**Desarrollo de un Software Web para el Control de Asistencia Estudiantil en COTECNOVA durante el periodo 2021**

**Natalia Agudelo Valdés**

**Juan David Hoyos Ramírez**

**Corporación de Estudios Tecnológicos del Norte del Valle**

**Unidad de Ingeniería, Arquitectura, Urbanismo y Afines**

**Tecnología en Sistemas de Información**

**Cartago**

**2021**

**Desarrollo de un Software Web para el Control de Asistencia Estudiantil en COTECNOVA durante el periodo 2021**

**Natalia Agudelo Valdés**

**Juan David Hoyos Ramírez**

**Proyecto integrador para optar por el título de Tecnólogo en Sistemas de Información**

**Director**

**IS Arvey Barahona Gómez**

**Corporación de Estudios Tecnológicos del Norte del Valle**

**Unidad de Ingeniería, Arquitectura, Urbanismo y Afines**

**Tecnología en Sistemas de Información**

**Cartago**

**2021**

# Resumen

Esta investigación, cuyo objetivo es desarrollar un software web para la asistencia estudiantil y envío de material en la Institución COTECNOVA en Cartago y objetivos específicos: Identificar los Requerimientos, Diseñar el sistema y Codificar el mismo, se presenta como un proyecto factible que buscar solucionar y explicar los inconvenientes que se tienen al tomar la asistencia de forma tradicional y su organización, como en sí, la forma en la que se envían los materiales vistos, este problema se abordó en una investigación de campo de tipo descriptiva, donde se refiere o se narra características y propiedades de la situación propuesta, la muestra del proyecto fueron los estudiantes de séptimo semestre de la jornada diurna en el programa de tecnología en sistemas. Esto dio a conocer como los docentes y estudiantes actúan en su entorno natural de clases y, en particular, en la situación actual que presenta el país por los diferentes confinamientos sufridos; así mismo, presenta las fallas o retardos en los procedimientos tomados y/o llevados a cabo debido a varios factores asociados, sea al tamaño del grupo o diferentes técnicas que aplica el docente para el llamado a lista.

El modelo en modelo en cascada permitió, inicialmente a través de la observación, y encuestas, identificar los requerimientos funcionales y no funcionales, para eso se utilizó la herramienta REM; con la herramienta Visual Paradigm se diseñan los diagramas de casos de uso, componentes, actividades y secuencia para modelar el sistema de datos propuesto; El código del software fue realizado en php y html, utilizando Boostrap y css se logró una vista más amigable al usuario.

**Palabras claves**: Asistencia a clase, Envío de material, Registrar clase, Consultar horario.

# Abstrac

This research, whose objective is to develop a web software for student assistance and sending material at the COTECNOVA Institution in Cartago and specific objectives: Identify the Requirements, Design the system and Code it, is presented as a feasible project to seek to solve and explain the inconveniences that are had when taking the assistance in a traditional way and its organization , as in itself, the way in which the materials seen are sent, this problem was addressed in a descriptive field research, where it refers or narrates characteristics and properties of the proposed situation, the sample of the project were the students of seventh semester of the daytime in the program of technology in systems. This made known how teachers and students act in their natural environment of classes and, in particular, in the current situation that the country presents due to the different confinements suffered; likewise, it presents the failures or delays in the procedures taken and / or carried out due to several associated factors, either the size of the group or different techniques that the teacher applies for the call to list.

The cascading model allowed, initially through observation, and surveys, to identify the functional and non-functional requirements, for that the REM tool was used; with the Visual paradigm tool, the diagrams of use cases, components, activities and sequence are designed to model the proposed data system; The software code was made in php and html, using Boostrap and css, a more user-friendly view.

**Keywords:** Class attendance, Sent material, Register class, Check schedule .**Tabla de contenidos**

Pág.

[Resumen 3](#_Toc75005417)

[Abstrac 4](#_Toc75005418)

[Introducción 16](#_Toc75005419)

[Justificación 18](#_Toc75005420)

[Definición del problema 20](#_Toc75005421)

[Objetivos 27](#_Toc75005422)

[General 27](#_Toc75005423)

[Específicos 27](#_Toc75005424)

[Marco Referencial 28](#_Toc75005425)

[Antecedentes 28](#_Toc75005426)

[Marco Teórico 30](#_Toc75005427)

[Especificaciones de requerimientos 30](#_Toc75005428)

[IEEE 830 30](#_Toc75005429)

[Requisitos Funcionales 30](#_Toc75005430)

[Requisitos No Funcionales 31](#_Toc75005431)

[REM 31](#_Toc75005432)

[Desarrollo de un sistema de base de datos 33](#_Toc75005433)

[Modelo entidad relación 35](#_Toc75005434)

[Metodología de desarrollo 38](#_Toc75005435)

[Marco Conceptual 43](#_Toc75005436)

[Mysql Workbench 43](#_Toc75005437)

[PhpMyAdmin 44](#_Toc75005438)

[Visual Paradigm 44](#_Toc75005439)

[HTML 45](#_Toc75005440)

[JavaScript 46](#_Toc75005441)

[Marco Contextual 47](#_Toc75005442)

[Diseño metodológico 49](#_Toc75005443)

[Enfoque 50](#_Toc75005444)

[Métodos y Tipo de Investigación 50](#_Toc75005445)

[Técnicas de Recolección de Datos 51](#_Toc75005446)

[Instrumento Observación 51](#_Toc75005447)

[Instrumento Entrevista 52](#_Toc75005448)

[Población y Muestra 53](#_Toc75005449)

[Esquema Temático 54](#_Toc75005450)

[Capítulo 1: Identificación de requerimientos 54](#_Toc75005451)

[Observación 54](#_Toc75005452)

[Entrevista 55](#_Toc75005453)

[REM 57](#_Toc75005454)

[Participantes en el proyecto 57](#_Toc75005455)

[Descripción del sistema actual 58](#_Toc75005456)

[Objetivos del proyecto 59](#_Toc75005457)

[Catálogo de requisitos del sistema 61](#_Toc75005458)

[Requisitos de información 61](#_Toc75005459)

[Requisitos Funcionales 71](#_Toc75005460)

[Diagramas de caso de uso 73](#_Toc75005461)

[Definición de actores 74](#_Toc75005462)

[Casos de uso del sistema 75](#_Toc75005463)

[Requisitos no funcionales 96](#_Toc75005464)

[Capítulo 2: Diseño del sistema de información 98](#_Toc75005465)

[Diagrama de casos de uso 98](#_Toc75005466)

[Diagrama de componentes 101](#_Toc75005467)

[Diagrama del Administrador 102](#_Toc75005468)

[Diagrama del Docente 103](#_Toc75005469)

[Diagrama del Estudiante 103](#_Toc75005470)

[Diagrama de actividades 104](#_Toc75005471)

[Diagrama del Administrador 105](#_Toc75005472)

[Diagrama del Docente 106](#_Toc75005473)

[Diagrama del Estudiante 107](#_Toc75005474)

[Diagramas de secuencia 109](#_Toc75005475)

[Diagramas del Administrador 110](#_Toc75005476)

[Diagramas del Estudiante 120](#_Toc75005477)

[Diagramas del Docente 123](#_Toc75005478)

[Diagramas Generales 127](#_Toc75005479)

[Diseño de la base de datos 128](#_Toc75005480)

[Modelo entidad-relación 129](#_Toc75005481)

[Modelo Físico 130](#_Toc75005482)

[Implementación a phpmyadmin 131](#_Toc75005483)

[Plantilla Original del Bootstrap 136](#_Toc75005484)

[Capítulo 3: Codificación del software 137](#_Toc75005485)

[Base de datos Control de Asistencia 138](#_Toc75005486)

[Codificación del software 155](#_Toc75005487)

[Aclaración de las gestiones realizadas en los módulos del administrador. 166](#_Toc75005488)

[Conclusiones 178](#_Toc75005489)

[Recomendaciones 180](#_Toc75005490)

[Cronograma 181](#_Toc75005491)

[Bibliografía 182](#_Toc75005492)

[Anexos 187](#_Toc75005493)

**Lista de Tablas**

Pág.

Tabla 1 Organizaciones del proyecto 57

Tabla 2 Participantes del proyecto 58

Tabla 3 Objetivos del proyecto 59

Tabla 4 Requisito de estudiante 61

Tabla 5 Requisito de horario 62

Tabla 6 Requisito de tipo de usuario 63

Tabla 7 Requisito de clase 64

Tabla 8 Requisito de administrador 65

Tabla 9 Requisito de aula 66

Tabla 10 Requisito de materia 67

Tabla 11 Requisito de docente 68

Tabla 12 Requisito carrera 69

Tabla 13 Requisito material adjunto 70

Tabla 14 Requisitos funcionales 71

Tabla 15 Definición de actores 74

Tabla 16 Caso de uso 1 75

Tabla 17 Caso de uso 2 76

Tabla 18 Caso de uso 3 77

Tabla 19 Caso de uso 4 78

Tabla 20 Caso de uso 5 79

Tabla 21 Caso de uso 6 80

Tabla 22 Caso de uso 7 81

Tabla 23 Caso de uso 8 82

Tabla 24 Caso de uso 9 83

Tabla 25 Caso de uso 10 84

Tabla 26 Caso de uso 11 85

Tabla 27 Caso de uso 12 86

Tabla 28 Caso de uso 13 87

Tabla 29 Caso de uso 14 88

Tabla 30 Caso de uso 15 89

Tabla 31 Caso de uso 16 90

Tabla 32 Caso de uso 17 91

Tabla 33 Caso de uso 18 92

Tabla 34 Caso de uso 19 93

Tabla 35 Caso de uso 20 94

Tabla 36 Caso de uso 21 95

Tabla 37 Requisitos no funcionales 96

Tabla 38 Descripción del funcionamiento de los casos de uso 99

Tabla 39 Descripción del funcionamiento de los diagramas de componentes 104

Tabla 40 Descripción del funcionamiento de los diagramas de actividades 108

Tabla 41 Descripción del funcionamiento de los diagramas de secuencia del administrador 117

Tabla 42 Descripción del funcionamiento de los diagramas de secuencia del estudiante 121

Tabla 43 Descripción del funcionamiento de los diagramas de secuencia del docente 125

Tabla 44 Descripción del funcionamiento de los diagramas de secuencia globales 128

Tabla 45 Código Fuente de la Base de datos 138

**Lista de ilustraciones**

Pág.

Ilustración 1 Estructura del documento de requisitos del sistema 32

Ilustración 2 Ciclo de vida de una Base de Datos 34

Ilustración 3 Notaciones 36

Ilustración 4 Relaciones 37

Ilustración 5 Modelo en cascada 38

Ilustración 6 Formato de observación 52

Ilustración 7 Diagrama general de caso de uso del sistema actual 57

Ilustración 8 Diagrama de casos de uso 73

Ilustración 9 Diagrama de casos de uso 98

Ilustración 10 Diagrama de componentes administrador 102

Ilustración 11 Diagrama de componentes del docente 103

Ilustración 12 Diagrama de componentes del estudiante 103

Ilustración 13 Diagrama de actividades del administrador 105

Ilustración 14 Diagrama de actividades del docente 106

Ilustración 15 Diagrama de actividades del estudiante 107

Ilustración 16 Gestionar docente 110

Ilustración 17 Gestionar estudiantes 111

Ilustración 18 Gestionar horarios 112

Ilustración 19 Gestionar aulas 113

Ilustración 20 Gestionar materias 114

Ilustración 21 Gestionar carreras 115

Ilustración 22 Modificar datos 116

Ilustración 23 Consultar horario y datos del estudiante y datos de las asignaturas 120

Ilustración 24 Validar asistencia a clase 121

Ilustración 25 Clases Vistas 123

Ilustración 26 Consultar clase y datos del docente y de las asignaturas 124

Ilustración 27 Registrar Clase 125

Ilustración 28 Iniciar Sesión 127

Ilustración 29 Crear o Recuperar contraseña 127

Ilustración 30 Modelo entidad-relación 129

Ilustración 31 Modelo Físico 130

Ilustración 32 MySQL Workbench 131

Ilustración 33 Forward Engineer 132

Ilustración 34 Forward Engineer 133

Ilustración 35 Mensaje 134

Ilustración 36 phpmyadmin 135

Ilustración 37 phpmyadmin 136

Ilustración 38 Plantilla Original 137

Ilustración 39 Conexión a la base de datos en forma local 147

Ilustración 40 Etiquetas HTML del login 148

Ilustración 41 Vista grafica desde el navegador 149

Ilustración 42 Validación del login.php 150

Ilustración 43 Index del estudiante 151

Ilustración 44 Consulta del horario del estudiante 152

Ilustración 45 Validar Asistencia Estudiante 153

Ilustración 46 Validar Asistencia Estudiante 153

Ilustración 47 Index Administrador 154

Ilustración 48 Gestionar Docente 155

Ilustración 49 Editar estudiante 156

Ilustración 50 Modificar Estudiante 157

Ilustración 51 Eliminar Estudiante 158

Ilustración 52 Registro nuevo docente 159

Ilustración 53 Modificación Docente 159

Ilustración 54 Modificación del docente 160

Ilustración 55 Eliminar Docente 160

Ilustración 56 Registro del aula 161

Ilustración 57 Modificación Aula 161

Ilustración 58 Modificación Aula 162

Ilustración 59 Registro Materia 162

Ilustración 60 Modificación Materia 163

Ilustración 61 Modificación Materia 163

Ilustración 62 Eliminar Materia 164

Ilustración 63 Registro Horario 165

Ilustración 64 Modificar Horario 166

Ilustración 65 Modificar Horario 167

Ilustración 66 Eliminar Horario 168

Ilustración 67 Registro de Carreras 168

Ilustración 68 Actualizar Carrera 169

Ilustración 69 Actualizar Carrera 169

Ilustración 70 Eliminar Carrera 170

Ilustración 71 Index docente 170

Ilustración 72 Envió de Material 171

Ilustración 73 Consultar Clase 172

Ilustración 74 Cronograma de actividades 176

**Lista de Anexos**

Pág.

[Anexo A Formato de observación diligenciado 182](#_Toc75036353)

[Anexo B Formato de Entrevista 184](#_Toc75036354)

[Anexo C Comunicado de falta de asistencia 187](#_Toc75036355)

# Introducción

La presente investigación se refiere al tema de la toma de asistencia estudiantil y envío de material, la cual siempre ha sido uno de los mayores problemas en las diferentes instituciones llegando a ser incontrolable. Actualmente se han inventado varios mecanismos de control para esta mediante dispositivos tecnológicos, lo cual ha hecho un gran auge a las renovaciones de como se venía tomando anteriormente.

Una característica principal de esta toma es como los docentes poseen muchos métodos e influencias en el manejo de sus clases, así mismo, cómo reaccionan ante el tamaño de su grupo.

El documento presenta en su estructura inicialmente: el resumen, el abstract e introducción, que hacen referencia a los datos básicos para presentar el documento a nivel general; la justificación presenta el porqué y el para qué en beneficio del proyecto; la definición del problema es en donde se detalla de una manera amplia el problema y se identifica o plantea la pregunta que da origen a la investigación, seguido de esto, los objetivos presentan: el general, el cual dará solución a la pregunta de investigación y los específicos, los cuales apoyan y son los que se ejecutarán en otro apartado. En el marco referencial se encuentran los diferentes antecedentes que sirvieron de apoyo para la elaboración de este proyecto; el marco teórico muestra en detalle las bases por las cuales el proyecto busca dar sus pasos, entre ellos se da un detalle de: la especificación de requerimientos, el desarrollo de un sistema de bases de datos, los modelos de entidad relación y la metodología de desarrollo; el marco conceptual está compuesto por algunos conceptos que se presentan en el documento; el marco contextual ubica el campo en donde se desarrollará el proyecto. El diseño metodológico presenta el tipo de enfoque se utilizó junto al método y tipo de investigación usada, la recolección de datos, la población y muestra determinada para llevar a cabo el proyecto.

Finalmente, el esquema temático se desarrolla en una serie de capítulos en los cuales se busca dar paso a paso como el proyecto de investigación da solución a la problemática planteada inicialmente.

# Justificación

Como se ha podido observar en el planteamiento del problema, la toma de asistencia estudiantil siempre ha sido algo que ha estado presente en la vida universitaria y qué, en sus diversos factores incide negativamente en el desarrollo o manejo de las respectivas clases y/o asignaturas.

En tiempos actuales no solamente la población universitaria se ha visto afectada por esto, también se ha podido evidenciar que algunos estudiantes no obtienen el material visto en clase por distintas circunstancias, validándose así, de sus mismos apuntes o conocimientos previos adquiridos.

El desarrollo de este proyecto, y más precisamente, del software, busca mayor eficiencia y facilidad en la toma de asistencia a clases de los estudiantes, esta toma asistida mediante el aplicativo web, donde los docentes y estudiantes, inicialmente, deberán loguearse; el docente tendrá tres opciones las cuales constarán de: Consultar clases, donde allí podrá ver su respectivo horario con las asignaturas que dictará a lo largo de la semana. Registrar clases, donde podrá, como lo dice su nombre, registrar sus clases. En este apartado podrá escoger los grupos disponibles que se le hayan sido asignado y la respectiva asignatura que dicta para que, posteriormente, inicie el proceso de asistencia de sus estudiantes a la clase, también tendrá la opción de crear un nuevo código para que sus estudiantes lo ingresen o permanecer con el de por defecto, allí mismo podrá dejar en un apartado links a su disposición (material de trabajo), por último, tendrá la opción de ver sus Clases vistas, las cuales ha ido dando a lo largo de la semana con los estudiantes que registraron su asistencia.

Por otro lado, los estudiantes tendrán dos apartado, el primero donde podrá consultar su horario con sus respectivas matrículas asignadas en la semana y, el segundo, donde podrá pasar a registrar su asistencia a la clase, en este apartado deberá seleccionar la asignatura deseada e ingresar el código obtenido por el docente inicialmente para así completar el respectivo proceso.

De esta forma, el resultado general del proyecto permitirá mostrar alternativas de solución y agilización acorde con las necesidades de la institución y de su comunidad académica, permitiendo ser una guía para futuras investigaciones que puedan presentar el mismo problema.

Junto a esto, los usuarios (docentes y estudiantes) que posea la institución tendrán sus respectivos beneficios, los estudiantes podrán consultar su respectivo horario de clase junto con sus asistencias constatadas de una forma más calmada, de tener a la mano el material visto, mientras que los docentes no tendrán que preocuparse por el llamado del mismo, ejerciendo un mejor control sobre su clase y sobre sus estudiantes, evitando el tener que ir a algún tercero para hacer su respectivo proceso de registro.

# Definición del problema

La tecnología, a lo largo del tiempo, ha evolucionado exponencialmente llegando a ser de gran ayuda para el ser humano en muchos aspectos; entre estos puede destacarse los controles de acceso a las diferentes estancias, ya sea de trabajo, de instituciones o simplemente, de aulas de clase, donde se ha logrado notar como, durante estos procesos, el uso del tiempo es realmente notable.

El control del tiempo

(…) es el proceso de registrar cuánto tiempo ha pasado una persona realizando una actividad. En el pasado, esto se hacía usando partes de horas o fichas. Los empleados fichaban manualmente al entrar y salir del trabajo, o cumplimentaban un formulario en papel que detallaba a qué dedicaban su tiempo.

Hoy en día, el control del tiempo se puede completar digitalmente mediante programas informáticos. En algunos casos, los empleados aún anotan manualmente la cantidad de horas que han pasado trabajando en cada tarea o proyecto. Esta información se registra en el software en lugar de una ficha de papel. En otros casos, se utiliza una aplicación de temporizador para controlar automáticamente el tiempo empleado en una tarea (Everitt, 2020).

El adquirir herramientas para administrar el tiempo es atacar el problema de raíz. De alguna manera, todos tenemos que organizar nuestras tareas cotidianas, pero la actitud frente al tiempo del que se dispone para la ejecución de dichas tareas suele ser variable y compleja. El modo de determinar el empleo del tiempo tiene que ver con actos, hábitos, actitudes, con el tipo de cultura donde se vive e incluso con las nuevas tecnologías que agilizan la información. Según la Revista de la Universidad del Salle (2016), las personas necesitan con urgencia dar espacio para atender su vida personal, familiar, laboral y académica de manera equilibrada, dando el nivel de importancia que cada actividad merece.

De igual forma, en los espacios académicos, los diferentes docentes y estudiantes deben pensar en crear diferentes métodos para tener un mejor control de su tiempo, ya que en su efecto contrario podrían provocarse errores. El llevarlo de forma positiva conduce a ser mucho más productivos y eficientes; tener una mejor concentración y un mayor enfoque, generando así, una gran eficacia para lograr los diferentes objetivos propuestos en el menor tiempo posible.

Siempre se cuestiona

(…) lo que pasa dentro de las salas de clases, puede ser un misterio para la administración en muchas instituciones de Educación Superior, pareciera que al cerrarse la puerta solo el profesor y los alumnos saben cómo se pasó la materia, o si efectivamente asistió la cantidad de gente que se lee en una dudosa lista escrita a mano.

La gestión de las salas de clases se ve como una tarea rudimentaria y de poco valor para el desarrollo de una carrera profesional, pero es fundamental para toda entidad educativa. Múltiples estudios demuestran que la asistencia está estrictamente relacionada con la retención del estudiante y, por consiguiente, el éxito estudiantil.

La asistencia a clases influye en variables como la recepción correcta de los contenidos y, por tanto, en el éxito académico general de una institución. Así lo demuestra un artículo de Minnesota State University, que recopila varios estudios sobre el tema.

En el primero de los estudios que presentan en el artículo se midió el impacto de los compromisos de tiempo de los estudiantes en las distintas actividades que conlleva el aprendizaje, entre ellas la clase. El estudio comprobó que lejos del compromiso de tiempo más valioso e importante en un curso es la clase presencial, donde el estudiante además de asistir estaba cognitivamente participando de la clase. Por esto, además de la asistencia a clases, es de gran importancia que los contenidos sean pasados de una forma que cause interés (Uplanner, 2019).

En el Ministerio de Educación Nacional (2009), se destaca la sección de las fundamentaciones y orientaciones para la implementación del decreto 1290 de 2009, ubicándonos en el artículo número 6, este trata de que cada establecimiento educativo determinará los criterios de promoción escolar de acuerdo con el sistema institucional de evaluación de los estudiantes. Así mismo, el establecimiento educativo definirá el porcentaje de asistencia que incida en la promoción del estudiante. Cuando un establecimiento educativo determine que un estudiante no puede ser promovido, debe garantizarle en todos los casos, el cupo para que continúe con su proceso formativo.

En Colombia, el uso de las tecnologías en cuanto la toma de asistencia aún es muy básico, ya que en las instituciones se sigue usando el registro de asistencia de forma tradicional, formatos impresos, oral o lista de chequeo, cabe destacar que algunas ya han optado en la modernización de este proceso mediante los dispositivos tecnológicos e informáticos, como es el caso de la (Universidad de Antioquia, s.f), donde proponen la Tarjeta Integrada Personal (TIP) como un medio de identificación y control que permitirá el acceso a los diferentes servicios y beneficios de la comunidad académica (estudiantes, profesores y egresados), así como a los empleados, jubilados y contratistas.

Los usos de estas tecnologías han sido de gran impacto actualmente, este se muestra, principalmente, en cómo la dirección de las clases; el apogeo virtual ha llegado a todo el ámbito educativo y ha ido expandiéndose cada vez más. En ciertas ocasiones, el problema de llamar a los estudiantes no es de menos, las instituciones han tenido que adaptarse a todas estas nuevas situaciones de diferentes modos.

En la Corporación de Estudios Tecnológicos del Norte del Valle, COTECNOVA, el docente lleva un control de asistencia estudiantil de forma tradicional (llamados de lista en forma oral, lista de chequeos o simplemente se hace entrega de una hoja en la cual los estudiantes escriben sus nombres); el uso de este mecanismo puede ser poco eficaz a la hora de realizar el registro estudiantil a grupos numerosos, ya que gasta demasiado tiempo en el llamado, igualmente, el manejo de listados en papel puede resultar tedioso e incluso contaminante sino se hace un buen uso del mismo (reciclaje de papeles en los cuales puede usarse más de una vez); terminada la clase, el docente debe dejar un reporte del tema dictado y de los estudiantes que no asistieron a la misma en la oficina de Atención al Usuario o en Registro y Control.

Del mismo modo, la Institución ha tomado nuevas estrategias en cuanto a cómo presentar las clases y materiales a sus estudiantes. La modalidad de alternancia (presencial y mediada por tecnología). Ha surgido como un nuevo camino para adaptar a la comunidad académica en la situación actual; siendo así un tema de interés para el docente el cómo debería constatar la asistencia de sus estudiantes o adjuntar archivos vía online; se han presentado algunos métodos donde el docente pide a los estudiantes escribir su nombre en el chat mientras van quedando grabados para, posteriormente finalizada la clase o en sus horarios personales, revisar dicha grabación y verificar quien se presentó, o bien, realizar un llamado tradicional (oral); en ambos casos se presentan retardos en el inicio o un reprocesamiento de información

Por otro lado, debe tenerse en cuenta a los estudiantes que aún no se encuentran matriculados pero que, se les da el plazo de una semana para resolver su situación (procesos financieros, aceptación de crédito, entre otros), estos estudiantes no tienen forma de estar registrados en el sistema o demostrar que sí tuvieron las clases, exceptuando el hecho de que, por parte de los maestros, los ven y apuntan en algún papel o son tenidos en cuenta luego.

Según el Artículo 37 del Reglamento Estudiantil de COTECNOVA (1971), en donde se menciona que el estudiante que deje de asistir al 10% o más de las clases dictadas en una asignatura, perderá ésta por inasistencia. El docente deberá invalidar las notas obtenidas y en el registro de calificaciones reportar cero punto cero (0.0 perdida por inasistencia), y deberá de notificarle al director de unidad respectivo y a los alumnos afectados por la inasistencia.

La deserción dentro de los diferentes entornos (virtual, presencial) también puede afectar al sistema, esta es provocada debido a que el estudiantado tiene dificultades no solamente desde los medios, sino también en la adaptación psicológica a una nueva forma de enseñar, cabe mencionar que también se puede producir por fallas técnicas ya que no todos los usuarios (estudiantes, docentes) cuentan con los recursos disponibles, como por ejemplo, poseer una buena conexión a internet o dispositivos inteligentes (Smartphone, computador). Según el Revista Semana (2020), la deserción escolar es un gran problema ya que en el año 2020 la cifra de deserción había llegado a 100.000 estudiantes.

La institución requiere de un sistema que pueda ayudar a los docentes a agilizar el tiempo de cátedra en sus clases, la cual se dirigiría de una forma eficaz permitiendo visualizar sus listados de una forma más intuitiva a tiempo; como a los estudiantes el presentarse a las mismas y recibir su respectivo material, permitiendo consultar sus horarios y clases, estos procesos se verían reflejados en la disminución de costos y recursos. De lo contrario, de seguir la situación actual presentada, seguirán mostrándose decadencias en las respectivas clases ocasionando posibles deserciones o pérdidas de interés por parte de la comunidad académica.

Por lo anterior, surge la pregunta:

¿Cómo identificar la asistencia estudiantil a las clases en COTECNOVA de tal manera que se utilice el menor tiempo posible, sin importar el tamaño del grupo y que el envío de material visto puedan obtenerlos todos los alumnos en el periodo 2021?

# Objetivos

## General

* Desarrollar un Software Web que permita la administración de la asistencia estudiantil y envío de material en la Corporación de Estudios Tecnológicos del Norte del Valle.

## Específicos

* Identificar los requerimientos del sistema para el control de asistencia estudiantil y envío de material.
* Diseñar el sistema de información de asistencia académica de COTECNOVA.
* Codificar el Software Web según las especificaciones.

# Marco Referencial

## Antecedentes

Con la finalidad de brindar soporte a la Escuela Fiscal Rafael Aguilar, un control y reporte de las labores diarias de los docentes que laboran en la institución (Sandoval Illescas & Sigüenza Cárdenas, 2011) en Cuenca – Ecuador, presentan un método de desarrollo de software basado en especificación de requerimientos, análisis, planificación, diseño y documentación que están constituidos por un conjunto e procesos que van desde la identificación de necesidades del usuario hasta la entrega del producto final. Después que recolectaron y amplificaron los respectivos requisitos de la institución, queda en claro que los planteles educativos están en constante crecimiento y evolución, por lo que es necesario avanzar tecnológicamente para brindar un mejor servicio tanto a alumnos como a padres de familia, por ello surge, la necesidad de adquirir un sistema de control de asistencia para automatizar el proceso que se realizaba manualmente.

El aporte de esta investigación sirve para la realización del proyecto en cuanto nuevas perspectivas que puede presentar el docente en el manejo del sistema y/o funcionalidades implementadas, dando así, una vista mucho más amplia para el objeto de estudio propio.

Así mismo, en el municipio de Verapaz – El salvador, se elaboró sistema informático para el instituto nacional “San José Verapaz” del municipio de Verapaz, departamento San Vicente (Alberto & Delmy, 2008), donde dicho sistema surge por los diversos problemas por las cuales pasa el instituto, específicamente dentro del área de control de los expedientes de los alumnos ya sean que estén activos o si son egresados, cuyo sistema fue creado para la optimización de procesamiento de datos referente a los diferentes expedientes que se necesiten dentro del instituto, estos están basados por medios de documentos y entrevistas que fueron facilitados por la administración del instituto nacional “San José Verapaz” del departamento de San Vicente.

La utilidad de dicha información dentro del proyecto planteado es una solución que puede servir de apoyo ya que explica cómo se logró la optimización del procesamiento de datos referente a los documentos que dicho instituto necesitaba.

Por otro lado, la compañía Samsung ha impulsado un proyecto junto a la Universidad de Salamarca para controlar la asistencia de sus alumnos por medio del dispositivo móvil de los estudiantes y otro dispositivo NFC localizado dentro del aula de clase, el estudiante deberá tener la aplicación descargada en su respectivo móvil ya que este estará enlazada con la base de datos la cual tendrá la información del estudiante y un registro de las clases a las cuales ha asistido, lo que se busca con la idea planteada es que el estudiante “pase su smartphone cerca al dispositivo localizado dentro del aula, seguido esto recibirá una confirmación en su dispositivo móvil sobre su registro en la clase” (Hernández, 2012), en cuanto al docente, este también tendrá acceso a la aplicación y podrá ver la información almacenada en la base de datos la cual tendrá el  registro de los alumnos registrados.

Gracias a los conocimientos aportados por el proyecto que se lanzó entre Samsung y la Universidad Salamarca, se encontró que la alternativa propuesta en sus celulares con un respectivo software y/o sistema operativo eliminó muchas actividades manuales y facilitó el rastreo de sus estudiantes.

## Marco Teórico

### Especificaciones de requerimientos

IEEE 830. El estándar IEEE 830-1998 para el SRS(en inglés) o ERS (Especificación de requerimientos de software) según @admin (2010), “Es un conjunto de recomendaciones para la especificación de los requerimiento o requisitos de software el cual tiene como producto final la documentación de los acuerdos entre el cliente y el grupo de desarrollo para así cumplir con la totalidad de exigencias estipuladas”.

La Especificación de Requisitos de Software (ERS) es una de las fases más importante del proceso de desarrollo de software y es en esta fase donde se obtiene la descripción completa del comportamiento del futuro software o producto que se va a desarrollar, (Grupo Carricay, 2018).

Se destacan los requisitos funcionales y no funcionales, como lo plantea @Requeridos Blog (2018):

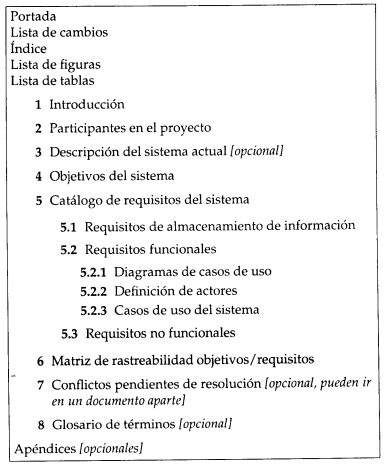
Requisitos Funcionales. Los requisitos funcionales son declaraciones de los servicios que prestará el sistema, en la forma en que reaccionará a determinados insumos. Cuando hablamos de las entradas, no necesariamente hablamos sólo de las entradas de los usuarios. Pueden ser interacciones con otros sistemas, respuestas automáticas, procesos predefinidos. En algunos casos, los requisitos funcionales de los sistemas también establecen explícitamente lo que el sistema no debe hacer. Es importante recordar esto: un RF puede ser también una declaración negativa. Siempre y cuando el resultado de su comportamiento sea una respuesta funcional al usuario o a otro sistema, es correcto. Y más aún, no sólo es correcto, sino que es necesario definirlo. Y eso nos lleva al siguiente punto.

Requisitos No Funcionales. Se trata de requisitos que no se refieren directamente a las funciones específicas suministradas por el sistema (características de usuario), sino a las propiedades del sistema: rendimiento, seguridad, disponibilidad. En palabras más sencillas, no hablan de “lo que” hace el sistema, sino de “cómo” lo hace. Alternativamente, definen restricciones del sistema tales como la capacidad de los dispositivos de entrada/salida y la representación de los datos utilizados en la interfaz del sistema.

Los requisitos no funcionales se originan en la necesidad del usuario, debido a restricciones presupuestarias, políticas organizacionales, la necesidad de interoperabilidad con otros sistemas de software o hardware, o factores externos tales como regulaciones de seguridad, políticas de privacidad, entre otros.

REM. REM (Requirements Management) es una herramienta experimental gratuita de Gestión de Requisitos diseñada para soportar la fase de Ingeniería de Requisitos de un proyecto de desarrollo software presentada por (Durán Toro, 2000), en su tesis se menciona el punto de “El documento de requisitos de sistemas” en el cual se expone la estructura que, posteriormente se seguirá en este proyecto de investigación.

Ilustración   
Estructura del documento de requisitos del sistema



Fuente: Depósito de investigación - Universidad de Sevilla, Un entorno metodológico de ingeniería de requisitos para sistemas de información

Se realizarán desde el primer punto hasta el sexto punto según lo requerido.

### Desarrollo de un sistema de base de datos

Una base de datos es

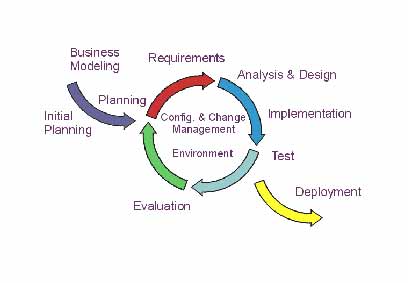
(…) una herramienta para recopilar y organizar datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En las bases de datos, se puede almacenar información sobre personas, productos, pedidos, o cualquier otra cosa.

Existen programas denominados sistemas gestores de bases de datos, abreviado DBMS, que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos DBMS, así como su utilización y administración, se estudian dentro del ámbito de la informática, (Delgado Licona, s.f).

Las bases de datos tienen un sistema de desarrollo el cual se divide en varios procesos, como lo plantea (Delgado Licona, s.f):

* Análisis.
* Diseño del modelo entidad / relación.
* Lenguaje SQL y base de datos final.

Ilustración   
Ciclo de vida de una Base de Datos



Fuente: Universidad Autónoma del Estado de Hidalgo, plataforma virtual

* Análisis: En este proceso se identifican todos los requisitos a fondo de lo que se quiere representar en la base de datos.
* Diseño del Modelo Entidad / Relación: Cuando se empieza a hacer el diseño de la base de datos un problema común es que el encargado del diseño de la misma lo haga directamente en la máquina. Esto hace que en un alto grado se pierda la conceptualidad del problema quedando tal diseño faltante de detalles.
* Lenguaje sql y base de datos final: Este es el último paso, consiste en codificar en lenguaje SQL el modelo relacional hecho anteriormente.

### Modelo entidad relación

El Modelo Entidad - Relación, presentado, creado por Peter P. Chen, menciona que este se centra más en la estructura lógica y abstracta de la información, siendo capaz de representar toda la semántica del mundo real por medio de entidades y relaciones.

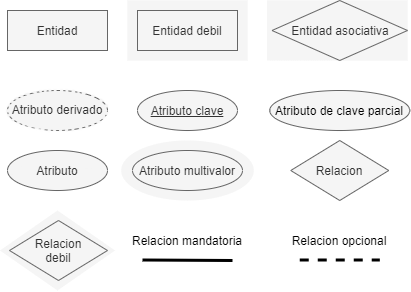
Para (Chen, 1977), el modelo de red,

(…) proporciona una visión más natural de los datos al separar las entidades y las relaciones (hasta cierto punto), pero su capacidad para lograr la independencia de los datos ha sido un desafío. El modelo relacional se basa en la teoría relacional y puede lograr un alto grado de independencia de los datos, pero puede perder información semántica importante sobre el mundo real. El modelo de conjunto de entidades, que se basa en la teoría de conjuntos, también logra un alto grado de independencia de datos, pero su visualización de valores como "3" o "rojo" puede no ser natural para algunas personas.

El modelo de entidad-relación adopta la visión más natural de que el mundo real consiste en entidades y relaciones. Incorpora parte de la información semántica importante sobre el mundo real.

La notación propuesta por Chen son las siguientes:

Ilustración 3   
Notaciones

  
*Fuente: Propia con ayuda de la herramienta draw.io*

**Entidad:** La entidad es aquello que se define como un objeto, persona o concepto. Esta entidad puede definirse por sus atributos.

**Entidad débil:** La entidad débil es aquella que no puede valerse por sí misma.

**Entidad asociativa:** Una entidad asociativa es aquella que relaciona entidades dentro de un conjunto.

**Atributo:** Es una característica de una entidad.

**Atributo Clave:** Es la característica principal de la entidad.

**Atributo de clave principal:** Es la que está relacionada con otra tabla a una llave principal.

**Atributo derivado:** Estos nacen a partir de otro atributo.

**Atributo multivalor:** Son aquello que tienen más de un valor.

**Relación:** Es aquella que da la interacción entre las entidades.

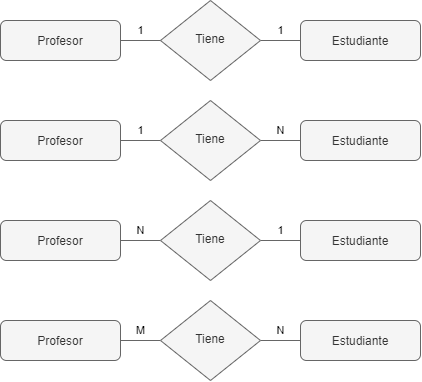
**Relación débil:** Como su mismo nombre lo indica, es débil.

**Relación mandatoria:** Es aquella que conecta las entidades con las relaciones.

**Relación opcional:** No es muy fuerte y, como su nombre lo indica, es opcional.

En cuanto a las relaciones en los diagramas se tiene qué:

Ilustración   
Relaciones

  
*Fuente: Propia con ayuda de la herramienta draw.io*

**1:N** De uno a muchos

**N:1** De muchos a uno

**M:N** De muchos a muchos

### Metodología de desarrollo

Royce propone un modelo compuesto por 7 fases que se ha de ejecutar en diversas vueltas (iteraciones):

Ilustración 5   
Modelo en cascada

  
*Fuente: Managing The Development Of Large Software Systems*

En español con el nombre de:

* Requisitos del sistema
* Requisitos de software
* Análisis
* Diseño
* Codificación
* Prueba
* Servicio u operaciones

En ocasiones las fases 1, 2 y 3 definidas por Royce se integran en una a modo de análisis, (Winston, 1970) propone 5 pasos para el desarrollo de modelo en cascada los cuales consisten en:

**Paso 1: El diseño del programa es lo primero**

El primer paso hacia una solución se realiza teniendo una fase de diseño preliminar del programa insertado entre la fase de generación de requisitos de software y la fase de análisis. Este procedimiento puede ser criticado sobre la base de que el diseñador del programa se ve obligado a diseñar en el vacío relativo del software inicial requisitos sin ningún análisis existente. Como resultado, su diseño preliminar puede estar sustancialmente erróneo en comparación a si tuviera que esperar hasta que se completara el análisis. Mediante esta técnica, el diseñador del programa asegura que el software no fallará debido al almacenamiento, razones de tiempo y flujo de datos. A medida que avanza el análisis en la fase siguiente, el diseñador del programa debe imponer al analista las limitaciones de almacenamiento, tiempo y funcionamiento de tal manera que perciba consecuencias. Si los recursos totales a aplicar son insuficientes o si el embrión de diseño operativo es incorrecto, se reconocerá en esta etapa anterior y la iteración con los requisitos y el diseño preliminar se podrán rehacer antes de que comience el diseño final, la codificación y la prueba.

¿Cómo se implementa este procedimiento? Se requieren los siguientes pasos:

1) Comenzar el proceso de diseño con diseñadores de programas, nada de analistas ni programadores.

2) Diseñar, definir y asignar los modos de procesamiento de datos, incluso a riesgo de equivocarse. Asignar procesamiento, funciones, diseñar la base de datos, definir el procesamiento de la base de datos, asignar tiempo de ejecución, definir interfaces y modos de procesamiento con el sistema operativo, describir el procesamiento de entrada y salida, y definir procedimientos operativos preliminares.

3) Escribir un documento de descripción general que sea comprensible, informativo y actual. Todos y cada uno de los trabajadores deben tener un conocimiento elemental del sistema. Al menos una persona debe tener un profundo conocimiento de la ingeniería del sistema que proviene en parte de haber tenido que escribir un documento de resumen.

**Paso 2: Documentar el diseño**

La primera regla para administrar el desarrollo de software es la documentación de los requisitos.

Para ello Royce propone que

1) Cada diseñador debe comunicarse con los diseñadores de interfaz, con su gerencia y posiblemente con el cliente. Un registro verbal es demasiado intangible para proporcionar una base adecuada para una interfaz o gestión.

2) Durante la fase inicial del desarrollo de software, la documentación es la especificación y la diseño. Hasta que comience la codificación, estos tres sustantivos (documentación, especificación, diseño) denotan algo. Si la documentación es mala el diseño es malo. Si la documentación aún no existe, aún no hay diseño, solo personas que piensan y hablan sobre el diseño, que tiene cierto valor, pero no mucho.

3) El valor monetario real de una buena documentación comienza en el proceso de desarrollo, durante la fase de prueba y continúa a través de las operaciones y el rediseño. El valor de la documentación puede ser descrita en términos de tres situaciones concretas y tangibles a las que se enfrenta todo director de programa.

**Paso 3: Hazlo dos veces**

Después de la documentación, el segundo criterio más importante para el éxito gira en torno a si el producto es totalmente original. Si el programa informático en cuestión se está desarrollando por primera vez, el orden importa por lo que la versión finalmente entregada al cliente para la implementación operativa sea en realidad la segunda versión en lo que se refiere a áreas críticas de diseño / operaciones.

Se debe tener una sensación intuitiva para el análisis, la codificación y diseño de programa. Debe detectarse rápidamente los puntos problemáticos en el diseño, modelarlos, modelar sus alternativas, olvídese de los aspectos sencillos del diseño que no vale la pena estudiar en este punto inicial, y finalmente llegar a un programa libre de errores. En cualquier caso, el punto de todo esto, como con una simulación, es que las cuestiones del tiempo, el almacenamiento, etc., que de otro modo serían cuestiones de juicio, ahora se pueden estudiar con precisión.

**Paso 4: planificar, controlar y supervisar las pruebas**

Sin lugar a dudas, el mayor usuario de los recursos del proyecto, ya sea mano de obra, tiempo de computadora o juicio de la gerencia, es la fase de prueba. Es la fase de mayor riesgo en términos de dólares y cronograma. Eso ocurre en el último punto de la programación cuando las alternativas de copia de seguridad están menos disponibles, si es que las hay.

Las tres recomendaciones anteriores para diseñar el programa antes de comenzar el análisis y la codificación, para documentarlo por completo, y para construir un modelo piloto, todos tienen como objetivo descubrir y resolver problemas antes de entrar a la fase de prueba.

**Paso 5: Involucrar al cliente**

Por alguna razón, lo que va a hacer un diseño de software está sujeto a una amplia interpretación incluso después de acuerdo previo. Es importante involucrar al cliente de manera formal para que se haya comprometido él mismo en puntos anteriores antes de la entrega final. Dar rienda suelta al contratista entre requisitos la definición y el funcionamiento invita a los problemas.

## Marco Conceptual

Mysql Workbench según (@Darkcrizt, s.f), es “un entorno gráfico para trabajar con bases de datos y servidores MySQL”, este cubre cinco temas principales los cuales son:

* Desarrollo SQL**:** Permite crear y administrar conexiones a servidores de bases de datos. Además de permitirle al usuario el poder configurar los parámetros de conexión, MySQL Workbench brinda la capacidad de ejecutar consultas SQL en las conexiones de la base de datos utilizando el Editor de SQL incorporado.
* Modelado de datos (diseño): Permite crear modelos del esquema de su base de datos de forma gráfica, aplicar ingeniería inversa y directa entre un esquema y una base de datos activa, y editar todos los aspectos de su base de datos utilizando el completo Editor de tablas. El editor de tablas proporciona facilidades de fácil uso para editar tablas, columnas, índices, disparadores, particiones, opciones, inserciones y privilegios, rutinas y vistas.
* Administración del servidor: Permite administrar instancias del servidor MySQL al administrar usuarios, realizar copias de seguridad y recuperación, inspeccionar datos de auditoría, ver el estado de la base de datos y monitorear el rendimiento del servidor MySQL.
* Migración de datos: Permite migrar de Microsoft SQL Server, Microsoft Access, Sybase ASE, SQLite, SQL Anywhere, PostreSQL y otras tablas, objetos y datos RDBMS a MySQL. La migración también admite la migración de versiones anteriores de MySQL a las últimas versiones.
* Soporte de MySQL Enterprise: Soporte para productos Enterprise como MySQL Enterprise Backup, MySQL Firewall y MySQL Audit.

PhpMyAdmin según (HN Datacenter en Chile, s.f), es un software de código abierto, diseñado para manejar la administración y gestión de bases de datos MySQL a través de una interfaz gráfica de usuario. PhpMyAdmin tiene la mayor lista de características; soporta todas las operaciones de uso común tales como la navegación, creación, modificación de las bases de datos MySQL, las tablas, campos e índices.

Visual ParadigmSegún (EcuRed, s.f), es una herramienta CASE: Ingeniería de Software Asistida por Computación. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación.

Dentro de esta herramienta se usarán los diagramas:

* Diagramas de caso de uso: En estos diagramas se exponen todos los requerimientos que deberá tener el sistema, así mismo, qué actores se identifican dentro del sistema y el rol que efectúa en el mismo.
* Diagramas de clase: Estos diagramas exponen la estructura del sistema, como está formado, que atributos tiene, que métodos tiene y la relación entre los mismos.
* Diagramas de componentes: En estos diagramas se representa las relaciones del sistema mediante una vista de diseño estática.
* Diagramas de actividades: Como su mismo nombre lo indica, los diagramas de actividades representan el paso a paso que deben hacer los actores del sistema cuando están ejecutando un caso de uso.
* Diagramas de secuencia: En estos diagramas se evidencia por partes, ya sea front-end (vista principal), backend (parte lógica funcional de una página) y la BD (base datos) interactúan entre sí y como los actores, al hacer un caso de uso, manejan el sistema.

HTML (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language), según (MDN-contributors, 2021 B), “es el componente más básico de la Web. Define el significado y la estructura del contenido web. Además de HTML, generalmente se utilizan otras tecnologías para describir la apariencia/presentación de una página web (CSS) o la funcionalidad/comportamiento (JavaScript)”.

**CSS** Hojas de Estilo en Cascada (del inglés Cascading Style Sheets) o CSS, como plantea (MDN-contributors, 2021 A), “es el lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML (en-US) (incluyendo varios languages basados en XML como SVG, MathML o XHTML). CSS describe como debe ser renderizado el elemento estructurado en la pantalla, en papel, en el habla o en otros medios”.

JavaScriptes un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones de primera clase, basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa (por ejemplo programación funcional), (MDN-contributors, 2021 C).

## Marco Contextual

**Cotecnova**

La (Corporación de Estudios Tecnológicos del Norte del Valle) fue fundada el 21 de septiembre de 1971, su carácter Académico es tecnológico, facultada para ofrecer programas hasta el nivel profesional por ciclos propedéuticos, esta nació gracias al empeño de insignes representantes cívicos cartagüeños, entre quienes es meritorio destacar al desaparecido Don Jesús Ernesto Aulestia, al Doctor Alfonso Delgado Arango y el excelentísimo Monseñor José Gabriel Calderón; unidos bajo el ideal común de ampliar las posibilidades de desarrollo social, técnico, moral y cultural de la comunidad estudiantil de la ciudad como centro intelectual.

La Universidad se encuentra ubicada en Cartago Valle del Cauca en la calle 10 con 3 # 3-95, esta cuenta con tres bloques diferentes dentro de ella; el bloque A que está compuesto por dos pisos, el primer piso con seis laboratorios, un consultorio médico, dos auditorios, oficinas y la cafetería, el segundo piso con 16 aulas. El bloque B en el primer piso lo conforma cuatro laboratorios, un auditorio y la biblioteca, en el segundo piso, se encuentran siete aulas de clase y el auditorio de teatro. El bloque C lo compone el auditorio general, espacio para cultivos, los baños, y un laboratorio.

En el año 2021, COTECNOVA imparte 1 [carrera universitaria](https://carrerasuniversitarias.com.co/universidades/corporacion-de-estudios-tecnologicos-del-norte-del-valle/carreras-universitarias), 11 [carreras cortas](https://carrerasuniversitarias.com.co/universidades/corporacion-de-estudios-tecnologicos-del-norte-del-valle/carreras-cortas) y una 1 [licenciatura](https://carrerasuniversitarias.com.co/universidades/corporacion-de-estudios-tecnologicos-del-norte-del-valle/licenciaturas), es además, una universidad virtual. Actualmente imparte 6 [carreras virtuales](https://carrerasuniversitarias.com.co/universidades/corporacion-de-estudios-tecnologicos-del-norte-del-valle/carreras-virtuales), entre ellas 1 [licenciatura virtual](https://carrerasuniversitarias.com.co/universidades/corporacion-de-estudios-tecnologicos-del-norte-del-valle/licenciaturas-virtuales), la institución ofrece 3 jornadas las cuales son: Diurna, Nocturna y Sabatina.

Para el periodo académico 2021-1 cuenta con 170 estudiantes en la jornada diurna, 254 estudiantes en la jornada diurna y 22 estudiantes en la jornada sabatina, para un total de 446 estudiantes. Así mismo, su planta de docentes es de 71 profesionales, (Corporación de Estudios Tecnológicos del Norte del Valle, 2021).

# Diseño metodológico

Desde la ingeniería de sistemas, la metodología a implementar será el modelo en cascada propuesto por Royce; se desarrollarán las 7 etapas inicialmente mencionadas por él, por lo que en el proyecto no se planea seguir los 5 pasos resumidos propuestos. De estas siete etapas, se explican de la siguiente manera:

* Requisitos del sistema: Para esta etapa se llevarán a cabo las entrevistas propuestas para lograr identificar los diferentes requisitos necesarios para el software, fuera de esto se hablará con los diferentes docentes para recolectar datos importantes del mismo.
* Requisitos de software: En esta etapa se detallará que requisitos necesita el sistema web (requisitos de computadora y/o, como su nombre lo indica, de software).
* Análisis: Aquí se revisan todos los datos obtenidos de los pasos anteriores y se trata de regir correctamente los parámetros y/o detalles que puedan surgir en el proyecto.
* Diseño: En esta etapa se llevará a cabo la realización de los diferentes tipos de diagramas necesarios para el desarrollo, como son el caso de los diagramas de uso, actividades, etc. Para llevar a cabos dichos diagramas se utilizará la herramienta Visual Paradigm.
* Codificación: El lenguaje que se utilizara para el software será PHP, que se llevara a cabo dentro de las herramientas de Sublime Text y Visual Code, junto con el manejo de bases de datos de la herramienta MYSQL.
* Prueba: Para lograr dicha etapa se llevará a cabo diversas pruebas del software para verificar su funcionamiento correcto de sus funciones y sus interfaces.
* Servicio u operaciones: En esta etapa se destacará más que todo el servicio del sistema y/o posible mantenimiento que pueda dársele.

## Enfoque

El tipo de enfoque que ayudará en este proyecto de investigación será el enfoque cualitativo, “este consiste en el estudio de métodos de recolección de datos de tipo descriptivo y de observaciones para descubrir de manera discursiva categorías conceptuales”, (@Sanjahingu, 2014).

## Métodos y Tipo de Investigación

El método que se empleará será: el método inductivo, “el cual parte de fenómenos particulares para llegar a generalizaciones. Esto se refiere a pasar de los resultados obtenidos de la observación y experimentación con elementos particulares a la formulación de hipótesis, principios y leyes de tipo general”, (@Sanjahingu, 2014).

La investigación descriptiva “se refiere o narra características y propiedades de un objeto, sujeto o situación específica, sin emplear juicios de valor y en procura de altos niveles de objetividad” (@Sanjahingu, 2014).

## Técnicas de Recolección de Datos

Las técnicas de recolección de información a utilizar en el proyecto, definidas según (Liberato, 2013) serán:

• La observación: “Este permite estudiar los aspectos más significativos de los objetos, hechos, situaciones sociales o personas en el contexto donde se desarrollan normalmente; permitiendo la comprensión de la verdadera realidad del fenómeno”.

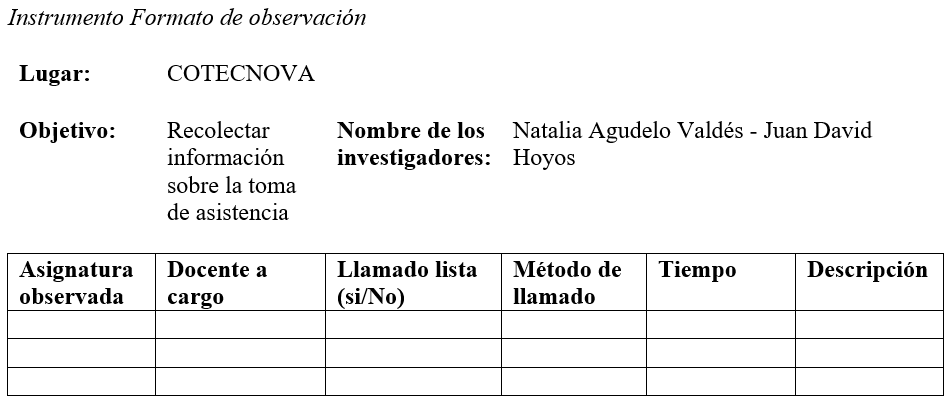
• La entrevista: “Es una conversación dirigida, con un propósito especifico que usa un formato de preguntas y respuestas. Se establece así un diálogo, donde una de las partes busca recoger informaciones y la otra se nos presenta como fuente de estas informaciones”.

### Instrumento Observación

Este instrumento se elabora dentro de una ficha descriptiva en donde, por asignaturas, se tomarán datos de la observación de las clases de manera no participativa.

A continuación, se muestra el formato para la observación:

Ilustración Formato de observación



*Fuente: Elaboración propia*

### Instrumento Entrevista

Esta entrevista se hace con el propósito de saber que opinan los docentes en cuanto al sistema que han llevado a lo largo de los años y el cómo, actualmente, las tecnologías han podido o no influir en éste.

La entrevista es de carácter estructurado y semiestructurado, consta de 8 preguntas en las cuales encontrarán preguntas abiertas, cerradas de selección múltiple.

Para una mejor observación y detalle consultar el Anexo Formato de entrevista.

## Población y Muestra

La población a ser estudiada se tomará como un grupo focal, el cual será un grupo de ingeniería en sistemas de la jornada diurna.

La muestra en primer lugar son los estudiantes de séptimo semestre de la jornada diurna en el programa de tecnología en sistemas de información, la cual pertenece a la ingeniería de sistemas por ciclos propedéuticos en COTECNOVA, pero en realidad, en un producto final, es para toda la comunidad académica.

# Esquema Temático

## Capítulo 1: Identificación de requerimientos

En este capítulo se expondrán las técnicas de recolección de datos con sus respectivos análisis, igualmente, se especifican los requerimientos del sistema, con algunos lineamientos del REM.

### Observación

El proceso se realizó durante la jornada diurna, se tomaron 6 asignaturas de séptimo semestre de la carrera Técnico en Sistemas de Información, teniendo en cuenta el sistema de alternancia planteado por COTECNOVA. Según los datos recolectados mediante el formato de observación (Anexo Formato de observación diligenciado), se obtuvo el siguiente análisis:

La mayoría de los docentes durante las clases prefieren hacer su control de asistencia minutos antes de finalizar la mismo. Tienen diferentes métodos destacando entre ellos el oral, el cual consiste en que el docente llama a los estudiantes y en su respectivo formulario marca si asisten o no; otro método es la escritura virtual, la cual se lleva a cabo en los medios tecnológicos utilizados para ver las clases (Google Meet), el estudiante por medio del chat escribe el nombre completo junto con la palabra presente, estos nombres quedan grabados durante el transcurso de la clase para luego ser exportados en un documento de texto; por último, se tiene el visual, en el cual el docente hace un reconocimiento rápido de los estudiantes (En el aula de clase o plataforma virtual) y determina que estudiantes han faltado a la clase.

Cabe mencionar que, con el último punto mencionado anteriormente algunos docentes ya reconocen los rostros de sus estudiantes (también por el tamaño de su grupo, pequeño/mediano) así que, para ellos, en ocasiones, el llamado de manera visual resulta más cómodo.

El tiempo intermedio para estos diferentes métodos de llamado fueron alrededor de 5 minutos como mínimo y el tiempo máximo no superó los 15 minutos.

Durante este análisis también se notó que algunos estudiantes durante las clases llevadas con instrumentos tecnológicos (virtuales) solo ingresaban a la clase para “constatar su asistencia” pero en ningún momento interactuaban con el chat o micrófono, ni cuando los docentes solicitan su participación o afirmación para determinar la asistencia.

Resaltando el punto anterior, cuando se presenta una inasistencia, es de conocimiento previo que al estudiante faltante se le hace llegar un correo notificando su falta a la clase (Anexo comunicado de falta de asistencia), este procedimiento es manejado por el sector de registro y control ya que allí es donde normalmente los docentes hacen su gestión; este correo no adjunta ningún archivo y/o tema tratado de la clase dictada y, en algunos casos, el carro no se presenta.

### Entrevista

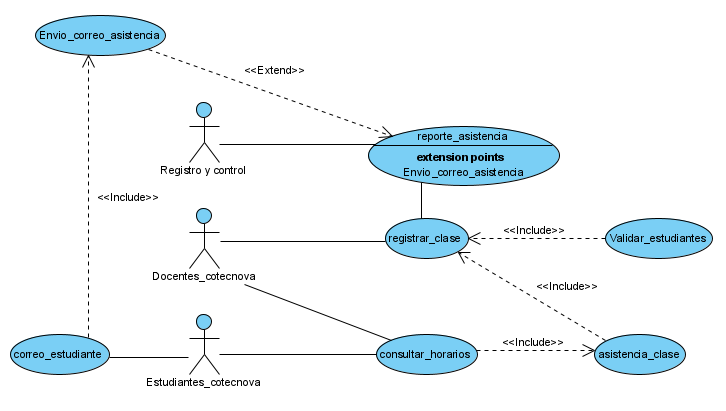
El proceso se realizó durante la jornada diurna, se tomaron 5 docentes de séptimo semestre de la carrera Técnico en Sistemas de Información, teniendo en cuenta el sistema de alternancia planteado por COTECNOVA, Según los datos recolectados mediante la entrevista (Anexo link [Entrevista - Formularios de Google](https://docs.google.com/forms/d/13FMQuKFiKA6ENUpz0VMLYhxHTCd1fJ3oNuizwKewESg/edit)), se obtuvo el siguiente análisis:

La mayoría de los docente encuestados consideraron que el llamado a lista interrumpe el proceso normal de la clase ya que este método tradicional reduce el tiempo de la misma, ya que no todos los estudiantes están presentes en el llamado inicial y toca volver a realizar dicho proceso para validar a los estudiantes faltantes, junto con esto se pudo determinar que entre más grande el grupo de estudiantes la experiencia se vuelve aun peor a la hora realizar el llamado a lista.

La mayoría de docentes utiliza los llamados tradicionales y lista de chequeo a la hora de realizar el respectivo control sin importar si se encuentra en clases presenciales o por medio de tecnologías virtuales, cabe destacar que los docentes apoyan la idea de que el uso de las tecnologías les ayudaría a facilitar el trabajo de la toma de asistencia, ya que estas les ayudarían a optimizar el tiempo de la misma.

Una vez finalizada la clase los docentes en su totalidad reportan las inasistencias de sus estudiantes ya que estos deben de realizar el respectivo control, estos también notifican a dichos estudiantes que no asistieron a la clase, donde según los docentes lo hacen para que estos no se atrasen en las actividades, dicho lo anterior, los docentes sienten el deber de enviar los materiales o contenido de la clase ya sea que si hayan asistido o no a la misma.

Ilustración   
Diagrama general de caso de uso del sistema actual



*Fuente: Fuente propia con la ayuda de la herramienta visual paradigm*

### REM

Para el desarrollo de los requerimientos del sistema se utilizó el programa REM, este sirve para la recolección de los mismos y ayudó a identificar los problemas y casos que puedan surgir en el proyecto para así, en profundidad, explicar cómo funciona el sistema.

#### Participantes en el proyecto

Tabla 1   
Organizaciones del proyecto

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Organización** | **Dirección** | **Teléfono** | **Fax** | **Comentarios** |
| **Cotecnova** | Cl. 10 #3-95, Cartago, Valle del Cauca | 2111804 | 213 44 21. Cartago, Valle del Cauca | Ninguno |
| **Grupo** | No aplica | No aplica | No aplica | Ninguno |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 2   
Participantes del proyecto

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Participante** | **Organización** | **Rol** | **Desarrollador** | **Cliente** | **Usuario** | **Comentarios** |
| **Natalia Agudelo Valdés** | Grupo | Desarrollador principal | Sí | No | No | Ninguno |
| **Juan David Hoyos** | Grupo | Desarrollador principal | Sí | No | No | Ninguno |
| **Arvey Barahona** | Grupo | Profesor | No | Sí | Sí | Ninguno |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

#### Descripción del sistema actual

En la corporación de Estudios Tecnológicos del Norte del Valle, fundada en 1971, el docente lleva un control de asistencia estudiantil de forma tradicional (documentos impresos como listas de chequeo, llamado de forma oral, entre otros); el uso de este mecanismo puede ser poco eficaz a la hora de realizar registro estudiantil a grupos numerosos, ya que gasta demasiado tiempo en el llamado a lista; terminada la clase, el docente debe dejar un reporte de los estudiantes que no asistieron a la misma.

#### Objetivos del proyecto

Tabla 3   
Objetivos del proyecto

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Objetivos** | **Descripción** | **Subobjetivos** | **Importancia** | **Urgencia** | **Estado** | **Estabilidad** | **Comentarios** |
| **Datos estudiantiles y docentes** | El sistema deberá *recolectar todos los datos acerca de los estudiantes y docentes* | Ninguno | vital | inmediatamente | en construcción | alta | Ninguno |
| **Control de estudiantes y docentes** | El sistema deberá *apuntar al control de estudiantes y docentes que han sido registrados y asignados a sus clases* | Ninguno | vital | inmediatamente | en construcción | alta | Ninguno |
| **Datos clases** | El sistema deberá *permitir registrar las clases de los docentes* | Ninguno | importante | hay presión | en construcción | alta | Ninguno |
| **Datos horarios** | El sistema deberá *permitir ingresar y modificar los respectivos datos de los horarios que fueron asignados a los estudiantes y docentes* | Ninguno | importante | hay presión | en construcción | alta | Ninguno |
| **Datos administrador** | El sistema deberá *almacenar los datos de los administrador* | Ninguno | vital | inmediatamente | en construcción | alta | Ninguno |

Continuación Tabla Objetivos del proyecto

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Objetivos** | **Descripción** | **Subobjetivos** | **Importancia** | **Urgencia** | **Estado** | **Estabilidad** | **Comentarios** |
| **Datos aulas** | El sistema deberá *almacenar los datos de las distintas aulas de clase* | Ninguno | importante | hay presión | en construcción | alta | Ninguno |
| **Datos Materias** | El sistema deberá *almacenar los datos de las distintas materias que se usarán para los horarios* | Ninguno | vital | inmediatamente | en construcción | alta | Ninguno |
| **Datos carreras** | El sistema deberá *almacenar la información de todas las carreras disponibles* | Ninguno | vital | inmediatamente | en construcción | alta | Ninguno |
| **Material adjunto** | El sistema deberá permitir a los docentes adjuntar sus links de material de trabajo visto en la clase | Ninguno | importante | puede esperar | en construcción | alta | Ninguno |
| **Iniciar sesión** | El sistema deberá contar con un inicio de sesión para todos los usuarios | Ninguno | vital | inmediatamente | en construcción | alta | Ninguno |

Fuente: Fuente propia, con ayuda de la herramienta REM

#### Catálogo de requisitos del sistema

##### Requisitos de información

Tabla 4   
Requisito de estudiante

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0001** | **Estudiante** | |
| **Versión** | 1.0 ( 03/10/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos estudiantes y docentes * [OBJ] Control de estudiantes y docentes | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a los estudiantes. En concreto: | |
| **Datos específicos** | * Id ( Identificador del estudiante) * Documento ( Numero de T.I o C.C) * Nombres ( Nombres completos de la persona) * Apellidos ( Apellidos completos de la persona) * Jornada ( Diurna, Nocturna, Sabatina) * Semestre ( I, II, III...) * clave ( clave del estudiante) * Horario\_id ( Id del horario correspondiente) * Carrera\_id ( Id de la carrera correspondiente) * tipo\_usuario\_id\_tipo\_usuario ( Id del tipo de usuario) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 500 año(s) | 999 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | media | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 5   
Requisito de horario

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0002** | **Horario** | |
| **Versión** | 1.0 ( 23/10/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos estudiantes y docentes * [OBJ] Datos Materias * [OBJ] Datos horarios * [OBJ] Datos aulas | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a el horario del respectivo tipo de usuario. En concreto: | |
| **Datos específicos** | * Id (*Identificador del horario*) * Hora (*Hora a la que se ve la respectiva materia*) * Aulas\_id (*Aula en la que se dará la respectiva materia*) * materia\_id\_materia (*id de la materia del horario*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 500 año(s) | 999 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | media | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 6   
Requisito de tipo de usuario

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0003** | **Tipo de usuario** | |
| **Versión** | 1.0 ( 23/10/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos estudiantes y docentes * [OBJ] Datos administrador | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a los tipos de usuarios. En concreto: | |
| **Datos específicos** | * Id\_tipo\_usuario (*1, 2 o 3*) * Nombre del usuario (*Estudiante, Administrador o Docente*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 500 año(s) | 999 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | media | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 7 Requisito de clase

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0004** | **Clase** | |
| **Versión** | 1.0 ( 23/10/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos clases * [OBJ] Datos estudiantes y docentes * [OBJ] Datos aulas | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a las clases. En concreto: | |
| **Datos específicos** | * Id\_clase (*Identificador de la clase*) * Dia (*Dia de la clase*) * horario\_id (*Id del horario*) * docente\_id\_docente (*Id del docente*) * estudiante\_id\_estudiante (*Id del estudiante*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 500 año(s) | 999 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | hay presión | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 8  
Requisito de administrador

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0005** | **Administrador** | |
| **Versión** | 1.0 ( 23/10/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos administrador | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a el administrador. En concreto: | |
| **Datos específicos** | * Id\_administrador (*Identificador del administrador*) * Documento (*Numero del documento*) * Nombres (*Nombres completos del administrador*) * Apellidos (*Apellidos completos del administrador*) * Tipo de usuario\_id (*Id del tipo*) * Clave (*clave del administrador*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 500 año(s) | 999 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 9   
Requisito de aula

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0006** | **Aula** | |
| **Versión** | 1.0 ( 09/11/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos aulas | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a las aulas. En concreto: | |
| **Datos específicos** | * Id\_aula (*Del aula*) * Nombre (*Del aula*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 999 año(s) | 500 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | importante | |
| **Urgencia** | hay presión | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 10   
Requisito de materia

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0007** | **Materia** | |
| **Versión** | 1.0 ( 11/11/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos Materias | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a las materias. En concreto: | |
| **Datos específicos** | * Id\_materia (*De la materia*) * Nombre (*De la materia*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 999 año(s) | 500 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 11   
Requisito de docente

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0008** | **Docente** | |
| **Versión** | 1.0 ( 16/11/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos estudiantes y docentes * [OBJ] Control de estudiantes y docentes | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a los docentes. En concreto: | |
| **Datos específicos** | * Id (*Identificador del docente*) * Documento (*Numero de C.C*) * Nombres (*Nombres completos de la persona*) * Apellidos (*Apellidos completos de la persona*) * Tipo de usuario\_id (*Id que identifica que tipo de usuario es*) * Clave (*Clave del docente*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 500 año(s) | 999 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | media | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 12   
Requisito carrera

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0010** | **Carrera** | |
| **Versión** | 1.0 ( 16/11/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Datos carreras | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a las carreras. En concreto: | |
| **Datos específicos** | * Id\_carrera (*Identificador de la carrera*) * Nombre (*Nombre de la carrera*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 500 año(s) | 999 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 500 | 999 |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 13   
Requisito material adjunto

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **IRQ-0011** | **Material adjunto** | |
| **Versión** | 1.0 ( 18/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | * [OBJ] Material adjunto | |
| **Descripción** | El sistema deberá almacenar la información correspondiente a los links adjuntados por los docentes. En concreto: | |
| **Datos específicos** | * Id\_material ( *1, 2, 3..*) * link ( *https://...*) | |
| **Tiempo de vida** | **Medio** | **Máximo** |
| 999 año(s) | 500 año(s) |
| **Ocurrencias simultáneas** | **Medio** | **Máximo** |
| 999 | 500 |
| **Importancia** | importante | |
| **Urgencia** | puede esperar | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

#### Requisitos Funcionales

Tabla 14   
Requisitos funcionales

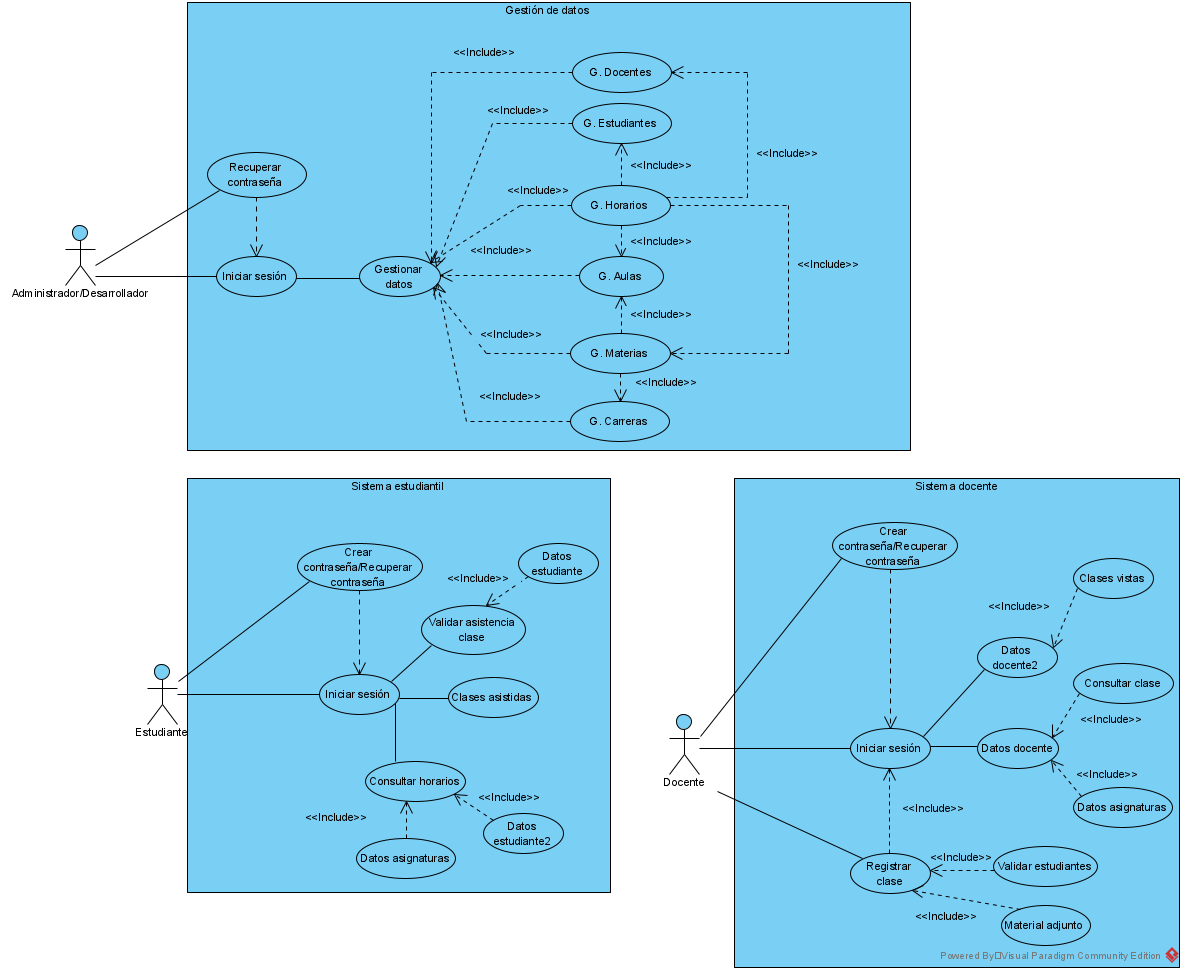
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Requisitos funcionales** | **Dependencias** | **Descripción** | **Importancia** | **Urgencia** | **Estado** | **Estabilidad** | **Comentarios** |
| **Recuperar contraseña/crear contraseña** | * [OBJ] Datos estudiantes y docentes * [OBJ] Iniciar sesión | El sistema deberá permitir que los usuarios recuperen su contraseña si la han olvidado o la creen. | vital | inmediatamente | en construcción | media | Ninguno |
| **Iniciar sesión** | * [[OBJ] Iniciar sesión](file:///D:\Natalia\Documents\Descargas\Universidad\Septimo%20semestre\Proyecto%20integrador\Proyecto_REM_Actualizado\COTECNOVA_REM_Default_Spanish_CRS.html#OBJ-0011) * [[OBJ] Datos estudiantes y docentes](file:///D:\Natalia\Documents\Descargas\Universidad\Septimo%20semestre\Proyecto%20integrador\Proyecto_REM_Actualizado\COTECNOVA_REM_Default_Spanish_CRS.html#OBJ-0001) | El sistema deberá contar con un inicio de sesión | vital | inmediatamente | en construcción | alta | Ninguno |
| **Gestionar datos** | * [OBJ] Datos estudiantes y docentes * [OBJ] Datos horarios * [OBJ] Datos aulas * [OBJ] Datos carreras * [OBJ] Datos Materias | El sistema deberá permitir al administrador gestionar los datos (crear, modificar, eliminar) de los Docentes, estudiantes, horarios, aulas, materias y carreras | vital | inmediatamente | en construcción | alta | Ninguno |
| **Validar asistencia a clases** | * [OBJ] Control de estudiantes y docentes | El sistema deberá permitir al estudiante validar su asistencia a la respectiva clase. | vital | inmediatamente | en construcción | alta | Ninguno |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Requisitos funcionales** | **Dependencias** | **Descripción** | **Importancia** | **Urgencia** | **Estado** | **Estabilidad** | **Comentarios** |
| **Consultar horarios** | * [OBJ] Datos estudiantes y docentes * [OBJ] Datos horarios | El sistema deberá permitir al estudiante consultar su horario. | vital | inmediatamente | en construcción | media | Ninguno |
| **Consultar clases** | * [OBJ] Datos clases | El sistema deberá permitir al docente consultar sus clases | vital | inmediatamente | en construcción | media | Ninguno |
| **Datos estudiantes** | * [OBJ] Datos estudiantes y docentes | El sistema deberá mostrar los datos del estudiante. | vital | inmediatamente | en construcción | baja | Ninguno |
| **Datos docentes** | * [OBJ] Datos estudiantes y docentes | El sistema deberá mostrar los datos del docente. | vital | hay presión | en construcción | baja | Ninguno |
| **Datos asignaturas** | * [OBJ] Datos Materias | El sistema deberá mostrar las asignaturas matriculadas (respectivamente a docentes y a estudiantes). | vital | inmediatamente | en construcción | baja | Ninguno |
| **Registrar clase** | * [OBJ] Datos clases | El sistema deberá permitir al docente registrar su clase | vital | inmediatamente | en construcción | media | Ninguno |
| **Validar estudiantes** | * [OBJ] Datos estudiantes y docentes * [OBJ] Control de estudiantes y docentes | El sistema deberá El sistema deberá permitir al docente validar la asistencia de sus estudiantes a la clase | importante | inmediatamente | En construcción | baja | Ninguno |
| **Clases vistas** | * [OBJ] Datos clases | El sistema deberá permitir al docente consultar las clases vistas. | importante | inmediatamente | En construcción | media | Ninguno |
| **Clases asistidas** | * [OBJ] Datos clases | El sistema deberá permitir al estudiante consultar las clases vistas. | importante | inmediatamente | En construcción | media | Ninguno |
| **Material adjunto** | * [OBJ] Material adjunto | El sistema deberá permitir que los docentes adjunten links a sus clases y/o envíen material | importante | puede esperar | En construcción | alta | Ninguno |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

##### Diagramas de caso de uso

Ilustración   
Diagrama de casos de uso



Fuente: Fuente propia con ayuda de la herramienta Visual Paradigm

##### Definición de actores

Tabla 15   
Definición de actores

|  |  |  |
| --- | --- | --- |
| **Actores** | **Descripción** | **Comentarios** |
| **Desarrollador/Administrador** | Este actor representa *a la persona que se encarga de administrar el sistema* | Ninguno |
| **Docente** | Este actor representa *al docente como tal, el cual maneja su propio horario* | Ninguno |
| **Estudiante** | Este actor representa *al estudiante como tal para la asistencia a clases* | Ninguno |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

##### Casos de uso del sistema

Tabla 16   
Caso de uso 1

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0001** | **Iniciar sesión** | |
| **Versión** | 1.0 ( 16/10/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** | Ninguno | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando los usuarios inicien sesión | |
| **Precondición** | Debe de existir una BD para rectificar los datos Deben existir los usuarios en la BD | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El sistema carga el formulario respectivo |
| 2 | El sistema recibe los datos del usuario |
| 3 | El sistema el sistema valida que todos los campos estén llenos |
| 4 | El sistema el sistema rectifica que la BD esté conectada |
| 5 | El sistema el sistema envía los datos a la BD y los rectifica |
| 6 | El sistema devuelve el login respectivo |
| **Postcondición** | Se confirman los datos y el usuario puede iniciar sesión | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si los campos no están llenos, el sistema aloja una alerta, a continuación este caso de uso continúa |
| 5 | Si el usuario no existe, el sistema arroja un mensaje de alerta, a continuación este caso de uso continúa |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | registro de un nuevo estudiante | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 17   
Caso de uso 2

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0002** | **Registrar clase** | |
| **Versión** | 1.0 ( 16/10/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0004] Clase | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando un docente desea registrar su clase | |
| **Precondición** | Debe existir en el usuario en la bd  Debe tener sus respectivos horarios de clase | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Docente (ACT) ingresa a la plataforma |
| 2 | El actor Docente (ACT) inicia sesión como docente |
| 3 | El sistema valida la existencia del docente |
| 4 | El actor Docente (ACT) procede al módulo de registrar clases |
| 5 | El actor Docente (ACT) selecciona la asignatura deseada |
| 6 | El actor Docente (ACT) selecciona el grupo deseado |
| 7 | El actor Docente (ACT) tiene la opción de crear un nuevo código o permanecer con el existente |
| 8 | El actor Docente (ACT) El docente cierra la clase con sus estudiantes registrados |
| **Postcondición** | El docente puede registrar su clase | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el docente no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Registro de clase | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 18   
Caso de uso 3

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0003** | **Gestionar Docentes** | |
| **Versión** | 1.0 ( 28/10/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0008] Docente   [IRQ-0005] Administrador | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el desarrollador/administrador entre al sistema y quiera gestionar los docentes | |
| **Precondición** | Debe existir un Administrador/Desarrollador registrado | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Desarrollador/Administrador (ACT) ingresa a la plataforma |
| 2 | El actor Desarrollador/Administrador (ACT) inicia sesión como administrador |
| 3 | El sistema valida la existencia del administrador |
| 4 | El actor Desarrollador/Administrador (ACT) procede al módulo de gestionar docente |
| 5 | El actor Desarrollador/Administrador (ACT) selecciona la acción deseada (Agregar, editar, eliminar) y procede con el respectivo formulario |
| **Postcondición** | El administrador gestiona al docente | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el administrador no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Registro de un docente en el sistema | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 19   
Caso de uso 4

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0004** | **Gestionar Estudiantes** | |
| **Versión** | 1.0 ( 28/10/2019 ) | |
| **Autores** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0005] Administrador   [IRQ-0001] Estudiante | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el desarrollador/administrador entre al sistema y quiera gestionar los datos del estudiante | |
| **Precondición** | Debe existir un Administrador/Desarrollador registrado | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Desarrollador/Administrador (ACT) ingresa a la plataforma |
| 2 | El actor Desarrollador/Administrador (ACT) inicia sesión como administrador |
| 3 | El sistema valida la existencia del administrador |
| 4 | El actor Desarrollador/Administrador (ACT) procede al módulo de gestionar estudiante |
| 5 | El actor Desarrollador/Administrador (ACT) selecciona la acción deseada (Agregar, editar, eliminar) y procede con el respectivo formulario |
| **Postcondición** | El administrador gestiona al estudiante | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el administrador no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Registro de un estudiante en el sistema | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 20   
Caso de uso 5

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0005** | **Gestionar horarios** | |
| **Versión** | 1.0 ( 28/10/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0002] Horario   [IRQ-0006] Aula   [IRQ-0007] Materia   [IRQ-0008] Docente   [IRQ-0001] Estudiante | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el desarrollador/administrador entre al sistema y quiera gestionar los datos de los horarios | |
| **Precondición** | Deben existir estudiantes registrados en el sistema  Deben existir docentes registrados en el sistema  Debe existir un administrador en el sistema | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Desarrollador/Administrador (ACT) ingresa a la plataforma |
| 2 | El actor Desarrollador/Administrador (ACT) inicia sesión como administrador |
| 3 | El sistema valida la existencia del administrador |
| 4 | El actor Desarrollador/Administrador (ACT) procede al módulo de gestionar horarios |
| 5 | El actor Desarrollador/Administrador (ACT) selecciona el horario a gestionar |
| 6 | El actor Desarrollador/Administrador (ACT) selecciona la acción deseada (Agregar, editar, eliminar) y procede con el respectivo formulario |
| **Postcondición** | El administrador gestiona el horario | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el administrador no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 21   
Caso de uso 6

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0006** | **Gestionar aulas** | |
| **Versión** | 1.0 ( 09/11/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0006] Aula | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el desarrollador/administrador entre al sistema y quiera gestionar los datos de las aulas | |
| **Precondición** | Debe existir un Administrador/Desarrollador registrado | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Desarrollador/Administrador (ACT) ingresa a la plataforma |
| 2 | El actor Desarrollador/Administrador (ACT) inicia sesión como administrador |
| 3 | El sistema valida la existencia del administrador |
| 4 | El actor Desarrollador/Administrador (ACT) procede al módulo gestionar aulas |
| 5 | El actor Desarrollador/Administrador (ACT) selecciona la acción deseada (Agregar, editar, eliminar) y procede con el respectivo formulario |
| **Postcondición** | El administrador gestiona las aulas | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el administrador no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 22   
Caso de uso 7

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0007** | **Gestionar materias** | |
| **Versión** | 1.0 ( 11/11/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0007] Materia | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el desarrollador/administrador entre al sistema y quiera gestionar los datos de las materias | |
| **Precondición** | Debe existir un Administrador/Desarrollador registrado | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Desarrollador/Administrador (ACT) ingresa a la plataforma |
| 2 | El actor Desarrollador/Administrador (ACT) inicia sesión como administrador |
| 3 | El sistema valida la existencia del administrador |
| 4 | El actor Desarrollador/Administrador (ACT) procede al módulo de gestionar materias |
| 5 | El actor Desarrollador/Administrador (ACT) selecciona la acción deseada (Agregar, editar, eliminar) y procede con el respectivo formulario |
| **Postcondición** | El administrador gestiona las materias | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el administrador no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 23   
Caso de uso 8

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0008** | **Gestionar carreras** | |
| **Versión** | 1.0 ( 16/11/2019 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0010] Carrera | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el desarrollador/administrador entre al sistema y quiera gestionar los datos de las carreras | |
| **Precondición** | Debe existir un Administrador/Desarrollador registrado | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Desarrollador/Administrador (ACT) ingresa a la plataforma |
| 2 | El actor Desarrollador/Administrador (ACT) inicia sesión como administrador |
| 3 | El sistema valida la existencia del administrador |
| 4 | El actor Desarrollador/Administrador (ACT) procede al módulo de gestionar carreras |
| 5 | El actor Desarrollador/Administrador (ACT) selecciona la acción deseada (Agregar, editar, eliminar) y procede con el respectivo formulario |
| **Postcondición** | El administrador gestiona las carreras | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el administrador no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 24   
Caso de uso 9

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0009** | **Crear contraseña/Recuperar contraseña** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | * Juan David Hoyos | |
| **Fuentes** | * Juan David Hoyos * Natalia Agudelo Valdés | |
| **Dependencias** | Ninguno | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el usuario quiera crear/recuperar su contraseña | |
| **Precondición** | El usuario debe estar registrado en el sistema | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El sistema carga el formulario respectivo |
| 2 | El sistema recibe los datos del usuario |
| 3 | El sistema el sistema valida que todos los campos estén llenos |
| 4 | El sistema el sistema rectifica que la BD esté conectada |
| 5 | El sistema el sistema envía los datos a la BD y los rectifica |
| 6 | El sistema devuelve el login respectivo |
| **Postcondición** | La contraseña se cambia exitosamente | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si los campos no están llenos, el sistema aloja una alerta, a continuación este caso de uso continúa |
| 5 | Si el usuario no existe, el sistema arroja un mensaje de alerta, a continuación este caso de uso continúa |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | registro de un nuevo estudiante | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 25   
Caso de uso 10

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0010** | **Gestionar datos** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0006] Aula   [IRQ-0001] Estudiante   [IRQ-0002] Horario   [IRQ-0007] Materia   [IRQ-0010] Carrera   [IRQ-0008] Docente | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el administrador inicie sesión y quiera gestionar los datos | |
| **Precondición** | El administrador debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Desarrollador/Administrador (ACT) ingresa a la plataforma |
| 2 | El actor Desarrollador/Administrador (ACT) inicia sesión como administrador |
| 3 | El sistema valida la existencia del administrador |
| 4 | El actor Desarrollador/Administrador (ACT) procede al módulo de gestionar datos |
| 5 | El actor Desarrollador/Administrador (ACT) Gestiona los datos deseados |
| **Postcondición** | El administrador gestiona los datos deseados | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el administrador no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 26   
Caso de uso 11

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0011** | **Datos docente** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0008] Docente | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el docente inicia sesión y está en su página principal | |
| **Precondición** | El docente debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Docente (ACT) ingresa a la plataforma |
| 2 | El actor Docente (ACT) inicia sesión como docente |
| 3 | El sistema valida la existencia del docente |
| 4 | El actor Docente (ACT) entra a su página principal |
| **Postcondición** | El docente observa sus respectivos datos | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el docente no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 27   
Caso de uso 12

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0012** | **Datos estudiante** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0001] Estudiante | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el estudiante inicia sesión y está en su página principal | |
| **Precondición** | El estudiante debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Estudiante (ACT) ingresa a la plataforma |
| 2 | El actor Estudiante (ACT) inicia sesión como estudiante |
| 3 | El sistema valida la existencia del estudiante |
| 4 | El actor Estudiante (ACT) entra a su página principal |
| **Postcondición** | El estudiante observa sus respectivos datos | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el estudiante no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 28   
Caso de uso 13

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0013** | **Datos asignaturas estudiantes** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0001] Estudiante   [IRQ-0006] Aula   [IRQ-0007] Materia   [IRQ-0008] Docente   [IRQ-0002] Horario | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el estudiante inicia sesión y esta entra al módulo de consultar horario | |
| **Precondición** | El estudiante debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Estudiante (ACT) ingresa a la plataforma |
| 2 | El actor Estudiante (ACT) inicia sesión como estudiante |
| 3 | El sistema valida la existencia del estudiante |
| 4 | El actor Estudiante (ACT) entra a su página principal |
| 5 | El actor Estudiante (ACT) ingresa al módulo de consultar horarios |
| 6 | El actor Estudiante (ACT) observa los datos de las asignaturas matriculadas |
| **Postcondición** | El estudiante observa las matrículas asignadas en consultar horarios | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el estudiante no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 29   
Caso de uso 14

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0014** | **Datos asignaturas docentes** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0002] Horario   [IRQ-0006] Aula   [IRQ-0007] Materia | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el docente inicia sesión y esta entra al módulo de consultar clases | |
| **Precondición** | El docente debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Docente (ACT) ingresa a la plataforma |
| 2 | El actor Docente (ACT) inicia sesión como docente |
| 3 | El sistema valida la existencia del docente |
| 4 | El actor Docente (ACT) entra a su página principal |
| 5 | El actor Docente (ACT) ingresa al módulo de consultar clases |
| 6 | El actor Docente (ACT) observa los datos de las materias asignadas |
| **Postcondición** | El docente observa las materias que da en consultar clases | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el docente no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 30   
Caso de uso 15

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0015** | **Consultar clases** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0002] Horario   [IRQ-0006] Aula   [IRQ-0007] Materia   [IRQ-0008] Docente | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el docente inicia sesión y esta entra al módulo de consultar clases | |
| **Precondición** | El docente debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Docente (ACT) ingresa a la plataforma |
| 2 | El actor Docente (ACT) inicia sesión como docente |
| 3 | El sistema valida la existencia del docente |
| 4 | El actor Docente (ACT) entra a su página principal |
| 5 | El actor Docente (ACT) ingresa al módulo de consultar clases |
| 6 | El actor Docente (ACT) observa los datos de sus clases |
| **Postcondición** | El docente consulta sus clases | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el docente no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 31   
Caso de uso 16

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0016** | **Consultar horario** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0007] Materia   [IRQ-0002] Horario   [IRQ-0006] Aula   [IRQ-0001] Estudiante   [IRQ-0008] Docente | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el estudiante inicia sesión y esta entra al módulo de consultar horario | |
| **Precondición** | El estudiante debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Estudiante (ACT) ingresa a la plataforma |
| 2 | El actor Estudiante (ACT) inicia sesión como estudiante |
| 3 | El sistema valida la existencia del estudiante |
| 4 | El actor Estudiante (ACT) entra a su página principal |
| 5 | El actor Estudiante (ACT) ingresa al módulo de consultar horarios |
| 6 | El actor Estudiante (ACT) observa los datos de las asignaturas matriculadas |
| **Postcondición** | El estudiante observa su horario | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el estudiante no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 32   
Caso de uso 17

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0017** | **Clases vistas** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0007] Materia   [IRQ-0002] Horario   [IRQ-0001] Estudiante | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el docente inicia sesión y esta entra al módulo de clases vistas | |
| **Precondición** | El docente debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Docente (ACT) ingresa a la plataforma |
| 2 | El actor Docente (ACT) inicia sesión como docente |
| 3 | El sistema valida la existencia del docente |
| 4 | El actor Docente (ACT) entra a su página principal |
| 5 | El actor Docente (ACT) ingresa al módulo de clases vistas |
| 6 | El actor Docente (ACT) selecciona el grupo deseado |
| 7 | El actor Docente (ACT) observa los estudiantes registrados a la clase |
| **Postcondición** | El docente consulta sus clases vistas | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el docente no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 33   
Caso de uso 18

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0018** | **Validar estudiantes** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0001] Estudiante   [IRQ-0002] Horario   [IRQ-0007] Materia | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el docente inicia sesión y esta entra al módulo de registrar clases | |
| **Precondición** | El docente debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Docente (ACT) ingresa a la plataforma |
| 2 | El actor Docente (ACT) inicia sesión como docente |
| 3 | El sistema valida la existencia del docente |
| 4 | El actor Docente (ACT) entra a su página principal |
| 5 | El actor Docente (ACT) ingresa al módulo de registrar clase |
| 6 | El actor Docente (ACT) selecciona el grupo deseado |
| 7 | El actor Docente (ACT) selecciona la materia deseada |
| 8 | El actor Docente (ACT) observa a los estudiantes registrarse en la clase |
| **Postcondición** | El docente consulta validar a sus estudiantes en la clase | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el docente no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 34   
Caso de uso 19

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0019** | **Validar asistencia** | |
| **Versión** | 1.0 ( 17/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0001] Estudiante   [IRQ-0007] Materia | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el estudiante inicia sesión y esta entra al módulo de validar asistencia | |
| **Precondición** | El estudiante debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Estudiante (ACT) ingresa a la plataforma |
| 2 | El actor Estudiante (ACT) inicia sesión como estudiante |
| 3 | El sistema valida la existencia del estudiante |
| 4 | El actor Estudiante (ACT) entra a su página principal |
| 5 | El actor Estudiante (ACT) ingresa al módulo de validar asistencia |
| 6 | El actor Estudiante (ACT) selecciona la materia deseada |
| 7 | El actor Estudiante (ACT) ingresa el código proporcionado por el docente |
| 8 | El actor Estudiante (ACT) se registra en la clase |
| **Postcondición** | El estudiante valida su asistencia a la clase | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el estudiante no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla 35   
Caso de uso 20

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0020** | **Material adjunto** | |
| **Versión** | 1.0 ( 18/05/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0011] Material adjunto | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el docente desee adjuntar links de materiales cuando está registrando la clase y sus estudiantes | |
| **Precondición** | Debe existir el docente en la BD | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Docente (ACT) ingresa a la plataforma |
| 2 | El actor Docente (ACT) inicia sesión como docente |
| 3 | El sistema valida la existencia del docente |
| 4 | El actor Docente (ACT) entra a su página principal |
| 5 | El actor Docente (ACT) ingresa al módulo de registrar clase |
| 6 | El actor Docente (ACT) selecciona al grupo deseado |
| 7 | El actor Docente (ACT) selecciona la materia deseada |
| 8 | El actor Docente (ACT) adjunta los links en el campo disponible |
| **Postcondición** | El docente adjunta los links | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el docente no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | importante | |
| **Urgencia** | puede esperar | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

Tabla   
Caso de uso 21

|  |  |  |
| --- | --- | --- |
| **Elemento** | **Descripción** | |
| **UC-0021** | **Clases asistidas** | |
| **Versión** | 1.0 ( 10/06/2021 ) | |
| **Autores** | Natalia Agudelo Valdés | |
| **Fuentes** | Juan David Hoyos  Natalia Agudelo Valdés | |
| **Dependencias** |  [IRQ-0007] Materia   [IRQ-0002] Horario   [IRQ-0001] Estudiante | |
| **Descripción** | El sistema deberá comportarse tal como se describe en el siguiente caso de uso cuando el estudiante inicia sesión y esta entra al módulo de clases asistidas | |
| **Precondición** | El estudiante debe existir | |
| **Secuencia normal** | **Paso** | **Acción** |
| 1 | El actor Estudiante (ACT) ingresa a la plataforma |
| 2 | El actor Estudiante (ACT) inicia sesión como estudiante |
| 3 | El sistema valida la existencia del estudiante |
| 4 | El actor Estudiante (ACT) entra a su página principal |
| 5 | El actor Estudiante (ACT) ingresa al módulo de clases asistidas |
| 6 | El actor Estudiante (ACT) selecciona la clase y observa los datos deseados |
| **Postcondición** | El estudiante consulta sus clases vistas | |
| **Excepciones** | **Paso** | **Acción** |
| 3 | Si el estudiante no existe, el sistema arroja una alerta, a continuación este caso de uso queda sin efecto |
| **Rendimiento** | **Paso** | **Tiempo máximo** |
| - | - |
| **Frecuencia esperada** | 999 veces por año(s) | |
| **Importancia** | vital | |
| **Urgencia** | inmediatamente | |
| **Estado** | en construcción | |
| **Estabilidad** | alta | |
| **Comentarios** | Ninguno | |

*Fuente: Fuente propia, con ayuda de la herramienta REM*

### Requisitos no funcionales

Tabla 37   
Requisitos no funcionales

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Requisitos no funcionales** | **Dependencias** | **Descripción** | **Importancia** | **Urgencia** | **Estado** | **Estabilidad** | **Comentarios** |
| **Procesamiento de datos** | Ninguno | El sistema deberá *procesar x cantidad de datos* | vital | inmediatamente | en construcción | alta | Ninguno |
| **Respaldo** | Ninguno | El sistema deberá *respaldarse cada 24 horas* | vital | inmediatamente | en construcción | alta | Ninguno |
| **Modificación de la Base de datos** | Ninguno | El sistema deberá *permitir que los datos modificados en la base de datos sean actualizados para todos los usuarios que acceden en menos de 2 segundos* | vital | hay presión | en construcción | media | Ninguno |
| **Advertencias** | Ninguno | El sistema deberá *proporcionar mensajes de error o sonidos de alerta en caso de que sean necesarios* | importante | inmediatamente | en construcción | media | Ninguno |
| **Comunicación** | Ninguno | El sistema deberá *permitir comunicación externas entre los servidores de datos, la aplicación y el cliente del sistema deben estar cifradas en algún algoritmo para garantizar seguridad.* | vital | hay presión | en construcción | alta | Ninguno |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Requisitos no funcionales** | **Dependencias** | **Descripción** | **Importancia** | **Urgencia** | **Estado** | **Estabilidad** | **Comentarios** |
| **Asignación** | Ninguno | El sistema deberá asignar a cada usuario un identificador único para luego ser utilizado en todos los procesos que se realicen sobre este. | vital | Hay presión | En construcción | media | Ninguno |
| **Administración** | Ninguno | El sistema deberá permitir tener uno o varios administradores. | importante | Hay presión | En construcción | baja | Ninguno |
| **Validación** | Ninguno | El sistema deberá validar automáticamente y saber los permisos de los usuarios. | importante | Hay presión | En construcción | media | Ninguno |
| **Seguridad** | Ninguno | El sistema deberá contar con encriptación para garantizar la seguridad en los datos almacenados. | vital | Hay presión | En construcción | alta | Ninguno |
| **Envió automatizado** | Ninguno | El sistema deberá permitir el envío automatizado del registro del docente y de los estudiantes que estuvieron en una clase a una base de datos superior o global. | vital | Hay presión | En construcción | media | Ninguno |

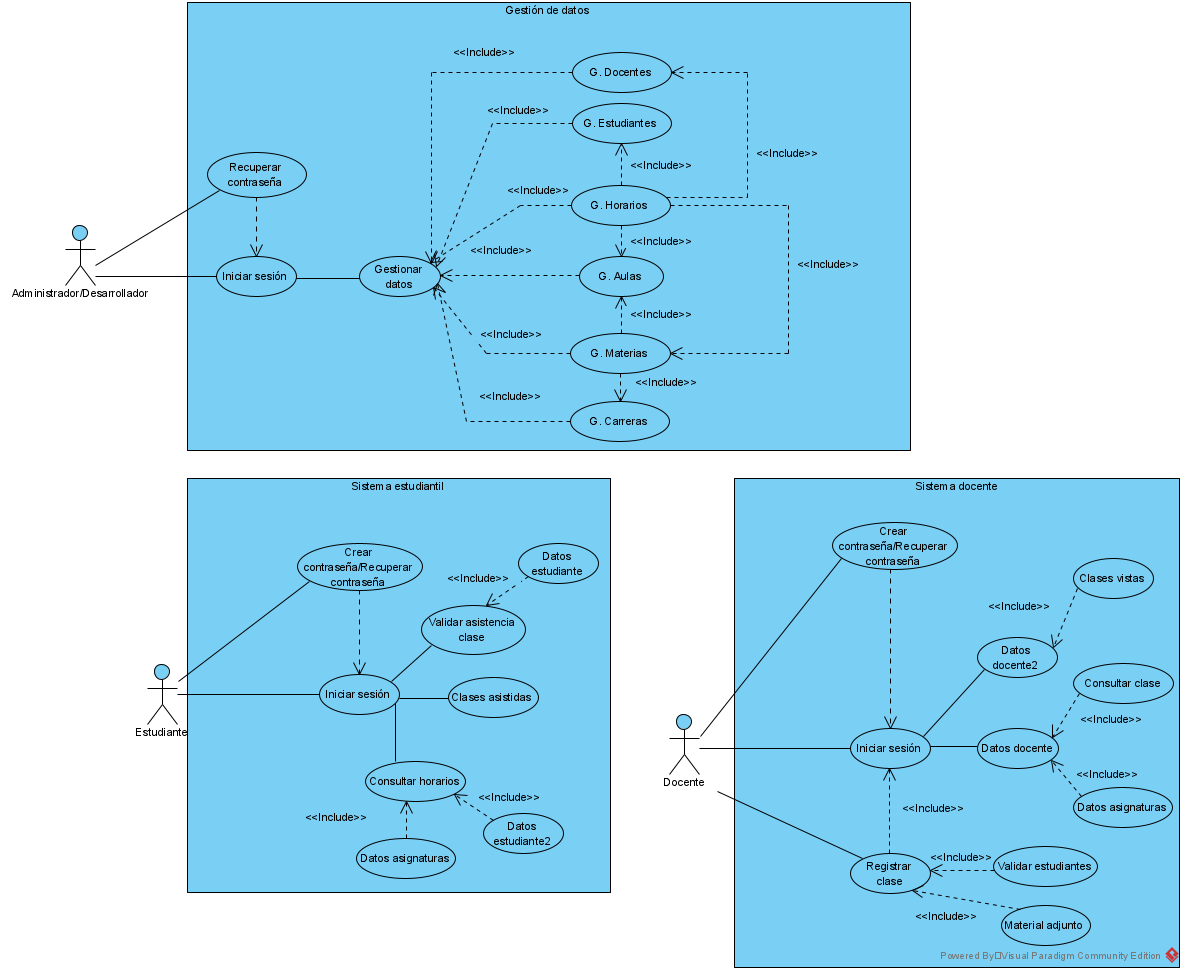
*Fuente: Fuente propia, con ayuda de la herramienta REM*

## Capítulo 2: Diseño del sistema de información

En este capítulo se expondrá todo el contenido propuesto en cuanto al diseño del sistema de información planteado anteriormente, dentro de él se observan los casos de uso, diagramas de clase, diagramas de componentes, diagramas de actividades y diagramas de secuencia, diseñados con la herramienta Visual Paradigm; así mismo, como se diseñó la base de datos y su estructura.

### Diagrama de casos de uso

Ilustración   
Diagrama de casos de uso

  
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Según el diagrama de casos de uso planteado anteriormente se tiene qué, el funcionamiento de cada caso consiste en:

Tabla 38   
Descripción del funcionamiento de los casos de uso

|  |  |  |
| --- | --- | --- |
| **Caso de uso** | **Nombre** | **Descripción** |
| **UC-0001** | **Iniciar sesión** | Este caso de uso tiene relación con los tres actores existentes en el sistema en donde, los actores cuenten con un sistema de login por el cual podrán ingresar a la plataforma para hacer su respectivo proceso. |
| **UC-0002** | **Registrar clase** | Este caso de uso se relaciona con el docente, el docente debe iniciar primeramente en la plataforma, una vez realizado este proceso podrá ingresar a su apartado donde iniciará el proceso de registro, allí debe escoger el grupo y la asignatura deseada para continuar con el registro. |
| **UC-0003** | **Gestionar Docentes** | Este caso de uso se relaciona con administrador, el administrador debe ingresar a la plataforma, una vez hecho ese proceso debe acceder al módulo correspondiente para poder realizar la gestión del docente. |
| **UC-0004** | **Gestionar Estudiantes** | Este caso de uso se relaciona con administrador, el administrador debe ingresar a la plataforma, una vez hecho ese proceso debe acceder al módulo correspondiente para poder realizar la gestión del estudiante. |
| **UC-0005** | **Gestionar Horarios** | Este caso de uso se relaciona con administrador, el administrador debe ingresar a la plataforma, una vez hecho ese proceso debe acceder al módulo correspondiente para poder realizar la gestión de los horarios. |
| **UC-0006** | **Gestionar Aulas** | Este caso de uso se relaciona con administrador, el administrador debe ingresar a la plataforma, una vez hecho ese proceso debe acceder al módulo correspondiente para poder realizar la gestión de las aulas. |
| **UC-0007** | **Gestionar Materias** | Este caso de uso se relaciona con administrador, el administrador debe ingresar a la plataforma, una vez hecho ese proceso debe acceder al módulo correspondiente para poder realizar la gestión de las materias. |
| **UC-0008** | **Gestionar Carreras** | Este caso de uso se relaciona con administrador, el administrador debe ingresar a la plataforma, una vez hecho ese proceso debe acceder al módulo correspondiente para poder realizar la gestión de las carreras. |
| **UC-0009** | **Crear contraseña/Recuperar contraseña** | Este caso de uso tiene relación con los tres actores existentes en el sistema en donde, los actores deben dar clic en texto subrayado del login para realizar el respectivo proceso de crear la contraseña o recuperarla. |
| **UC-0010** | **Gestionar Datos** | Este caso reúne los anteriores casos mencionados anteriormente de “gestión” en donde el actor administrador, procede a hacer las respectivas gestiones deseadas. |
| **UC-0011** | **Datos Docentes** | Este caso de uso va relacionado al docente como con el sistema, ya que el sistema debe comportarse tal como se describe para que así el docente cuando inicie sesión pueda observar los diferentes datos almacenados sobre él. |
| **UC-0012** | **Datos estudiantes** | Este caso de uso va relacionado al estudiante como con el sistema, ya que el sistema debe comportarse tal como se describe para que así el estudiante cuando inicie sesión pueda observar los diferentes datos almacenados sobre él. |
| **UC-0013** | **Datos asignaturas estudiantes** | Este caso de uso va relacionado con el estudiante, primeramente debe iniciar sesión en el aplicativo web, posteriormente debe dirigirse a su apartado consultar horarios donde podrá obtener la información de sus asignaturas. |
| **UC-0014** | **Datos asignaturas docentes** | Este caso de uso va relacionado con el docente, primeramente debe iniciar sesión en el aplicativo web, posteriormente debe dirigirse a su apartado consultar clases donde podrá obtener la información de las asignaturas que dictará. |
| **UC-0015** | **Consultar Clases** | Este caso de uso va relacionado con el docente, primeramente debe iniciar sesión en el aplicativo web, posteriormente debe dirigirse a su apartado consultar clases donde podrá obtener la información deseada. |
| **UC-0016** | **Consultar Horario** | Este caso de uso va relacionado con el estudiante, primeramente debe iniciar sesión en el aplicativo web, posteriormente debe dirigirse a su apartado consultar horario donde podrá obtener la información deseada. |
| **UC-0017** | **Clases Vistas** | Este caso de uso va relacionado con el docente, primeramente debe iniciar sesión en el aplicativo web, posteriormente debe dirigirse a su apartado de clases vistas donde podrá obtener la información deseada. |
| **UC-0018** | **Validar Estudiantes** | Este caso de uso va relacionado con el docente, este deberá acceder a la plataforma mediante el proceso de login, luego de ello deberá acceder al módulo de registrar clase donde escogerá la asignatura y el grupo deseado, luego de ello iniciará el proceso de validar la asistencia de sus estudiantes a la clase. |
| **UC-0019** | **Validar asistencia** | Este caso de uso va relacionado con el estudiante, este deberá acceder a la plataforma mediante el proceso de login, una vez hecho el respectivo proceso el estudiante deberá acceder al módulo Validar asistencia y seleccionar la asignatura deseada, luego de ello deberá ingresar el código suministrado por el docente y así registrarse en la clase. |
| **UC-0020** | **Material adjunto** | Este caso de uso va relacionado con el docente, este deberá acceder a la plataforma mediante el proceso de login, luego de ello deberá acceder al módulo de registrar clase donde escogerá la asignatura y el grupo deseado, allí tendrá un apartado en donde podrá dejar los links de material de trabajo para su clase. Otra opción para ello es ir al módulo de clases vistas y seleccionar la clase que se dio, allí puede editar el material dejado anteriormente. |
| **UC-0021** | **Clases Asistidas** | Este caso de uso va relacionado con el estudiante, primeramente debe iniciar sesión en el aplicativo web, posteriormente debe dirigirse a su apartado de clases asistidas donde podrá obtener la información deseada. |

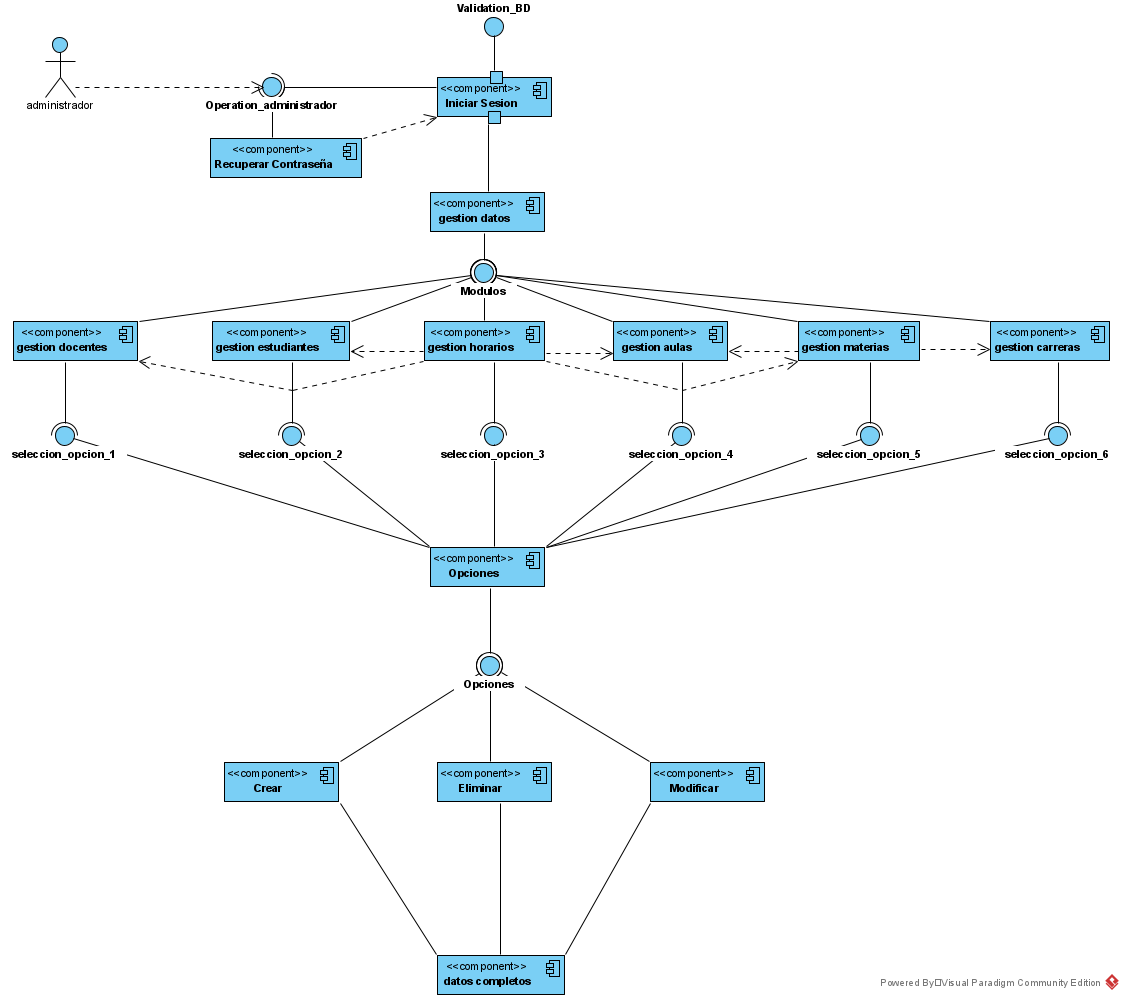
*Fuente: Elaboración propia*

### Diagrama de componentes

Estos diagramas están divididos según el rol del usuario, en este caso, se presentan tres usuarios los cuales son: Administrador, Docente y Estudiante.

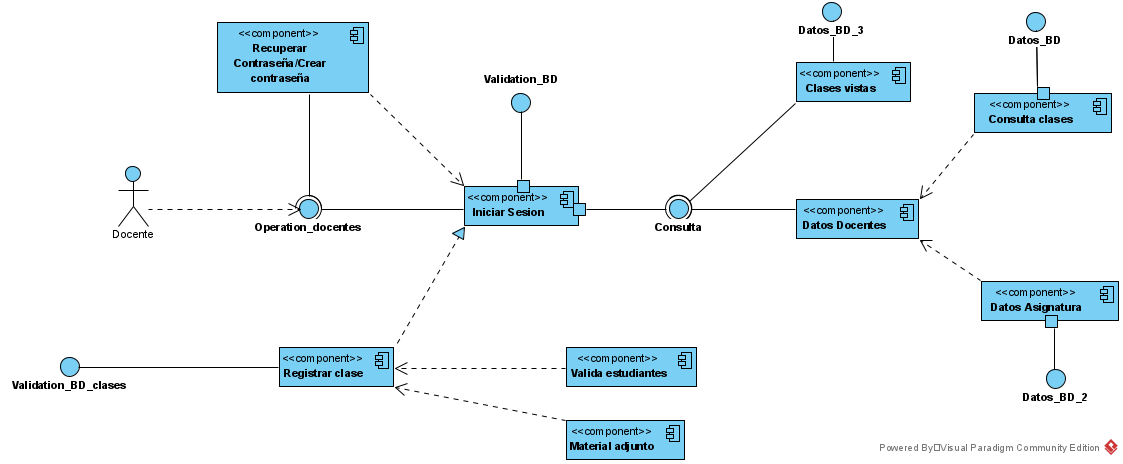
#### Diagrama del Administrador

Ilustración   
Diagrama de componentes administrador

  
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

#### Diagrama del Docente

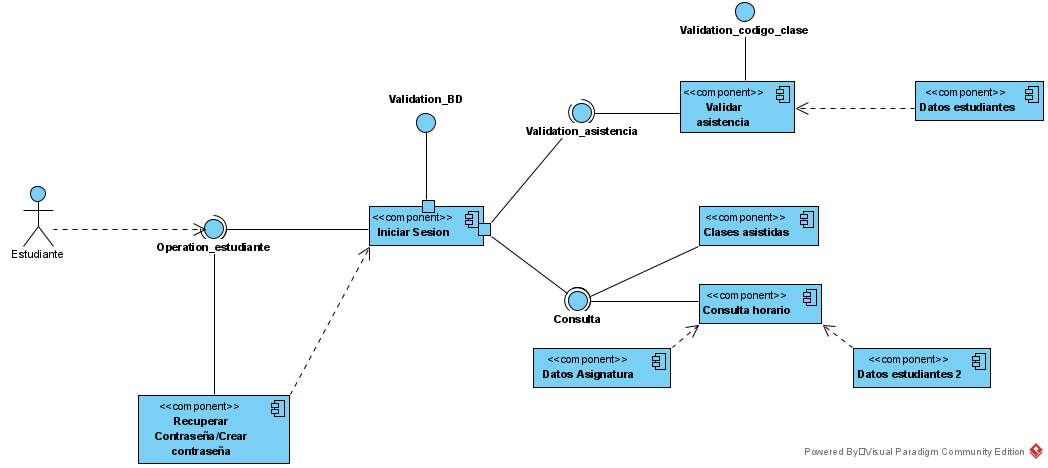
Ilustración   
Diagrama de componentes del docente

****

*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

#### Diagrama del Estudiante

Ilustración   
Diagrama de componentes del estudiante

****

*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Tabla   
Descripción del funcionamiento de los diagramas de componentes

|  |  |  |
| --- | --- | --- |
| **Diagrama Componentes** | **Usuario** | **Descripción** |
| **DC-00001** | **Administrador** | Este diagrama va relacionado al administrador, en este se muestran los diferentes componentes que tiene el software en cuanto a las funciones del administrador, el cual tendrá que hacer el proceso de login para poder acceder a los diferentes módulos de gestión de cada proceso y hacer sus respectivas operaciones en cada una de ellas. |
| **DC-00002** | **Docente** | Este diagrama va relacionado con el docente, en este se muestran los diferentes componentes que tiene el software en cuanto a las funciones del docente, el cual tendrá que hacer el proceso de login para poder acceder a los diferentes módulos, este podrá registrar las clases, validar a los estudiantes, consultar sus clases por dar y sus clases vistas. |
| **DC-00003** | **Estudiante** | Este diagrama va relacionado con el estudiante, en este se muestran los diferentes componentes que tiene el software en cuanto a las funciones del estudiante, el cual tendrá que hacer el proceso de login para poder acceder a los diferentes módulos, este podrá consultar su horario, validar su asistencia a clase y consultar sus clases vistas. |

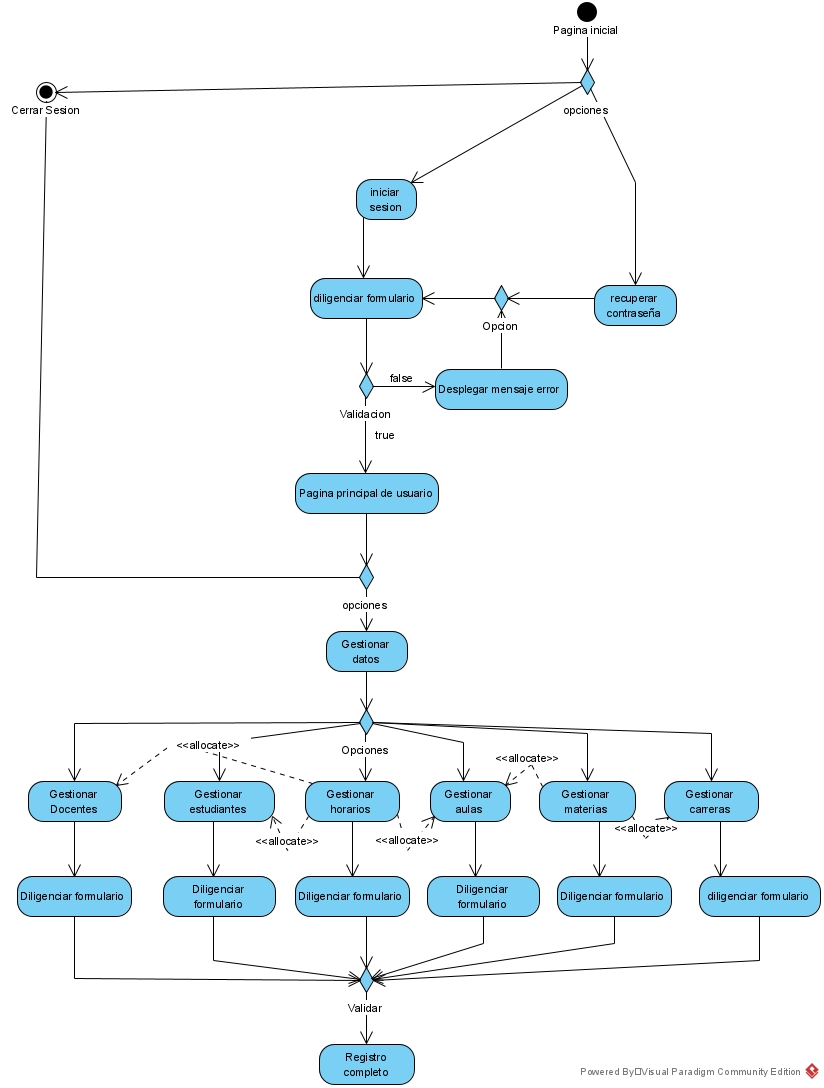
*Fuente: Elaboración propia*

### Diagrama de actividades

Estos diagramas están divididos según el rol del usuario, en este caso, se presentan tres usuarios los cuales son: Administrador, Docente y Estudiante.

#### Diagrama del Administrador

Ilustración   
Diagrama de actividades del administrador

****

*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

#### Diagrama del Docente

Ilustración   
Diagrama de actividades del docente

****

*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

#### Diagrama del Estudiante

Ilustración   
Diagrama de actividades del estudiante

****

*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Tabla   
Descripción del funcionamiento de los diagramas de actividades

|  |  |  |
| --- | --- | --- |
| **Diagrama Actividades** | **Usuario** | **Descripción** |
| **DA-00001** | **administrador** | Este diagrama va relacionado al administrador, en este se muestran las diferentes actividades que tiene el software en cuanto a las funciones del administrador, el administrador deberá elegir entre las opciones que saldrán en el login, dependiendo cual elija tendrá diferentes respuestas en el sistema, la primera si escoge la opción de iniciar sesión este deberá llenar un formulario de login, una vez hecho el proceso anterior dispondrá de varias opciones para hacer diferentes gestiones en los datos almacenados, dependiendo de qué gestión desee hacer tendrá que completar ciertos campos de información sobre estos y en caso de no iniciar sesión, la segunda, realizará el proceso de recuperar su contraseña para así realizar el procedimiento de la primera opción. |
| **DA-00002** | **Docente** | Este diagrama va relacionado con el docente, en este se muestran las diferentes actividades que tiene el software en cuanto a las funciones del docente, este deberá elegir entre las opciones que saldrán en el login, dependiendo cual elija tendrá diferentes respuestas en el sistema, la primera si escoge la opción de iniciar sesión este deberá llenar un formulario de login, una vez hecho el proceso anterior dispondrá de varias opciones dependiendo de que desee hacer, entre ellos tiene: registrar la clase, ver las clases vistas o consultarlas, en caso opuesto, la segunda opción del login, podrá crear o recuperar su contraseña y así realizar las demás actividades mencionadas anteriormente |
| **DA-00003** | **Estudiante** | Este diagrama va relacionado con el estudiante, en este se muestran las diferentes actividades que tiene el software en cuanto a las funciones del estudiante, este deberá elegir entre las opciones que saldrán en el login, dependiendo cual elija tendrá diferentes respuestas en el sistema, la primera si escoge la opción de iniciar sesión este deberá llenar un formulario de login, una vez hecho el proceso anterior dispondrá de varias opciones dependiendo de qué desee hacer tendrá que completar ciertos campos de información sobre estos, entre las actividades que podrá realizar esta consultar su horario de clases, consultar sus clases vistas y validar su asistencia en la clase llenando algunos campos solicitados por el sistema, en caso de ir por la segunda opción del login este podrá crear o recuperar su contraseña para así ejecutar las demás actividades. |

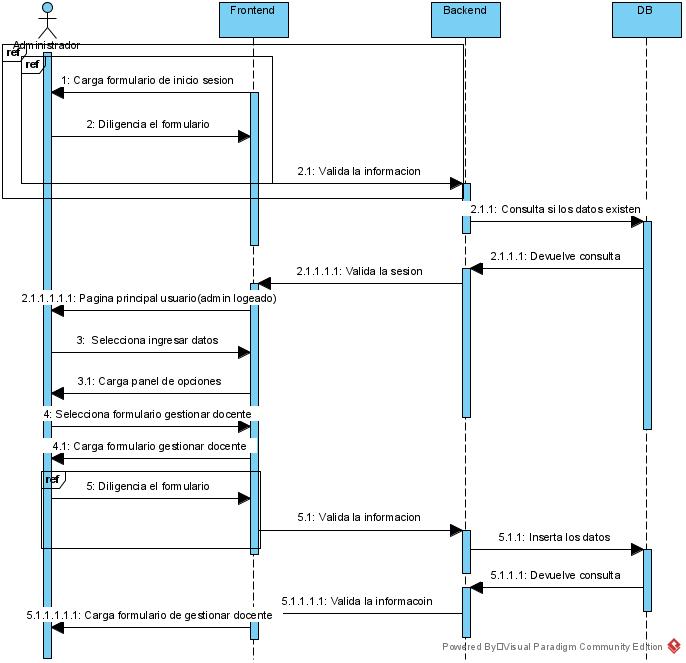
*Fuente: Elaboración propia*

### Diagramas de secuencia

Estos diagramas están divididos según el rol del usuario, en este caso, se presentan tres usuarios los cuales son: Administrador, Docente y Estudiante, de igual forma, por cada rol de usuario se identifican sus propios diagramas (acciones) que forman parten del sistema.

#### Diagramas del Administrador

Ilustración   
Gestionar docente

****

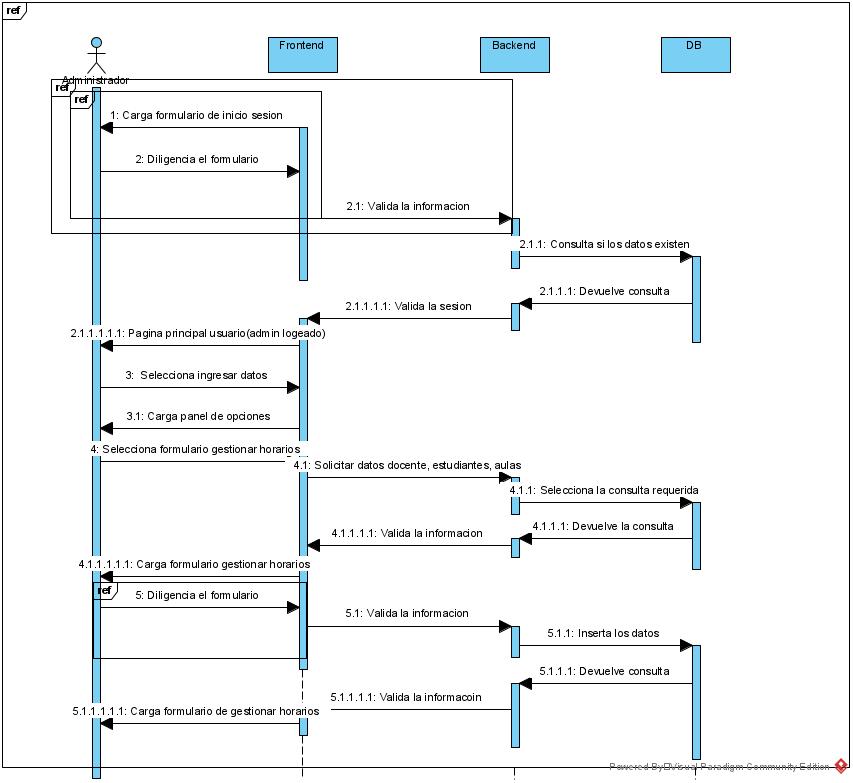
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Gestionar estudiantes

****

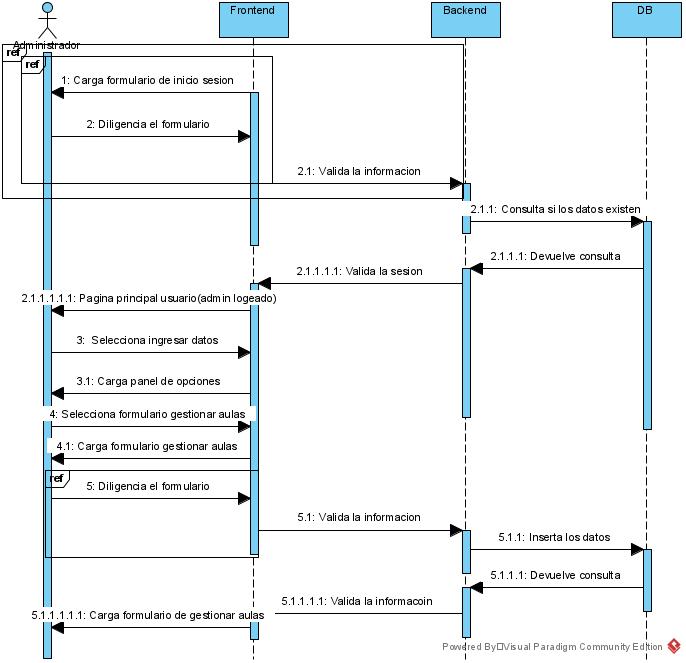
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Gestionar horarios

****

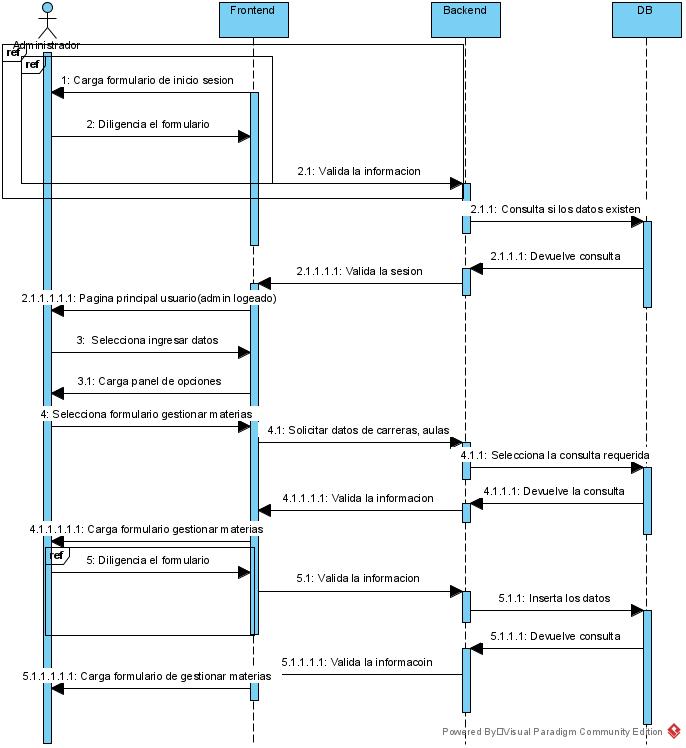
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Gestionar aulas

****

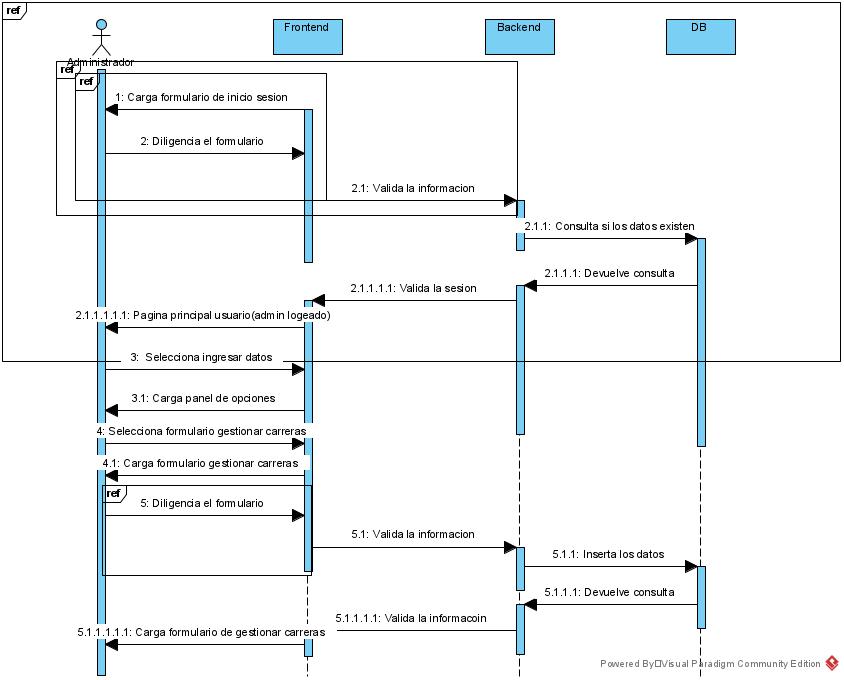
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Gestionar materias

****

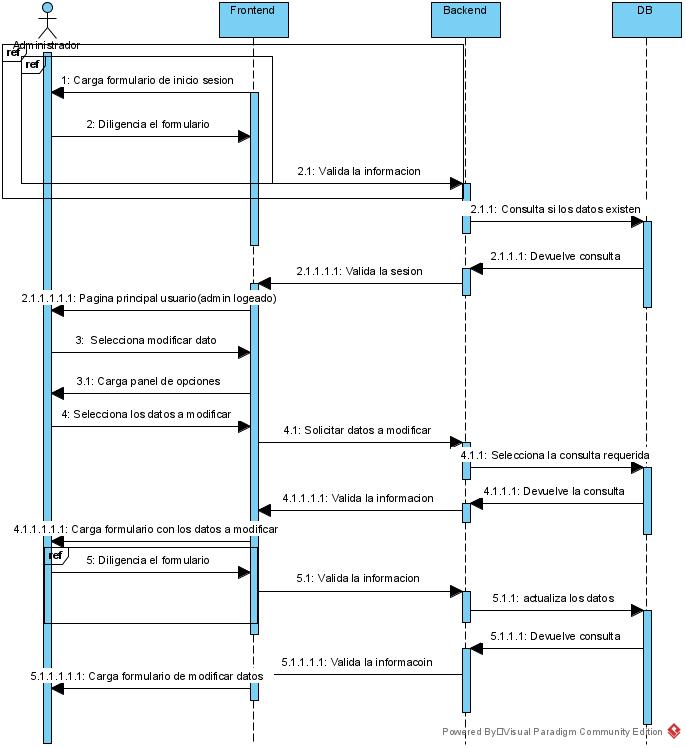
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Gestionar carreras

****

*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Modificar datos

****

*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

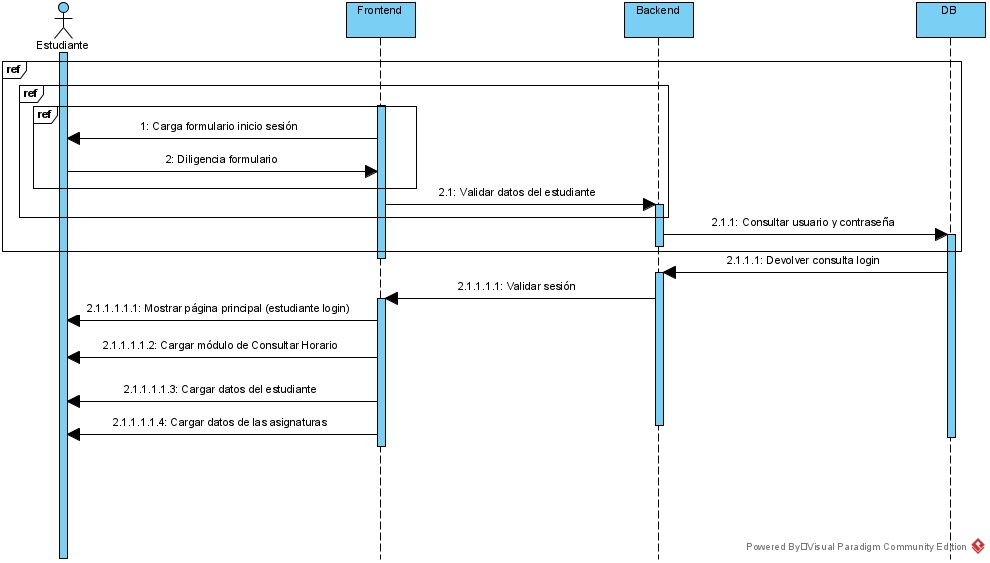
Tabla   
Descripción del funcionamiento de los diagramas de secuencia del administrador

|  |  |  |
| --- | --- | --- |
| **Diagrama Secuencia** | **Nombre** | **Descripción** |
| **DS-00001** | **Gestionar docente** | Este diagrama va relacionado al administrador, en el frontend cargara el formulario de inicio sesión donde el administrador deberá llenarlo con los campos solicitados, estos serán enviados al backend donde se realizara la validación de los datos ingresados los cuales están conectados a la BD donde se hace la consulta de los datos para determinar si estos existen y devolver la consulta por backend con la validación de la sesión volviendo así al frontend donde cargara el menú principal del administrador, allí se cargaran los diferentes módulos que puede gestionar y seleccionar el módulo de gestionar docente y completar los datos pedidos por el formulario, una vez hecho el proceso estos datos serán enviados por medio del backend para así validar la información y hacer la respectiva consulta en la BD, si todo sale devolverá la consulta con la información validada y volver al frontend cargando nuevamente el formulario deseado. |
| **DS-00002** | **Gestionar estudiantes** | Este diagrama va relacionado al administrador, en el frontend cargara el formulario de inicio sesión donde el administrador deberá llenarlo con los campos solicitados, estos serán enviados al backend donde se realizara la validación de los datos ingresados los cuales están conectados a la BD donde se hace la consulta de los datos para determinar si estos existen y devolver la consulta por backend con la validación de la sesión volviendo así al frontend donde cargara el menú principal del administrador, allí se cargaran los diferentes módulos que puede gestionar y seleccionar el módulo de gestionar estudiantes y completar los datos pedidos por el formulario, una vez hecho el proceso estos datos serán enviados por medio del backend para así validar la información y hacer la respectiva consulta en la BD, si todo sale devolverá la consulta con la información validada y volver al frontend cargando nuevamente el formulario deseado. |
| **DS-00003** | **Gestionar horarios** | Este diagrama va relacionado al administrador, en el frontend cargara el formulario de inicio sesión donde el administrador deberá llenarlo con los campos solicitados, estos serán enviados al backend donde se realizara la validación de los datos ingresados los cuales están conectados a la BD donde se hace la consulta de los datos para determinar si estos existen y devolver la consulta por backend con la validación de la sesión volviendo así al frontend donde cargara el menú principal del administrador, allí se cargaran los diferentes módulos que puede gestionar y seleccionar el módulo de gestionar horarios y completar los datos pedidos por el formulario, una vez hecho el proceso estos datos serán enviados por medio del backend para así validar la información y hacer la respectiva consulta en la BD, si todo sale devolverá la consulta con la información validada y volver al frontend cargando nuevamente el formulario deseado. |
| **DS-00004** | **Gestionar aulas** | Este diagrama va relacionado al administrador, en el frontend cargara el formulario de inicio sesión donde el administrador deberá llenarlo con los campos solicitados, estos serán enviados al backend donde se realizara la validación de los datos ingresados los cuales están conectados a la BD donde se hace la consulta de los datos para determinar si estos existen y devolver la consulta por backend con la validación de la sesión volviendo así al frontend donde cargara el menú principal del administrador, allí se cargaran los diferentes módulos que puede gestionar y seleccionar el módulo de gestionar horarios y completar los datos pedidos por el formulario, una vez hecho el proceso estos datos serán enviados por medio del backend para así validar la información y hacer la respectiva consulta en la BD, si todo sale devolverá la consulta con la información validada y volver al frontend cargando nuevamente el formulario deseado. |
| **DS-00005** | **Gestionar materias** | Este diagrama va relacionado al administrador, en el frontend cargara el formulario de inicio sesión donde el administrador deberá llenarlo con los campos solicitados, estos serán enviados al backend donde se realizara la validación de los datos ingresados los cuales están conectados a la BD donde se hace la consulta de los datos para determinar si estos existen y devolver la consulta por backend con la validación de la sesión volviendo así al frontend donde cargara el menú principal del administrador, allí se cargaran los diferentes módulos que puede gestionar y seleccionar el módulo de gestionar materias y completar los datos pedidos por el formulario, una vez hecho el proceso estos datos serán enviados por medio del backend para así validar la información y hacer la respectiva consulta en la BD, si todo sale devolverá la consulta con la información validada y volver al frontend cargando nuevamente el formulario deseado. |
| **DS-00006** | **Gestionar carreras** | Este diagrama va relacionado al administrador, en el frontend cargara el formulario de inicio sesión donde el administrador deberá llenarlo con los campos solicitados, estos serán enviados al backend donde se realizara la validación de los datos ingresados los cuales están conectados a la BD donde se hace la consulta de los datos para determinar si estos existen y devolver la consulta por backend con la validación de la sesión volviendo así al frontend donde cargara el menú principal del administrador, allí se cargaran los diferentes módulos que puede gestionar y seleccionar el módulo de gestionar carreras y completar los datos pedidos por el formulario, una vez hecho el proceso estos datos serán enviados por medio del backend para así validar la información y hacer la respectiva consulta en la BD, si todo sale devolverá la consulta con la información validada y volver al frontend cargando nuevamente el formulario deseado. |
| **DS-00007** | **Modificar Datos** | Este diagrama va relacionado al administrador, en el frontend cargara el formulario de inicio sesión donde el administrador deberá llenarlo con los campos solicitados, estos serán enviados al backend donde se realizara la validación de los datos ingresados los cuales están conectados a la BD donde se hace la consulta de los datos para determinar si estos existen y devolver la consulta por backend con la validación de la sesión volviendo así al frontend donde cargara los módulos otorgados por el sistema allí el administrador deberá seleccionar la opción de modificar el dato, esto cargara el panel de opciones y deberá seleccionar los datos a modificar, estos serán enviados al backend sonde se solicitaran los datos a modificar conectada con la BD donde se seleccionara la consulta requerida, la BD devolverá la consulta con la información validada por el backend hasta volver al frontend donde se cargara el nuevo formulario con los datos a modificar, el administrador deberá diligenciar el formulario y nuevamente estos datos serán enviados al backend donde se validara la información llegando a la BD donde se actualizarán los datos, una vez hecho el proceso la BD devolverá la consulta por el backend con la información validada hasta finalmente volver al frontend con la carga del formulario de modificar datos. |

*Fuente: Elaboración propia*

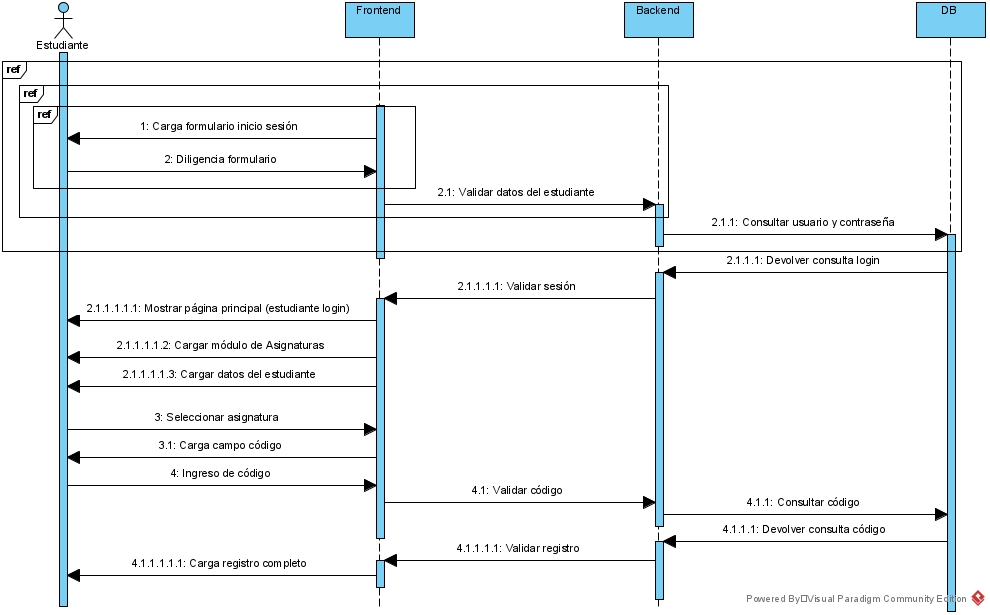
#### Diagramas del Estudiante

Ilustración   
Consultar horario y datos del estudiante y datos de las asignaturas



*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
 Validar asistencia a clase



*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

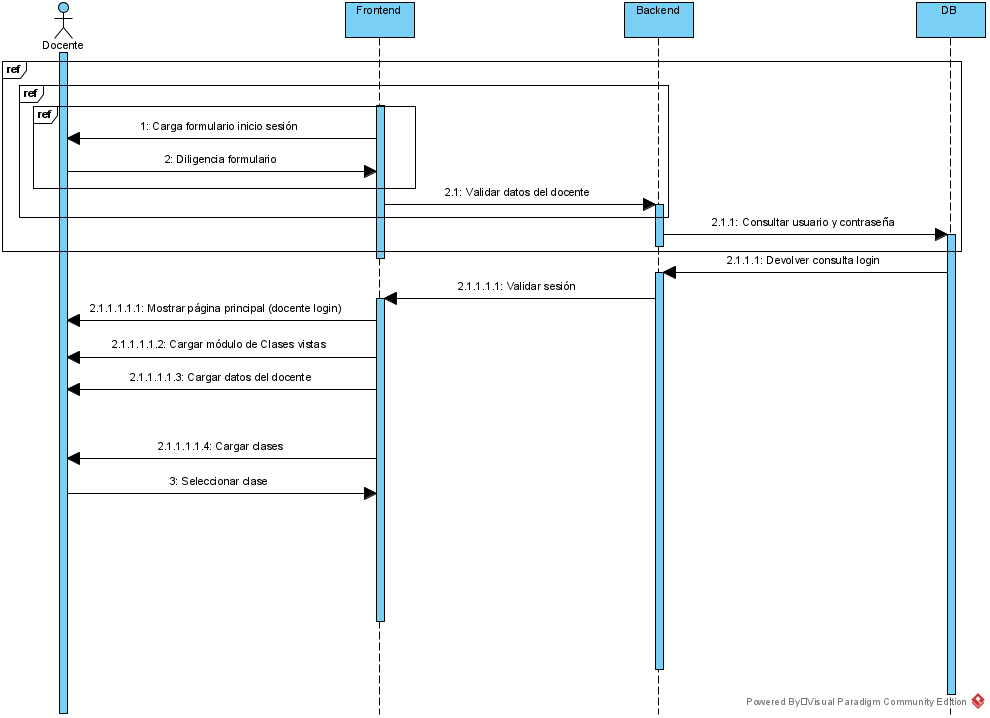
Tabla   
Descripción del funcionamiento de los diagramas de secuencia del estudiante

|  |  |  |
| --- | --- | --- |
| **Diagrama Secuencia** | **Nombre** | **Descripción** |
| **DS-00001** | **Consultar horario y datos del estudiante y datos de las asignaturas** | Este diagrama va relacionado al estudiante, por medio del frontend se cargara el formulario de inicio de sesión donde el estudiante deberá diligenciar el formulario con los datos sugeridos, estos datos serán enviados por medio del backend para validar los datos del estudiante llegando a la BD donde se consulta el usuario y la contraseña del estudiante, esto devolverá la consulta del login al backend con la información validada llegando al frontend donde se mostrara la página principal del estudiante, allí deberá cargar el módulo de consultar horarios y cargar los datos del estudiante junto con los datos de las asignaturas. |
| **DS-00002** | **Validar asistencia a clase** | Este diagrama va relacionado al estudiante, por medio del frontend se cargara el formulario de inicio de sesión donde el estudiante deberá diligenciar el formulario con los datos sugeridos, estos datos serán enviados por medio del backend para validar los datos del estudiante llegando a la BD donde se consulta el usuario y la contraseña del estudiante, esto devolverá la consulta del login al backend con la información validada llegando al frontend donde se mostrara la página principal del estudiante, allí deberá cargarse el módulo de asignaturas y cargar los datos del estudiante, allí el estudiante deberá seleccionar la asignatura deseada, luego se cargara un campo especifico donde deberá ingresar el código suministrado por el docente, esta información será enviada por el backend para validar el código ingresado por el estudiante llegando a la BD donde se hará la respectiva consulta, luego esta consulta será devuelta por el backend para validar el registro llegando finalmente al frontend donde se cargara el registro completo. |

*Fuente: Elaboración propia*

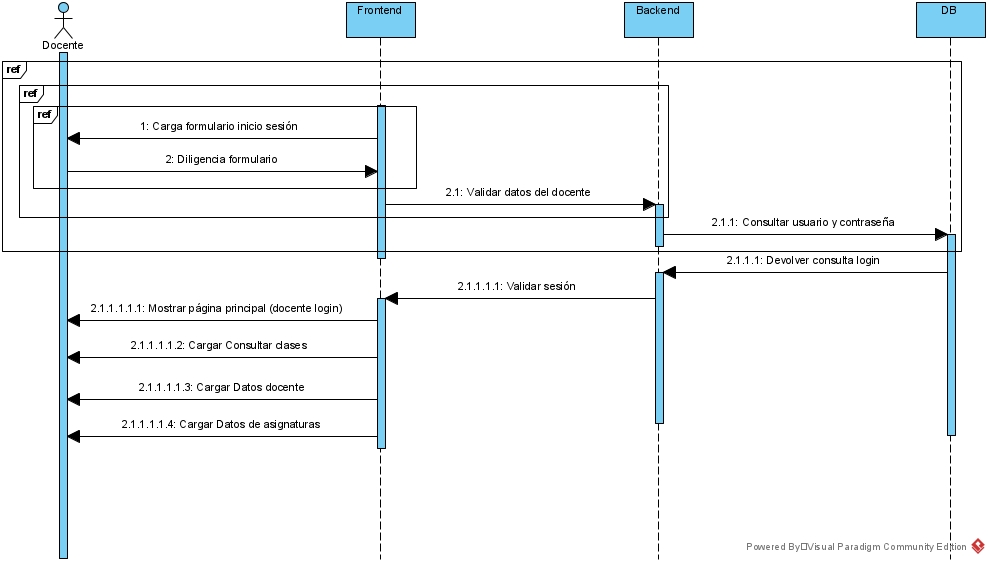
#### Diagramas del Docente

Ilustración   
Clases Vistas



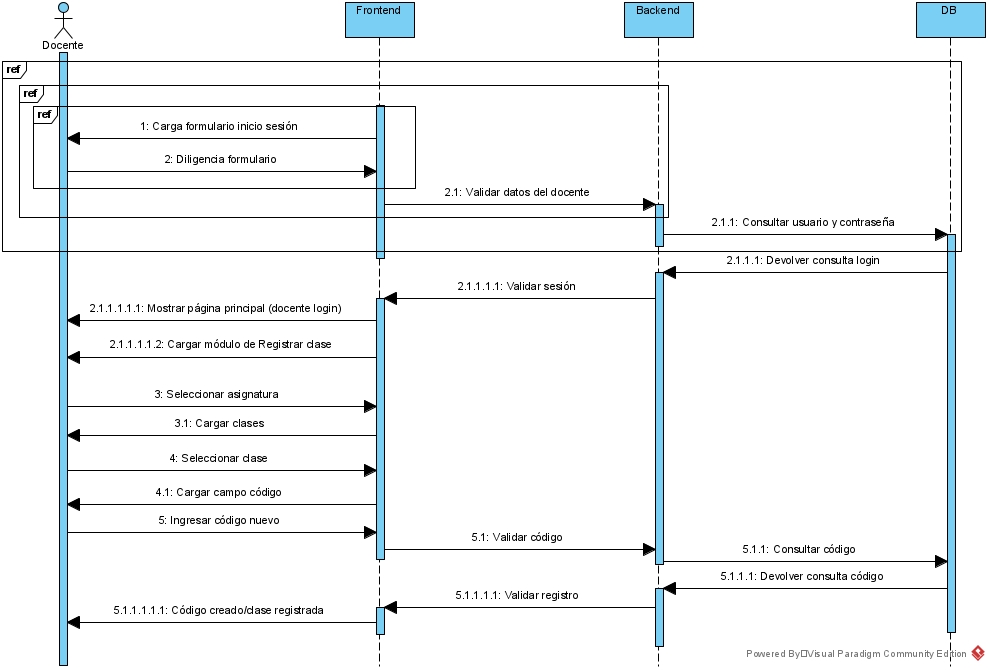
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Consultar clase y datos del docente y de las asignaturas



*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Registrar Clase



*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

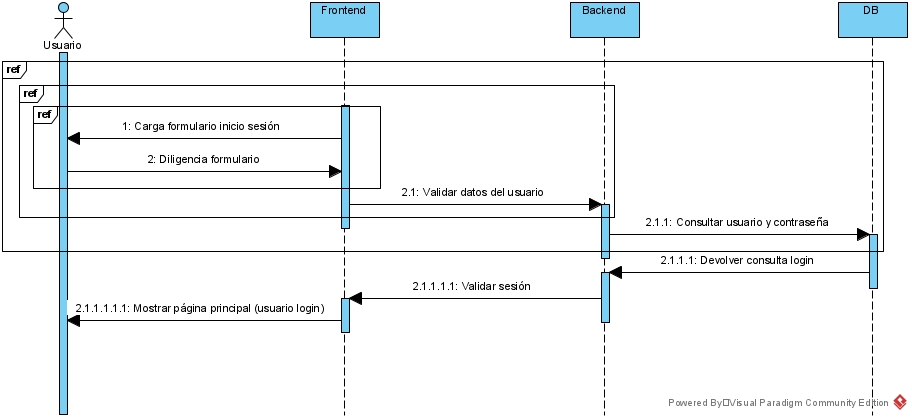
Tabla   
Descripción del funcionamiento de los diagramas de secuencia del docente

|  |  |  |
| --- | --- | --- |
| **Diagrama Secuencia** | **Nombre** | **Descripción** |
| **DS-00001** | **Clases vistas** | Este diagrama va relacionado al docente, por medio del frontend se cargara el formulario de inicio de sesión donde el docente deberá diligenciar el formulario con los datos sugeridos, estos datos serán enviados por medio del backend para validar los datos del docente llegando a la BD donde se consulta el usuario y la contraseña del docente, esto devolverá la consulta del login al backend con la información validada llegando al frontend, donde se cargara la página principal del docente, junto el módulo de las clases vistas la cual seleccionara el usuario esto cargara los datos del docente y las clases, allí el docente selecciona la clase que quiere consultar. |
| **DS-00002** | **Consultar clase y datos del docente y de las asignaturas** | Este diagrama va relacionado al docente, por medio del frontend se cargara el formulario de inicio de sesión donde el docente deberá diligenciar el formulario con los datos sugeridos, estos datos serán enviados por medio del backend para validar los datos del docente llegando a la BD donde se consulta el usuario y la contraseña del docente, esto devolverá la consulta del login al backend con la información validada llegando al frontend donde cargara la página principal del docente , también cargara el módulo de consultar clases la cual será seleccionada por el usuario, esto cargara los datos del docente y las de las asignaturas. |
| **DS-00003** | **Registrar clase** | Este diagrama va relacionado al docente, por medio del frontend se cargara el formulario de inicio de sesión donde el docente deberá diligenciar el formulario con los datos sugeridos, estos datos serán enviados por medio del backend para validar los datos del docente llegando a la BD donde se consulta el usuario y la contraseña del docente, esto devolverá la consulta del login al backend con la información validada llegando al frontend donde cargara la página principal del docente y cargara el módulo de registrar clase, la cual será seleccionada por el usuario, en el cual tendrá que seleccionar la clase, y cargara un campo específico para ingresar el código de la clase, la información suministrada será enviada por el backend donde se validara el código realizando la consulta con la BD, esta devolverá la consulta del código por el backend con la validación del registro llegando nuevamente al frontend con el código creado y la clase registrada. |

*Fuente: Elaboración propia*

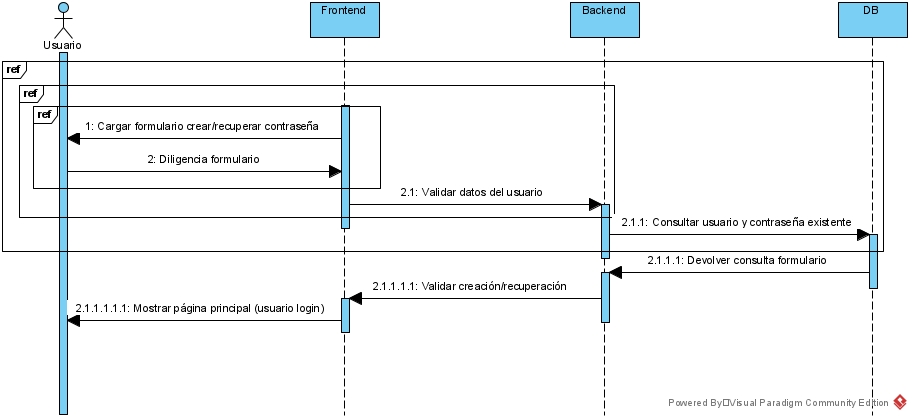
#### Diagramas Generales

Ilustración   
Iniciar Sesión



*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Ilustración   
Crear o Recuperar contraseña



*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

Tabla   
Descripción del funcionamiento de los diagramas de secuencia globales

|  |  |  |
| --- | --- | --- |
| **Diagrama Secuencia** | **Nombre** | **Descripción** |
| **DS-00001** | **Iniciar sesión** | Este diagrama va relacionado a los 3 usuarios planteados, en el frontend se cargara el formulario de inicio de sesión el cual el usuario deberá diligenciarlo con los datos sugeridos estos datos serán enviados al backend para validar dichos datos, llegando a la BD donde se realiza la consulta de usuario y contraseña, luego esta consulta será devuelta al login pasando por el bakend con la validación de la sesión llegando al frontend donde se mostrara la página principal del usuario. |
| **DS-00002** | **Crear o recuperar contraseña** | Este diagrama va relacionado a los 3 usuarios planteados, en el frontend se cargara el formulario de crear/recuperar contraseña, allí el usuario deberá diligenciar el formulario, esta información se validara pasando por el backend hasta llegar a la BD donde se hará la consulta del usuario y contraseña existente, luego esta será devuelta por el backend con la validación de la creación o recuperación del usuario, llegando finalmente al frontend donde se mostrara la página principal del usuario ingresado. |

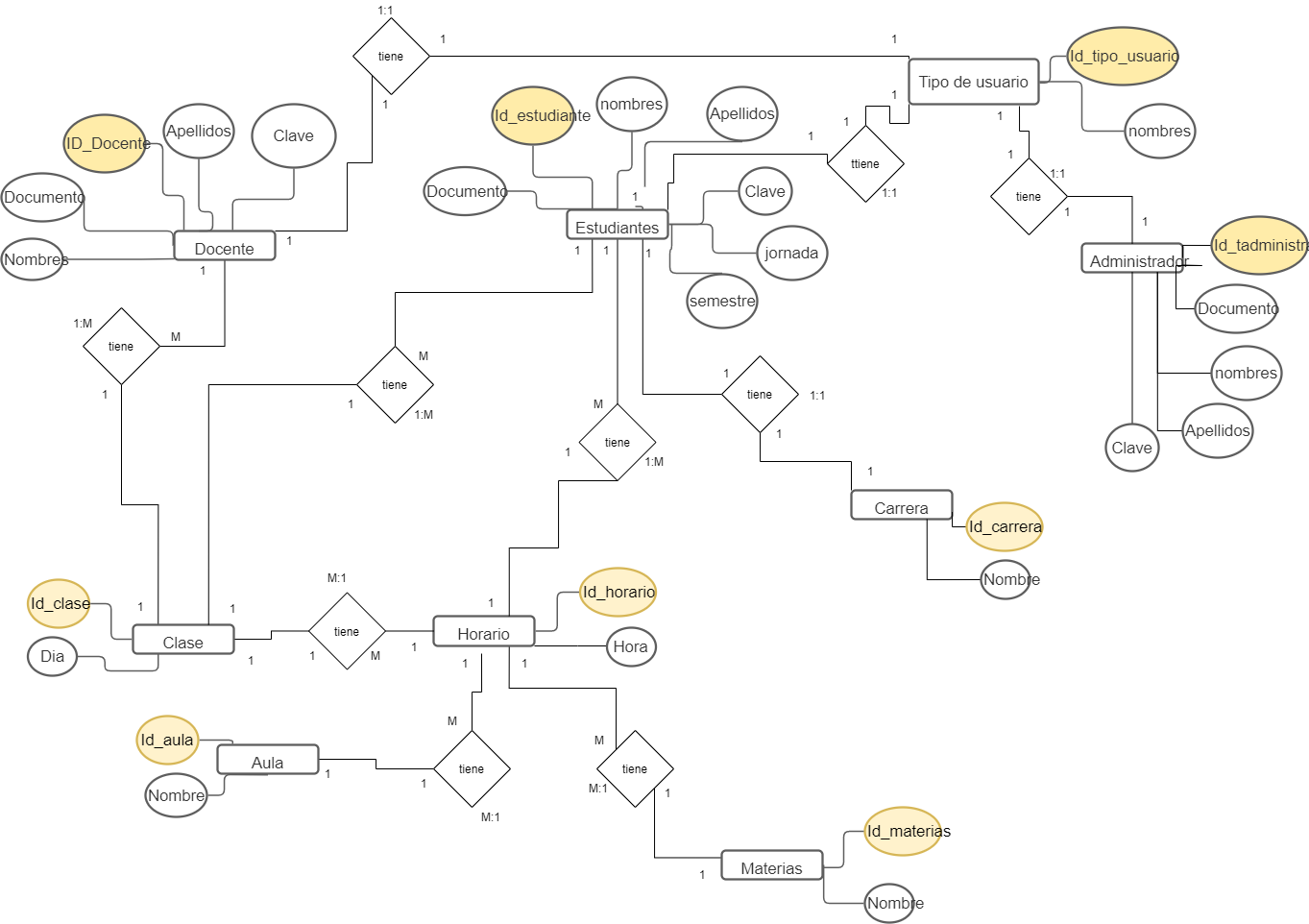
*Fuente: Elaboración propia*

### Diseño de la base de datos

Según los requisitos de información planteados dentro del diseño de sistema de información, nace el siguiente modelo entidad-relación, donde se muestran las entidades con sus respectivos atributos.

#### Modelo entidad-relación

Ilustración   
Modelo entidad-relación



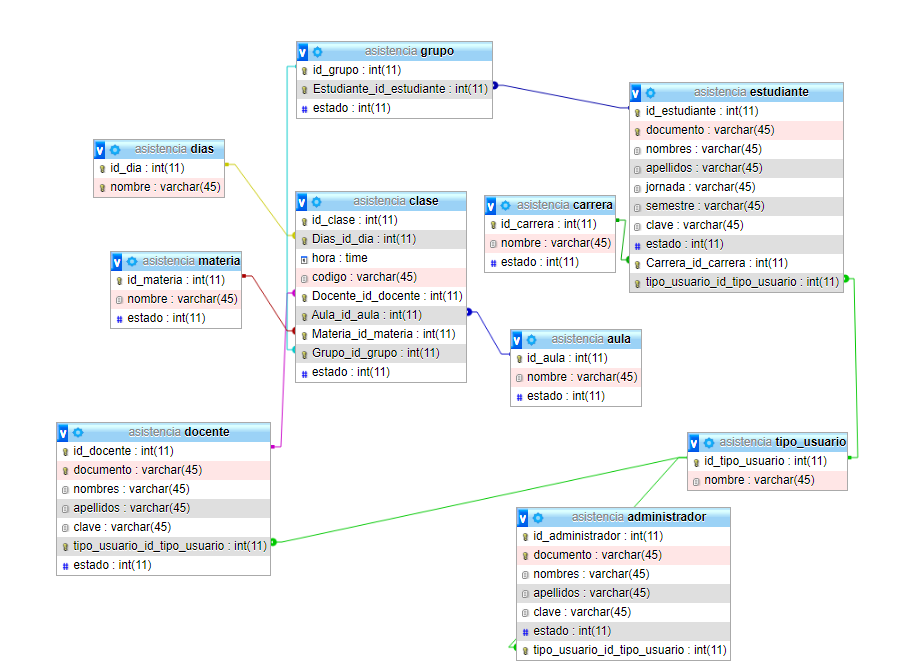
*Fuente: Elaboración propia con ayuda de la herramienta Visual Paradigm*

En este diagrama entidad-relación podemos observar 9 entidades las cuales son: Docente, Estudiantes, Tipo de usuario, administrador, Carrera, Horario, Clase, Aula y Materias.

#### Modelo Físico

Para este modelo se utilizó el programa MySql Workbench el cual es una herramienta visual de diseño de bases de datos que integra desarrollo de software, administración de bases de datos, diseño de bases de datos, gestión y mantenimiento para el sistema de base de datos MySQL.

Ilustración   
Modelo Físico



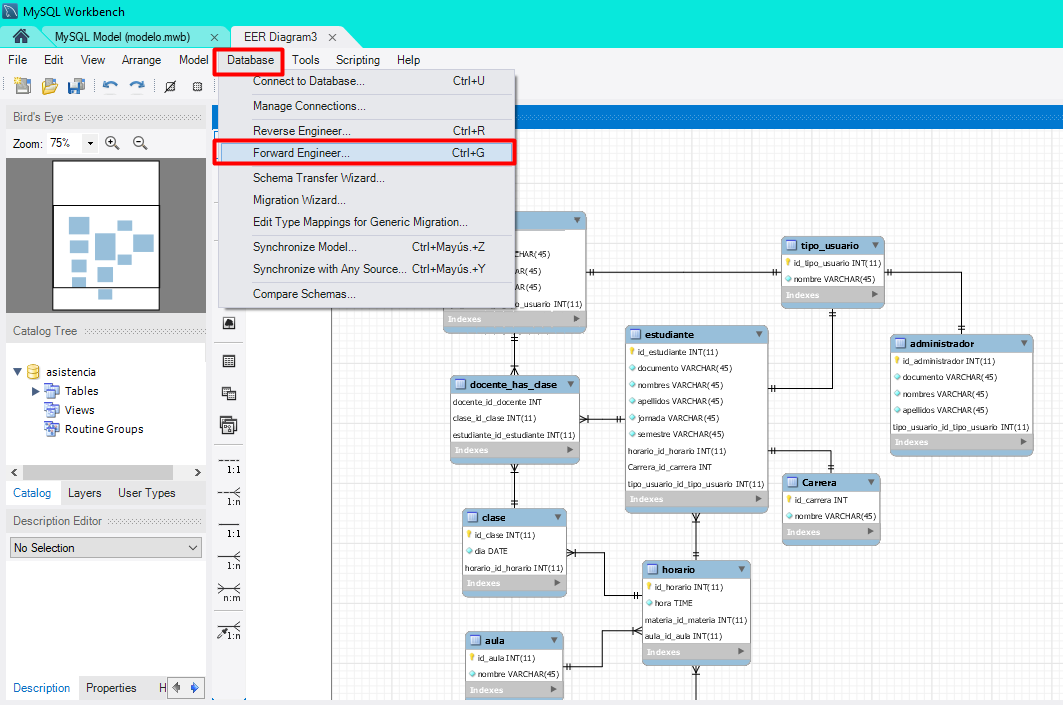
Fuente: Elaboración propia con ayuda de la herramienta MySQL Workbench

En esta etapa se normaliza el modelo entidad relación expuesto anteriormente, dando así como resultado nuevas entidades como se observa en la imagen de una forma más “organizada”.

#### Implementación a phpmyadmin

Cuando se finaliza con el modelo físico, es momento de comprobar si el modelo hecho está formado correctamente para su posterior subida a un servidor MySQL, para ello simplemente debemos dirigirnos a la pestaña database y, a continuación, forward engineer

Ilustración   
MySQL Workbench



Fuente: Elaboración propia con ayuda de la herramienta MySQL Workbench

Al darle nos abrirá una pestaña como en la que se muestra en la siguiente imagen donde simplemente nos pide alguna conexión que debe crearse inicialmente en la pantalla principal del programa; a continuación daremos next hasta ubicarnos en la siguiente pestaña.

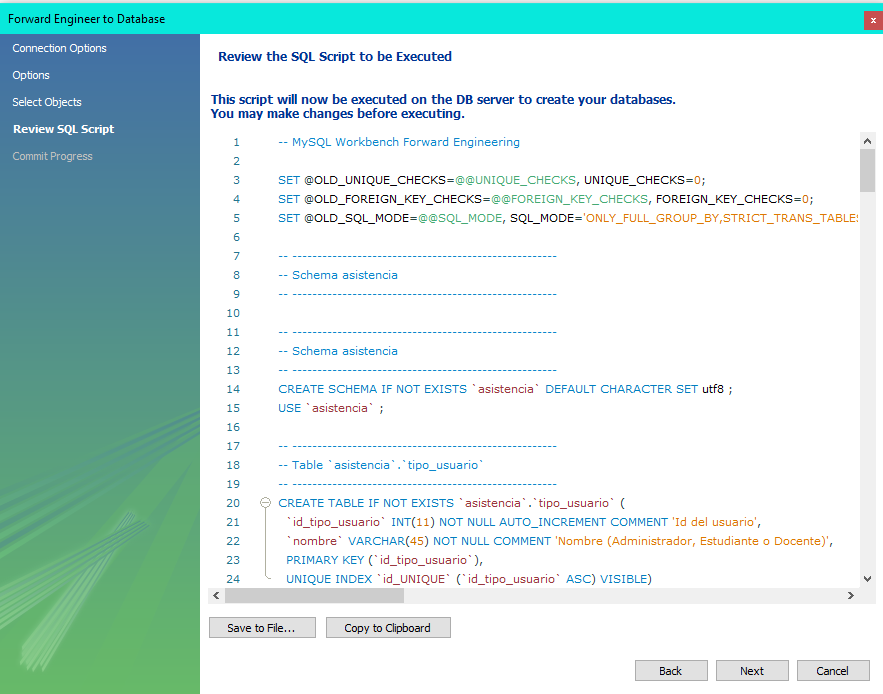
Ilustración   
Forward Engineer



Fuente: Elaboración propia con ayuda de la herramienta MySQL Workbench

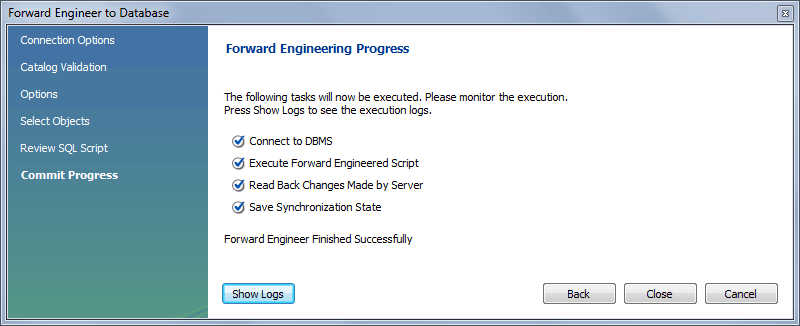
En esta imagen se aprecia el código de todo el esquema del modelo que se creó en un inicio, aquí es donde el programa se encarga de verificar que todo esté en orden, si es así, simplemente saldrá un mensaje de que todo está bien y que el esquema ha sido exportado exitosamente

Ilustración   
Forward Engineer



Fuente: Elaboración propia con ayuda de la herramienta MySQL Workbench

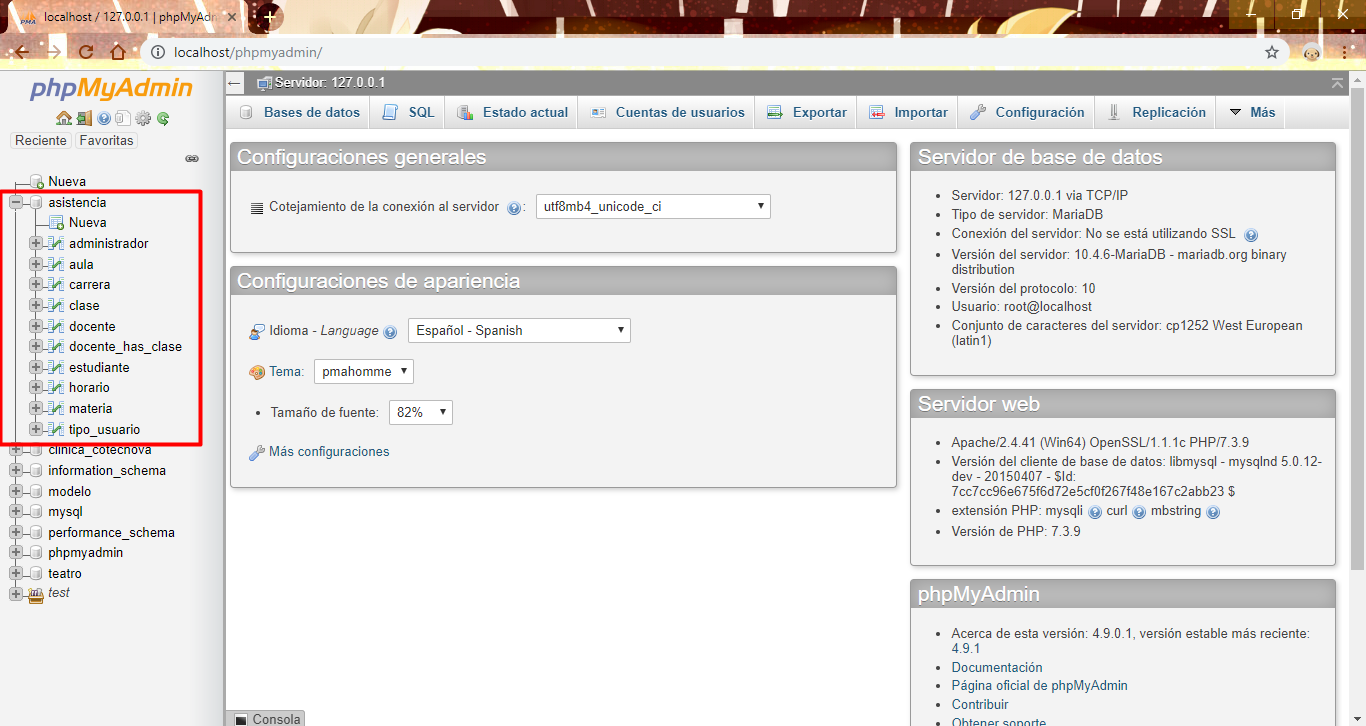
Ilustración   
Mensaje



Fuente: Elaboración propia con ayuda de la herramienta MySQL Workbench

Después de haber hecho lo mostrado anteriormente en las imágenes se pasa a comprobar en algún navegador la dirección localhost/phpmyadmin (para esto debemos tener funcionando un programa externo, en este caso se usó el xampp).

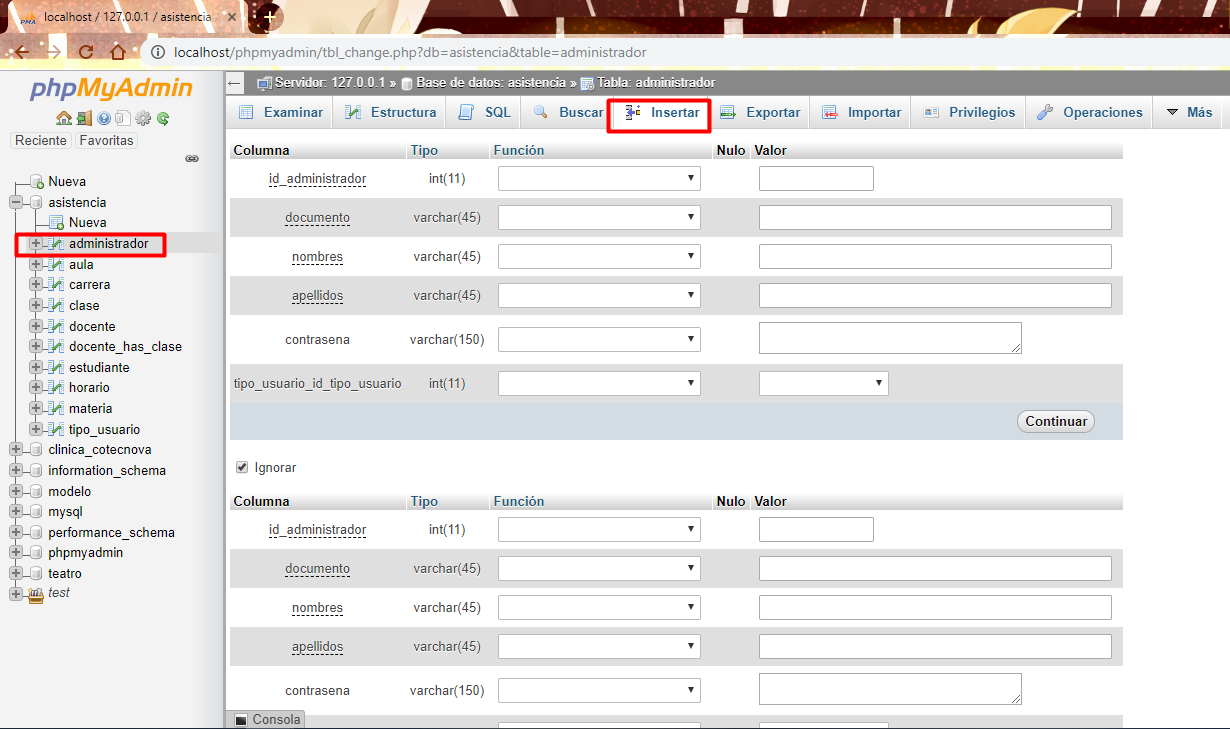
Ilustración   
phpmyadmin



Fuente: Elaboración propia

En la imagen puede observarse que en la parte izquierda del navegador se encuentra el esquema cargado que hicimos anteriormente en el programa, este ahora está transformado en una base de datos la cual se encuentra vacía en un principio, pero con la ayuda de phpmyadmin puede llenarse de una forma más ágil y rápida.

Ilustración   
phpmyadmin



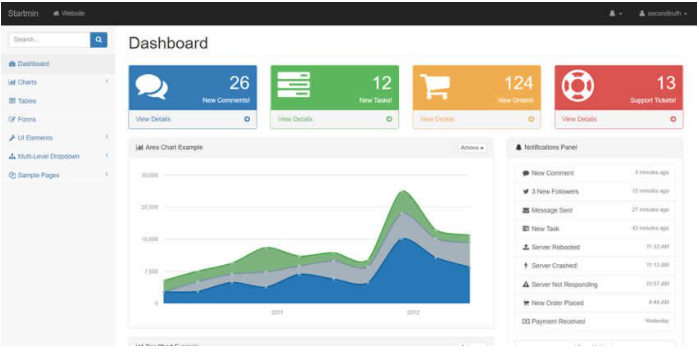
Fuente: Fuente propia

Luego de completar el proceso de llenar cada tabla con sus respectivos datos, está lista para ser integrada y/o usada con una parte gráfica.

#### Plantilla Original del Bootstrap

Para el desarrollo del aplicativo web se usara una plantilla completamente editable y gratis ofrecida por la página (tednologia), la cual tiene un gran repertorio de plantillas editables para los diferentes usuarios.

Ilustración   
Plantilla Original



Fuente: blog.tednologia

## Capítulo 3: Codificación del software

El lenguaje de programación que se implementara el software será PHP versión 7.2.32 ya que este es un lenguaje libre y abierto, su entorno de desarrollo es de rápido y fácil configuración, además este cuenta con un acceso a las bases de datos cuya versión es la 5.0.2, se usara el servidor que nos ofrece phpmyadmin el cual es MariaDB versión 10.4.13, cabe aclarar que este también será usado con las etiquetas HTML 5, el framework front-end bootstrap en su versión 4 y javascript.

### Base de datos Control de Asistencia

A continuación se presenta el modelo de la base datos detallando su composición del código generado.

Tabla   
Código Fuente de la Base de datos

|  |
| --- |
| **Código Fuente de la base de datos de control de asistencia** |
| -- MySQL Workbench Forward Engineering  SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;  SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;  SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';  -- -----------------------------------------------------  -- Schema asistencia  -- -----------------------------------------------------  -- -----------------------------------------------------  -- Schema asistencia  -- -----------------------------------------------------  CREATE SCHEMA IF NOT EXISTS `asistencia` DEFAULT CHARACTER SET utf8 ;  USE `asistencia` ;  -- -----------------------------------------------------  -- Table `asistencia`.`Tipo\_usuario`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Tipo\_usuario` (  `id\_tipo\_usuario` INT(11) NOT NULL AUTO\_INCREMENT COMMENT 'Id del usuario',  `nombre` VARCHAR(45) NOT NULL COMMENT 'Nombre (Administrador, Estudiante o Docente)',  PRIMARY KEY (`id\_tipo\_usuario`),  UNIQUE INDEX `id\_UNIQUE` (`id\_tipo\_usuario` ASC) )  ENGINE = InnoDB  AUTO\_INCREMENT = 7  DEFAULT CHARACTER SET = utf8;  -- -----------------------------------------------------  -- Table `asistencia`.`Administrador`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Administrador` (  `id\_administrador` INT(11) NOT NULL AUTO\_INCREMENT COMMENT 'Identificador del administrador',  `documento` VARCHAR(45) NOT NULL COMMENT 'Documento del administrador',  `nombres` VARCHAR(45) NOT NULL COMMENT 'Nombres del administrador',  `apellidos` VARCHAR(45) NOT NULL COMMENT 'Apellidos del administrador',  `clave` VARCHAR(45) NOT NULL,  `estado` INT(11) NOT NULL,  `tipo\_usuario\_id\_tipo\_usuario` INT(11) NOT NULL,  PRIMARY KEY (`id\_administrador`, `tipo\_usuario\_id\_tipo\_usuario`),  UNIQUE INDEX `documento\_UNIQUE` (`documento` ASC) ,  UNIQUE INDEX `id\_UNIQUE` (`id\_administrador` ASC) ,  INDEX `fk\_administrador\_tipo\_usuario1\_idx` (`tipo\_usuario\_id\_tipo\_usuario` ASC) ,  CONSTRAINT `fk\_administrador\_tipo\_usuario1`  FOREIGN KEY (`tipo\_usuario\_id\_tipo\_usuario`)  REFERENCES `asistencia`.`Tipo\_usuario` (`id\_tipo\_usuario`)  ON DELETE RESTRICT  ON UPDATE CASCADE)  ENGINE = InnoDB  AUTO\_INCREMENT = 4  DEFAULT CHARACTER SET = utf8;  -- -----------------------------------------------------  -- Table `asistencia`.`Aula`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Aula` (  `id\_aula` INT(11) NOT NULL AUTO\_INCREMENT COMMENT 'Id del aula',  `nombre` VARCHAR(45) NOT NULL COMMENT 'Nombre del aula',  `estado` INT(11) NOT NULL,  PRIMARY KEY (`id\_aula`),  UNIQUE INDEX `id\_UNIQUE` (`id\_aula` ASC) )  ENGINE = InnoDB  AUTO\_INCREMENT = 6  DEFAULT CHARACTER SET = utf8;  -- -----------------------------------------------------  -- Table `asistencia`.`Docente`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Docente` (  `id\_docente` INT(11) NOT NULL AUTO\_INCREMENT COMMENT 'Identificador del docente',  `documento` VARCHAR(45) NOT NULL,  `nombres` VARCHAR(45) NOT NULL COMMENT 'Nombres del docente\n\n',  `apellidos` VARCHAR(45) NOT NULL,  `clave` VARCHAR(45) NOT NULL,  `tipo\_usuario\_id\_tipo\_usuario` INT(11) NOT NULL,  `estado` INT NOT NULL,  PRIMARY KEY (`id\_docente`, `tipo\_usuario\_id\_tipo\_usuario`),  UNIQUE INDEX `id\_docente\_UNIQUE` (`id\_docente` ASC) ,  INDEX `fk\_docente\_tipo\_usuario1\_idx` (`tipo\_usuario\_id\_tipo\_usuario` ASC) ,  UNIQUE INDEX `documento\_UNIQUE` (`documento` ASC) ,  CONSTRAINT `fk\_docente\_tipo\_usuario1`  FOREIGN KEY (`tipo\_usuario\_id\_tipo\_usuario`)  REFERENCES `asistencia`.`Tipo\_usuario` (`id\_tipo\_usuario`)  ON DELETE RESTRICT  ON UPDATE CASCADE)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table `asistencia`.`Materia`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Materia` (  `id\_materia` INT(11) NOT NULL AUTO\_INCREMENT COMMENT 'Id de la materia',  `nombre` VARCHAR(45) NOT NULL COMMENT 'Nombre de la materia',  `estado` INT(11) NOT NULL,  PRIMARY KEY (`id\_materia`),  UNIQUE INDEX `id\_UNIQUE` (`id\_materia` ASC) )  ENGINE = InnoDB  AUTO\_INCREMENT = 6  DEFAULT CHARACTER SET = utf8;  -- -----------------------------------------------------  -- Table `asistencia`.`Carrera`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Carrera` (  `id\_carrera` INT NOT NULL AUTO\_INCREMENT,  `nombre` VARCHAR(45) NOT NULL,  `estado` INT(11) NOT NULL,  PRIMARY KEY (`id\_carrera`),  UNIQUE INDEX `id\_carrera\_UNIQUE` (`id\_carrera` ASC) )  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table `asistencia`.`Estudiante`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Estudiante` (  `id\_estudiante` INT(11) NOT NULL AUTO\_INCREMENT COMMENT 'Identificador del estudiante',  `documento` VARCHAR(45) NOT NULL COMMENT 'Documento del estudiante',  `nombres` VARCHAR(45) NOT NULL COMMENT 'Nombres del estudiante',  `apellidos` VARCHAR(45) NOT NULL COMMENT 'Apellidos del estudiante',  `jornada` VARCHAR(45) NOT NULL COMMENT 'Jornada del estudiante',  `semestre` VARCHAR(45) NOT NULL,  `clave` VARCHAR(45) NOT NULL,  `estado` INT(11) NOT NULL,  `Carrera\_id\_carrera` INT NOT NULL,  `tipo\_usuario\_id\_tipo\_usuario` INT(11) NOT NULL,  PRIMARY KEY (`id\_estudiante`, `Carrera\_id\_carrera`, `tipo\_usuario\_id\_tipo\_usuario`),  UNIQUE INDEX `documento\_UNIQUE` (`documento` ASC) ,  UNIQUE INDEX `id\_UNIQUE` (`id\_estudiante` ASC) ,  INDEX `fk\_estudiante\_Carrera1\_idx` (`Carrera\_id\_carrera` ASC) ,  INDEX `fk\_estudiante\_tipo\_usuario1\_idx` (`tipo\_usuario\_id\_tipo\_usuario` ASC) ,  CONSTRAINT `fk\_estudiante\_Carrera1`  FOREIGN KEY (`Carrera\_id\_carrera`)  REFERENCES `asistencia`.`Carrera` (`id\_carrera`)  ON DELETE RESTRICT  ON UPDATE CASCADE,  CONSTRAINT `fk\_estudiante\_tipo\_usuario1`  FOREIGN KEY (`tipo\_usuario\_id\_tipo\_usuario`)  REFERENCES `asistencia`.`Tipo\_usuario` (`id\_tipo\_usuario`)  ON DELETE RESTRICT  ON UPDATE CASCADE)  ENGINE = InnoDB  AUTO\_INCREMENT = 2  DEFAULT CHARACTER SET = utf8;  -- -----------------------------------------------------  -- Table `asistencia`.`Grupo`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Grupo` (  `id\_grupo` INT(11) NOT NULL AUTO\_INCREMENT,  `Estudiante\_id\_estudiante` INT(11) NOT NULL,  `estado` INT(11) NOT NULL,  PRIMARY KEY (`id\_grupo`, `Estudiante\_id\_estudiante`),  INDEX `fk\_Grupo\_Estudiante1\_idx` (`Estudiante\_id\_estudiante` ASC) ,  CONSTRAINT `fk\_Grupo\_Estudiante1`  FOREIGN KEY (`Estudiante\_id\_estudiante`)  REFERENCES `asistencia`.`Estudiante` (`id\_estudiante`)  ON DELETE RESTRICT  ON UPDATE CASCADE)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table `asistencia`.`Dias`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Dias` (  `id\_dia` INT(11) NOT NULL AUTO\_INCREMENT,  `nombre` VARCHAR(45) NOT NULL,  PRIMARY KEY (`id\_dia`),  UNIQUE INDEX `id\_dia\_UNIQUE` (`id\_dia` ASC) ,  UNIQUE INDEX `nombre\_UNIQUE` (`nombre` ASC) )  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table `asistencia`.`Clase`  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS `asistencia`.`Clase` (  `id\_clase` INT(11) NOT NULL AUTO\_INCREMENT COMMENT 'Id de la clase',  `Dias\_id\_dia` INT(11) NOT NULL,  `hora` TIME NOT NULL,  `codigo` VARCHAR(45) NOT NULL,  `Docente\_id\_docente` INT(11) NOT NULL,  `Aula\_id\_aula` INT(11) NOT NULL,  `Materia\_id\_materia` INT(11) NOT NULL,  `Grupo\_id\_grupo` INT(11) NOT NULL,  PRIMARY KEY (`id\_clase`, `Dias\_id\_dia`, `Docente\_id\_docente`, `Aula\_id\_aula`, `Materia\_id\_materia`, `Grupo\_id\_grupo`),  UNIQUE INDEX `id\_UNIQUE` (`id\_clase` ASC) ,  INDEX `fk\_Clase\_Docente1\_idx` (`Docente\_id\_docente` ASC) ,  INDEX `fk\_Clase\_Aula1\_idx` (`Aula\_id\_aula` ASC) ,  INDEX `fk\_Clase\_Materia1\_idx` (`Materia\_id\_materia` ASC) ,  INDEX `fk\_Clase\_Grupo1\_idx` (`Grupo\_id\_grupo` ASC) ,  INDEX `fk\_Clase\_Dias1\_idx` (`Dias\_id\_dia` ASC) ,  CONSTRAINT `fk\_Clase\_Docente1`  FOREIGN KEY (`Docente\_id\_docente`)  REFERENCES `asistencia`.`Docente` (`id\_docente`)  ON DELETE RESTRICT  ON UPDATE CASCADE,  CONSTRAINT `fk\_Clase\_Aula1`  FOREIGN KEY (`Aula\_id\_aula`)  REFERENCES `asistencia`.`Aula` (`id\_aula`)  ON DELETE RESTRICT  ON UPDATE CASCADE,  CONSTRAINT `fk\_Clase\_Materia1`  FOREIGN KEY (`Materia\_id\_materia`)  REFERENCES `asistencia`.`Materia` (`id\_materia`)  ON DELETE RESTRICT  ON UPDATE CASCADE,  CONSTRAINT `fk\_Clase\_Grupo1`  FOREIGN KEY (`Grupo\_id\_grupo`)  REFERENCES `asistencia`.`Grupo` (`id\_grupo`)  ON DELETE RESTRICT  ON UPDATE CASCADE,  CONSTRAINT `fk\_Clase\_Dias1`  FOREIGN KEY (`Dias\_id\_dia`)  REFERENCES `asistencia`.`Dias` (`id\_dia`)  ON DELETE RESTRICT  ON UPDATE CASCADE)  ENGINE = InnoDB  AUTO\_INCREMENT = 3  DEFAULT CHARACTER SET = utf8;  SET SQL\_MODE=@OLD\_SQL\_MODE;  SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;  SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS; |

Fuente: Fuente propia Gracias a la Herramienta PHPMYADMIN

### Codificación del software

Para realizar la vinculación del software con su base de datos tenemos que crear la respectiva conexión, dicha conexión se encuentra en la carpeta modelo dentro de los archivos editables del software.

Ilustración   
Conexión a la base de datos en forma local



Fuente: Fuente propia con la ayuda de la herramienta sublime text

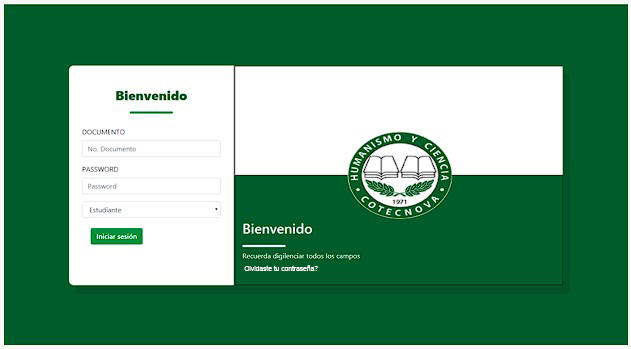
Una vez hecho la conexión procederemos a crear la interfaz gráfica del login, el cual dará inicio al software, este se creara a partir de etiquetas HTML, una vez que esté listo iremos al navegador y escribiéremos en la URL lo siguiente: localhost/nombre\_carpeta/login.php.

Ilustración   
Etiquetas HTML del login



Fuente: Fuente propia con la ayuda de la herramienta sublime text

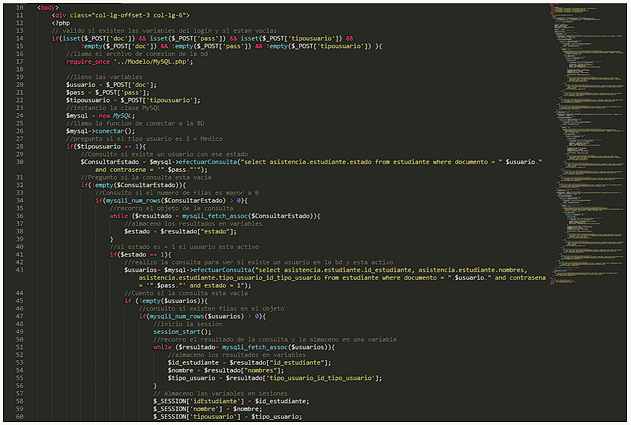
Ilustración   
Vista grafica desde el navegador



Fuente: Fuente propia

El login es controlado por medio de un controlador alojado en la carpeta Controladores, dentro de esta carpeta se encuentra el archivo loginValidacion.php el cual está conformado por las diferentes conexiones y consultas de la base de datos, el resultado de estas consultas se validara con la información presentada en los inputs del formulario, si son iguales se iniciara la sesión del usuario, el usuarios se selecciona en el select del formulario, actualmente son tres tipos de usuarios (Administrador, Docente, estudiante).

Ilustración   
Validación del login.php



Fuente: Fuente propia con la ayuda de la herramienta sublime text

Una vez realizado el proceso de login cada usuario ingresara a su respectivo index, en el caso del estudiante en su index tendrá los módulos de consultar horario y validar asistencia, para el docente tendrá consultar clase, registrar clase y enviar material, el administrador dispondrá de las diferentes gestiones de la información almacenada en la base de datos en los módulos de gestionar estudiantes, docentes, aulas, materias, horario y carreras.

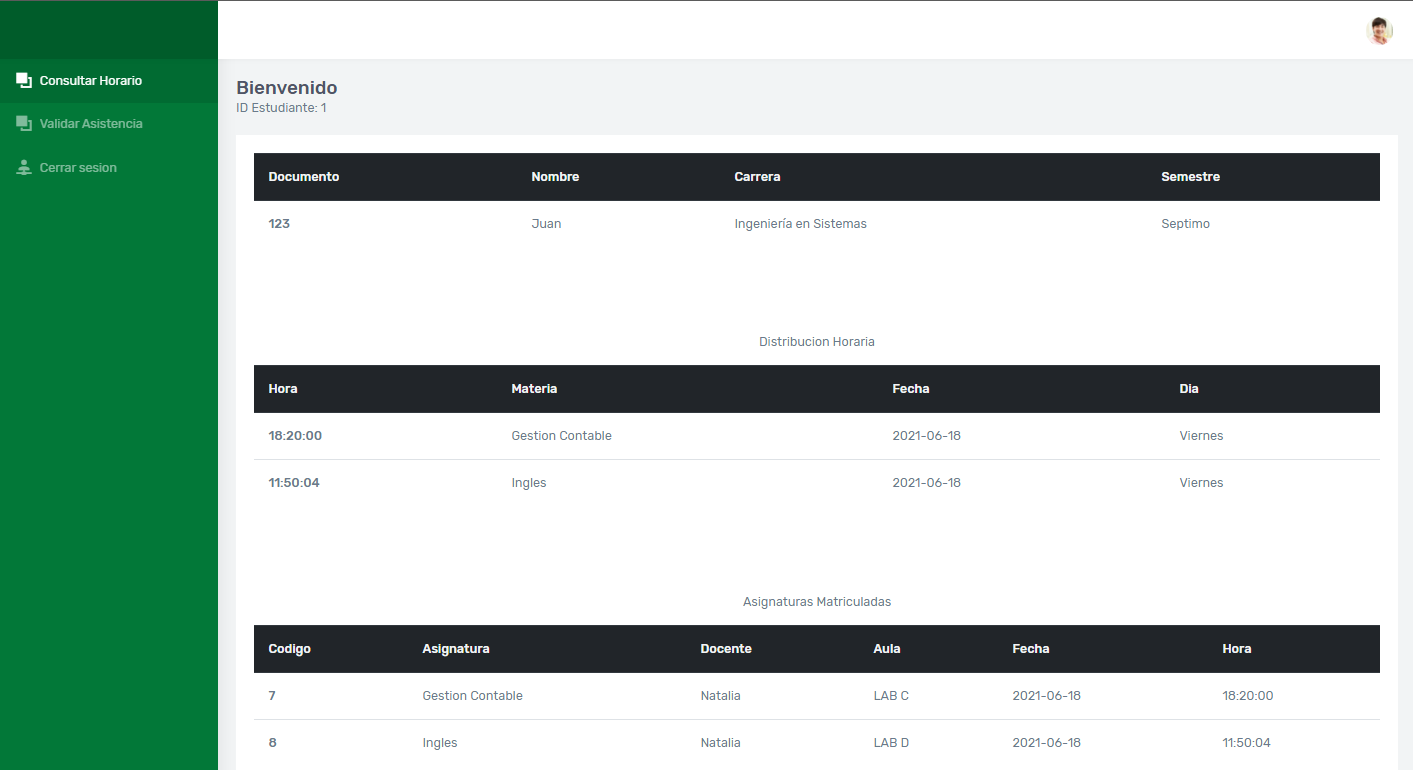
Ilustración   
Index del estudiante



Fuente: Fuente propia

En el index del estudiante podrá acceder a los módulos de consultar horario y validar la asistencia, junto con esto se mostrar en la vista principal la información del estudiante para validar su sesión.

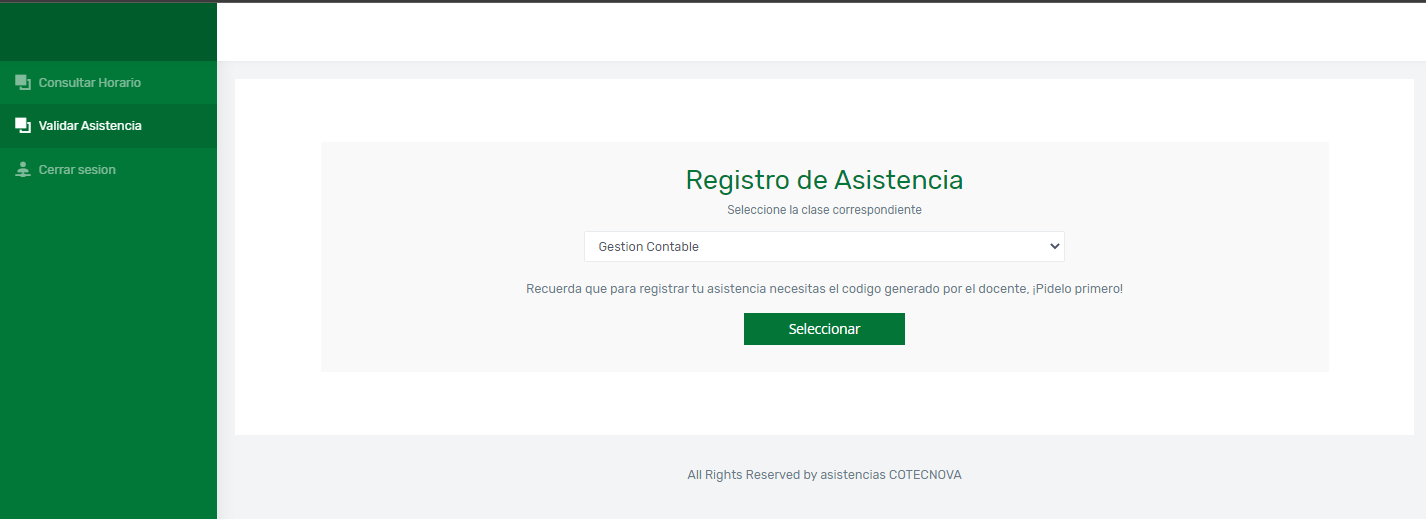
Ilustración   
Consulta del horario del estudiante



Fuente: Fuente propia

Aquí el estudiante podrá encontrar nuevamente su información personal, además de esto podrá ver su distribución del horario y las materias que este tiene asignadas, esta información es traída gracias a los controladores en los cuales se encuentran las diferentes consultas de la Base de Datos.

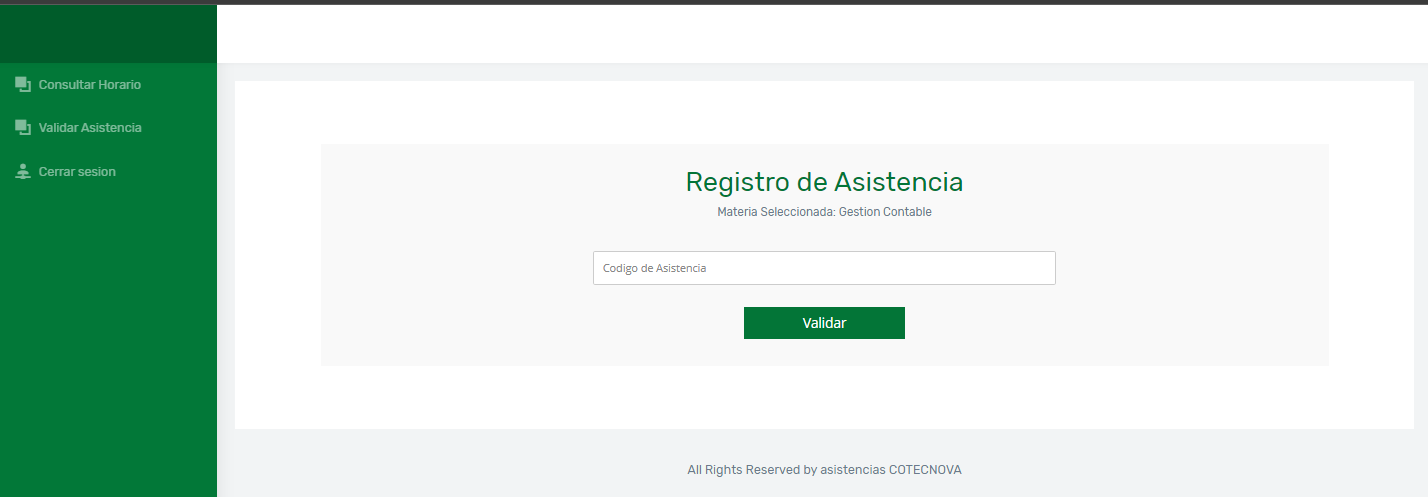
Ilustración   
Validar Asistencia Estudiante



Fuente: Fuente propia

El estudiante deberá seleccionar la materia a la cual desea realizar su asistencia, este se selecciona por medio de un select el cual está conectado a las BD por medio de las consultas internas del código, una vez seleccionada la materia dará clic en el botón de seleccionar, este mostrar un nuevo formulario donde deberá ingresar el código de la clase.

Ilustración   
Validar Asistencia Estudiante



Fuente: Fuente propia

El estudiante deberá ingresar el código suministrado por el docente para así validar su asistencia, esta será validada por medio del controlador donde estará la respectiva consulta con al BD.

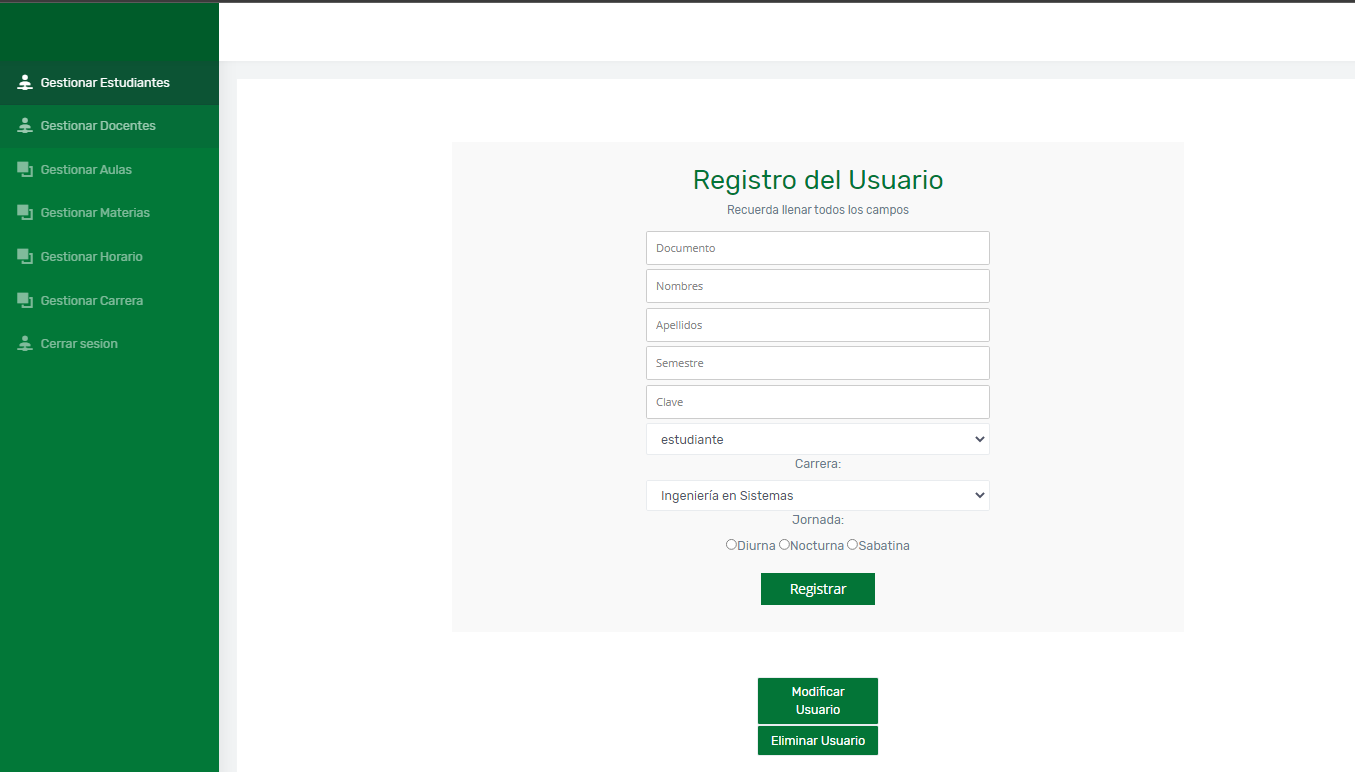
Ilustración   
Index Administrador



Fuente: Fuente propia

Una vez el administrador este en su respectivo index contara con los módulos de gestionar estudiantes, docentes, aulas, materias, horario y carrera.

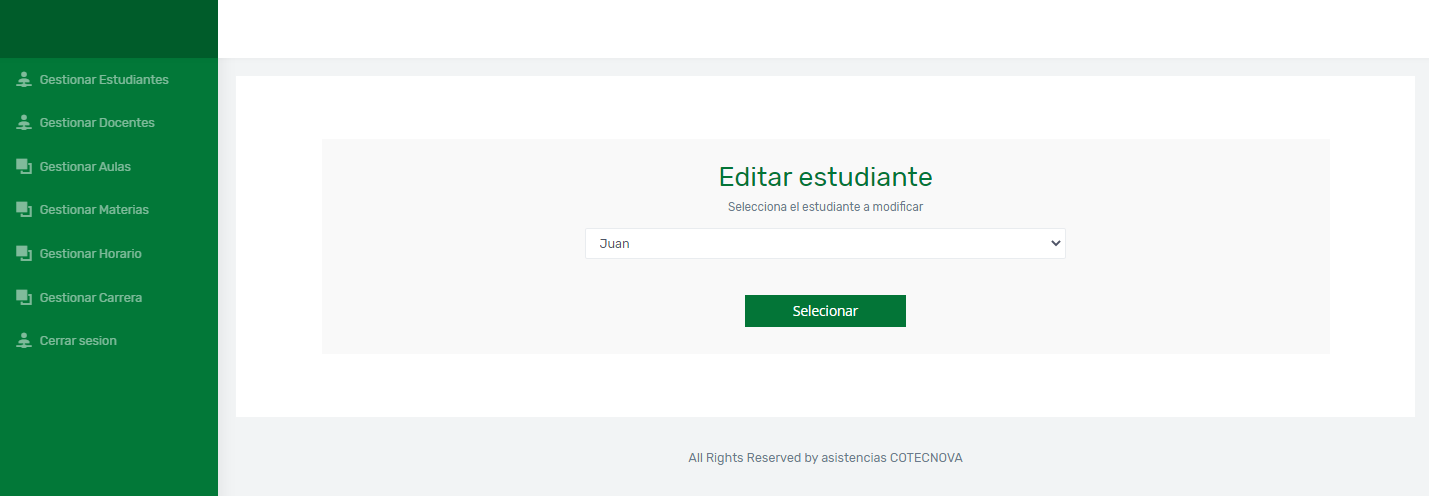
Ilustración   
Gestionar Docente



Fuente: Fuente propia

El administrador podrá ingresar a un nuevo estudiante o respectivamente modificar o eliminar a uno de los que ya estén existentes, en el caso de registrar a un nuevo estudiante deberá ingresar los datos solicitados del formulario y dar clic sobre el botón de registrar estos datos serán enviados a la BD por medio del controlador.

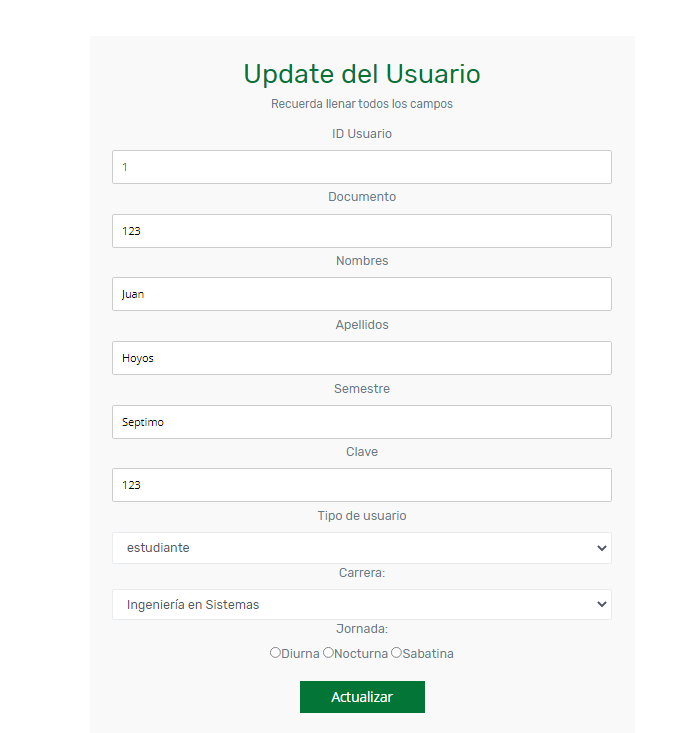
Ilustración   
Editar estudiante



Fuente: Fuente propia

El administrador deberá seleccionar al estudiante que desea modificar por medio del select el cual trae la información de los estudiantes por medio del controlador conectado a la BD, una vez seleccionado deberá dar clic sobre el respectivo botón, este enviara todos los datos del estudiante a un nuevo formulario donde los podrá modificar.

Ilustración   
Modificar Estudiante



Fuente: Fuente propia

Una vez modificado los datos del estudiante deberá dar clic sobre el respectivo botón el cual hará los cambios del estudiante gracias al controlador usado para la respectiva consulta y gestión.

Ilustración   
Eliminar Estudiante



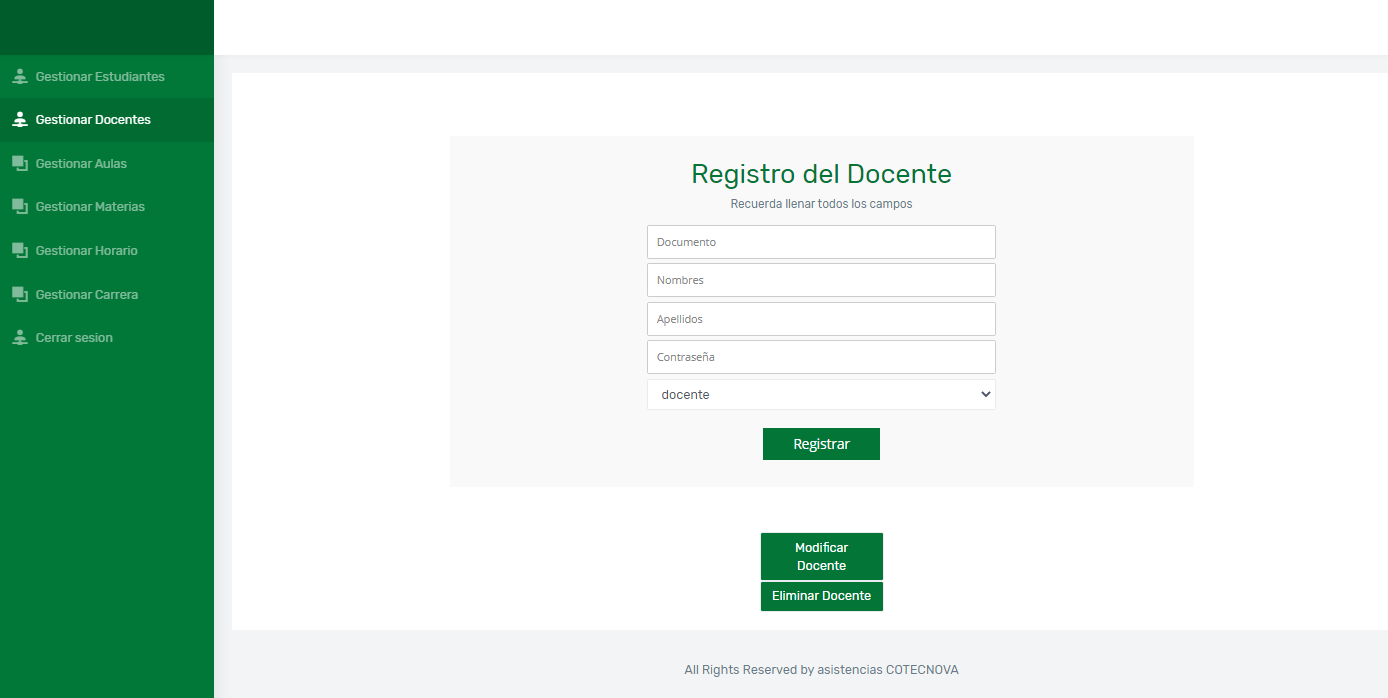
Fuente: Fuente propia

El administrador deberá seleccionar al estudiante que desee eliminar por medio del select suministrado, el cual está conectado a la BD por su respectiva consulta, una vez seleccionado deberá dar clic sobre el respectivo botón, este será enviado a un nuevo controlador el cual hará los respectivos cambios en la BD, estos no se eliminan del todo, sino que estos serán deshabilitados por medio de los estados ubicados en la BD.

#### Aclaración de las gestiones realizadas en los módulos del administrador.

Cabe aclarar que los procedimientos de registro, eliminar y modificar en los módulos del administrador son los mismos para todos, es decir se realiza el mismo proceso que en la gestión de estudiantes.

Ilustración   
Registro nuevo docente



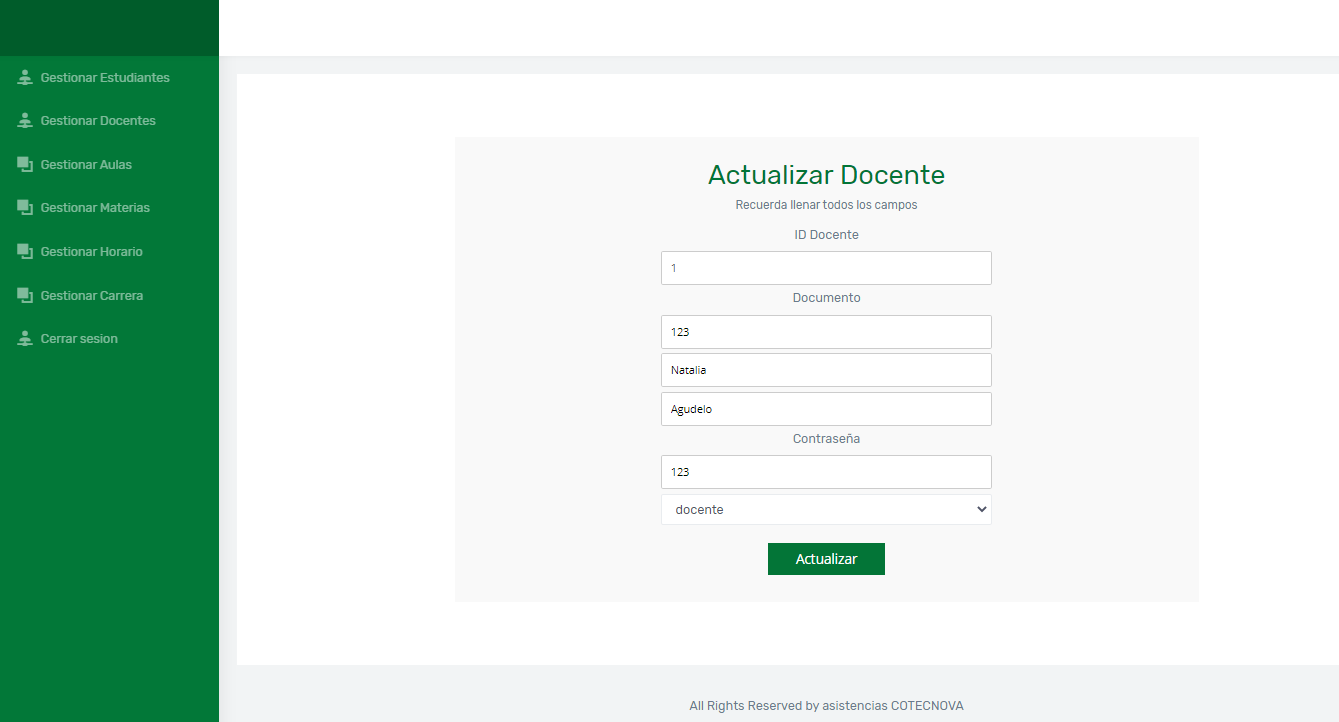
Fuente: Fuente propia

Ilustración   
Modificación Docente



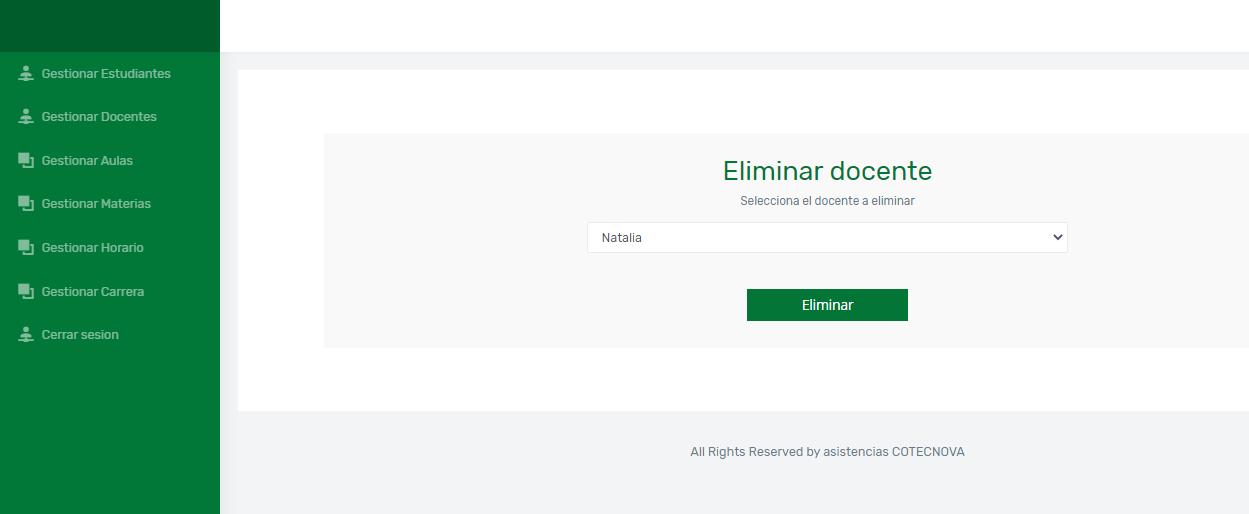
Fuente: Fuente propia

Ilustración   
Modificación del docente



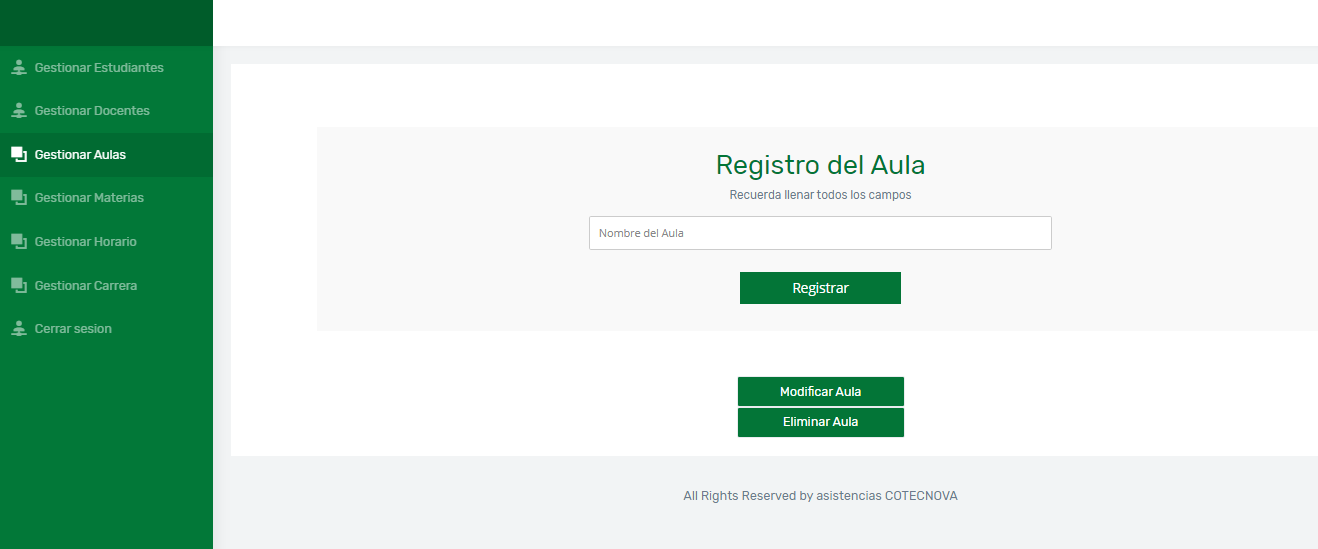
Fuente: Fuente propia

Ilustración   
Eliminar Docente



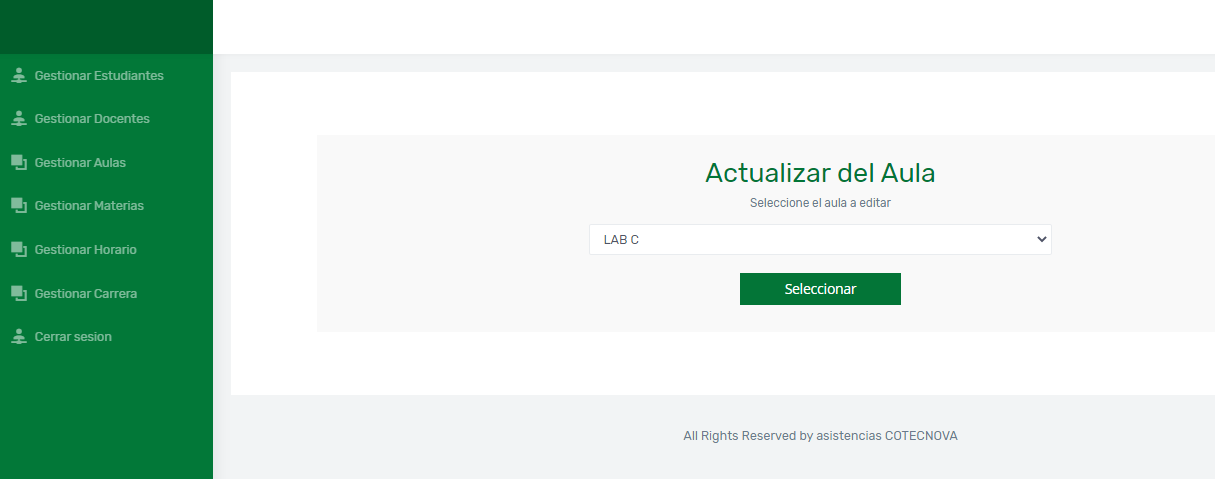
Fuente: Fuente propia

Ilustración   
Registro del aula



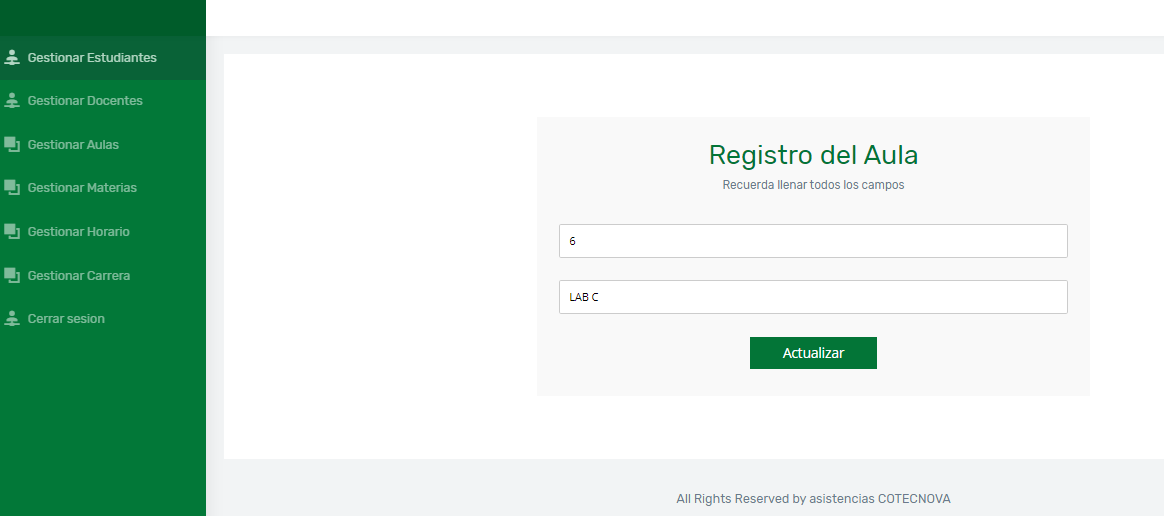
Fuente: Fuente propia

Ilustración   
Modificación Aula



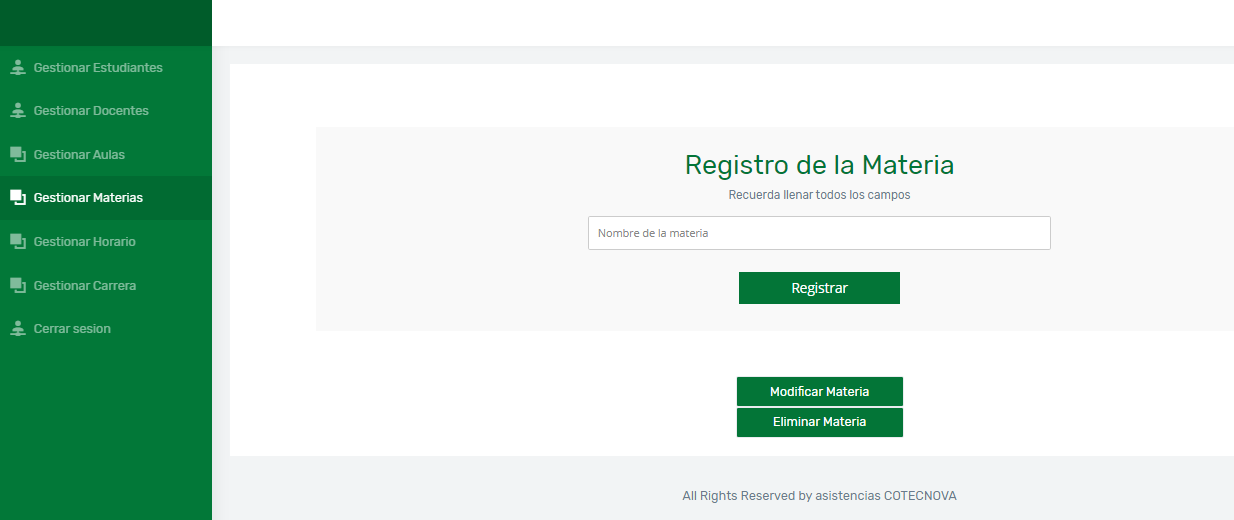
Fuente: Fuente propia

Ilustración   
Modificación Aula



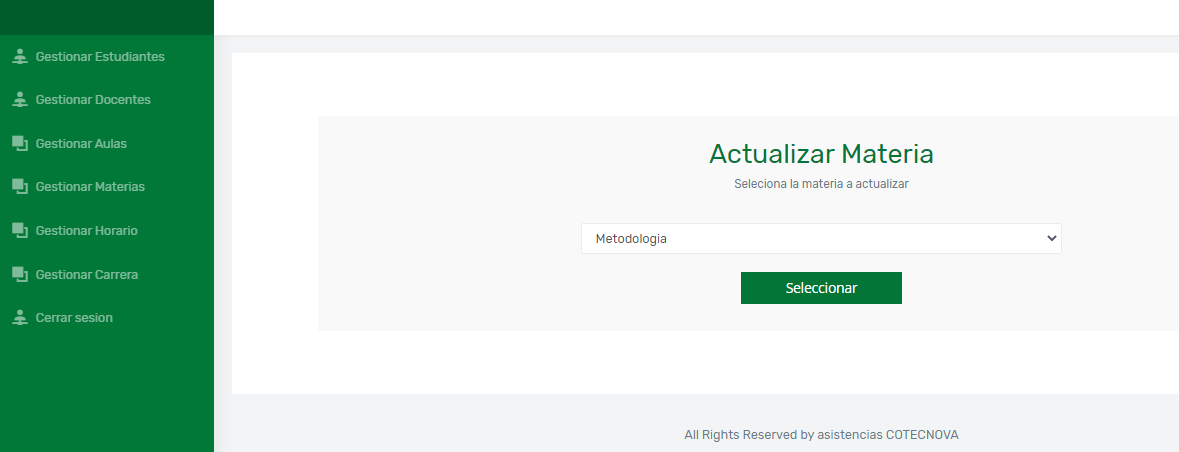
Fuente: Fuente propia

Ilustración   
Registro Materia



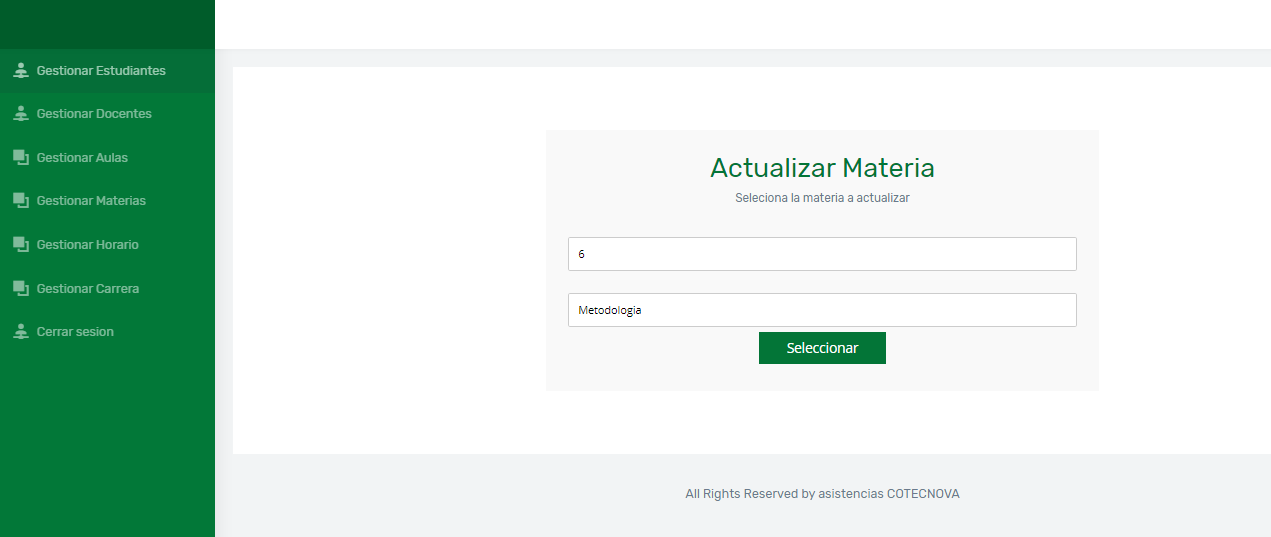
Fuente: Fuente propia

Ilustración   
Modificación Materia



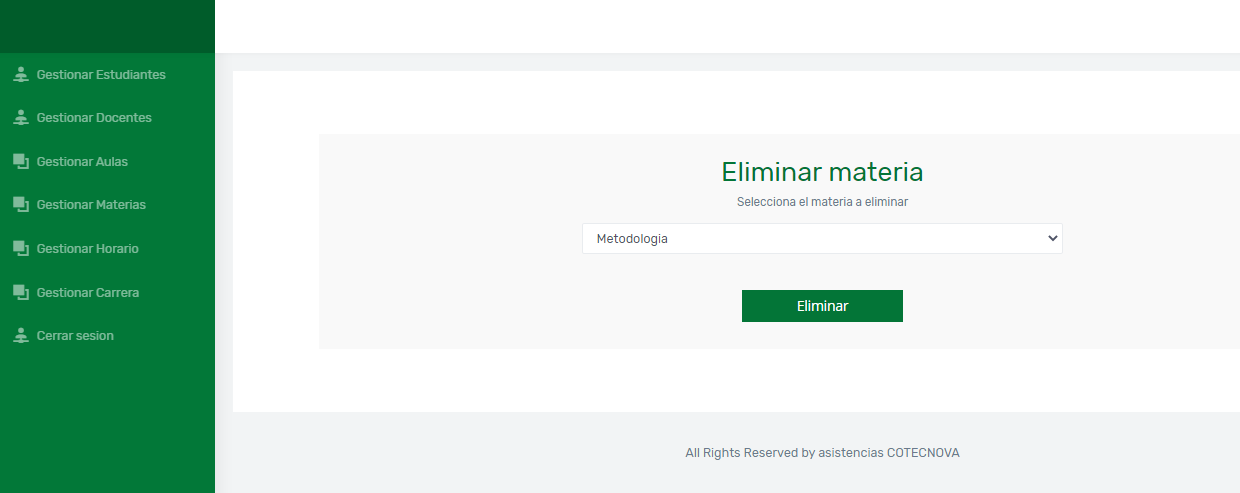
Fuente: Fuente propia

Ilustración   
Modificación Materia



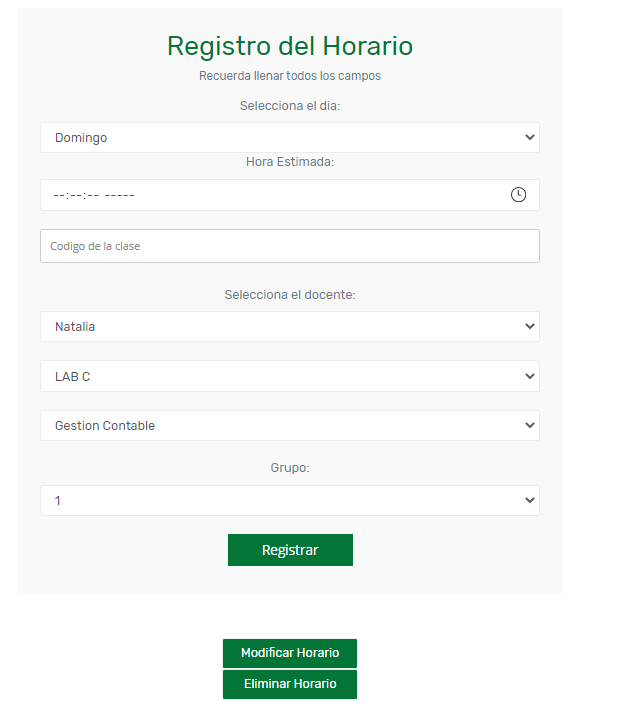
Fuente: Fuente propia

Ilustración   
Eliminar Materia



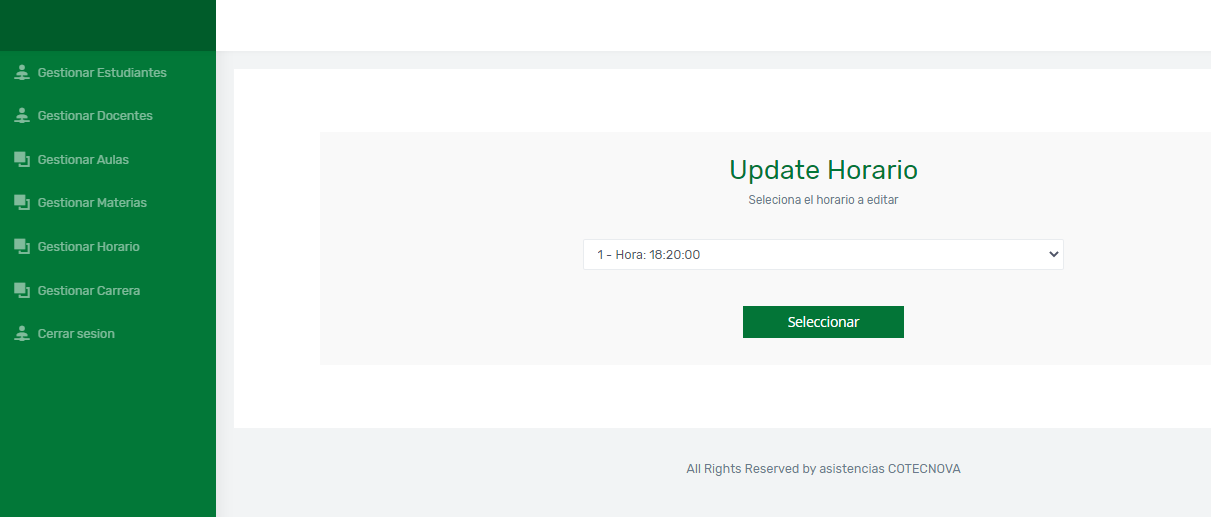
Fuente: Fuente propia

Ilustración   
Registro Horario



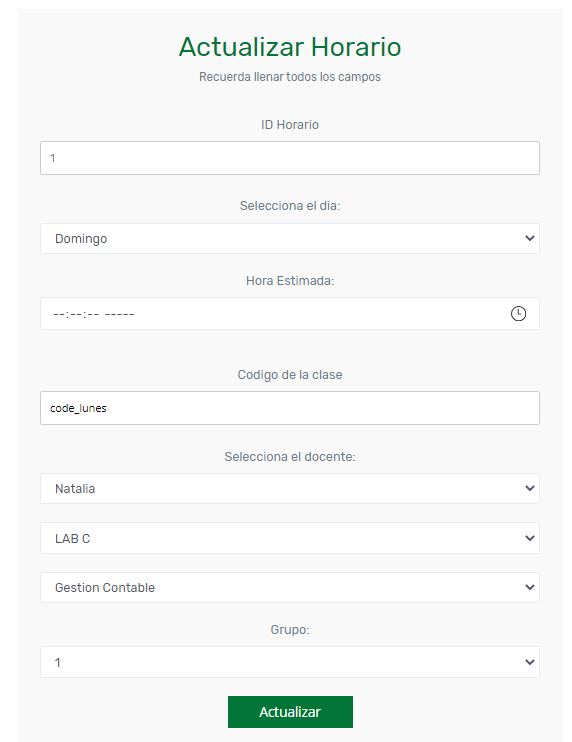
Fuente: Fuente propia

Ilustración   
Modificar Horario



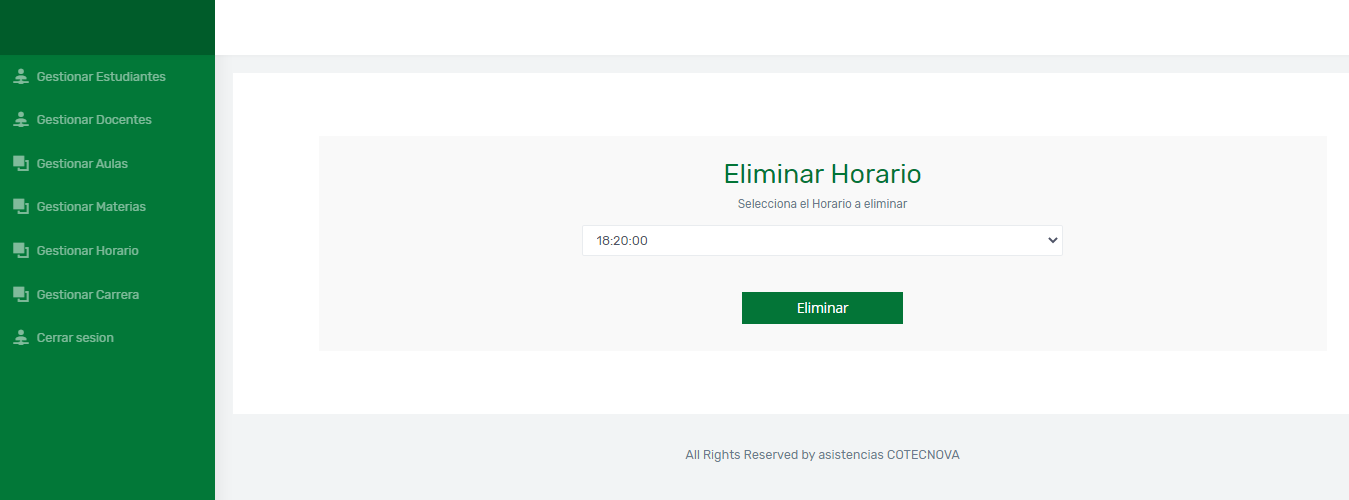
Fuente: Fuente propia

Ilustración   
Modificar Horario



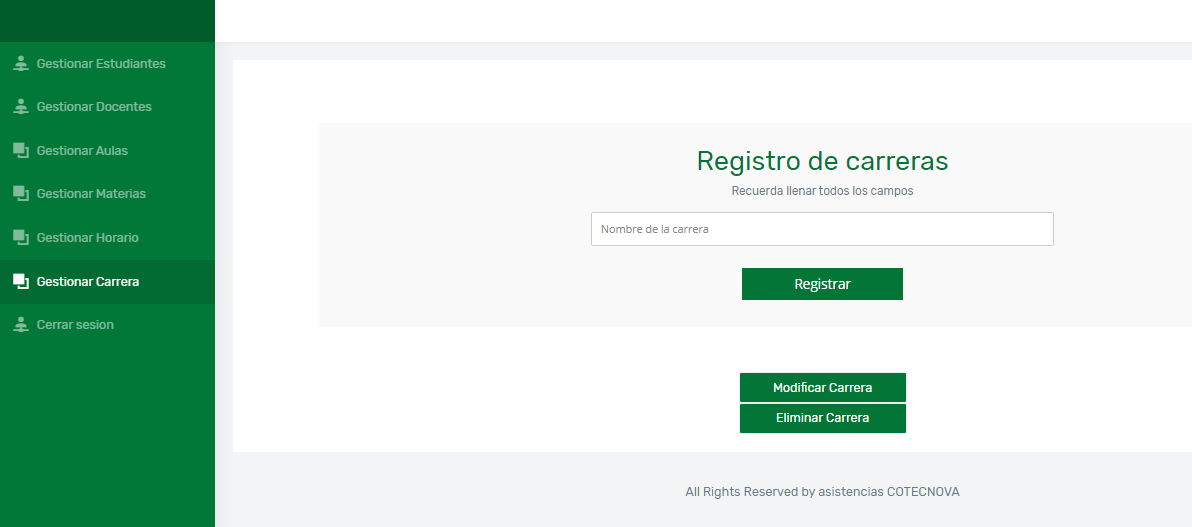
Fuente: Fuente propia

Ilustración   
Eliminar Horario



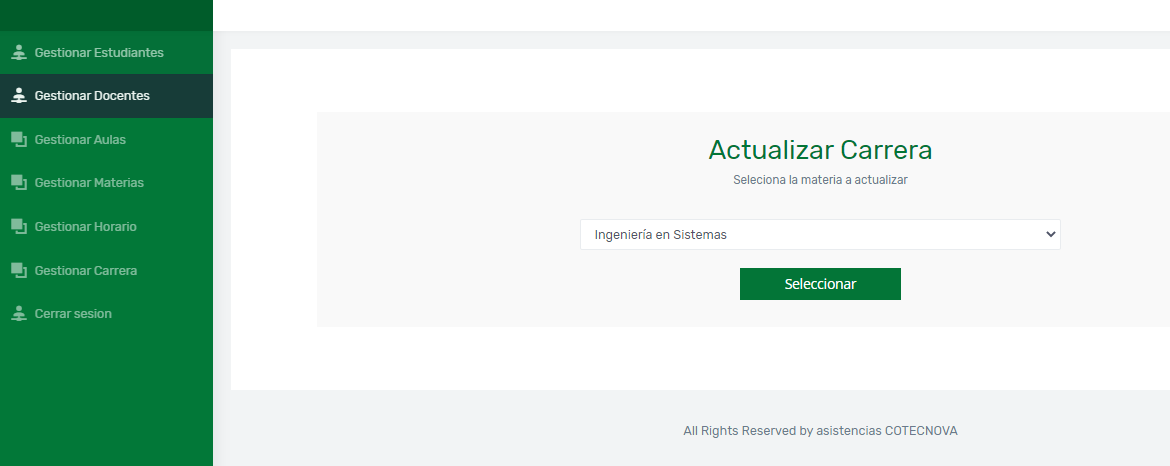
Fuente: Fuente propia

Ilustración   
Registro de Carreras



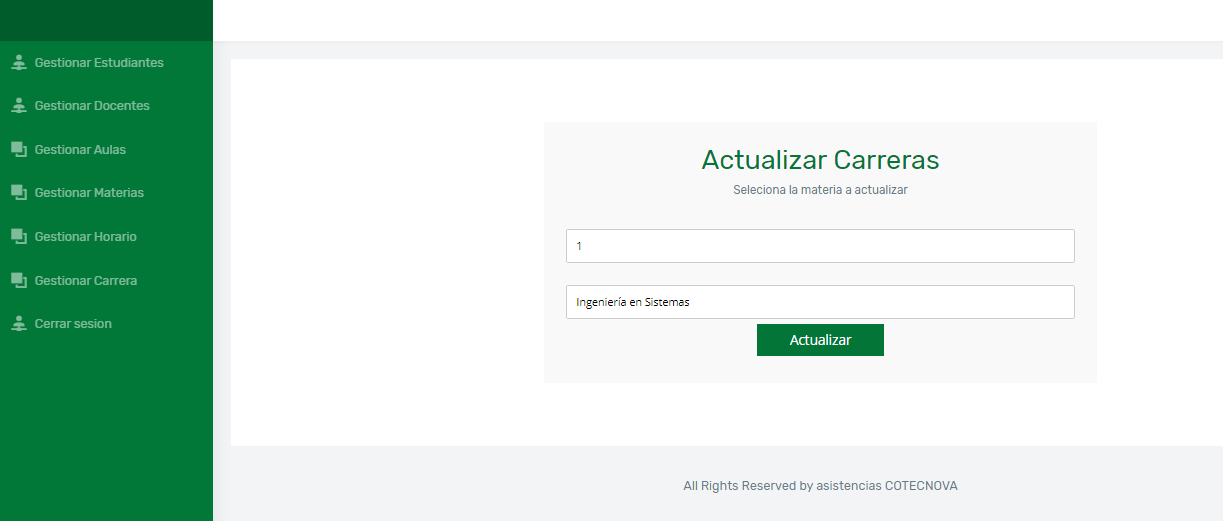
Fuente: Fuente propia

Ilustración   
Actualizar Carrera



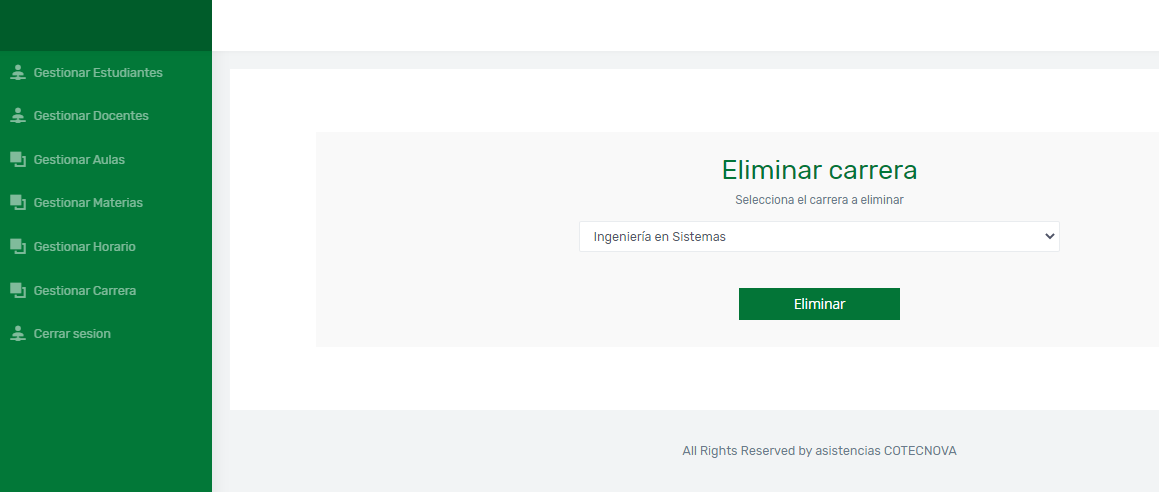
Fuente: Fuente propia

Ilustración   
Actualizar Carrera



Fuente: Fuente propia

Ilustración   
Eliminar Carrera



Fuente: Fuente propia

Ilustración   
Index docente



Fuente: Fuente propia

El docente dispondrá de los módulos de consultar clase, registrar clase y enviar material.

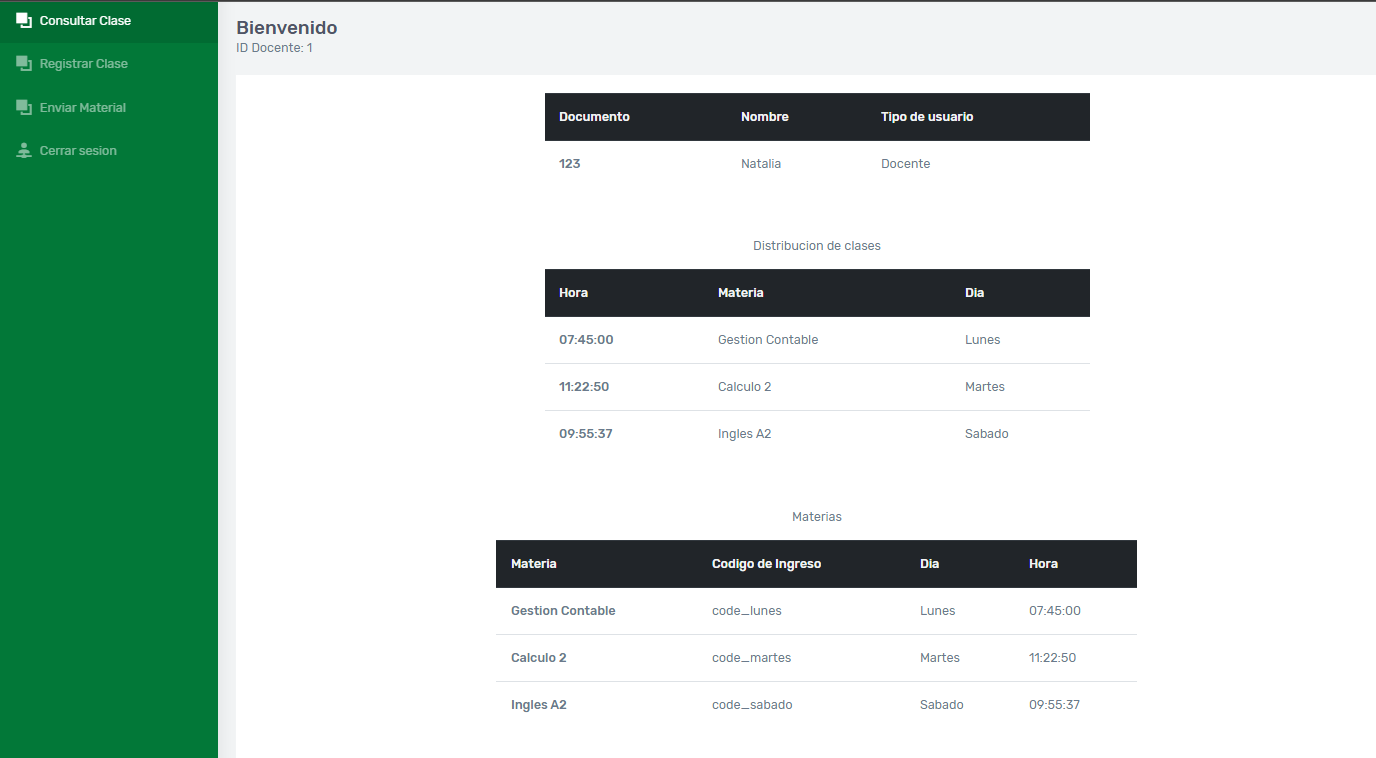
Ilustración   
Envió de Material



Fuente: Fuente propia

El docente deberá diligenciar los datos solicitados en el formulario para así enviar el respectivo correo electrónico, esto gracias al controlador donde se encuentra la configuración del mismo.

Ilustración   
Consultar Clase



Fuente: Fuente propia

En el módulo de consultar clase el docente podrá encontrar 3 tablas diferentes, en la primera tabla encontrara la información personal, en la segunda la distribución de las clases las cuales dará y la tercera son las materias que da el docente, estas consultas son traídas gracias al controlador de la página el cual está conectado a la BD.

# Conclusiones

El objetivo principal de este documento fue abordar el problema a la hora de realizar el control de llamado a lista dentro de la institución universitaria COTECNOVA, cuya solución ayudaría tanto a estudiantes como docentes logrando la optimización de dicho de control.

Así pues, la aportación principal consiste en la implementación de un software web para el control de asistencia y envió de material visto en las clases, en el cual tendría la intervención por parte de los administrativos, docentes y estudiantes, por ello, gracias a los objetivos planteados, se llegaron a las siguientes conclusiones:

* La identificación de requerimientos presentados en el capítulo 1 del esquema temático, se logró determinar los instrumentos de investigación necesarios para el desarrollo del documento, entre estos se encuentran la observación, la entrevista y la herramienta REM, los cuales tuvieron un papel fundamental dentro del documento, ya que por medio de la observación se pudo determinar los tiempos y métodos que utilizan los docentes a la hora de ejercer el proceso de control de asistencia, junto con ello se desarrolló la entrevista a partir de lo observado en la observación, logrando así la realización de una serie de preguntas para determinar las opiniones de los docentes y el desarrollo del proceso del control de asistencia y envío de material, por último, la herramienta REM ayudó a la hora de categorizar los diferentes componentes necesarios para el desarrollo del software recolectando los diferentes requisitos necesarios (funcionales – no funcionales – sistema), junto con los diagramas de casos de uso desarrollados según las problemáticas encontradas durante la investigación, cabe aclarar que dentro de la herramienta REM se tuvieron en cuenta los participantes del proyecto, la descripción del sistema actual y los objetivos del proyecto.
* En el diseño del sistema de información plasmado en el capítulo 2 del esquema temático se logró el desarrollo de los diagramas necesarios para llegar al desarrollo del software y su base de datos, junto a ello, la solución más óptima para la misma, logrando identificar los diferentes procesos para optimizar la problemática planteada. Cabe aclarar que, para el desarrollo del proyecto, se tuvieron en cuenta 3 actores los cuales fueron: los administradores, los docentes y estudiantes, los cuales tienen un papel fundamental dentro del sistema, estos actores se tuvieron en cuenta durante todo el desarrollo.

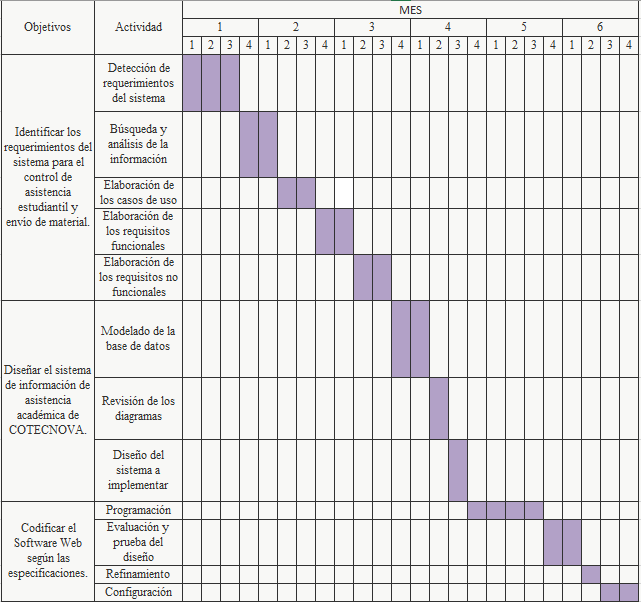
# Recomendaciones

Se deberá contar con una conexión de internet estable para el correcto funcionamiento del software ya que no funciona de manera offline.

En cuanto al envió del material tanto los docentes y usuarios dispuestos a recibir dicho correo deberán hacer una configuración en los mismos para poder recibir el contenido del correo.

# Cronograma

Ilustración   
Cronograma de actividades

  
*Fuente: Propia con ayuda de la herramienta Excel*

# Bibliografía

@admin. (2010, Octubre 15). *SRS / ERS Especificación de requerimientos de software*. Retrieved from Universidad ICESI: http://www.icesi.edu.co/departamentos/tecnologias\_informacion\_comunicaciones/proyectos/lisa/home/analisis/srs/srs

@Darkcrizt. (s.f). *MySQL Workbench un entorno gráfico para trabajar con MySQL [Blog]*. Retrieved from Ubunlog: https://ubunlog.com/mysql-workbench-un-entorno-grafico-para-trabajar-con-mysql/

@RequeridosBlog. (2018, Abril 20). *Requerimientos Funcionales y No Funcionales, ejemplos y tips [Blog]*. Retrieved from Medium: https://medium.com/@requeridosblog/requerimientos-funcionales-y-no-funcionales-ejemplos-y-tips-aa31cb59b22a

@Sanjahingu. (2014, Enero 28). *Métodos, Tipos y Enfoques de Investigación [Blog]*. Retrieved from Investigación: http://sanjahingu.blogspot.com/2014/01/metodos-tipos-y-enfoques-de.html

Alberto, M. R., & Delmy, P. V. (2008, Agosto). *Elaboración e implemenatción de un sistema informático para el instituto nacional "San José Verapaz" del municipio de verapaz, departamento de San Vicente.* Retrieved from Universidad de el Salvador: http://ri.ues.edu.sv/id/eprint/3375/1/70102338.pdf

Chen, P. (1977, Marzo). *The Entity Relationship Model – Toward a Unified View of Data.* Retrieved from Extra Materials: https://dspace.mit.edu/bitstream/handle/1721.1/47432/entityrelationshx00chen.pdf

Corporación de Estudios Tecnológicos del Norte del Valle. (2021). Retrieved from COTECNOVA: https://www.cotecnova.edu.co/

Corporación de Estudios Tecnológicos del Norte del Valle. (s.f). *Quiénes Somos*. Retrieved from Corporación de Estudios Tecnológicos del Norte del Valle: https://www.cotecnova.edu.co/index.php/quienes-somos/#tab-id-1

COTECNOVA. (1971). *Reglamento Estudiantil.* Retrieved from COTECNOVA: https://www.cotecnova.edu.co/wp-content/uploads/2016/09/Reglamento\_Estudiantil\_2005.pdf

Delgado Licona, E. (s.f). *Diseño de base de datos*. Retrieved from Universidad Autónoma del Estado de Hidalgo: http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro14/15\_proceso\_de\_desarrollo\_de\_la\_base\_de\_datos.html

Durán Toro, A. (2000, Septiembre 19). *Un entorno metodológico de ingeniería de requisitos para sistemas de información*. Retrieved from Depósito de investigación - Universidad de Sevilla: https://idus.us.es/handle/11441/15365

EcuRed. (s.f). *Visual Paradigm*. Retrieved from EcuRed: https://www.ecured.cu/Visual\_Paradigm

Everitt, J. (2020, Mayo 10). *Claves para entender los conceptos básicos del control del tiempo de los proyectos [Blog]*. Retrieved from Wrike: https://www.wrike.com/es/blog/claves-para-entender-los-conceptos-basicos-del-control-del-tiempo-de-los-proyectos/#:~:text=El%20control%20del%20tiempo%20es%20el%20proceso%20de%20registrar%20cu%C3%A1nto,a%20qu%C3%A9%20dedicaban%20su%20tiempo.

Frank Codd, E. (1970, Junio). *A Relational Model of Data for Large Shared Data Banks.* Retrieved from Penn Engineering: https://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf

Grupo Carricay. (2018, Junio 25). *¿Cómo escribir un buen documento de especificación de requisitos de software?* Retrieved from Medium: https://medium.com/grupo-carricay/c%C3%B3mo-escribir-un-buen-documento-de-especificaci%C3%B3n-de-requisitos-de-software-fd8bb3b5a39a

Hernández, E. (2012, Enero 30). *España: Tecnología NFC para controlar la asistencia en la Universidad de Salamanca*. Retrieved from FayerWayer: https://www.fayerwayer.com/2012/01/espana-tecnologia-nfc-para-controlar-la-asistencia-en-la-universidad-de-salamanca/

HN Datacenter en Chile. (s.f). *¿Qué es phpMyAdmin?* Retrieved from HN Datacenter en Chile: https://www.hostname.cl/blog/que-es-phpmyadmin

Liberato, G. (2013). *Técnicas de Recolección de Datos.* Retrieved from UTIM: https://gabriellebet.files.wordpress.com/2013/01/tecnicas-de-recoleccic3b3n4.pdf#targetText=La%20recolecci%C3%B3n%20de%20datos%20se,y%20el%20diccionario%20de%20datos.

MDN-contributors. (2021 A, 19 Mayo). *CSS*. Retrieved from MDN Web Docs: https://developer.mozilla.org/es/docs/Web/CSS

MDN-contributors. (2021 B, Mayo 18). *HTML: Lenguaje de etiquetas de hipertexto*. Retrieved from MDN Web docs: https://developer.mozilla.org/es/docs/Web/HTML

MDN-contributors. (2021 C, Mayo 18). *JavaScript*. Retrieved from MDN Docs: https://developer.mozilla.org/es/docs/Web/JavaScript

mineducacion. (s.f). *Vigilancia y Control*. Retrieved from mineducacion: https://www.mineducacion.gov.co/1759/w3-article-304225.html?\_noredirect=1

Ministerio de Educación Nacional. (2009). *Fundamentaciones y orientaciones para la implementacion del decreto 1290 de 2009.* Retrieved from Ministerio de Educación Nacional: https://www.mineducacion.gov.co/1621/articles-213769\_archivo\_pdf\_evaluacion.pdf

Publicaciones Semana S.A. (2020, Noviembre 19). Deserción escolar, un problema que ya supera los 100.000 estudiantes en 2020 [Web]. *Semana*. Retrieved from Semana: https://www.semana.com/pais/articulo/desercion-escolar-de-estudiantes-en-colombia-durante-el-2020/307215/

Revista de la Universidad de la Salle. (2016, Enero 7). *Revista de la Universidad de La Salle*. Retrieved from La administración del tiempo: una prioridad en la vida: https://ciencia.lasalle.edu.co/cgi/viewcontent.cgi?article=1481&context=ruls

Sandoval Illescas, J. X., & Sigüenza Cárdenas, R. E. (2011). *Analisis, Diseño e Implementación del Sistema de Control de Asistencia de Personal Docente Administrativo de la escuela Fiscal Mixta para Rafael Aguilar Pesantez.* Retrieved from Repositorio Institucional - Universidad Politécnica Salesiana: https://dspace.ups.edu.ec/bitstream/123456789/1636/14/UPS-CT002157.pdf

Universidad de Antioquia. (s.f). *Tarjeta Integrada personal*. Retrieved from Universidad de Antioquia: http://www.udea.edu.co/wps/portal/udea/web/inicio/institucional/tarjeta-integrada-personal

Uplanner. (2019, Marzo 12). *Por qué el control de asistencia es clave para el éxito estudiantil [Blog]*. Retrieved from Uplanner: https://www.uplanner.com/es/blog/por-que-el-control-de-asistencia-es-clave-para-el-exito-estudiantil/

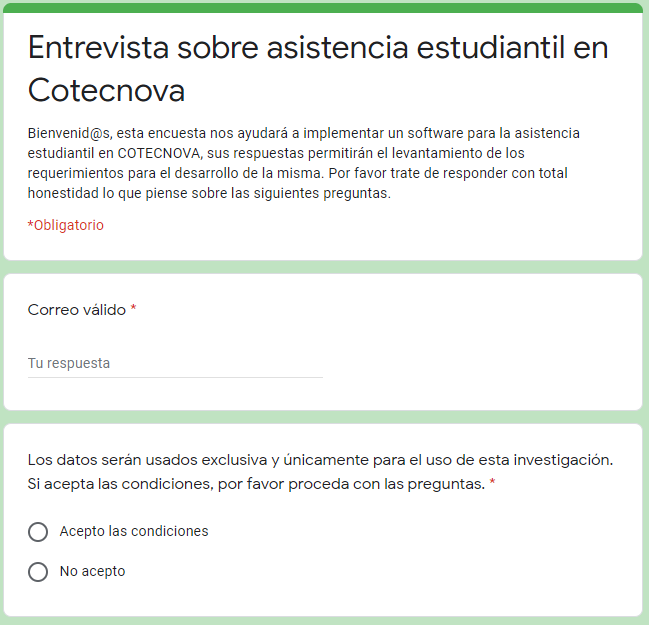
Winston, R. (1970). MANAGING THE DEVELOPMENT OF LARGE SOFTWARE SYSTEMS. In R. Winston, *MANAGING THE DEVELOPMENT OF LARGE SOFTWARE SYSTEMS* (pp. 328, 331-338).

# Anexos

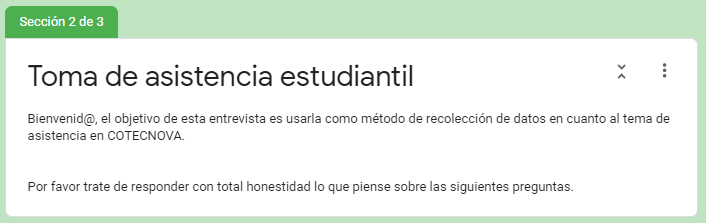
Anexo   
Formato de observación diligenciado

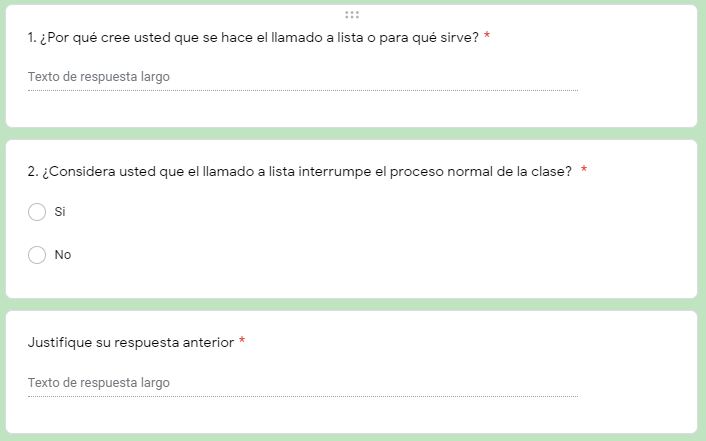
|  |  |
| --- | --- |
| **Lugar:** | COTECNOVA |
| **Objetivo:** | Recolectar información sobre la toma de asistencia | **Nombre de los investigadores:** Natalia Agudelo Valdés - Juan David Hoyos | | | |
| **Asignatura observada** | **Docente a cargo** | **Llamado lista (si/No)** | **Método de llamado** | **Tiempo** | **Descripción** |
| Inteligencia artificial | Rafael Rubiano | Sí | Oral | 5-8 minutos | El docente disponible el espacio al finalizar la clase para hacer el llamado de atención. |
| “” | “” | Sí | Escritura virtual | Indefinido | El docente en el transcurso de la clase pide a los alumnos hacer escritura de su nombre en el chat para el llamado de asistencia |
| “” | “” | No | - | - | El docente no hizo llamado alguno o alusión del mismo durante la clase. |
| Metodología de la investigación | Arvey Barahona | Sí | Visual | No calculable | El docente en el transcurso de la clase observa los nombres de los estudiantes y menciona/pregunta algunos que no están. |
| “” | “” | Sí | Escritura virtual | Indefinido | El docente en el transcurso de la clase pide a los alumnos hacer escritura de su nombre en el chat para el llamado de asistencia |
| “” | “” | No | - |  | El docente no hizo llamado alguno o alusión del mismo durante la clase. |
| Inglés A-2 | Jairo Alonso Orjuela | Sí | Oral | 8-10 minutos | El docente disponible el espacio al finalizar la clase para hacer el llamado de atención. |
| “” |  | Sí | Oral | 8-10 minutos | El docente disponible el espacio al finalizar la clase para hacer el llamado de atención. |
| “” |  | Sí | Oral | 8-10 minutos | El docente disponible el espacio al finalizar la clase para hacer el llamado de atención. |
| Proyecto integrador de tecnología | Arvey Barahona | Sí | Visual | No calculable | El docente en el transcurso de la clase observa los nombres de los estudiantes y menciona/pregunta algunos que no están. |
| “” | “” | Sí | Visual | No calculable | El docente en el transcurso de la clase observa los nombres de los estudiantes y menciona/pregunta algunos que no están. |
| “” | “” | No | - | - | El docente no hizo llamado alguno o alusión del mismo durante la clase. |
| Ciencia, tecnología y sociedad | Sonia Helena Godoy | Sí | Oral | 10 minutos o más | El docente disponible el espacio al finalizar la clase para hacer el llamado de atención. |
| “” | “” | Sí | Oral | 10 minutos o más | El docente disponible el espacio al finalizar la clase para hacer el llamado de atención. |
| “” | “” | Sí | Oral | 10 minutos o más | El docente disponible el espacio al finalizar la clase para hacer el llamado de atención. |
| Diseño de sistemas de información | Julio Cesar Velosa | Si | Escritura virtual | Indefinido | El docente en el transcurso de la clase pide a los alumnos hacer escritura de su nombre en el chat para el llamado de asistencia |
| “” | “” | Sí | Visual | No calculable | El docente en el transcurso de la clase observa los nombres de los estudiantes y menciona/pregunta algunos que no están. |
| “” | “” | Sí | Visual | No calculable | El docente en el transcurso de la clase observa los nombres de los estudiantes y menciona/pregunta algunos que no están. |

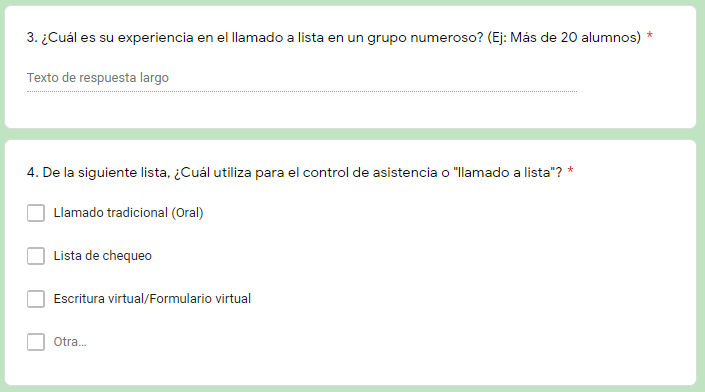
Anexo   
Formato de Entrevista

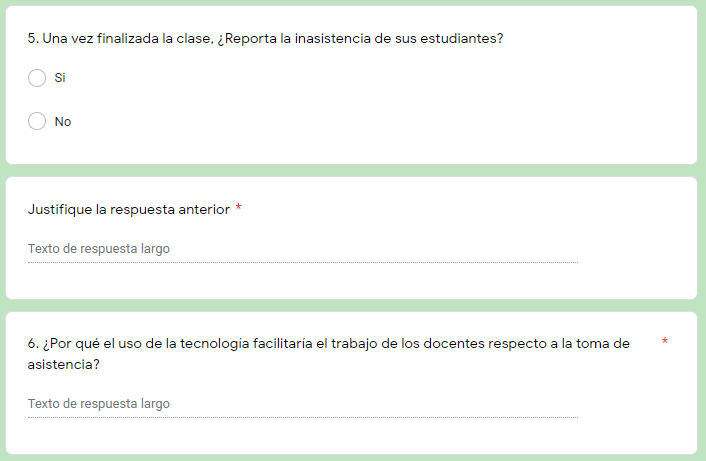


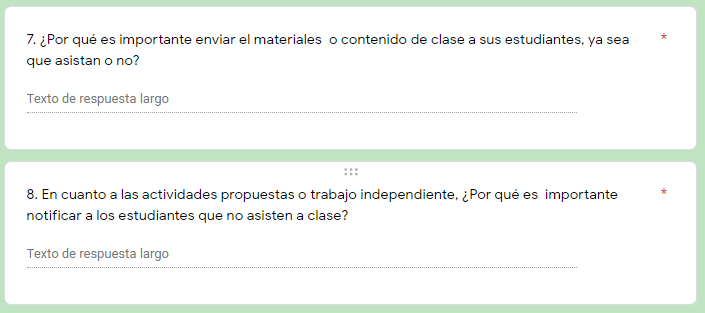
Continuación Formato de Entrevista









  
*Fuente: Elaboración propia con ayuda de la herramienta Formularios de Google*

Anexo   
Comunicado de falta de asistencia

**

*Fuente: Comunicado enviado por parte de la Universidad*