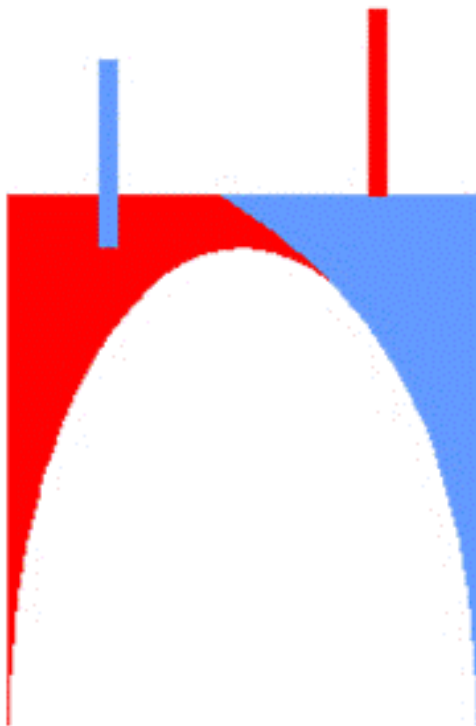


RESTRESV

Proyecto 2º DAM



Proyecto 2º DAM
Jorge Tomas Marco
Tutor: Raúl Olles Clavelo



Índice

1.Datos identificativos del proyecto	2
2.Resumen ejecutivo	2
Español.....	2
Ingles	Error! Bookmark not defined.
3.Datos Descriptivos	3
3.1Trabajo de investigación:	4
3.2Dificultad	6
3.3Creatividad e innovación	6
4.Desarrollo de proyecto	6
4.1 Acuerdo de proyecto	6
4.1.1 Generales	6
4.1.2 Requisitos Funcionales.....	6
4.1.3 Requisitos no funcionales	7
4.1.4 Especificaciones técnicas	7
4.2. Actividades y Cronología	8
Diagrama de Gantt	8
4.2.2Documento de análisis y diseño	10
4.2.2.3 Modelo Físico de Datos	19
4.2.3 Pruebas de sistema	27
4.2.4Documentacion para el cliente.....	27
4.3 Metodología	28
4.4 Recursos	29
4.5Dificultad del proyecto.....	29
4.6Trabajo de investigación	30
5.Presupuesto	30
5.1 Resumen ejecutivo.....	31
5.2 Pruebas del Sistema	31
6.Resultados obtenidos y conclusiones	31
7.Bibliografía	31



1.Datos identificativos del proyecto

2.Resumen ejecutivo

Español

La propuesta que plantea RestResv surge a causa del movimiento en el sector de la hostelería a realizar reservas con confirmación por correo electrónico y a su vez la mejora de organización y modernización de estas.

·En España cada año restaurantes pierden la capacidad de llenar porque los clientes acostumbra a no acudir a las reservas y no avisar, esto nos lleva a que estos restaurantes están perdiendo dinero por no tener una capacidad de organización suficiente para evitar estas situaciones.

Muchos restaurantes han buscado realizar un cobro obligatorio por adelantado para evitar estos casos, pero esto en muchos casos no es bien recibido por los clientes.

Esta aplicación además de evitar estas situaciones, nos daría mucha información sobre el servicio diario mejorando así la organización diaria. Tendría poco riesgo puesto que al incluir la página web y aplicación más personalizada a cada empresa nos daría una ventaja de cercanía con el restaurante, evitando así a las aplicaciones y páginas webs genéricas y mejorando la experiencia respecto a estas últimas.

En definitiva, existen muchas webs grandes en las cual el restaurante es uno más y no destaca o incluso se arriesga a ser criticado negativamente de manera injustificada como ya ha ocurrido en las reseñas de Google o en TripAdvisor. En este caso tendríamos una web y una aplicación vinculadas a una base de datos y así gestionar correctamente el restaurante.



Ingles

The proposal that RestResv raises arises because of the movement in the hostelry sector to make bookings with confirmation by email and in turn the improvement of organization and modernization of these.

In Spain every year restaurants lose the capacity to fill because customers don't usually go to reservations and not notify us, because of that these restaurants are losing money because they don't have sufficient organizational capacity to avoid these situations.

Many restaurants have look for solutions and forced people to make a mandatory payment in advance to avoid these cases, but this is not well received by customers in many cases.

This application, in addition to avoiding these situations, would give us a lot of information about the daily service as well as the daily organization. It would have little risk since by including the most personalized web page and application for each company it would give us an advantage of proximity to the restaurant, thus preventing generic applications and web pages and improving the experience.

In conclusion, there are many large websites in which the restaurant doesn't stand out or even risks being unjustifiably criticized negatively, as has already happened in Google or TripAdvisor reviews. In this case we have a website and an application linked to a database and therefore correctly manage the restaurant.



3.Datos Descriptivos

La empresa se dedica a realizar diferentes proyectos dependiendo de lo que piden los usuarios desde web hasta aplicaciones móviles y busca pequeñas empresas debido a su capacidad actual de trabajo, aunque dependiendo del proyecto se podría contratar gente o solicitar ayuda externa.

El proyecto se debe a la demanda de una organización más actual, y que permita coordinarse con una página web, además de la posibilidad de dar avisos a los clientes por correo electrónico. En este caso se busca cubrir la necesidad que tienen muchos restaurantes respecto a las reservas que finalmente no acuden al establecimiento. Se busca poder usar el correo electrónico a modo de aviso y confirmación de la reserva por medio de un link, desde el cual con una contraseña y correo podrán gestionar sus reservas.

La posibilidad de reservar, tendría dos posibilidades, una, la reserva por parte de web desde un formulario online, ubicado en una página web sencilla. Y por otra parte tendríamos la aplicación desde la que el restaurante podría crear las reservas de todos aquellos clientes que no quieran usar la web.

La reserva desde un formulario web, desde el que guardaremos los datos en una base de datos, utilizando el teléfono de contraseña y el correo de usuario para el usuario que crea la reserva, si este ya existiese simplemente se añadiría a su lista de reservas.

Luego desde la aplicación la posibilidad de reservar funciona igual, pero además, podremos ver las reservas que se han realizado y gestionarlas.

También es interesante saber el número de reservas que no se llegan a efectuar cuando el restaurante este completo, con esta información podremos llegar a conocer cuál es el posible crecimiento del local en este caso concreto, que además, es un dato muy interesante que no suelen tener los programas de esta clase.

3.1 Trabajo de investigación:

En otras aplicaciones web similares no hay una reserva online por parte del cliente, es decir es el usuario (responsable del restaurante), el que tiene que añadir al cliente que reserva desde la propia aplicación.

Por último, están las aplicaciones que te permiten reservar en muchos restaurantes distintos, pero estas funcionan más como ranking de restaurantes, con puntuaciones y comentarios que como gestión de restaurantes como por ejemplo OpenTable o thefork.



Para explicar sus fortalezas y debilidades respecto a otras aplicaciones en el mercado he realizado un análisis DAFO. Un análisis DAFO es una técnica para analizar y poner de relieve la situación actual de tu negocio, y poder tomar las decisiones estratégicas adecuadas. Es una herramienta habitual tanto en un plan estratégico como en un plan de negocio o un estudio de mercado. Mediante un análisis del entorno externo y las características internas de tu negocio, esta herramienta de gestión empresarial permite obtener una representación gráfica de sus: debilidades, amenazas, oportunidades y fortalezas





3.2 Dificultad

La dificultad de este proyecto es sobre todo el uso de nuevas herramientas como usar Javafx para hacer las interfaces, o como realizar la parte de la página web desde visual studio code, para realizar la web hemos usado html, css y algún script de javascript y php



3.3 Creatividad e innovación

4. Desarrollo de proyecto

4.1 Acuerdo de proyecto

4.1.1 Generales

Buscamos cubrir la necesidad de las empresas de restauración, que estas puedan tener una organización moderna y que les permita a su vez acceder a datos menos habituales como la cantidad de clientes perdidos por lleno del local.

Esta aplicación nos permite realizar consultas sobre los asistentes diarios al restaurante, así como si estos han aceptado o no la reserva, y así evitar perder posibles mesas. Además, la aplicación tendrá un registro para que solo puedan acceder personas autorizadas.

Buscamos que el proyecto sea un programa que pueda seguir modificándose según demanden los clientes y tenerlo a como un servicio mensual con asistencia incluida en caso de problema.

4.1.2 Requisitos Funcionales

1. Iniciar sesión con usuario y contraseña para cada empleado.
2. Poder ver en la aplicación el número de mesas reservadas.
3. Se debe poder cerrar sesión haciendo que te vuelvas a loguear.
4. Posibilidad de ver número de reservas canceladas y confirmadas.
5. Recibir confirmación desde el correo por parte del cliente.
6. Tener una web donde se registre el cliente



4.1.3 Requisitos no funcionales

1. Tener Correo electrónico.
2. Tener la aplicación instalada.
3. Tiene que ser una interfaz intuitiva puesto que va a ser usada por gente no habituada a nuevas tecnologías.
4. Tener teléfono móvil

4.1.4 Especificaciones técnicas

Vamos a usar diferentes programas para nuestro proyecto,

-Xampp: Como servidor



-PhpMyadminn: Para gestionar las bases de datos.





-Java: Como lenguaje principal de la aplicación.



-Html y Css para crear la página web en local



-Window builder: Para la interfaz gráfica



4.2. Actividades y Cronología

Diagrama de Gantt

Jefe de proyectos: Jorge Tomás

Fecha de comienzo: 27/03/2023

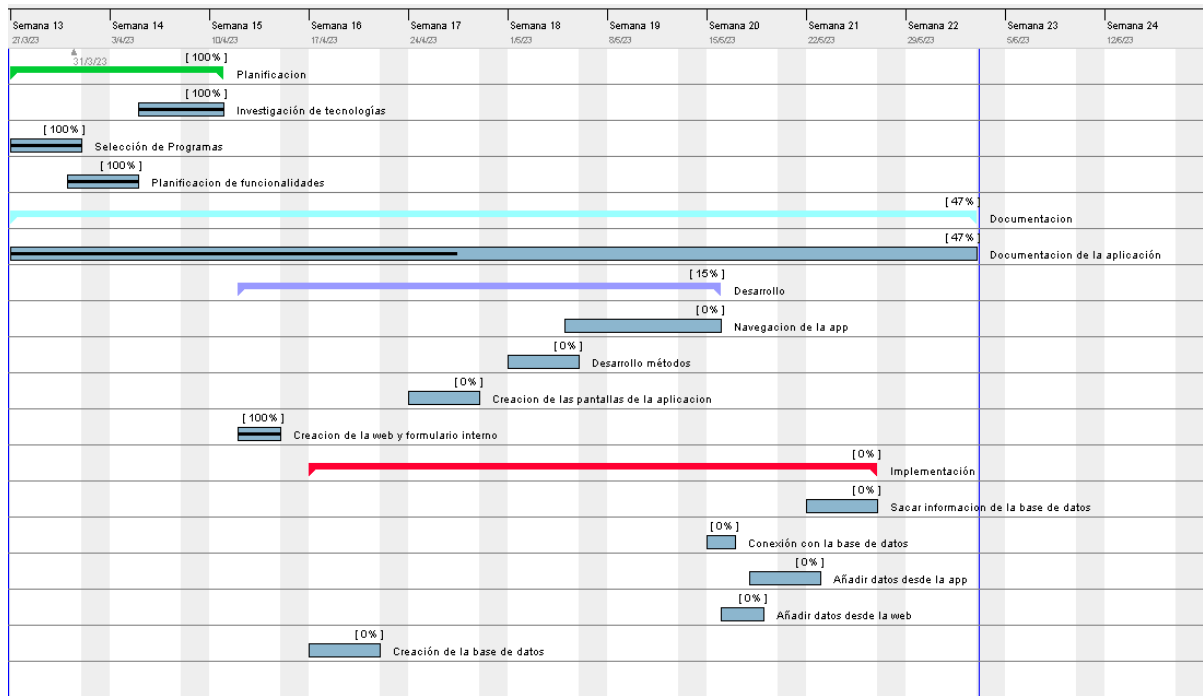
Fecha de finalización más pronto: 09/06/2023

El diagrama de Gantt es una herramienta de gestión de proyectos que permite visualizar de forma gráfica la planificación y el progreso de las tareas y actividades de un proyecto. Consiste en un gráfico de barras horizontales que muestra la duración y la fecha de inicio y finalización de cada tarea, así como su relación con otras tareas en el proyecto.



El diagrama de Gantt es útil porque permite a los gestores de proyectos y miembros del equipo visualizar el progreso y el calendario del proyecto de forma clara y concisa. También ayuda a identificar posibles retrasos o problemas en la planificación y a tomar medidas correctivas para garantizar que el proyecto se complete a tiempo y dentro del presupuesto. Además, puede utilizarse para comunicar el progreso del proyecto a los interesados y las partes interesadas.

Nombre	Fecha de inicio	Fecha de fin	Duración	Progreso
▼ Planificación	27/3/23	10/4/23	11	100
Investigación de tecnologías	5/4/23	10/4/23	4	100
Selección de Programas	27/3/23	31/3/23	5	100
Planificación de funcionalidades	31/3/23	4/4/23	3	100
▼ Documentación	27/3/23	2/6/23	50	47
Documentación de la aplicación	27/3/23	2/6/23	50	47
▼ Desarrollo	12/4/23	15/5/23	24	15
Navegación de la app	5/5/23	15/5/23	7	0
Desarrollo métodos	1/5/23	5/5/23	5	0
Creación de las pantallas de la aplicación	24/4/23	28/4/23	5	0
Creación de la web y formulario interno	12/4/23	14/4/23	3	100
▼ Implementación	17/4/23	26/5/23	30	0
Sacar información de la base de datos	22/5/23	26/5/23	5	0
Conexión con la base de datos	15/5/23	16/5/23	2	0
Añadir datos desde la app	18/5/23	22/5/23	3	0
Añadir datos desde la web	16/5/23	18/5/23	3	0
Creación de la base de datos	17/4/23	21/4/23	5	0



4.2.2 Documento de análisis y diseño

4.2.2.1 Casos de uso

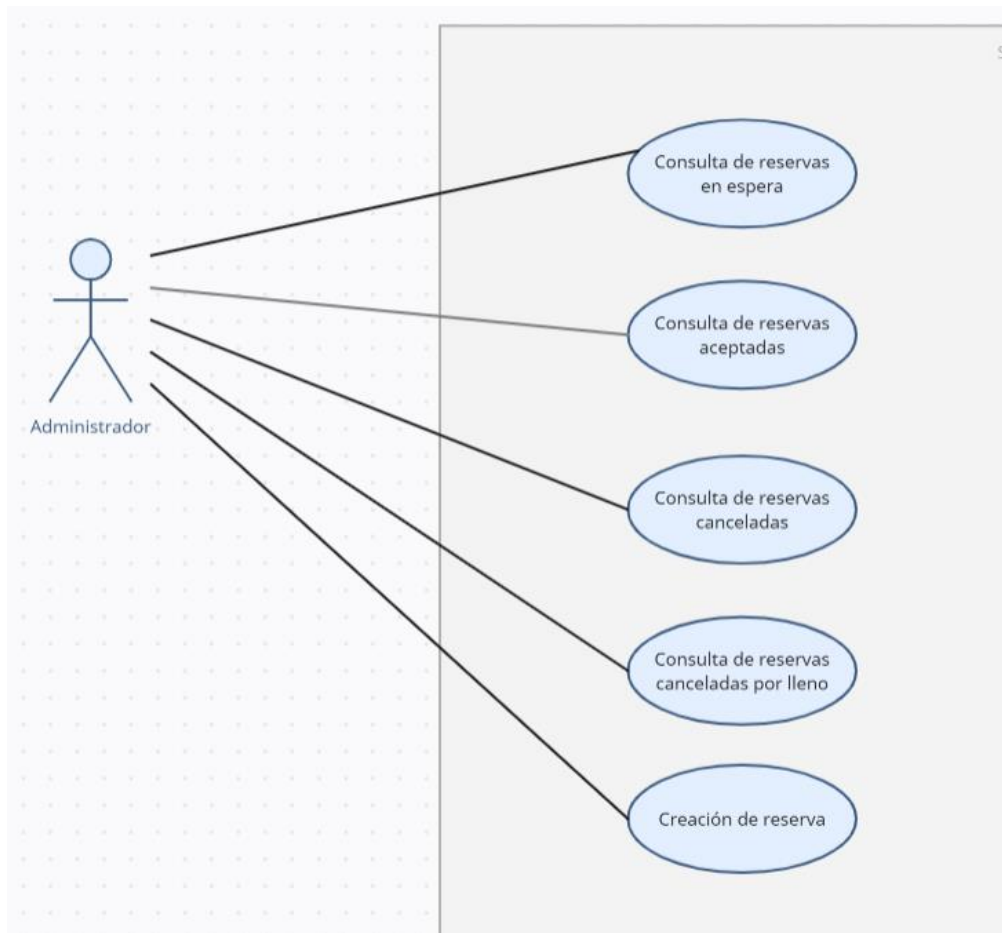


Usuario crea una reserva desde la web



Actor	Cliente
Resumen	Realiza una reserva desde la web
Precondiciones	Tener teléfono y correo e internet
Postcondiciones	---
Incluye	---
Extiende	---
Hereda de	---
Flujo de Eventos	
1.Desde un dispositivo entra en la página web y rellena los campos para realizar la reserva	

*Todo administrador debe logearse para entrar a la aplicación, pero esto no es una finalidad, es una forma de asegurarse que entra alguien del restaurante. Es por eso que no aparece en los casos de uso



Administrador consulta las reservas en espera

Actor	Administrador
Resumen	Consulta de reservas en espera
Precondiciones	Ser parte del staff del local y que haya reservas
Postcondiciones	---
Incluye	---
Extiende	---
Hereda de	---

Flujo de Eventos

1.Desde la aplicación después de registrarse debe hacer clic en el botón de la consulta de reserva en espera



Administrador consulta las reservas aceptadas	
Actor	Administrador
Resumen	Consulta de reservas aceptadas
Precondiciones	Ser parte del staff del local y que haya reservas
Postcondiciones	---
Incluye	---
Extiende	---
Hereda de	---
Flujo de Eventos	
1.Desde la aplicación después de registrarse debe hacer clic en el botón de la consulta de reserva aceptadas	

Administrador consulta las reservas canceladas	
Actor	Administrador
Resumen	Consulta de reservas canceladas
Precondiciones	Ser parte del staff del local y que haya reservas realizadas
Postcondiciones	---
Incluye	---
Extiende	---
Hereda de	---
Flujo de Eventos	
1.Desde la aplicación después de registrarse debe hacer clic en el botón de la consulta de reservas canceladas	

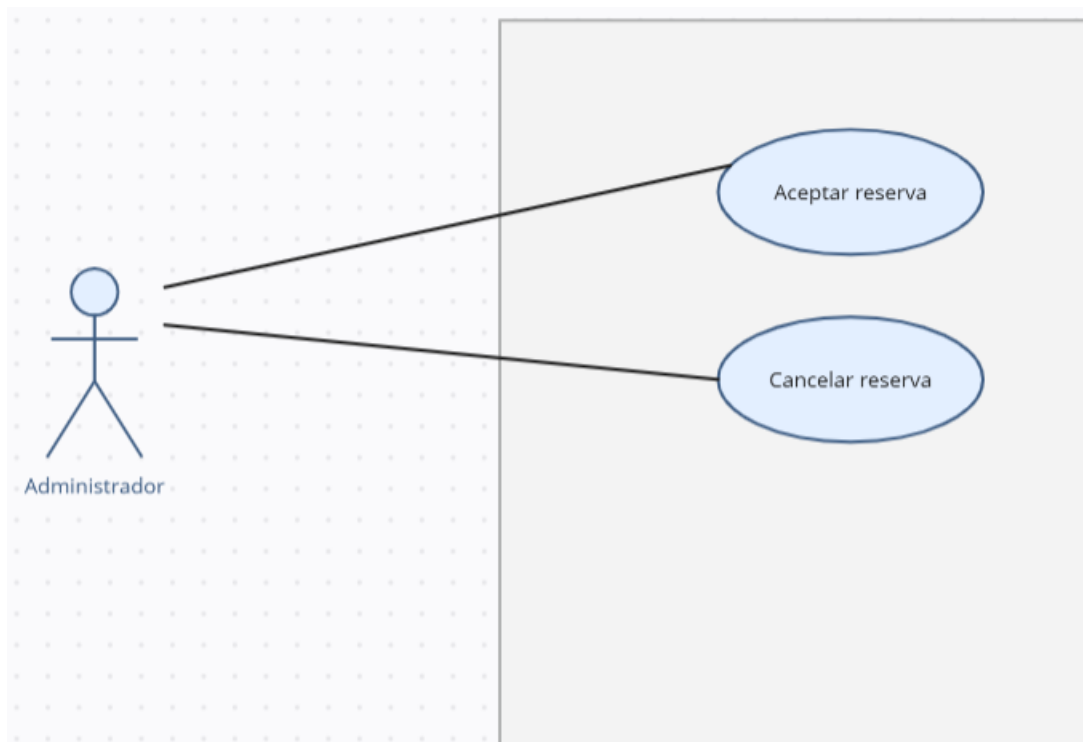


Administrador consulta las reservas que no se han producido por lleno en el restaurante	
Actor	Administrador
Resumen	Consulta de reservas canceladas por lleno
Precondiciones	Ser parte del staff del local, que el restaurante este lleno en ese turno.
Postcondiciones	---
Incluye	---
Extiende	---
Hereda de	---
Flujo de Eventos	
1.Desde la aplicación después de registrarse debe hacer clic en el botón de la consulta de reservas canceladas por lleno	

Administrador crea una reserva	
Actor	Administrador
Resumen	Creación de reserva desde la aplicación
Precondiciones	Ser parte del staff del local
Postcondiciones	Tiene que validarla el cliente en su momento
Incluye	---
Extiende	---
Hereda de	---
Flujo de Eventos	
1.Desde la aplicación después de registrarse debe hacer clic en el botón de la creación de reserva desde la aplicación	



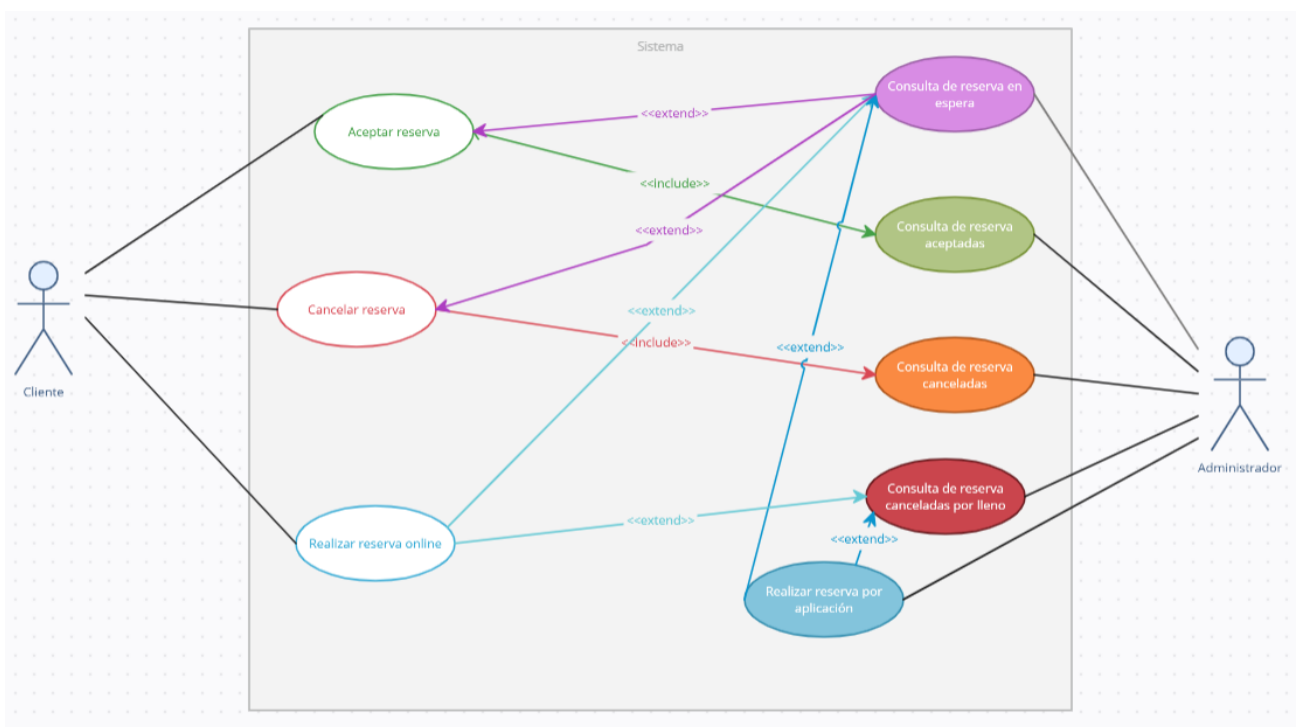
2. Deberá rellenar los campos correctamente y terminar con la reserva desde la aplicación.



Cliente acepta una reserva realizada por la web	
Actor	Cliente
Resumen	Acepta una reserva
Precondiciones	Haber hecho una reserva
Postcondiciones	---
Incluye	---
Extiende	---
Hereda de	---
Flujo de Eventos	
1. Acepta desde el correo electrónico	

**Administrador cancela una reserva realizada por la web**

Actor	Cliente
Resumen	Cancela la reserva
Precondiciones	Haber hecho una reserva
Postcondiciones	---
Incluye	---
Extiende	---
Hereda de	---

Flujo de Eventos**1.Cancela desde el correo electrónico**

Aquí está el diagrama de casos de uso que nos muestra cuales son las diferentes acciones que pueden llevar a cabo en la aplicación en el caso del cliente cabe destacar que no tiene acceso a la aplicación, pero desde el correo electrónico está capacitado para confirmar o cancelar la reserva incluyendo la cancelación en caso de no respuesta afirmativa dependiendo del tiempo que le cueste responder. **En este caso el CASE es creately.**



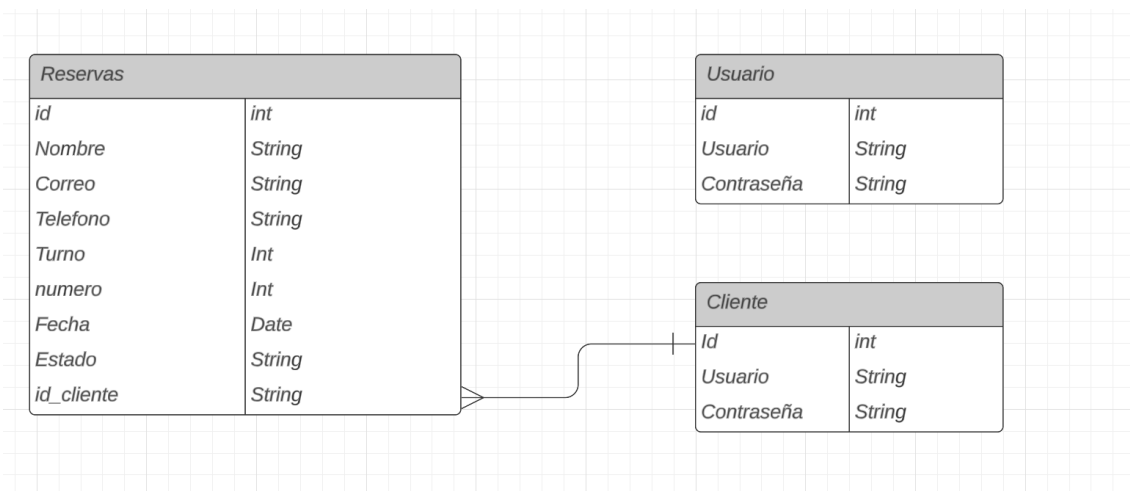
4.2.2.2 Modelo Entidad relacion

Un diagrama de bases de datos es una herramienta visual utilizada en el diseño y modelado de bases de datos relacionales. Consiste en una representación gráfica de la estructura de la base de datos, incluyendo las tablas, las relaciones entre las tablas y los campos de cada tabla.

En un diagrama de bases de datos, cada tabla se representa como un rectángulo, con el nombre de la tabla en la parte superior y los campos de la tabla en la parte inferior. Las relaciones entre las tablas se representan mediante líneas que conectan los campos que actúan como claves primarias y claves foráneas en las tablas relacionadas.

Los diagramas de bases de datos son útiles porque permiten a los diseñadores y desarrolladores de bases de datos visualizar la estructura de la base de datos y planificar su diseño antes de implementarla. También pueden ayudar a identificar posibles problemas en la estructura de la base de datos y a optimizar su rendimiento.

Además, los diagramas de bases de datos pueden utilizarse para comunicar la estructura de la base de datos a otros miembros del equipo de desarrollo y a las partes interesadas, lo que facilita la colaboración y la toma de decisiones informadas sobre el diseño y la implementación de la base de datos.



En este caso vemos que la base de datos tiene una relación un cliente con muchas reservas, es decir cada cliente puede tener varias reservas en su lista.

Por otro lado, tenemos a los usuarios, estos son principalmente para acceder a la aplicación del restaurante, pero no está relacionado con las otras tablas.



4.2.2.3 Diagrama de clases

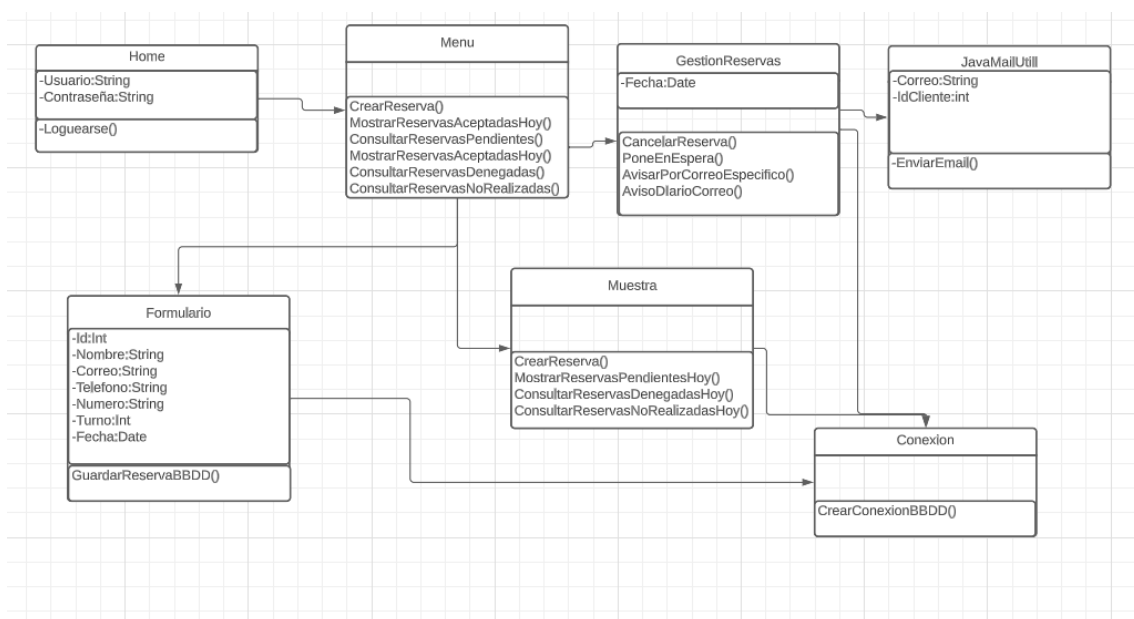
Un diagrama de clases es una herramienta visual utilizada en programación orientada a objetos para representar la estructura de clases y objetos en un sistema. En un diagrama de clases, se muestran las clases, las interfaces, los atributos y los métodos de un sistema, así como las relaciones entre ellas.

Cada clase en un diagrama de clases representa un objeto o una entidad en el sistema, y puede tener atributos que describen sus características y métodos que definen su comportamiento. Las interfaces en el diagrama de clases especifican los métodos que deben ser implementados por las clases que las utilizan.

Las relaciones entre las clases se representan mediante líneas y flechas que indican la naturaleza de la relación. Por ejemplo, una relación de "herencia" indica que una clase es una extensión de otra clase, mientras que una relación de "asociación" indica que una clase utiliza otra clase como parte de su implementación.

El diagrama de clases es una herramienta útil para modelar la estructura de un sistema, lo que ayuda a los desarrolladores a entender cómo interactúan las clases y a planificar la implementación del sistema. También puede utilizarse para comunicar la estructura del sistema a otros miembros del equipo de desarrollo y a las partes interesadas.

El CASE utilizado para en este caso es lucidchart.





4.2.2.3 Modelo Físico de Datos

El script de creación sería:

Tabla de clientes:

```
CREATE TABLE `clientes` (  
  `id` int(11) NOT NULL,  
  `usuario` varchar(11) NOT NULL,  
  `contraseña` varchar(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_general_ci;
```

Tabla de reservas:

```
CREATE TABLE `reservas` (  
  `id` int(11) NOT NULL,  
  `nombre` varchar(12) NOT NULL,  
  `correo` varchar(25) NOT NULL,  
  `telefono` varchar(20) NOT NULL,  
  `turno` int(11) NOT NULL,  
  `numero` int(11) NOT NULL,  
  `fecha` date NOT NULL,  
  `Estado` varchar(10) NOT NULL,  
  `id_cliente` varchar(20) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_general_ci;
```

Tabla de usuario:

```
CREATE TABLE `usuario` (  
  `id` int(11) NOT NULL,
```



```
`nombre` varchar(11) NOT NULL,  
`contraseña` varchar(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_general_ci;
```

-- Indices de la tabla `clientes`

```
ALTER TABLE `clientes`  
  ADD PRIMARY KEY (`id`),  
  ADD UNIQUE KEY `usuario` (`usuario`);
```

-- Indices de la tabla `reservas`

```
ALTER TABLE `reservas`  
  ADD PRIMARY KEY (`id`),  
  ADD KEY `rel` (`id_cliente`),  
  ADD UNIQUE KEY `telefono` (`telefono`),  
  ADD UNIQUE KEY `correo` (`correo`);
```

-- Indices de la tabla `usuario`

```
ALTER TABLE `usuario`  
  ADD PRIMARY KEY (`id`);
```

-- AUTO_INCREMENT de la tabla `clientes`

```
ALTER TABLE `clientes`  
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
  AUTO_INCREMENT=29;
```

-- AUTO_INCREMENT de la tabla `reservas`

```
ALTER TABLE `reservas`
```



```
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=74;
```

```
-- AUTO_INCREMENT de la tabla `usuario`
```

```
--
```

```
ALTER TABLE `usuario`
```

```
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
```

```
-- Filtros para la tabla `reservas`
```

```
ALTER TABLE `reservas`
```

```
ADD CONSTRAINT `rel` FOREIGN KEY (`id_cliente`) REFERENCES  
`clientes` (`usuario`) ON DELETE CASCADE ON UPDATE CASCADE;
```

```
COMMIT;
```

Modelo relacional de la base de datos:

Clave Primaria

Foreign key

Unicos

-Usuario(id,nombre,contraseña)

-Clientes(**id**,nombre,contraseña)

-Reservas(**id**,nombre,correo,telefono,turno,numero,fecha,estado,**id_cliente**)

4.2.2.4 Diagrama de secuencia

Un diagrama de secuencia es una herramienta de modelado visual utilizada en la programación orientada a objetos para representar la interacción entre objetos y componentes de un sistema a lo largo del tiempo. En un diagrama de secuencia, se muestran los objetos participantes y los mensajes que se envían entre ellos para realizar una tarea o una función.

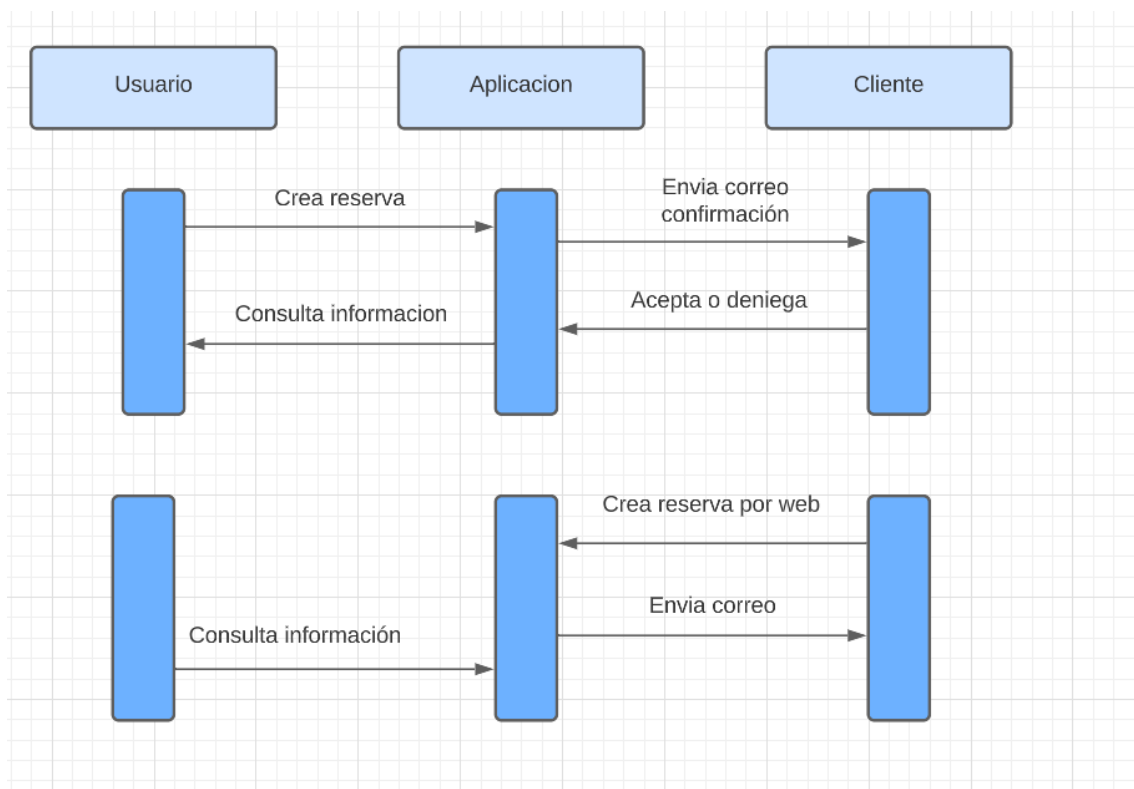
Cada objeto en un diagrama de secuencia se representa como una caja vertical, con el nombre del objeto en la parte superior. Los mensajes que se envían entre los objetos se representan mediante flechas horizontales que conectan las cajas de los objetos. La secuencia en la que se envían los mensajes se muestra mediante la ubicación vertical de las flechas.



Los diagramas de secuencia son útiles porque permiten a los desarrolladores visualizar y comprender el flujo de control y la interacción entre los objetos en un sistema. Esto ayuda a detectar problemas en el diseño o en la implementación del sistema antes de que se produzcan errores en la ejecución del sistema. Además, los diagramas de secuencia pueden utilizarse para documentar el diseño del sistema y comunicarlo a otros miembros del equipo de desarrollo y a las partes interesadas.

En resumen, un diagrama de secuencia es una herramienta visual útil para modelar y entender la interacción entre objetos y componentes en un sistema a lo largo del tiempo.

En este caso hemos usado lucidchart como herramienta CASE.





4.2.2.5 Desarrollo de interfaces

The screenshot shows a web application window titled 'El Palacio'. On the left is a black and white photograph of a formal dining room with tables, chairs, and a large window with curtains. On the right, the title 'El Palacio' is displayed in a large, bold font. Below the title are two input fields: 'USERNAME' and 'PASSWORD'. A red button labeled 'Login' is positioned below the password field.

Aquí Introducimos Usuario y Contraseña. Una vez escrito usuario y contraseña si no hemos acertado nos saltara una notificación en la pantalla diciéndonos que algo ha ido mal en caso contrario, es decir que tanto usuario contraseña hayan sido acertados nos llevara a otra página que vemos a continuación.

The screenshot shows the main menu of the 'El Palacio' application. The title 'El Palacio' is at the top. Below it is a large, empty rectangular box on the left. On the right is a vertical sidebar with a dark background and white text. The sidebar contains five menu items: 'Crear reserva', 'Reservas en curso', 'Reservas canceladas', 'Reservas no realizadas por lleno', and 'Gestionar reservas'. At the bottom of the sidebar is a red button labeled 'Volver'.



Una vez nos hayamos logueado en la aplicación, tendremos diferentes opciones a elegir.

La primera opción la vemos a la izquierda la cual es la pantalla más grande de todas en esta nos aparecerán las reservas del día de servicio, que ya han sido contrastadas por el usuario.

Después empezaremos por la ventana de la derecha, por el botón inferior que nos permitirá volver a la página anterior.

El siguiente botón nos permitirá gestionar todas las reservas.

El posterior, que es muy útil, que nos explicará que reservas no se han podido producir por lleno, es una información muy interesante puesto que nos puede ayudar a saber cuántas mesas más podríamos haber ofrecido, lo cual es útil para restaurantes en crecimiento.

El cuarto botón de la pantalla derecha empezando por abajo, el llamado Reservas canceladas, nos indica que reservas no han sido confirmadas por el cliente, esto puede ser necesario en caso de queja de cliente.

El botón que se encuentra justo encima de este nos indicará que reservas están en espera de ser confirmadas por el cliente en el día present.

Y por último en la primera pantalla situada a la derecha de la pantalla arriba del todo, este será un botón que nos dará la opción de crear reservas, para esto nos llevará a otra página a modo de formulario.

El Palacio

Nombre	<input type="text"/>	Numero de clientes	<input type="text"/>
Correo electronico	<input type="text"/>	Fecha	<input type="text"/>
Telefono	<input type="text"/>	Numero de turno	<input type="text"/>



En esta ventana tendremos seis opciones, todas deberán ser rellenadas correctamente para poder producirse una reserva.

Empezando por la primera a la izquierda, tenemos el nombre del cliente en esa reserva.

Después a su derecha, el número de clientes que van a producirse en la reserva.

La que se encuentra a su debajo a la izquierda será el correo del usuario que se usará a modo de contraseña del cliente.

A la derecha de este se encuentra la fecha que tendrá formato dd-MM-yyyy.

Debajo el teléfono que se usará a modo de usuario.

A su derecha se encuentra el turno que deben elegir entre 1 y 2.

Y por último un botón para enviar toda la información y crear la reserva.

Por parte del cliente también va a poder reservar desde la web pinchando en reservas nos llevara a un formulario y funcionara muy parecido al que hay en la aplicación.

The screenshot shows a web application window titled "Gestion de reservas". At the top, there are three input fields: "Fecha:" with a date picker, "Turno:" with a dropdown menu set to "1", and "Estado:" with a dropdown menu set to "Espera". Below these is a large text area containing a list of reservations. Each line in the list follows the format: "Turno: [Turno] Nº Personas: [Nº Personas] Fecha: [Fecha] Estado: [Estado] IdReserva: [IdReserva] IdCliente: [IdCliente]". The list is scrollable. At the bottom of the window, there is a row of five buttons: "Cancelar reserva", "Espera", "Volver" (highlighted in red), "Avisar", and "Confirmacion".

En esta pantalla podemos gestionar las reservas del restaurante.

En la fecha podemos elegir una fecha concreta a modo de buscador, después lo mismo con el turno que nos acortara más la búsqueda y por ultimo el estado. El



estado especifica al turno, esto quiere decir que si cambias el turno de 1 a 2 te restablecerá la búsqueda a todos (All).

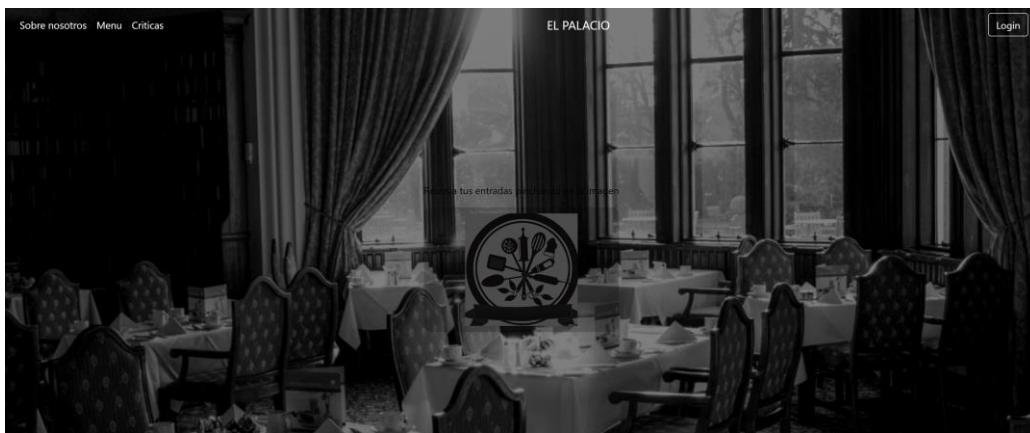
Con el primer botón empezando por la izquierda podemos cancelar una o varias reservas que **hayan sido aceptadas anteriormente**.

El segundo botón cambia una o varias reservas ha estado en espera.

El tercer botón sirve para volver a la ventana anterior.

El Cuarto sirve para mandar un correo de confirmación a o los clientes de una o varias reservas.

Y el ultimo manda confirmación a todos los clientes con reserva en el día presente.



Una vez hagamos clic en el icono del centro aparecerá el formulario y si le damos arriba a la derecha a login nos mandará a un registro web que pondremos el teléfono de usuario y el correo de contraseña.





Aquí veremos una web desde la cual podremos aceptar o anular una reserva poniendo un id de nuestra lista de reservas.

Reservas

ID	Nombre	Correo	Telefono	Fecha	Estado	Turno
1 -	manolo	manolo@hotmail	698563587	2023-05-10	Espera	2
18 -	paco	paco@asd	6663332252	0000-00-00	Espera	2
19 -	jorge	jorge@jorge	999888777	0000-00-00	Espera	2
23 -	Query4	Query4@Query4	999888777	0000-00-00	Espera	2
24 -	Query4	Query4@Query4	999888777	0000-00-00	Espera	2
27 -	adsf3erb	adsf@asa	999888777	0000-00-00	Espera	2
32 -	mardsodf	asfd@agfds	999888777	0000-00-00	Espera	2

Email address

La siguiente es la ventana del formulario para crear una reserva

El_Palacio

Nombre	Número de clientes
<input type="text" value="Nombre"/>	<input type="text"/>
Correo Electronico	Fecha
<input type="text" value="nombre@example.com"/>	<input type="text" value="17/04/23"/>
Teléfono	Numero de turno
<input type="text" value="632157874"/>	<input type="text" value="1 o 2"/>

4.2.3 Pruebas de sistema

4.2.4 Documentacion para el cliente

4.3 Metodología

Para el desarrollo de esta aplicación hemos utilizado el sistema Kanban que es un sistema de gestión visual de flujo de trabajo que se originó en la industria manufacturera japonesa, y que hoy en día se utiliza en una amplia variedad de sectores y entornos. La palabra "Kanban" significa "tarjeta" o "etiqueta" en japonés, y es precisamente el elemento visual clave en este sistema.



Este sistema se basa en la idea de que el trabajo fluye de manera más eficiente y efectiva cuando se limita la cantidad de trabajo en proceso y se centra en la finalización de las tareas individuales antes de comenzar nuevas. Para lograr esto, utilizamos una pizarra en el cual vemos todos los trabajos y los dividimos en diferentes etapas de proceso, estas están divididas en ".

Cada tarea se representa mediante una tarjeta, esta contiene información descriptiva de la tarea a realizar, la persona que la realiza, su fecha de inicio y su fecha de finalización prevista. Mientras las tareas se van realizando pasan de una etapa de proceso a otra y movemos respectivamente las tarjetas en nuestro tablero.

El sistema Kanban también utiliza principios de limitación de trabajo en proceso (WIP) y de mejora continua. Establecer unos límites en la cantidad de trabajo dentro de cada etapa de nuestro proceso nos ayuda a evitar sobrecargarnos de trabajo y a su vez se logra una mejora continua debido a la revisión periódica del proceso.

En resumen, el sistema Kanban es una herramienta efectiva para la gestión visual de tareas y proyectos, que permite limitar el trabajo en proceso, mejorar la eficiencia y la calidad, y fomentar la colaboración y la comunicación en el equipo.

Este sistema es tan efectivo que se ha hecho popular en muchas empresas importantes entre las cuales podemos destacar a:



El sistema Kanban se ha popularizado en todo el mundo y ha sido adoptado por muchas empresas líderes en diversos sectores. Algunas de las empresas más conocidas que utilizan Kanban son Toyota, Amazon, Microsoft, Spotify, Airbus o Zara.

He elegido este sistema porque me parece el más visual a la hora de organizarse y permite ver en qué fase del proyecto estas según cuantos “post it” te queden en cada columna. Además, es compatible con el diagrama de Gant, puesto que podemos usar diferentes colores para cada una de los subapartados. Además, personalmente es el sistema de trabajo que más me convence para realizar un proyecto ya sea en grupo o de manera individual, como es este caso. Además permite organizarse y ver en que estado tenemos los proyectos.

4.4 Recursos

Infraestructura y mantenimiento

- Ordenador:
 - Ryzen 5 5500 3.6GHz
 - 16GB RAM DDR
 - ASUS Prime B550M-A AMD AM4 (3ª generación Ryzen)
 - Windows 10
- Servidor de base de datos: PhpmyAdmin.

Humanos

-En este caso el proyecto ha sido realizado de manera individual.

Aplicaciones utilizadas

-**Eclipse**: Es necesario para crear todo el código de java de la aplicación y añadir a esta, una funcionalidad.

-**Visual studio**: Para realizar la parte de la página web.

-**Office**: Lo necesitamos para la parte de la documentación.

4.5 Dificultad del proyecto

Este proyecto ha resultado complicado por su tamaño, al ser el primer proyecto de cierto tamaño que se realiza y usar WindowBuilder, cabe destacar el envío de emails que ha cambiado ligeramente la forma de enviarlos recientemente o los filtros de la parte de gestión de reservas. Además de que he querido implementar una web con bastante código de php desde cero hasta enviar, recoger y editar datos de la base de datos y usar distintos programas a los usados en clase.



También el tamaño y complejidad de la documentación ha hecho complicado compaginarlo con el tiempo usado para las practicas.

4.6 Trabajo de investigación

En este proyecto ha habido muchos puntos que han implicado usar elementos externos al curso, el primero de estos es el hecho de implementar una web con html, css, y scripts hechos con Javascript y php.

Además, he usado librerías diferentes de los usados en clase como es la librería de WindowBuilder o la de JavaMail

Y por último, también he usado programas nuevos como el visual studio code o bitbucket con sourcetree.

5. Presupuesto

Gasto

Para llevar a cabo el presupuesto dividiremos cada tarea por horas y coste.

Cada tarea conlleva que un profesional distinto haga un trabajo, así que voy a desglosar el presupuesto por profesional.

7 horas Analista Junior = $14.3 * 7 = 100.1$

24 horas Diseñador Interfaces = $16.15 * 24 = 387.6€$

65 horas Programador Junior = $14 * 65 = 910€$

9 horas Administrador BBDD = $12,82 \text{ por hora.} * 9 = 115.38€$

2 horas Diseñador BBDD = $15,56 * 2 = 31.12€$

22 horas Diseñador Web = $12.31 * 22 = 270.82 €$

Recursos personales utilizados (Equipo, electricidad) = 520€

Subtotal=2.335,02€

I.V.A. 21%=490,3542€

Gasto total=2.825,3472 €

Beneficio

Como es un producto al cual le queremos sacar beneficio.



Gastos: 2.825,3472€

Porcentaje de beneficio: 15% de 2.825,3472€ = 423,80613€

Beneficios = 2.825,3472€ + 423,80613€ = 3.249,15333€

Coste total=3.249,15333€
Beneficio total=423,80613€ €

z

5.1 Resumen ejecutivo

5.2 Pruebas del Sistema

Pruebas unitarias: Consiste en la prueba de fragmentos de código de forma que comprobemos que funciona correctamente. Se ha comprobado que todos los elementos funcionan de manera adecuada, que las referencias internas y externas están correctas y son relativas.

Pruebas de integración: Prueba que diferentes o módulos o servicios funcionan correctamente. Por ejemplo, he probado a la conexión con la base de datos y ha funcionado correctamente, así como todas sus operaciones de inserción y consulta.

Pruebas funcionales: Son pruebas similares a las de integración pero que buscan tener valores concretos, en este caso el cambio de estado de los productos al confirmar la reserva funciona correctamente.

Pruebas de humo: Consiste en probar el funcionamiento básico de la aplicación en un entorno controlado, antes de dar acceso a terceros.

6.Resultados obtenidos y conclusiones

7.Bibliografía

- [https://www.agilealliance.org/glossary/kanban/#q=~\(infinite~false~filters~\(postType~\(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video\)\)~tags~\(~'kanban\)\)~searchTerm~'~sort~false~sortDirection~'asc~page~1\)](https://www.agilealliance.org/glossary/kanban/#q=~(infinite~false~filters~(postType~(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video))~tags~(~'kanban))~searchTerm~'~sort~false~sortDirection~'asc~page~1))
- <https://www.infoautonomos.com/plan-de-negocio/analisis-dafo/>
- <https://www.edrawsoft.com/es/how-to-create-database-diagram.html#:~:text=Un%20diagrama%20de%20base%20de%20datos%20es%20una%20herramienta%20visual,a%20la%20que%20est%C3%A9%20conectado.>
- <https://getbootstrap.esdocu.com/>
- <https://www.php.net/manual/es/intro-what-is.php>



- <https://www.wordreference.com>
-