



FACULTAD DE INGENIERÍA

Carrera Profesional de Ingeniería de Sistemas e Informática

Curso Integrador I: Sistemas – Software (30348)

Título: Diseño de Aplicación Web aplicado a automatización de la gestión de servicios
en empresa SPA UTP

Ciclo: VI

Autores:

Gamarra Mamani, Carlos Alberto
Gutierrez Thaquima, Luz Mery
Mamani Huanca, Jhoel Rene
Quispe Huamani, Anthony Alfredo

Docente: Ing. Aníbal Sardón Paniagua

Arequipa - Perú
2023

HISTORIAL DE VERSIONES

Fecha de Elaboración	Versión	Elaborado por	Descripción	Revisado por	Fecha de Revisión
22/04/2023	1.0	Equipo	Avance de Proyecto Final 1	Anthony Q.	22/04/2023
13/05/2023	2.0	Equipo	Avance de Proyecto Final 2	Anthony Q.	13/05/2023
27/05/2023	3.0	Equipo	Avance de Proyecto Final 3	Anthony Q.	27/05/2023
10/06/2023	4.0	Equipo	Avance de Proyecto Final 4	Jhoel M.	10/06/2023

Contenido

CAPITULO 01: ASPECTOS GENERALES.....	5
1. DEFINICIÓN DEL PROBLEMA.....	5
I. Descripción del problema	5
II. Diagramas de Procesos de Negocio	6
2. PLANTEAMIENTO DE ALTERNATIVAS DE SOLUCIÓN	9
I. Aplicativo Móvil.....	9
II. Aplicación de escritorio con JFrame.....	11
III. Página web con Servidor de base de datos.....	12
3. DEFINICIÓN DE OBJETIVOS.....	15
I. Objetivo General.....	15
II. Objetivos específicos.....	16
III. Alcances y Limitaciones	16
IV. Justificación.....	17
V. ESTADO DEL ARTE.....	17
CAPÍTULO 02: MARCO TEÓRICO	19
1. FUNDAMENTO TEÓRICO.....	19
I. PÁGINAS WEB	19
II. SERVIDORES	19
III. JAVA PERSISTENCE API	21
IV. SERVLETS	22
V. INTERFAZ DE PÁGINAS WEB	22
CAPÍTULO 03: DESARROLLO DE LA SOLUCIÓN	23
1. ANÁLISIS DEL SISTEMA	23
I. Requisitos Funcionales	23
II. Requisitos No Funcionales.....	27
2. DISEÑO DEL SISTEMA.....	29
I. Casos de Uso.....	29
II. Diseño de Clases	37
III. Diseño de la Base de datos	37
IV. Código Fuente	38
ANEXOS	42

RESUMEN

En este trabajo se muestra desarrollo de aplicación web para gestión de servicios en una empresa del rubro SPA, se presenta los diferentes procesos que comprenden el funcionamiento de la empresa. Asimismo, se presenta desarrollo de la implementación, desde análisis de los requerimientos funcionales y no funcionales, diseños de la base de datos y configuración de las funcionalidades de la aplicación web para automatizar los diferentes procesos de la empresa. Se utiliza java como lenguaje de programación fundamental para el proyecto y algunos FrameWorks que se analizarán durante el desarrollo para una efectiva solución de los requerimientos del sistema.

Palabras clave: Automatización, sistemas de gestión, java, página web, servlets, JPA.

ABSTRACT

This work shows the development of a web application for service management in a SPA company, the different processes that comprise the operation of the company are presented. It also presents the development of the implementation, from analysis of functional and non-functional requirements, database design and configuration of the web application functionalities to automate the different processes of the company. Java is used as the fundamental programming language for the project and some FrameWorks that will be analyzed during the development for an effective solution of the system requirements.

Keywords: Automation, management systems, java, web page, servlets, JPA.

Archivos del Proyecto se encuentran en repositorio de GitHub:
Link: https://github.com/AnthonyQuispeHuamani/SPA_UTP

CAPITULO 01: ASPECTOS GENERALES

1. DEFINICIÓN DEL PROBLEMA

La gestión de procesos en sistemas para una empresa del rubro SPA es importante para maximizar la satisfacción por el servicio gracias a una eficiente distribución de horarios y servicios disponibles con registro. En este trabajo, el caso de estudio es para empresa UTP, que brinda diferentes servicios los cuales no contienen un sistema de gestión automatizado, presentando inconvenientes en registros de servicios y correctas asignaciones de tareas en la empresa.

I. Descripción del problema

La empresa SPA UTP es una empresa que llegó a Arequipa a ofrecer sus servicios de SPA urbano donde los clientes podrán encontrarán un refugio de paz, armonía y podrán complacer sus sentidos, conectando con su fuerza espiritual a través de una experiencia única y personal. Además de, contar con un equipo especializado y de alta categoría que realiza terapias que unen los cuatro elementos: tierra, agua, fuego y aire garantizando a sus clientes una experiencia inolvidable.

Este trabajo tiene como caso de aplicación de la implementación de sistema de gestión en la empresa SPA UTP, a continuación se presenta la misión, visión, servicios que ofrece y problemas encontrados.

Misión

Ser una empresa dedicada al buen servicio de terapias de spa, entregar un servicio de calidad en todas las categorías de servicio que brindemos tanto corporativos como industriales y particulares.

Visión

Ser la mejor empresa de Spa en el departamento de Arequipa. Dándoles servicio de calidad a nuestros clientes.

La empresa SPA UTP ofrece diferentes servicios como:

- Servicio Categoría de SPA.
 1. La ducha Vichy.
 2. Sauna · Exfoliación · Vichy · Masaje Ritual Relax.
 3. Jacuzzi.
- Servicio Categoría de Tratamientos Corporales
 1. Envolturas Hidratantes de Miel
 2. Choco terapia.
 3. Exfoliación con sales del Mar Muerto.
 4. Fango remineralizante.
- Servicio Categoría de Relajación y Masajes
 1. Ritual Con Piedras Volcánicas
 2. Masaje Royal Reductor
 3. Ritual Relax

Actualmente la empresa tiene problemas respecto al manejo y control de datos de sus clientes por el uso de un software inadecuado para almacenar información de importancia para la empresa SPA UTP. Por otro lado, la empresa no cuenta con un proceso eficiente de registro de citas de los clientes. En consecuencia, esto podría traer futuras situaciones perjudiciales para sus clientes y la misma empresa. Por último, la empresa cuenta con vías ineficientes de comunicación como es el WhatsApp ya que abría congestión en las llamadas y los mensajes recibidos, evitando la atención a varios clientes.

II. Diagramas de Procesos de Negocio

• Registro de Citas

1. Un cliente solicita una cita por teléfono o comunicándose por el WhatsApp de la empresa.

El cliente indica sus datos personales (nombres, apellidos, dirección, teléfono), los servicios solicitados

2. El encargado atiende al cliente, consulta en las hojas de cálculo, los horarios de atención y para ver las disponibilidades de tiempo de los técnicos que aplican los masajes y tratamientos.

El encargado le dice al cliente las fechas y horas disponibles.

3. El cliente elige la fecha y hora de la visita.

4. El encargado selecciona o registra los datos del cliente (en caso no se encuentre en la base datos), la fecha y hora de la visita. Separa el espacio en los horarios de los técnicos, separa la sala donde se atenderá al cliente y selecciona los insumos necesitados para la atención del cliente.

5. El encargado calcula el total a cobrar de acuerdo con las tarifas por los servicios brindados y envía por correo electrónico al cliente su orden de atención con el monto a cobrar.

6. El cliente paga por el servicio usando algún medio electrónico (Yape, Plin, Transferencia Bancaria), o puede pagar al contado al momento de visitar el SPA y recibir los servicios.

• Programación de los técnicos

1. El administrador ingresa al sistema con sus credenciales.

2. El administrador registra el servicio con datos del mismo para luego consultar si existe disponibilidad de trabajador. Ya que se verificará si el trabajador se encuentra ocupado o no está habilitado para realizar la tarea.

3. Caso estuviese habilitado, se procede a notificar el registro.

4. Opcional, el administrador podrá entregar a cada técnico la lista de los servicios que deberá realizar en cada salida, (impreso o por WhatsApp) con todos los detalles de los clientes, servicios y tiempos.

5. Una vez notificado o generado el registro se podrá dirigirse al menú para

esperar un nuevo servicio a generar o finalizar sesión.

- **Proceso de Atención en el SPA**

1. El cliente ingresa a la sala de atención, donde se encuentra esperando la/el técnico, con los insumos necesarios para ofrecer los servicios solicitados.
2. Luego de finalizar los servicios, el técnico llena la orden de atención para que pase por caja para pagar.
3. El cliente pasa por la caja, el encargado imprime su comprobante de pago.
4. En caso el cliente haya pagado con anticipación por los servicios por algún medio electrónico (Yape, Plin, Transferencia Bancaria), el encargado imprime su comprobante de pago.
5. El encargado entrega el comprobante de pago al cliente
6. El cliente se retira del SPA.

Diagrama Registro de Cita

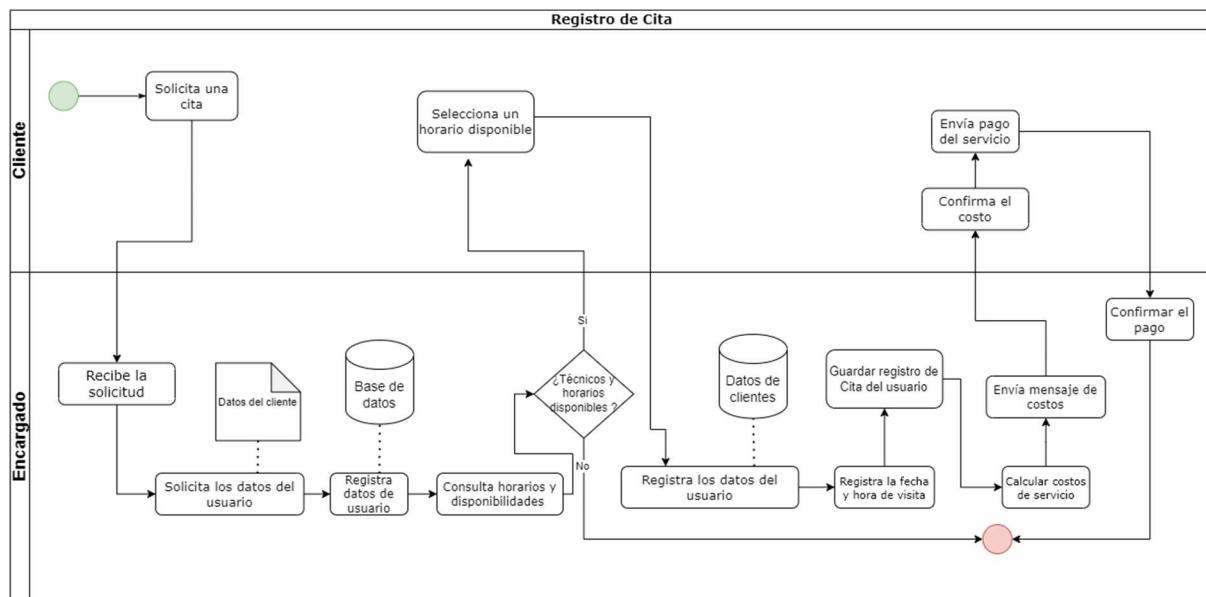


Figura: Diagrama Registro de Cita

Diagrama programación de los técnicos

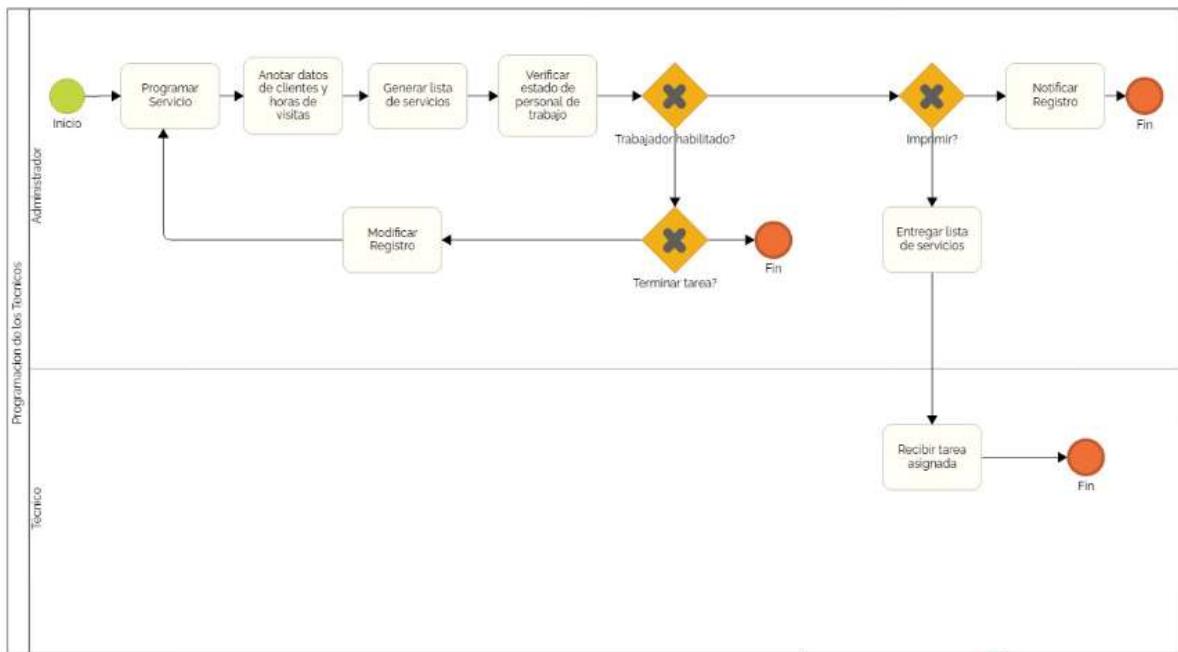


Figura: Diagrama programación de los técnicos

Diagrama Proceso de atención en el SPA

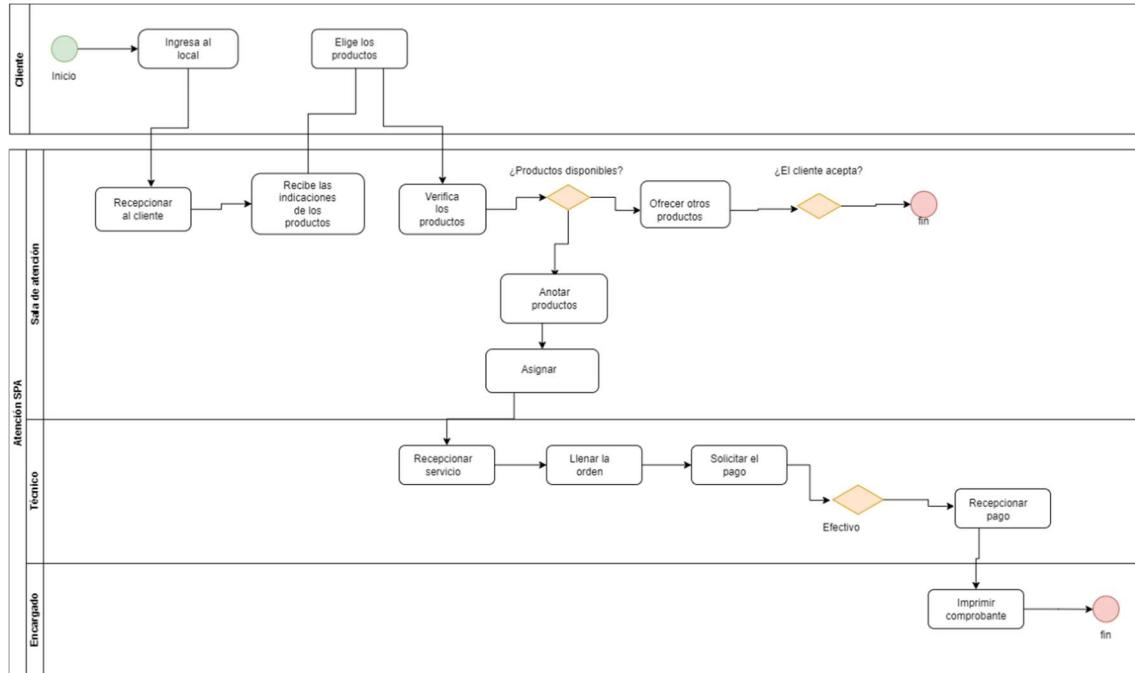


Figura: Diagrama proceso de atención

2. PLANTEAMIENTO DE ALTERNATIVAS DE SOLUCIÓN

En este apartado se describen las alternativas de solución que pueden ser implementadas para automatizar la gestión y administración de servicios para la empresa SPA UTP.

- Alternativa solución 1: Aplicativo móvil.
- Alternativa solución 2: Aplicación de escritorio con JFrame.
- Alternativa solución 3: Página web con Servidor de base de datos.

I. Aplicativo Móvil

Como primera alternativa de solución planteamos un aplicativo móvil ya que estos aplicativos han demostrado monitorizar y gestionar los diferentes procesos que se realizan en diferentes empresas. Además, permitiría a la empresa tener una conexión más directa e instantánea con sus clientes, aumentaría su visibilidad, podría conectarse con clientes en movimiento y estos podrían acceder de una manera fácil y didáctica al inventario de SPA UTP para agendar rápidamente y sin problemas algún servicio de la empresa.

- El aplicativo móvil podrá ser creado con las herramientas de desarrollo:
- Java como lenguaje de programación del aplicativo móvil.
- MySQL como gestor de base de datos.

Ventana de Ingreso:

Ventana de Registro de los diferentes tipos de usuarios como el cliente, vendedor o administrador registrados por el aplicativo móvil.

Ventana de Información:

Podemos visualizar el menú principal del aplicativo móvil donde nos ofrecen su diferente categorías servicio que la empresa SPA UTP.

En la parte inferior podremos encontrar información sobre la empresa.



Figura: Ventanas de Ingreso e Información

Ventana de Programación de Cita:

Ventana para poder registrar de citas las diferentes citas programadas por el usuario(vendedor).

Ventana de Información de servicios:

Ventana intuitiva llena de información sobre los diferentes paquetes dirigidas a los clientes.



Figuras: Programación de citas y Reporte de citas

Ventana de Reporte

Ventana de búsqueda de citas mediante diferentes datos registrados en la base de datos.

Ventana de Registro de Usuario:

Ventana para registrar a un nuevo usuario mediante sus datos.



Figuras: Ventanas reporte y registro usuario

II. Aplicación de escritorio con JFrame.

Como segunda alternativa de solución planteamos una aplicación de escritorio con uso de JFrame y datos almacenados en archivos .XML. Esta aplicación podría ser ejecutada desde cualquier escritorio en que vayamos a tener la aplicación. Además, podremos generar datos e importar datos desde XML y no necesariamente utilizando gestor de base de datos ni servidor web. Es una alternativa de solución sencilla, pero podría cubrir las necesidades de la empresa de manera fácil y didáctica en uso para el administrador SPA UTP para agendar rápidamente y sin problemas algún servicio de la empresa. La desventaja será que será necesaria la presencia del cliente en oficinas de la empresa para poder generar su cita, ya que el aplicativo estará corriendo en un escritorio como servidor a la vez.

- Se utiliza java como lenguaje de programación.
- Se utiliza JFrame para generar ventanas en las cuales deberemos de implementar las interacciones.
- Se utiliza archivo XML para escribir y recopilar datos.
- Se utiliza un modelo MVC, de forma que podamos interpretar y construir mejor la información de funcionalidad.

A continuación, se presentan capturas de las ventanas en ejecución:



Figura: Formulario de login



Figura: Formulario de registro de usuario



Formulario Menú

III. Página web con Servidor de base de datos

Mejorar la interface de comunicación entre el cliente y servicio, mejorando así la atención a varios usuarios al mismo tiempo, ya sea, al momento de reservar una cita, así como, el control en los insumos, además, se mejorará el control de datos con respecto a los clientes y sus citas, ya que actualmente se está usando Excel para el registro de los servicios y los datos del cliente, el cual podría causar perdida de datos o fallos en los datos registrados.

Las siguientes paginas están creabas bajo las siguientes herramientas de desarrollo

- JavaScript, HTML, CSS para la creación del fronted, y demás interfaces y campo visual del usuario.
- Java para el backend y como gestor de la base de datos y parte lógica, se utilizará MySQL.

MOCKUPS Y DESCRIPCIÓN:



Figura: Logo empresa SPA UTP

Ventana de Ingreso

En este apartado se podrá introducir las credenciales para poder acceder al sistema, se deberá verificar internamente si el usuario existe para dar acceso. Sino, deberá registrarse.

INGRESE

Correo Electronico

Contraseña

[¿No tienes una cuenta? REGISTRATE](#)

INGRESAR

Figuras: Página para el ingreso del cliente o del trabajador para acceder a las distintas opciones.

Ventana de Registro

Página para el registro del cliente o del trabajador.

REGISTRESE

Nombre

Apellidos Paterno

Apellidos Materno

Correo Electronico

Contraseña

Direccion

Referencia

DNI

REGISTRAR

Figuras: Registro

Ventana de Reservación de citas:

Página para la reservación de una cita nueva. Esta opción puede variar dependiendo al Trabajador o cliente y las visualizaciones de ciertas características.

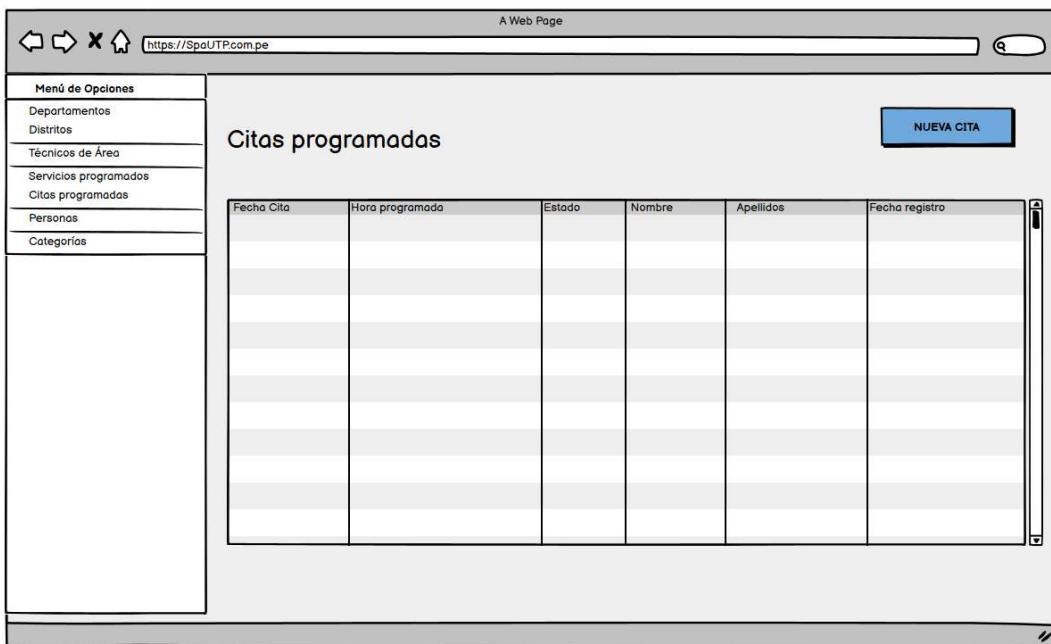


Figura: Reservación cita

Ventanas de Reporte de servicios:

REPORTE DE SERVICIO					
Filtrar					
Por Fec	10/09/2022	Por Ti	Chocoterapia	<input checked="" type="checkbox"/> Buscar	
Por Catego	TRATAMIENTOS CORPORAL				
ID	Cliente	DNI	Distrito	Dirección	Fecha
4471	Alejandro Martínez	83667108	Cerro Colorado	Calle España 204	10/09/2022
4741	Jose Luis	71540082	Cerro Colorado	Calle Peregrinos 109	11/09/2022

Figuras: Reporte Servicio

Funcionalidades de la Página Web:

A Web Page

<https://LavanderiaUTP.com.pe>

SPA UTP

Menú de Opciones	<h3>Reporte de servicios por cliente</h3> <p>Filtrar</p> <p>Por Fecha <input type="text" value="10/09/2022"/> <input type="button" value=""/></p> <p>Por Mont <input type="text" value="S/50.00 - S/100.0"/> <input type="button" value=""/></p> <p><input checked="" type="checkbox"/> Buscar</p>						
Departamentos	Cliente	DNI	Teléfono	Fecha Entrega	Tratamiento programado	Monto total	Estado
Distritos	Alejandro Martínez	83667108	741 869 115	10/09/2022	Chocoterapia	S/ 85	Pagado
Técnicos de Área							
Servicios programados							
Citas programadas							
Personas							
Categorías							

Figura: Filtros Búsqueda

Figura: Filtros Búsqueda

3. DEFINICIÓN DE OBJETIVOS

I. Objetivo General

El objetivo general del presente trabajo es analizar, diseñar e implementar un sistema simulado de información en un entorno página Web, que permita la gestión y administración de servicios de la empresa SPA UTP por medio de algoritmos realizados en Java y otros recursos de base de datos.

II. Objetivos específicos

- Modelar la arquitectura de información necesaria para administrar (registrar, editar y/o deshabilitar manualmente) las rutas de transporte como los horarios del sistema con interacción de registro de usuarios.
- Establecer los mecanismos de carga masiva de información de rutas para la arquitectura definida, considerando aspectos tales como servicios que se brindan, actores que participan en el sistema y exportaciones de datos como informes.
- Diseñar un algoritmo Java junto a Frameworks que nos permitan generar una página Web para la interacción de un usuario a fin de optimizar la creación y asignación de servicios minimizando costos innecesarios y, asimismo, maximizando la satisfacción por el servicio gracias a una eficiente distribución de horarios y servicios disponibles con registro.
- Implementar un prototipo del sistema de información en un entorno simulado adecuado para esta solución usando herramientas, controles, estilos y formatos comunes basados en estándares conocidos de Java para la interacción del cliente, registro y personalización de servicios en preferencia que brinda la empresa.

III. Alcances y Limitaciones

El proyecto consiste en desarrollar una aplicación para la atención en el Spa UTP.
Los módulos que se manejarán son:

Ingreso y registro en el sistema.- Es donde el usuario tendrá que poner todos sus datos para que se pueda llevar un registro de su atención y así mismo pueda dar seguimiento de sus pendientes o solicitudes de atención de la Spa.

Reservación o Cita.- El usuario registrará la cita indicando sus datos personales y eligiendo un tratamiento.

Atención en Spa.- Este módulo está diseñado para que los trabajadores de la empresa hagan el ingreso del cliente a un servicio en el spa con sus datos previamente registrados.

Reportes.- Aquí los trabajadores de la empresa podrán ver toda la información de los servicios realizados con el detalle de todos los clientes y atenciones, tenemos:

- Reportes de servicios por tipos de servicio.
- Reporte de servicios por cliente.
- Resumen de servicios por Personal.

Así mismo el sistema contará con los siguientes servicios a seleccionar:

- SPA
- TRATAMIENTOS CORPORALES
- RELAJACION Y MASAJES

FUERA DEL ALCANCE

1. No se considera el módulo de pagos y tesorería.
2. No se considera el área de contabilidad.

LIMITACIONES

1. No se tiene acceso a los datos anteriores de los registros.
2. No se trasferirá los datos registrados anteriores de las hojas de cálculo a la aplicación, ya que faltaran datos por completar.
3. NO se tiene acceso a los medios de pago. Por ende, la aplicación no manejará cuentas bancarias, transferencias u otro medio de pago dentro de la aplicación.

IV. Justificación

Actualmente en las pequeñas y abarcando a grandes empresas presentan dificultad de en sus sistemas de gestión de sus servicios y/o productos ventas que conservan. Es así que es normal presentar niveles bajos de rendimiento en la atención a los múltiples clientes, perjudicando a la clientela, y causando problemas de orden y control en la empresa.

Además, se necesita mejorar en control de los datos de los clientes y su correcto almacenamiento en una base de datos estable, esto ayudara a mejorar la velocidad en la atención a los clientes que vienen de forma recurrente, además de gestionar mejor los servicios que se ofrecen. Al no implementar esta mejora con la página web, podría perjudicar a largo plazo la forma de trabajo de la empresa, ya que actualmente trabajan con archivos Excel para el manejo de datos de sus clientes, y el uso de WhatsApp para coordinar las citas y almacenan los datos de sus clientes en archivos Excel, esto podría causar perdida de datos y la gestión de estos datos. Por ende, este proyecto se orienta en la mejora y optimización del servicio al cliente y la gestión de los procesos del servicio para los trabajadores

V. ESTADO DEL ARTE

PROYECTO 1:

- La referencia principal de este trabajo es el diseño Sistema de gestión de empresa para Ventas de Navinda Chandrasiri, donde implementa un diseño Modelo-Vista-Controlador (MVC), utilizando JSPs y sus clases con sus métodos CRUD. (Chandrasiri, 2022)

PROYECTO 2:

- El segundo proyecto que se revisó es guía de SAINI VIKASH, en su proyecto (Gestión de relaciones comerciales en línea) nos muestra la implementación de servlets para hacer las comunicaciones con respecto a las funciones como el registro de datos, y con esto nosotros podemos utilizar para el registro, listado y eliminación de los respectivos datos. (SAINI, 2015) (F., 2018)

PROYECTO 3:

- Como tercer proyecto guía nos ayudamos de Gregggy Gianini F., en su (Proyecto de programación Orientada a objetos), para la implementación de daos, para así implementar las funciones básicas y principales de cada uno de las entidades el cual se repiten estas funciones. (F., 2018)

PROYECTO 4:

- El proyecto “Diseño e implementación de una tienda virtual”, este proyecto utiliza como base de datos MySQL.
- Se está utilizando las principales funciones para la gestión de los datos como es el caso de las consultas, ya sean, las de insertar, modificar y

eliminar, que en nuestro caso son las funciones de create, Edit, y Destroy.

- Para la implementación estamos usando el servidor de datos MySQL, el cual su definición según Ruiz Antonio MySQL es:
- “El servidor de datos relacionales mas popular,desarrollado y proporcionado por MySQL.es una empresa cuyo negocio consiste en proporcionar servicios en torno al servidor de bases de datos MySQL” (Ruiz, 2010).
- También se está utilizando el servidor Apache. “Que es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1 (RFC 2616) y la noción de sitio virtual.” (Ruiz, 2010).

CAPÍTULO 02: MARCO TEÓRICO

1. FUNDAMENTO TEÓRICO

I. PÁGINAS WEB

La página web es la unidad básica de la World Wide Web, (la Web). Se puede definir como un documento electrónico que contiene información específica de un tema en particular y que es almacenado en algún sistema de cómputo que se encuentra conectado a la red de información Internet.

Se pueden identificar etapas de una etapa web: Planificación, Estructuración, Diseño-Implementación y Publicación.

II. SERVIDORES

- **Servidores de Base de Datos**

Un servidor de base de datos es un programa informático que posibilita que otros programas puedan acceder a uno o varios sistemas de bases de datos a través de una red.



Figura: Referencial Datos de información

En el contexto de bases de datos, existen diferentes tipos de servidores que se utilizan para administrar y almacenar la información. La elección del tipo de servidor de bases de datos dependerá de los requisitos de la aplicación y las necesidades particulares del proyecto. A continuación, se mencionan los más comunes:

1. Servidor de bases de datos relacional: Es el tipo más común de servidor de bases de datos. Aquí, los datos se organizan en tablas con filas y columnas, y se utilizan lenguajes de consulta como SQL (Structured Query Language) para acceder y manipular los datos. Ejemplos populares de servidores de bases de datos relacionales son MySQL, Oracle Database y Microsoft SQL Server.
2. Servidor de bases de datos NoSQL: Estos servidores de bases de datos están diseñados para manejar grandes volúmenes de datos no estructurados o semi estructurados. Los sistemas NoSQL se basan en modelos de datos flexibles y permiten un escalado horizontal más fácilmente que los sistemas relacionales. Algunos ejemplos de servidores de bases de datos NoSQL son MongoDB, Cassandra y Redis.

3. Servidor de bases de datos en memoria: Estos servidores almacenan los datos en la memoria principal del servidor en lugar de en un disco duro, lo que permite un acceso y procesamiento de datos más rápidos. Son especialmente útiles para aplicaciones que requieren una alta velocidad de lectura y escritura. Ejemplos de servidores de bases de datos en memoria son Redis, Memcached e IBM Db2 BLU.
4. Servidor de bases de datos en la nube: Estos servidores de bases de datos se ejecutan en la nube, lo que significa que los datos y la infraestructura subyacente están alojados y administrados por un proveedor de servicios en la nube, como Amazon Web Services (AWS), Microsoft Azure o Google Cloud Platform (GCP). Algunos ejemplos de servidores de bases de datos en la nube son Amazon RDS, Azure SQL Database y Google Cloud Spanner.

- **Servidores de Páginas Web**

Un servidor web es un software y un hardware que utiliza el protocolo HTTP (Hypertext Transfer Protocol) y otros protocolos para responder a las peticiones de los clientes realizadas a través de la World Wide Web. La principal función de un servidor web es mostrar el contenido de un sitio web almacenando, procesando y entregando las páginas web a los usuarios. Además de HTTP, los servidores web también soportan SMTP (Simple Mail Transfer Protocol) y FTP (File Transfer Protocol), utilizados para el correo electrónico, la transferencia de archivos y el almacenamiento.



Figura: Referencial páginas web server

Existen varios tipos de servidores web utilizados para alojar y entregar sitios web y aplicaciones en Internet. A continuación, te presentaré algunos de los tipos más comunes de servidores web:

1. Servidor web Apache: Apache HTTP Server, también conocido como Apache, es uno de los servidores web más populares y ampliamente utilizados en el mundo. Es un software de código abierto y multiplataforma que puede ser instalado en diferentes sistemas operativos, como Linux, Windows y macOS. Apache es conocido por su estabilidad, flexibilidad y capacidad para soportar una amplia gama de módulos y complementos.
2. Servidor web Nginx: Nginx (pronunciado "engine-x") es otro servidor web

de código abierto que ha ganado popularidad en los últimos años. Se destaca por su alto rendimiento y eficiencia en el manejo de múltiples solicitudes simultáneas. Nginx es especialmente adecuado para servir contenido estático, actuar como proxy inverso y equilibrador de carga, y manejar aplicaciones web de alto tráfico.

3. Servidor web Microsoft IIS: Internet Information Services (IIS) es el servidor web desarrollado por Microsoft para sistemas operativos Windows. Está integrado con el sistema operativo Windows Server y se utiliza comúnmente para alojar aplicaciones web en entornos basados en Microsoft. IIS admite tecnologías como ASP.NET y ofrece características avanzadas de seguridad y administración.
4. Servidor web Lighttpd: Lighttpd (también conocido como "Lighty") es un servidor web de código abierto y ligero que se centra en el rendimiento y la eficiencia. Está diseñado para ser rápido y consumir pocos recursos del sistema, lo que lo hace adecuado para servidores con limitaciones de hardware o donde se requiere un alto rendimiento en situaciones de alta carga.
5. Servidor web Tomcat: Apache Tomcat es un contenedor de servlets de código abierto que se utiliza para alojar aplicaciones web basadas en Java. Aunque técnicamente no es un servidor web completo, Tomcat se combina a menudo con un servidor web como Apache para entregar aplicaciones Java en la web.

En nuestro trabajo, utilizaremos el servidor Tomcat versión 9, que previamente es iniciado desde Netbeans.

III. JAVA PERSISTENCE API

JPA es la propuesta estándar que ofrece Java para implementar un Framework Object Relational Mapping (ORM), que permite interactuar con la base de datos por medio de objetos, de esta forma, JPA es el encargado de convertir los objetos Java en instrucciones para el Manejador de Base de Datos (MDB).



Figura: Referencial JPA

Cuando empezamos a trabajamos con bases de datos en Java lo primero que nos enseñan es a utilizar el API de JDBC el cual nos permite realizar consultas directas a la base de datos a través de consultas SQL nativas. JDBC por mucho tiempo fue la única forma de interactuar con las bases de datos, pero representaba un gran problema y es que Java es un lenguaje orientado a objetos y se tenía que convertir los atributos de las clases en una consulta SQL como SELECT, INSERT, UPDATE, DELETE, etc. lo que ocasionaba un gran esfuerzo de trabajo y un provocaba muchos errores en tiempo de ejecución, debido principalmente a que las consultas SQL se tenían que generar frecuentemente al vuelo.

IV. SERVLETS

V. INTERFAZ DE PÁGINAS WEB

CAPÍTULO 03: DESARROLLO DE LA SOLUCIÓN

1. ANÁLISIS DEL SISTEMA

I. Requisitos Funcionales

RF-001	Es sistema debe permitir registrar usuarios
Dependencias	Usuario nuevo sin registro
Descripcion	El sistema deberá permitir que el usuario ingresar la información:
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-002	Es sistema debe permitir Actualizar o modificar los datos del usuario
Dependencias	El usuario ya debe estar registrado
Descripcion	El sistema deberá permitir que el usuario pueda actualizar toda su informacion. Nombres y apellidos.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-003	El sistema debe permitir ingresar (Logeo)
Dependencias	El usuario debe ya estar registrado El usuario debe ya estar asignado a un tipo de Usuario
Descripcion	El sistema deberá permitir al usuario ingresar:
Importancia	Alto
Estabilidad	Alta
Comentarios	Cuando valide al usuario mostrara de acuerdo con el tipo de usuario los accesos

RF-004	El sistema debe permitir registrar una o mas citas al usuario
Dependencias	El usuarios deben iniciar seccion. El usuario debe seleccionar la pestaña de citas programadas. El usuario debe seleccionar el boton "Añadir nueva cita".
Descripcion	El sistema abrirá una ventana "Nueva Cita". El sistema pedirá ingresar los datos: fecha de cita, hora cita y cliente (este ya debe tener una cuenta).
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguna

RF-005	<u>El sistema debe Mostrar los reporte de las citas solicitadas</u>
Dependencias	El usuario debe iniciar seccion. El usuario debe dirigirse a la pestaña de citas programadas. El usuario debio registrar citas al sistema previamente.
Descripcion	El sistema debera mostrar todos los reportes de citas programadas con sus respectivos datos: fecha cita, hora programada, nombre, apellidos, estado, creado, modificar y acciones.
Importancia	Alto
Estabilidad	Alta
Comentarios	Esto aplica para reservaciones

RF-006	<u>El sistema debe dar permisos particulares al administrador</u>
Dependencias	El usuario debe estar registrado como administrador
Descripcion	El sistema reconocera al usuario "Administrador" y le dara facultad para poder modificar o eliminar informacion del sistema.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-007	<u>El sistema debe permitir mostrar los servicios SPA UTP</u>
Dependencias	El usuario debe estar registrado al sistema. El usuario debe navegar en el sistema.
Descripcion	El sistema mostrara las categorias de tratamiento que ofrece SPA UTP.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-008	<u>El sistema debe permitir al usuario registrar un usuario y delegar el tipo de usurio.</u>
Dependencias	El usuario debe iniciar seccion. El usuario selecciona la pestaña de personas. El usuario selecciona el boton "Añadir persona".
Descripcion	El sistema habre una ventana y muestra los datos a ingresar para registrar. El sistema lista las opciones de tipos de usuarios disponibles.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-009	Contar con un CRUD (create, read, update, delete) en el sistema
Dependencias	El usuario debe iniciar seccion.
Descripcion	<p>El sistema podra realizar estas funciones en varias ventanas.</p> <p>Actualizar Modificar Visualizar Borrar Según en el contexto que la empresa lo requiera.</p>
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-010	El sistema lista las opciones de distritos disponibles.
Dependencias	<p>El usuario debe iniciar seccion.</p> <p>El usuario selecciona la pestaña de personas.</p> <p>El usuario selecciona el boton "Añadir persona".</p>
Descripcion	<p>El sistema habre una ventana "Registrar un nuevo miembro".</p> <p>El sistema mostrara varios opciones de distritos del departamento.</p>
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-011	El sistema registra los datos del usuario en la base de datos.
Dependencias	El usuario debe haber pasado por el proceso de registro de usuario.
Descripcion	El sistema recopila todos los datos ingresados del usuario en la base de datos.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-012	El sistema habre uns ventana de interfaz de usuario, de acuerdo al tipo de usuario.
Dependencias	El usuario debe iniciar seccion.
Descripcion	<p>El sistema reconocera el tipo de usuario que inicio seccion.</p> <p>El sistema mostrara mas pestañas de opciones.</p>
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

RF-013	El sistema debe permitir iniciar seccion mediante correo electronico y contraseña
Dependencias	El usuario debe estar registrado al sistema. El usuario debe dirigirse a la ventana "Logueate para ingresar"
Descripcion	El sistema reconocera un correo electronico.
	El sistema encripta la contraseña
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno
RF-014	El sistema debe mostrar la lista de los usuario existente.
Dependencias	El usuario debe iniciar seccion. El usuario debe dirigirse a la ventana "Selecciona la pestaña de personas"
Descripcion	El sistema recarga y muestra la lista con los usuarios existentes y nuevos.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno
RF-015	El sistema debe permitir el ingresar datos obligatoriamente.
Dependencias	El usuario debe iniciar seccion.
Descripcion	El sistema debe obligar al usuario colocar todos los datos requeridos. En los diferentes procesos que requieran datos de usuario.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno
RF-016	El sistema debe contar con un CRUD en la ventana "Registro citas"
Dependencias	El usuario debe iniciar seccion. El usuario debe dirigirse a la ventana "Registro de citas"
Descripcion	El sistema muestra el registro de cita para que el usuario pueda reservar.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno
RF-017	El sistema debe contar con un CRUD en la ventana "Nueva persona"
Dependencias	El usuario debe iniciar seccion. El usuario debe dirigirse a la ventana "Nueva persona"
Descripcion	El sistema muestra nueva persona
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguno

II. Requisitos No Funcionales

RN-001	Rendimiento [REQREND]:
Dependencias	Ninguna
Descripción	Los registros de usuarios, consultas, modificaciones de cuentas y reservas se procesarán rápidamente.
Importancia	Alto
Estabilidad	Alta
Comentarios	Ninguna

RN-002	Escalabilidad[REQESCA]:
Dependencias	Ninguna
Descripción	El sistema debe ser capaz de escalar hacia arriba y hacia abajo según disposición.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-003	Seguridad [REQSEG]:
Dependencias	Ninguna
Descripción	El sistema deberá ser íntegro y protegido contra el acceso no autorizado.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-004	Confiabilidad [REQCON]:
Dependencias	Ninguna
Descripción	El sistema bloqueará temporalmente la cuenta de un usuario si el usuario comete más de 3 intentos fallidos al ingresar a su cuenta.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-005	Disponibilidad [REQDISP]:
Dependencias	Ninguna
Descripcion	La aplicación debe estar disponible las 24 horas del dia y los 7 dias de la semana.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-006	Usabilidad [REQUSA]:
Dependencias	Ninguna
Descripcion	El sistema deberá ser intuitivo y de interfaz amigable para nuestros clientes y personal de SPA UTP.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-007	Mantenimiento [REQUMAN]:
Dependencias	Ninguna
Descripcion	El sistema debe ser facil de mantener y actualizar.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-008	Compatibilidad [REQUCON]:
Dependencias	Ninguna
Descripcion	El sistema solo podra funcionar sobre navegar Web: Google Chrome estable 107, 106, 105 y Firefox Beta 97.0, 96.0, 95.0 El sistema debe ser compatible con otros sistemas.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-009	Fiabilidad [REQFIA]:
Dependencias	Ninguna
Descripcion	El sistema debe ser confiable y cumplir con los requisitos del usuario.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-010	Portatilidad [REQPORT]:
Dependencias	Ninguna
Descripción	El sistema deberá de ser utilizado en diferentes equipos de cómputo o móviles que tengan instalado un navegador Chrome, Mozilla o Edge para la página web y Windows 10 o superior para la aplicación de escritorio.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

RNF-011	Actuacion [REQACTU]:
Dependencias	Ninguna
Descripción	El sistema debe poder manejar el numero requerido de usuarios sin ninguna degradacion en el rendimiento.
Importancia	Alta
Estabilidad	Alta
Comentarios	Ninguna

2. DISEÑO DEL SISTEMA

I. Casos de Uso

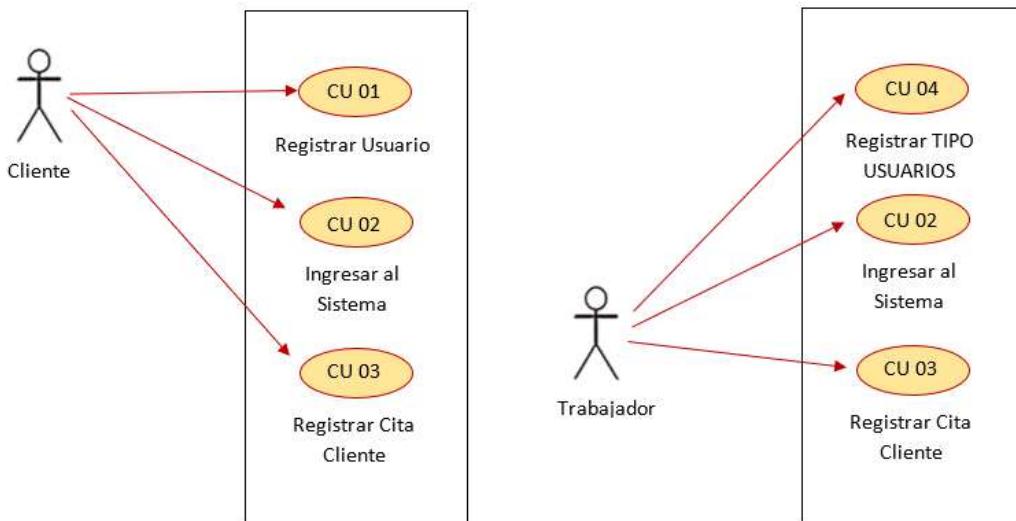


Figura: Diagrama de Caso de uso

- **CASO DE USO 1 : REGISTRAR USUARIO**

CU 01	REGISTRAR USUARIO	
Breve descripción:	El usuario puede ingresar sus datos para registrarse en el sistema. El trabajador puede ingresar los datos del cliente para registrarlos en el sistema	
Actores principales	Usuario	
Actores Secundarios	Trabajador	
Pre-condición	Ninguno	
Flujo Principal	El usuario	El sistema
	<ol style="list-style-type: none"> 1. El usuario Ingresar Datos de Usuario (Nombre, Apellido, DNI, Teléfono, Dirección) 2. El Usuario selecciona la opción de elegir su distrito. 4. El usuario selecciona el distrito correspondiente. 5. Ingresa su Correo y su contraseña 6. Selecciona “REGISTRAR” 	<ol style="list-style-type: none"> 3. El sistema lista las opciones de distritos disponibles. 7. El Sistema registra los datos del usuario en la base de datos. 8. El sistema redirige al usuario a la página de logeo (Ingreso de usuario).
Flujo Secundario (Alternativo)	Ninguno	

The screenshot shows a user registration form for 'SRA SUTR'. The main form on the left contains fields for Nombres, Apellidos, DNI, Teléfono, Dirección, Email, Contraseña, and Repita la contraseña. It also includes a checkbox for accepting terms and conditions and a 'Registrar' button. Below the form is a link to 'Ya tengo una cuenta'. To the right, a modal window titled 'Alto Selva Alegre' lists various locations: Arequipa, Cayma, Cerro Colorado, Characato, Chiguata, Jacobo Hunter, Jose Luis Bustamante y Rivero, La Joya, Mariano Melgar, Miraflores, Mollebaya, Paucarpata, Pocsi, Polobaya, Quequeña, Sabandia, Sachaca, and San Juan de Siguas. This modal has its own set of registration fields and a 'Registrar' button.

Registrar un nuevo miembro

Nombres

Apellidos

DNI

Teléfono

Dirección

Alto Selva Alegre

Email

Contraseña

Repita la contraseña

Acepto los **términos y condiciones**

Registrar

Ya tengo una cuenta

Alto Selva Alegre

Arequipa
Cayma
Cerro Colorado
Characato
Chiguata
Jacobo Hunter
Jose Luis Bustamante y Rivero
La Joya
Mariano Melgar
Miraflores
Mollebaya
Paucarpata
Pocsi
Polobaya
Quequeña
Sabandia
Sachaca
San Juan de Siguas

Alto Selva Alegre

Email

Contraseña

Repita la contraseña

Acepto los **términos y condiciones**

Registrar

Ya tengo una cuenta

Figuras: Registro de Usuario

- **CASO DE USO 2 : INGRESAR AL SISTEMA**

CU 02	INGRESAR AL SISTEMA	
Breve descripción:	El sistema permite el ingreso a los accesos de acuerdo al tipo de usuario	
Actores principales	Trabajador, Cliente	
Actores Secundarios	Ninguno	
Pre-condición	El trabajador debió logearse correctamente y debe tener al usuario registrado previamente	
Flujo Principal	El usuario	El sistema
	1. El Trabajador o Usuario ingresan su correo y contraseña	<ol style="list-style-type: none"> 2. El sistema valida si es un correo correcto. 3. El sistema valida si está registrado en el sistema. 4. El sistema abre la ventana de interface de usuario, de acuerdo al tipo de usuario se mostraran más pestañas de opciones.
Flujo Secundario (Alternativo)	Ninguno	



Figura: Ingresar al Sistema

- **CASO DE USO 3 : REGISTRAR CITA**

CU 03	REGISTRAR CITA	
Breve descripción:	El sistema muestra y permite crear una cita nueva para un usuario.	
Actores principales	Trabajador	
Actores Secundarios	Usuario(Cliente)	
Pre-condición	El trabajador debió logearse correctamente y debe tener al usuario registrado previamente	
Flujo Principal	El usuario 1. El Trabajador selecciona la pestaña de citas programadas. 3. El Trabajador selecciona el “Botón añadir nueva Cita”. 5. El Trabajador selecciona la fecha y hora de la cita del cliente. 6. El trabajador selecciona “Guardar”	El sistema 2. El sistema Lista las citas ya registradas mostrando la fecha y hora de la cita, así como el estado y el nombre de quien es el cliente que solicito la cita. 4. El sistema abre una ventana que le da la opción de ingresar la fecha y la hora de cita y seleccionar al cliente 7. El sistema guarda los datos y recarga la lista con la nueva cita agregada.
Flujo Secundario (Alternativo)	Ninguno	

Citas Programadas							
Show	entries	Search:					
Fecha Cita	Hora Programada	Nombres	Apellidos	Estado	Creado	Modificado	Acciones
No data available in table							
Showing 0 to 0 of 0 entries							

Figura: Citas Programadas

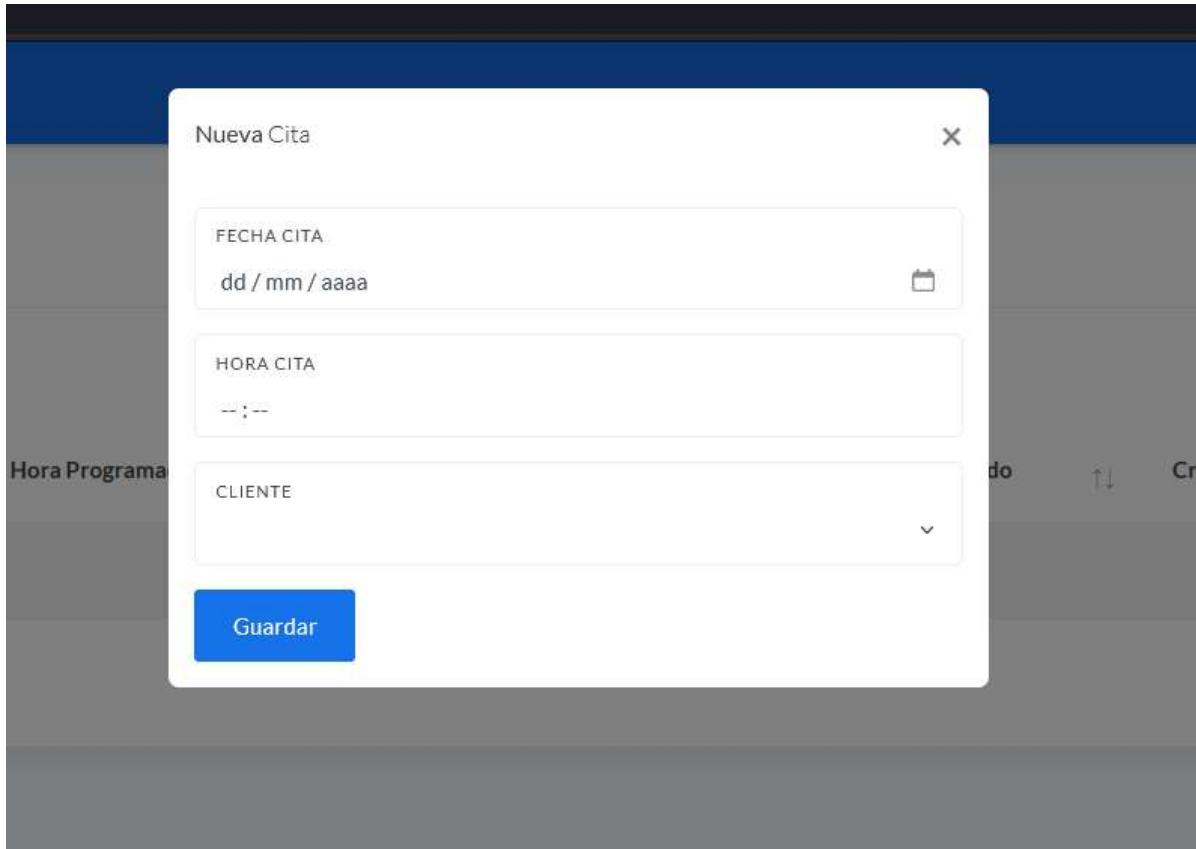


Figura: Registro de Nueva Cita

- **CASO DE USO 4 : REGISTRAR TIPO USUARIOS**

CU 03	REGISTRAR TIPO DE USUARIO
Breve descripción:	El sistema permite al trabajador registrar un usuario y elegir qué tipo de usuario es (Administrador, Cliente, Empleado)
Actores principales	Administrador
Actores Secundarios	Empleado
Pre-condición	El Administrador debió ingresar correctamente al sistema.

Flujo Principal	El usuario	El sistema
	<p>1. El Administrador selecciona la pestaña de personas</p> <p>3. El Administrador selecciona el botón “Añadir Persona”.</p> <p>5. El Administrador Ingresar Datos de la persona a registrar (Nombre, Apellido, DNI, Teléfono, Dirección)</p> <p>6. El Administrador selecciona la opción de elegir su distrito.</p> <p>8. El Administrador selecciona el distrito correspondiente.</p> <p>9. El Administrador selección la opción de elegir tipo de persona</p> <p>11. El Administrador selecciona el tipo de usuario correspondiente.</p> <p>12. Ingrasa su Correo y su contraseña, selecciona “REGISTRAR”</p>	<p>2. El sistema Lista las Personas registradas en el sistema, ya sean cliente, empleado o Administrador.</p> <p>4. El sistema Abre una ventana y muestra los datos a ingresar para registrar.</p> <p>7. El sistema lista las opciones de distritos disponibles.</p> <p>10. El sistema lista las opciones de tipos de usuario disponibles.</p> <p>13. El sistema guarda los datos y recarga la lista con el nuevo usuario.</p>
Flujo Secundario (Alternativo)	Ninguno	

Nueva Persona

NOMBRES

APELLIDOS

DNI

TELÉFONO

DIRECCIÓN

DISTRITO
Alto Selva Alegre

EMAIL

PASSWORD

TIPO DE PERSONA
Administrador

Guardar

Figura: Registro de usuario y asignación de tipo

Personas												Multi-Persons
Show: <input type="text"/> entries												Search: <input type="text"/>
Nombres	Apellidos	DNI	Teléfono	Dirección	Email	Estado	Tipo de Persona	Creado	Modificado	Acciones		
No data available in table												
Showing 0 to 0 of 0 entries												

Figura: Usuarios Registrados

II. Diseño de Clases

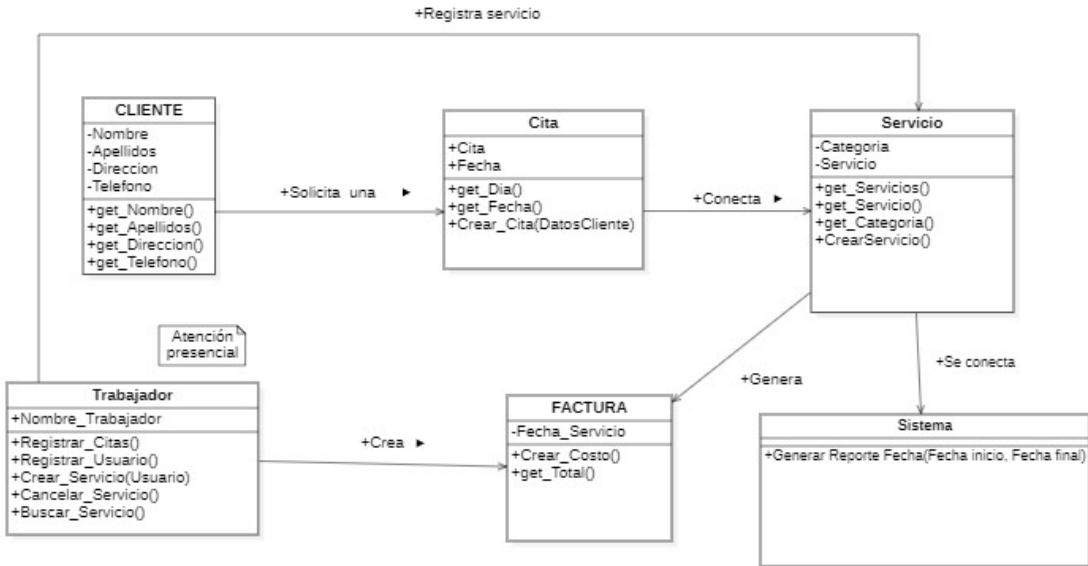


Figura: Diseño de Clases

III. Diseño de la Base de datos

• Modelo Entidad – Relación

El modelo entidad relación que se muestra resume el primer planteamiento de relaciones de entidades junto con su descripción de atributos a considerar. Se considera a Persona como entidad que puede asignar una cita que esta contiene un servicio. Asimismo, la persona puede generar el comprobante para proceder con registro final. En el trabajo integrador, se presentan actores de administrador, técnicos y clientes. Los cuales son descritos por persona, mediante tipo de persona como atributo que sirve como clasificación de actor.

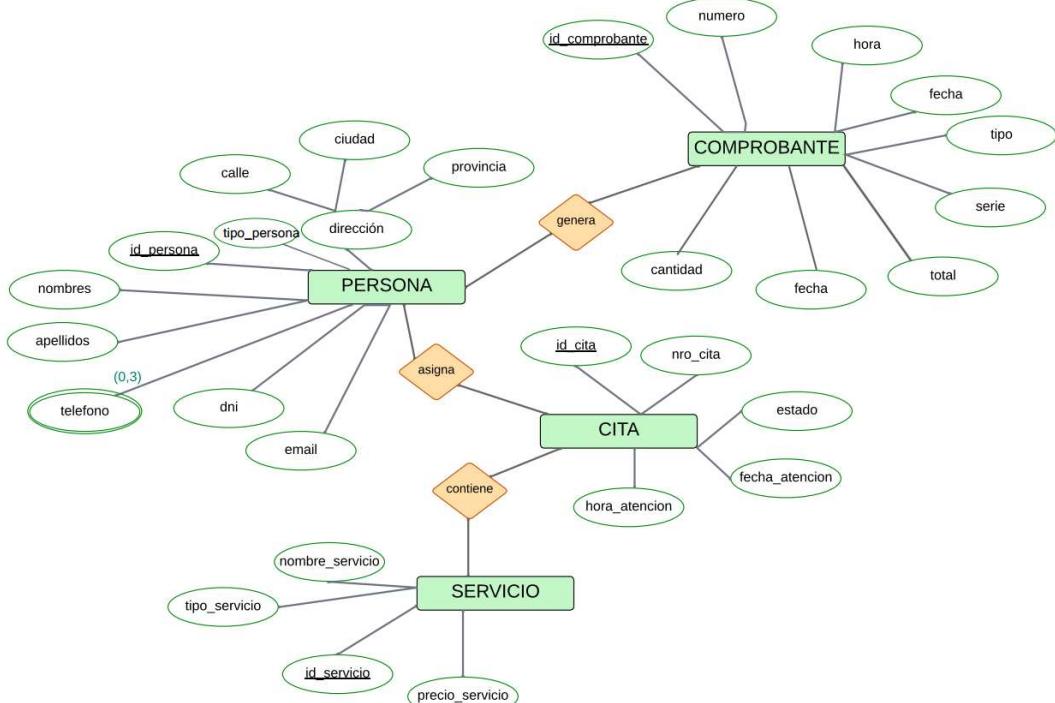


Figura: Modelo Entidad-Relación

- **Diseño Físico**

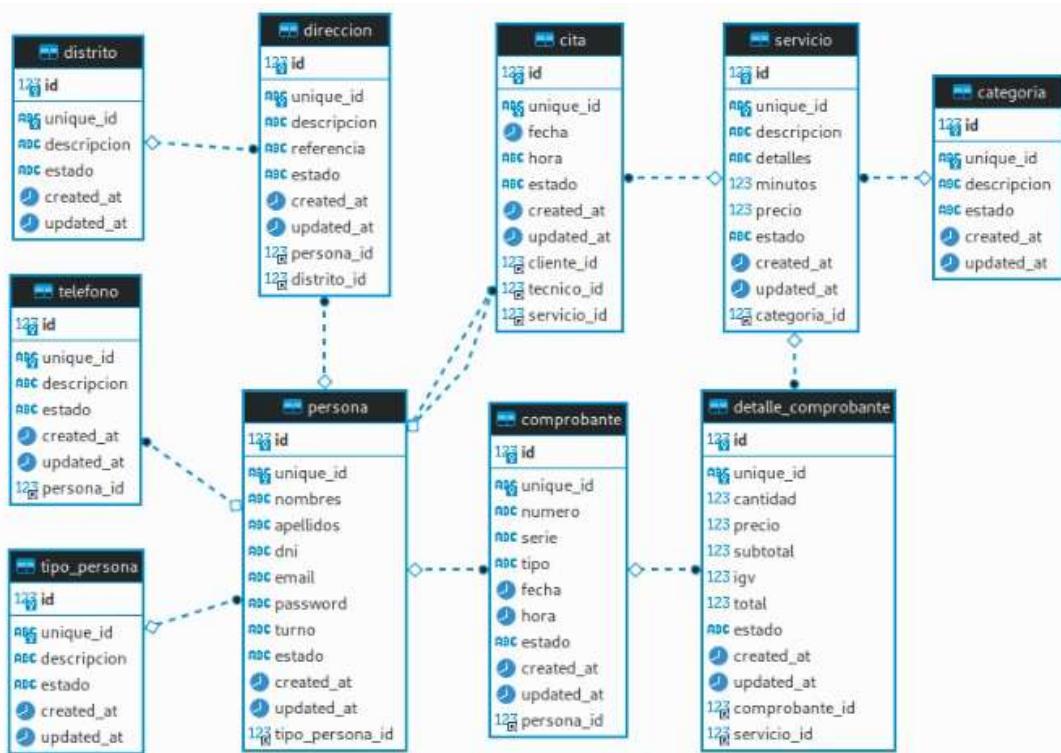


Figura: Diseño Físico BD.

IV. Código Fuente

- **Organización de los paquetes.-**

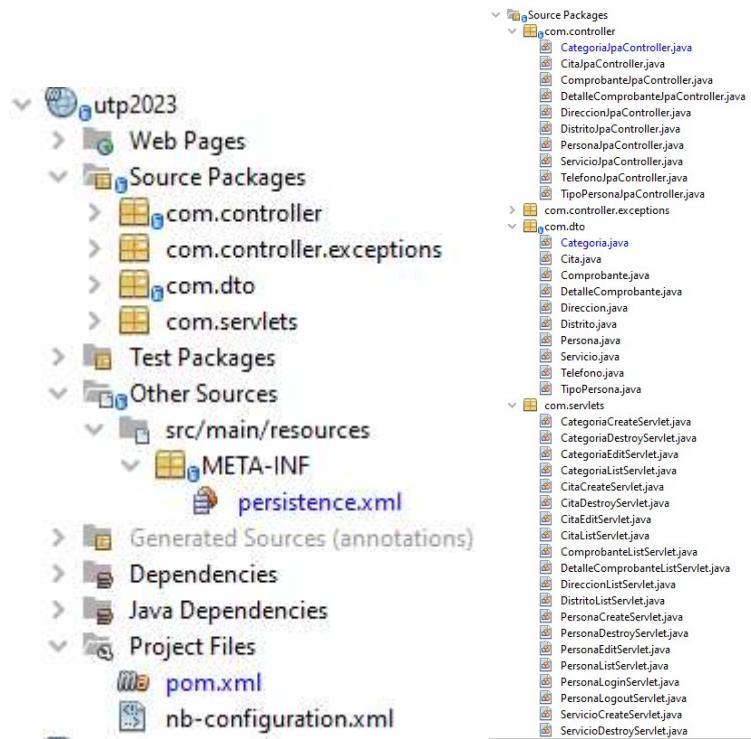


Figura: Captura de paquetes del proyecto en Netbeans.

- **Modelo MVC.-**

En este trabajo la propuesta de arquitectura es separar los códigos de acuerdo a distintas responsabilidades. El patrón MVC se usa como referencia ya que también estaremos trabajando con otras aplicaciones que servirán para complementar el proyecto. Lo utilizamos a fin de tener un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

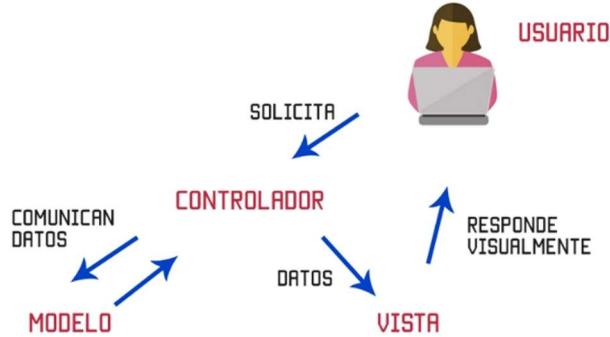


Figura: Modelo MVC.

En este trabajo, el proyecto desarrollado en IDE Netbeans se divide en MVC donde tenemos nuestro jsp, controller, dto y servlets.

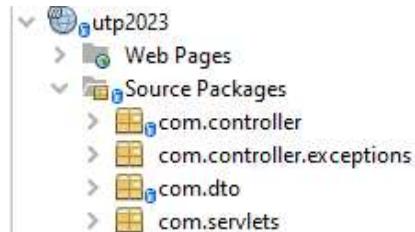


Figura: Referencial proyecto Netbeans.

- El jsp serán todas nuestras pantallas de visualización, mientras nuestros controller asumirán el rol de sentencias de conexión de la base de datos, donde tendremos varios métodos que realiza un CRUD.

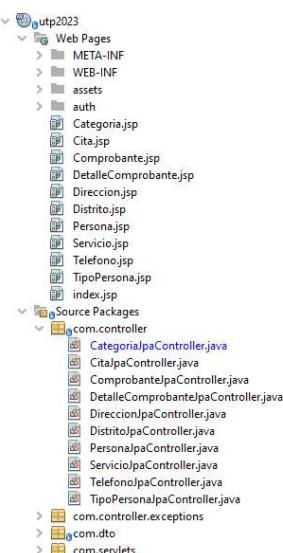


Figura: Archivos JSP y Controller

```

27 |     public void create(Cita cita) {
28 |         EntityManager em = null;
29 |         try {
30 |             em = getEntityManager();
31 |             em.getTransaction().begin();
32 |             Persona clienteId = cita.getClienteId();
33 |             if (clienteId != null) {
34 |                 clienteId = em.getReference(clienteId.getClass(), clienteId.getId());
35 |                 cita.setClienteId(clienteId);
36 |             }
37 |             Servicio servicioId = cita.getServicioId();
38 |             if (servicioId != null) {
39 |                 servicioId = em.getReference(servicioId.getClass(), servicioId.getId());
40 |                 cita.setServicioId(servicioId);
41 |             }
42 |             Persona tecnicoId = cita.getTecnicoId();
43 |             if (tecnicoId != null) {
44 |                 tecnicoId = em.getReference(tecnicoId.getClass(), tecnicoId.getId());
45 |                 cita.setTecnicoId(tecnicoId);
46 |             }
47 |             em.persist(cita);
48 |             if (clienteId != null) {
49 |                 clienteId.getCitaList().add(cita);
50 |                 clienteId = em.merge(clienteId);
51 |             }
52 |             if (servicioId != null) {
53 |                 servicioId.getCitaList().add(cita);
54 |                 servicioId = em.merge(servicioId);
55 |             }
56 |             if (tecnicoId != null) {
57 |                 tecnicoId.getCitaList().add(cita);
58 |                 tecnicoId = em.merge(tecnicoId);
59 |             }
60 |             em.getTransaction().commit();
61 |         } finally {
62 |             if (em != null) {
63 |                 em.close();
64 |             }
65 |         }
66 |     }

```

Figura: Programación de crear cita (paquete controller)

- En nuestro dto tendremos todas las clases que serán nuestros objetos con atributos, constructor y sus métodos get y set y en los servlets tendremos nuestro proceso de manejo de datos para cumplir con la tarea que envía nuestro jsp.

```

@javax.persistence.Entity
@javax.persistence.AdditionalCriteria("this.estado <> 'eliminado'")
@javax.persistence.Table(name = "persona")
@javax.persistence.NamedQueries({
    @javax.persistence.NamedQuery(name = "Persona.findAll", query = "SELECT p FROM Persona p"))
public class Persona implements Serializable {

    private static final long serialVersionUID = 1L;
    @javax.persistence.Id
    @javax.persistenceGeneratedValue(strategy = GenerationType.IDENTITY)
    @javax.persistence.Basic(optional = false)
    @javax.persistence.Column(name = "id")
    private Long id;
    @javax.persistence.Basic(optional = false)
    @javax.persistence.NotNull
    @javax.persistence.Size(min = 1, max = 36)
    @javax.persistence.Column(name = "unique_id")
    private String uniqueId;
    @javax.persistence.Size(max = 255)
    @javax.persistence.Column(name = "nombres")
    private String nombres;
    @javax.persistence.Size(max = 255)
    @javax.persistence.Column(name = "apellidos")
    private String apellidos;
}

```

Figura: Paquetes dto. Visualización de atributos Persona.java

- Cada módulo tiene los sevlets llamados CreateServlets, DestroyServlets, editServlets y listServlets que instanciaran a los objetos del dto y controller para cumplir con la tarea correspondiente que el jsp manda este Servet devuelve la tarea correspondiente mediante un método processRequest al jsp.

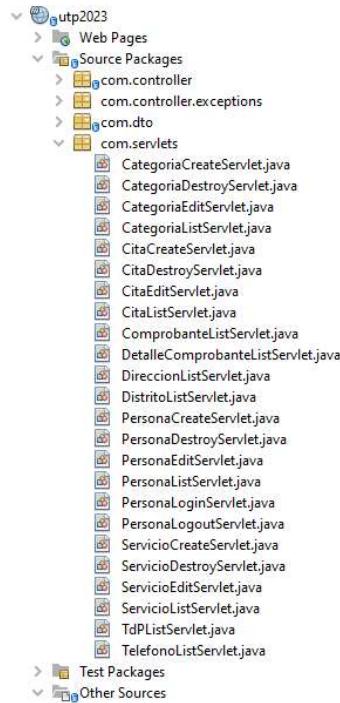


Figura: Servlets package

```

1 package com.servlets;
2
3 import com.controller.CategoriaJpaController;
4 import com.dto.Categoria;
5 import java.io.IOException;
6 import javax.persistence.EntityManagerFactory;
7 import javax.persistence.Persistence;
8 import javax.servlet.ServletException;
9 import javax.servlet.annotation.WebServlet;
10 import javax.servlet.http.HttpServlet;
11 import javax.servlet.http.HttpServletRequest;
12 import javax.servlet.http.HttpServletResponse;
13
14 @WebServlet(name = "CategoriaCreateServlet", urlPatterns = {"/*CategoriacreateServlet"})
15 public class CategoriaCreateServlet extends HttpServlet {
16
17     /**
18      * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
19      * methods.
20      *
21      * @param request servlet request
22      * @param response servlet response
23      * @throws ServletException if a servlet-specific error occurs
24      * @throws IOException if an I/O error occurs
25      */
26     protected void processRequest(HttpServletRequest request, HttpServletResponse response)
27     throws ServletException, IOException {
28         response.setContentType("text/html;charset=UTF-8");
29
30         System.out.println("Bandera Categoria Create Servlet");
31         try {
32             EntityManagerFactory emf = Persistence.createEntityManagerFactory("com.spa_utp2023_war_1.0PU");
33
34             CategoriaJpaController jpacObject = new CategoriaJpaController(emf);
35             Categoria categoria = new Categoria();

```

Figura: Servlet Categoria

ANEXOS

1. REPOSITORIO GITHUB

Archivos del Proyecto se encuentran en repositorio de GitHub:

Link: https://github.com/AnthonyQuispeHuamani/SPA_UTP

2. CRONOGRAMA

ETAPA1

CRONOGRAMA DE ACTIVIDADES						
Numero de trabajo	Nombre de tarea	Duracion dias	Comienzo	Fin	Predecesor a del trabajo	Responsable
	Sistema SPA UTP	126				
	INICIO		4 de Abril de 2023	21 de Julio de 2023		
T1	<i>Definicion del problema</i>	5	3 de Abril de 2023	8 de Abril de 2023		Analista y desarrollador de Sistemas
T1.1	Descripcion del problema	1	3 de Abril de 2023	4 de Abril de 2023		Todos
T1.2	Avance de Project Charter	1	4 de Abril de 2023	5 de Abril de 2023	T1.1	Todos
T1.3	Modelo Lean Canvas	1	5 de Abril de 2023	6 de Abril de 2023	T1.2	Todos
T1.4	Diagrama de procesos	1	6 de Abril de 2023	7 de Abril de 2023	T1.3	Todos
T1.5	Planteamientos alternativos de solucion	1	7 de Abril de 2023	8 de Abril de 2023	T1.4	Todos
T2	<i>Definicion de objetivos</i>	4	10 de Abril de 2023	14 de Abril de 2023	T1	
T2.1	Objetivo General	1	10 de Abril de 2023	11 de Abril de 2023		Analista y desarrollador de Sistemas-Administrador del proyecto
T2.2	Objetivos Especificos	1	11 de Abril de 2023	12 de Abril de 2023	T2.1	Analista y desarrollador de Sistemas-Administrador del proyecto
T2.3	Alcances y limitaciones	1	12 de Abril de 2023	13 de Abril de 2023	T2.2	Analista y desarrollador de Sistemas-Administrador del proyecto
T2.4	Justificacion	1	13 de Abril de 2023	14 de Abril de 2023	T2.3	Analista y desarrollador de Sistemas-Administrador del proyecto
T3	<i>Analisis y diseño del sistema</i>	6	17 de Abril de 2023	23 de Abril de 2023	T2	
T3.1	Requisitos funcionales	1	17 de Abril de 2023	18 de Abril de 2023		Analista y desarrollador de Sistemas
T3.2	Requisitos no funcionales	1	18 de Abril de 2023	19 de Abril de 2023	T3.1	Analista y Desarrollador de Sistemas
T3.3	Diseño del sistema	1	19 de Abril de 2023	20 de Abril de 2023	T3.2	Desenñador Web-Desarrollador Front-End
T3.4	Casos de uso	1	20 de Abril de 2023	21 de Abril de 2023	T3.3	Analista y desarrollador de Sistemas-Desarrollador Front-End
T3.5	Diseño de clases	1	21 de Abril de 2023	22 de Abril de 2023	T3.4	Desarrollador de base de datos
T3.6	Diseño de base de datos	1	22 de Abril de 2023	23 de Abril de 2023	T3.5	Desarrollador de base de datos

	ETAPA1				
<i>Registro de citas</i>					
T4.1	<i>Diseño de paginas de registro</i>	8	2 de Mayo de 2023	10 de Mayo de 2023	
T4.1.1	Pagina de registro de usuario	2	2 de Mayo de 2023	4 de Mayo de 2023	Disenñador Web -Desarrollador Front-End
T4.1.2	Pagina de registro de trabajador	2	4 de Mayo de 2023	6 de Mayo de 2023	T4.1.1
T4.1.3	Pagina de reservacion de citas	2	6 de Mayo de 2023	8 de Mayo de 2023	T4.1.2
T4.1.4	Pagina de gestion de servicios	2	8 de Mayo de 2023	10 de Mayo de 2023	T4.1.3
T4.2	<i>Creacion yconexion de la base de datos</i>	6	11 de Mayo de 2023	17 de Mayo de 2023	
T4.2.1	Creacion de la base de datos	3	11 de Mayo de 2023	14 de Mayo de 2023	Desarrollador de base de datos
T4.2.2	Conección de la base de datos	3	14 de Mayo de 2023	17 de Mayo de 2023	T4.2.1
T4.3	<i>Funciones de registro de clientes</i>	5			
T4.3.1	Funcion registro de clientes	3			Desarrollador Back-end-Desarrollador de base de datos
T4.3.2	Funcion filtro de registro de clientes	2			T4.3.1
T4.4	<i>Funcion de registro de personal</i>	4			
T4.4.1	Funcion registro de personal	2			Desarrollador Back-end-Desarrollador de base de datos
T4.4.2	Funcion filtro de registro de personal	2			T4.4.1
T4.5	<i>Funciones de registro de citas</i>	4			
T4.5.1	Funcion registro de citas	2			Desarrollador Back-end-Desarrollador de base de datos
T4.5.2	Funcion filtro de citas (fechas,reservaciones)	2			T4.5.1
T4.6	<i>Funciones de registro de servicios</i>	4			
T4.6.1	Funcion de registro de nuevo servicios	2			Desarrollador Back-end-Desarrollador de base de datos
T4.6.2	Funcion filtro de tipos de servicios-categorias	2			T4.6.1
T4.7	<i>Edicion de datos y mantenimiento de personal (Eliminar,Editar)</i>	4	18 de Mayo de 2023	20 de Mayo de 2023	
T4.7.2	Integracion de funcion de editar personal	2	19 de Mayo de 2023	20 de Mayo de 2023	T4.7.1
T4.8	<i>Edicion de datos y mantenimiento de clientes (Eliminar,Editar)</i>	5	20 de Mayo de 2023	22 de Mayo de 2023	
T4.8.1	Integracion de funcion de eliminar cliente	3	20 de Mayo de 2023	21 de Mayo de 2023	Desarrollador Back-end
T4.8.2	Integracion de funcion de editar cliente	2	21 de Mayo de 2023	22 de Mayo de 2023	T4.8.1
T4.9	<i>Edicion de datos y mantenimiento de citas (Eliminar,Editar)</i>	4	21 de Mayo de 2023	23 de Mayo de 2023	
T4.9.1	Integracion de funcion de cancelar cita	2	21 de Mayo de 2023	22 de Mayo de 2023	Desarrollador Back-end
T4.9.2	integracion de funcion de editar cita(Fecha)	2	22 de Mayo de 2023	23 de Mayo de 2023	T4.9.1
T4.10	<i>Edicion de datos y mantenimiento de servicios(Eliminar,Editar)</i>	4	23 de Mayo de 2023	25 de Mayo de 2023	
T4.10.1	Integracion de funcion de eliminar servicio	2	23 de Mayo de 2023	24 de Mayo de 2023	Desarrollador Back-end
T4.10.2	Integracion de funcion de editar servicio	2	24 de Mayo de 2023	25 de Mayo de 2023	T4.10.1
T4.11	<i>Ingreso del usuario al sistema(Logeo)</i>	6	25 del Mayo del 2023	29 de Mayo de 2023	
T4.11.1	Creacion de pagina del logeo	2	25 de Mayo de 2023	27 de Mayo de 2023	Disenñador Web
T4.11.2	Conección de la base de datos	2	27 de Mayo de 2023	28 de Mayo de 2023	T4.11.2
T4.11.3	Creacion de filtros de logeo(password,email,etc)	2	28 de Mayo de 2023	29 de Mayo de 2023	T4.11.3

3. MATRIZ DE TRAZABILIDAD

REQUISITOS	CASOS DE USO											
	Código	Descripción	CU1	CU2	CU3	CU4	CU5	CU6	CU7	CU8	CU9	CU10
	R1	El sistema debe permitir registrar usuarios										
	R2	El sistema debe permitir Actualizar o modificar los datos del usuario										
	R3	El sistema debe permitir ingresar (Logeo)										
	R4	El sistema debe permitir Registrar una cita seleccionando al usuario										
	R5	El sistema debe Mostrar los reporte de las citas solicitadas										
	R6	El sistema debe dar permisos particulares al administrador										
	R7	El sistema debe presentar informes en formato PDF										
	R8											

Figura: Matriz de Trazabilidad

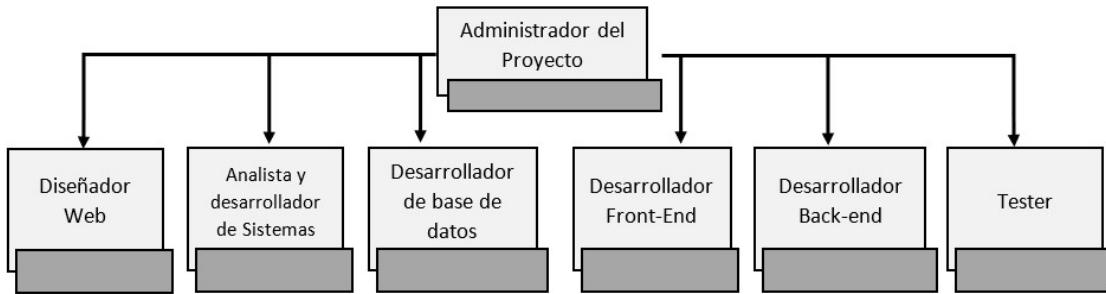
4. PRESUPUESTO

CI01-UTP-P001		DESCRIPCIÓN RECURSOS Y PRESUPUESTOS EJECUCIÓN : SISTEMA INTEGRADO SPA UTP			
Nro	Tipo	RECURSO	DESCRIPCIÓN	VALOR	CANT. TOTAL
1	Software	StartUML	Herramienta de UML	S/ 0	4 S/ 0
2	Software	Java (JDK) 20 Windows X64	Software necesario para programar en Java	S/ 0	4 S/ 0
3	Software	MySQL WorkBench 8.0	Gestor de base de datos	S/ 0	4 S/ 0
4	Software	WildFly, GlassFish o Tomcat v9	Servidor de aplicaciones java EE	S/ 0	4 S/ 0
5	Software	Apache Netbeans v16	Entorno de desarrollo	S/ 0	4 S/ 0
6	Software	Visual Studio Code	Editor de código apoyo	S/ 0	4 S/ 0
7	Aplicación	Zoom / Meet / Discord / WhatsApp	Reuniones / Información del equipo	S/ 0	4 S/ 0
8	Aplicación	GitHub	Sistema de control de versiones	S/ 0	4 S/ 0
9	Software	MsProject	Diagrama de Gantt	S/ 0	1 S/ 0
10	Online	Draw.io	Diagramas de procesos	S/ 0	1 S/ 0
12	Equipo de Trabajo	Administrador de Proyecto	Encargado del proyecto	S/ 11.000	1 S/ 11.000
13	Equipo de Trabajo	Diseñador Web	Desarrollo de página Web	S/ 4.600	1 S/ 4.600
14	Equipo de Trabajo	Analista y desarrollador de Sistemas	Diseñador y Documentador del sistema	S/ 4.200	1 S/ 4.200
15	Equipo de Trabajo	Desarrollador de Base de Datos	Desarrollador y diseñador de la base de datos	S/ 4.500	1 S/ 4.500
16	Equipo de Trabajo	Desarrollador Front-END	Desarrollador Entorno Visual	S/ 3.200	1 S/ 3.200
17	Equipo de Trabajo	Desarrollador Back-END	Configuración funcionamiento	S/ 5.000	1 S/ 5.000
18	Equipo de Trabajo	Tester	Pruebas de software	S/ 1.100	1 S/ 1.100
19	Hardware	PC core i5 16GB RAM, 120 GB ROM	Equipo para el desarrollo S.O. Windows 10 X64	S/ 1.000	5 S/ 5.000
20	Conectividad	Internet Fibra Óptica	100Mbps	S/ 100	5 S/ 500
21	Otros	Instalaciones de desarrollo	Alquiler	S/ 250	5 S/ 1.250
22	Otros	Energía eléctrica y Agua	Consumo durante el proyecto	S/ 150	5 S/ 750
23	Otros	Muebles: Escritorio, Impresora y demás	Durante el Proyecto	S/ 600	5 S/ 3.000
24	Otros	Ejecución de reportes físico	Documentación	S/ 100	5 S/ 500
				Total	S/ 44.600

Nota: Se considera presupuesto para Equipo de Trabajo pago por todo el proyecto.

Figura: Presupuesto

5. EQUIPO DE TRABAJO



I. Descripción

Especialistas	Descripción
Administrador del proyecto	<ul style="list-style-type: none"> - Seguimiento de tareas. - Documentación del proceso. - Gestión de recursos. - Organización de las reuniones del grupo de trabajo.
Analista y desarrollador de Sistemas	<ul style="list-style-type: none"> - Planificar la actividad o trabajo de análisis y diseño de sistemas. - Diseño de los casos de uso. Investigar los diferentes problemas de la empresa. - Diseñar o adaptar un sistema informático para mejorar el funcionamiento de la empresa. - Examinar cuidadosamente el nuevo sistema. - Asegurar que el nuevo sistema cumpla con sus funciones correspondientes. - Responsable de todo el proceso de análisis de las necesidades de la empresa.
Desarrollador de base de datos	<ul style="list-style-type: none"> - Define la estructura lógica y física de una o más bases de datos. - Diseñar bases de datos eficaces, fiables y estables. - Modificar la base de datos según las solicitudes. - Solucionar problemas y errores de funcionamiento de la base de datos. - Adaptará la base de datos a la empresa y a sus necesidades.
Desarrollador Front-End	<ul style="list-style-type: none"> - Encargado de programar el navegador de una web. - Responsable de ver el desarrollo visual de la página web. - Controlar tipografía, formas del diseño y la interactividad - Adaptar el diseño web a diversos dispositivos y navegadores. - Es el encargado de trabajar la arquitectura interna de la web.
Desarrollador Back-end	<ul style="list-style-type: none"> - Es el encargado de trabajar la arquitectura interna de la web. - Ocuparse de la seguridad de la información al momento de ser utilizada o solicitada por el sistema. - Desarrollo de la lógica del negocio del sistema. - Gestionar e integrar la base de datos necesaria. - Escribir códigos en diferentes lenguajes de programación. - Optimizar la página web para que rinda mejor y de forma más rápida.
Diseñador Web	<ul style="list-style-type: none"> - Diseñar las páginas web. - Desarrollar la página web y procesos - Sugiere formas de mejorar el proceso de manejo de las páginas web
Tester	<ul style="list-style-type: none"> - Planifica y lleva a cabo pruebas para comprobar si funcionan correctamente. - Detecta potenciales fallos. - Evalúa el funcionamiento general del software. - Sugiere formas de mejorar el software desarrollado.

CONCLUSIONES

RECOMENDACIONES

ANEXOS

PROJECT CHARTER

- 1. NOMBRE DE PROYECTO**
- 2. DESCRIPCIÓN DEL PROYECTO**

- 3. DEFINICIÓN DEL PRODUCTO**

3.1 CARACTERÍSTICAS DEL SISTEMA

- **3.2 ALCANCE**

 - **3.3 CONTEXTO DEL SISTEMA**
- #### **3.2.1 ENTORNO DE OPERACIÓN**
- #### **3.2.2 ENLACE REPOSITORIO**
- #### **4.1 OBJETIVOS DEL PROYECTO**
- **OBJETIVO GENERAL**
 - **OBJETIVO ESPECIFICOS**

 - **4.2 EQUIPO DE TRABAJO**

4.2.1. Administrador de Proyecto

4.2.2. Analista y diseñador de Sistemas

4.2.3. Desarrollador Front-End

4.2.4. Desarrollador Back-End

4.2.4. Tester

4.2.5 Desarrollador de Base de Datos

- **4.3 RECURSOS Y PRESUPUESTO**

PROBLEMA	SOLUCION	Proposición de Valor única	Ventaja especial	Segmento de clientes
<ul style="list-style-type: none"> Carencia de autoservicio, ya que, al atender por teléfono y mensajes vía WhatsApp, se observa una limitada capacidad de atención a numerosos clientes. Causando la perdida de potenciales clientes por la falta de atención 24/7. Mala gestión de datos, al usar archivos Excel para el registro de datos de los clientes y demás. Puede causar pérdidas de datos incongruencias y una limitada automatización. Falta de gestión de cobros. Se observa una carencia de un sistema de cálculo de costos de servicio. Los trabajadores calculan los precios a cobrar de forma personal y puede presentarse algún error causando perdida de ganancias 	<ul style="list-style-type: none"> Integrar un sistema de atención al cliente vía web, para que el cliente consulte, reserve o solicite una cita. Implementar un sistema de gestión de base de datos para la automatización y optimización de la gestión de datos de los clientes, reservas, citas, trabajadores, costos y servicios. Se automatizará la gestión de precios y cálculo de costos de los servicios por medio de la página web. 	<p>Un sistema claro y fácil de usar, para mejorar la atención al cliente y reduciendo el tiempo de atención, y así mejorando el control de los servicios.</p>	<ul style="list-style-type: none"> Una rápida atención y sencilla para nuestros clientes. No será necesario ir de forma presencial al local para solicitar una cita o preguntar por los servicios. Acceso a los servicios y sus ofertas y precios de forma rápida y disponible a cualquier hora 	Dirigido a empresas que se dediquen al rubro de servicios spa, servicios de relajación, tratamiento de belleza y terapias de salud.
Alternativas	Métricas claves <ul style="list-style-type: none"> WhatsApp business Sistema de control de reservas de atención Implementar un gestor de datos de clientes simple, datos de 		Canales <ul style="list-style-type: none"> Redes sociales. Internet por medio de la publicidad. 	Early Adopters <ul style="list-style-type: none"> Personas que tengan rutinas de trabajo excesivas. Personas que necesiten un tratamiento de

clientes y datos de servicio.	mes <ul style="list-style-type: none"> • Número de nuevos clientes al mes 		<ul style="list-style-type: none"> • Correos electrónicos. • Oficina principal 	belleza. <ul style="list-style-type: none"> • Personas que deseen un tratamiento de salud física.
Estructura de costes		Flujo de ingresos		
<ul style="list-style-type: none"> • Equipos informáticos y servicios: Computadoras e internet, Mantenimiento de página web y servicios de gestión de página web. • Costo de personal: Jefe de proyecto, analista de sistemas, programadores de base de datos, desarrolladores de Back-end y Front-end. • Servicios de marketing online, diseño de imágenes publicitarias y demás. 			<ul style="list-style-type: none"> • Costo de pago por el sistema web. • Costo de pago mensual por su soporte, mantenimiento y optimización e integración de nuevas funciones. 	

1.5 MODELO LEAN CANVAS