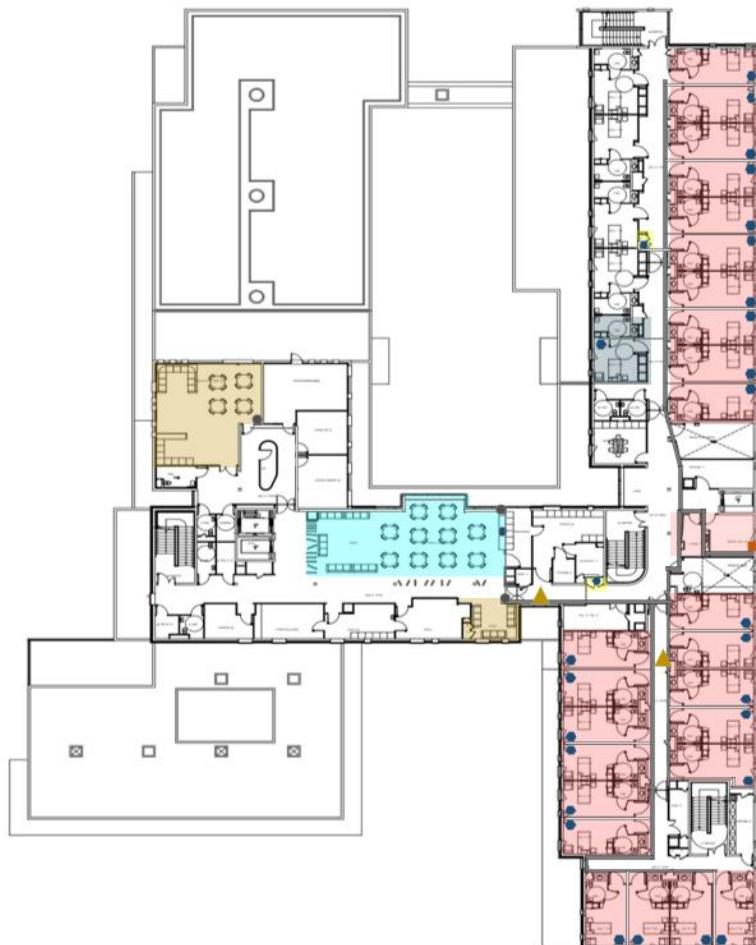
Referencias:

S.I.BI.M
I.T.I-C.E.T.P
3° BB
614

Plano planta baja:



Plano planta alta:



Esquema vertical:

Fundamentación

Utilizamos la topología de árbol ya que provee una conexión sencilla y no requiere de muchas conexiones ni dispositivos intermedios.

Servicios montados en el Router.

VLAN (Virtual Local Area Network)

Una VLAN, son_redes lógicas independientes dentro de una misma red física. Varias VLAN pueden coexistir en un único dispositivo de red físico o en una única red física. Son útiles para administrar la red, separando segmentos lógicos de una LAN, que no deberían intercambiar datos usando la red local (aunque podrían hacerlo a través de un router o un Switch de capa 3 del modelo OSI).

VLAN 10 (192.168.10.0):Servidores públicos
VLAN 20 (192.168.20.0):Servidores privados
VLAN 30 (192.168.30.0):Consultorios
VLAN 40 (192.168.40.0):Analisis clinico
VLAN 50 (192.168.50.0): Salas de reuniones
VLAN 60 (192.168.60.0): Consultorios triage
VLAN 70 (192.168.70.0): Puestos para autoconsulta
VLAN 80 (192.168.80.0): Puestos de gerencia
VLAN 90 (192.168.90.0): Puestos de atencion al cliente
VLAN 100 (192.168.100.0): Red pública

Configuración en Router:

```
Enable
Configure terminal
Interface gigabitethernet 0/0
No shutdown
Encapsulation dot1q 10 native
Ip address 192.168.10.1 255.255.255.0
Encapsulation dot1q 20 native
Ip address 192.168.20.1 255.255.255.0
Encapsulation dot1q 30 native
Ip address 192.168.30.1 255.255.255.0
Encapsulation dot1q 40 native
Ip address 192.168.40.1 255.255.255.0
Encapsulation dot1q 50 native
Ip address 192.168.50.1 255.255.255.0
Encapsulation dot1q 60 native
Ip address 192.168.60.1 255.255.255.0
Encapsulation dot1q 70 native
Ip address 192.168.70.1 255.255.255.0
Encapsulation dot1q 80 native
Ip address 192.168.80.1 255.255.255.0
Encapsulation dot1q 90 native
Ip address 192.168.90.1 255.255.255.0
Encapsulation dot1q 100 native
```



```
Ip address 192.168.100.1 255.255.255.0  
exit
```

DHCP (Dynamic Host Communication Protocol)

Brinda un servicio sobre una red IP, la cual asigna de forma dinámica (automático) direcciones IPs a los equipos y a otros como por ejemplo: DNS, Gateway y Máscara de red.

Subredes

Subred_1: 192.168.10.0 255.255.255.0

IPs fijas: 192.168.1.1 a 192.168.1.10, 192.168.10.1 a 192.168.10.3,
192.168.20.1 a 192.168.20.5

IPs dinámicas: 192.168.1.11 a 192.168.10.0, 192.168.10.4 a 192.168.20.0,
192.168.20.6 a 255.255.255.255

Gateway: 192.168.1.1

Dns: 192.168.1.9

Subred_2: 192.168.110.0 255.255.254.0

IPs fijas: 192.168.110.1 a 192.168.110.10

IPs dinámicas: 192.168.110.0, 192.168.110.11 a 255.255.255.255

Gateway: 192.168.110.1

Dns: 192.168.110.10

Configuración en el router:

Enable

Configure terminal

Interface gigabitethernet 0/0

ip address 192.168.1.1

exit

ip dhcp pool Subred_1

network 192.168.10.0 255.255.255.0

default_router 192.168.1.1

dns-server 192.168.1.9

exit

ip dhcp excluded-address 192.168.1.1 192.168.1.10

ip dhcp excluded-address 192.168.10.1 192.168.10.3

ip dhcp excluded-address 192.168.20.1 192.168.20.5

exit

Interface gigabitethernet 0/1

ip address 192.168.110.1

ip dhcp pool Subred_2

network 192.168.110.0 255.255.254.0

default_router 192.168.110.1

dns-server 192.168.110.10

exit



```
ip dhcp excluded-address 192.168.110.1 192.168.110.10
exit
copy running-config startup-config
```

Access-list (Lista de acceso)

Son utilizadas para la seguridad en redes LAN, MAN y WAN
Es una lista secuencial de instrucciones que permite o niega, aplicadas sobre los protocolos de capa superiores o direcciones. La lista de acceso es una herramienta importante para controlar el tráfico entrante o saliente de una red.

Permisos access-list:

VLAN 10 denegadas: VLAN 20 y 100.
VLAN 20 denegadas: VLAN 10 y 30 a 90.
VLAN 30 denegadas: todo.
VLAN 40 denegadas: todo.
VLAN 50 denegadas: todo.
VLAN 60 denegadas: todo.
VLAN 70 denegadas: todo.
VLAN 80 denegadas: todo.
VLAN 90 denegadas: todo.
VLAN 100 denegada: todo excepto: VLAN 10

Configuración en el router:

```
enable
configure terminal
access-list 10 permit 192.168.20.0 0.0.0.255
access-list 10 permit 192.168.100.0 0.0.0.255
access-list 20 permit 192.168.10.0 0.0.0.255
access-list 20 permit 192.168.30.0 0.0.0.255
access-list 20 permit 192.168.40.0 0.0.0.255
access-list 20 permit 192.168.50.0 0.0.0.255
access-list 20 permit 192.168.60.0 0.0.0.255
access-list 20 permit 192.168.70.0 0.0.0.255
access-list 20 permit 192.168.80.0 0.0.0.255
access-list 20 permit 192.168.90.0 0.0.0.255
access-list 30 deny any
access-list 40 deny any
access-list 50 deny any
access-list 60 deny any
access-list 70 deny any
access-list 80 deny any
access-list 90 deny any
access-list 100 deny any
access-list 100 permit 192.168.10.0 0.0.0.255
```

Manual de procedimientos ante fallas del sistema

Falla de uno o dos discos duros de un servidor:

todos los servidores pueden soportar la pérdida de 2 discos duros sin perder información, pero de no reemplazarlos no soportaría la pérdida de otro disco, por lo cual ante la falla de uno o dos discos se debe proceder por reemplazarlos y restablecer los datos a esos discos duros.

Falla de tres o más discos duros de un servidor:

Al perder 3 discos duros todos los datos del sistema se corromperán, se debe proceder a reemplazar los discos duros averiados, reinstalar el sistema y posteriormente se deberán ejecutar el script “scriptMenu.sh” para instalar las necesidades básicas al inicializar el sistema.

Falla total del servidor:

Se debe proceder a reemplazar el servidor averiado, reinstalar el sistema y posteriormente se deberán ejecutar el script “scriptMenu.sh” para instalar las necesidades básicas al inicializar el sistema.

Incendio en sala de servidores:

Se debe proceder a reemplazar el equipamiento afectado, realizar la configuración de red en el equipo reemplazado, reinstalar el sistema en los servidores, y posteriormente se deberán ejecutar el script “scriptMenu.sh” para instalar las necesidades básicas al inicializar el sistema y realizar la restauración de la copia de seguridad ubicada en el servidor de respaldo de otra sucursal. El respaldo se debe realizar manualmente desde otra sucursal.

Incendio en una sucursal:

Se debe proceder a reemplazar la totalidad del equipamiento afectado, realizar la configuración de red en el equipo reemplazado, reinstalar el sistema en los puestos de trabajo y servidores, posteriormente se deberán ejecutar el script “scriptMenu.sh” en los mismos para instalar las necesidades básicas al inicializar el sistema y realizar la restauración de la copia de seguridad ubicada en el servidor de respaldo de otra sucursal. El respaldo se debe realizar manualmente desde otra sucursal.

Pérdida total de datos

Si se pierden datos en los servidores, se debe proceder a reinstalar el sistema y posteriormente se deberán ejecutar el script “scriptMenu.sh” para realizar la restauración de la copia de seguridad almacenada en el servidor de respaldos, si el servidor afectado es el de respaldos la restauración debe hacerse con el respaldo almacenado en otra sucursal.

Fallos por virus informáticos en servidor

Si se contaminan los datos en un servidor, se debe proceder a realizar un análisis a través de la herramienta de antivirus y en caso de este no ser eliminado se procederá a reinstalar el sistema y posteriormente se deberán ejecutar el script “scriptMenu.sh” para realizar la restauración de la copia de seguridad almacenada en el servidor de respaldos, si el servidor afectado es el de respaldos la restauración debe hacerse con el respaldo almacenado en otra sucursal.

Fallos por virus informáticos en puesto de trabajo

Si se contaminan los datos en los puestos de trabajo, se debe proceder a realizar un análisis a través de la herramienta de antivirus y en caso de este no ser eliminado se procederá a reinstalar el sistema, para realizar la restauración de la copia de seguridad almacenada en el servidor de respaldos, si el servidor afectado es el de respaldos la restauración debe hacerse con el respaldo almacenado en otra sucursal.

Robo de equipos en una sucursal:

Se debe proceder a reemplazar el equipamiento perdido, realizar la configuración de red en el equipo reemplazado perdido, reinstalar el sistema en los servidores perdidos, y posteriormente se deberán ejecutar el script “scriptMenu.sh”. El restablecimiento del sistema debe realizar manualmente desde otra sucursal, si el servidor de respaldos fue afectado, de lo contrario se podrá restablecer el sistema localmente.

Problemas eléctricos en una sucursal:

Si la energía se corta, el equipamiento informático fundamental permanecerá en funcionamiento gracias a que las UPS les distribuirá energía eléctrica por al menos 30 minutos.

Software de monitoreo (PandoraFMS)

Requerimientos mínimos

Hardware

Hardware	PEQUEÑO: Hasta 500 agentes o 5000 módulos	MEDIANA: Hasta 2000 agentes o 10000 módulos	GRANDE: Para más de 4000 agentes*
CPU	1 núcleo a 2 GHz	2 núcleos a 2,5 GHz	4 núcleos a 3 GHz
RAM	4 GB	8 GB	16 GB
Disco Duro	7200 rpm	15K rpm o SSD	SSD
Espacio en disco	20GB mínimo 40GB recomendado	60GB mínimo 120GB recomendado	120GB mínimo 250GB recomendado

Software

Software	Requisitos	
Sistema Operativo	Windows Server (2003 o superior) RedHat Enterprise (RHEL) 7.X CentOS 7.X (Recomendado) SLES 11 SP1 o superior OpenSUSE 11.X o superior Debian 5, 6, 7 o superior Ubuntu 11 o superior	
	FreeBSD 9.X y 10.X Solaris 10/OpenSolaris	Pandora FMS no da soporte oficial en estas plataformas

Base de datos

BBDD	Detalles
MySQL Standard	Versión 5.5 Para la instalación estándar se requiere un usuario con privilegios de creación en la BBDD de Pandora FMS. En caso de no poseer dicho usuario se podrá realizar una instalación manual.
Percona XTraDB	La instalación de Percona XTraDB se recomienda para entornos amplios de Pandora FMS donde se vayan a crear más de 4000 agentes. Versión 5.5 Para la instalación estándar se requiere un usuario con privilegios de creación en la BBDD de Pandora FMS. En caso de no poseer dicho usuario se podrá realizar una instalación manual.

PandoraFMS

Pandora FMS es un software de monitorización que cuenta con las siguientes funciones:

- Monitorización de red

Es un programa que detecta si hay componentes defectuosos o lentos dentro de la red e informa al administrador.

- Monitorización de servidores

Es un programa que detecta si dentro del servidor hay algún fallo, en el caso de haberlo lo informa al administrador.

- Experiencia de Usuario

Una interfaz que indica cuando y donde se produjo un error o un enfentecimiento en un periodo de tiempo corto.

- Monitorización SAP

En resumidas cuentas es un plug-in que automáticamente te envía un informe en el caso que haya un fallo.

- Monitorización de aplicaciones

Sirve para decidir si extraer las instancias de un servidor Oracle o repartirlo entre agentes.

- Monitorización de entornos virtuales

Ayuda a optimizar el rendimiento de la infraestructura (almacenamiento).

- Gestión de incidencias

Mejora el gestionamiento de toda interrupción o reducción de la calidad que no está prevista en el servicio.

- Control Remoto

Es un complemento que permite control total de cualquier servidor desde cualquier lugar mediante consola.

- Monitorización para MSP

La Metaconsola de Pandora FMS es una versión específica para la monitorización de MSP, diseñada para trabajar en entornos que requieren un gran volumen de dispositivos y que necesitan ofrecer a cada cliente una interfaz personalizada, limitando las funcionalidades de cada uno.

- Monitorización de negocio

Le ofrecemos un cuadro de mando con sus propias necesidades y diseñado conjuntamente con nosotros, con el que obtendrá datos vitales para su negocio, sin actualizaciones programadas, ni plazos de días para obtener información. Estos datos estarán actualizados en tiempo real, disponibles 24/7 y accesibles en cualquier lugar: en su oficina, su casa, en cualquier teléfono y tablet. También, podrá establecer alertas que le avisen de cualquier anomalía.

- Monitorización de servicios

La monitorización de servicios es el puente entre la tecnología y el negocio, es lo que le permite identificar fácilmente dónde está el origen del problema entre toda infraestructura tecnológica.

- Monitorización de Mainframe

Unifica la recolección de datos desde una base de datos, sistemas abiertos, redes y aplicaciones de negocio.

- Monitorización de IoT

El Internet de las Cosas, más conocido como IoT, ha supuesto una verdadera revolución en todos los sentidos. Con el creciente aumento de dispositivos produciendo y enviando información, será necesario monitorizar tanto las máquinas como los datos generados.

- Inventario

Podrá recibir avisos ante cualquier cambio en sus activos TI (p.e. un usuario nuevo en un sistema, una aplicación nueva corriendo en el mismo, un cambio en la versión de hardware, etc.). Existen varios tipos de informes de inventario que le permitirán extraer listados específicos por tipo de elemento de inventario (p.e.: licencia de Windows) o por grupos de agentes (todos los paquetes de software instalados en mis equipos).

- Recolección de logs

Pandora FMS Enterprise integra un recolector de logs que funciona de forma paralela a la monitorización, reutilizando toda su infraestructura. Podrá recoger logs y eventos de Windows en cualquiera de las máquinas que ya tiene monitorizadas, sin esfuerzo, con la misma licencia y sin tener que instalar nada adicional.

Soporte

Características de soporte	Lite & Enterprise
Solución de problemas (*)	Sí
Solución de dudas	Sí
Soporte vía web	Sí
Soporte telefónico	No
Tiempo max. respuesta (*)	4h
Respuesta de emergencia	No
Desarrollo de plugins	No



Licenciamiento

LITE
850€ / Año



Desde 35 agentes

Licencia por suscripción

3 tickets de soporte

Todas las funciones Enterprise (*)

(*) Sin experiencia de usuario

(*) Sin Satellite server

Anexos

Anexo Análisis y Diseño de Aplicaciones: Anexo I: Métricas

AJUSTE DE COMPLEJIDAD						
	0	1	2	3	4	5
	No influenci	Incidental	Moderado	Medio	Significativ	Esencial
1. ¿Requiere el sistema copias de seguridad y recuper. fiables?					Si	
2. ¿Se requiere comunicac. de datos ?						Si
3.¿ Existen funciones de func. distribuido?		Si				
4. ¿ Es crítico el rendimiento?				Si		
5. ¿ Se ejecutará el sistema en						
8. ¿ Se actualizan los archivos maestro en forma interactiva ?		Si				
9. ¿ Son complejas las entradas, las salidas, los archivos o las peticiones?		Si				
10. ¿ Es complejo el procesamiento interno ?					Si	
11. ¿ Se diseñará el código para ser reutilizable ?					Si	
12. ¿ Están incluidas en el diseño la conversión y la instalación ?				Si		
13. ¿ Se diseñará el sistema para múltiples instalaciones en diferentes organizaciones ?					Si	
14. ¿ Se diseñará la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario ?					Si	
	0	3	2	12	20	5
PF.= Cuenta-Total * (0,65+0,001* sumatoria de Fi)	=		Fi =	42		
				126		

TAMAÑO DEL PROYECTO y COSTOS DEL PROYECTO.

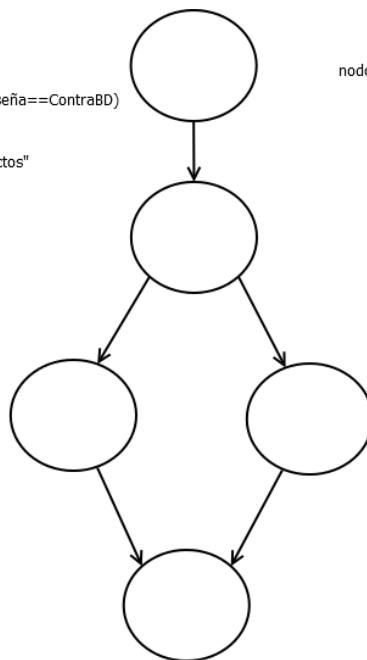
LENGUAJE DE PROGRAMACIÓN		LDC/PF	TOTALES	
Ensamblador		320	40302,08	
C		128	16120,832	
COBOL		105	13224,12	
FORTRAN		105	13224,12	
PASCAL		90	11334,96	
ADA		70	8816,08	
LENGUAJES ORIENTADO A OBJET.		30	3778,32	
LENGUAJES DE 4a.GENERACION		20	2518,88	
GENERADORES DE CÓDIGO		15	1889,16	
HOJAS DE CÁLCULO		6	755,664	
LENGUAJES GRÁFICOS (ICONOS)		4	503,776	
VALOR ESPERADO = (OPTIM +4PROBABLE+PESIMISTA)/6				
VALOR ESPERADO = 10169,978 líneas				
Según valores tomados de la bibliografía específica , se escriben 620 Líneas/me: por persona				
	tiempo total	16,40	1 persona	4,10079758 Cantidad
	Costos=	49209,571	dólares	
Se consideró un salario de U\$S		20	por hora	
Trabajando		150	horas por mes	

Cálculo de métricas por McCabe

Nº1

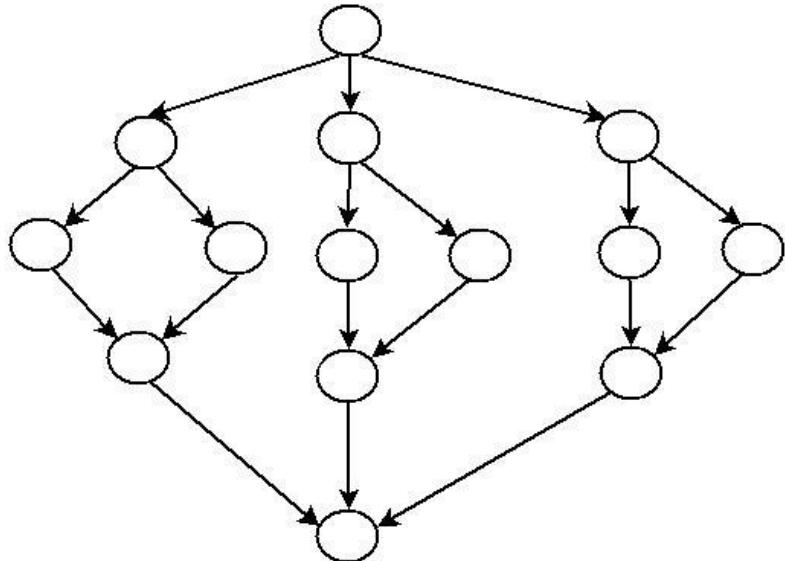
```
ingreso Usuario  
ingreso Contraseña  
Si (Usuario==UsuarioBD) && (Contraseña==ContraBD)  
DoyPermisos()  
Sino  
Escribir "Usuario y contraseña incorrectos"  
fin si
```

nodos-(lineas+1)=-1



Nº2

Ingrese op
Si op==1
ingreso Primer nombre
ingreso Segundo nombre
ingreso Primer apellido
ingreso Segundo apellido
ingreso Cédula
ingreso Edad
ingreso Dirección
ingreso Teléfono
ingreso contraseña
ingrese tipo personal médico
Si tipo personal médico==médico
ingreso especialidad
Sistema ingresa datos
fin si
Si op==2
ingreso CI
Si CI existe
Sistema carga datos
fin si
ingreso Primer nombre
ingreso Segundo nombre
ingreso Primer apellido
ingreso Segundo apellido
ingreso Cédula
ingreso Edad
ingreso Dirección
ingreso Teléfono
ingreso contraseña
ingrese tipo personal médico
Si tipo personal médico==médico
ingreso especialidad
fin si
Sistema ingresa nuevos datos
Si op==3
ingreso CI
Si CI existe
Sistema carga datos
fin si
Sistema Elimina datos cargados
fin si



Complejidad cidómatica:
 $14 - (18 + 1) = -5$

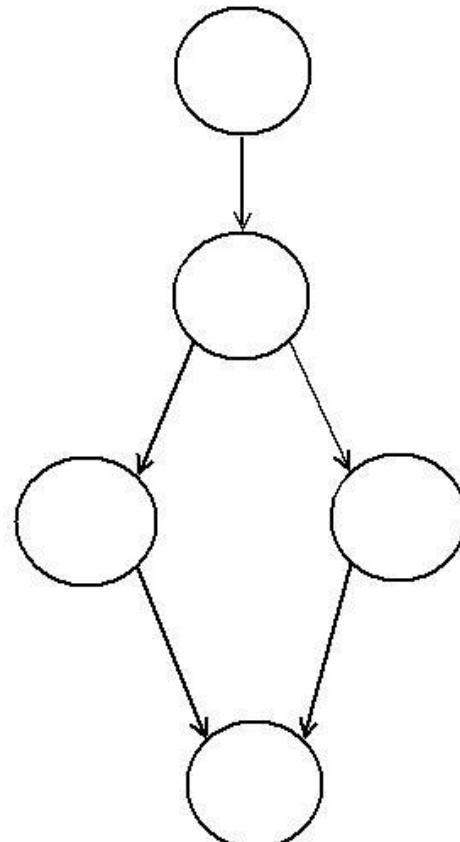
Nº3

```

ingreso Primer nombre
ingreso Segundo nombre
ingreso Primer apellido
ingreso Segundo apellido
ingreso Cédula
ingreso Edad
ingreso Sexo
ingreso Dirección
ingreso Estado civil
ingreso Nombre del acompañante
if paciente existe
    sistema no ingresa datos
sino
    sistema ingresa datos

```

Complejidad cidomática
 $5-(5+1)=-1$



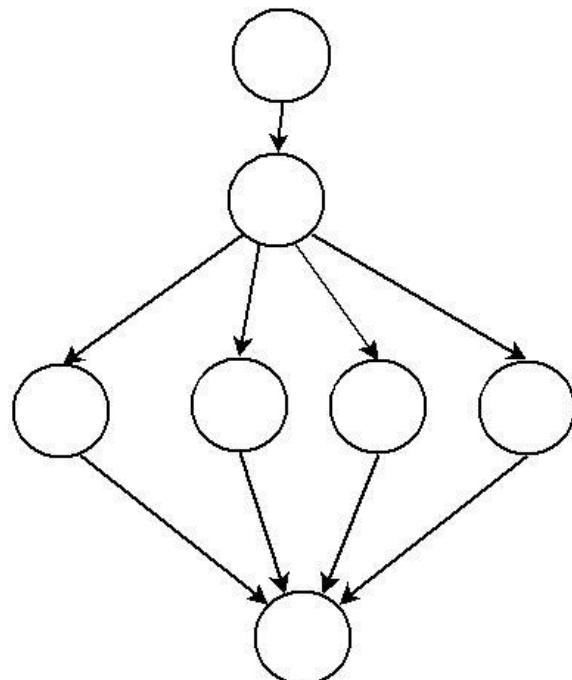
Nº4

```

Ingresar resultado entrevista inicial
Ingresa datos inspección
Ingresa constantes vitales
Ingresa temperatura
Ingresa pulso
Ingresa frecuencia respiratoria
Ingresa frecuencia cardiaca
Ingresa síntoma
si (síntoma == fiebre)
    sugiere formulario del tipo fiebre
Si (síntoma == dolor)
    sugiere formulario del tipo dolor
Si (síntoma == malestar)
    sugiere formulario del tipo malestar
si (síntoma == otro síntoma)
    sugiere formulario del tipo otro
Fin si

```

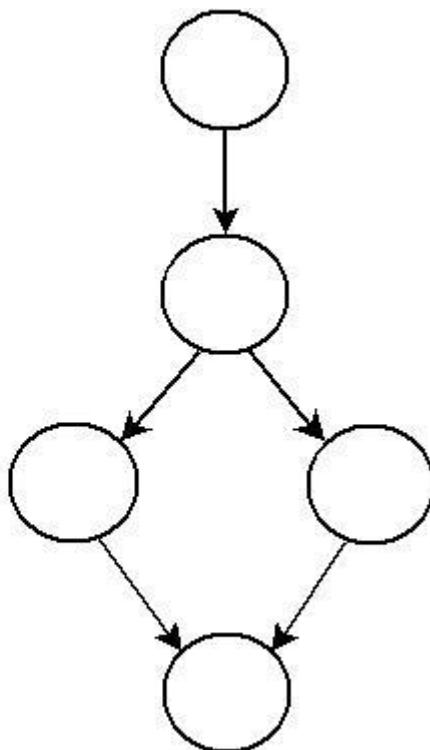
Complejidad cidomática:
 $7-(9+1) = -3$



Nº5

Ingreso diagnóstico
ingresa op
si op == 1
ingresa tipo de diagnóstico
ingresa descripción de diagnóstico
si op== 2
ingresa resultado análisis clínico
fin si

Complejidad cidomática:
 $5-(5+1)=-1$

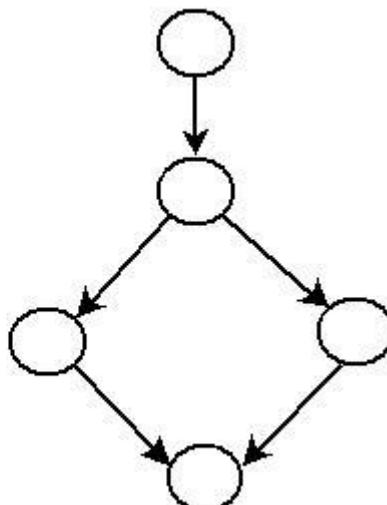


Nº6

Ingreso tratamientos

Introduce opcion
Si (opcion==1)
Muestra registro
controles
Si (opcion==2)
Ingresa nombre
Ingresa fecha inicio
Ingresa descripción
Ingresa fecha fin
Fin si

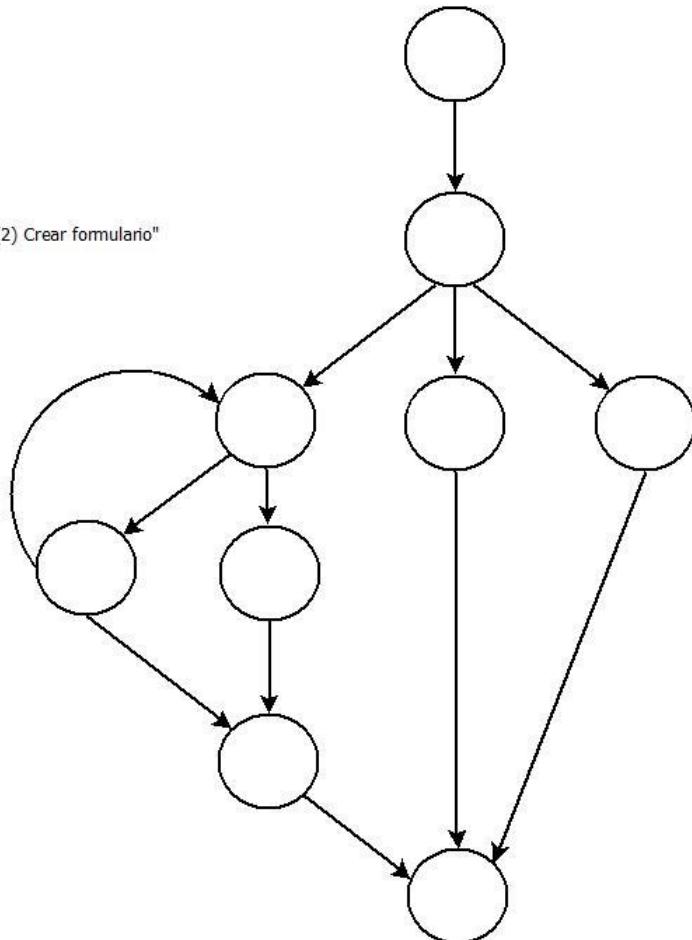
Complejidad cidomática:
 $5-(5+1)=-1$



Nº7

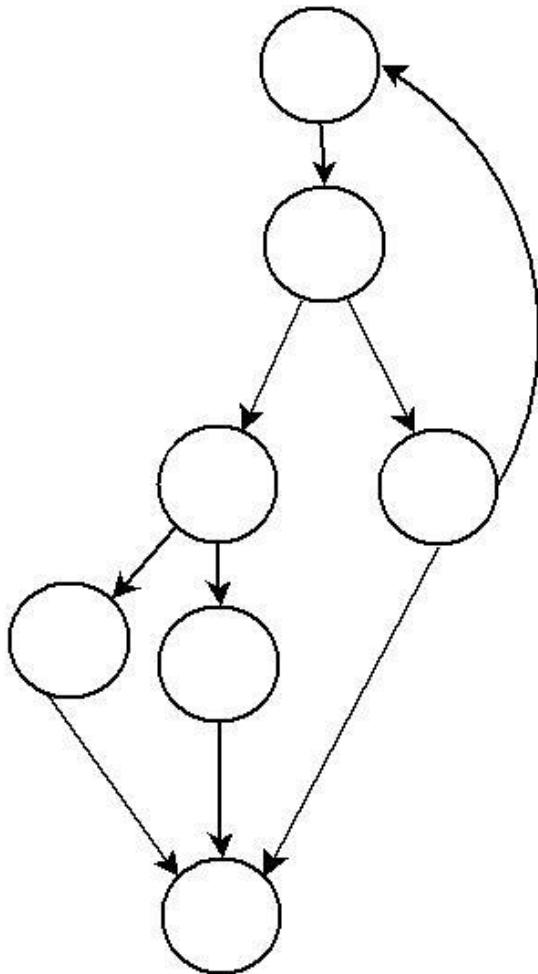
ABM Formularios de entrevista

```
bandera=0
ingreso op
si op == 1
ingreso tipo
mientras bandera=0
ingreso pregunta
Escribir "Desea ingresar otra pregunta (1) o (2) Crear formulario"
Leer opc
Si opc == 1
bandera = 0
Sino bandera = 1
Crea formulario
FinSi
FinMientras
Si op == 2
ingresa tipo formulario
carga formularios por tipo
ingresa nueva pregunta
Sino op == 3
ingresa tipo formulario
carga formularios por tipo
ingresa id formulario
elimina formulario
FinSi
```

Complejidad cidómatica:
 $9 - (12 + 1) = -4$ 

Nº8

```
Consulta información del paciente
Ingresar CI del paciente
si CI existe
introduce opcion
si (opcion ==1)
Carga datos personales del paciente
si (opcion==2)
Carga historial clínico del paciente
sino
loop
Fin si
Complejidad=7-(9+1)=-3
```

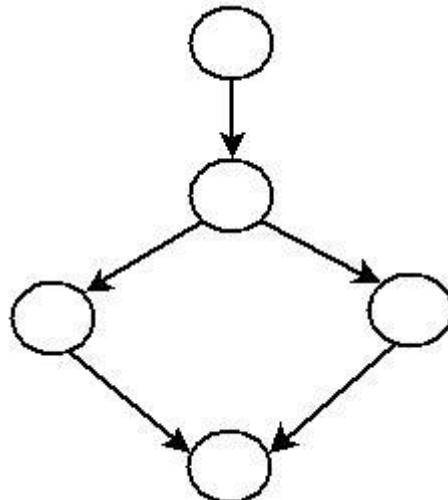


Nº9

Consultar información de diagnóstico

```
Ingresa CI
Si CI existe
Muestra diagnósticos
realizados
Sino
Escribir "CI
no existe"
Fin si
```

Complejidad cídomática:
 $5-(5+1)=-1$

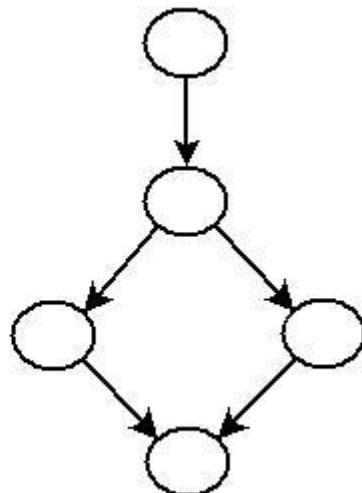


Nº10

Consultar resultado del análisis clínico

Ingresar CI
Si CI existe
Muestra análisis
clínicos realizados
Sino
Escribir "CI
no existe"
Fin si

Complejidad cídomática:
 $5 - (5+1) = -1$

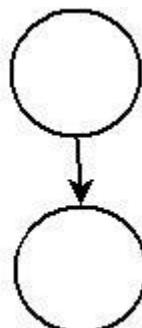


Nº11

Salir del sistema

selecciona salir del sistema

Complejidad=2-(1+1)=0



Anexo II: Agenda de reuniones

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 03 de Junio de 2020		HORA INICIO: 17:50		HORA CIERRE: 19:50
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Aspectos generales sobre la primera entrega de la asignatura de Proyecto.			
2	Aspectos 1, 2 y 3 del proyecto sobre la asignatura Inglés.			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Traducir los puntos 1, 2 y 3 realizados de la primera entrega de la asignatura de Inglés.	Agustín Acosta Santiago Iriart	4/6/2020	SI
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 09 de Junio de 2020		HORA INICIO: 17:45		HORA CIERRE: 21:20
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Primera entrega - Punto 1"			
2	Taller de Mantenimiento III "Primera entrega - Punto 1"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 12 de Junio de 2020		HORA INICIO: 19:20		
		HORA CIERRE: 21:35		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Primera entrega - Punto 1, y 2"			
2	Taller de Mantenimiento III "Primera entrega - Punto 3 y 4"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Terminar lo el punto Justificación del sistema operativo a utilizar en servidores.	Agustin Acosta Gonzalo Gularte	18/6/2020	SI
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Sábado, 13 de Junio de 2020		HORA INICIO: 17:10	HORA CIERRE: 19:55	
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Primera entrega - Punto 3, 4 y 5"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 16 de Junio de 2020	HORA INICIO: 18:10	HORA CIERRE: 21:00		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Primera entrega - Punto 1, 2 y 4"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 17 de Junio de 2020	HORA INICIO: 17:40	HORA CIERRE: 20:00		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Primera entrega"			
2	Proyecto "Métricas punto de función"			
3	Respaldo de archivos primera entrega			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 20 de Junio de 2020		HORA INICIO: 17:10		HORA CIERRE: 19:10
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Primera entrega - Punto 1 y 4"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 23 de Junio de 2020		HORA INICIO: 17:45		HORA CIERRE: 21:00
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	A.D.A "Primera entrega - Punto 2, 3 y 6"			
2	Formación Empresarial "Primera entrega (Corrección)"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Realizar Shell Script creación, modificación y eliminación de usuarios y grupos.	Gonzalo Gularte	3/7/2020	SI

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 29 de Junio de 2020		HORA INICIO: 09:55 HORA CIERRE:12:40		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Corrección primera entrega"			
2	Taller de Mantenimiento III "Primera entrega corrección"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 1 de Julio de 2020		HORA INICIO: 18:30 HORA CIERRE: 20:30		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Taller de Mantenimiento III "Primera entrega - Punto 2"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 2 de Julio de 2020		HORA INICIO: 18:00		
HORA CIERRE: 20:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Analisis y diseño de aplicaciones "Primera entrega - Especificacion de requerimientos"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Sábado, 4 de Julio de 2020		HORA INICIO: 18:00		
HORA CIERRE: 21:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Analisis y diseño de aplicaciones "Primera entrega - Especificacion de requerimientos"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA		ACTA DE REUNIONES FORMALES		
Fecha: Miércoles, 8 de Julio de 2020		HORA INICIO: 18:00		HORA CIERRE: 22:00
ASISTENTES				
1	Agustin Acosta - Integrante		Asistencia	
2	Santiago Iriart - Sub-Coordinador		Asistencia	
3	Gonzalo Gularte - Coordinador		Asistencia	
TEMAS TRATADOS				
1	Analisis y diseño de aplicaciones "Primera entrega - Especificacion de requerimientos"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Llamar y consultar a Microsoft por el licenciamiento VLK.	Agustin Acosta	10/7/2020	SI
GSA		ACTA DE REUNIONES FORMALES		
Fecha: Jueves, 9 de Julio de 2020		HORA INICIO: 12:00		HORA CIERRE: 15:10
ASISTENTES				
1	Agustin Acosta - Integrante		Asistencia	
2	Santiago Iriart - Sub-Coordinador		Asistencia	
3	Gonzalo Gularte - Coordinador		Asistencia	
TEMAS TRATADOS				
1	Sistemas Operativos III "Primera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 10 de Julio de 2020		HORA INICIO: 17:45		
HORA CIERRE: 19:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Estudio de roles"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 13 de Julio de 2020		HORA INICIO: 11:25		
HORA CIERRE: 13:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
3	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Corrección primer entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA		ACTA DE REUNIONES FORMALES		
Fecha: Lunes, 13 de Julio de 2020		HORA INICIO: 17:00	HORA CIERRE: 18:25	
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Proyecto "Planificación"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Elección y presupuesto Servidores Taller	Bruno Sunes	14/7/2020	SI
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 14 de Julio de 2020		HORA INICIO: 18:00	HORA CIERRE: 21:00	
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Base de Datos II "D.E.R"			
2	Formación Empresarial "Creación formularios 0351 y 0352, creación contrato SRL"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 15 de Julio de 2020		HORA INICIO: 18:00		
HORA CIERRE: 21:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia (Justificada)		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Base de Datos II "D.E.R"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 16 de Julio de 2020		HORA INICIO: 18:00		HORA CIERRE: 21:45
ASISTENTES				
1	Agustín Acosta - Integrante		Asistencia	
2	Bruno Sunes - Integrante		Asistencia	
3	Santiago Iriart - Sub-Coordinador		Asistencia	
4	Gonzalo Gularte - Coordinador		Asistencia	
TEMAS TRATADOS				
1	Base de Datos II "D.E.R"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Domingo, 19 de Julio de 2020		HORA INICIO: 17:00		HORA CIERRE: 20:35
ASISTENTES				
1	Agustín Acosta - Integrante		Asistencia	
2	Bruno Sunes - Integrante		Asistencia	
3	Santiago Iriart - Sub-Coordinador		Asistencia	
4	Gonzalo Gularte - Coordinador		Asistencia	
TEMAS TRATADOS				
1	Programación III "Primera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Relevamiento	Agustín Acosta	26/7/2020	SI



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 21 de Julio de 2020		HORA INICIO: 16:00	HORA CIERRE: 17:35	
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Programación III "Primera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 24 de Julio de 2020		HORA INICIO: 18:00	HORA CIERRE: 20:00	
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Base de Datos II "DER"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Domingo 26 de Julio de 2020		HORA INICIO: 17:00		
		HORA CIERRE: 20:20		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Proyecto "GANTT"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes 27 de Julio de 2020		HORA INICIO: 15:00		
		HORA CIERRE: 17:40		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Base De Datos II "Primera Entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles 29 de Julio de 2020	HORA INICIO: 15:00	HORA CIERRE: 18:10		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Proyecto "Perfil de recursos"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves 30 de Julio de 2020	HORA INICIO: 16:15	HORA CIERRE: 17:30		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia (Justificada)		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Programación III "Punto 1"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves 30 de Julio de 2020		HORA INICIO: 19:00		
		HORA CIERRE: 21:20		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Programación III "Punto 1 y 3"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes 31 de Julio de 2020		HORA INICIO: 18:00		
		HORA CIERRE: 21:15		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Programación III "Punto 4"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Sábado 1 de Agosto de 2020		HORA INICIO: 15:00		
		HORA CIERRE: 17:25		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Análisis y diseño de aplicaciones "Cálculo de métricas del proyecto por McCabe"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Sábado 1 de Agosto de 2020		HORA INICIO: 18:25		
		HORA CIERRE: 21:35		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Bruno Sunes - Integrante	Asistencia		
3	Santiago Iriart - Sub-Coordinador	Asistencia		
4	Gonzalo Gularte - Coordinador	Asistencia		
TEMAS TRATADOS				
1	Análisis y diseño de aplicaciones "Estudio de factibilidades"			
2	Proyecto "Ciclo de vida(Formulación e implementación)"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA			
ACTA DE REUNIONES FORMALES			
Fecha: Domingo 2 de Agosto de 2020		HORA INICIO: 15:00	HORA CIERRE: 02:00
ASISTENTES			
1	Agustin Acosta - Integrante	Asistencia	
2	Bruno Sunes - Integrante	Asistencia	
3	Santiago Iriart - Sub-Coordinador	Asistencia	
4	Gonzalo Gularte - Coordinador	Asistencia	
TEMAS TRATADOS			
1	Armado de carpetas para la primera entrega		
COMPRISOS DE ESTA REUNIÓN			
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento
1	No hay compromisos asignados a integrantes.		Cumplido SI/NO

GSA		
ACTA DE REUNIONES FORMALES		
a: Miércoles, 05 de Agosto de 2020	HORA INICIO: 18:00	HORA CIERRE: 20:00
ASISTENTES		
Agustín Acosta - Integrante		Asistencia
Santiago Iriart - Sub-Coordinador		Asistencia
Gonzalo Gularte - Coordinador		Asistencia
Bruno Sunes - Integrante		Asistencia
TEMAS TRATADOS		
Correcciones programación.		
COMPRISOS DE ESTA REUNIÓN		
ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento
Cumplido SI/NO		
No hay compromisos asignados a integrantes.		
GSA		
ACTA DE REUNIONES FORMALES		
a: Domingo, 09 de Agosto de 2020	HORA INICIO: 18:00	HORA CIERRE: 20:45
ASISTENTES		
Agustín Acosta - Integrante		Asistencia
Santiago Iriart - Sub-Coordinador		Asistencia
Gonzalo Gularte - Coordinador		Asistencia
Bruno Sunes - Integrante		Asistencia
TEMAS TRATADOS		
BASE DE DATOS II "DDL (Sentencias SQL de creación de tablas y estructuras necesarias)"		
COMPRISOS DE ESTA REUNIÓN		
ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento
Cumplido SI/NO		
No hay compromisos asignados a integrantes.		

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 11 de Agosto de 2020	HORA INICIO: 18:00	HORA CIERRE: 20:00		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
TEMAS TRATADOS				
1	BASE DE DATOS II "DDL (Sentencias SQL de creación de tablas y estructuras necesarias)"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 13 de Agosto de 2020	HORA INICIO: 10:55	HORA CIERRE: 12:40		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
TEMAS TRATADOS				
1	Correcciones base de datos.			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA		GSA		
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 13 de Agosto de 2020		HORA INICIO: 17:00		
		HORA CIERRE: 19:30		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
TEMAS TRATADOS				
1	Correcciones base de datos.			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA		GSA		
ACTA DE REUNIONES FORMALES				
Fecha: Sábado, 15 de Agosto de 2020		HORA INICIO: 18:00		
		HORA CIERRE: 20:20		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
TEMAS TRATADOS				
1	Correcciones base de datos.			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 17 de Agosto de 2020		HORA INICIO: 18:00		
HORA CIERRE: 21:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Correcciones sistemas operativos.			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 19 de Agosto de 2020		HORA INICIO: 18:00		
HORA CIERRE: 20:30				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Plan Estratégico"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 20 de Agosto de 2020	HORA INICIO: 15:00	HORA CIERRE: 17:00		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Plan Estratégico"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 21 de Agosto de 2020	HORA INICIO: 17:00	HORA CIERRE: 18:50		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Formación Empresarial "Plan Estratégico"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Sábado, 22 de Agosto de 2020	HORA INICIO: 15:30	HORA CIERRE: 18:20		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Base de Datos II "Punto 3 y 4"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Domingo, 23 de Agosto de 2020	HORA INICIO: 15:30	HORA CIERRE: 17:00		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Base de Datos II "Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Elaboración texto Inglés punto 4	Gonzalo Gularte	27/8/2020	SI
2	Traducción texto Inglés punto 4	Santiago Iriart Agustín Acosta	30/8/2020	SI
3	Menú Scripts	Gonzalo Gularte	17/9/2020	SI
4	Detalle y esquema de la forma en la cual se interconectan los puestos	Bruno Sunes	11/9/2020	NO

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 26 de Agosto de 2020		HORA INICIO: 18:00		
HORA CIERRE: 20:00				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Inglés "Punto 1 - Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 26 de Agosto de 2020		HORA INICIO: 18:30		
HORA CIERRE: 20:45				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programación "Código alta, baja, modificación y consulta"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 28 de Agosto de 2020		HORA INICIO: 17:00		
HORA CIERRE: 18:50				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programación "Código alta, baja, modificación y consulta"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 31 de Agosto de 2020		HORA INICIO: 18:00		
HORA CIERRE: 20:10				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Taller de Mantenimiento III "Servicios provistos por los servidores" Y Base de Datos "Permisos"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 01 de Setiembre de 2020		HORA INICIO: 15:00		
		HORA CIERRE: 17:30		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programacion III "Metodos alta, baja, modificacion y busqueda"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 02 de Setiembre de 2020		HORA INICIO: 17:00		
		HORA CIERRE: 21:15		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programacion III "Metodos alta, baja, modificacion y busqueda" y Sociología "Formulación de preguntas"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA		ACTA DE REUNIONES FORMALES		GSA
Fecha: Jueves, 03 de Setiembre de 2020		HORA INICIO: 15:00		HORA CIERRE: 19:05
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programacion III "Metodos alta, baja, modificacion y busqueda"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA		ACTA DE REUNIONES FORMALES		
Fecha: Viernes, 04 de Setiembre de 2020		HORA INICIO: 15:00		HORA CIERRE: 18:40
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programacion III "Metodos alta, baja, modificacion y busqueda" y T.M III "Cableado estructurado"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA		ACTA DE REUNIONES FORMALES		
Fecha: Sábado, 05 de Setiembre de 2020		HORA INICIO: 16:30	HORA CIERRE: 19:30	
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programacion III "Metodos alta, baja, modificacion y busqueda"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA		ACTA DE REUNIONES FORMALES		
Fecha: Lunes, 07 de Setiembre de 2020		HORA INICIO: 17:30	HORA CIERRE: 20:00	
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia (Justificada)		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programacion III "Metodos alta, baja, modificacion y busqueda"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 08 de Setiembre de 2020		HORA INICIO: 15:00		
HORA CIERRE: 17:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Formación Empresarial "Plan Operativo"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 08 de Setiembre de 2020		HORA INICIO: 20:00		
HORA CIERRE: 21:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Formación Empresarial "Plan Operativo"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 09 de Setiembre de 2020		HORA INICIO: 18:00		
		HORA CIERRE: 21:00		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 10 de Setiembre de 2020		HORA INICIO: 15:00		
		HORA CIERRE: 19:00		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 11 de Setiembre de 2020		HORA INICIO: 17:00		
HORA CIERRE: 19:10				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Domingo, 13 de Setiembre de 2020		HORA INICIO: 15:30		
HORA CIERRE: 17:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Detalle y esquema de la forma en la cual se interconectan los puestos	Bruno Sunes	16/9/2020	

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 14 de Setiembre de 2020		HORA INICIO: 18:00		
HORA CIERRE: 20:25				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunés - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Segunda entrega"			
2	Programación III "Segunda entrega"			
3	Inglés "Segunda entrega"			
4	Asignación de tareas			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Diagrama Warnier-Orr, Diagrama de estados.	Gonzalo Gularte	18/9/2020	SI
2	Programar RNE.	Zendhel Rosales	17/9/2020	SI
3	ABMC programación.	Santiago Iriart Gonzalo Gularte	17/9/2020	SI
4	Cálculo de materiales necesarios para la imp. de la red y el equip. Informático.	Bruno Sunés Zendhel Rosales	17/9/2020	SI
5	Direccionamiento IP usando VLSM para cada VLAN.	Bruno Sunés Agustín Acosta	17/9/2020	SI
6	Detalles de las UPS para el equipamiento crítico que interconecta las instituciones.	Bruno Sunés	16/9/2020	SI
7	Script de respaldo.	Agustín Acosta Gonzalo Gularte	17/9/2020	SI

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 16 de Setiembre de 2020		HORA INICIO: 18:00		
HORA CIERRE: 23:00				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 17 de Setiembre de 2020		HORA INICIO: 10:00		
HORA CIERRE: 00:00				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Segunda entrega"			
2	Programación "Segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 18 de Setiembre de 2020		HORA INICIO: 07:00		
HORA CIERRE: 20:45				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Finalización y entrega de carpetas de la segunda entrega			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Sábado 19 de Setiembre de 2020		HORA INICIO: 15:00		
HORA CIERRE: 20:50				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	A.D.A "Analisis Costo-Beneficio"			
2	Sistemas Operativos III "Respaldos"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA			
ACTA DE REUNIONES FORMALES			
Fecha: Domingo, 20 de Setiembre de 2020		HORA INICIO: 16:15	HORA CIERRE: 22:00
ASISTENTES			
1	Agustín Acosta - Integrante	Asistencia	
2	Santiago Iriart - Sub-Coordinador	Asistencia	
3	Gonzalo Gularte - Coordinador	Asistencia	
4	Bruno Sunes - Integrante	Asistencia	
5	Zendhel Rosales - Integrante	Asistencia	
TEMAS TRATADOS			
1	Sistemas Operativos III "Scripts de respaldos"		
COMPRISOS DE ESTA REUNIÓN			
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento
1	No hay compromisos asignados a integrantes.		
GSA			
ACTA DE REUNIONES FORMALES			
Fecha: Lunes, 21 de Setiembre de 2020		HORA INICIO: 16:00	HORA CIERRE: 22:00
ASISTENTES			
1	Agustín Acosta - Integrante	Asistencia	
2	Santiago Iriart - Sub-Coordinador	Asistencia	
3	Gonzalo Gularte - Coordinador	Asistencia	
4	Bruno Sunes - Integrante	Asistencia	
5	Zendhel Rosales - Integrante	Asistencia	
TEMAS TRATADOS			
1	A.D.A "Warnier-Orr"		
2	Proyecto "Plan de contingencias, GANTT Por asignatura"		
COMPRISOS DE ESTA REUNIÓN			
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento
1	No hay compromisos asignados a integrantes.		



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 22 de Setiembre de 2020		HORA INICIO: 16:00 HORA CIERRE: 20:30		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	A.D.A "UML, Análisis Costo-Beneficio"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	Diccionario de Datos	Bruno Sunes	23/9/2020	SI
2	Diagrama de estados	Gonzalo Gularte	23/9/2020	SI



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 23 de Setiembre de 2020		HORA INICIO: 10:00	HORA CIERRE: 14:00	
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia (Justificada)		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Proyecto "GANNT Y PERT TOTALES"			
2	A.D.A "Diagrama de Estados"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 23 de Setiembre de 2020		HORA INICIO: 17:00	HORA CIERRE: 20:00	
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Armado de carpetas "Proyecto y A.D.A"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 25 de Setiembre de 2020		HORA INICIO: 17:00		
HORA CIERRE: 18:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Calendario			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 28 de Setiembre de 2020		HORA INICIO: 17:30		
HORA CIERRE: 18:30				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Taller de Mantenimiento III "Contratación de servicio de INTERNET para la institución"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 1 de Octubre de 2020	HORA INICIO: 17:00	HORA CIERRE: 17:50		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Taller de Mantenimiento III "Contratación de servicio de INTERNET para la institución"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	"ABSTRACT" Inglés.	Agustin Acosta // Santiago Iriart // Gonzalo Gularte	6/10/2020	
2	"Análisis F.O.D.A" Inglés.	Agustin Acosta // Santiago Iriart // Gonzalo Gularte	4/10/2020	SI
3	"Guión presentación venta software" Inglés.	Agustin Acosta // Santiago Iriart // Gonzalo Gularte	8/10/2020	
4	"Correcciones segunda entrega" Sistemas Operativos.	Agustin Acosta Gonzalo Gularte	7/10/2020	

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 02 de Octubre de 2020	HORA INICIO: 15:30	HORA CIERRE: 19:00		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Plan de comercialización"			
2	Inglés "SWOT Analysis"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Sábado, 03 de Octubre de 2020		HORA INICIO: 18:30		
HORA CIERRE: 22:10				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Base de Datos II "Consultas SQL"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 05 de Octubre de 2020		HORA INICIO: 11:25		
HORA CIERRE: 14:00				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia(Justificada)		
2	Santiago Iriart - Sub-Coordinador	Asistencia(Justificada)		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia(Justificada)		
TEMAS TRATADOS				
1	Base de Datos II "Consultas SQL"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 05 de Octubre de 2020		HORA INICIO: 18:30 HORA CIERRE: 20:35		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Base de Datos II "Consultas SQL"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 06 de Octubre de 2020		HORA INICIO: 15:30		
HORA CIERRE: 17:00				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularde - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Programación III "Usuarios"			
2	Formación Empresarial "Plan de comercialización"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 07 de Octubre de 2020		HORA INICIO: 16:00		
HORA CIERRE: 17:00				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
3	Gonzalo Gularde - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Sistemas Operativos III "Correcciones segunda entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 07 de Octubre de 2020		HORA INICIO: 16:00 HORA CIERRE: 17:00		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
3	Gonzalo Gularde - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programacion III "Usuarios"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA		ACTA DE REUNIONES FORMALES		GSA
Fecha: Jueves, 08 de Octubre de 2020		HORA INICIO: 18:00		HORA CIERRE: 20:15
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Sistemas operativos "Control de logs del sistema"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA		ACTA DE REUNIONES FORMALES		
Fecha: Lunes, 12 de Octubre de 2020		HORA INICIO: 18:00		HORA CIERRE: 21:35
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Programación "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 13 de Octubre de 2020		HORA INICIO: 16:00		
HORA CIERRE: 20:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programación "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 14 de Octubre de 2020		HORA INICIO: 15:00		
HORA CIERRE: 17:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia (Justificada)		
2	Santiago Iriart - Sub-Coordinador	Asistencia (Justificada)		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Formación Empresarial "Plan Financiero"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 15 de Octubre de 2020		HORA INICIO: 16:00 HORA CIERRE: 19:40		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Formación Empresarial "Plan Financiero"			
2	Sistemas Operativos III "Control de logs scripts respaldo"			
3	Programación III "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 16 de Octubre de 2020		HORA INICIO: 16:00	HORA CIERRE: 18:30	
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia (Justificada)		
TEMAS TRATADOS				
1	Programación III "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 19 de Octubre de 2020		HORA INICIO: 18:00	HORA CIERRE: 21:15	
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programación III "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 20 de Octubre de 2020	HORA INICIO: 15:00	HORA CIERRE: 18:40		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programación III "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 21 de Octubre de 2020	HORA INICIO: 18:00	HORA CIERRE: 21:40		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Inglés "ABSTRACT"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 22 de Octubre de 2020		HORA INICIO: 15:00		HORA CIERRE: 21:50
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Presentación final proyecto			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 23 de Octubre de 2020		HORA INICIO: 15:30		HORA CIERRE: 17:10
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Presentación final proyecto			
2	Formación Empresarial "Plan Financiero"			
3	Taller de Mantenimiento III "Contratación de internet"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Domingo, 25 de Octubre de 2020		HORA INICIO: 18:00		
HORA CIERRE: 20:35				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia (Justificada)		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Presentación final proyecto			
2	Taller de mantenimiento III "Selección antivirus"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Lunes, 26 de Octubre de 2020	HORA INICIO: 18:00	HORA CIERRE: 20:00		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Presentación final proyecto			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 27 de Octubre de 2020	HORA INICIO: 15:00	HORA CIERRE: 20:10		
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia(Justificada)		
TEMAS TRATADOS				
1	Programación "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 28 de Octubre de 2020		HORA INICIO: 18:00		
HORA CIERRE: 21:05				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programación "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 29 de Octubre de 2020		HORA INICIO: 15:00		
HORA CIERRE: 21:30				
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	Programación "Tercera entrega"			
2	Formación empresarial "Plan comercialización"			
3	Sistemas operativos "Script actualización, script mysql, script ssh, script firewall"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA		
ACTA DE REUNIONES FORMALES		
Fecha: Viernes, 30 de Octubre de 2020		HORA INICIO: 15:00
ASISTENTES		
1	Agustin Acosta - Integrante	Asistencia
2	Santiago Iriart - Sub-Coordinador	Asistencia
3	Gonzalo Gularte - Coordinador	Asistencia
4	Bruno Sunes - Integrante	Asistencia
5	Zendhel Rosales - Integrante	Asistencia

TEMAS TRATADOS		
1	Programación "Tercera entrega"	
2	Sistemas operativos "Tercera entrega"	
3	Taller de Mantenimiento III "Correcciones de la segunda entrega"	

COMPRISOS DE ESTA REUNIÓN		
ITEM	ASUNTO	RESPONSABLE
1	No hay compromisos asignados a integrantes.	

GSA		
ACTA DE REUNIONES FORMALES		
Fecha: Sábado, 31 de Octubre de 2020		HORA INICIO: 15:00
ASISTENTES		

1	Agustín Acosta - Integrante	Asistencia
2	Santiago Iriart - Sub-Coordinador	Asistencia
3	Gonzalo Gularte - Coordinador	Asistencia
4	Bruno Sunes - Integrante	Asistencia
5	Zendhel Rosales - Integrante	Asistencia

TEMAS TRATADOS		
1	Sistemas operativos "Tercera entrega"	
2	Taller de Mantenimiento III "Correcciones de la segunda entrega, Manual de procedimientos ante fallas del sist."	

COMPRISOS DE ESTA REUNIÓN		
ITEM	ASUNTO	RESPONSABLE
1	No hay compromisos asignados a integrantes.	

GSA		
ACTA DE REUNIONES FORMALES		
Fecha: Domingo, 1 de Noviembre de 2020		HORA INICIO: 16:00
ASISTENTES		
1	Agustin Acosta - Integrante	Asistencia
2	Santiago Iriart - Sub-Coordinador	Asistencia
3	Gonzalo Gularte - Coordinador	Asistencia
4	Bruno Sunes - Integrante	Asistencia
5	Zendhel Rosales - Integrante	Asistencia

TEMAS TRATADOS		
1	Sistemas operativos "Tercera entrega"	
2	A.D.A "Estudio de herramientas C.A.S.E"	
3	Inglés "Corrección ABSTRACT"	
COMPRISOS DE ESTA REUNIÓN		
ITEM	ASUNTO	RESPONSABLE
1	No hay compromisos asignados a integrantes.	

GSA		
ACTA DE REUNIONES FORMALES		
Fecha: Lunes, 2 de Noviembre de 2020		HORA INICIO: 15:30
ASISTENTES		
1	Agustin Acosta - Integrante	Asistencia
2	Santiago Iriart - Sub-Coordinador	Asistencia
3	Gonzalo Gularte - Coordinador	Asistencia
4	Bruno Sunes - Integrante	Asistencia
5	Zendhel Rosales - Integrante	Asistencia
TEMAS TRATADOS		
1	Sistemas operativos "Video explicativo del uso de los Scripts"	
2	A.D.A "Carta Jerárquica"	
COMPRISOS DE ESTA REUNIÓN		
ITEM	ASUNTO	RESPONSABLE
1	No hay compromisos asignados a integrantes.	



GSA		GSA		
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 3 de Noviembre de 2020		HORA INICIO: 15:00		
HORA CIERRE: 18:00				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularde - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia(Justificada)		
TEMAS TRATADOS				
1	Taller de Mantenimiento III "Manual de procedimientos ante fallas del sist., Selección software de monitoreo"			
2	Programación "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA		GSA		
ACTA DE REUNIONES FORMALES				
Fecha: Martes, 3 de Noviembre de 2020		HORA INICIO: 20:00		
HORA CIERRE: 21:30				
ASISTENTES				
1	Agustin Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularde - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia(Justificada)		
TEMAS TRATADOS				
1	Inglés "Guion presentación"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			

GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Miércoles, 4 de Noviembre de 2020	HORA INICIO: 15:00	HORA CIERRE: 21:15		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	A.D.A "Manual de administrador"			
2	Programación III "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			
GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Jueves, 5 de Noviembre de 2020	HORA INICIO: 15:00	HORA CIERRE: 22:10		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	A.D.A "Manual de usuario"			
2	Programación III "Tercera entrega"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



GSA				
ACTA DE REUNIONES FORMALES				
Fecha: Viernes, 6 de Noviembre de 2020		HORA INICIO: 17:00 HORA CIERRE: 22:20		
ASISTENTES				
1	Agustín Acosta - Integrante	Asistencia		
2	Santiago Iriart - Sub-Coordinador	Asistencia		
3	Gonzalo Gularte - Coordinador	Asistencia		
4	Bruno Sunes - Integrante	Asistencia		
5	Zendhel Rosales - Integrante	Asistencia		
TEMAS TRATADOS				
1	A.D.A "Manual de usuario"			
2	Programación III "Tercera entrega"			
3	Proyecto "Diagrama GANTT"			
COMPRISOS DE ESTA REUNIÓN				
ITEM	ASUNTO	RESPONSABLE	Fecha máxima de cumplimiento	Cumplido SI/NO
1	No hay compromisos asignados a integrantes.			



Anexo III: Manual de administrador

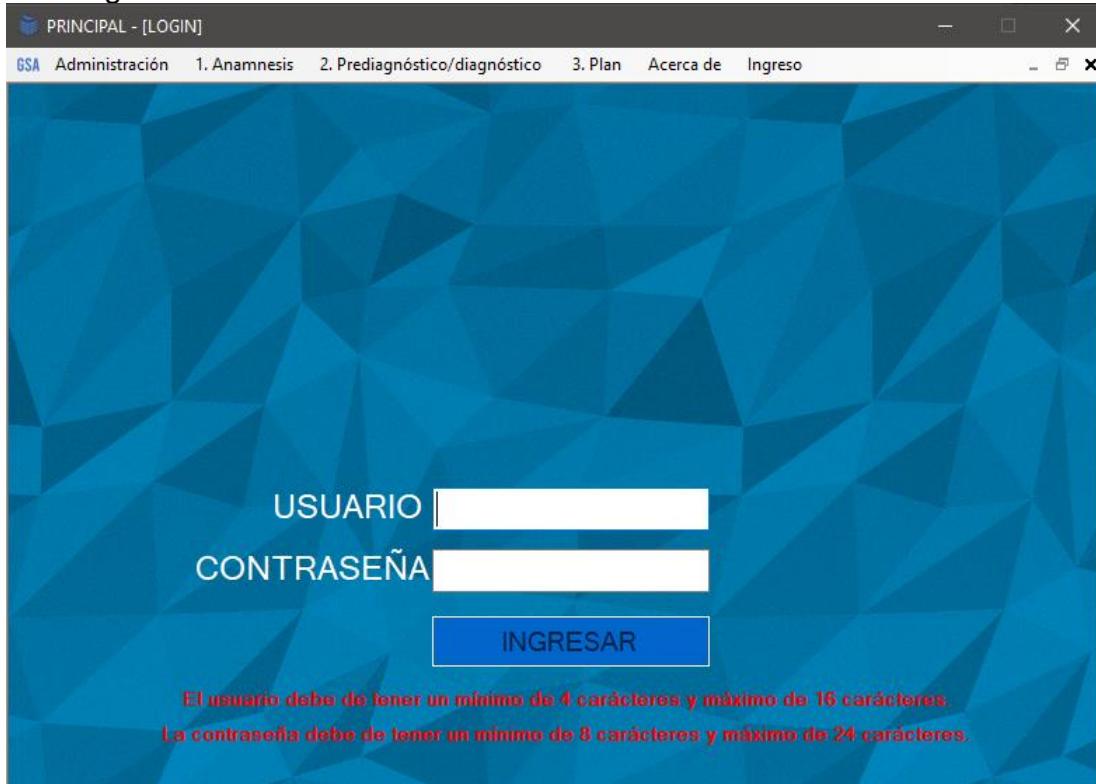
Login

Para ingresar al sistema se deberá de dirigir al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Ingreso” y luego ingresar a la opción de “Login”.



-Si la sesión no está iniciada

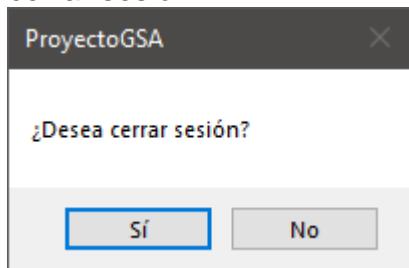
Deberá de ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, estos datos deberán de pertenecer a un usuario que ya esté registrado en el sistema.





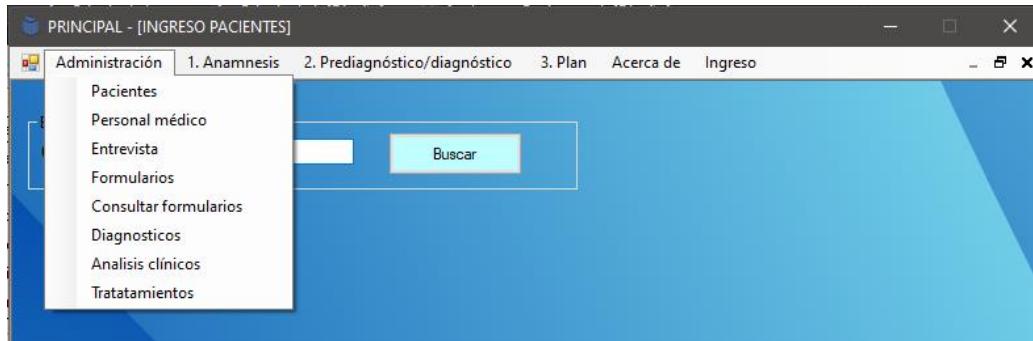
-Si la sesión está iniciada

El sistema desplegará una ventana emergente la cual preguntará si desea cerrar sesión.

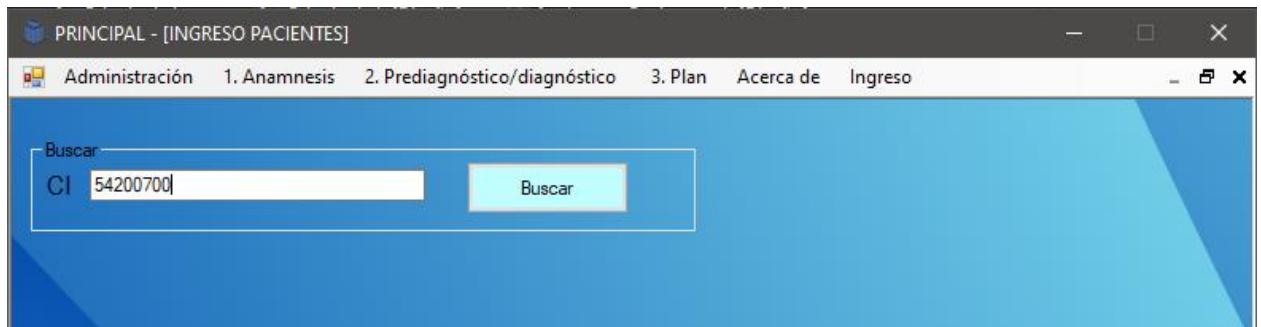


Administración - Pacientes

Para ingresar un paciente deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Pacientes”.



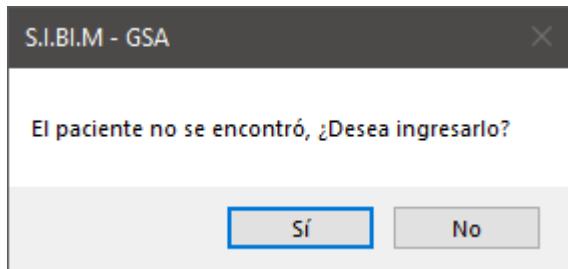
Una vez dentro de “Pacientes” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



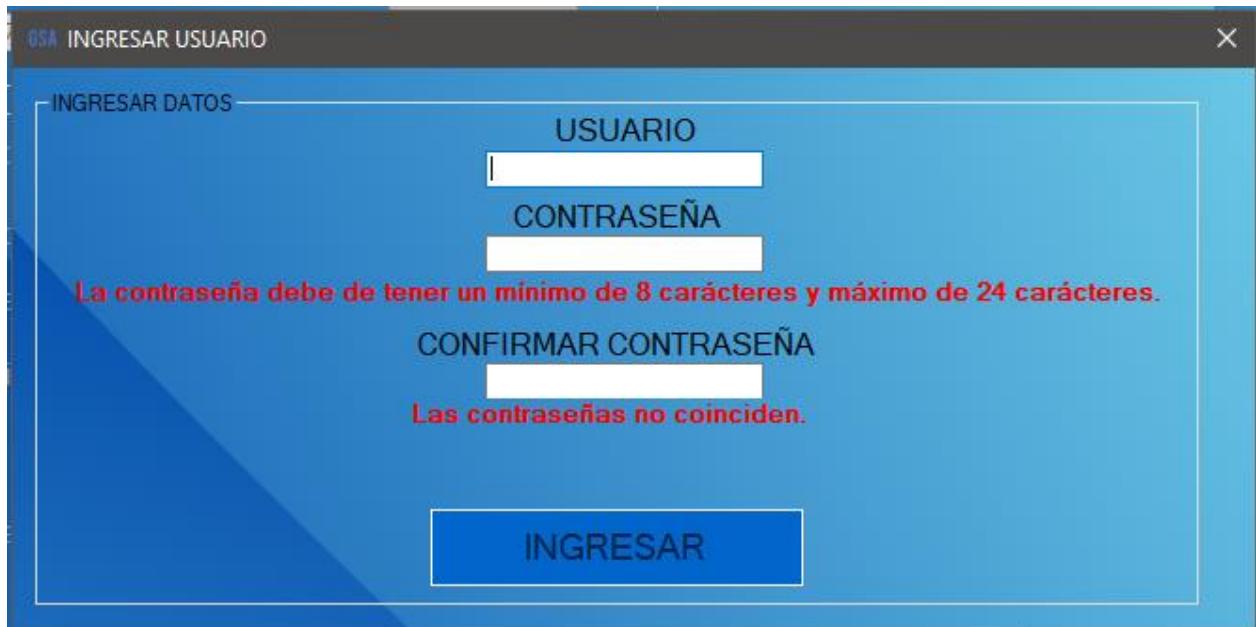
Ya ingresada la cédula, pueden surgir posibles alternativas, una si el paciente no existe, otra si el paciente existe, la tercera si el paciente está ingresado como personal médico y cuarta si el paciente fue dado de baja.

-Si el paciente no existe

Aparecerá una ventana emergente mencionando que el paciente no se encontró y si se desea ingresar al mismo.



Al seleccionar la opción “Sí” se desplegará una ventana emergente titulada “INGRESAR USUARIO” donde se tendrá que ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, estos datos son los que se utilizarán al iniciar sesión en el programa.



Una vez ingresados los datos solicitados, deberá de pulsar el botón “INGRESAR” y automáticamente se cerrará la ventana y se le habilitará el ingreso de información perteneciente al paciente.

PRINCIPAL - [INGRESO PACIENTES]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

Buscar
CI 54200700 Buscar

Datos

Primer nombre	Gonzalo	Segundo nombre	Lisandro
Primer apellido	Gularde	Segundo apellido	Atola
Fecha nacimiento	2002/12/02 18:52:34	Edad	17 años, 11 meses y 2 días
Estado civil	Soltero/a	Sexo	M
E-Mail	gonzalogularatemvd@gmail.com	Calle	Quijote
Nº Puerta	2885	Esquina	Dr. Gustavo Gallinal

Ingresar Eliminar Modificar Cancelar

Ya ingresados los datos solicitados deberá de pulsar el botón “Ingresar”, una vez pulsado el sistema le notificará si el paciente se ingresó exitosamente o no, si este se ingresó correctamente aparecerá una ventana emergente titulada “INGRESAR TELEFONO” en donde tendrá que ingresar el o los números pertenecientes al paciente.

GSA INGRESAR TELEFONO

INGRESAR DATOS

Teléfono 094742318 Agregar Quitar

Una vez ingresado el número de teléfono deberá presionar el botón “Agregar” para confirmar el ingreso del teléfono, en el caso de que se haya ingresado un teléfono erróneo, podrá seleccionar de la lista desplegable y presionar el botón “Quitar” para eliminar el teléfono del paciente. Ya ingresados el o los número del paciente podrá proceder a cerrar la ventana emergente.

-Si el paciente existe

El sistema cargará los datos pertenecientes al paciente con la cédula de identidad ingresada previamente y se habilitarán las funciones de eliminar o modificar al paciente.

PRINCIPAL - [INGRESO PACIENTES]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

Buscar
CI 54200700 Buscar

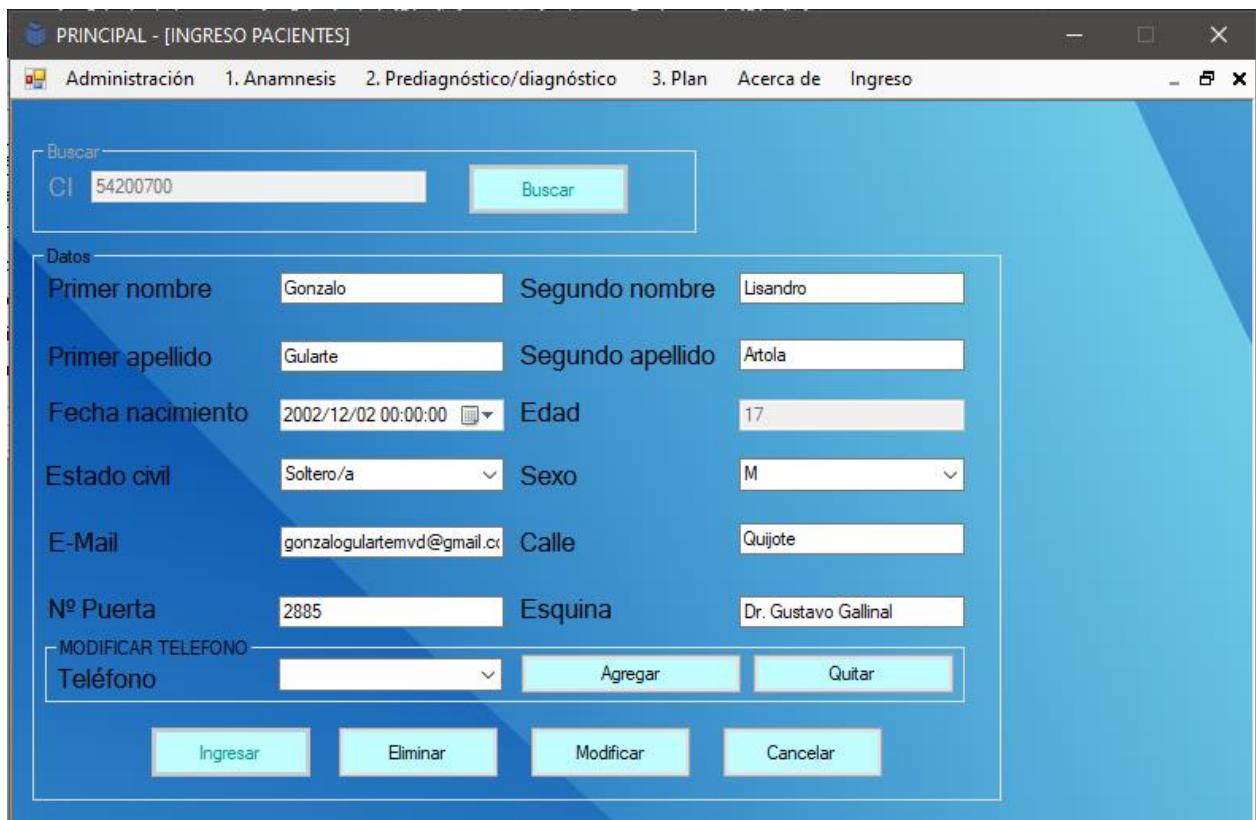
Datos

Primer nombre	Gonzalo	Segundo nombre	Lisandro
Primer apellido	Gularde	Segundo apellido	Artola
Fecha nacimiento	2002/12/02 00:00:00	Edad	17
Estado civil	Soltero/a	Sexo	M
E-Mail	gonzalogularatemvd@gmail.com	Calle	Quijote
Nº Puerta	2885	Esquina	Dr. Gustavo Gallinal

MODIFICAR TELEFONO

Teléfono	Agregar	Quitar
----------	---------	--------

Ingresar Eliminar Modificar Cancelar

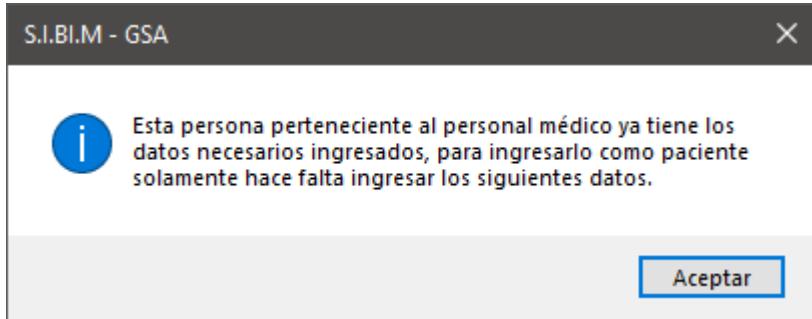


En el caso que desee eliminar el paciente se deberá de pulsar el botón de "Eliminar", si desea modificar el paciente deberá de reemplazar la información de los cuadros de texto por la que se desee modificar y posteriormente deberá de pulsar el botón de "Modificar".

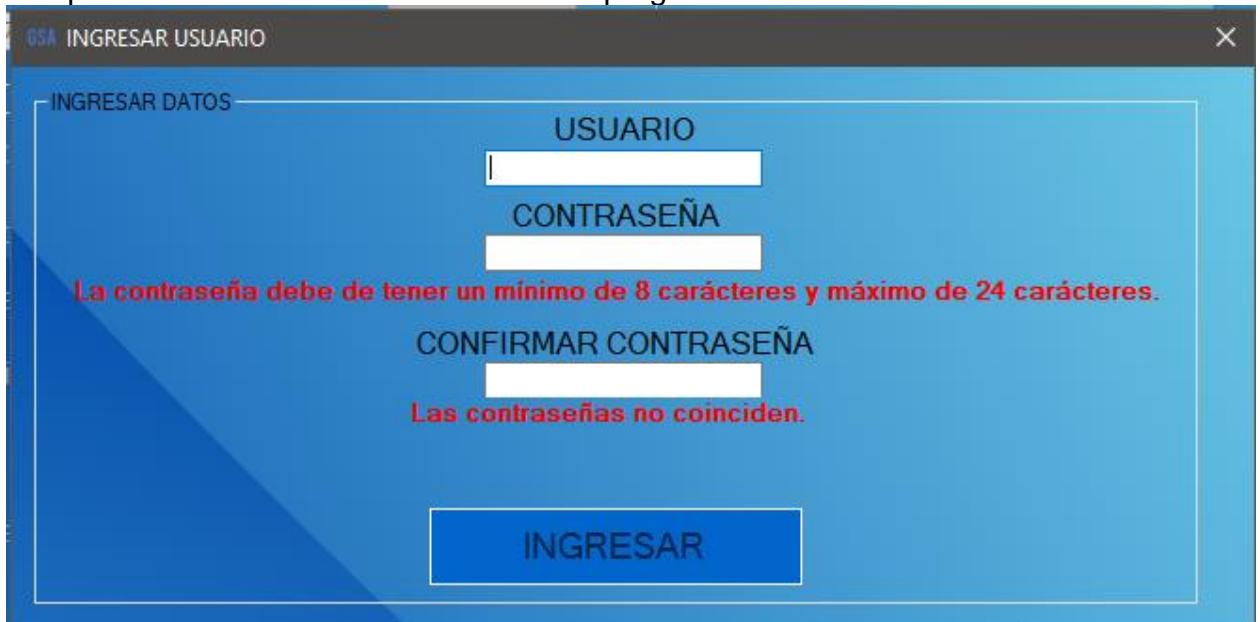
-Si el paciente está ingresado como personal médico
En este caso el sistema desplegará una ventana emergente en donde se le informará que la persona con la cédula de identidad ingresada previamente está dentro del personal médico y solamente será necesario ingresar



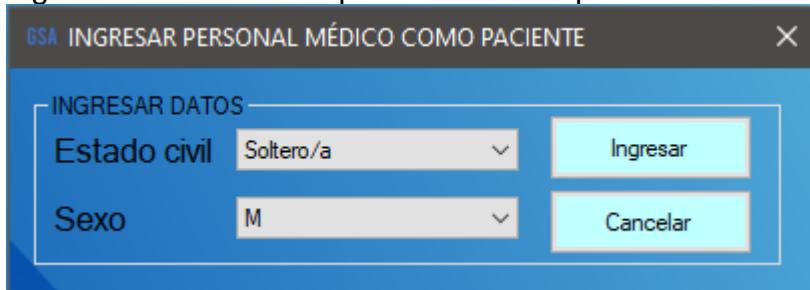
determinados datos.



Al presionar la opción “Aceptar” se desplegará una ventana emergente titulada “INGRESAR USUARIO” donde se tendrá que ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, estos datos son los que se utilizarán al iniciar sesión en el programa.



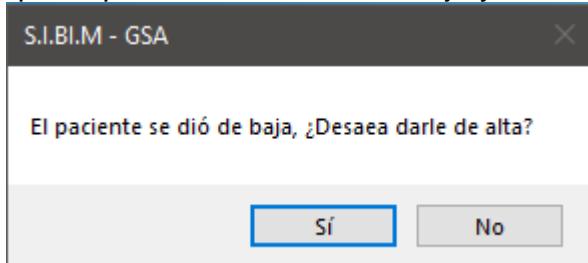
Una vez ingresados los datos solicitados, deberá de pulsar el botón “INGRESAR” y automáticamente se cerrará la ventana y se le habilitará el ingreso de información perteneciente al paciente.



Ya ingresados los datos solicitados deberá de pulsar el botón “Ingresar”, una vez pulsado el sistema le notificará si el paciente se ingresó exitosamente o no.

-Si el paciente existe pero fue dado de baja

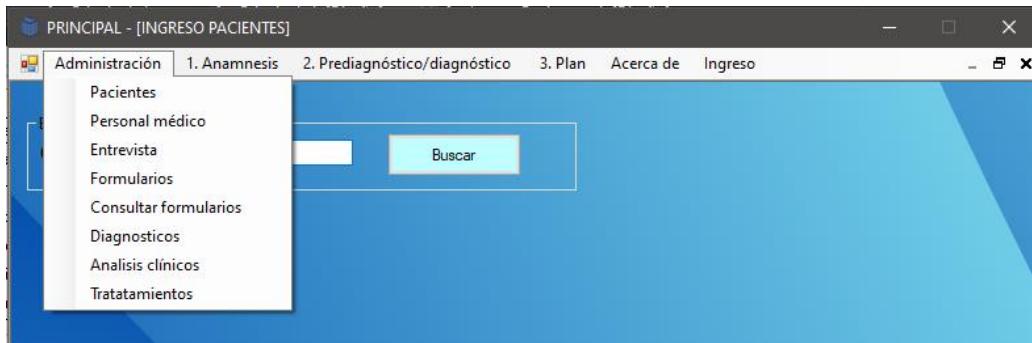
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



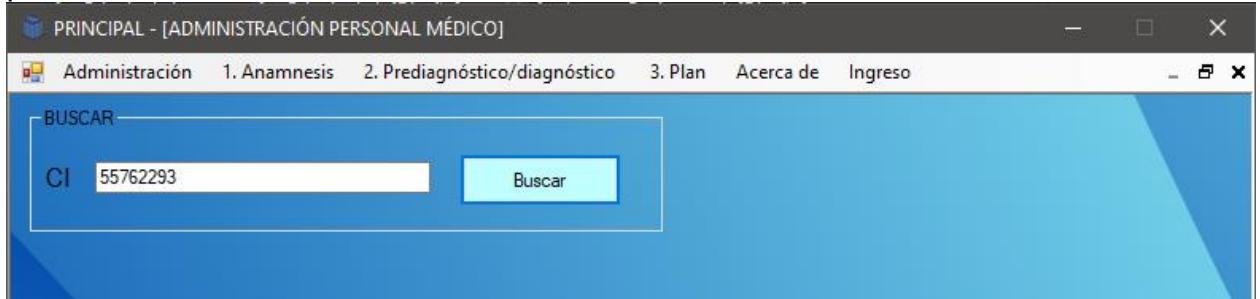
Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

Administración - Personal médico

Para ingresar un médico o auxiliar deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Personal médico”



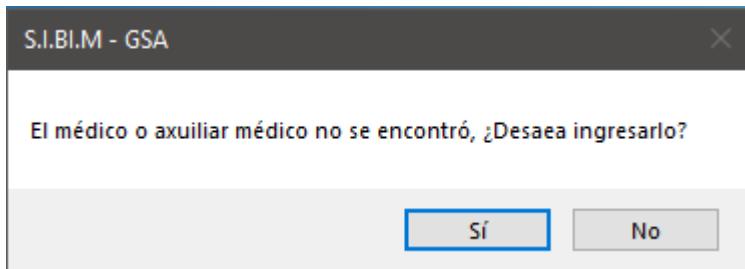
Una vez dentro de “Personal médico” será necesario previamente ingresar la cédula de identidad del médico o auxiliar, sin puntos ni guiones y luego presionar el botón de “Buscar”.



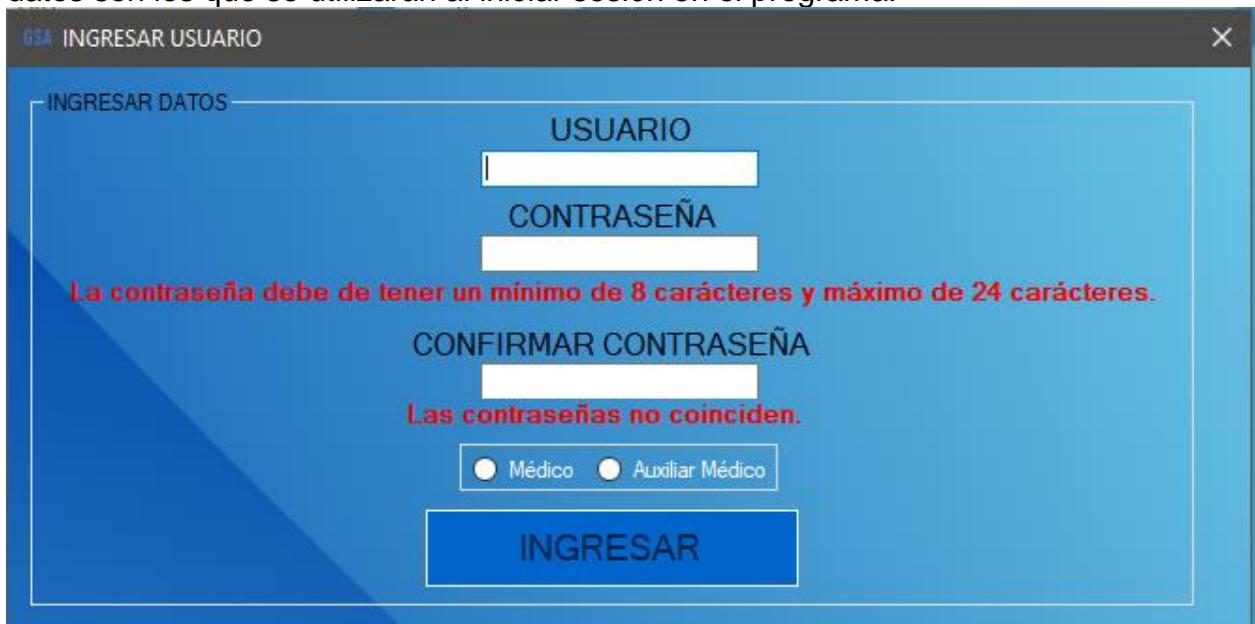
Ya ingresada la cédula, pueden surgir posibles alternativas, una si el médico o auxiliar médico no existe, otra si el médico o auxiliar médico existe, la tercera si el médico o auxiliar médico está ingresado como paciente y cuarta si el médico o auxiliar médico fue dado de baja.

-Si el médico o auxiliar médico no existe

Aparecerá una ventana emergente mencionando que el médico o auxiliar médico no se encontró y si se desea ingresar al mismo.



Al seleccionar la opción “Sí” se desplegará una ventana emergente titulada “INGRESAR USUARIO” donde se tendrá que ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, también se deberá de indicar si el personal médico es médico o auxiliar médico, estos datos son los que se utilizarán al iniciar sesión en el programa.



Una vez ingresados los datos solicitados, deberá pulsar el botón “INGRESAR” y automáticamente se cerrará la ventana y se le habilitará el ingreso de información perteneciente al médico o auxiliar médico.

PRINCIPAL - [ADMINISTRACIÓN PERSONAL MÉDICO]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR:

CI: 55762293 Buscar

INGRESAR DATOS

Tipo personal	<input checked="" type="radio"/> Médico	<input type="radio"/> Auxiliar Médico	
Primer nombre	Santiago	Segundo nombre	Miguel
Primer apellido	Iriart	Segundo apellido	Guibemau
Fecha nacimiento	2002/10/22 19:21:24	Edad	18 años, 0 meses y 13 días
Calle	Suecia	Nº Puerta	1719
Esquina	Grecia	E-Mail	a@edu.uy
Especialidad	Traumatólogo		

Ingresar Eliminar Modificar Cancelar

Ya ingresados los datos solicitados deberá de pulsar el botón “Ingresar”, una vez pulsado el sistema le notificará si el médico o auxiliar médico se ingresó exitosamente o no, si este se ingresó correctamente aparecerá una ventana emergente titulada “INGRESAR TELEFONO” en donde tendrá que ingresar el o los números pertenecientes al médico o auxiliar médico.

GSA INGRESAR TELEFONO

INGRESAR DATOS

Teléfono: 093982489 Agregar Quitar

Una vez ingresado el número de teléfono deberá presionar el botón “Agregar” para confirmar el ingreso del teléfono, en el caso de que se haya ingresado un teléfono erróneo, podrá seleccionar de la lista desplegable y presionar el botón “Quitar” para eliminar el teléfono del médico o auxiliar médico. Ya ingresados el o los números del médico o auxiliar médico podrá proceder a cerrar la ventana emergente.



-Si el médico o auxiliar médico existe

El sistema cargará los datos pertenecientes al médico o auxiliar médico con la cédula de identidad ingresada previamente y se habilitarán las funciones de eliminar o modificar al médico o auxiliar médico.

PRINCIPAL - [ADMINISTRACIÓN PERSONAL MÉDICO]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR

CI: 55762293 Buscar

INGRESAR DATOS

Tipo personal Médico Auxiliar Médico

Primer nombre Santiago Segundo nombre Santiago

Primer apellido Iriart Segundo apellido Guibemau

Fecha nacimiento 2002/10/22 00:00:00 Edad 18

Calle Suecia Nº Puerta 1719

Esquina Grecia E-Mail 093982489

Especialidad Traumatólogo

MODIFICAR TELEFONO

Teléfono Agregar Quitar

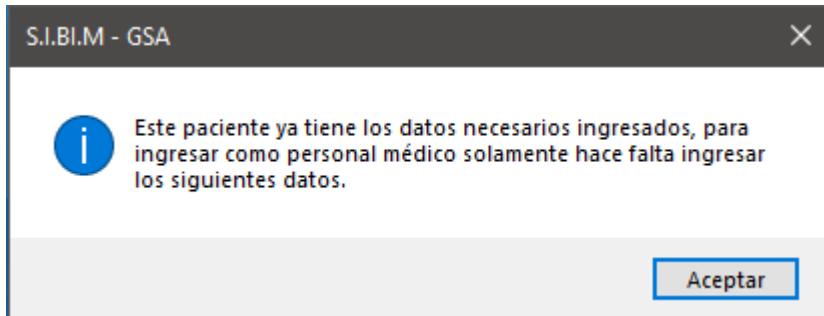
Ingresar Eliminar Modificar Cancelar

En el caso que desee eliminar el médico o auxiliar médico se deberá de pulsar el botón de "Eliminar", si desea modificar el médico o auxiliar médico deberá de reemplazar la información de los cuadros de texto por la que se deseé modificar y posteriormente deberá de pulsar el botón de "Modificar".

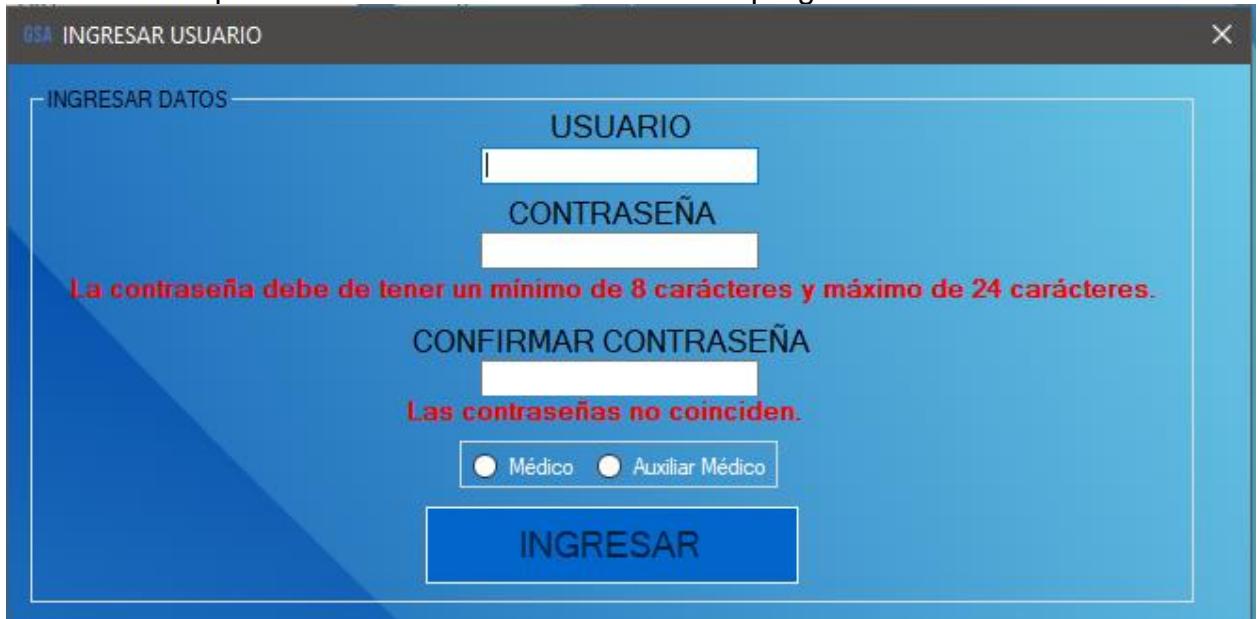


-Si el médico o auxiliar médico está ingresado paciente

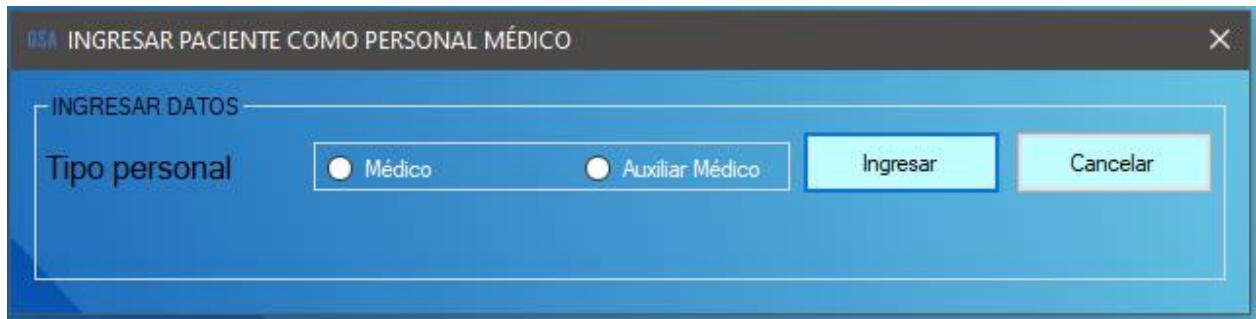
En este caso el sistema desplegará una ventana emergente en donde se le informará que la persona con la cédula de identidad ingresada previamente está dentro del personal médico y solamente será necesario ingresar determinados datos.



Al presionar la opción “Aceptar” se desplegará una ventana emergente titulada “INGRESAR USUARIO” donde se tendrá que ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, también se deberá de indicar si el personal médico es médico o auxiliar médico, estos datos son los que se utilizarán al iniciar sesión en el programa.

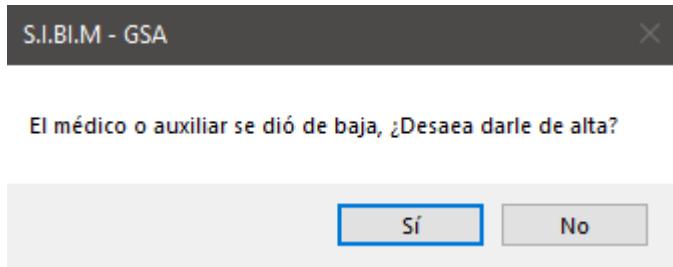


Una vez ingresados los datos solicitados, deberá pulsar el botón “INGRESAR” y automáticamente se cerrará la ventana y se le habilitará el ingreso de información perteneciente al médico o auxiliar médico.



Ya ingresados los datos solicitados deberá de pulsar el botón “Ingresar”, una vez pulsado el sistema le notificará si el médico o auxiliar médico se ingresó exitosamente o no.

-Si el médico o auxiliar médico existe pero fue dado de baja
Para este caso el sistema desplegará una ventana emergente informando de que el médico o auxiliar médico fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón “Sí” el sistema notificará si el médico o auxiliar médico se dio de alta exitosamente o no.



Anexo IV: Manual de usuario

MANUAL DE USUARIO Usuario - Paciente

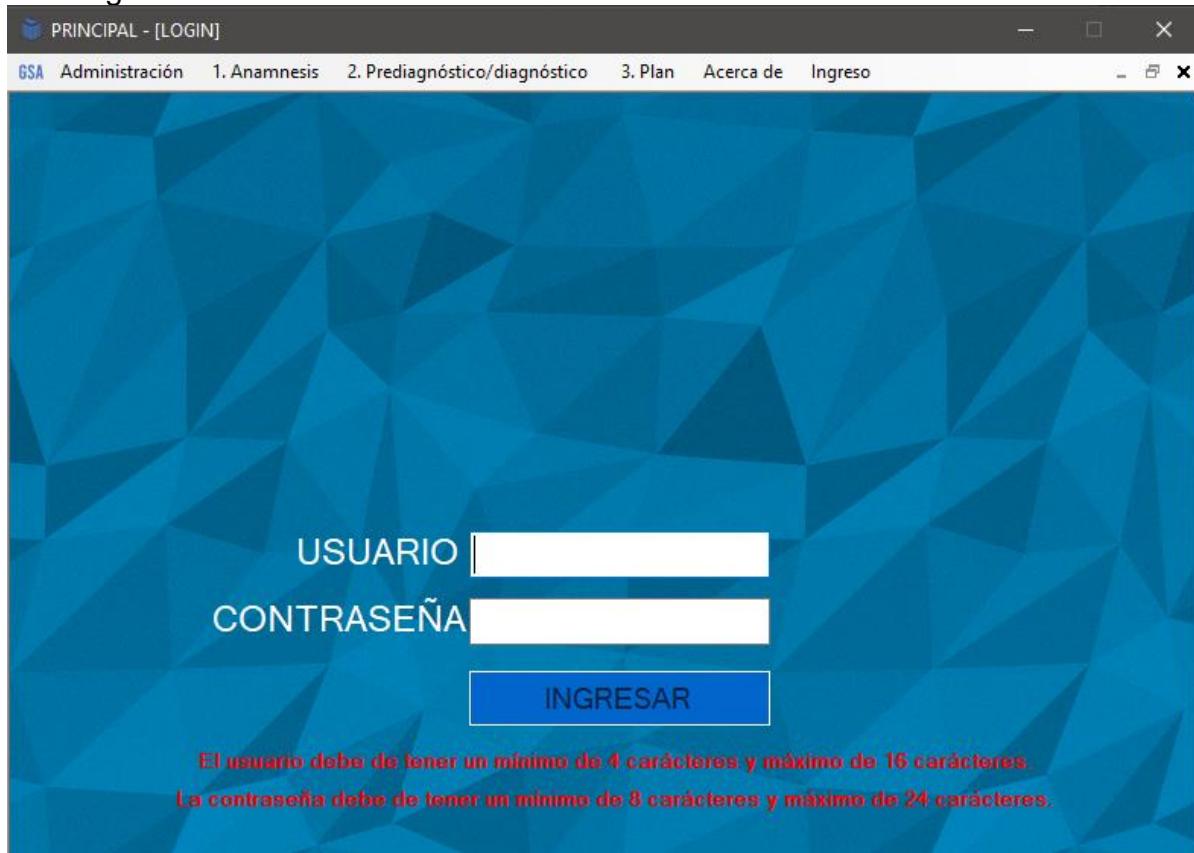
Login

Para ingresar al sistema se deberá dirigir al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Ingreso” y luego ingresar a la opción de “Login”.



-Si la sesión no está iniciada

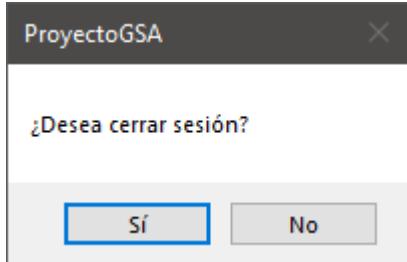
Deberá de ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, estos datos deberán de pertenecer a un usuario que ya esté registrado en el sistema.





-Si la sesión está iniciada

El sistema desplegará una ventana emergente la cual preguntará si desea cerrar sesión.





Consultar análisis clínicos

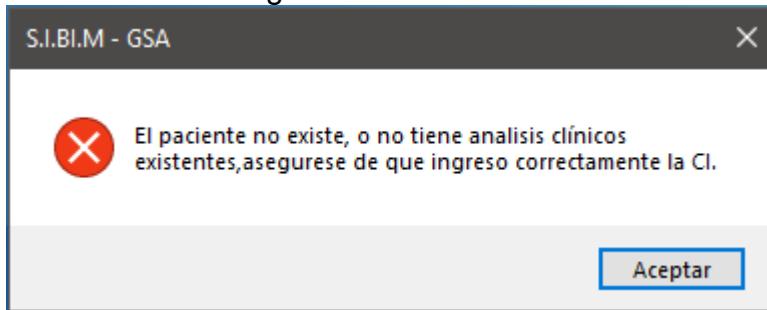
Para consultar un análisis clínico deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “3. Plan” y luego ingresar a la opción de “3.3 Consultar análisis clínicos”.



Una vez dentro de “3.3 Consultar análisis clínicos” será necesario previamente ingresar su cédula de identidad, sin puntos ni guiones y luego presionar el botón de “Buscar”.

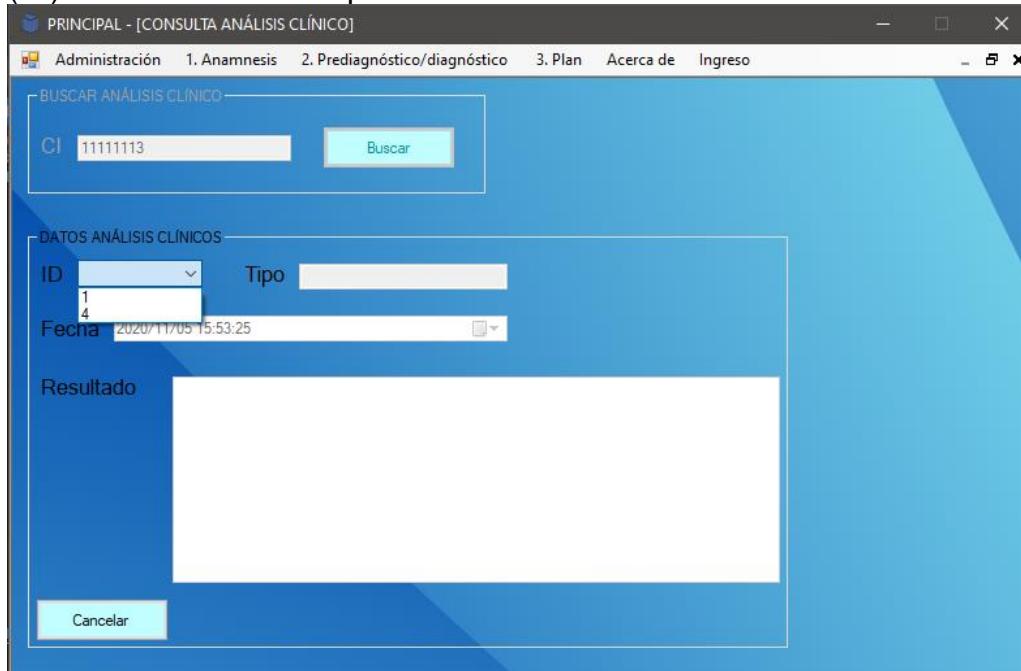


-Si el paciente no existe o no tiene análisis clínicos registrados
En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente o el paciente con dicha cédula de identidad no tiene análisis clínicos registrados en la institución médica.

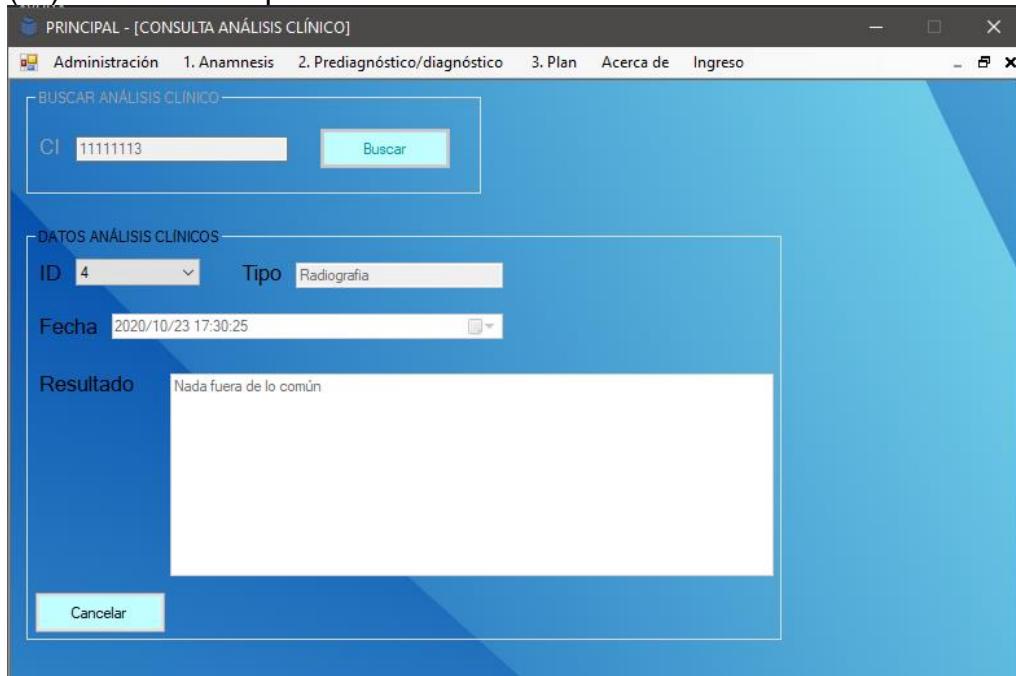


-Si el paciente existe y tiene análisis clínicos registrados

Ya ingresada la cédula, se habilitará una agrupación de datos pertenecientes a un análisis clínico en donde tendremos que seleccionar el número identificador (ID) del análisis clínico que deseamos consultar.



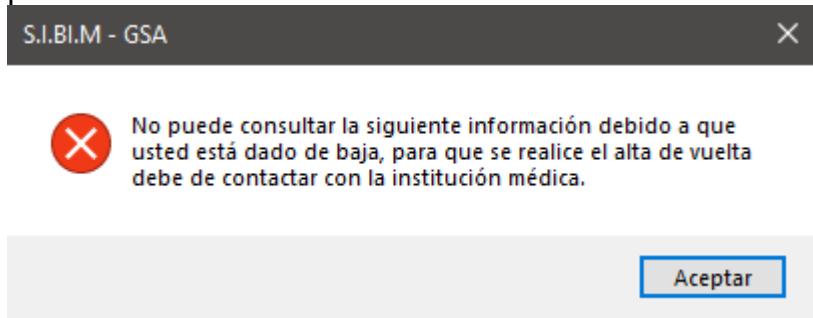
Una vez seleccionado el número identificador (ID) el sistema cargará en pantalla los datos pertenecientes al análisis clínico con el número identificador (ID) seleccionado previamente.





-Si el paciente existe pero fue dado de baja

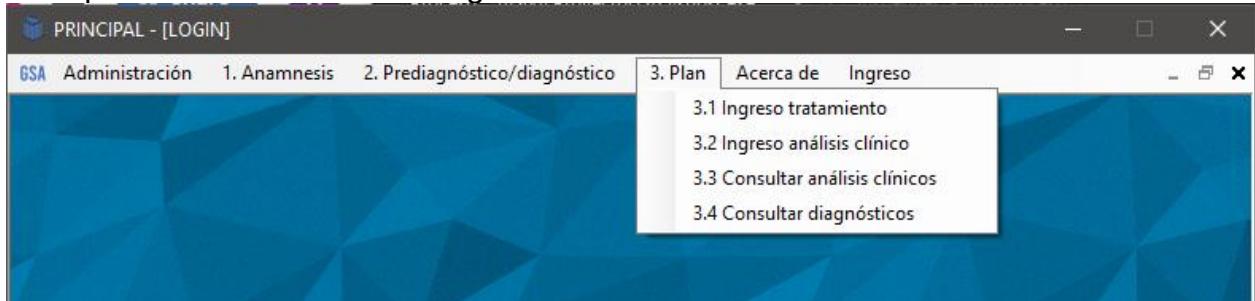
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y que debe contactar con la institución médica para realizar el alta del mismo.



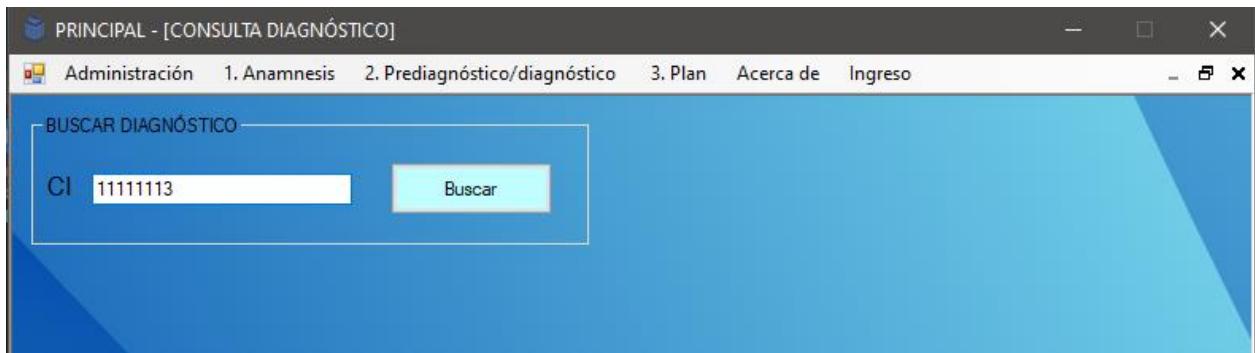


Consultar diagnósticos

Para consultar un diagnóstico deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “3. Plan” y luego ingresar a la opción de “3.4 Consultar diagnósticos”.

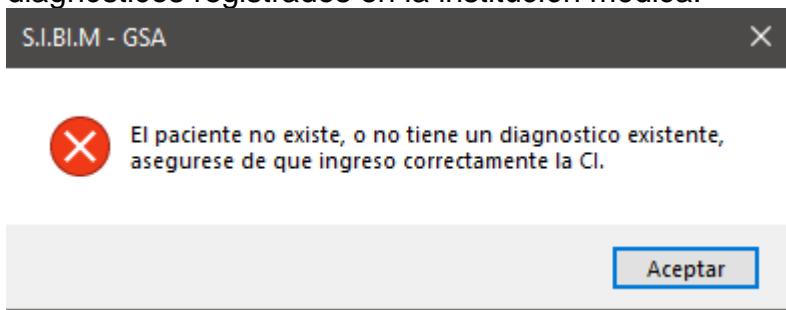


Una vez dentro de “3.3 Consultar diagnósticos” será necesario previamente ingresar su cédula de identidad, sin puntos ni guiones y luego presionar el botón de “Buscar”.



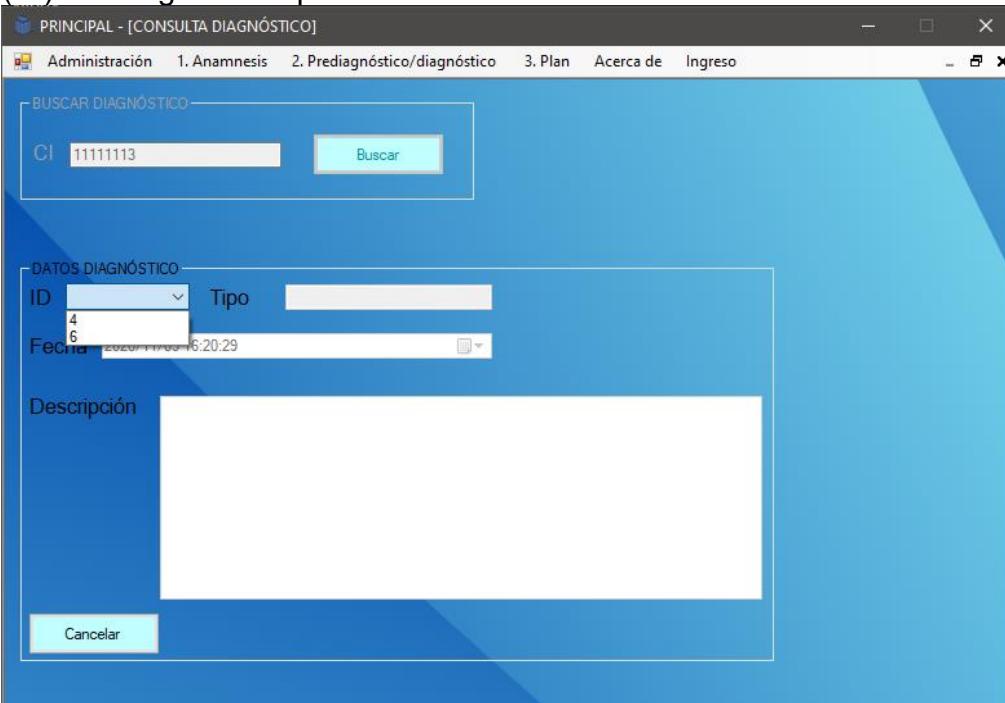
-Si el paciente no existe o no tiene diagnósticos registrados

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente o el paciente con dicha cédula de identidad no tiene diagnósticos registrados en la institución médica.



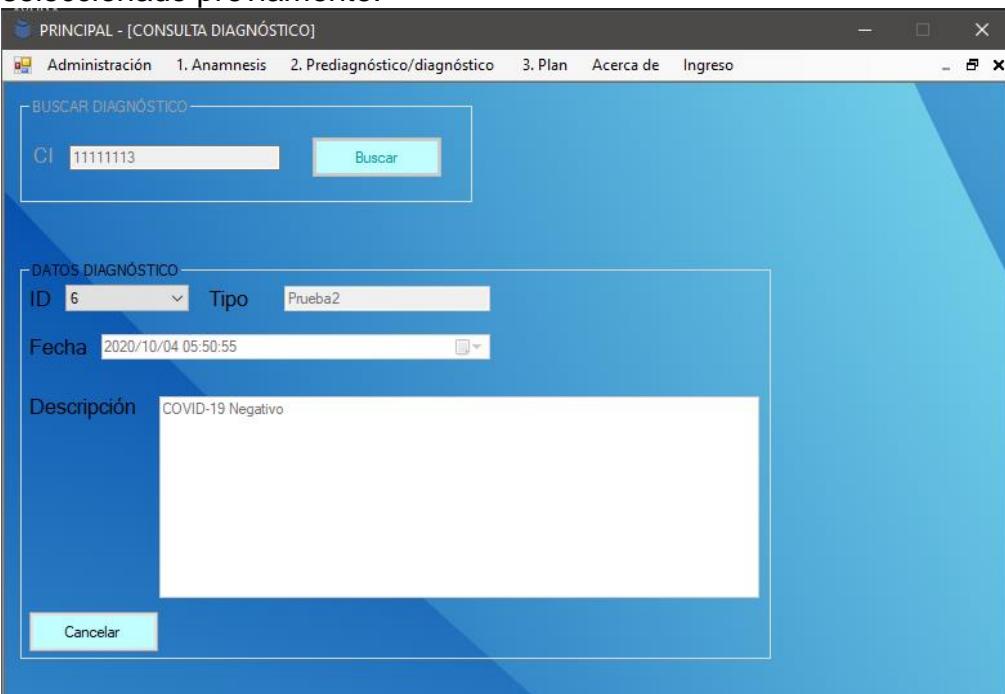
-Si el paciente existe y tiene diagnósticos registrados

Ya ingresada la cédula, se habilitará una agrupación de datos pertenecientes a un diagnóstico en donde tendremos que seleccionar el número identificador (ID) del diagnóstico que deseamos consultar.



The screenshot shows a software window titled "PRINCIPAL - [CONSULTA DIAGNÓSTICO]". The menu bar includes "Administración", "1. Anamnesis", "2. Prediagnóstico/diagnóstico", "3. Plan", "Acerca de", and "Ingreso". The main area has a header "BUSCAR DIAGNÓSTICO" with a search field containing "CI 11111113" and a "Buscar" button. Below this is a section titled "DATOS DIAGNÓSTICO" with fields for "ID" (dropdown menu showing "4" and "6"), "Tipo" (text input "Prueba2"), and "Fecha" (date input "2020/10/03 16:20:29"). A large text area labeled "Descripción" contains "COVID-19 Negativo". At the bottom left is a "Cancelar" button.

Una vez seleccionado el número identificador (ID) el sistema cargará en pantalla los datos pertenecientes al diagnóstico con el número identificador (ID) seleccionado previamente.

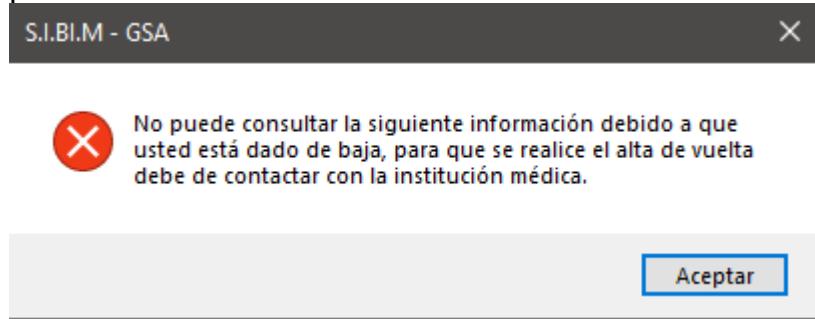


The screenshot shows the same software window as the previous one, but now the "ID" dropdown is set to "6". The "Descripción" text area now displays "COVID-19 Negativo". All other fields remain the same as in the first screenshot.



-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y que debe contactar con la institución médica para realizar el alta del mismo.





Usuario - Médico

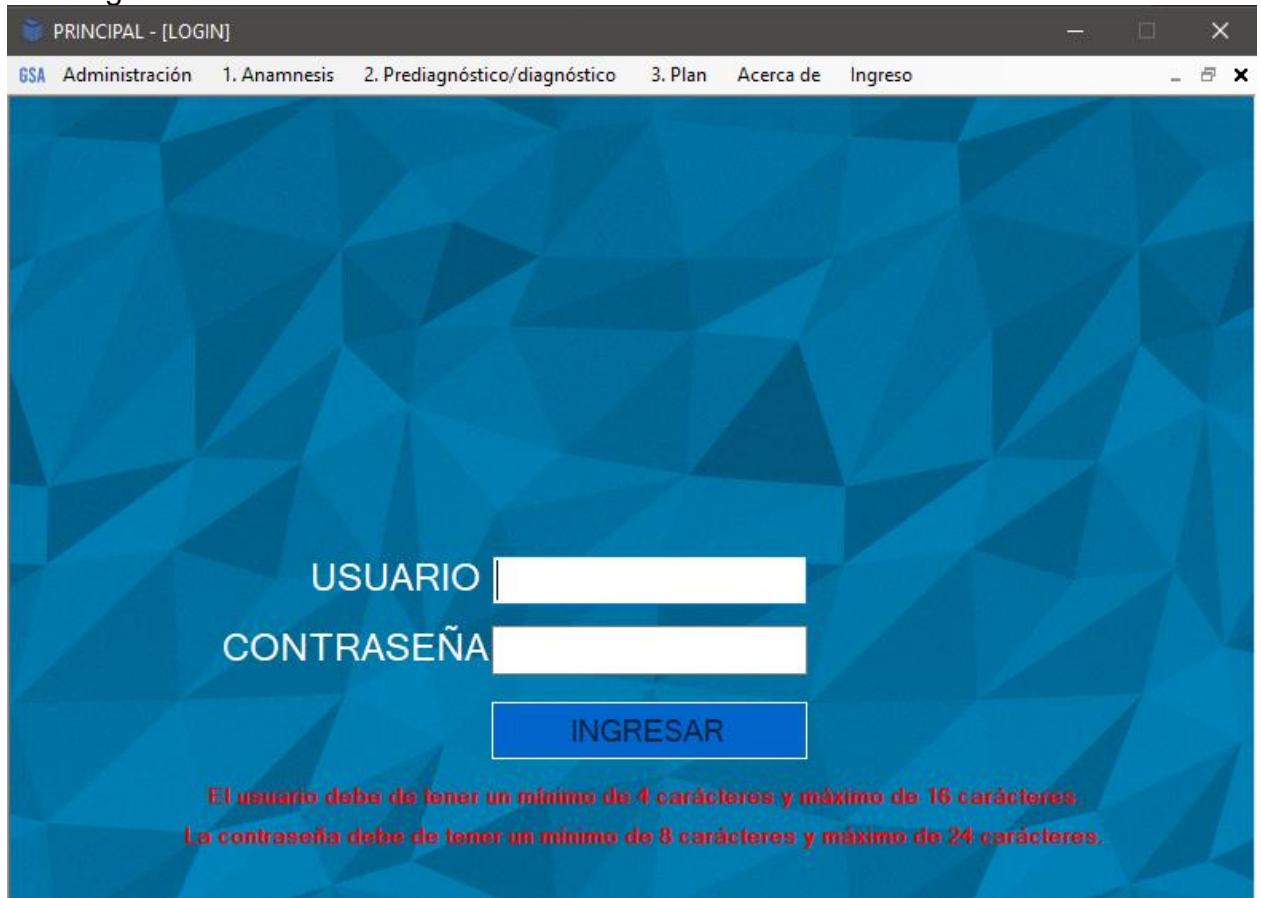
Login

Para ingresar al sistema se deberá de dirigir al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Ingreso” y luego ingresar a la opción de “Login”.



-Si la sesión no está iniciada

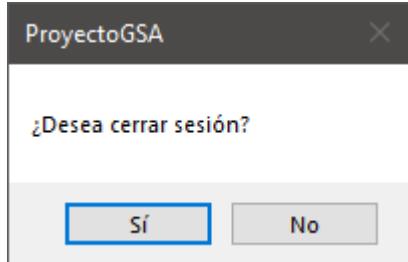
Deberá de ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, estos datos deberán de pertenecer a un usuario que ya esté registrado en el sistema.





-Si la sesión está iniciada

El sistema desplegará una ventana emergente la cual preguntará si desea cerrar sesión.



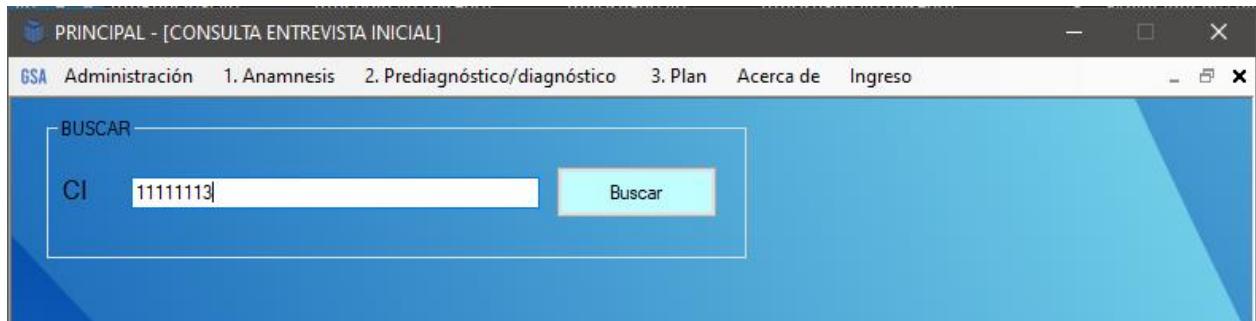


Administración - Entrevista

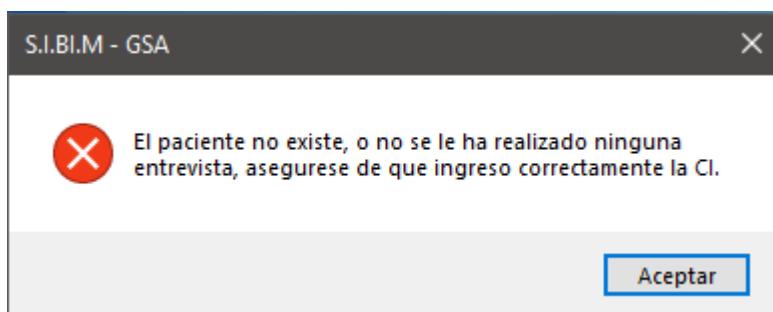
Para consultar una entrevista deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Entrevista”.



Una vez dentro de “Entrevista” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.

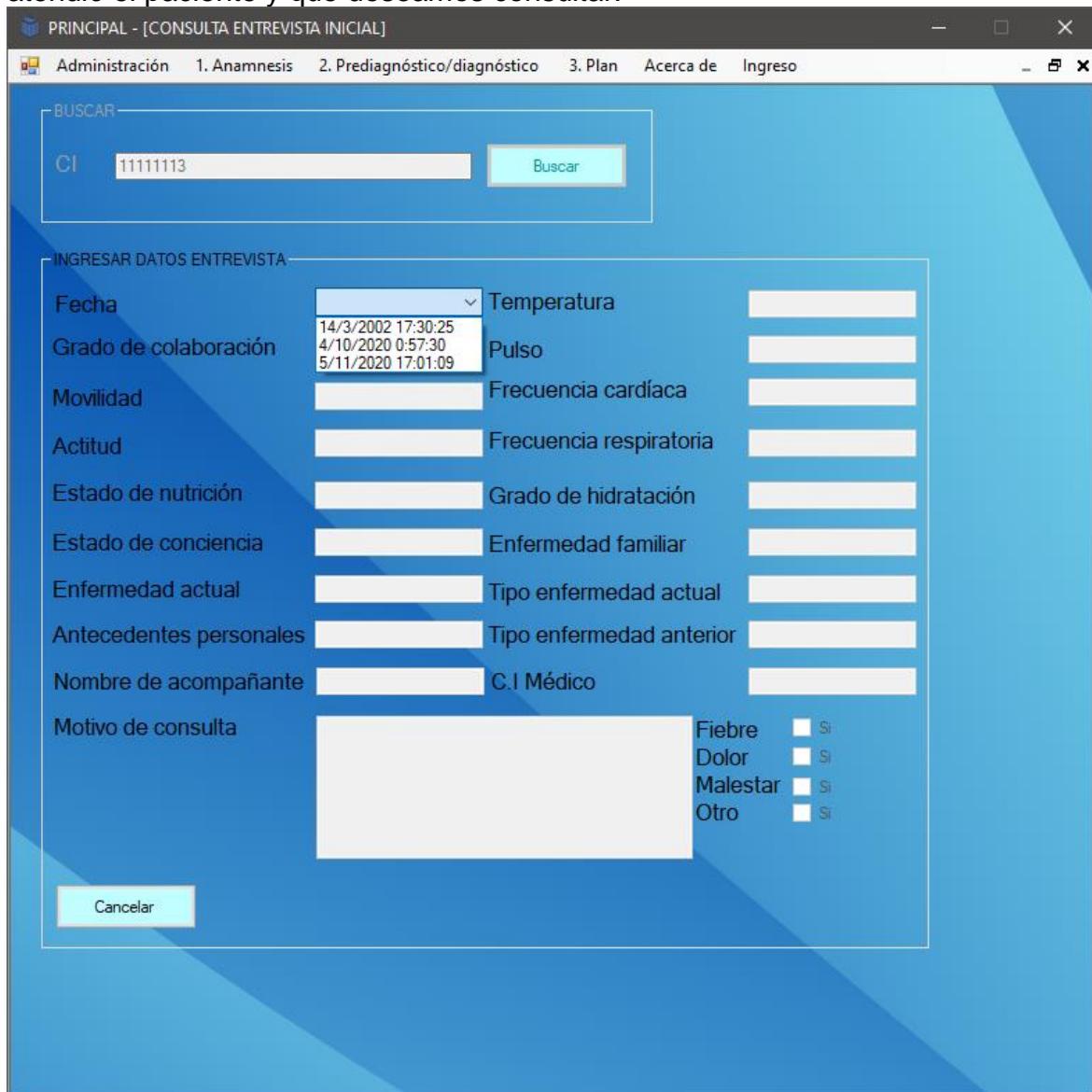


-Si el paciente no existe o no se le ha realizado una entrevista
En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente o el paciente con dicha cédula de identidad no se ha realizado entrevistas en la institución médica.



-Si el paciente existe y tiene entrevistas registradas

Ya ingresada la cédula, se habilitará una agrupación de datos pertenecientes a una entrevista en donde tendremos que seleccionar la fecha en la que se atendió el paciente y que deseamos consultar.



The screenshot shows a Windows application window titled "PRINCIPAL - [CONSULTA ENTREVISTA INICIAL]". The menu bar includes "Administración", "1. Anamnesis", "2. Prediagnóstico/diagnóstico", "3. Plan", "Acerca de", and "Ingreso". The main area has a blue header "BUSCAR" and a search field with "CI 11111113" and a "Buscar" button. Below this is a section titled "INGRESAR DATOS ENTREVISTA" with a table of data fields:

Fecha	14/3/2002 17:30:25	Temperatura	
Grado de colaboración	4/10/2020 0:57:30 5/11/2020 17:01:09	Pulso	
Movilidad		Frecuencia cardíaca	
Actitud		Frecuencia respiratoria	
Estado de nutrición		Grado de hidratación	
Estado de conciencia		Enfermedad familiar	
Enfermedad actual		Tipo enfermedad actual	
Antecedentes personales		Tipo enfermedad anterior	
Nombre de acompañante		C.I Médico	
Motivo de consulta		Fiebre	<input type="checkbox"/> Sí
		Dolor	<input type="checkbox"/> Sí
		Malestar	<input type="checkbox"/> Sí
		Otro	<input type="checkbox"/> Sí

At the bottom left is a "Cancelar" button.

Una vez seleccionado la fecha en la que se atendió el paciente, el sistema cargará en pantalla los datos pertenecientes a la entrevista con la fecha seleccionada previamente.

PRINCIPAL - [CONSULTA ENTREVISTA INICIAL]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR

CI 11111113 Buscar

INGRESAR DATOS ENTREVISTA

Fecha	5/11/2020 17:01:09	Temperatura	365
Grado de colaboración	Colabora	Pulso	Normal
Movilidad	Total	Frecuencia cardíaca	Normal
Actitud	Normal	Frecuencia respiratoria	Apta
Estado de nutrición	Comun	Grado de hidratación	Normal
Estado de conciencia	Conciente	Enfermedad familiar	Diabetes
Enfermedad actual		Tipo enfermedad actual	
Antecedentes personales	Diabetes	Tipo enfermedad anterior	Mellitus tipo uno
Nombre de acompañante	Agustín Iriart	C.I Médico	11111110
Motivo de consulta	Malestar estomacal	Fiebre	<input type="checkbox"/> Sí
		Dolor	<input type="checkbox"/> Sí
		Malestar	<input checked="" type="checkbox"/> Sí
		Otro	<input type="checkbox"/> Sí

C Cancelar

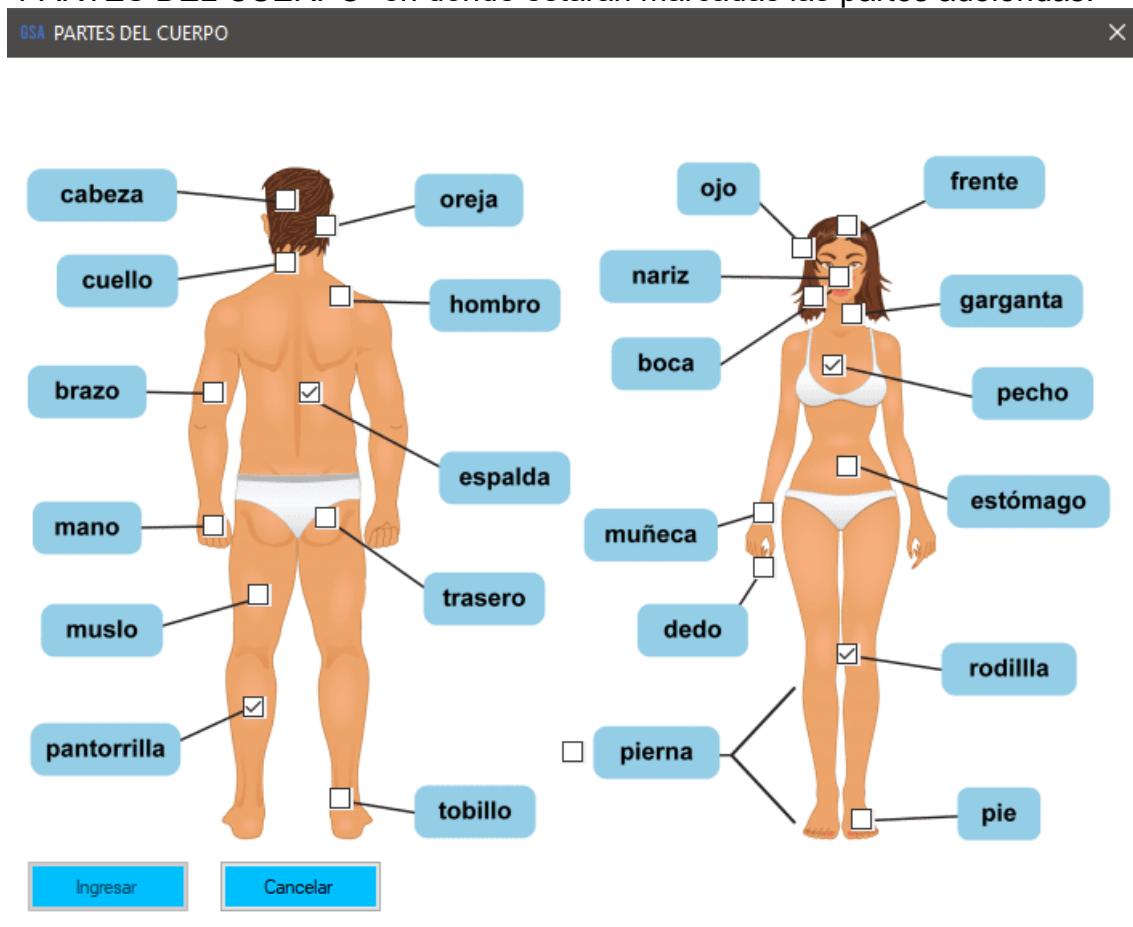
-Si el paciente registra que tuvo dolor en la entrevista

Si el sistema carga los datos de la entrevista de un paciente y este tuvo el síntoma de dolor en la misma, se habilitará la opción para consultar las partes del cuerpo en las que presentó dolor.

Fiebre	<input type="checkbox"/> Sí
Dolor	<input checked="" type="checkbox"/> Sí
Malestar	<input type="checkbox"/> Sí
Otro	<input type="checkbox"/> Sí

Consultar dolor en las partes del cuerpo

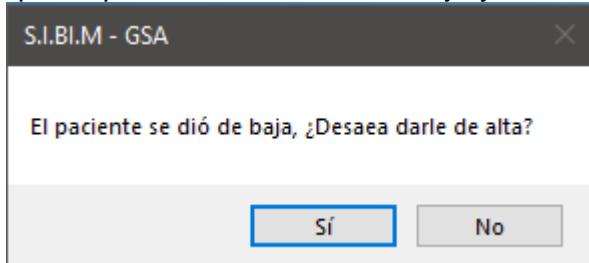
Una vez pulsado el botón para consultar las partes del cuerpo en las que el paciente posee dolor, el sistema desplegará una ventana emergente titulada "PARTES DEL CUERPO" en donde estarán marcadas las partes adoloridas.





-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

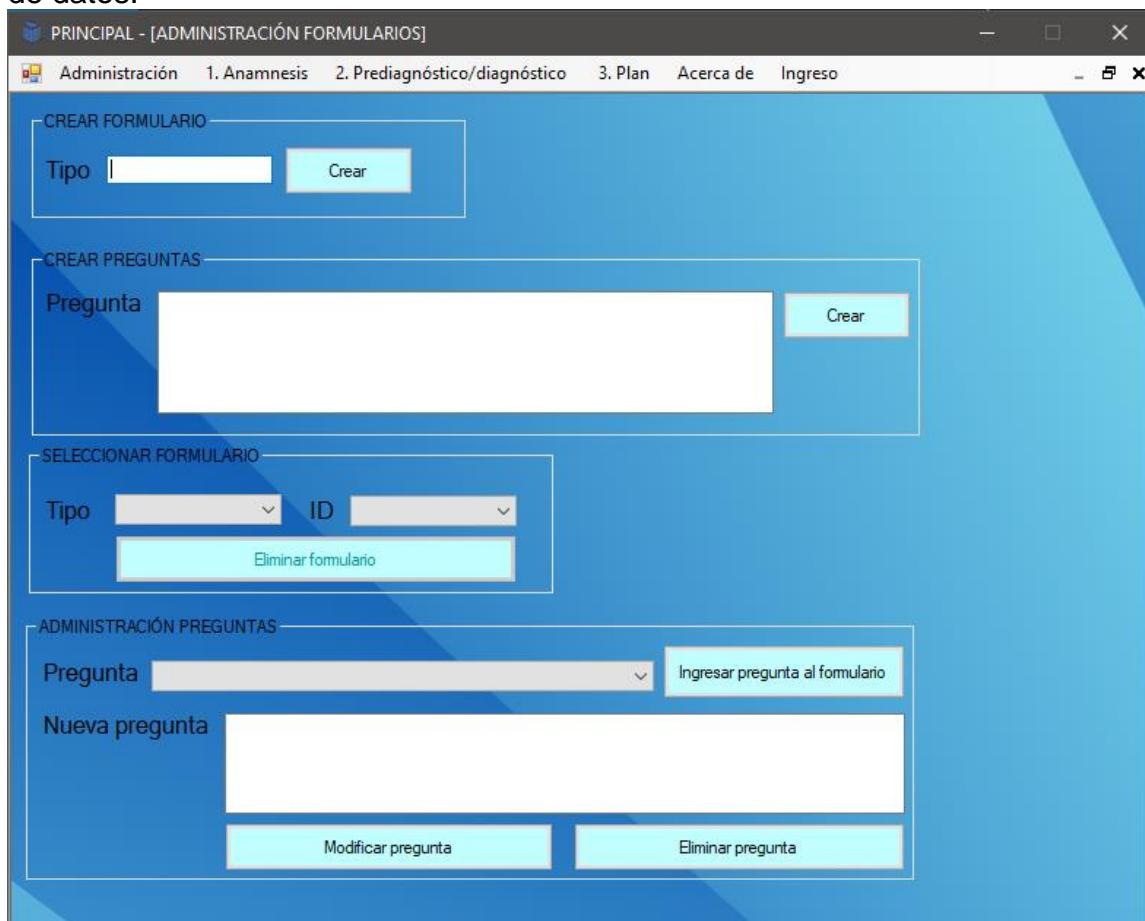


Administración - Formularios

Para administrar formularios deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Formularios”.



Una vez dentro de “Formulario” el sistema habilitará el ingreso y modificación de datos.





-Crear formulario

Para crear un formulario será necesario ingresar en el cuadro de texto que se ubica a la derecha de la etiqueta con texto “Tipo”, el tipo que corresponde al formulario que se desea crear.

The screenshot shows a software interface titled "PRINCIPAL - [ADMINISTRACIÓN FORMULARIOS]". In the top menu bar, there are several options: "Administración", "1. Anamnesis", "2. Prediagnóstico/diagnóstico", "3. Plan", "Acerca de", and "Ingreso". Below the menu, a sub-section titled "CREAR FORMULARIO" is displayed. It contains a label "Tipo" followed by a text input field containing the value "Dolor". To the right of the input field is a blue "Crear" button. The background of the main window is light blue.

-Crear preguntas

Para crear una pregunta será necesario ingresar en el cuadro de texto que se ubica a la derecha de la etiqueta con texto “Pregunta”, la pregunta que se desea crear.

The screenshot shows a software interface titled "CREAR PREGUNTAS". On the left side, there is a label "Pregunta" next to a large, empty text input field. On the right side, there is a blue "Crear" button. The background is light blue.

-Eliminar formulario

Para eliminar un formulario será necesario previamente seleccionar el tipo de formulario y luego el número identificador (ID) perteneciente al formulario que se desea eliminar.

The screenshot shows a software interface titled "SELECCIONAR FORMULARIO". It features two dropdown menus: one for "Tipo" set to "Dolor" and another for "ID" set to "2". Below these dropdowns is a blue "Eliminar formulario" button.

Una vez seleccionados los datos solicitados anteriormente, deberá de pulsar el botón de “Eliminar formulario”, si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que el formulario se eliminó con éxito, de lo contrario el sistema notificará que hubo errores al eliminar el formulario.



-Ingresar preguntas al formulario

Para ingresar preguntas al formulario será necesario previamente seleccionar el tipo de formulario y luego el número identificador (ID) perteneciente al formulario al cual se le quiere agregar las preguntas.

SELECCIONAR FORMULARIO

Tipo Dolor ID 1

Eliminar formulario

Una vez seleccionados los datos solicitados anteriormente, deberá seleccionar la pregunta en la lista desplegable que se desea agregar.

ADMINISTRACIÓN PREGUNTAS

Pregunta Dolor ¿A?

Dolor ¿B?

Dolor ¿C?

Malestar ¿D?

Malestar ¿F?

Fiebre ¿G?

Fiebre ¿H?

Fiebre ¿I?

Fiebre ¿J?

Nueva pregunta

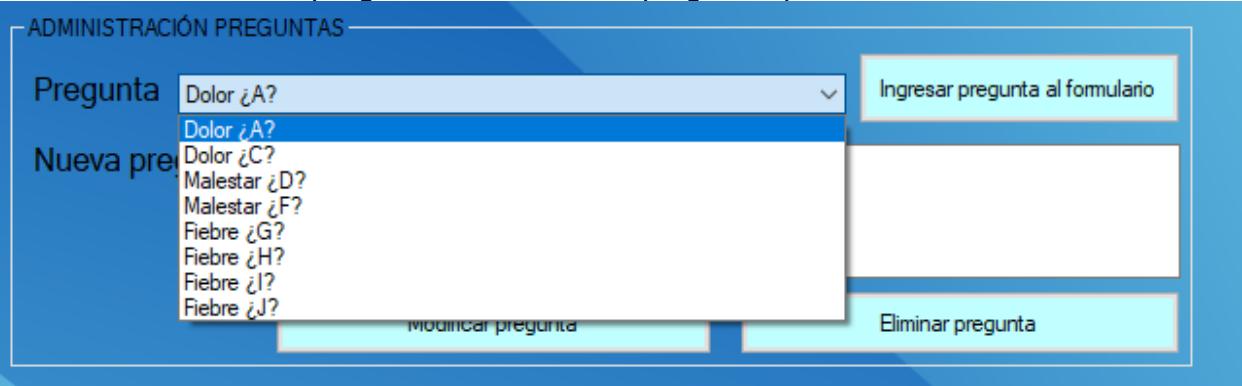
Ingresar pregunta al formulario

Eliminar pregunta

Una vez seleccionada la pregunta, deberá de pulsar el botón “Ingresar pregunta al formulario”, si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó la pregunta al formulario con éxito, de lo contrario el sistema notificará que hubo errores al ingresar la pregunta al formulario.

-Modificar pregunta

Deberá seleccionar la pregunta en la lista desplegable que se desea eliminar.



ADMINISTRACIÓN PREGUNTAS

Pregunta Dolor ¿A?

Dolor ¿A?
Dolor ¿C?
Malestar ¿D?
Malestar ¿F?
Fiebre ¿G?
Fiebre ¿H?
Fiebre ¿I?
Fiebre ¿J?

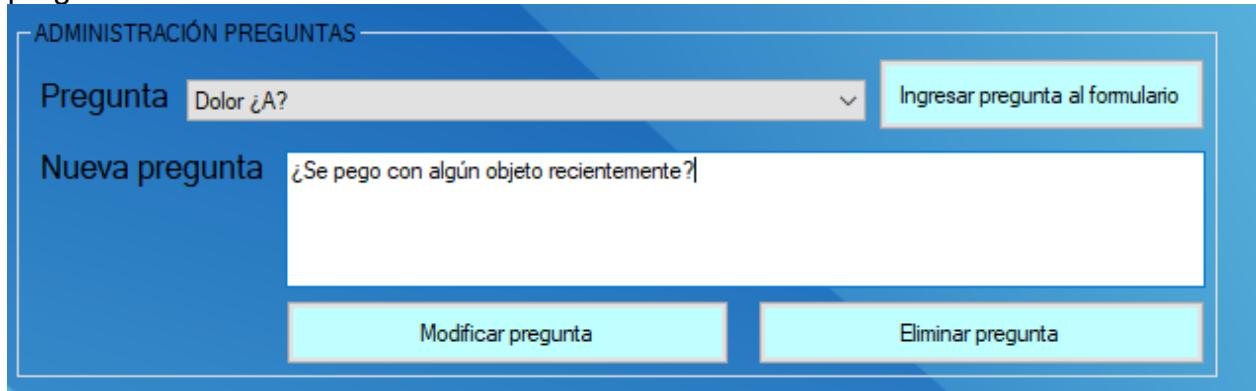
Nueva pregunta

Ingresar pregunta al formulario

Modificar pregunta

Eliminar pregunta

Una vez seleccionada la pregunta, tendrá que pulsar el botón “Eliminar pregunta”.



ADMINISTRACIÓN PREGUNTAS

Pregunta Dolor ¿A?

Nueva pregunta ¿Se pego con algún objeto recientemente?

Ingresar pregunta al formulario

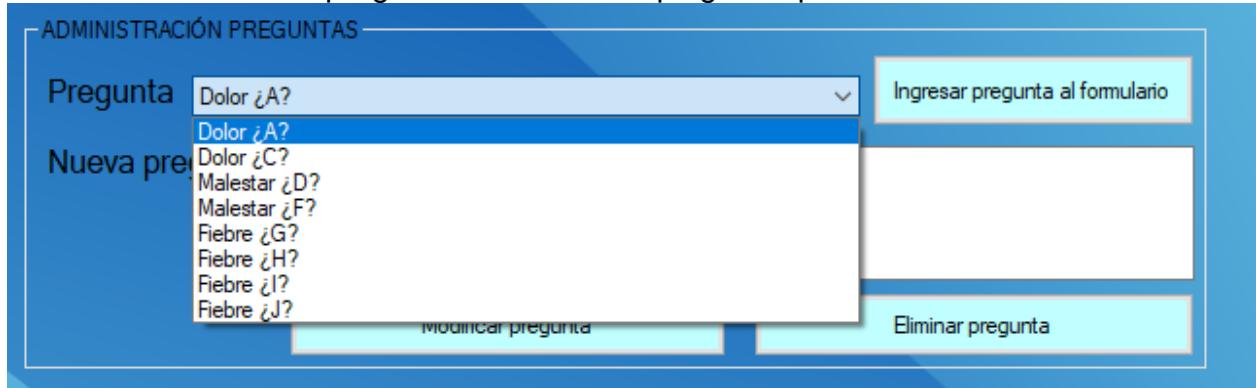
Modificar pregunta

Eliminar pregunta

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se eliminó la pregunta con éxito, de lo contrario el sistema notificará que hubo errores al eliminar la pregunta.

-Eliminar pregunta

Deberá seleccionar la pregunta en la lista desplegable que se desea modificar.

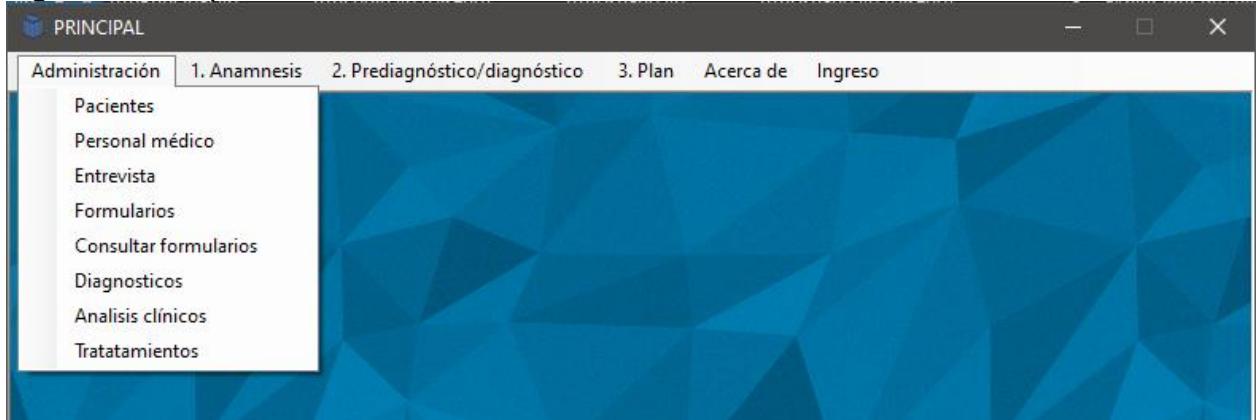


Una vez seleccionada la pregunta, tendrá que ingresar la nueva pregunta que deseé en el cuadro de texto ubicado a la derecha de la etiqueta de texto “Nueva pregunta”.

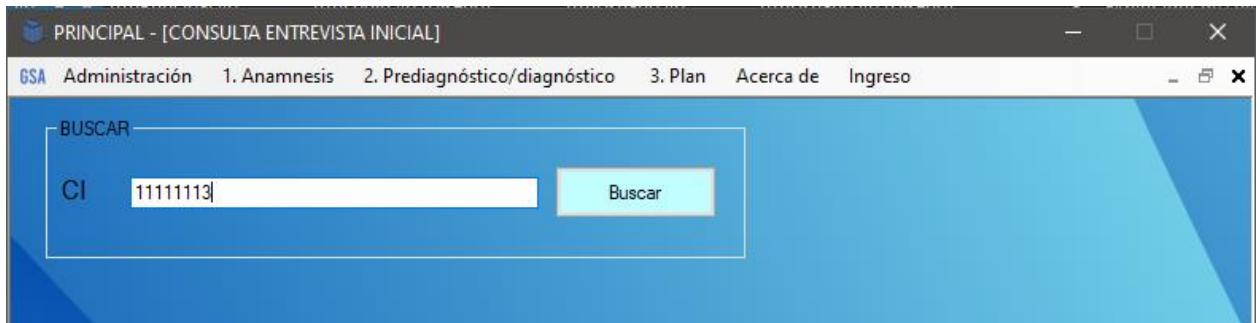


Administración - Consultar formularios

Para consultar los formularios deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Consultar formularios”.

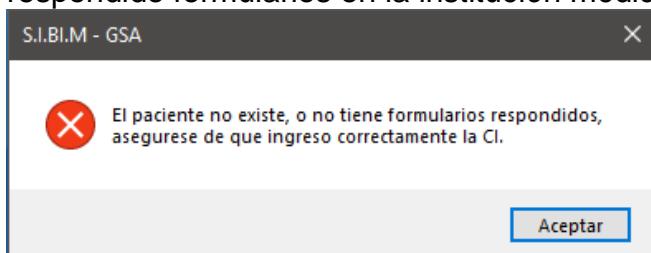


Una vez dentro de “Consultar formularios” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



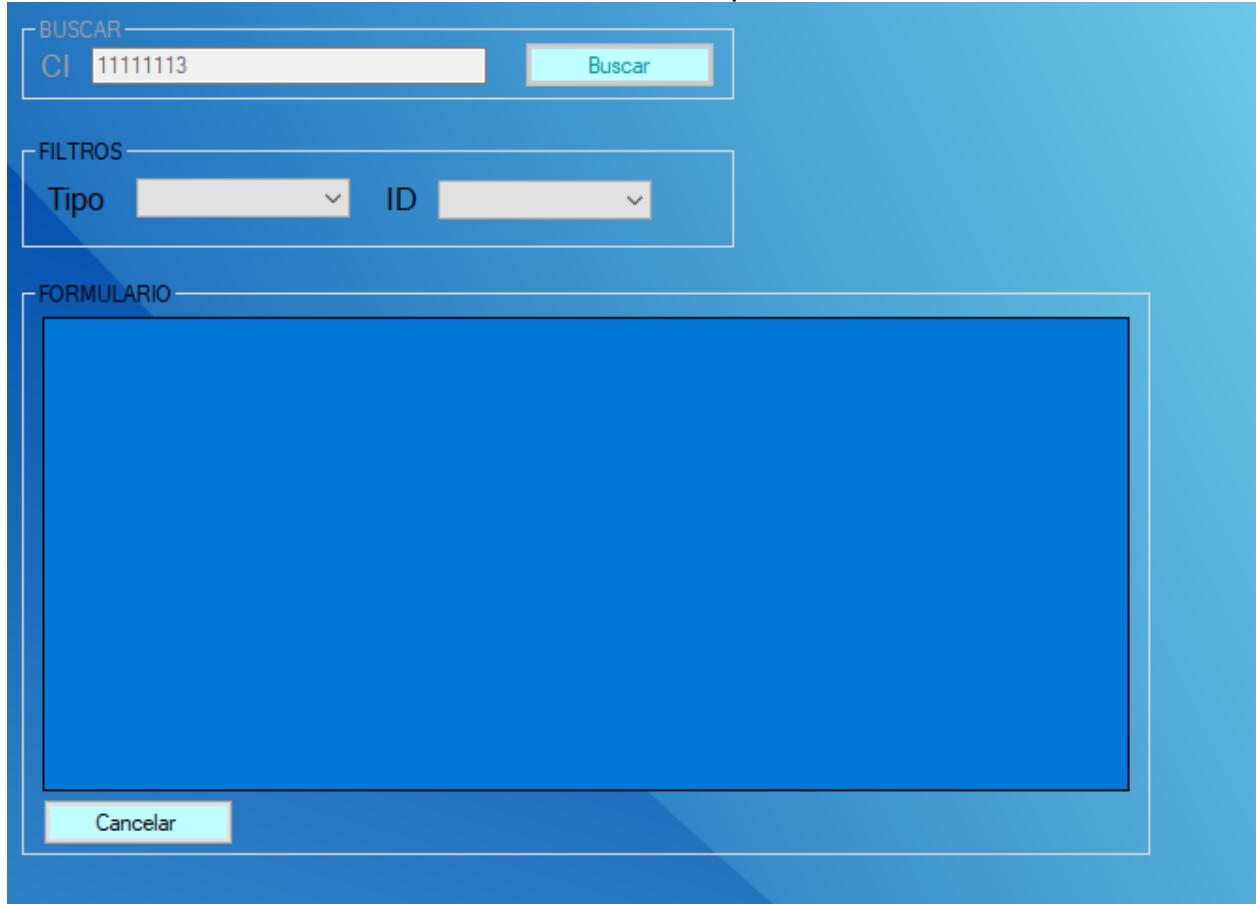
-Si el paciente no existe o no respondió ningún formulario

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente o el paciente con dicha cédula de identidad no ha respondido formularios en la institución médica.



-Si el paciente existe y respondió formularios

En este caso el sistema procederá a habilitar una agrupación de datos, donde deberemos de seleccionar los datos solicitados en pantalla.



The screenshot shows a user interface for searching patient forms. At the top left is a "BUSCAR" button with a dropdown menu containing the text "CI". Next to it is a text input field containing the number "11111113", followed by a "Buscar" button. Below this is a "FILTROS" section with two dropdown menus: "Tipo" and "ID". A large rectangular area labeled "FORMULARIO" occupies the center, which is currently empty. At the bottom left of this area is a "Cancelar" button.

Deberá de seleccionar el tipo de formulario y luego procederá a seleccionar el número identificador (ID) del formulario.



The screenshot shows a "FILTROS" section with two dropdown menus. The first dropdown is set to "Tipo" and has "Dolor" selected. The second dropdown is set to "ID" and has "5" selected.



Una vez seleccionados los datos mencionados anteriormente, el sistema cargará en pantalla las preguntas en conjunto con las respuestas a cada una de ellas.

FORMULARIO

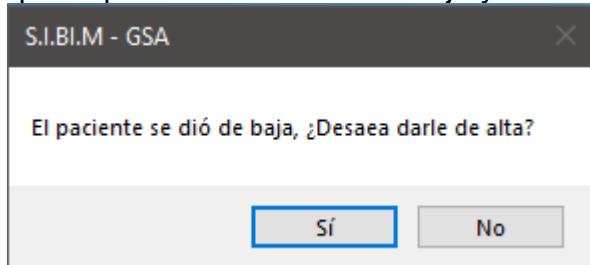
Pregunta	Respuesta
¿Se golpeó en esa zona recientemente?	No tuvo un golpe en esa zona recientemente, la causa es...
¿Hace cuanto siente el dolor?	Aproximadamente 4 días.
Del nivel 1 a 10, ¿Qué nivel de dolor siente?	Nivel 4
¿El dolor es permanente o solamente al realizar cierta actividad?	De momento es permanente.

< >

Cancelar

-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.

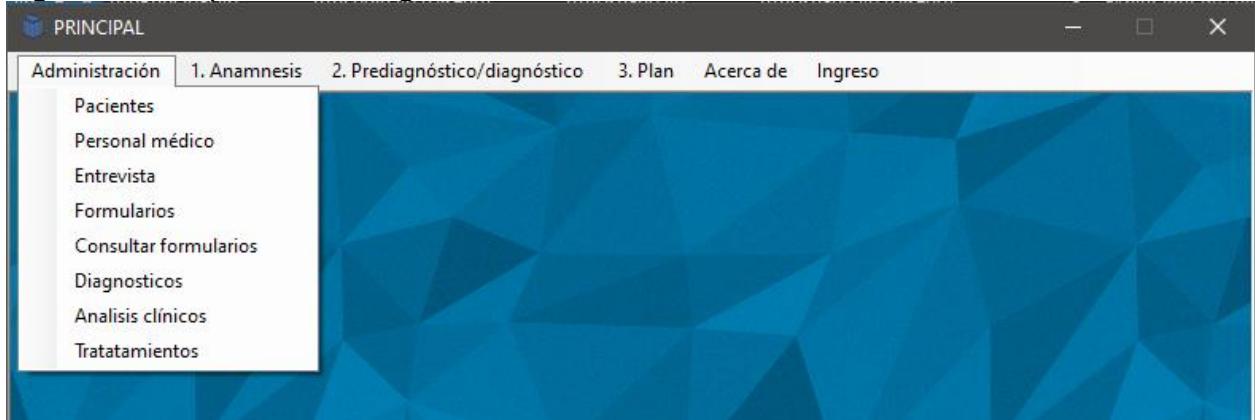


Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

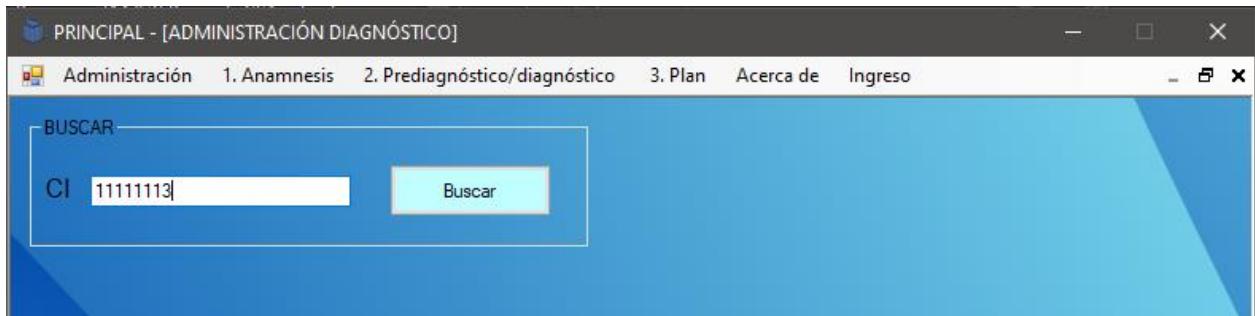


Administración - Diagnósticos

Para administrar los diagnósticos deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Diagnosticos”.

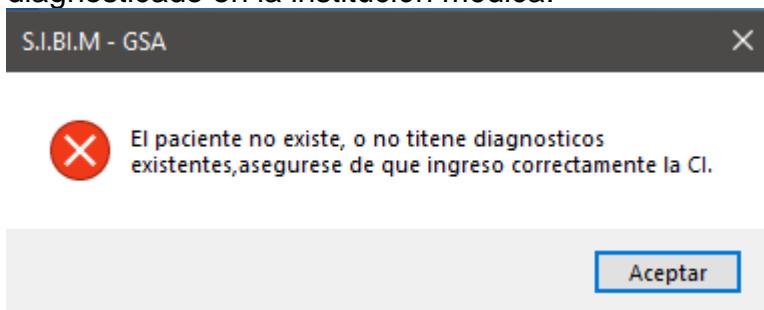


Una vez dentro de “Diagnosticos” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



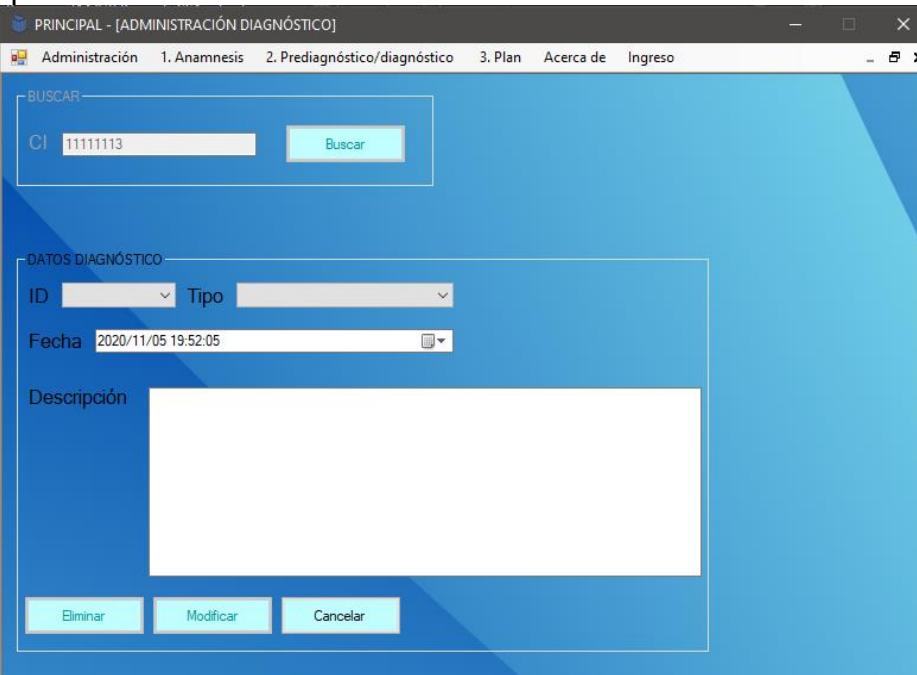
-Si el paciente no existe o no ha sido diagnosticado

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente o el paciente con dicha cédula de identidad no ha sido diagnosticado en la institución médica.



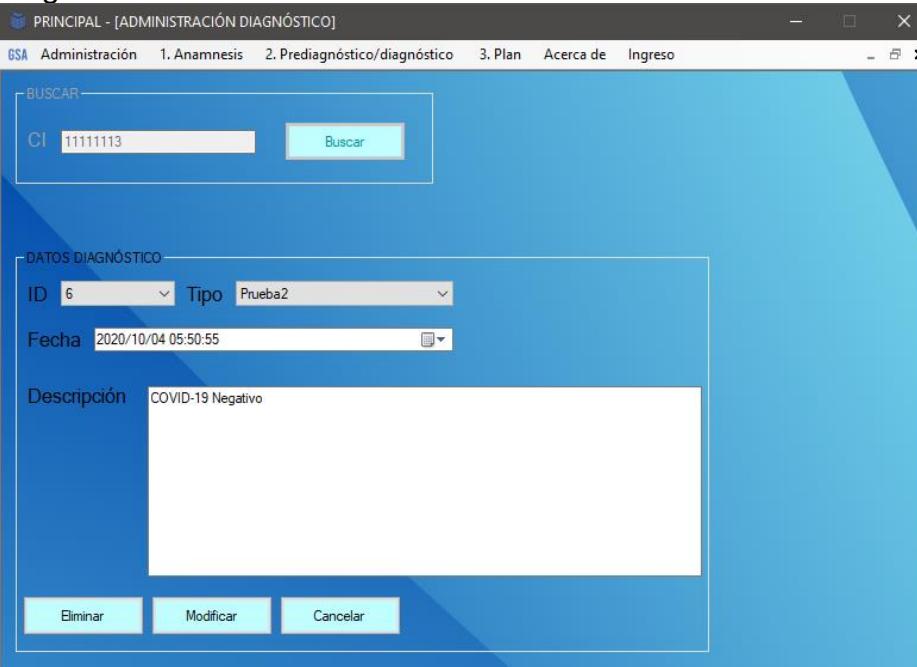
-Si el paciente existe y ha sido diagnosticado

En este caso el sistema procederá a habilitar una agrupación de datos, donde deberemos de seleccionar el número identificador (ID) del diagnóstico que queremos administrar.



The screenshot shows the 'PRINCIPAL - [ADMINISTRACIÓN DIAGNÓSTICO]' window. At the top, there is a navigation bar with links: Administración, 1. Anamnesis, 2. Prediagnóstico/diagnóstico, 3. Plan, Acerca de, and Ingreso. Below the navigation bar is a search section labeled 'BUSCAR' with a text input field containing 'CI 11111113' and a 'Buscar' button. The main area is titled 'DATOS DIAGNÓSTICO' and contains fields for 'ID' (dropdown menu), 'Tipo' (dropdown menu), 'Fecha' (date picker showing '2020/11/05 19:52:05'), and a large 'Descripción' text area. At the bottom of this section are three buttons: 'Eliminar', 'Modificar', and 'Cancelar'.

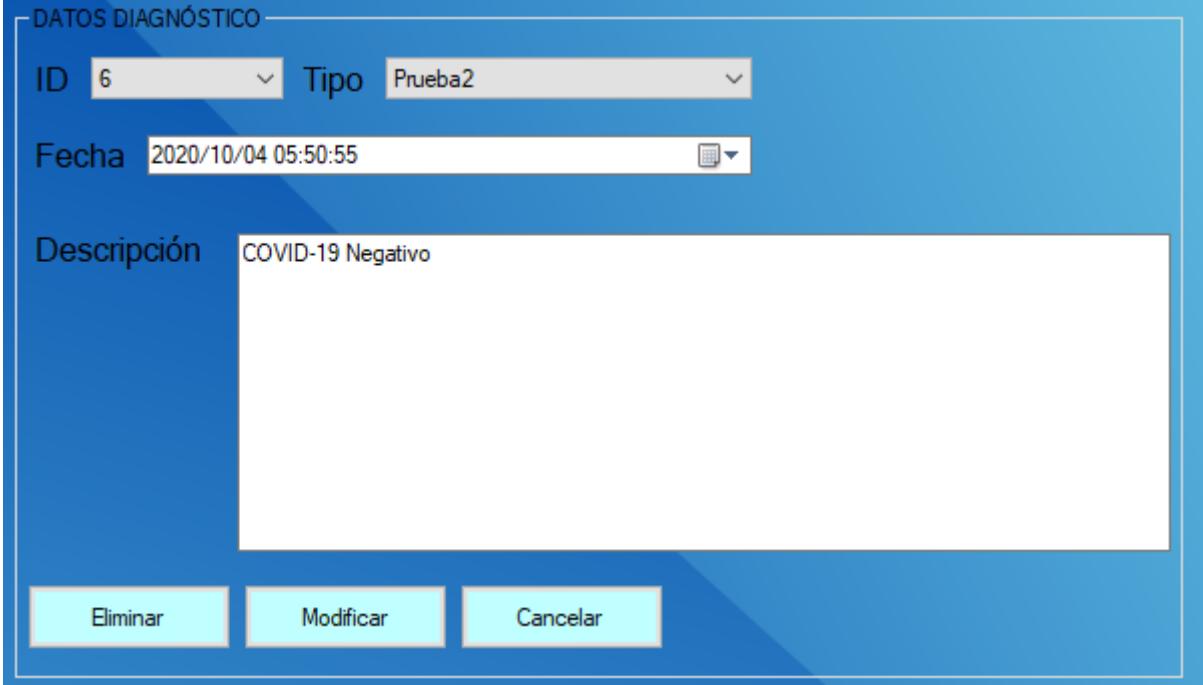
Una vez seleccionado el número identificador (ID) del diagnóstico que se desea administrar el sistema cargará en pantalla los datos pertenecientes al diagnóstico seleccionado.



The screenshot shows the same 'PRINCIPAL - [ADMINISTRACIÓN DIAGNÓSTICO]' window after a selection. The 'ID' field now displays '6' and the 'Tipo' field displays 'Prueba2'. The 'Fecha' field still shows '2020/10/04 05:50:55'. The 'Descripción' text area now contains the text 'COVID-19 Negativo'. The bottom buttons remain the same: 'Eliminar', 'Modificar', and 'Cancelar'.

-Eliminar diagnóstico

Para eliminar un diagnóstico se deberá de seleccionar el número identificador (ID) del diagnóstico que queremos eliminar y posteriormente pulsar el botón "Eliminar".



The screenshot shows a software interface for managing diagnostic data. At the top, there's a title bar labeled "DATOS DIAGNÓSTICO". Below it, there are three input fields: "ID" containing the value "6", "Tipo" set to "Prueba2", and "Fecha" showing the timestamp "2020/10/04 05:50:55". A text area labeled "Descripción" contains the text "COVID-19 Negativo". At the bottom of the dialog are three buttons: "Eliminar" (highlighted in light blue), "Modificar", and "Cancelar".

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se eliminó el diagnóstico con éxito, de lo contrario el sistema notificará que hubo errores al eliminar el diagnóstico.

-Modificar diagnóstico

Para modificar un diagnóstico se deberá de seleccionar el número identificador (ID) del diagnóstico que queremos modificar.

DATOS DIAGNÓSTICO

ID	6	Tipo	Prueba2
Fecha	2020/10/04 05:50:55		
Descripción	COVID-19 Negativo		

Eliminar **Modificar** **Cancelar**

Una vez seleccionado el número identificador (ID) del diagnóstico que se desea modificar, debe proceder a cambiar la información en los cuadros de textos y listas desplegables que deseé modificar, ya reemplazada la información, deberá de pulsar el botón “Modificar” para aplicar los cambios.

DATOS DIAGNÓSTICO

ID	6	Tipo	Heroico
Fecha	2019/06/06 05:50:55		
Descripción	No tiene ninguna enfermedad.		

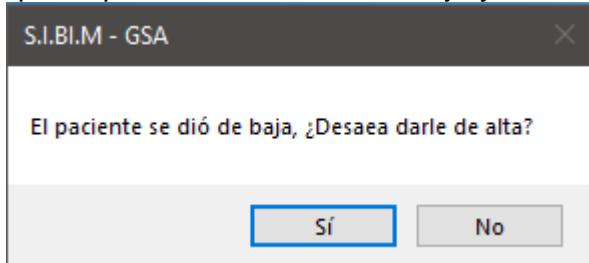
Eliminar **Modificar** **Cancelar**

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se modificó el diagnóstico con éxito, de lo contrario el sistema notificará que hubo errores al modificar el diagnóstico.



-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.



Administración - Análisis clínicos

Para administrar los análisis clínicos deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Análisis clínicos”.

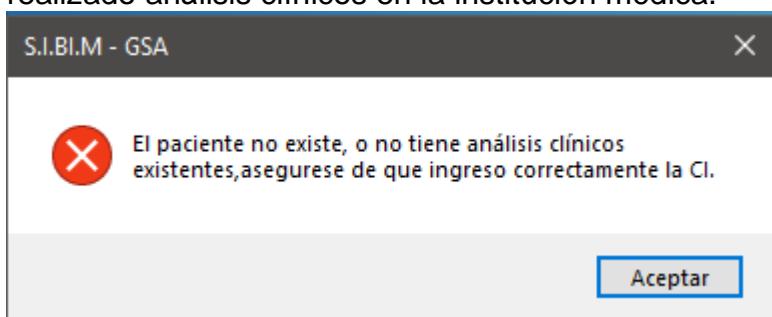


Una vez dentro de “Análisis clínicos” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



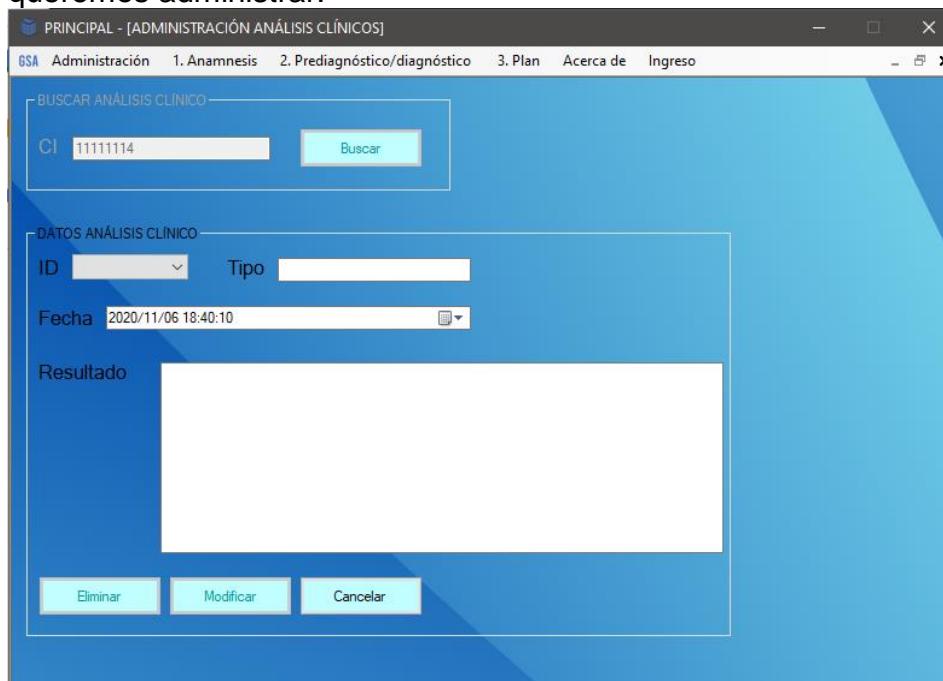
-Si el paciente no existe o no ha tenido análisis clínicos

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente o el paciente con dicha cédula de identidad no se ha realizado análisis clínicos en la institución médica.



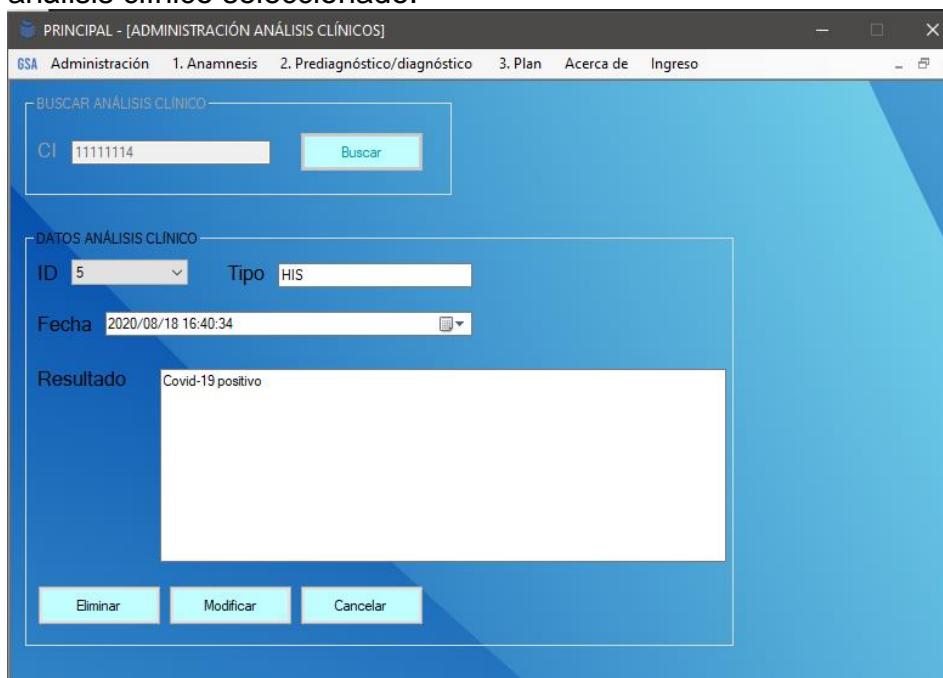
-Si el paciente existe y se ha realizado análisis clínicos

En este caso el sistema procederá a habilitar una agrupación de datos, donde deberemos de seleccionar el número identificador (ID) del análisis que queremos administrar.



The screenshot shows the 'PRINCIPAL - [ADMINISTRACIÓN ANÁLISIS CLÍNICOS]' window. In the 'BUSCAR ANÁLISIS CLÍNICO' section, the 'CI' field contains '11111114' and the 'Buscar' button is visible. In the 'DATOS ANÁLISIS CLÍNICO' section, the 'ID' dropdown is set to '5' and the 'Tipo' dropdown is set to 'HIS'. The 'Fecha' field shows '2020/11/06 18:40:10'. The 'Resultado' area is empty. At the bottom are 'Eliminar', 'Modificar', and 'Cancelar' buttons.

Una vez seleccionado el número identificador (ID) del análisis clínico que se desea administrar el sistema cargará en pantalla los datos pertenecientes al análisis clínico seleccionado.



The screenshot shows the same window after selecting ID 5. The 'DATOS ANÁLISIS CLÍNICO' section now displays 'ID 5' and 'Tipo HIS'. The 'Fecha' field shows '2020/08/18 16:40:34'. The 'Resultado' area contains the text 'Covid-19 positivo'. The bottom buttons remain 'Eliminar', 'Modificar', and 'Cancelar'.



-Eliminar análisis clínico

Para eliminar un análisis clínico se deberá de seleccionar el número identificador (ID) del análisis clínico que queremos eliminar y posteriormente pulsar el botón “Eliminar”.

DATOS ANÁLISIS CLÍNICO

ID	5	Tipo	HIS
Fecha	2020/08/18 16:40:34		
Resultado	Covid-19 positivo		
Eliminar Modificar Cancelar			

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se eliminó el análisis clínico con éxito, de lo contrario el sistema notificará que hubo errores al eliminar el análisis clínico.

-Modificar análisis clínico

Para modificar un análisis clínico se deberá de seleccionar el número identificador (ID) del análisis clínico que queremos modificar.

DATOS ANÁLISIS CLÍNICO

ID	5	Tipo	HIS
Fecha	2020/08/18 16:40:34	<input type="button" value=""/>	
Resultado	Covid-19 positivo		
<input type="button" value="Eliminar"/> <input type="button" value="Modificar"/> <input type="button" value="Cancelar"/>			

Una vez seleccionado el número identificador (ID) del análisis clínico que se desea modificar, debe proceder a cambiar la información en los cuadros de textos que deseé modificar, ya reemplazada la información, deberá de pulsar el botón “Modificar” para aplicar los cambios.

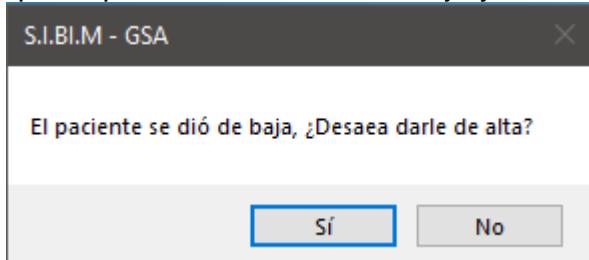
DATOS ANÁLISIS CLÍNICO

ID	5	Tipo	HIS
Fecha	2020/08/18 16:40:34	<input type="button" value=""/>	
Resultado	Covid-19 NEGATIVO		
<input type="button" value="Eliminar"/> <input type="button" value="Modificar"/> <input type="button" value="Cancelar"/>			

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se modificó el análisis clínico con éxito, de lo contrario el sistema notificará que hubo errores al modificar el análisis clínico.

-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.

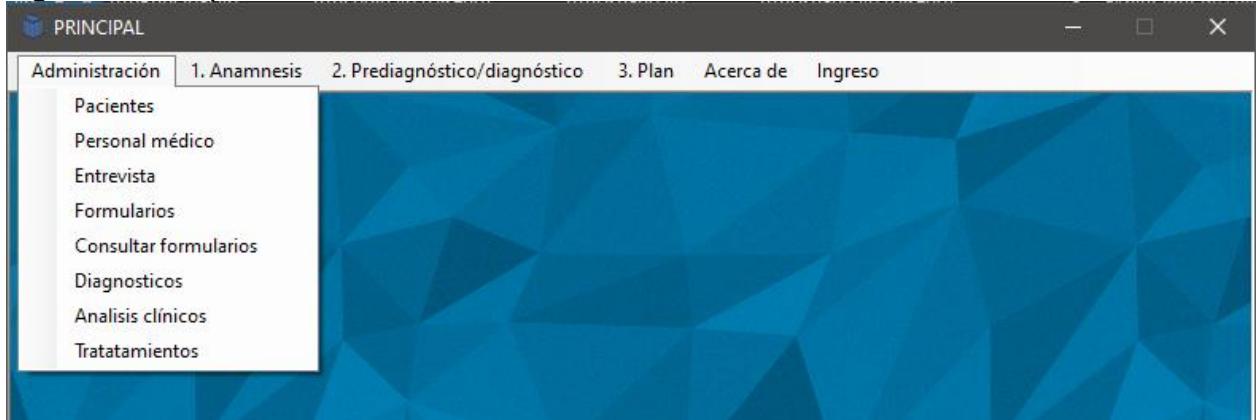


Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

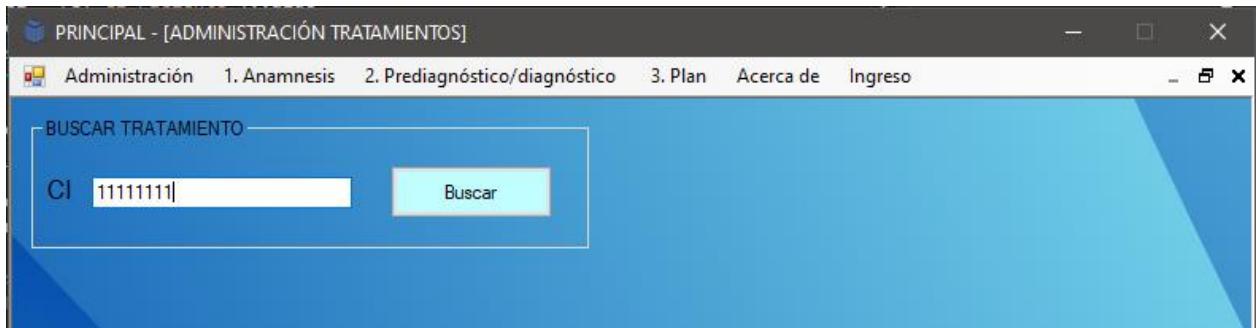


Administración - Tratamientos

Para administrar los tratamientos deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Tratamientos”.

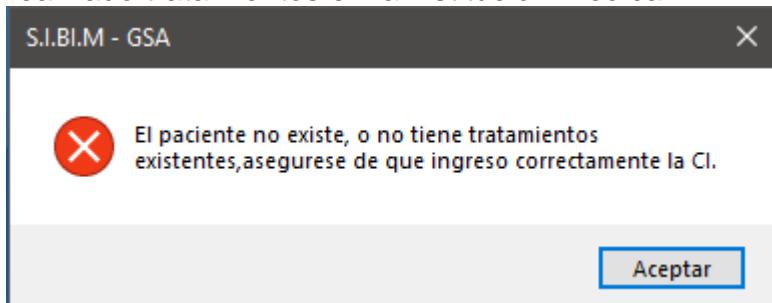


Una vez dentro de “Tratamientos” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



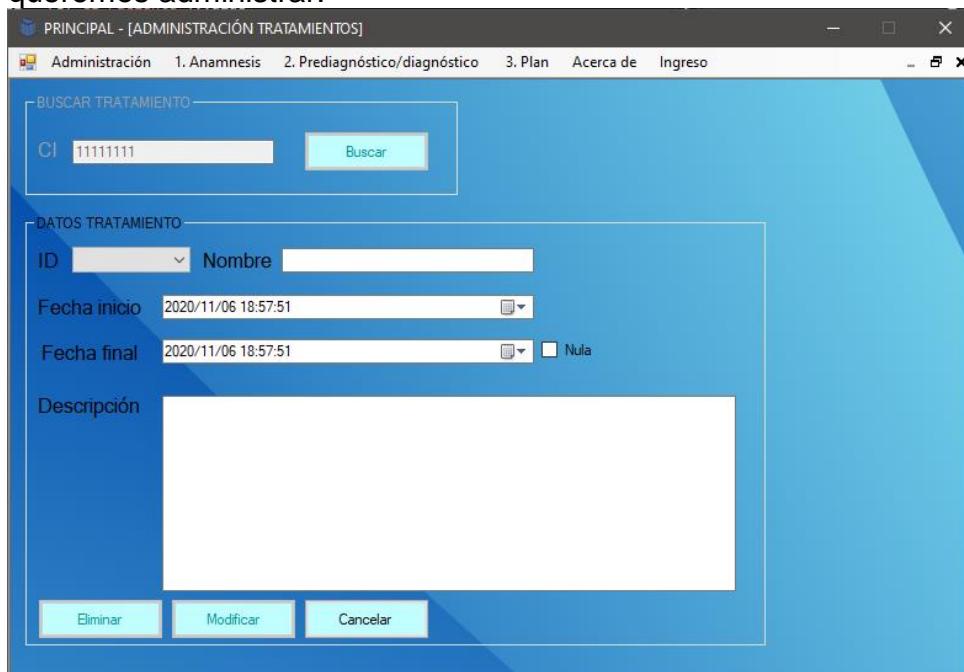
-Si el paciente no existe o no ha tenido tratamientos

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente o el paciente con dicha cédula de identidad no se ha realizado tratamientos en la institución médica.



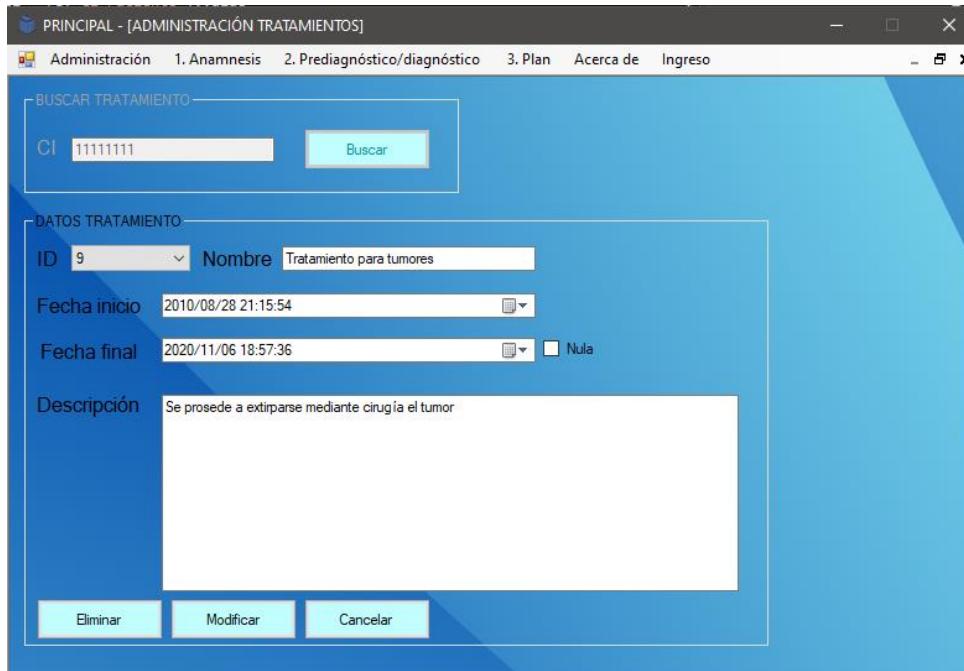
-Si el paciente existe y se ha realizado tratamientos

En este caso el sistema procederá a habilitar una agrupación de datos, donde deberemos de seleccionar el número identificador (ID) del tratamiento que queremos administrar.



The screenshot shows a software window titled 'PRINCIPAL - [ADMINISTRACIÓN TRATAMIENTOS]'. At the top, there is a menu bar with options: Administración, 1. Anamnesis, 2. Prediagnóstico/diagnóstico, 3. Plan, Acerca de, and Ingreso. Below the menu, there is a search section labeled 'BUSCAR TRATAMIENTO' with a text input field containing '11111111' and a 'Buscar' button. The main area is titled 'DATOS TRATAMIENTO' and contains fields for 'ID' (dropdown menu), 'Nombre' (text input), 'Fecha inicio' (date/time picker set to '2020/11/06 18:57:51'), 'Fecha final' (date/time picker set to '2020/11/06 18:57:51' with a checked 'Nula' checkbox), and a large 'Descripción' text area. At the bottom of this section are three buttons: 'Eliminar', 'Modificar', and 'Cancelar'.

Una vez seleccionado el número identificador (ID) del tratamiento que se desea administrar el sistema cargará en pantalla los datos pertenecientes al tratamiento seleccionado.



The screenshot shows the same software window as before, but now it displays specific treatment details. The 'Nombre' field is populated with 'Tratamiento para tumores', and the 'Descripción' field contains the text 'Se prosede a extirparse mediante cirugía el tumor'. The other fields remain the same as in the previous screenshot.



-Eliminar tratamiento

Para eliminar un tratamiento se deberá de seleccionar el número identificador (ID) del tratamiento que queremos eliminar y posteriormente pulsar el botón "Eliminar".

DATOS TRATAMIENTO

ID	9	Nombre	Tratamiento para tumores
Fecha inicio	2010/08/28 21:15:54 <input type="button" value="Calendar"/>		
Fecha final	2020/11/06 19:10:46 <input type="button" value="Calendar"/> <input type="checkbox"/> Nula		
Descripción	Se prosede a extirparse mediante cirugía el tumor		

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se eliminó el tratamiento con éxito, de lo contrario el sistema notificará que hubo errores al eliminar el tratamiento .

-Modificar tratamiento

Para modificar un tratamiento se deberá de seleccionar el número identificador (ID) del tratamiento que queremos modificar.

DATOS TRATAMIENTO

ID	9	Nombre	Tratamiento para tumores
Fecha inicio	2010/08/28 21:15:54	<input type="button" value=""/>	
Fecha final	2020/11/06 19:10:46	<input type="checkbox"/> Nula	
Descripción	Se prosede a extirparse mediante cirugía el tumor		
<input type="button" value="Eliminar"/> <input type="button" value="Modificar"/> <input type="button" value="Cancelar"/>			

Una vez seleccionado el número identificador (ID) del tratamiento que se desea modificar, debe proceder a cambiar la información en los cuadros de textos que deseé modificar, ya reemplazada la información, deberá de pulsar el botón “Modificar” para aplicar los cambios.

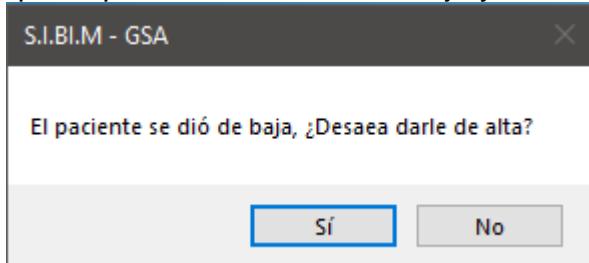
DATOS TRATAMIENTO

ID	9	Nombre	Tratamiento para tumores
Fecha inicio	2010/08/28 21:15:54	<input type="button" value=""/>	
Fecha final	NULA	<input checked="" type="checkbox"/> Nula	
Descripción	Quimioterapia		
<input type="button" value="Eliminar"/> <input type="button" value="Modificar"/> <input type="button" value="Cancelar"/>			

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se modificó el tratamiento con éxito, de lo contrario el sistema notificará que hubo errores al modificar el tratamiento.

-Si el paciente existe pero fue dado de baja

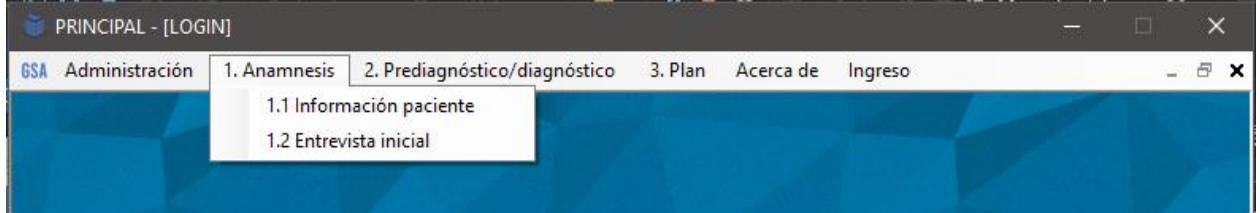
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



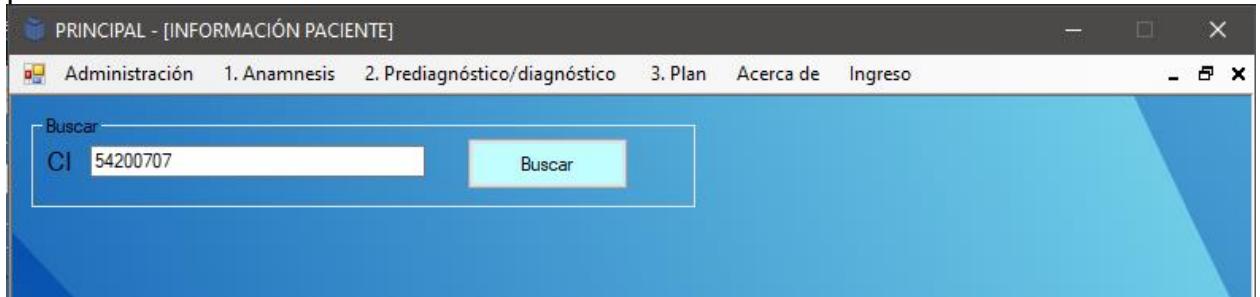
Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

1. Anamnesis - 1.1 Información paciente

Para Ingresar la entrevista inicial deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “1. Anámnasis” y luego ingresar a la opción de “1.2 Información paciente”.

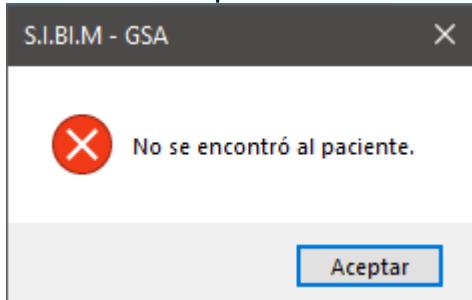


Una vez dentro de “1.1 Información paciente” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



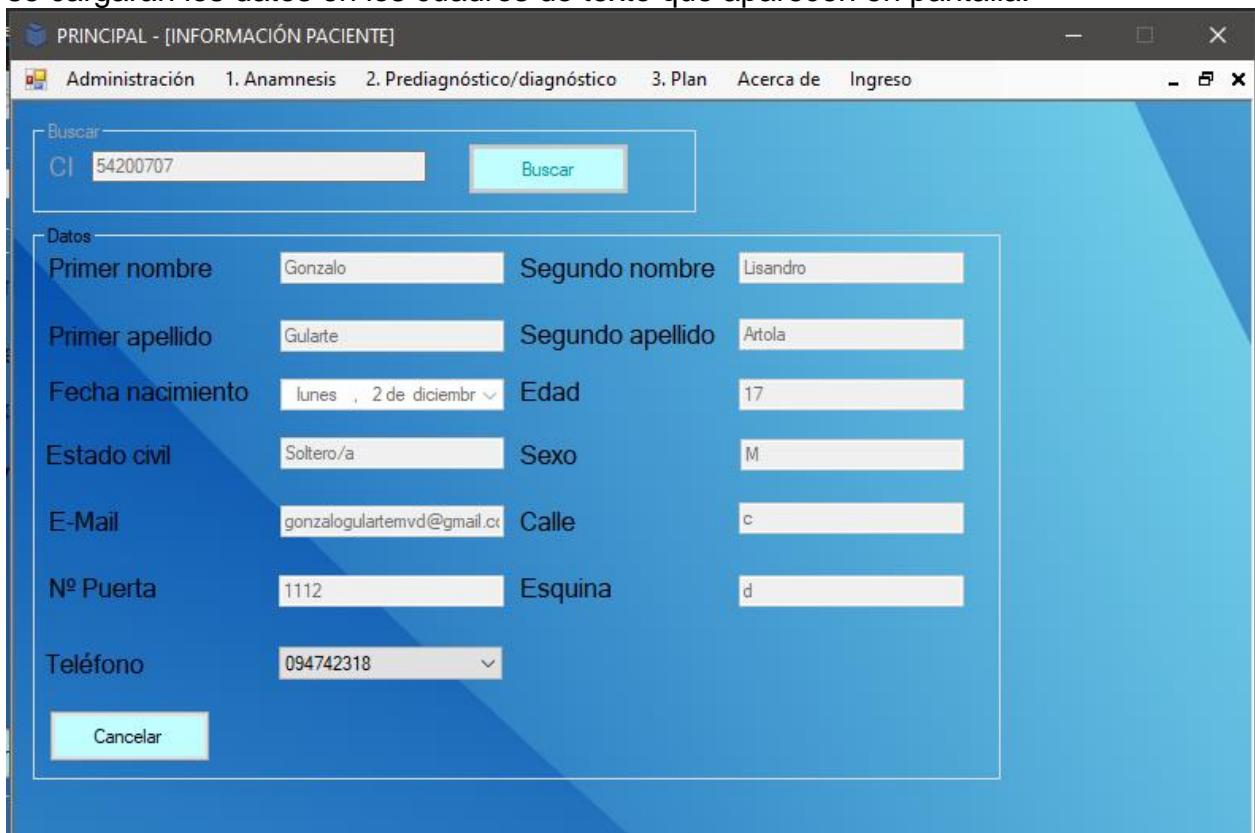
-Si el paciente no existe

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente.



-Si el paciente existe

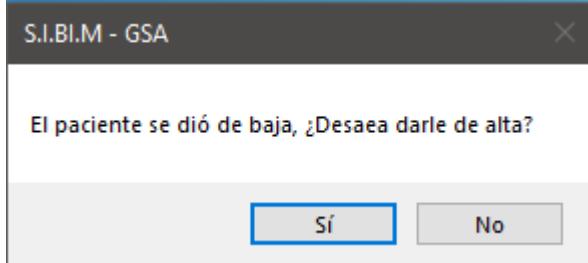
En este caso el sistema procederá a habilitar una agrupación de datos, donde se cargarán los datos en los cuadros de texto que aparecen en pantalla.



Datos	
Primer nombre	Gonzalo
Segundo nombre	Lisandro
Primer apellido	Gularde
Segundo apellido	Artola
Fecha nacimiento	lunes , 2 de diciembre
Edad	17
Estado civil	Soltero/a
Sexo	M
E-Mail	gonzalogularatemvd@gmail.com
Calle	c
Nº Puerta	1112
Esquina	d
Teléfono	094742318

-Si el paciente existe pero fue dado de baja

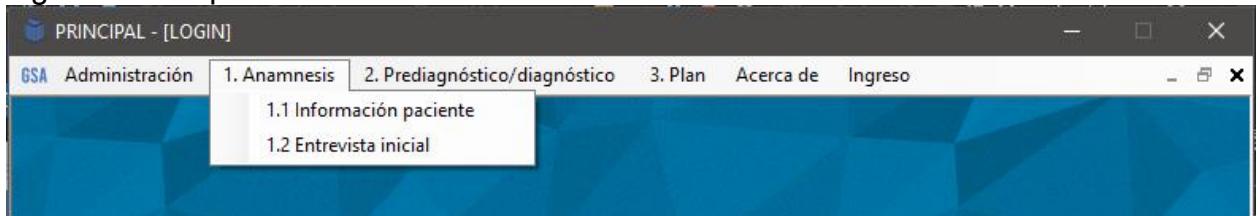
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



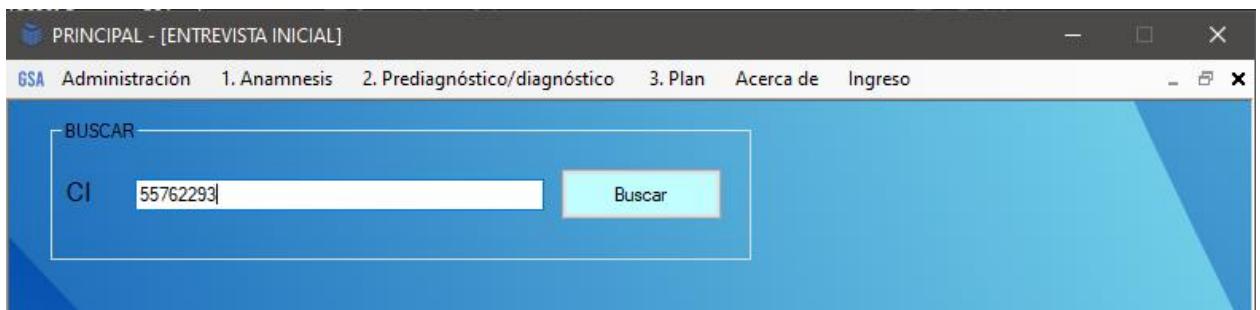
Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

1. Anamnesis - 1.2 Entrevista inicial

Para Ingresar la entrevista inicial deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “1. Anamnesis” y luego ingresar a la opción de “1.2 Entrevista Inicial”.

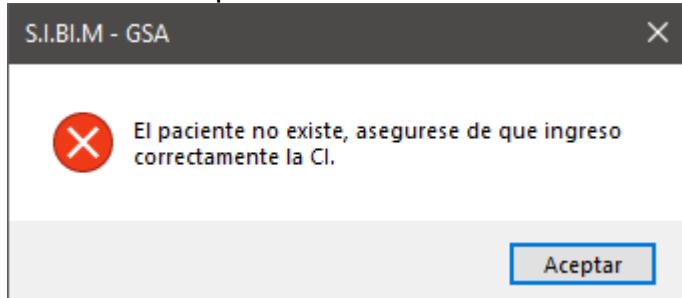


Una vez dentro de “1.2 Entrevista Inicial” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



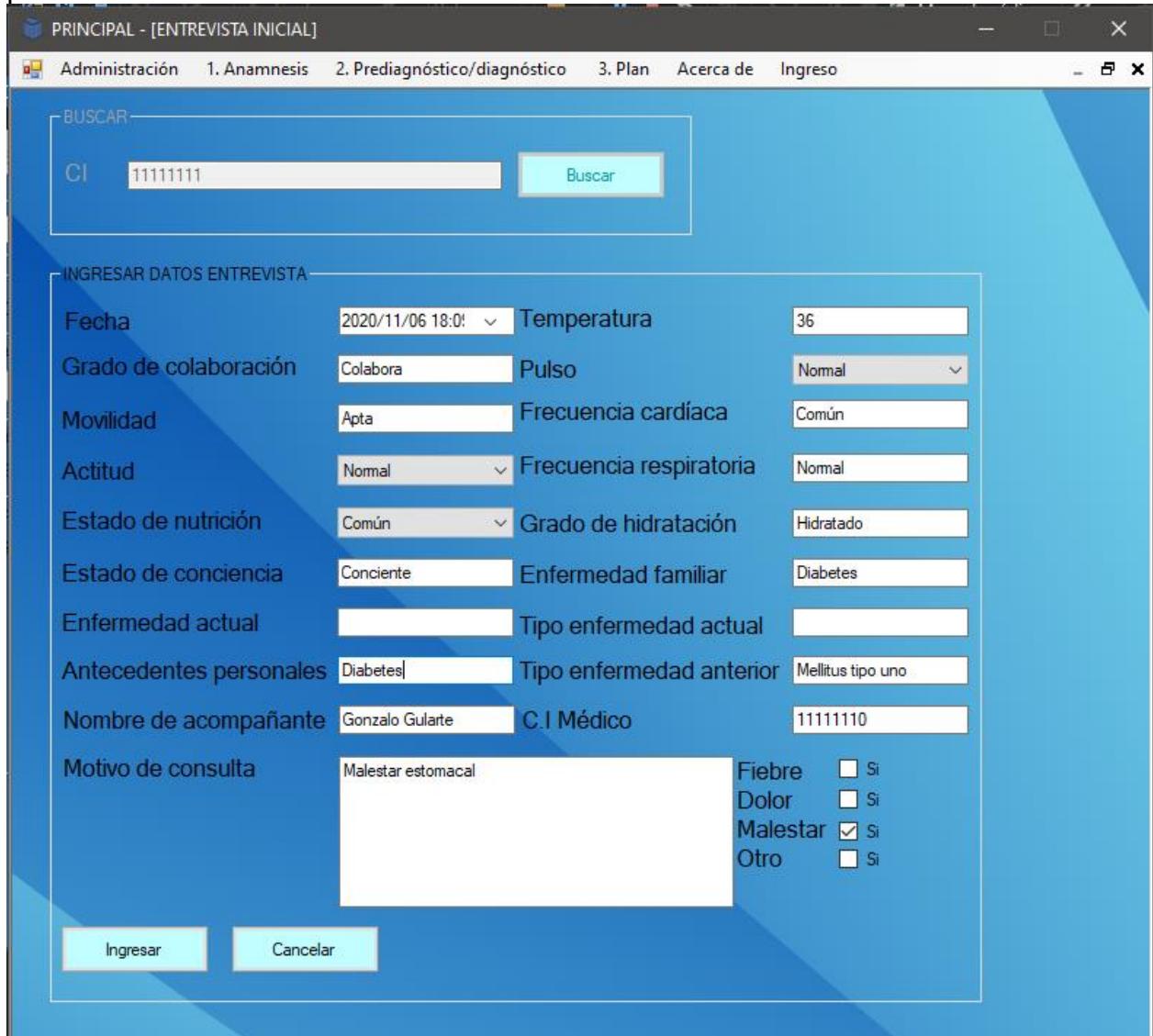
-Si el paciente no existe

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente.



-Si el paciente existe

En este caso el sistema procederá a habilitar una agrupación de datos, donde deberemos ingresar la información en los cuadros de texto que aparecen en pantalla.



PRINCIPAL - [ENTREVISTA INICIAL]

BUSCAR

CI 11111111 Buscar

INGRESAR DATOS ENTREVISTA

Fecha	2020/11/06 18:01	Temperatura	36
Grado de colaboración	Colabora	Pulso	Normal
Movilidad	Apta	Frecuencia cardíaca	Común
Actitud	Normal	Frecuencia respiratoria	Normal
Estado de nutrición	Común	Grado de hidratación	Hidratado
Estado de conciencia	Conciente	Enfermedad familiar	Diabetes
Enfermedad actual		Tipo enfermedad actual	
Antecedentes personales	Diabetes	Tipo enfermedad anterior	Mellitus tipo uno
Nombre de acompañante	Gonzalo Gularte	C.I Médico	11111110
Motivo de consulta	Malestar estomacal	Fiebre	<input type="checkbox"/> Si
		Dolor	<input type="checkbox"/> Si
		Malestar	<input checked="" type="checkbox"/> Si
		Otro	<input type="checkbox"/> Si

Ingresar Cancelar

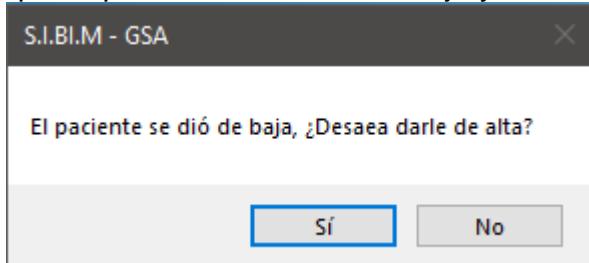
Una vez habilitada la agrupación de datos, debe proceder a cambiar la información en los cuadros de textos y listas desplegables que deseé ingresar, ya ingresada la información, deberá de pulsar el botón “Ingresar” para aplicar los cambios.

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó la entrevista inicial con éxito, de lo contrario el sistema notificará que hubo errores al ingresar la entrevista inicial.



-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.

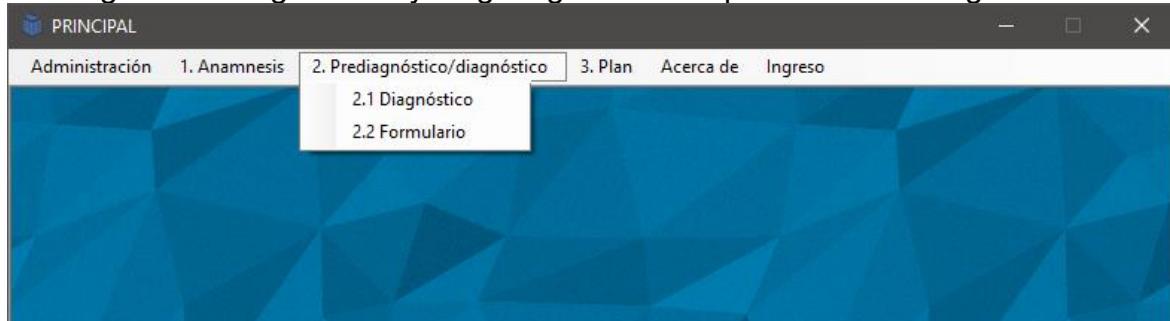


Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

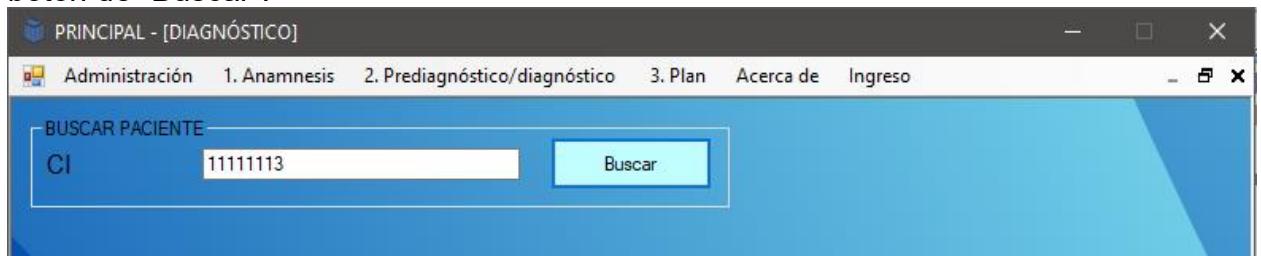


2. Prediagnóstico/diagnóstico - 2.1 Diagnóstico

Para ingresar un diagnóstico deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “2. Prediagnóstico/diagnóstico” y luego ingresar a la opción de “2.1 Diagnóstico”.

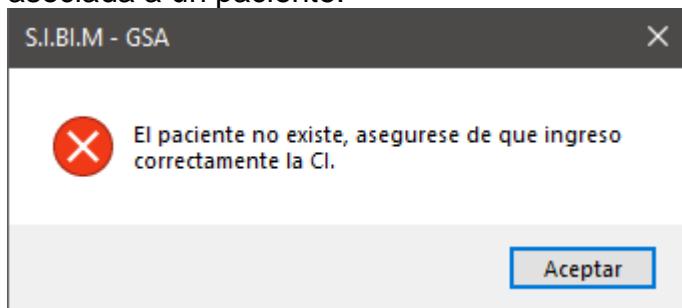


Una vez dentro de “2.1 Diagnóstico” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



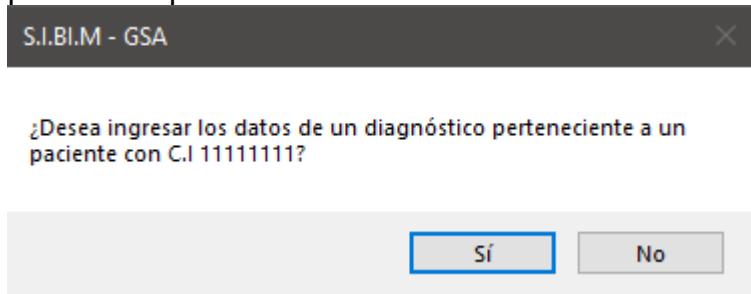
-Si el paciente no existe

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente.

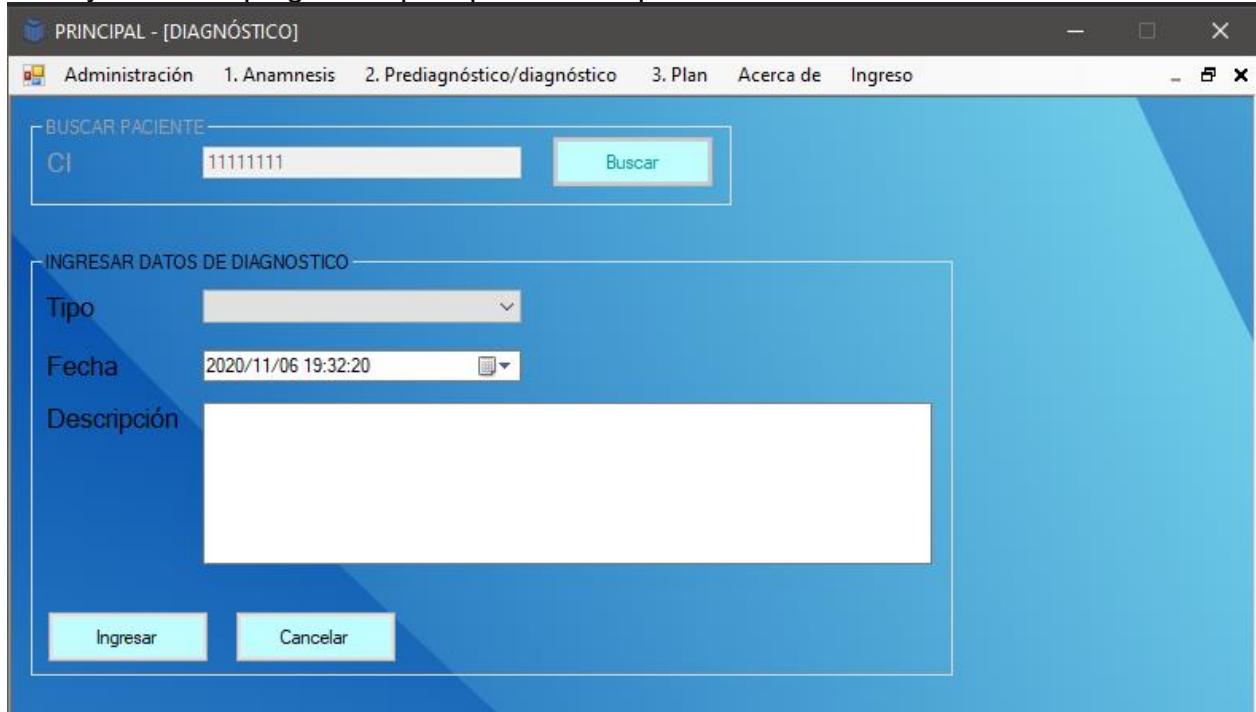


-Si el paciente existe

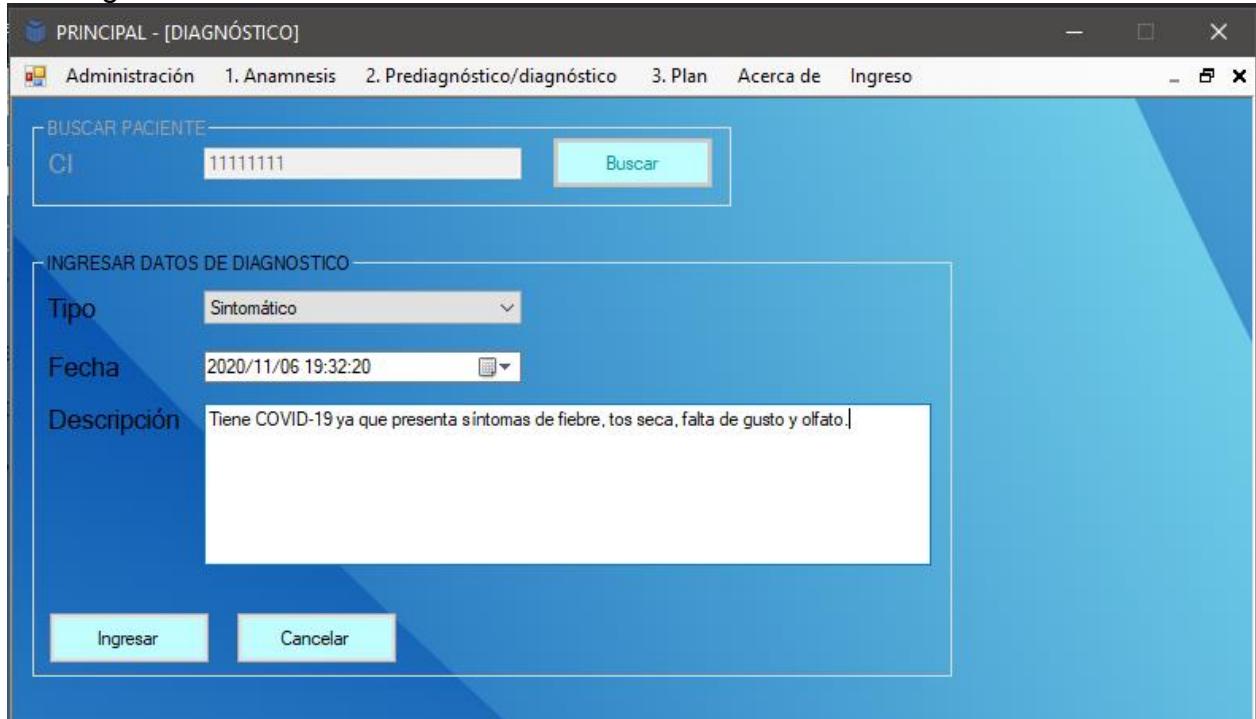
En este caso el sistema procederá a mostrar una ventana emergente que preguntará si se desea ingresar los datos de un diagnóstico perteneciente a un paciente con la cédula de identidad ingresada anteriormente y deberá de proceder a pulsar el botón “Sí”.



Una vez pulsado el botón “Sí” el sistema habilitará una agrupación de datos en donde se tendrá que ingresar los datos de un diagnóstico en los cuadros de texto y listas desplegables que aparecen en pantalla.



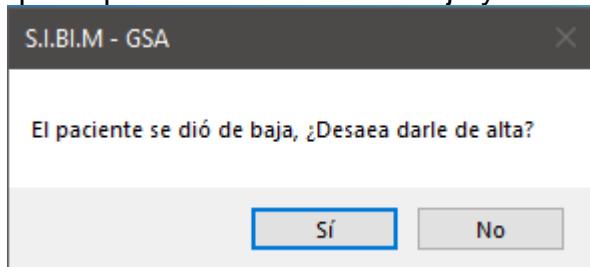
Ya ingresados los datos solicitados en los cuadros de texto y listas desplegables, se procederá a pulsar el botón “Ingresar” para realizar el ingreso del diagnóstico.



Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó el diagnóstico con éxito, de lo contrario el sistema notificará que hubo errores al ingresar el diagnóstico.

-Si el paciente existe pero fue dado de baja

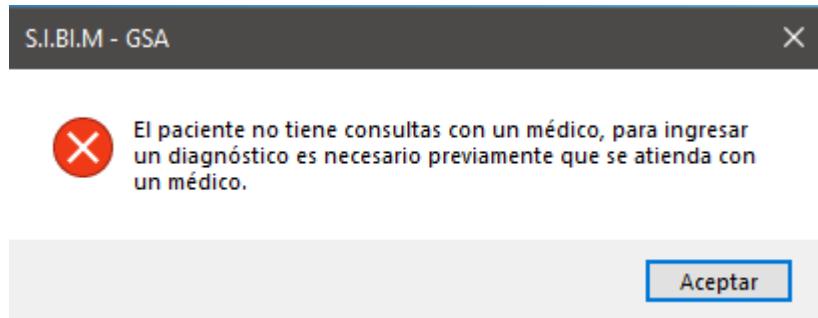
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

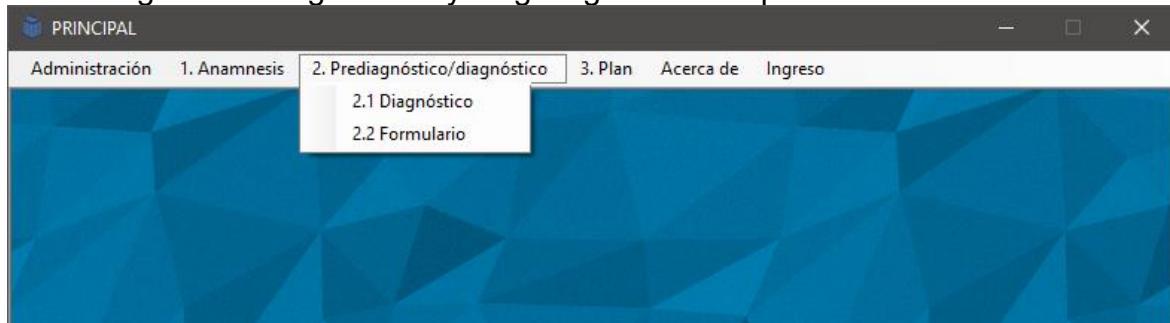


-Si el paciente existe pero no tuvo consultas con un médico
Para este caso el sistema desplegará una ventana emergente informando de que el paciente no tiene entrevistas realizadas con un médico y que para ingresar un diagnóstico es necesario previamente haber consultado con un médico.

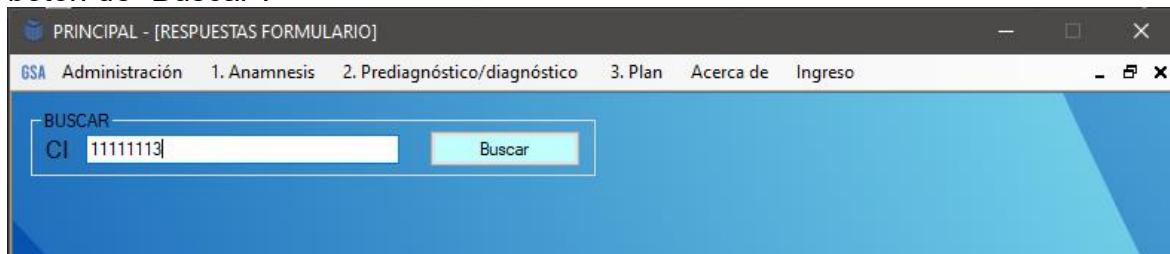


2. Prediagnóstico/diagnóstico - 2.2 Formulario

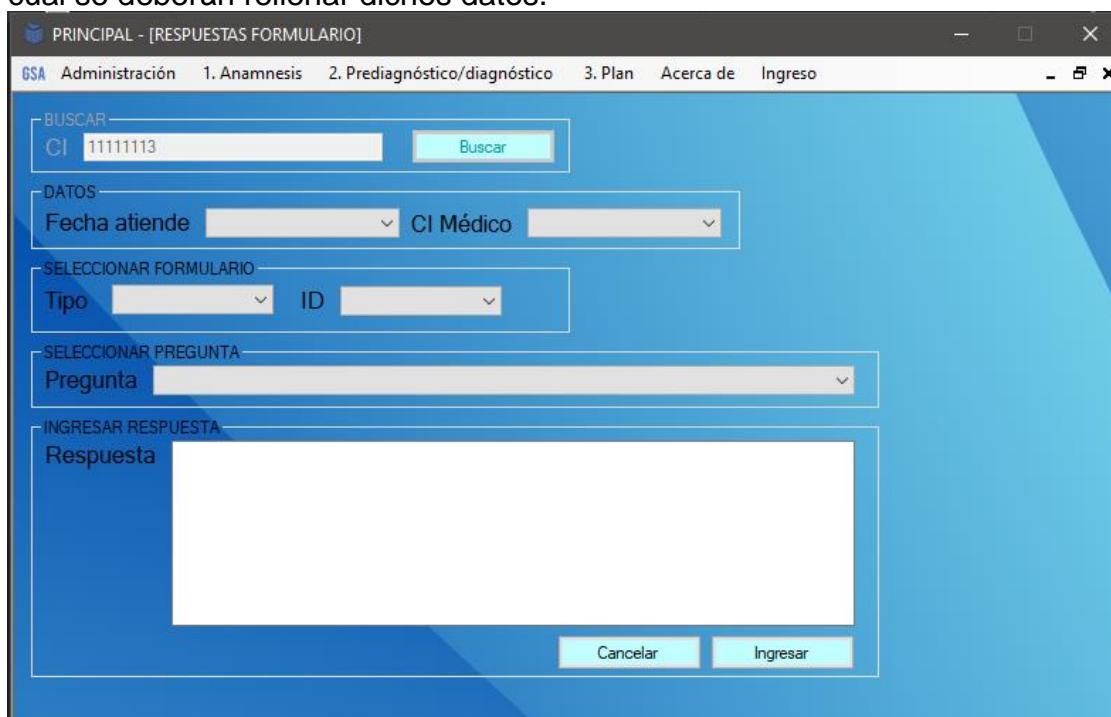
Para contestar las preguntas de un formulario deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección “2.Prediagnóstico/diagnóstico” y luego ingresar a la opción de “2.2 Formulario”.



Una vez dentro de “2.2 Formulario” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



Luego se habilitará una agrupación de datos pertenecientes al formulario en la cual se deberán llenar dichos datos.

A screenshot of the 'RESPUESTAS FORMULARIO' data entry screen. The window title is 'PRINCIPAL - [RESPUESTAS FORMULARIO]'. The menu bar includes 'GSA', 'Administración', '1. Anamnesis', '2. Prediagnóstico/diagnóstico' (highlighted in blue), '3. Plan', 'Acerca de', and 'Ingreso'. The screen is divided into several sections: 'BUSCAR' (with CI 11111113 and Buscar button), 'DATOS' (with 'Fecha atiende' and 'CI Médico' dropdowns), 'SELECCIONAR FORMULARIO' (with 'Tipo' and 'ID' dropdowns), 'SELECCIONAR PREGUNTA' (with 'Pregunta' dropdown), and 'INGRESAR RESPUESTA' (with a large text area for 'Respuesta' and 'Cancelar'/'Ingresar' buttons). The background has a blue gradient.

En fecha atiende, el sistema desplegará las fechas en las cuales el paciente se atendió y solo tendrá que seleccionar la fecha de la cual desea responder las preguntas .

DATOS

Fecha atiende	14/3/2002 17:30:25	CI Médico
SELECCIONAR FORMULARIO	4/10/2020 0:57:30	
Tipo	5/11/2020 17:01:09	
	5/11/2020 17:19:29	

En “CI Médico” le aparecerá la cédula del médico con el cual el paciente se atendió en la fecha anteriormente ingresada.

DATOS

Fecha atiende	5/11/2020 17:19:29	CI Médico
SELECCIONAR FORMULARIO		11111110

En el tipo de formulario seleccionará el tipo de formulario del cual desea contestar las preguntas.

SELECCIONAR FORMULARIO

Tipo	ID
Dolor	
Malestar	
Fiebre	

Luego de seleccionar el tipo, deberá seleccionar el número identificador (ID) del formulario a contestar.

SELECCIONAR FORMULARIO

Tipo	ID
Dolor	5
	1
SELECCIONAR PREGUNTA	5

Una vez ingresados los datos anteriores, en “Pregunta” se le desplegarán las preguntas que ese formulario posee.

SELECCIONAR PREGUNTA

Pregunta	¿Se golpeó en esa zona recientemente?
INGRESAR RESPUESTA	¿Hace cuanto siente el dolor?
Respuesta	Del nivel 1 a 10, ¿Que nivel de dolor siente?
	¿El dolor es permanente o solamente al realizar cierta actividad?



Al seleccionar la pregunta que desea responder, ingresará el texto en "Respuesta" y al finalizar le dará al botón "Ingresar" con el cual ingresará la respuesta anteriormente escrita en "Respuesta" al formulario ingresado al comienzo.

SELECCIONAR PREGUNTA

Pregunta	¿Se golpeó en esa zona recientemente?
----------	---------------------------------------

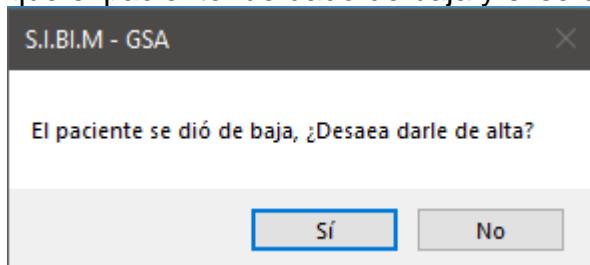
INGRESAR RESPUESTA

Respuesta	No tuvo un golpe en esa zona recientemente, la causa del dolor es desconocida.
-----------	--

Cancelar Ingresar

-Si el paciente existe pero fue dado de baja

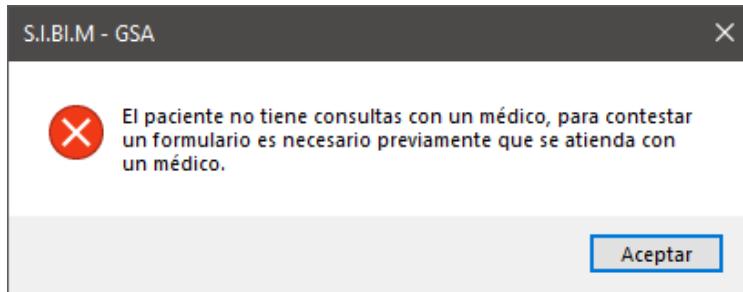
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón "Sí" el sistema notificará si el paciente se dio de alta exitosamente o no.

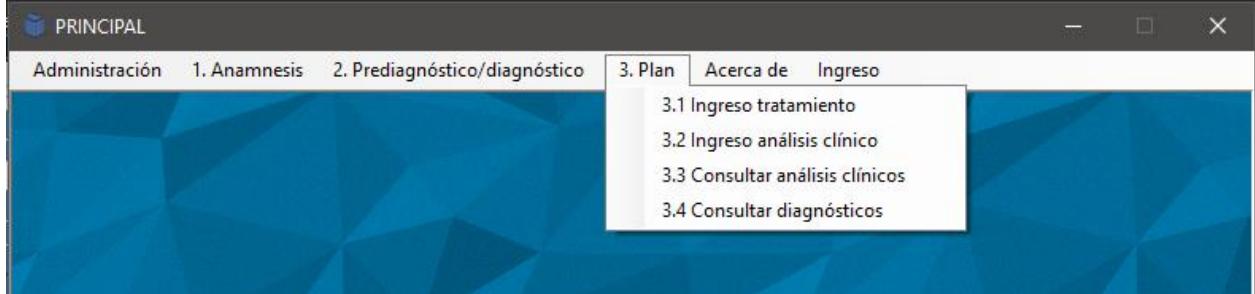
-Si el paciente existe pero no tuvo consultas con un médico

Para este caso el sistema desplegará una ventana emergente informando de que el paciente no tiene entrevistas realizadas con un médico y que para contestar un formulario es necesario previamente haber consultado con un médico.

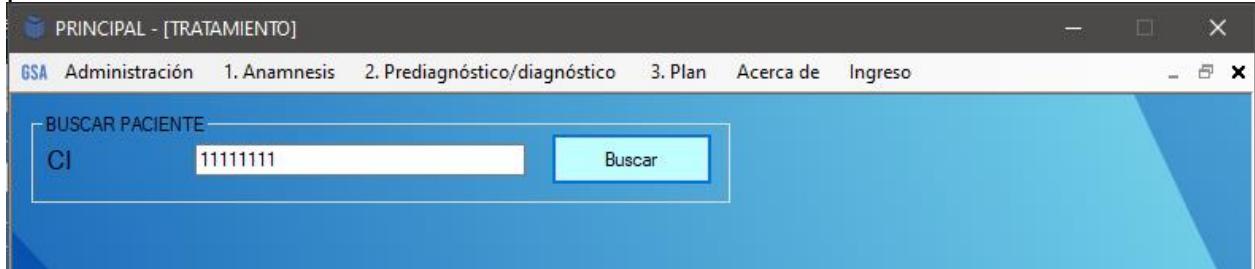


3. Plan - 3.1 Ingreso tratamiento

Para Ingresar un tratamiento deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “3. Plan” y luego ingresar a la opción de “3.1 Ingreso tratamiento”.

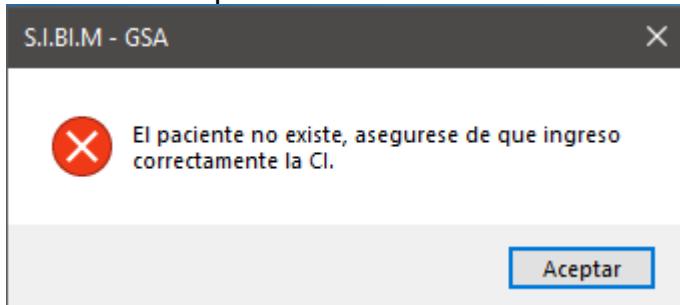


Una vez dentro de “3.1 Ingreso tratamiento” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



-Si el paciente no existe

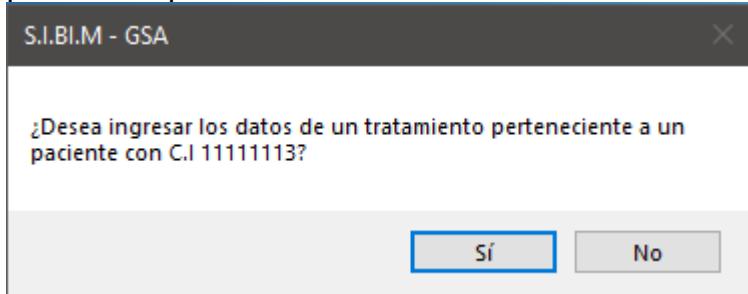
En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente



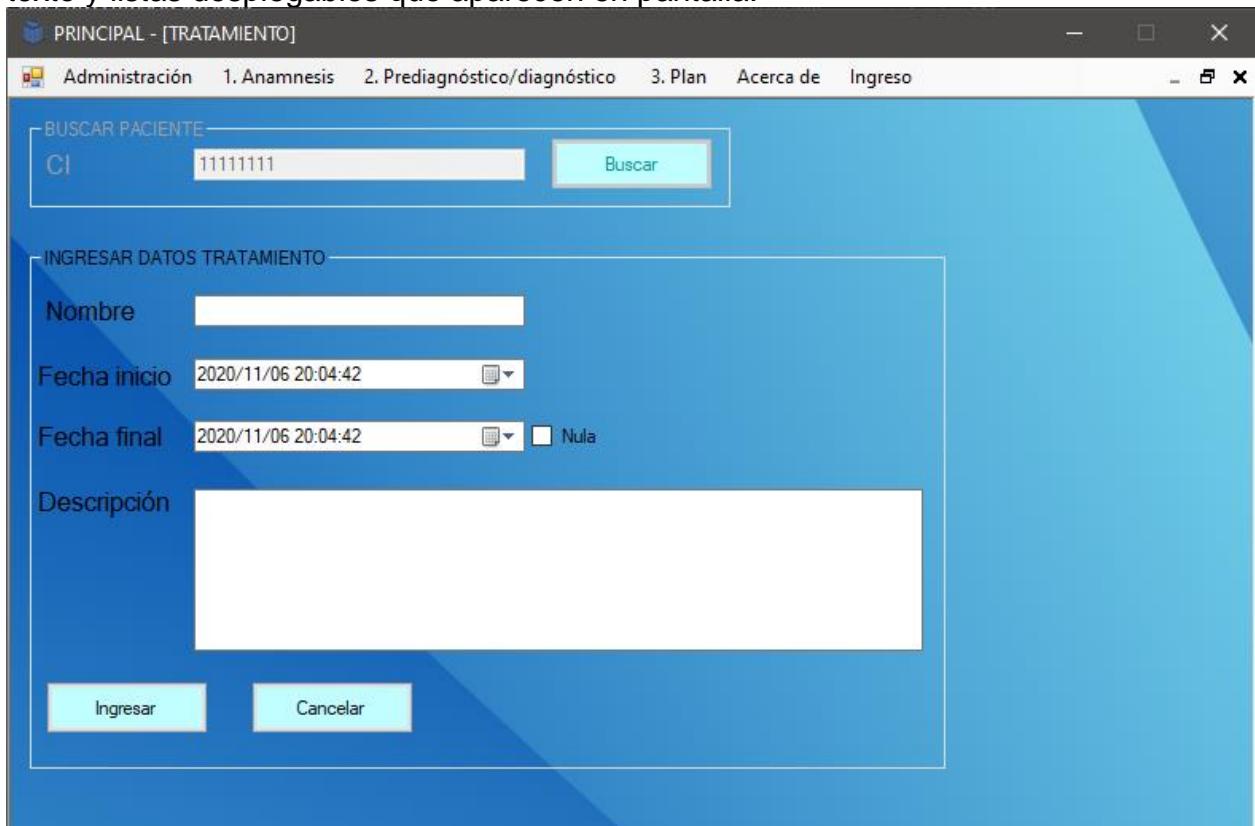


-Si el paciente existe

En este caso el sistema procederá a mostrar una ventana emergente que preguntará si se desea ingresar los datos de un tratamiento perteneciente a un paciente con la cédula de identidad ingresada anteriormente y deberá de proceder a pulsar el botón “Sí”.



Una vez pulsado el botón “Sí” el sistema habilitará una agrupación de datos en donde se tendrá que ingresar los datos de un tratamiento en los cuadros de texto y listas desplegables que aparecen en pantalla.



Ya ingresados los datos solicitados en los cuadros de texto y listas desplegables, se procederá a pulsar el botón “Ingresar” para realizar el ingreso del tratamiento.



PRINCIPAL - [TRATAMIENTO]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR PACIENTE
CI: 11111111 Buscar

INGRESAR DATOS TRATAMIENTO

Nombre: Fiebre

Fecha inicio: 2020/11/06 20:04:42

Fecha final: 2021/01/10 20:04:42 Nula

Descripción: Perifar 500

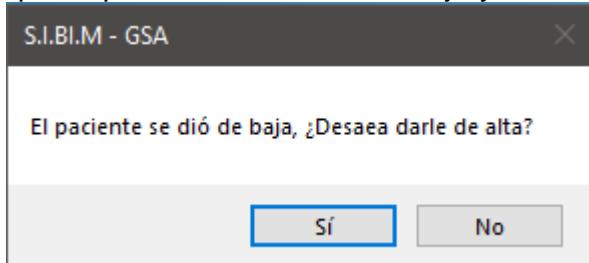
Ingresar Cancelar

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó el tratamiento con éxito, de lo contrario el sistema notificará que hubo errores al ingresar el tratamiento.



-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.

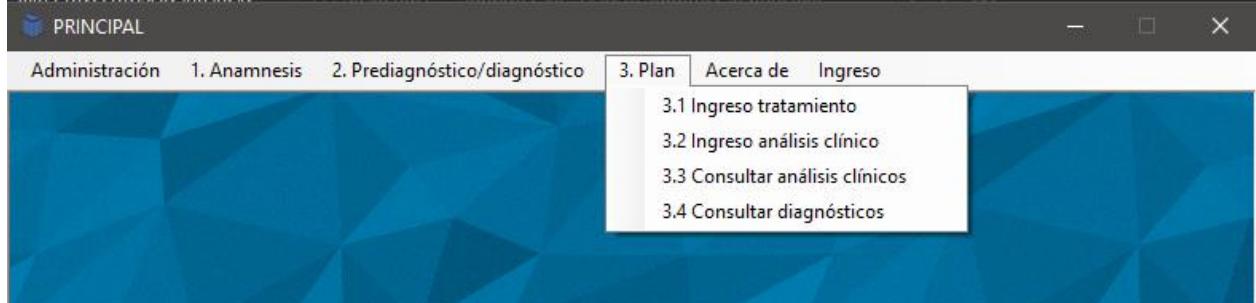


Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

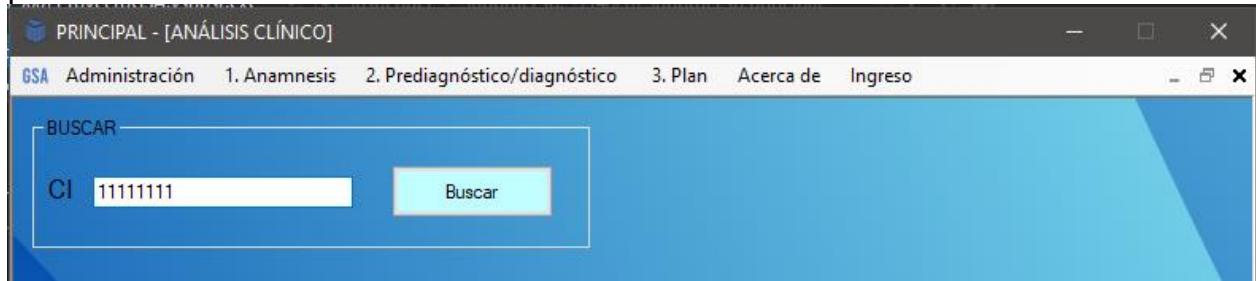


3. Plan - 3.2 Ingreso análisis clínico

Para Ingresar el análisis clínico deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “3. Plan” y luego ingresar a la opción de “3.2 Ingreso análisis clínico”.

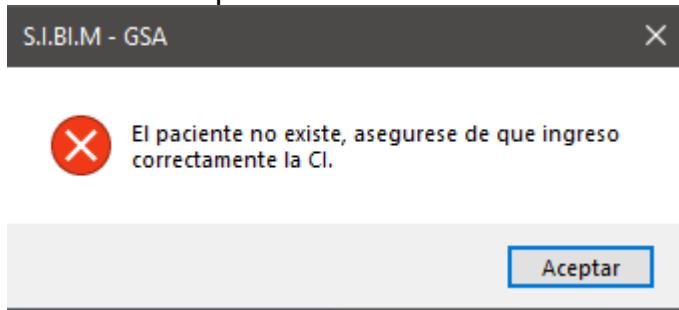


Una vez dentro de “3.2 Ingreso análisis clínico” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”



-Si el paciente no existe

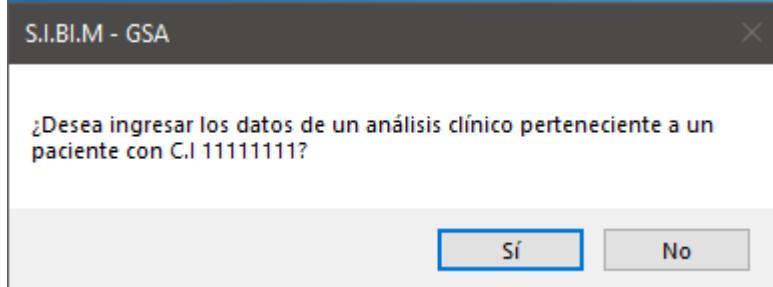
En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente



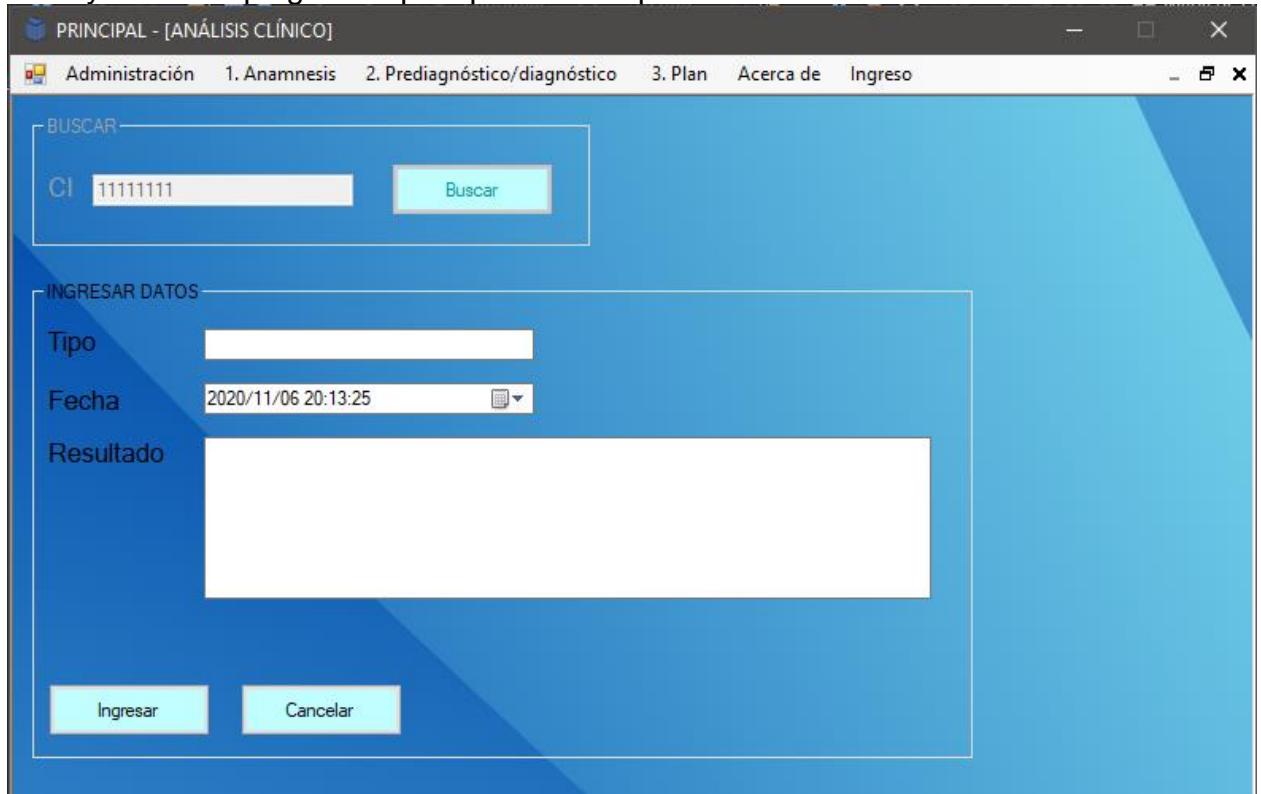


-Si el paciente existe

En este caso el sistema procederá a mostrar una ventana emergente que preguntará si se desea ingresar los datos de un análisis clínico perteneciente a un paciente con la cédula de identidad ingresada anteriormente y deberá de proceder a pulsar el botón “Sí”.



Una vez pulsado el botón “Sí” el sistema habilitará una agrupación de datos en donde se tendrá que ingresar los datos de un tratamiento en los cuadros de texto y listas desplegables que aparecen en pantalla.



Ya ingresados los datos solicitados en los cuadros de texto y listas desplegables, se procederá a pulsar el botón “Ingresar” para realizar el ingreso del análisis clínico.



PRINCIPAL - [ANÁLISIS CLÍNICO]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR

CI 11111111 Buscar

INGRESAR DATOS

Tipo Radiografía

Fecha 2020/11/06 20:13:25

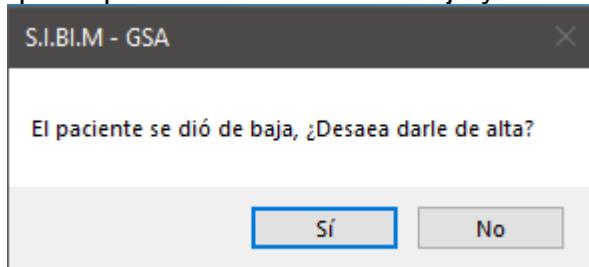
Resultado Hombro deslocado

Ingresar Cancelar

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó el análisis clínico con éxito, de lo contrario el sistema notificará que hubo errores al ingresar el análisis clínico.

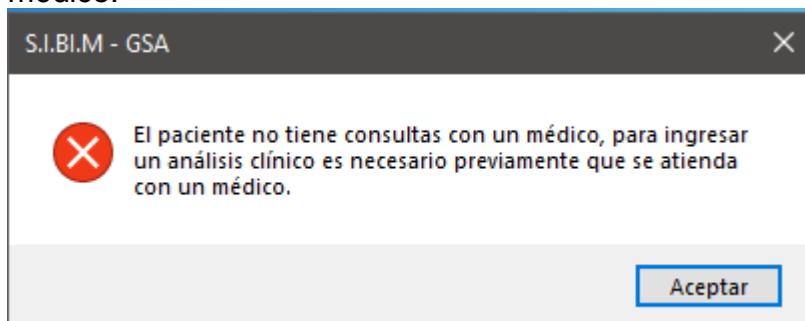
-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón "Sí" el sistema notificará si el paciente se dio de alta exitosamente o no.

-Si el paciente existe pero no tuvo consultas con un médico
Para este caso el sistema desplegará una ventana emergente informando de que el paciente no tiene entrevistas realizadas con un médico y que para ingresar un análisis clínico es necesario previamente haber consultado con un médico.





Usuario - Auxiliar Médico

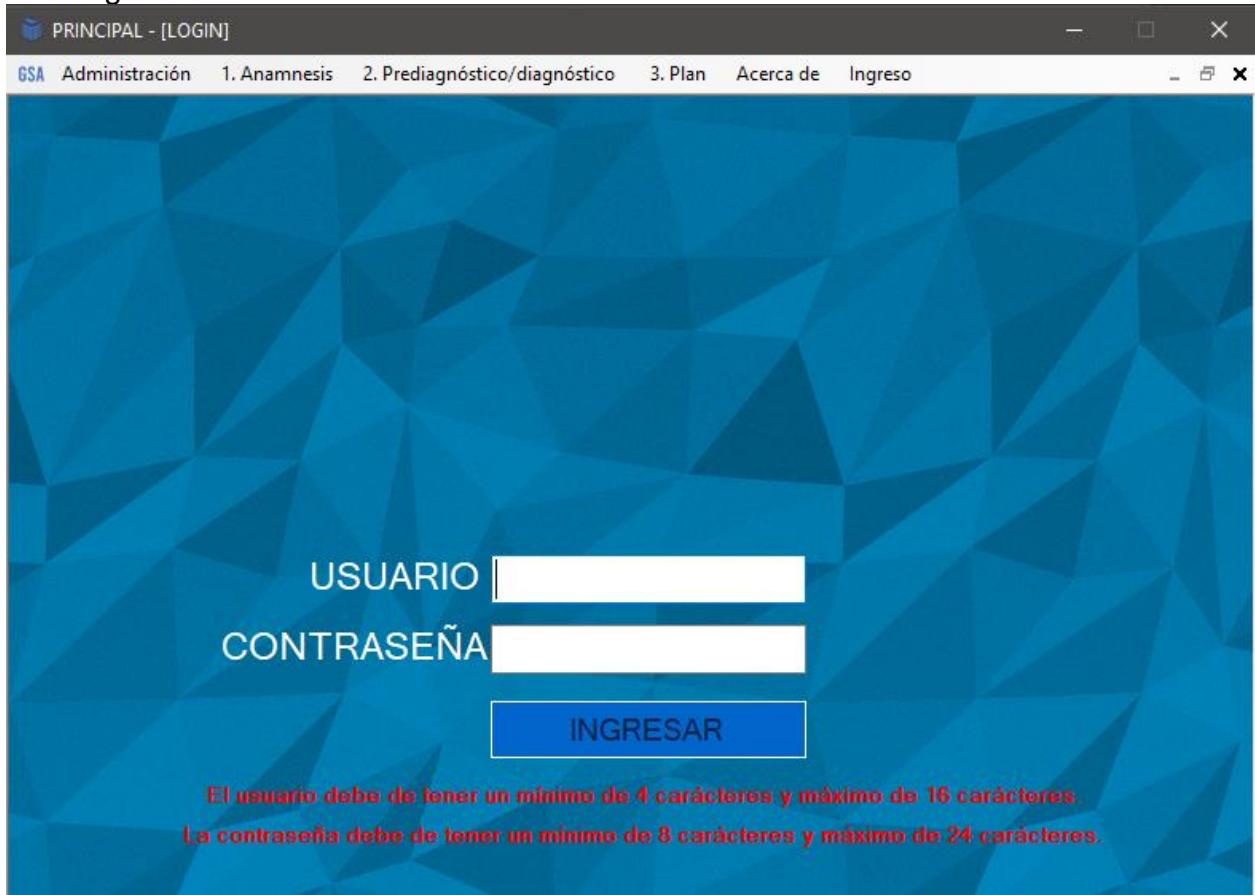
Login

Para ingresar al sistema se deberá de dirigir al menú ubicado en la parte superior del programa, deberá entrar a la sección de "Ingreso" y luego ingresar a la opción de "Login".



-Si la sesión no está iniciada

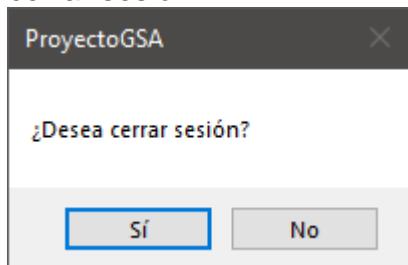
Deberá de ingresar un usuario, sin espacios y que cumpla con el requisito de mínimo 4 caracteres y máximo de 24 caracteres junto con una contraseña, sin espacios y que deberá cumplir un requisito de mínimo de 8 caracteres y máximo 24 caracteres, estos datos deberán de pertenecer a un usuario que ya esté registrado en el sistema.





-Si la sesión está iniciada

El sistema desplegará una ventana emergente la cual preguntará si desea cerrar sesión.

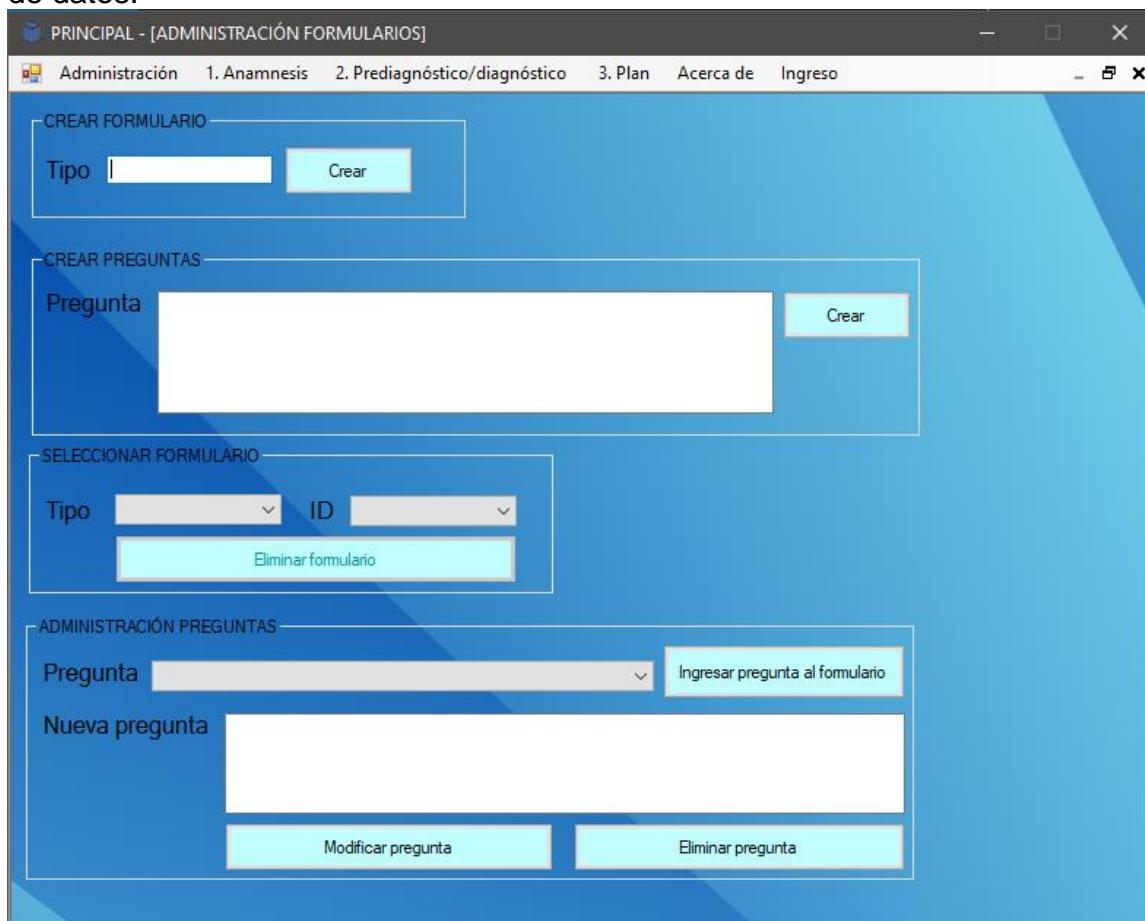


Administración - Formularios

Para administrar formularios deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “Administración” y luego ingresar a la opción de “Formularios”.



Una vez dentro de “Formulario” el sistema habilitará el ingreso y modificación de datos.





-Crear formulario

Para crear un formulario será necesario ingresar en el cuadro de texto que se ubica a la derecha de la etiqueta con texto “Tipo”, el tipo que corresponde al formulario que se desea crear.

The screenshot shows a software interface titled "PRINCIPAL - [ADMINISTRACIÓN FORMULARIOS]". In the top menu bar, there are several options: "Administración", "1. Anamnesis", "2. Prediagnóstico/diagnóstico", "3. Plan", "Acerca de", and "Ingreso". Below the menu, a sub-section titled "CREAR FORMULARIO" is displayed. It contains a label "Tipo" followed by a text input field containing the text "Anamnesis". To the right of the input field is a blue "Crear" button. The background of the main window is light blue.

-Crear preguntas

Para crear una pregunta será necesario ingresar en el cuadro de texto que se ubica a la derecha de la etiqueta con texto “Pregunta”, la pregunta que se desea crear.

The screenshot shows a software interface titled "CREAR PREGUNTAS". On the left side, there is a label "Pregunta" next to a large white text input field. On the right side, there is a blue "Crear" button. The background is light blue.

-Eliminar formulario

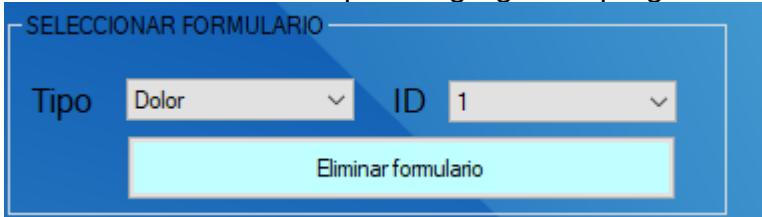
Para eliminar un formulario será necesario previamente seleccionar el tipo de formulario y luego el número identificador (ID) perteneciente al formulario que se desea eliminar.

The screenshot shows a software interface titled "SELECCIONAR FORMULARIO". It features two dropdown menus: "Tipo" set to "Dolor" and "ID" set to "2". Below these dropdowns is a blue "Eliminar formulario" button.

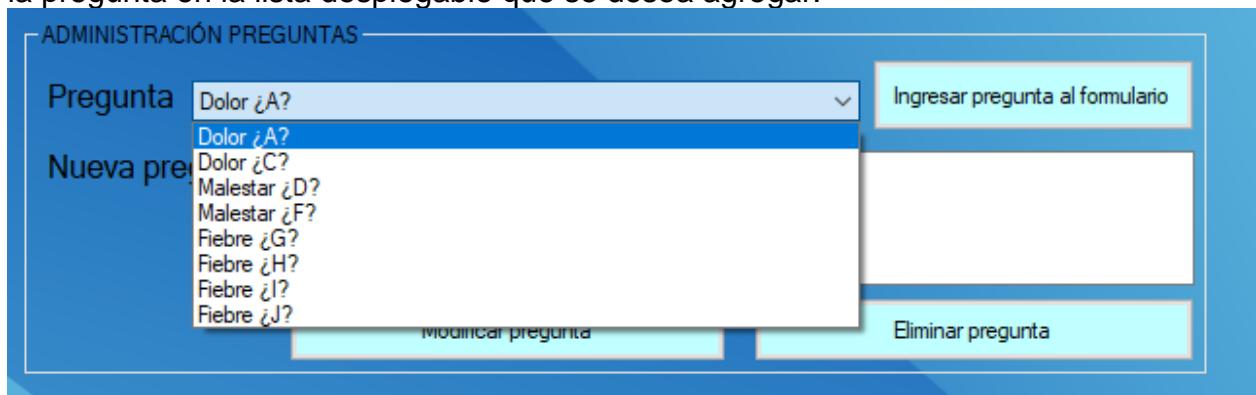
Una vez seleccionados los datos solicitados anteriormente, deberá de pulsar el botón de “Eliminar formulario”, si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que el formulario se eliminó con éxito, de lo contrario el sistema notificará que hubo errores al eliminar el formulario.

-Ingresar preguntas al formulario

Para ingresar preguntas al formulario será necesario previamente seleccionar el tipo de formulario y luego el número identificador (ID) perteneciente al formulario al cual se le quiere agregar las preguntas.



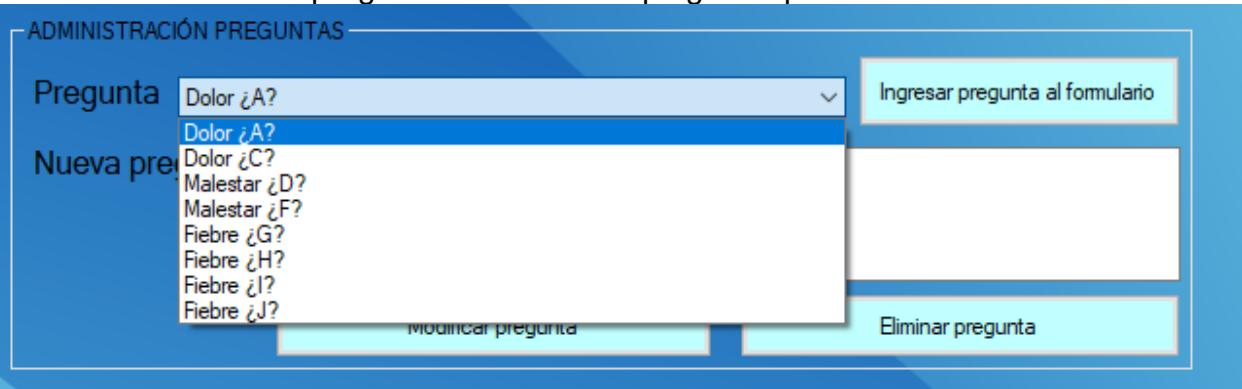
Una vez seleccionados los datos solicitados anteriormente, deberá seleccionar la pregunta en la lista desplegable que se desea agregar.



Una vez seleccionada la pregunta, deberá de pulsar el botón “Ingresar pregunta al formulario”, si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó la pregunta al formulario con éxito, de lo contrario el sistema notificará que hubo errores al ingresar la pregunta al formulario.

-Modificar pregunta

Deberá seleccionar la pregunta en la lista desplegable que se desea eliminar.



ADMINISTRACIÓN PREGUNTAS

Pregunta Dolor ¿A?

Dolor ¿A?
Dolor ¿C?
Malestar ¿D?
Malestar ¿F?
Fiebre ¿G?
Fiebre ¿H?
Fiebre ¿I?
Fiebre ¿J?

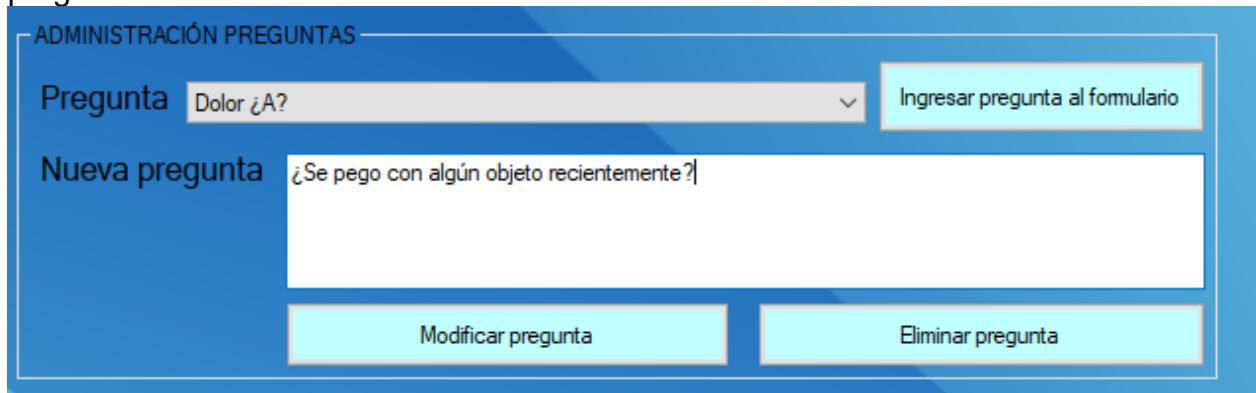
Nueva pregunta

Ingresar pregunta al formulario

Modificar pregunta

Eliminar pregunta

Una vez seleccionada la pregunta, tendrá que pulsar el botón “Eliminar pregunta”.



ADMINISTRACIÓN PREGUNTAS

Pregunta Dolor ¿A?

Nueva pregunta ¿Se pego con algún objeto recientemente?

Ingresar pregunta al formulario

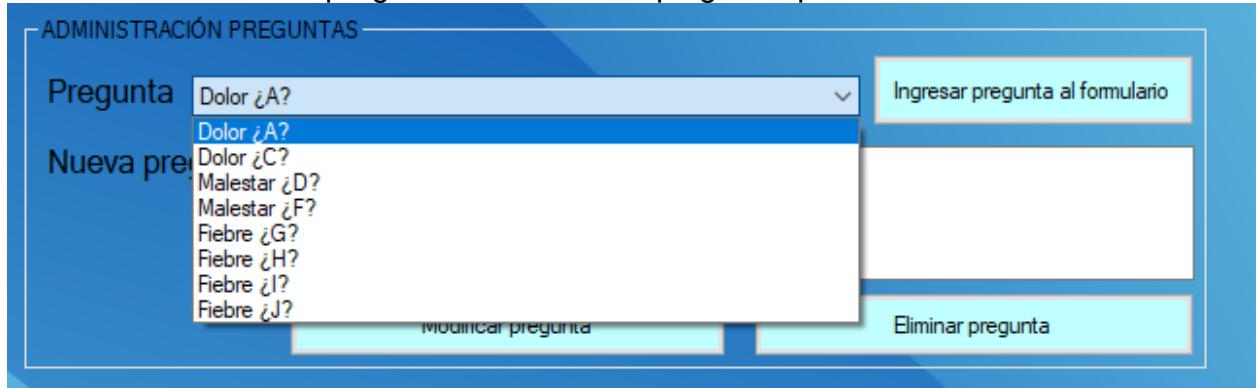
Modificar pregunta

Eliminar pregunta

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se eliminó la pregunta con éxito, de lo contrario el sistema notificará que hubo errores al eliminar la pregunta.

-Eliminar pregunta

Deberá seleccionar la pregunta en la lista desplegable que se desea modificar.

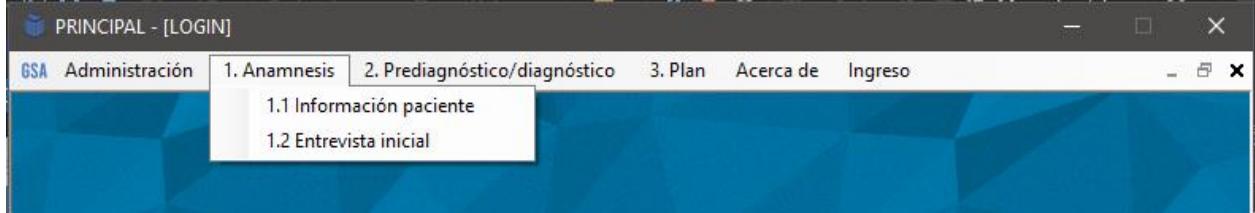


Una vez seleccionada la pregunta, tendrá que ingresar la nueva pregunta que deseé en el cuadro de texto ubicado a la derecha de la etiqueta de texto "Nueva pregunta".

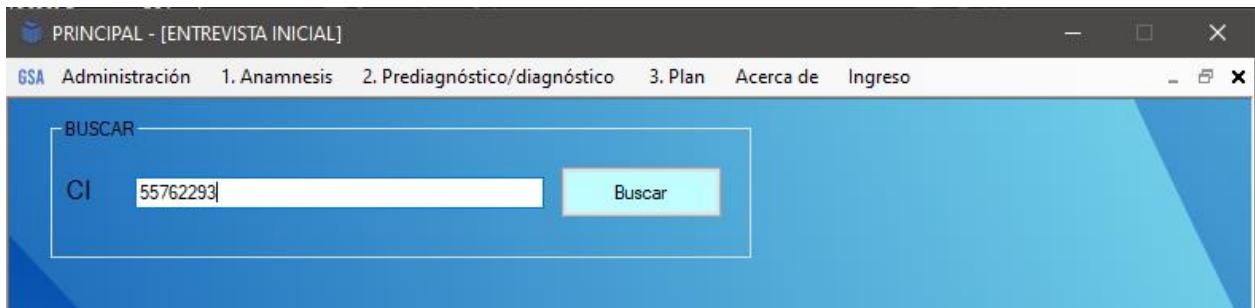


1. Anamnesis - 1.2 Entrevista inicial

Para Ingresar la entrevista inicial deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “1. Anámnasis” y luego ingresar a la opción de “1.2 Entrevista Inicial”.

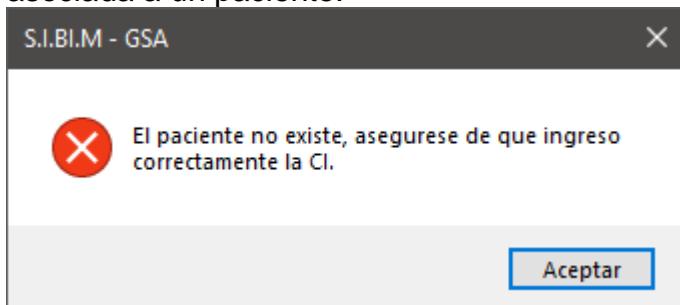


Una vez dentro de “1.2 Entrevista Inicial” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



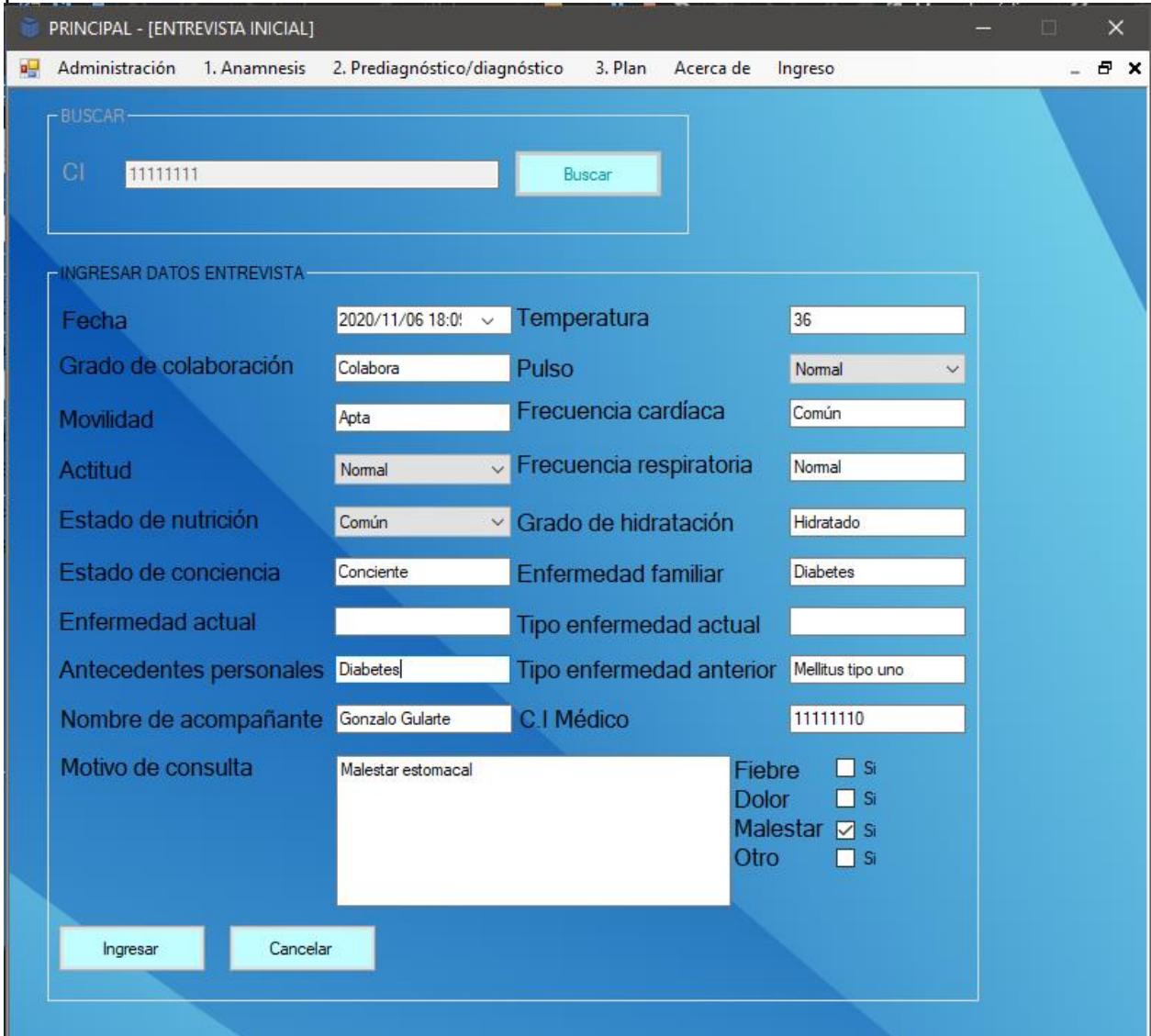
-Si el paciente no existe

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente.



-Si el paciente existe

En este caso el sistema procederá a habilitar una agrupación de datos, donde deberemos ingresar la información en los cuadros de texto que aparecen en pantalla.



The screenshot shows the 'PRINCIPAL - [ENTREVISTA INICIAL]' application window. At the top, there is a menu bar with options: Administración, 1. Anamnesis, 2. Prediagnóstico/diagnóstico, 3. Plan, Acerca de, and Ingreso. Below the menu, there is a search bar labeled 'BUSCAR' with a placeholder 'CI' and a text input field containing '1111111111'. A 'Buscar' button is next to the input field. The main area is titled 'INGRESAR DATOS ENTREVISTA' and contains a grid of data entry fields. The fields include:

Fecha	2020/11/06 18:01	Temperatura	36
Grado de colaboración	Colabora	Pulso	Normal
Movilidad	Apta	Frecuencia cardíaca	Común
Actitud	Normal	Frecuencia respiratoria	Normal
Estado de nutrición	Común	Grado de hidratación	Hidratado
Estado de conciencia	Conciente	Enfermedad familiar	Diabetes
Enfermedad actual		Tipo enfermedad actual	
Antecedentes personales	Diabetes	Tipo enfermedad anterior	Mellitus tipo uno
Nombre de acompañante	Gonzalo Gularte	C.I Médico	11111110
Motivo de consulta	Malestar estomacal	Fiebre	<input type="checkbox"/> Si
		Dolor	<input type="checkbox"/> Si
		Malestar	<input checked="" type="checkbox"/> Si
		Otro	<input type="checkbox"/> Si

At the bottom of the form are two buttons: 'Ingresar' (Enter) and 'Cancelar' (Cancel).

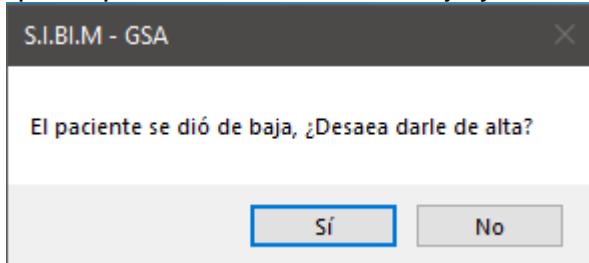
Una vez habilitada la agrupación de datos, debe proceder a cambiar la información en los cuadros de textos y listas desplegables que deseé ingresar, ya ingresada la información, deberá de pulsar el botón "Ingresar" para aplicar los cambios.

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó la entrevista inicial con éxito, de lo contrario el sistema notificará que hubo errores al ingresar la entrevista inicial.



-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.

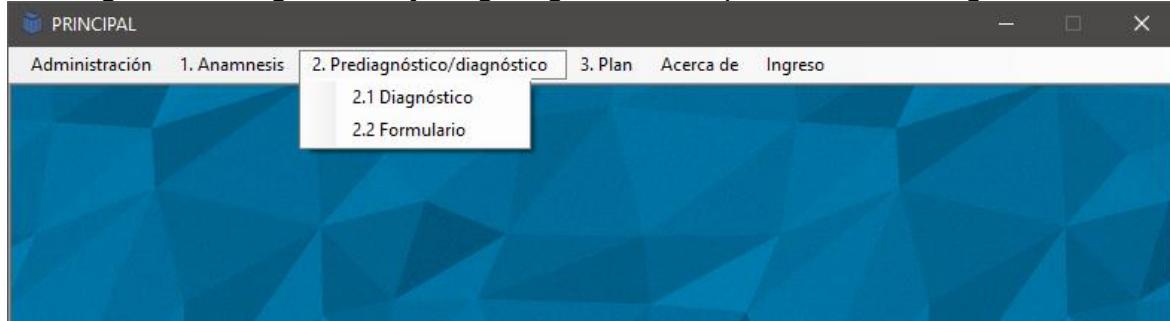


Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

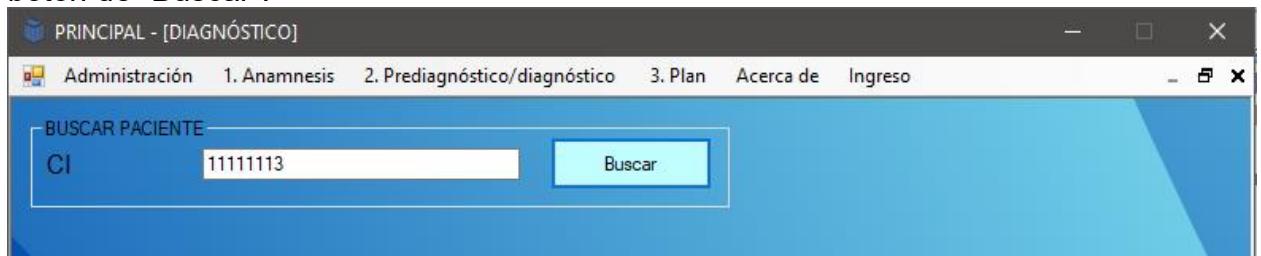


2. Prediagnóstico/diagnóstico - 2.1 Diagnóstico

Para ingresar un diagnóstico deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “2. Prediagnóstico/diagnóstico” y luego ingresar a la opción de “2.1 Diagnóstico”.

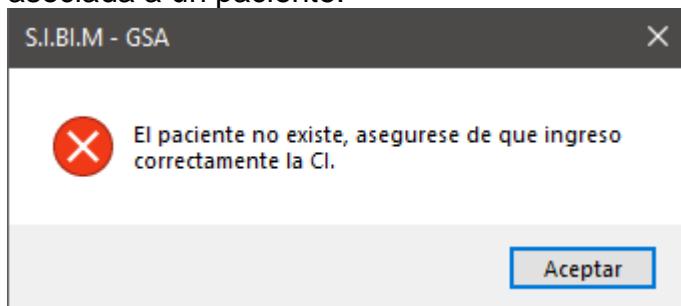


Una vez dentro de “2.1 Diagnóstico” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



-Si el paciente no existe

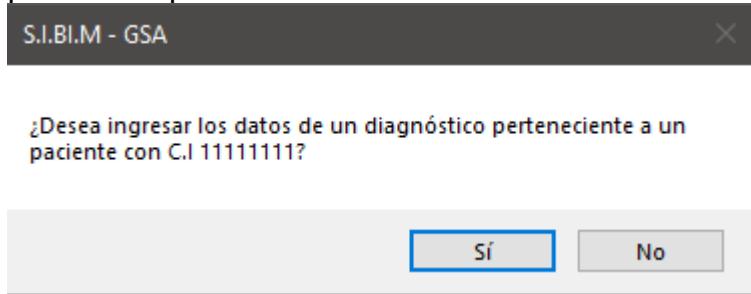
En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente.



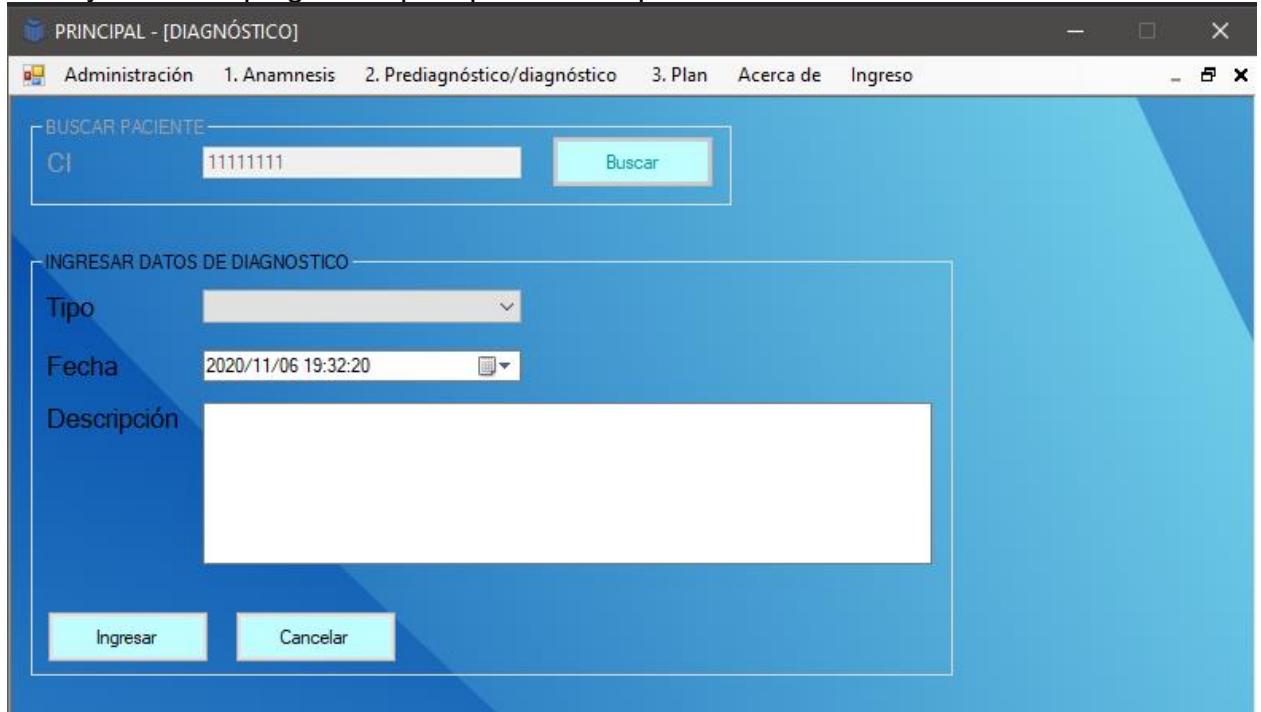


-Si el paciente existe

En este caso el sistema procederá a mostrar una ventana emergente que preguntará si se desea ingresar los datos de un diagnóstico perteneciente a un paciente con la cédula de identidad ingresada anteriormente y deberá de proceder a pulsar el botón “Sí”.



Una vez pulsado el botón “Sí” el sistema habilitará una agrupación de datos en donde se tendrá que ingresar los datos de un diagnóstico en los cuadros de texto y listas desplegables que aparecen en pantalla.



Ya ingresados los datos solicitados en los cuadros de texto y listas desplegables, se procederá a pulsar el botón “Ingresar” para realizar el ingreso del diagnóstico.



PRINCIPAL - [DIAGNÓSTICO]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR PACIENTE
CI: 11111111 Buscar

INGRESAR DATOS DE DIAGNÓSTICO

Tipo: Sintomático

Fecha: 2020/11/06 19:32:20

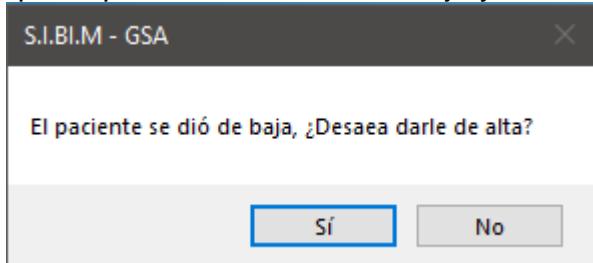
Descripción: Tiene COVID-19 ya que presenta síntomas de fiebre, tos seca, falta de gusto y olfato.

Ingresar Cancelar

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó el diagnóstico con éxito, de lo contrario el sistema notificará que hubo errores al ingresar el diagnóstico.

-Si el paciente existe pero fue dado de baja

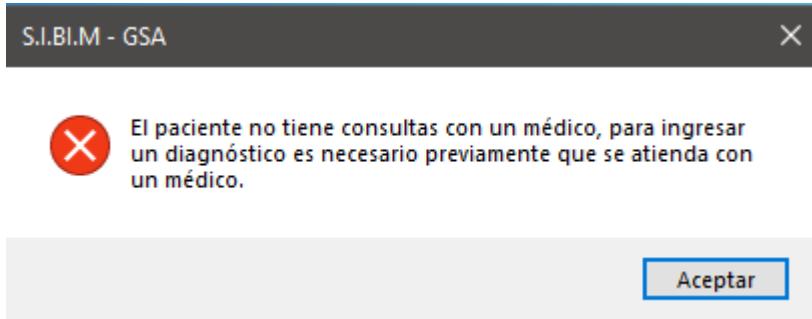
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón "Sí" el sistema notificará si el paciente se dio de alta exitosamente o no.

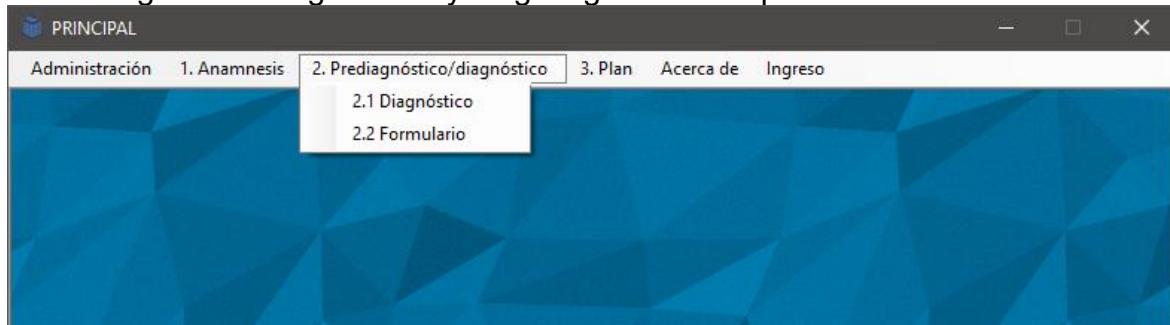
-Si el paciente existe pero no tuvo consultas con un médico

Para este caso el sistema desplegará una ventana emergente informando de que el paciente no tiene entrevistas realizadas con un médico y que para ingresar un diagnóstico es necesario previamente haber consultado con un médico.

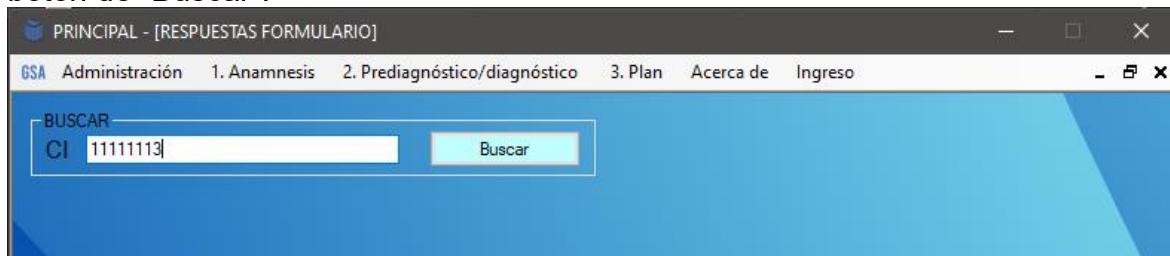


2. Prediagnóstico/diagnóstico - 2.2 Formulario

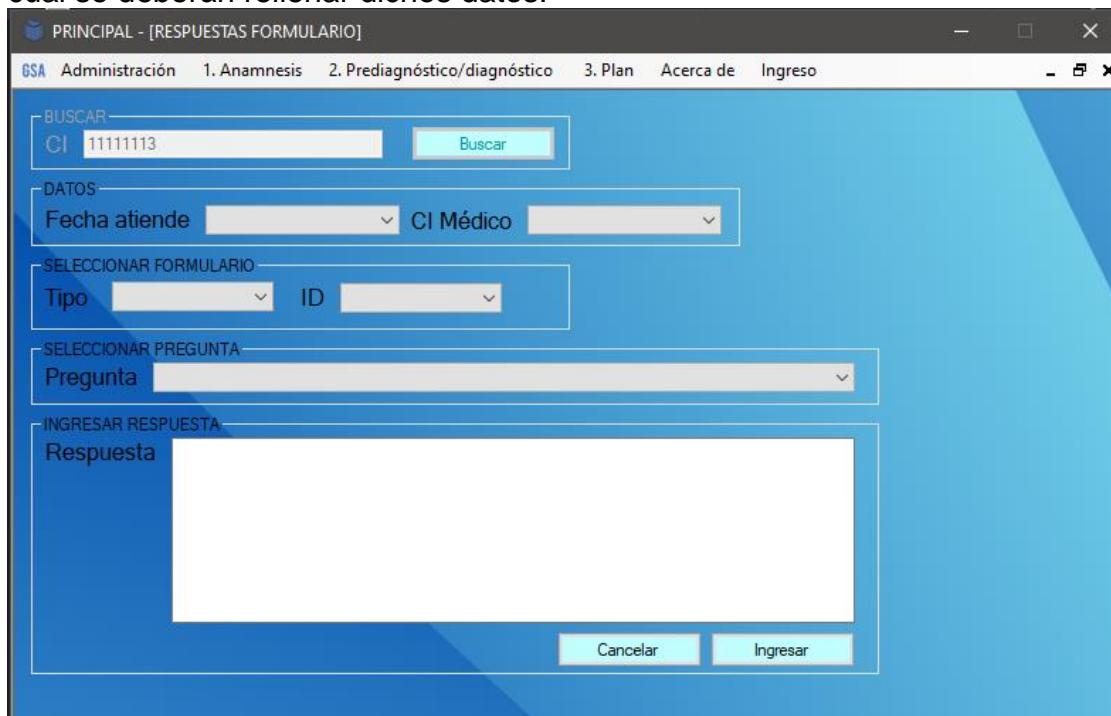
Para contestar las preguntas de un formulario deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección “2.Prediagnóstico/diagnóstico” y luego ingresar a la opción de “2.2 Formulario”.



Una vez dentro de “2.2 Formulario” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



Luego se habilitará una agrupación de datos pertenecientes al formulario en la cual se deberán llenar dichos datos.

A screenshot of the 'RESPUESTAS FORMULARIO' data entry screen. The window title is 'PRINCIPAL - [RESPUESTAS FORMULARIO]'. The menu bar includes 'GSA', 'Administración', '1. Anamnesis', '2. Prediagnóstico/diagnóstico' (highlighted in blue), '3. Plan', 'Acerca de', and 'Ingreso'. The screen is divided into several sections: 'BUSCAR' (with CI 11111113 and Buscar button), 'DATOS' (with 'Fecha atiende' and 'CI Médico' dropdowns), 'SELECCIONAR FORMULARIO' (with 'Tipo' and 'ID' dropdowns), 'SELECCIONAR PREGUNTA' (with 'Pregunta' dropdown), and 'INGRESAR RESPUESTA' (with a large text area for 'Respuesta' and 'Cancelar'/'Ingresar' buttons). The background has a blue gradient.

En fecha atiende, el sistema desplegará las fechas en las cuales el paciente se atendió y solo tendrá que seleccionar la fecha de la cual desea responder las preguntas .

DATOS

Fecha atiende	14/3/2002 17:30:25	CI Médico
SELECCIONAR FORMULARIO	4/10/2020 0:57:30	
Tipo	5/11/2020 17:01:09	
	5/11/2020 17:19:29	

En “CI Médico” le aparecerá la cédula del médico con el cual el paciente se atendió en la fecha anteriormente ingresada.

DATOS

Fecha atiende	5/11/2020 17:19:29	CI Médico
SELECCIONAR FORMULARIO		11111110

En el tipo de formulario seleccionará el tipo de formulario del cual desea contestar las preguntas.

SELECCIONAR FORMULARIO

Tipo	ID
Dolor	
Malestar	
Fiebre	

Luego de seleccionar el tipo, deberá seleccionar el número identificador (ID) del formulario a contestar.

SELECCIONAR FORMULARIO

Tipo	ID
Dolor	5
	1
SELECCIONAR PREGUNTA	5

Una vez ingresados los datos anteriores, en “Pregunta” se le desplegarán las preguntas que ese formulario posee.

SELECCIONAR PREGUNTA

Pregunta	¿Se golpeó en esa zona recientemente?
INGRESAR RESPUESTA	¿Hace cuanto siente el dolor?
Respuesta	Del nivel 1 a 10, ¿Que nivel de dolor siente?
	¿El dolor es permanente o solamente al realizar cierta actividad?



Al seleccionar la pregunta que desea responder, ingresará el texto en “Respuesta” y al finalizar le dará al botón “Ingresar” con el cual ingresará la respuesta anteriormente escrita en “Respuesta” al formulario ingresado al comienzo.

SELECCIONAR PREGUNTA

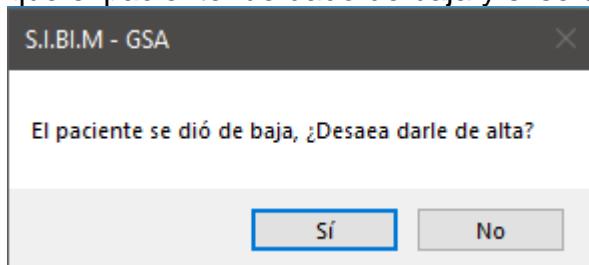
Pregunta ¿Se golpeó en esa zona recientemente?

INGRESAR RESPUESTA

Respuesta No tuvo un golpe en esa zona recientemente, la causa del dolor es desconocida.

-Si el paciente existe pero fue dado de baja

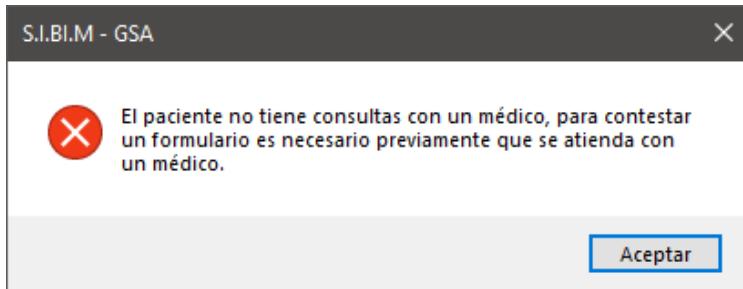
Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón “Sí” el sistema notificará si el paciente se dio de alta exitosamente o no.

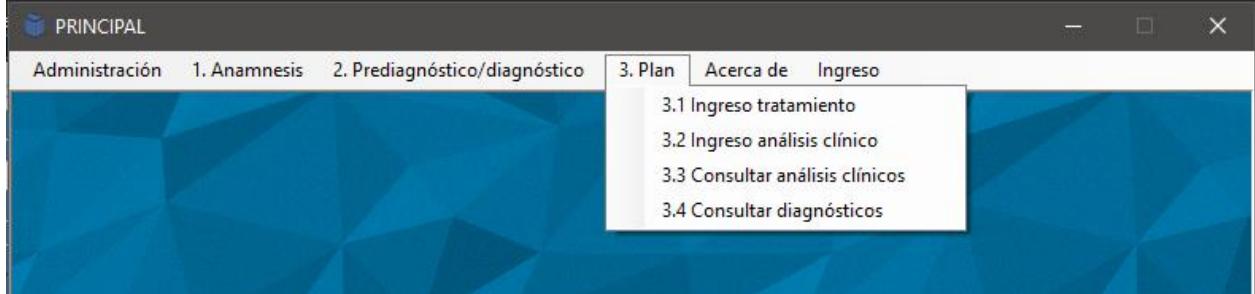


-Si el paciente existe pero no tuvo consultas con un médico
Para este caso el sistema desplegará una ventana emergente informando de que el paciente no tiene entrevistas realizadas con un médico y que para contestar un formulario es necesario previamente haber consultado con un médico.

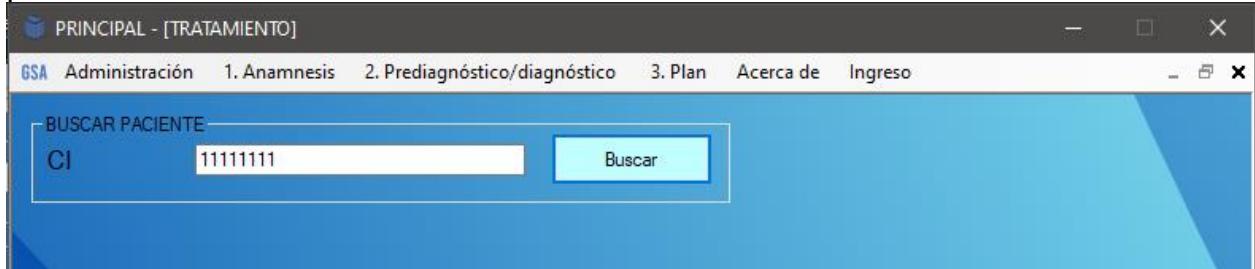


3. Plan - 3.1 Ingreso tratamiento

Para Ingresar un tratamiento deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “3. Plan” y luego ingresar a la opción de “3.1 Ingreso tratamiento”.

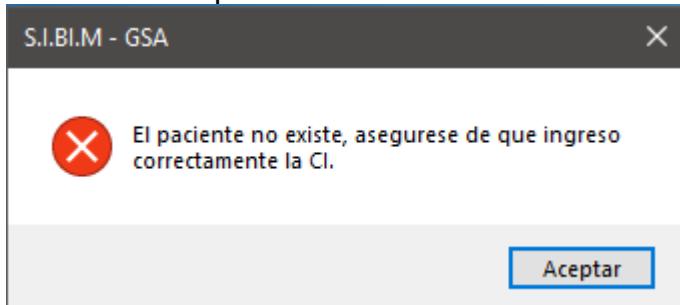


Una vez dentro de “3.1 Ingreso tratamiento” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”.



-Si el paciente no existe

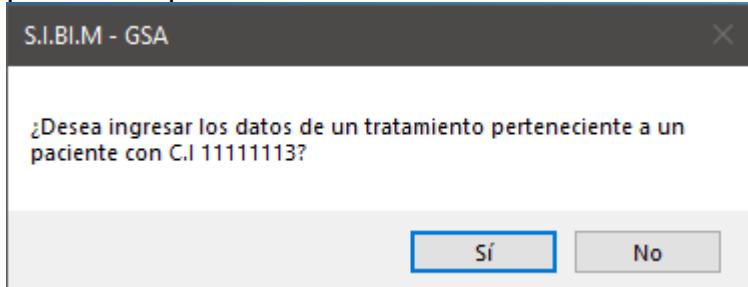
En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente



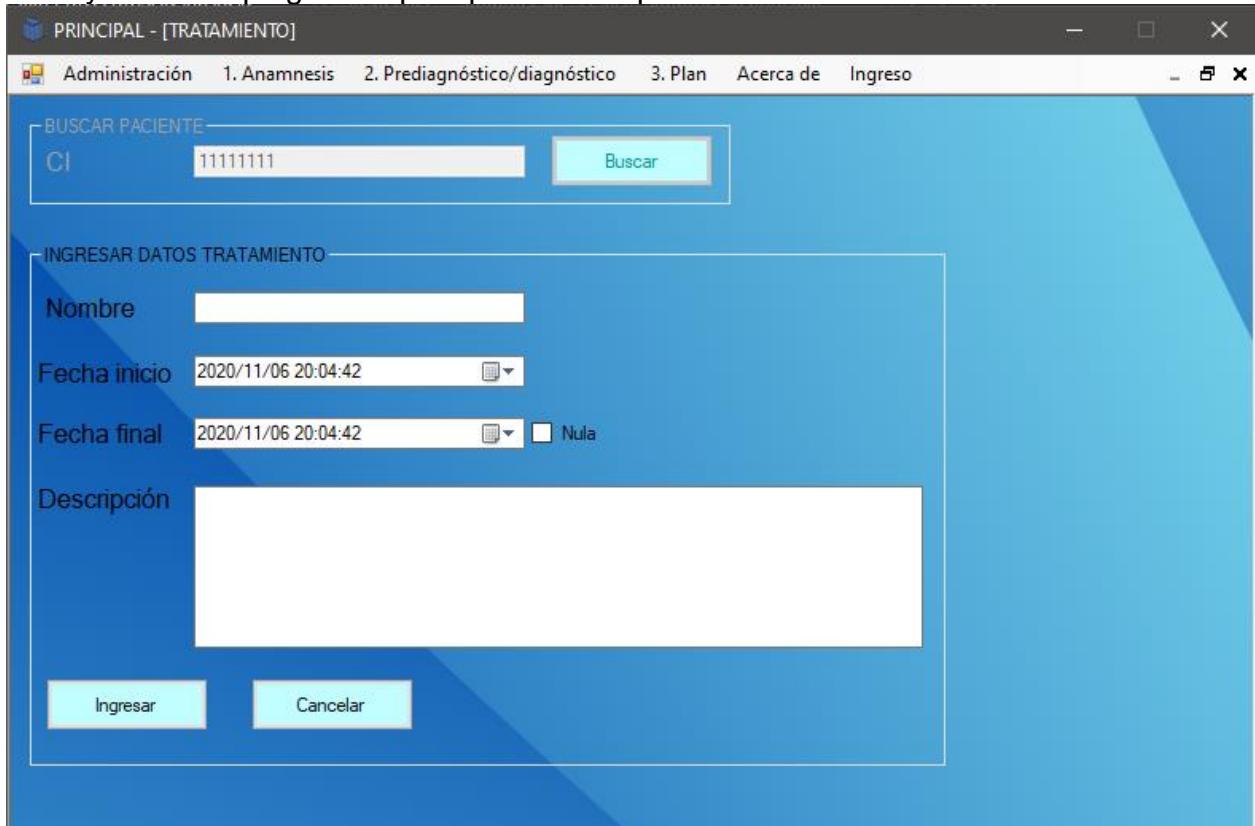


Si el paciente existe

En este caso el sistema procederá a mostrar una ventana emergente que preguntará si se desea ingresar los datos de un tratamiento perteneciente a un paciente con la cédula de identidad ingresada anteriormente y deberá de proceder a pulsar el botón “Sí”.



Una vez pulsado el botón “Sí” el sistema habilitará una agrupación de datos en donde se tendrá que ingresar los datos de un tratamiento en los cuadros de texto y listas desplegables que aparecen en pantalla.



Ya ingresados los datos solicitados en los cuadros de texto y listas desplegables, se procederá a pulsar el botón “Ingresar” para realizar el ingreso del tratamiento.



PRINCIPAL - [TRATAMIENTO]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR PACIENTE
CI: 11111111 Buscar

INGRESAR DATOS TRATAMIENTO

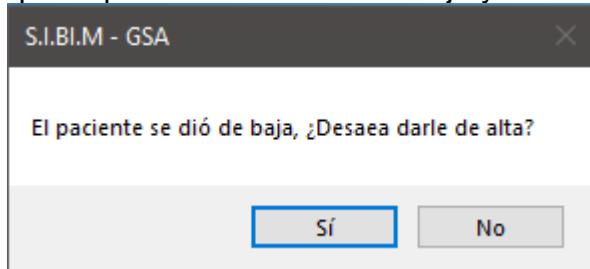
Nombre: Fiebre
Fecha inicio: 2020/11/06 20:04:42
Fecha final: 2021/01/10 20:04:42 Nula
Descripción: Perifar 500

Ingresar Cancelar

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó el tratamiento con éxito, de lo contrario el sistema notificará que hubo errores al ingresar el tratamiento.

-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.

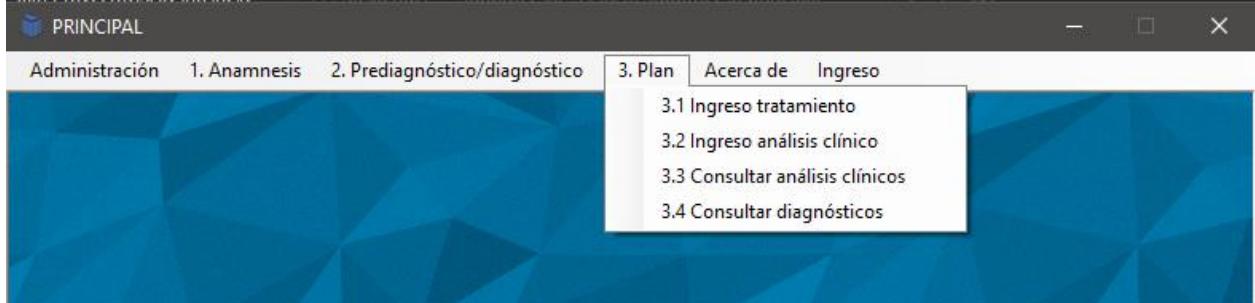


Al presionar el botón "Sí" el sistema notificará si el paciente se dio de alta exitosamente o no.

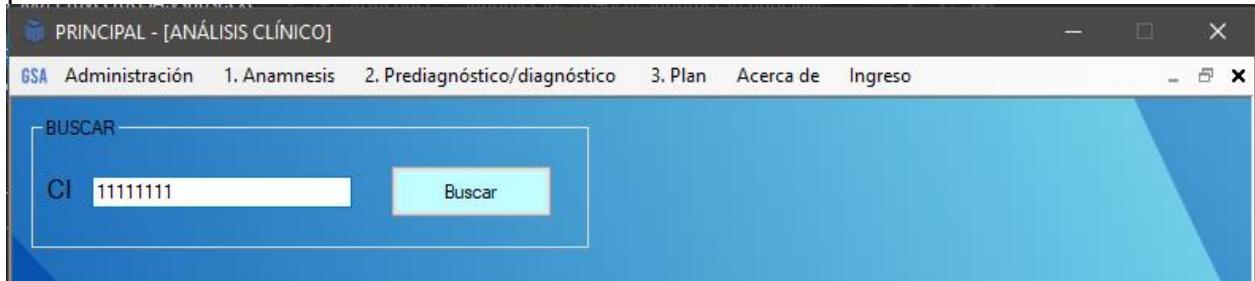


3. Plan - 3.2 Ingreso análisis clínico

Para Ingresar el análisis clínico deberá dirigirse al menú ubicado en la parte superior del programa, deberá entrar a la sección de “3. Plan” y luego ingresar a la opción de “3.2 Ingreso análisis clínico”.

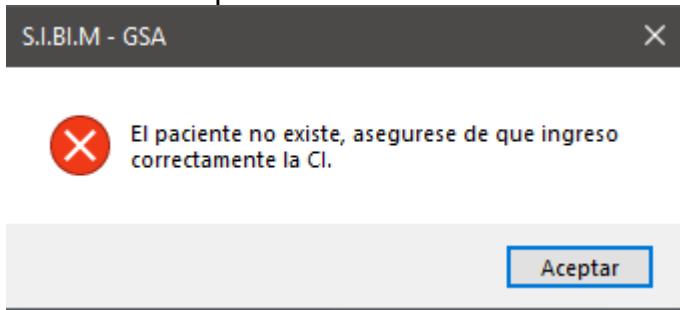


Una vez dentro de “3.2 Ingreso análisis clínico” será necesario previamente ingresar la cédula de identidad del paciente, sin puntos ni guiones y luego presionar el botón de “Buscar”



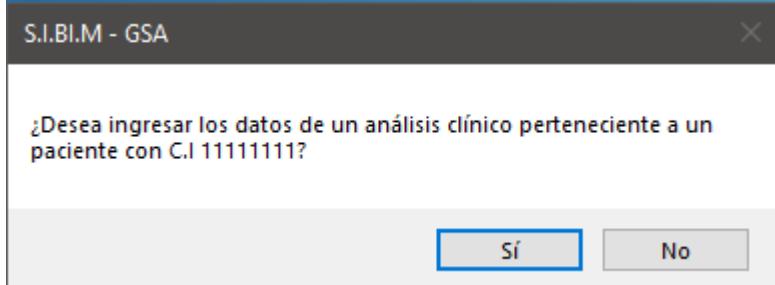
-Si el paciente no existe

En este caso el sistema procederá a mostrar una ventana emergente que notificará que la cédula de identidad que se ingresó previamente no está asociada a un paciente

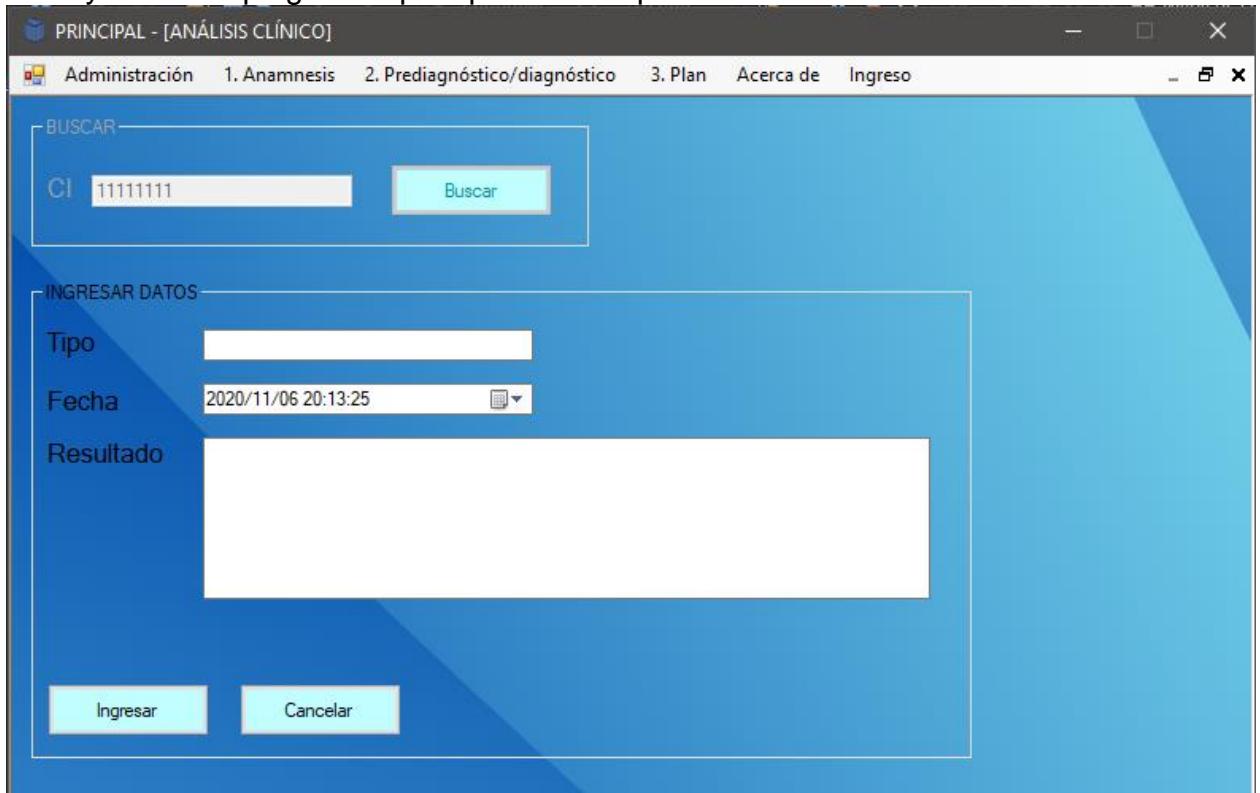


-Si el paciente existe

En este caso el sistema procederá a mostrar una ventana emergente que preguntará si se desea ingresar los datos de un análisis clínico perteneciente a un paciente con la cédula de identidad ingresada anteriormente y deberá de proceder a pulsar el botón “Sí”.



Una vez pulsado el botón “Sí” el sistema habilitará una agrupación de datos en donde se tendrá que ingresar los datos de un tratamiento en los cuadros de texto y listas desplegables que aparecen en pantalla.



Ya ingresados los datos solicitados en los cuadros de texto y listas desplegables, se procederá a pulsar el botón “Ingresar” para realizar el ingreso del análisis clínico.



PRINCIPAL - [ANÁLISIS CLÍNICO]

Administración 1. Anamnesis 2. Prediagnóstico/diagnóstico 3. Plan Acerca de Ingreso

BUSCAR

CI 11111111 Buscar

INGRESAR DATOS

Tipo Radiografía

Fecha 2020/11/06 20:13:25

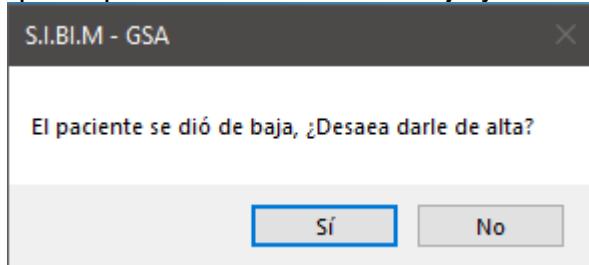
Resultado Hombro deslocado

Ingresar Cancelar

Si la operación se efectúa exitosamente, el sistema desplegará una ventana emergente notificando que se ingresó el análisis clínico con éxito, de lo contrario el sistema notificará que hubo errores al ingresar el análisis clínico.

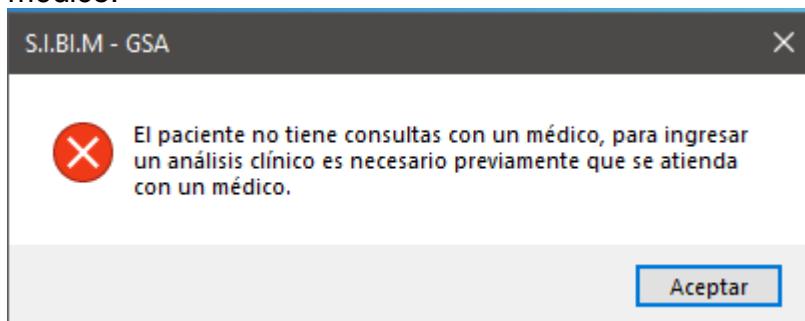
-Si el paciente existe pero fue dado de baja

Para este caso el sistema desplegará una ventana emergente informando de que el paciente fue dado de baja y si se desea realizar el alta del mismo.



Al presionar el botón "Sí" el sistema notificará si el paciente se dio de alta exitosamente o no.

-Si el paciente existe pero no tuvo consultas con un médico
Para este caso el sistema desplegará una ventana emergente informando de que el paciente no tiene entrevistas realizadas con un médico y que para ingresar un análisis clínico es necesario previamente haber consultado con un médico.



Anexo Formacion Empresarial

Anexo I: Formularios DGI 0351 y 0352

DGI DIRECCIÓN GENERAL IMPOSITIVA		BPS Instituto de Seguridad Social		DECLARACIÓN DE REGISTRO										0351					
				INSCRIPCIÓN Y ACTUALIZACIÓN EMPRESAS Y OTRAS ENTIDADES UNIPERSONALES O PLURIPERSONALES										VERSIÓN 04					
RUBRO 1 - IDENTIFICACIÓN																			
BPS	Nº de REGISTRO DE CONTRIBUYENTE				Nº de REGISTRO EMPRESA				Nº DE RUT				DG						
	C.I.		ATYR																
Acto que se realiza				BPS	DGI					Vigencia del acto									
Inicio de actividades				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>					Día	Mes	Año							
Reinicio de actividades										01	03	2050							
Modificación (completar sólo los campos que se actualizan)																			
RUBRO 2 - DATOS DE LA ENTIDAD																			
Nombre o denominación		GSA																	
Tipo de entidad																			
Persona física		Núcleo Familiar			Condominio			Sucesión Indivisa											
Sociedad de Hecho		Sociedad Colectiva			Sociedad de Respons.Limitada			<input checked="" type="checkbox"/> Soc. en comandita simple											
Soc. en comandita por Acc.		SA con Acciones Nominativas			SA con Acciones al Portador			Cooperativas											
Pers. de Derecho Público		Asociaciones			Entidad pluripersonal no resid.			Fundaciones											
Fideicomiso		Grupo de Interés Económico			Asoc/soc. Agrarias			Entidad Gremial											
Comisión Administradora		Repres. Diplomat. Extranjeras			Organismos internacionales			Otros											
Fecha de constitución		Día	Mes	Año	Inscripción en el Registro Nat. Com.				Número	Día	Mes	Año							
		01	03	2020						11	3	2020							
Nombre de fantasía																			
Residente		Si	No	No residente con Establecimiento permanente			Si	No	País de residencia										
RUBRO 3 - DOMICILIO FISCAL DEL LOCAL PRINCIPAL																			
Calidad en que ocupa el domicilio fiscal		Propietario	<input checked="" type="checkbox"/>	Arrendatario	Sub-arrendat.	Comodatario	Usufructuario	Otros											
Departamento	Localidad				Calle	Número	Apart.	Código postal											
Montevideo	Montevideo	Quijote				2885		11600											
Complemento del domicilio	Nº torre, Edif., Block				Calle interna		Manzana	Solar											
Ruta	Km.	Nombre del establecimiento			Nº de Padrón	Nº de Secc. Judicial	Nº de Secc. Policial	Paraje											
Otros detalles de ubicación: (E) Entre calle y calle)																			
Contactos	Persona o empresa			Teléfono fijo	Fax	Teléfono móvil	Correo electrónico												
	Gonzalo Gularde			24863382		094742318	gsasrlcorp@gmail.com												
RUBRO 4 - DOMICILIO CONSTITUIDO																			
Indicar si el Domicilio Constituido es igual al Fiscal <input checked="" type="checkbox"/>																			
Departamento	Localidad				Calle	Número	Apart.	Código postal											
Complemento del domicilio	Nº torre, Edif., Block				Calle interna		Manzana	Solar											
Ruta	Km.	Nombre del establecimiento			Nº de Padrón	Nº de Secc. Judicial	Nº de Secc. Policial	Paraje											
Otros detalles de ubicación: (E) Entre calle y calle)																			
Contactos	Persona o empresa			Teléfono fijo	Fax	Teléfono móvil	Correo electrónico												
RUBRO 5 - ACTIVIDAD ECONÓMICA																			
Actividad Principal	Desarrollo de software													Alta	Baja	Grupo	Sub G	Cap	Band
Actividad Secundaria	Diseño e instalación de Cableado Estructurado													<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				
Actividad Secundaria	Testing													<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				
RUBRO 6 - OBLIGACIONES																			
Alta	Baja	Obligación	Características	01	02	03	04	05	Alta	Baja	Obligación	Características	01	02	03	04	05		
<input checked="" type="checkbox"/>		IRAE	<input checked="" type="checkbox"/>						IMESI NUM. 1								IMESI NUM. 15		
		IVA - Construcc.							IMESI NUM. 2								IMESI NUM. 16		
		IVA - Serv. Pers.							IMESI NUM. 3								IMESI NUM. 17		
		IVA - AGROP.							IMESI NUM. 4								IMESI NUM. 18		
<input checked="" type="checkbox"/>		IVA - GRAL	<input checked="" type="checkbox"/>						IMESI NUM. 5								IMESI NUM. 19		
		IVA - Peq. Emp.							IMESI NUM. 6								MONOTRIBUTO		
<input checked="" type="checkbox"/>		PAT - Entidades	<input checked="" type="checkbox"/>						IMESI NUM. 7								IRPF I		
		PAT - P.F.							IMESI NUM. 8								IRPF II		
		ITP							IMESI NUM. 9								IRNR		
		ICOSA							IMESI NUM. 10										
		IMEBA							IMESI NUM. 11										
		IMEBA ADIC.							IMESI NUM. 12										
		FIS							IMESI NUM. 13										
		ENT.ASEGURA.							IMESI NUM. 14										
														Características					
														01 Contribuyente 02 Agente de retención 03 Agente de percepción 04 Resp. por oblig. tributarias de 3ros 05 Responsable sustituto					
Firma Representante/Autorizado							Aclaración de Firma												

S.I.BI.M

I.T.I-C.E.T.P

3° BB

790

		Nº de RUT										
RUBRO 7 - OTROS DATOS												
Condición												
Importador	Exportador			<input checked="" type="checkbox"/> Asimilado a Exportador		Imprenta Autorizada						
Usuario directo Z. Franca	Usuario indirecto Z. Franca			<input checked="" type="checkbox"/> Administradora de Crédito		Entidad Colaboradora						
Establecim. De Temporada	No Contrib./ Cont. Exonerado			<input checked="" type="checkbox"/> Ent.no resid. sin activ. empr.		Ent. resid. Atribuidora de rent.						
Imprenta Autorizada WEB	Comercio autorizado Tax free			<input checked="" type="checkbox"/> Registro INAC - CVA 90 días		Registro INAC - CVA 30 días						
Otros (detallar)												
Balance	Día 31	Mes 12	Día	Mes	Día	Mes						
Antecedentes												
En caso de inicio o reinicio por Transferencia, indicar la forma	Compra total		Compra parcial		Sucesión	Escisión	Fusión					
Otros (detallar)												
Nombre o denominación del antecesor					Nº de Registro de Empresa	Nº de R U T						
RUBRO 8 - RÉGIMEN DE APORTACIÓN AL B.P.S.												
Industria y comercio	<input checked="" type="checkbox"/>		Servicios Personales		Rural	Construcción	Civil					
RUBRO 9 - DATOS DEL TITULAR												
Persona Física (indicar)												
Alta <input checked="" type="checkbox"/> Permanencia <input checked="" type="checkbox"/>												
Type of Vinculos	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS
Dueño			Director		Socio	2	9	Cónyuge Colab.		Administrador		Síndico
Ingresar Código (*)	Socio sin Administración			Socio Administrador conjunto			Otro (detallar)			Socio Administrador indistinto		
Representante												
Tipo de documento								Nº de documento		País de origen del documento		
C.I.	<input checked="" type="checkbox"/> DNI		Pasaporte		54200707							
1º Apellido	2º Apellido		1º Nombre		2º Nombre							
Gularde	Artola		Gonzalo		Lisandro							
Fecha de Nacimiento	Estado Civil		Sexo		Firma							
02/12/2020	Soltero		Masculino									
Tipo de residencia								NO Residente	Residente	<input checked="" type="checkbox"/>	País de residencia	
Domicilio Particular												
Departamento	Localidad		Calle		Número		Adjunto	Apart.	Código postal			
Montevideo	<input checked="" type="checkbox"/> Montevideo		Quijote		2885							
Completo del domicilio	Nº torre, Edif., Block		Calle interna				Manzana	Solar				
Ruta	Km.	Nombre del establecimiento	Nº de Padrón	Nº de Secc. Judicial	Nº de Secc. Policial		Paraje					
Otros detalles de ubicación: (Ej. Entre calle y calle)												
Teléfono fijo	Fax		Teléfono móvil		Correo electrónico		Otro correo electrónico o contacto					
24863382			094742318		gonzalogular@gmail.com							
Persona Física (indicar)								Alta <input checked="" type="checkbox"/> Permanencia <input checked="" type="checkbox"/>				
Type of Vinculos	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS
Dueño			Director		Socio	02	9	Cónyuge Colab.		Administrador		Síndico
Ingresar Código (*)	Socio sin Administración			Socio Administrador conjunto			Otro (detallar)			Socio Administrador indistinto		
Representante												
Tipo de documento								Nº de documento		País de origen del documento		
C.I.	<input checked="" type="checkbox"/> DNI		Pasaporte		55762293		Uruguay					
1º Apellido	2º Apellido		1º Nombre		2º Nombre							
Iriart	Guibernet		Santiago		Miguel							
Fecha de Nacimiento	Estado Civil		Sexo		Firma							
22/10/2002	Soltero		Masculino									
Tipo de residencia								NO Residente	Residente	<input checked="" type="checkbox"/>	País de residencia	
Domicilio Particular												
Departamento	Localidad		Calle		Número		Adjunto	Apart.	Código postal			
Montevideo	<input checked="" type="checkbox"/> Montevideo		Suecia		1719				12800			
Completo del domicilio	Nº torre, Edif., Block		Calle interna				Manzana	Solar				
Ruta	Km.	Nombre del establecimiento	Nº de Padrón	Nº de Secc. Judicial	Nº de Secc. Policial		Paraje					
Otros detalles de ubicación: (Ej. Entre calle y calle)												
Teléfono fijo	Fax		Teléfono móvil		Correo electrónico		Otro correo electrónico o contacto					
23113887			093982489		santiagoirart9@gmail.com							
Observaciones												
Firma del Representante o Autorizado								TIMBRE PROFESIONAL				
Aclaración de firma												
Documento de Identidad												

El/Vos firmante/s declara/n que los datos establecidos en este formulario son correctos y completos y que conoce/n las sanciones aplicables en caso de falsa declaración, previsto en los Art. 96 del Código Tributario y 239 del Código Penal.

Impreso Secc. Publicaciones D.G.I.



DECLARACIÓN DE REGISTRO
ACTUALIZACIÓN / ANEXO DE INSCRIPCIÓN
PERSONAS FÍSICAS VINCULADAS
 EMPRESAS Y OTRAS ENTIDADES
 UNIPERSONALES O PLURIPERSONALES

0352
 VERSIÓN 03

DGI
 DIRECCIÓN GENERAL
 I M P O S I T I V A

RUBRO 1 - IDENTIFICACIÓN		
Nº de REGISTRO DE CONTRIBUYENTE		Nº de REGISTRO EMPRESA
BPS	C.I.	
	ATYR	DGI

RUBRO 10- DATOS DE PERSONAS FÍSICA VINCULADAS												
Persona Física Vinculada (Indicar)			Alta	<input checked="" type="checkbox"/> Permanencia								
Tipo de Vínculos Ingresar Código (*)	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS
	Dueño		Director		Socio	02 01	Cónyuge colab.		Administrador		Síndico	
	Socio sin Administración			Socio Administrador conjunto			Socio Administrador indistinto					
	Representante			Otro (detallar)								
Tipo de documento			Nº de documento			País de origen del documento						
C.I.	<input checked="" type="checkbox"/> DNI	Pasaporte	5.213.710-8			Uruguay						
1° Apellido		2° Apellido	1° Nombre			2° Nombre						
Acosta		Bernardo	Agustín			Luciano						
Fecha de Nacimiento	Estado Civil	Sexo	Firma									
18/04/2002	Soltero	Masculino										
Tipo de residencia	No Residente	Residente	<input checked="" type="checkbox"/>	País de residencia								
Domicilio Particular												
Departamento	Localidad		Calle			Número	Adjunto	Apart.	Código postal			
Montevideo	Montevideo		Ramón y Cajal			2467	1		11600			
Complemento del domicilio			Nº torre, Edif., block	Calle interna			Manzana	Solar				
Otros detalles de ubicación: (Ej.: Entre calle y calle; Ruta, Km. Nombre establecimiento, No. de Padrón, Sec. Jud., Sec. Pol., Paraje)												
Teléfono fijo	Fax	Teléfono móvil	Correo electrónico			Otro correo electrónico o contacto						
24808806		091027326	agustin14acosta@gmail.com									

En los casos de Persona Física RESIDENTE sin actividad empresarial, el domicilio que se declara corresponde al domicilio fiscal a todos los efectos tributarios.												
Persona Física Vinculada (Indicar)			Alta	<input checked="" type="checkbox"/> Permanencia								
Tipo de Vínculos Ingresar Código (*)	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS
	Dueño		Director		Socio	02 01	Cónyuge colab.		Administrador		Síndico	
	Socio sin Administración			Socio Administrador conjunto			Socio Administrador indistinto					
	Representante			Otro (detallar)								
Tipo de documento			Nº de documento			País de origen del documento						
C.I.	<input checked="" type="checkbox"/> DNI	Pasaporte										
1° Apellido		2° Apellido	1° Nombre			2° Nombre						
Sunes		Pianavilla	Bruno			Daniel						
Fecha de Nacimiento	Estado Civil	Sexo	Firma									
14/10/2002	Soltero	Masculino										
Tipo de residencia	No Residente	Residente	<input checked="" type="checkbox"/>	País de residencia								
Domicilio Particular												
Departamento	Localidad		Calle			Número	Adjunto	Apart.	Código postal			
Montevideo	Montevideo		Tampico			945	B		11900			
Complemento del domicilio			Nº torre, Edif., block	Calle interna			Manzana	Solar				
Otros complementos o detalles de ubicación: (Ej.: Entre calle y calle; Ruta, Km. Nombre establecimiento, No. de Padrón, Sec. Jud., Sec. Pol., Paraje)												
Teléfono fijo	Fax	Teléfono móvil	Correo electrónico			Otro correo electrónico o contacto						
23049665		092111308	brunosunes@gmail.com									

En los casos de Persona Física RESIDENTE sin actividad empresarial, el domicilio que se declara corresponde al domicilio fiscal a todos los efectos tributarios.												
Persona Física Vinculada (Indicar)			Alta	Permanencia								
Tipo de Vínculos Ingresar Código (*)	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS	VF	SS
	Dueño		Director		Socio		Cónyuge colab.		Administrador		Síndico	
	Socio sin Administración			Socio Administrador conjunto			Socio Administrador indistinto					
	Representante			Otro (detallar)								
Tipo de documento			Nº de documento			País de origen del documento						
C.I.	DNI	Pasaporte										
1° Apellido		2° Apellido	1° Nombre			2° Nombre						
Fecha de Nacimiento	Estado Civil	Sexo	Firma									

Firma del Representante o Autorizado	Aclaración de Firma
--------------------------------------	---------------------

Anexo II: Contrato de forma jurídica

SOCIEDAD DE RESPONSABILIDAD LIMITADA. En la ciudad de Montevideo, el 01 de marzo del 2020, entre: Por una parte: Agustín Luciano Acosta Bernardo, domiciliado en Ramón y Cajal 2467 apto 01, titular de la CI 5.213.710-8, Santiago Miguel Iriart Guibernau, domiciliado en calle Suecia 1719, titular de la CI 5.576.229-3, Bruno Daniel Sunes Pianavilla, domiciliado en calle Tampico 945 B, titular de la CI 5.258.229-1, Gonzalo Lisandro Gularte Artola, domiciliado en calle Quijote 2885, titular de la CI 5.420.070-7, todos uruguayos, mayores de edad, y solteros, siendo sus domicilios de esta ciudad **CONVIENEN** en celebrar un contrato de sociedad de responsabilidad limitada según las disposiciones de la Ley N° 16.060 y las estipulaciones que pactan y transcriben a continuación: **PRIMERO:** Los comparecientes constituyen una sociedad de responsabilidad limitada que se denominará "**GSA SRL**". **SEGUNDO:** El domicilio será en la ciudad de Montevideo, pudiendo establecer sucursales en cualquier ciudad del país. **TERCERO:** El objeto de la sociedad será la comercialización de todo tipo de servicios informáticos en general, según las disposiciones legales pertinentes, pudiendo incluso exportar, importar y representar a otras personas físicas o jurídicas, sean nacionales o extranjeras. **CUARTO:** La sociedad tendrá una duración de 30 años, pero después del segundo año, cualquiera de los socios podrá disolver la sociedad, comunicándolo por escrito a los administradores con una anticipación de 90 días previos a la terminación del ejercicio corriente. **QUINTO:** El capital de la sociedad será de pesos uruguayos cien mil (100.000), el que se dividirá en cien cuotas de pesos uruguayos mil (1.000) cada una, correspondiéndole a cada socio un número de veinticinco cuotas. **SEXTO:** En cumplimiento de sus aportes, los socios integran su capital, entregando la suma de pesos veinticinco mil (25.000) cada uno a la sociedad, suma por la cual ésta otorga a aquellos formal carta de pago. **SÉPTIMO:** La administración de la sociedad será ejercida indistintamente por cualquiera de los socios, sin perjuicio de ser necesaria la actuación conjunta de los socios para el caso de enajenación o gravamen de bienes inmuebles propiedad de la sociedad. **OCTAVO:** El ejercicio económico de la sociedad que se constituye, abarcará el lapso del 1º de enero al 31 de diciembre. Al final de cada ejercicio se hará un balance general para determinar el estado de los negocios sociales, el que deberá finalizar en un plazo de sesenta días a partir del cierre. **NOVENO:** De las ganancias de cada ejercicio se retirará un cinco por ciento (10%) para la formación de un fondo de reserva. El resto de las ganancias se distribuirá entre los socios en forma proporcional a sus aportes. **DÉCIMO:** La cesión de las cuotas sociales requerirá en todo caso el consentimiento unánime de los socios. **DÉCIMO PRIMERO:** En el caso de fallecimiento o incapacidad de cualquiera de los socios, la sociedad no se disolverá y continuará con los herederos o el cónyuge del socio fallecido, o el representante del incapaz. **DÉCIMO SEGUNDO:** Los comparecientes fijan domicilios especiales a todos los efectos de este contrato en los arriba indicados como suyos. **DÉCIMO TERCECRO:** Los comparecientes solicitan la certificación notarial de su firma a la Escribana Agustina Daniela Iriart Artola. **CERTIFICO QUE:** Las firmas que anteceden fueron puestas en mi presencia por las personas hábiles y de mi conocimiento, Agustín Luciano Acosta Bernardo, Santiago Miguel Iriart Guibernau, Bruno Daniel Sunes Pianavilla y Gonzalo Lisandro Gularte Artola, cuyos demás datos individualizantes surgen del documento precedente. **EN FE DE ELLO**, a solicitud de parte interesada y a efectos de su presentación en el Registro Nacional de Comercio y demás oficinas públicas y/o privadas que fuere menester, expido el presente que sello, signo y firmo en la ciudad de Montevideo, el 01 de marzo de dos mil veinte.

Firma de Socio 1: 

Firma de Socio 2: 

Firma de Socio 3: 

Firma de Socio 4: 

Firma de Escribana: 



Anexo Sistemas Operativos

Anexo I: Datos necesarios para usar el servidor

Usuario: gsaltI

Contraseña: gsa3bb



Bibliografía

Licencias Ciclo de vida Windows Microsoft

<https://support.microsoft.com/es-es/help/13853/windows-lifecycle-fact-sheet>

Windows 10 2019 Enterprise LTSC

<https://docs.microsoft.com/en-us/windows/whats-new/ltsc/>

Novedades Windows 10 2019 Enterprise LTSC

<https://docs.microsoft.com/en-us/windows/whats-new/ltsc/whats-new-windows-10-2019>

Diferencias entre licencias OEM, Retail y VLK

https://www.youtube.com/watch?v=8juoYw_IoMA

Licencias VLK Microsoft <https://www.youtube.com/watch?v=ZusmvqMRGHg>

CentOS 8 <https://www.centos.org/>

Ciclo de vida Windows Microsoft

<https://support.microsoft.com/es-es/help/13853/windows-lifecycle-fact-sheet>

Windows 10 2019 Enterprise LTSC

<https://docs.microsoft.com/en-us/windows/whats-new/ltsc/>

Novedades Windows 10 2019 Enterprise LTSC

<https://docs.microsoft.com/en-us/windows/whats-new/ltsc/whats-new-windows-10-2019>

Diferencias entre licencias OEM, Retail y VLK

https://www.youtube.com/watch?v=8juoYw_IoMA

Licencias VLK Microsoft

<https://www.youtube.com/watch?v=ZusmvqMRGHg>

Contactar para consultar precio licenciamiento

<https://www.microsoft.com/es-uy/solution-providers/home>

Información CentOS 8

<https://blog.desdelinux.net/se-ha-presentado-la-liberacion-de-centos-8-y-centos-stream-edition/>

Instalación CentOS 8

<https://josehuaman.com/centos-8-instalacion/>

Requisitos MySQL

<https://dev.mysql.com/doc/mysql-monitor/8.0/en/system-prereqs-reference.html>

Instalación MySQL en CentOS 8

<https://www.tecmint.com/install-mysql-on-centos-8/>

<https://www.solvetic.com/tutoriales/article/7867-como-instalar-o-desinstalar->



[mysql-en-centos-8/ https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-centos-8](https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-centos-8) <https://kifarunix.com/install-mysql-8-on-centos-8/>

Información SELinux

<https://soyadmin.com/que-es-selinux/>

Sistemas Operativos Modernos Tanenbaum

Material de clases dictadas por John Liencres

Roger S. Pressman “Ingeniería del Software - Un enfoque práctico” Séptima edición (1988)

Apuntes de clases dictadas por el docente Néstor Piazza Documentos varios de información

<https://nestor-piazza.webnode.com.uy/proyecto/>

<https://nestor-piazza.webnode.com.uy/analisis-y-dise%c3%b1o-de-aplicaciones/>

Apuntes de clase dictadas por el docente Renzo Bazzano

Apuntes de clase dictadas por el docente Fabian Martinez

Apuntes de clase dictadas por la docente Alejandra Meny

Apuntes de clase dictadas por el docente Fabian Saule

Apuntes de clase dictadas por el docente Gino Di Pascua

Apuntes de clase dictadas por la docente Rosa Nieves

Sistema de Bases de Datos “Elmasri Navathe”

Informe CUTI 2016