Pontificia Universidad Católica del Ecuador

Facultad De Ingeniería

Escuela de Sistemas



TEMA:

DESARROLLO DE UNA APLICACIÓN WEB, PARA LA GESTIÓN DE RESERVAS DE RESTAURANTES, DEL CENTRO NORTE DE QUITO.

AUTOR:

ERICK RICARDO HARO MORALES.

TRABAJO PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS DE INFORMACIÓN.

QUITO, MAYO – 2023

DEDICATORIA

A todas las personas que me han brindado su apoyo a lo largo de la carrera, en especial a mi familia.

AGRADECIMIENTO

A todas las personas que me han ayudado durante esta etapa de aprendizaje y crecimiento personal, en especial a mis padres que siempre me han apoyado para cumplir y seguir adelante con mis sueños y metas

RESUMEN

En el presente trabajo de titulación se presentará el proceso de desarrollo de una aplicación web para la gestión de reservas en el centro-norte del distrito metropolitano de Quito. La aplicación propuesta permite a los restaurantes registrarse en la misma y gestionar las reservas que realicen los comensales. Por otro lado, permite a los comensales ver los restaurantes registrados en la aplicación y realizar una reserva en los mismos.

ÍNDICE

ÍNDICE DE FIGURAS, GRÁFICOS Y TABLAS V

ÍNDICE DE ILUSTRACIONES V

ÍNDICE DE TABLAS VII

CAPÍTULO I: INTRODUCCIÓN 1

1. MARCO DE REFERENCIA 1

1.1. JUSTIFICACIÓN 1

1.2. Planteamiento del problema 1

1.3. Objetivo General 2

1.4. Objetivos Específicos 2

1.5. Antecedentes 2

1.6. Alcance 3

CAPÍTULO II: FUNDAMENTACIÓN TEÓRICA 4

2. Marco Teórico 4

2.1. Aplicaciones 4

2.1.1. Aplicaciones de escritorio 4

2.1.2. Aplicaciones móviles 5

2.1.3. Aplicaciones web 5

2.1.3.1. Aplicaciones web en tiempo real 6

2.2. Reservas 6
2.2.1. Reservas en restaurantes 6
2.3. Herramientas 7
2.3.1. Angular 7
2.3.2. React 7
2.3.3. .NET 8
2.3.4. NodeJS 8
2.3.5. SQL Server 8
2.3.6. PostgreSQL 9
2.3.7. Mapbox GL JS 9
2.3.8. Leaflet 9
2.3.9. Signal R 9
CAPÍTULO III: METODOLOGÍA 11
3. Metodología de desarrollo del plan de tesis 11
3.1. Enfoque Cualitativo 11
3.2. Investigación Aplicada 11
3.2.1. Método de investigación: Observación. 12
3.2.1.1. Observación Participativa 12
3.2.1.2. Observación No Participativa 13
3.3. Metodologías Ágiles para desarrollo de software. 13
3.3.1. Descripción de las metodologías 14

3.3.1.1. Scrum 14
3.3.1.2. Kanban 16
3.3.1.3. Por prototipos 16
3.3.2. Modelo que aplica 17
3.4. Elección de tipo de aplicación 18
3.5. Elección de Herramientas 19
3.5.1. Herramienta para el desarrollo del Front End 19
3.5.2. Herramienta para el desarrollo backend 19
3.5.3. Herramienta para obtener la ubicación y manejo de mapas 20
3.5.4. Motor de base de datos 21
CAPÍTULO IV: DESARROLLO DE LA INVESTIGACIÓN 22
4.1. Análisis de los requerimientos. 22
4.1.1. Propósito. 22
4.1.2. Funcionalidades del Producto. 22
4.2. Modelo de la base de datos. 24
4.3. Casos de uso 25
4.4. Diseño de las pantallas 31
CAPÍTULO V: IMPLEMENTACIÓN 35
5. Implementación de la aplicación 35
5.1. Pruebas funcionales. 35
5.1.1. Caso de uso: Registro del restaurante 35

5.1.2. Caso de uso: Login 38
5.1.3. Caso de uso: Ver las reservas....40
5.1.4. Caso de uso: Crear reserva (Restaurante) 44
5.1.5. Caso de uso: Lista de restaurantes 46
5.1.6. Caso de uso: Crear Reserva (Comensal) 47
CONCLUSIONES Y RECOMENDACIONES 51
6. Conclusiones: 51
7. Recomendaciones: 51
BIBLIOGRAFÍA 53
ANEXOS 56
Anexo A: 56

ÍNDICE DE FIGURAS, GRÁFICOS Y TABLAS

ÍNDICE DE ILUSTRACIONES

Figura 1 Modelo de la base de datos....25
Figura 2 Diseño de la pantalla home de restaurantes....31
Figura 3 Diseño de la pantalla de reservas....31
Figura 4 Diseño de la pantalla nueva reserva (restaurantes)....32
Figura 5 Diseño de la pantalla editar perfil....32
Figura 6 Diseño de la pantalla configuración....33
Figura 7 Diseño pantalla home....33
Figura 8 Diseño pantalla restaurante....34
Figura 9 Diseño pantalla Home....34
Figura 10 Error contraseña insegura....36
Figura 11 Error contraseñas no coinciden....36
Figura 12 Usuario creado con éxito....37
Figura 13 Seleccionar ubicación del restaurante....37
Figura 14 Registro de datos públicos exitoso....38
Figura 15 Configuración del restaurante exitosa....38
Figura 16 Login de restaurantes....39
Figura 17 Pantalla home de restaurantes....40
Figura 18 Error contraseña o usuario incorrectos....40
Figura 19 Pantalla home gestor de reservas....42
Figura 20 Pantalla para ver las reservas según la fecha....42
Figura 21 Editar reserva....43
Figura 22 Eliminar reserva....43

Figura 23 Reserva creada exitosamente 44
Figura 24 No hay disponibilidad para el horario elegido 45
Figura 25 La hora seleccionada esta fuera del horario del restaurante.....45
Figura 26 Pantalla home 46
Figura 27 Búsqueda de restaurante 47
Figura 28 Reserva creada con exito por un comensal.....48
Figura 29 Error de disponibilidad de reservas.....49
Figura 30 Error hora fuera de horario de atención 49
Figura 31 Pruebas de rendimiento.....50

ÍNDICE DE TABLAS

Tabla 1 Comparación de metodologías ágiles 17
Tabla 2 Comparación de tipos de aplicaciones 18
Tabla 3 Comparación de herramientas del front end 19
Tabla 4 Comparación de herramientas del back end 19
Tabla 5 Comparación de herramientas para mapas 20
Tabla 6 Comparación de Motores de base de datos 21
Tabla 7 Caso de uso CU1 25
Tabla 8 Flujo del caso de uso CU1 26
Tabla 9 Caso de uso CU2 26
Tabla 10 Flujo del caso de uso CU2 27
Tabla 11 Caso de uso CU3 27
Tabla 12 Flujo del caso de uso CU3 28
Tabla 13 Caso de uso CU4 28
Tabla 14 Flujo del caso de uso CU4 29
Tabla 15 Caso de uso CU5 29
Tabla 16 Flujo del caso de uso CU5 29
Tabla 17 Caso de uso CU6 30
Tabla 18 Flujo del caso de uso CU6 30

CAPÍTULO I: INTRODUCCIÓN

1. MARCO DE REFERENCIA

1.1. JUSTIFICACIÓN

En la actualidad, en el centro norte de la ciudad de Quito la apertura de nuevos restaurantes está en constante crecimiento, muchos de estos restaurantes debido al lugar en el que se encuentran y el tipo de atención que brindan hacia sus clientes, manejan un proceso de reservas. Al existir varios restaurantes que manejan reservas, se les dificulta a los clientes encontrar la información de estos de manera consolidada. A su vez los restaurantes pueden mejorar la eficiencia de este proceso.

Por lo expuesto anteriormente, se propone el desarrollo de una aplicación web para la gestión de reservas de restaurantes del centro norte de la ciudad de Quito. La aplicación ayudará a que los restaurantes tengan su información de contacto más visible y consolidada, permitiendo a los clientes encontrar con más facilidad los diferentes restaurantes que se registren en la aplicación, y brindarles una mayor disponibilidad para que estos puedan realizar sus reservas. Evitando que los clientes se pongan en contacto directo con los restaurantes.

1.2. Planteamiento del problema

El aumento de la oferta de restaurantes que manejan reservas en el centro norte de la ciudad de Quito hace que la información de estos se encuentre dispersa en diferentes medios. Por lo cual, a los clientes se les puede dificultar encontrar la información de estos, para poder realizar una reserva de manera rápida y eficiente.

Además, para los restaurantes manejar el proceso de reservas de manera manual puede traer varios problemas como pérdida de la información, errores del personal humano, etc. es por esto por lo que la evolución de la tecnología ha permitido hacer grandes avances en cuanto a

automatizar procesos mediante aplicaciones web. La automatización permite que los procesos sean más ágiles y eficientes (redhat, 2022). Por lo que, las aplicaciones web han beneficiado a varios negocios brindando competitividad a los mismos.

Por esta razón, la aplicación ayudara a que los restaurantes tengan su información más visible y fácil de encontrar, además de manejar las reservas de manera automatizada, permitiendo a los clientes realizar una reserva de manera rápida y eficiente.

1.3. Objetivo General

Desarrollar una aplicación web, para la gestión de reservas de restaurantes para los sectores centro – norte del Distrito Metropolitano de Quito.

1.4. Objetivos Específicos

- Analizar la situación actual de las reservas en los restaurantes del centro - norte del Distrito Metropolitano de Quito.
- Definir los requerimientos para el diseño de la aplicación web para la reserva de restaurantes.
- Desarrollar la aplicación web, para reservas en los restaurantes del centro - norte del Distrito Metropolitano de Quito.

1.5. Antecedentes

En la actualidad, el uso de la tecnología y los sistemas han permitido que los restaurantes hagan sus procesos más eficientes y ágiles. Por lo que han sido objeto de varias investigaciones y desarrollos de aplicaciones que automatizan estos procesos.

Desarrollos previos han implementado un apartado de reservas en restaurantes a través de aplicaciones web, que permiten programar pedidos y consultar opiniones sobre restaurantes. (Espasa Saval, 2021). Por otro lado, desarrollos de aplicaciones móviles se han enfocado en el no-show en los restaurantes, creando políticas para evitar esta problemática. (Romo Rivero, 2020).

Además, se han desarrollado plataformas web que también permiten la gestión de reservas en restaurantes, pero que solo son de un restaurante y no permiten el registro de otros restaurantes entre los cuales el usuario pueda elegir. (Mazo Jiménez, 2022). Sin embargo, también se han implementado aplicaciones web en las cuales los clientes se deben registrar para poder realizar reservas y cuentan con un registro para los restaurantes. (Yanez Rea, 2022). Y también existen desarrollos en los cuales se han implementado aplicación web como móvil, sin embargo, estas eran para un restaurante en específico. (Villao, 2021)

Estos antecedentes permiten tener una base sólida sobre las aplicaciones y plataformas que se han desarrollado para restaurantes en los últimos años.

1.6. Alcance

El presente proyecto de titulación finalizará, con el desarrollo de una aplicación web para reservas de restaurantes del centro norte de la ciudad de Quito, que contendrá:

- Login para permitir el ingreso a la aplicación a nivel usuario final (Restaurantes y Administrador).
- Modulo para la gestión de servicios para restaurantes. (Número de mesas, número de comensales, intervalo de tiempo entre reservas, horarios del restaurante, control de las reservas).
- Facilidades para que el cliente seleccione entre los restaurantes registrados, elija un horario para hacer la reserva e ingrese el número de comensales.

CAPÍTULO II: FUNDAMENTACIÓN TEÓRICA

2. Marco Teórico

2.1. Aplicaciones

Una aplicación es un tipo de software, desarrollado con el objetivo de cumplir una o varias tareas. Existen dos términos para nombrarlas, el primero es app, pero este se usa cuando estamos hablando de aplicaciones móviles. El segundo es aplicación que se usa comúnmente cuando estamos hablando de software para computadoras de escritorio y laptops. Aunque estos se pueden usar sin distinción. (Twilio, s.f.)

Las aplicaciones por lo general se usan para automatizar tareas. No obstante, cada día es más común ver que estas tienen usos más variados, además de que a lo largo del tiempo surgen necesidades más diversas a las cuales adaptarse. (Cristal, 2020)

Cuando se va a implementar una solución, existen diferentes tipos de aplicaciones entre los cuales elegir, esto depende del usuario final al que se dirige y el uso que va a tener la misma. Por lo general se usan 3 tipos de aplicaciones, aplicaciones de escritorio, aplicaciones móviles y aplicaciones web. Pueden existir casos en el que la solución que se va a implementar se dirige a diferentes tipos de usuarios, por lo cual se puede implementar más de un tipo de aplicación.

2.1.1. Aplicaciones de escritorio

Una aplicación de escritorio es un tipo de aplicación la cual se debe instalar en un computador, estas se pueden descargar de internet, de una tienda de aplicaciones o se pueden instalar a partir de un CD o DVD. (Microsoft, s.f.). Una ventaja de estas aplicaciones es que permiten realizar tareas offline, además de tareas que necesiten una conexión a internet.

2.1.2. Aplicaciones móviles

Una aplicación móvil es un tipo de aplicación que se descarga e instala en un dispositivo móvil. Estas aplicaciones sirven para brindar portabilidad a los usuarios, y permiten realizar tareas tanto offline como online.

Un concepto que cada vez va tomando más importancias es el de las PWA, que por sus siglas significan Aplicaciones web progresivas. Estas son aplicaciones las cuales están disponibles en navegadores y buscan brindar una experiencia muy similar al de las aplicaciones nativas. (Thomas , y otros, 2020)

Para que otros usuarios puedan descargar estas aplicaciones de manera segura, se usan diferentes tipos de tiendas de aplicaciones dependiendo del sistema operativo del dispositivo. Estas tiendas, con el objetivo de brindar seguridad a los usuarios, exigen a los desarrolladores seguir ciertos procedimientos, normas y aspectos de seguridad, para poder publicar una aplicación en las mismas.

2.1.3. Aplicaciones web

Las aplicaciones web son aplicaciones las cuales como su nombre indica se encuentran en la web, y permiten a los usuarios hacer uso de funcionalidades complejas sin tener que instalar una aplicación, si no solo haciendo uso de un navegador con conexión a internet. (Amazon Web Services, s.f.). Por lo que se puede acceder a estas aplicaciones desde diferentes dispositivos como celulares, computadoras, tablets, etc.

Estas aplicaciones se dividen en dos partes, las cuales son frontend y backend:

Frontend: Es el que se encarga de darle estilos a la página, para que la información que se presente al usuario sea de fácil comprensión. El frontend también es conocido como las interfaces graficas con las que interactúa el usuario, por lo que las tecnologías que se usan aquí

son HTML, CSS y JavaScript. Aunque se pueden usar ciertos frameworks que nos faciliten el trabajo. (Pérez, Quispe, Mullicundo, & Lamas, 2021)

Backend: El backend es en donde se desarrolla la lógica del negocio, manejo de datos y se tiene la conexión de la base de datos. Se puede usar diferentes lenguajes de programación y herramientas para el desarrollo del backend.

Por lo general, el envió de información mediante el frontend y backend depende de que el frontend haga una petición al backend, para que este le envié información. Pero existe otra manera de hacer un envío bidireccional de datos el cual se presenta a continuación.

2.1.3.1. Aplicaciones web en tiempo real

Las aplicaciones web en tiempo real permiten que el frontend pueda recibir datos del backend, sin necesidad de que el usuario actualice la página o presione algún botón, para que se haga una petición al servidor.

Un ejemplo de cómo funcionan estas aplicaciones son los chats, al tener abierto el chat recibimos los mensajes que envíen otros usuarios y estos aparecen en pantalla, sin la necesidad de actualizar la página.

2.2. Reservas

El principal objetivo de las reservas es el apartar un espacio por un tiempo, para poder hacer uso de este. La principal ventaja de esto es que se convierte en algo confiable para el cliente, ya que tiene la seguridad de que contará con el espacio que aparto en el tiempo definido. Existen varios tipos de reservas, pero en este caso nos centraremos en las reservas en los restaurantes.

2.2.1. Reservas en restaurantes

En el siglo 18 los restaurantes, normalmente tenían una mesa grande en la que comían todos los comensales, sin embargo, para poder tener más privacidad se podía reservar una sala privada, después en el siglo 19 aparecieron las reservas de mesas con el objetivo de que los

hombres ricos pudieran pretender a una mujer de una manera aceptada en la sociedad (Shah, 2014).

En la actualidad, se sigue manejando reservas en los restaurantes, en las que el cliente se pone en contacto con el restaurante para reservar una mesa, este debe brindar datos como el nombre de la persona, número de comensales, hora, número de contacto, etc. para que los restaurantes puedan apartar el espacio necesario.

2.3. Herramientas

2.3.1. Angular

Angular es un framework de desarrollo de aplicaciones web para la parte del cliente, este se basa en componentes. Además de contar con una gran colección de librerías y herramientas que facilitan el desarrollo de las aplicaciones. (Angular, 2022)

- Componentes: Son bloques de código que componen una aplicación. Estos están compuestos de una clase de una clase de TypeScript, una plantilla de HTML y una plantilla de estilos (CSS).

Este Framework permite una creación más sencilla de aplicaciones para el cliente, y además la estructura de las carpetas y archivos que nos facilita este framework, permite que el código sea mantenible y ayuda a la colaboración, ya que si otras personas acceden al código podrán encontrar las diferentes funciones fácilmente.

Entre las principales ventajas de angular se puede mencionar, la inyección de dependencia a los componentes, lo que va a facilitar hacer aplicaciones modulares, cuenta con una estructura y arquitectura para facilitar la escalabilidad de los proyectos. (Hutagikar & Hegde, 2020)

2.3.2. React

Es una biblioteca de Javascript la cual se enfoca en la creación de interfaces de usuario, esta librería permite este desarrollo a través de componentes reutilizables. Otra manera de ver los

componentes es como una función, estos pueden ir desde botones a interfaces más complejas. (React, s.f)

Esta librería facilita el desarrollo de interfaces de usuario, ya que se cuenta con una serie de herramientas las cuales se pueden incorporar a esta librería.

2.3.3. .NET

Es una plataforma que sirve para crear diferentes tipos de aplicaciones, está conformado de lenguajes de programación y bibliotecas. Para poder ampliar la plataforma de .net existe ASP.NET que proporciona API extras para compilar aplicaciones web. (Microsoft, s.f.)

También existe .NET Core, el cual es un framework multiplataforma, lo que significa que se puede utilizar en distintos sistemas operativos. (Macho, 2020) .Net nos permitirá construir el backend ya que esta plataforma tiene herramientas especializadas para este trabajo. Además de contar con varias librerías y paquetes que nos serán de mucha ayuda.

2.3.4. NodeJS

Es un entorno de tiempo de ejecución que se encuentra del lado del servidor, el principal uso que tiene es la creación de servidores web, además de estar basado en Javascript (NodeJs, 2023)

Esta herramienta permite construir la parte del servidor de nuestras aplicaciones. Cuenta con diferentes herramientas que se integran para facilitar el desarrollo de estos servidores. También existen paquetes de terceros que se pueden integrar con facilidad.

2.3.5. SQL Server

Es un sistema de gestión de base de datos relacionales, al ser un producto de Microsoft permite la fácil integración con productos de Microsoft. (Microsoft, 2022) Una de las principales ventajas de este, es que al desarrollar aplicaciones con este entorno podemos integrar entity framework facilitándonos así el proceso de desarrollo.

2.3.6. PostgreSQL

Es un sistema de gestión de base de datos relacionales, una de las principales características de este, es que es de código abierto. (postgresql, 2023) Este también permite la integración con herramientas como entity framework, sin embargo, al no ser un entorno propio de esta aplicación tiene ciertas dificultades al integrarse.

2.3.7. Mapbox GL JS

Mapbox GL JS es una librería de javascript que permite crear mapas en línea e interactuar con estos a través de las diferentes funcionalidades que presta, con esta librería se puede personalizar mapas para aplicaciones del lado del cliente. (Mapbox, s.f)

Esta librería además nos permite hacer conversión de coordenadas geográficas. Al igual que se puede transformar de una dirección común y corriente, a coordenadas geográficas. Por lo que esta herramienta nos puede ayudar a delimitar si una dirección se encuentra dentro de cierto sector.

2.3.8. Leaflet

Leaflet es una librería de javascript la cual permite la interacción con mapas y usar diferentes funcionalidades que tienen que ver con posiciones geográficas y coordenadas. (Leaflet, 2022)

Esta librería es de código abierto, además de que existen varias herramientas que se pueden acoplar a esta. También presenta funciones para hacer transformaciones de coordenadas geográficas, y también de direcciones comunes.

2.3.9. Signal R

Signal R es una librería de ASP.NET, que permite a las aplicaciones web tener funcionalidades en tiempo real, un uso muy común de esta librería es para realizar chats los cuales funcionan en tiempo real. (Bradygaster, 2022)

Por lo general, el servidor espera que el cliente solicite nuevos datos, con esta librería es posible que el servidor envíe los datos al cliente a medida que esta esté disponible.

CAPÍTULO III: METODOLOGÍA

3. Metodología de desarrollo del plan de tesis

3.1. Enfoque Cualitativo

Una investigación cualitativa es la que permite estudiar como una actividad, relaciones, materiales, etc. Se dan u ocurren en una situación o problema. (Vera Vélez). En este tipo de investigación existen varios tipos los cuales se pueden aplicar, pero el que es más útil en el caso de este trabajo es el de observación.

El objetivo de la investigación cualitativa fue conocer como un restaurante del centro norte de la ciudad de Quito, maneja sus reservas. Para lo cual se realizó una entrevista a uno de estos restaurantes (Referirse al Anexo A). Se plantearon los siguientes temas:

- Confirmaciones en las reservas.
- Intervalos de tiempo entre reservas
- Horarios definidos
- Distribución de las mesas a partir de las reservas

Como resultado de esta investigación, se pudo determinar que: existen restaurantes que realizan confirmaciones de las reservas; el tiempo de entre reservas se define según el restaurante y el tipo de servicios que se ofrezcan en el restaurante; cada restaurante tiene un horario definido de atención; las mesas se distribuyen a partir de cómo se hayan realizado las reservas, por lo que es importante hacer un contraste con el aforo del restaurante.

3.2. Investigación Aplicada

La investigación aplicada es una manera de encontrar soluciones para problemas en un contexto especifico. A diferencia de los otros tipos de investigación, la investigación aplicada sirve para aplicar el conocimiento existente en soluciones a problemas prácticos. (Cordero, 2008)

Esta investigación es usada en varios campos, pero uno de los que más nos interesa es el de la tecnología. En este se presentarán soluciones prácticas para problemas del mundo real, usando la tecnología.

El objetivo de esta investigación fue conocer si los restaurantes del centro norte de la ciudad de Quito manejan reservas. Se tomó un grupo de restaurantes aleatoriamente y se buscó el medio por el cual estos realizan las reservas.

Como resultado de esta investigación se encontró, que la mayoría de los restaurantes del centro norte de la ciudad de Quito manejan reservas. La mayoría de estos, reciben sus reservas a través de redes sociales, o incluso por teléfono.

Los resultados de la investigación sugieren que la mayoría de los restaurantes del centro norte de la ciudad de Quito, no se encuentran registrados en una aplicación web que les permita a los clientes realizar sus reservas. Por lo que el desarrollo de una de estas aplicaciones puede ser apropiado para este sector de la ciudad.

3.2.1. Método de investigación: Observación.

La investigación por observación es un método en el que se presta atención a una actividad, con el objetivo de sacar conclusiones o comprobar una hipótesis sobre esta. Permitiendo obtener datos sin la necesidad de depender de personas que intervengan en la investigación. Existen 2 tipos:

3.2.1.1. Observación Participativa

Es una investigación en la cual el investigador participa en la situación que se está estudiando. (Vera Vélez).

3.2.1.2. Observación No Participativa

Es una investigación en la cual el investigador toma datos mientras observa cómo se realiza la actividad que se está estudiando, se pueden usar simulaciones o estudios de casos para saber cómo los participantes actúan ante una situación específica. (Vera Vélez)

El objetivo de la investigación fue observar los medios por los cuales, los restaurantes del centro norte de la ciudad de Quito reciben sus reservas, para poder implementar una aplicación web que ayude a gestionar las reservas de estos restaurantes.

La investigación se realizó mediante la observación no participante. Se escogió un grupo de restaurantes de manera aleatoria, y se observaron las opciones que brindan estos para realizar una reserva.

Como resultado de la investigación, se observó que muchos de los restaurantes no cuentan con una aplicación web que permita ver su información y realizar una reserva. Estos restaurantes tienen mucha de su información en redes sociales, además de un link que redirige hacia el WhatsApp del restaurante para realizar las reservas.

Los resultados de la investigación sugieren que varios de los restaurantes del centro norte de la ciudad de Quito, no cuentan con una aplicación web que les permita gestionar las reservas. Por lo que sería apropiado el desarrollo de esta aplicación.

3.3. Metodologías Ágiles para desarrollo de software.

Las metodologías ágiles son un tipo de metodologías que pueden ser usadas para desarrollar software, la principal característica de estas es que se adaptan fácilmente a nuevos requerimientos que pueda tener la aplicación.

Entre las principales metodologías ágiles se encuentran Scrum, Kanban y por prototipos. Las cuales se pueden adaptar a diferentes situaciones, dependiendo de las necesidades y del software que se vaya a implementar.

3.3.1. Descripción de las metodologías

3.3.1.1. Scrum

Es una metodología ágil en la cual el equipo aprende a través de la experiencia, aborda problemas y mejora continuamente. Cuenta con una serie de reuniones y herramientas con el objetivo de estructurar el trabajo que se va a realizar.

Permite tener entregas de forma continua y, además, de ser el caso se pueden hacer integraciones del feedback que se reciba por parte del cliente al final de cada sprint. Por lo que hay una mejor respuesta en caso de que cambien las tendencias del mercado. (Drumond)

Entre los actores que intervienen se encuentran:

Desarrolladores: Los desarrolladores son el grupo de personas encargadas de la implementación del código, en la metodología tienen responsabilidades como:

- Definir los tiempos en las tareas que se encuentren designados en el sprint backlog.
- Adaptar las tareas al objetivo del sprint.
- Cumplir los criterios de aceptación de las tareas.

Product Owner: Es la persona responsable de incrementar el valor del producto, entre sus principales funciones en la metodología están:

- Comunicar al equipo las nuevas funcionalidades de la aplicación.
- Definir el orden de las tareas del product backlog.
- Garantizar que el product backlog y las tareas estén claras para todo el equipo.
- Definir los criterios de aceptación de las tareas.

Scrum Master: Es la persona responsable de asegurar que se cumpla con la metodología y que todas las personas del equipo cumplan su trabajo en el sprint y fechas acordadas, entre sus funciones se encuentran:

- Manejo de las daily scrum meetings.
- Ayudar a que el equipo entienda la planificación de tareas.
- Definir en la organización como se va a manejar la metodología.
- Garantizar que el equipo y la organización hagan el uso correcto de la metodología.

Para la correcta gestión de la metodología se manejan diferentes tipos de eventos, los cuales se presentan a continuación:

Sprint: Los sprints son periodos de tiempo en los que se cumplirá un objetivo del producto, pueden ser tanto de un mes como de una semana, pero deben ser consistentes en el tiempo. Cuando un sprint termina el próximo empieza. Cada uno tiene un objetivo, por lo cual todos los eventos que se presentan a continuación sucederán en cada sprint.

Sprint Planning: Es una reunión en la que todo el equipo asiste, donde lo primero que indica el Product Owner es como quiere mejorar o implementar algo nuevo al producto en el sprint. Una vez se tiene definido y todo el equipo conoce el objetivo del sprint, se planifica que tareas del product backlog se van a realizar en el sprint, basándose en el objetivo de ese sprint y se asignan a los desarrolladores. Una vez asignadas las tareas, los desarrolladores pueden dividir estas tareas en tareas más pequeñas, las cuales tomen un día o dos y estimaran el tiempo para cada una.

Daily Scrum: Es una reunión que se realiza al iniciar el día, a la que asisten el equipo de desarrolladores y el scrum master. El objetivo de esta reunión es tener un control de cómo se van realizando las actividades y si tienen algún impedimento. Por lo que, cada desarrollador en la reunión responderá 3 preguntas, las cuales son ¿Qué hice ayer?, ¿Qué hare hoy? Y ¿Existe alguna dependencia o impedimento para avanzar con el trabajo?

Sprint Review: Es una reunión a la cual asisten todos los miembros del equipo. El propósito de esta reunión es asegurarse que se cumplió con el objetivo del sprint. Para esto cada

desarrollador presenta su trabajo y el product owner puede comprobar que se hayan cumplido los criterios de aceptación de cada tarea.

Sprint Retrospective: Es una reunión en la cual se reúne el scrum master con el equipo de desarrollo para buscar mejoras en cuanto al cómo se realizaron las cosas en el último sprint.

Para el manejo de las tareas y documentación de esta metodología se usan diferentes herramientas, entre las que se encuentran:

Product Backlog: Es una lista ordenada de tareas que definen que se va a mejorar, implementar, corregir.

Sprint Backlog: Es una lista ordenada de las tareas que se van a realizar en el sprint, estas tareas provienen del Product Backlog.

3.3.1.2. Kanban

Es una metodología ágil que permite la visualización del flujo de trabajo a través de un tablero Kanban. Este tablero se divide en varias columnas como, pendiente, en curso y terminado. En el cual se crean tarjetas para la representación de tareas, y estas tarjetas irán cambiando de columna en el tablero dependiendo del estado de esta.

En esta metodología es fácil agregar nuevas tareas, y a la misma vez es fácil eliminar tareas que no son necesarias. Lo que permite tener un seguimiento visual del trabajo. A diferencia de Scrum este no maneja fechas de sprints, sino que es un proceso continuo.

3.3.1.3. Por prototipos

Es una metodología ágil que se basa en la mejora continua de prototipos, esta mejora se puede hacer a través del feedback del cliente o a través de pruebas del prototipo. Lo que permite que los equipos de desarrollo se puedan adaptar fácilmente a cambios y también a nuevas funcionalidades que se requieran desarrollar.

Los pasos que se siguen en esta metodología son:

1. Definición de requerimientos.
2. Diseño de la aplicación.
3. Creación de un prototipo inicial.
4. Pruebas o retroalimentación del cliente.
5. Mejora del prototipo.
6. Repetir el paso 4 y 5 hasta alcanzar el nivel de calidad que se desea.

En esta metodología no se definen reuniones para el seguimiento de las tareas de los desarrolladores como en scrum, ni tampoco se tiene una herramienta de visualización del flujo de trabajo, sin embargo, si se definen ciclos iterativos de trabajo.

3.3.2. Modelo que aplica

A continuación, se presenta una tabla comparativa de las metodologías antes mencionadas:

Tabla 1 Comparación de metodologías ágiles

	Scrum	**Kanban**	**Por prototipos**
Ventajas	• Tiene ciclos iterativos • Es adapta a proyectos en los que trabajan varias personas • Se recibe feedback y se trabaja sobre el en el siguiente ciclo iterativo	• Entrega continua de resultados • Presentación grafica del flujo del trabajo • Fácil de implementar	• Presenta prototipos iterativamente • Se adapta fácilmente a proyectos de 1 persona • Entrega rápida de valor • Se recibe feedback y se trabaja sobre el en el siguiente ciclo iterativo
Desventajas	• Es difícil de adaptar a proyectos de 1 persona • Tiene varias reuniones	• No tiene ciclos iterativos • No tiene fases en las cuales guiarse	• Los siguientes ciclos iterativos dependen del feedback del cliente o de pruebas • Los ciclos iterativos terminan con la presentación de un prototipo por lo que no son ciclos de tiempo cortos

Nota: Esta tabla muestra la comparación de 3 metodologías ágiles que se pueden usar.

La metodología que se elige a partir de la comparación es la metodología ágil por prototipos. Las razones que se presentan para elegir esta metodología es que permite la mejora continua del producto. Una de las principales ventajas que tiene esta sobre scrum, es que no necesita una cantidad de personas como mínimo para asignar roles y tener las reuniones.

Al ser un proyecto de 1 persona la asignación de roles carece de sentido al igual que muchos de los eventos que maneja scrum. En cambio, Kanban no permite trabajar con ciclos iterativos como se presenta en la metodología que se eligió.

La metodología ágil por prototipos permitirá manejar intervalos de tiempo que permitan implementar nuevas funcionalidades y mejoras, además de una adaptación fácil al equipo de trabajo con el que se cuenta.

3.4. Elección de tipo de aplicación

Tabla 2 Comparación de tipos de aplicaciones

	Web	**Mobile**	**Desktop**
Ventajas	• Se pueden acceder desde cualquier dispositivo que tenga un navegador • No necesitan ser instaladas • Si se actualiza la aplicación no se requiere que los dispositivos las instalen	• Mayor confiabilidad al ser descargadas desde una tienda de aplicaciones • Se pueden enviar notificaciones push • Acceso a funcionalidades del dispositivo	• Acceso a recursos del sistema operativo • Funcionamiento offline • Interfaz de usuario personalizable
Desventajas	• Poca accesibilidad a funcionalidades de los dispositivos • Depende de una conexión a internet	• Aprobación de tiendas de aplicaciones • Se necesitan instalar en el dispositivo • Cuando existe una actualización se debe instalar	• Requiere ser instalada en el dispositivo • No se puede acceder desde otros dispositivos • Requieren actualizaciones

Se decidió hacer una aplicación web, basándonos en la comparación de los distintos tipos de aplicaciones. En el presente trabajo de titulación se busca una mayor accesibilidad para todo tipo de usuarios. Permitiendo a los usuarios acceder a la aplicación desde cualquier dispositivo que

cuente con un navegador web. Con lo que se puede dar una mejor experiencia a los usuarios, ya que al usar este tipo de aplicación no tienen restricciones de sistema operativo, y no hay necesidad de instalar aplicaciones adicionales

3.5. Elección de Herramientas

3.5.1. Herramienta para el desarrollo del Front End

Tabla 3 Comparación de herramientas del front end

	Angular	**React**
Ventajas	• Utiliza TypeScript • Estructura de carpetas y archivos intuitiva • Fácil modularización de componentes • Documentación oficial	• Utiliza JavaScript • Reutilización de componentes • Fácil configuración Inicial • Documentación Oficial
Desventajas	• Curva de aprendizaje • Mayor complejidad de configuración inicial	• La estructura de las carpetas y archivos depende del desarrollador • Menos funcionalidades integradas

Después de comparar estas 2 herramientas, se decide trabajar con Angular. Esta decisión se basa principalmente en la experiencia previa que se tiene de la aplicación, además de que permite una buena personalización de los componentes. Esto será de gran ayuda al momento de desarrollar la aplicación, y al tener conocimientos previos sobre la herramienta no habrá inconveniente con la curva de aprendizaje.

3.5.2. Herramienta para el desarrollo backend

Tabla 4 Comparación de herramientas del back end

	.Net	**NodeJS**
Ventajas	• Amplio conjunto de librerías y paquetes • Cuenta con un ecosistema que permite el fácil desarrollo de las aplicaciones • Soporte por parte de microsoft	• Al ser basado en javascript su curva de aprendizaje es más fácil • Tiene un amplio ecosistema de módulos y paquetes a través de npm
Desventajas	• Mayor curva de aprendizaje • Tiene menor cantidad de paquetes que NodeJS	• La estabilidad puede ser afectada al tener un mal manejo de los callbacks • Dependencia de la comunidad para actualizaciones de librerías y módulos

En base a la comparación de estas dos herramientas por razones estratégicas y prácticas, se decide trabajar con .Net, en específico con .Net Core para el desarrollo de la aplicación. La principal razón fue la capacidad de aprovechar Entity Framework, al tener conocimientos sobre este framework habrá una ventaja al momento de desarrollar la aplicación.

Otra de las razones es el entorno de desarrollo que se tiene sobre .Net Core para la implementación de las aplicaciones. Además de las facilidades que brindan, la elección también se basó en la experiencia previa trabajando con .Net Core. Por lo que la curva de aprendizaje no será un impedimento.

3.5.3. Herramienta para obtener la ubicación y manejo de mapas

Tabla 5 Comparación de herramientas para mapas

	Leaflet	**Mapbox GL JS**
Ventajas	• Personalización mediante CSS • Fácil de usar • Buena documentación oficial • Buena integración con React • Personalización de mapas	• Personalización a través de Mapbox Studio • Fácil de usar • Buena documentación además de soporte técnico • Buena integración con Angular • Personalización de mapas
Desventajas	• Pueden existir errores al momento de usar la personalización mediante estilos • Existen varias herramientas para integración, pero muchas no cuentan con su debida documentación.	• Acepta otros proveedores de mapas, pero son limitados • Tiene muchas herramientas que se pueden usar, pero algunas son de costo

Una vez comparadas las herramientas, se tomó la decisión de usar Mapbox para la integración de mapas en nuestra aplicación. Una de las principales razones para elegir esta herramienta, es la capacidad de adaptarse mejor a las otras herramientas con las que vamos a trabajar. Además de que esta herramienta permite tener una amplia gama de opciones de personalización y estilos para los mapas.

3.5.4. Motor de base de datos

Tabla 6 Comparación de Motores de base de datos

	SQL Server	PostgreSQL
Ventajas	• Fácil integración con Entity Framework • Fuerte integración con herramientas de Microsoft como .Net	• Licencia de código abierto • Gran comunidad y soporte • Mayor enfoque en estándares de SQL
Desventajas	• Limitaciones en plataformas que no son de windows • Licencia propietaria y costo asociado	• Menor integración con herramientas y servicios de Microsoft • Algunas características pueden ser más complejas de configurar

Una vez comparadas las herramientas se eligió SQL Server como base de datos, debido a la integración que esta tiene con las herramientas que vamos a trabajar, que son del entorno de Microsoft. Además de que el uso de Entity Framework permite el desarrollo de la aplicación usando esta base de datos.

La compatibilidad de .Net Core con SQL Server permitirá aprovechar al máximo las capacidades que tienen estas dos herramientas, ya que las 2 pertenecen al entorno de Microsoft.

CAPÍTULO IV: DESARROLLO DE LA INVESTIGACIÓN

4.1. Análisis de los requerimientos.

4.1.1. Propósito.

Se tiene como objetivo definir los requerimientos de la aplicación web, los cuales se definen en base a la observación de los restaurantes que manejan reservas en el centro norte de la ciudad de Quito.

4.1.2. Funcionalidades del Producto.

En base a lo que se observó de como los restaurantes del centro norte de la ciudad de Quito manejan sus reservas, se definieron las principales funcionalidades:

F1. Registro de Restaurantes

Descripción: Los restaurantes podrán crear un usuario y contraseña, con los que después podrán acceder a la aplicación. Además de eso registrarán toda la información de este y configurarán las opciones que tendrán.

Requerimientos funcionales:

REQ-1: Crear registro y login. La aplicación tendrá una página con un formulario que permitirá a los restaurantes registrar su usuario y contraseña, también se comprobará que el restaurante se encuentre en el área permitida, y contará con una página en la que pueda iniciar sesión a partir de los datos registrados.

REQ-2: Crear registro de datos del restaurante. La aplicación tendrá una página con un formulario que permitirá a los restaurantes registrar toda la información que aparecerá en la aplicación.

REQ-3: Crear página de configuración del restaurante. La aplicación tendrá una página en la que se permitirá modificar las diferentes configuraciones con las que contaran los restaurantes.

F2. Gestionar las reservas

Descripción: Los restaurantes podrán ver las reservas que tienen para la fecha actual sin necesidad de actualizar la página, para las fechas de los otros días podrán filtrarlas según la fecha, podrán crear nuevas reservas, editar y borrar.

Requerimientos Funcionales:

REQ-4: Ver las reservas de la fecha actual. La aplicación tendrá una página que permita a los restaurantes ver las reservas, sin necesidad de actualizar la pantalla. En esta página también se permitirá asignar una mesa a una reserva cuando lleguen los comensales.

REQ-5: Ver las reservas según la fecha. La aplicación tendrá una página que permita a los restaurantes ver las reservas según la fecha que escojan. En esta página también se permitirá confirmar una reserva, editarla o eliminarla.

REQ-6: Crear nueva reserva. La aplicación tendrá una página que permita a los restaurantes crear una nueva reserva. En esta página se presentará un formulario en el que se ingresaran los datos de la reserva.

F3. Lista de restaurantes

Descripción: Existirá una página pública en la que los comensales que ingresen podrán ver la lista de los restaurantes. Los restaurantes que aparezcan en esta página serán los que tengan activa la opción para aparecer en el pool de restaurantes.

Requerimientos funcionales:

REQ-7: Página para lista de restaurantes. La aplicación tendrá una página en la que aparezcan todos los restaurantes, que tienen activada la opción para aparecer en el pool de restaurantes.

REQ-8: Búsqueda de restaurantes. En la página para lista de restaurantes se podrá buscar el restaurante por el nombre.

F4. Crear reserva.

Descripción: Cada restaurante tendrá una página en donde los comensales podrán crear una reserva. En esta página también se encontrará la información de cada restaurante.

Requerimientos funcionales:

REQ-9: Página por restaurante. Cada restaurante tendrá una página en la que se visualizará la información de este.

REQ-10: Nueva reserva. En esta página existirá un formulario para crear una nueva reserva. El comensal elegirá la fecha, hora, numero de comensales, celular, comentarios.

Requerimientos no funcionales

REQNF-1: La aplicación debe ser accesible desde un navegador web.

REQNF-2: La aplicación debe tener una base de datos y estar conectada a la misma.

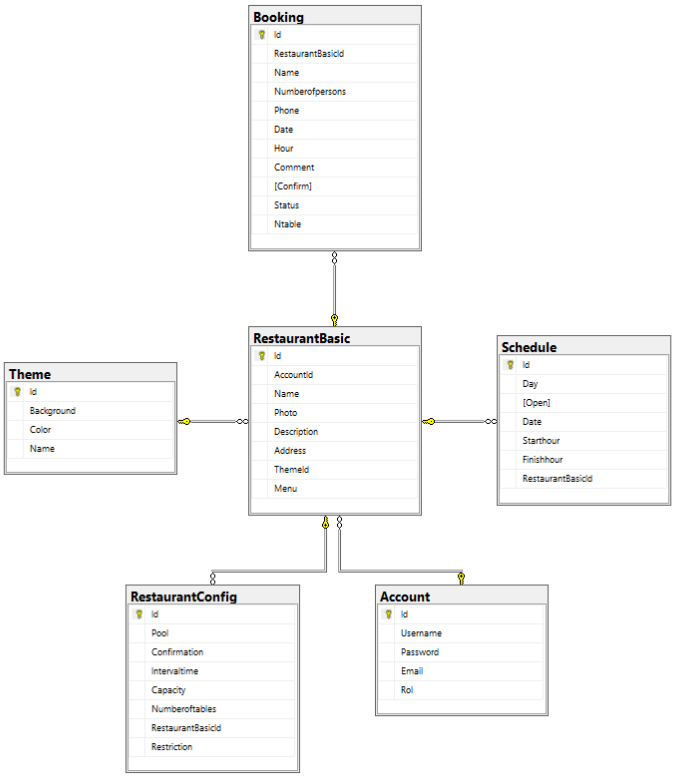
REQNF-3: La aplicación debe ser compatible con varios navegadores de internet.

REQNF-4: La aplicación debe tener interfaces intuitivas y amigables con el usuario.

4.2. Modelo de la base de datos.

En base a los requerimientos definidos se presenta el siguiente modelo para la base de datos:

Figura 1 Modelo de la base de datos



4.3. Casos de uso

A continuación, se presentan los casos de uso de las funcionalidades antes descritas:

Tabla 7 Caso de uso CU1

Caso de Uso	Registro del restaurante	**Identificador:** CU1
Actores	Restaurante	
Tipo	Primario	
Referencias	REQ-1, REQ-2, REQ-3	
Precondición		
Postcondición	CU2	
Descripción	Registrar los datos del restaurante y crear un usuario y contraseña	
Resumen	El restaurante presiona en el botón de registrar, llena el formulario y se registra el restaurante.	

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 8 Flujo del caso de uso CU1

Nro.	Ejecutor	Paso o Actividad
1	Restaurante	Presiona el botón de registrar
2	Sistema	Abre la página de registro
3	Restaurante	Ingresa usuario, contraseña y email
4	Restaurante	Presiona en siguiente
5	Sistema	Valida los datos ingresados
6	Sistema	Abre la siguiente página de registro de información
7	Restaurante	Ingresa los datos requeridos en el formulario
8	Restaurante	Presiona en siguiente
9	Sistema	Valida los datos ingresados
10	Sistema	Abre la siguiente página de configuración del restaurante
11	Restaurante	Elige y llena los datos requeridos
12	Restaurante	Presiona en registrar
13	Sistema	Valida los datos ingresados
14	Sistema	Crea el restaurante

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 9 Caso de uso CU2

Caso de Uso	Login	**Identificador:** CU2
Actores	Restaurante	
Tipo	Primario	
Referencias	REQ-1	
Precondición	CU1	
Postcondición		
Descripción	Inicio de sesión y validación de datos del restaurante	
Resumen	El restaurante presiona en el botón de inicio de sesión, ingresa el usuario y contraseña y el sistema valida sus datos	

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 10 Flujo del caso de uso CU2

Nro.	Ejecutor	Paso o Actividad
1	Restaurante	Presiona en iniciar sesión
2	Sistema	Abre la página para iniciar sesión
3	Restaurante	Ingresa usuario y contraseña
4	Sistema	Valida los datos ingresados
5	Sistema	Si los datos son incorrectos imprime error
6	Sistema	Si los datos son correctos envía al restaurante a la página de gestión de reservas.

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 11 Caso de uso CU3

Caso de Uso	Ver las reservas	**Identificador:** CU3
Actores	Restaurante	
Tipo	Primario	
Referencias	REQ-4, REQ-5	
Precondición		
Postcondición		
Descripción	Ver las reservas que un restaurante tiene para el mismo, día y para el resto de los días según la fecha.	
Resumen	El restaurante presiona en el botón de home y se desplegaran las reservas que sean para ese mismo día, si el restaurante presiona sobre la reserva podrá ver la información de esta. El restaurante presiona el botón de reservas y aplasta en el botón para elegir una fecha, al elegir la fecha se podrán ver las reservas de la fecha elegida	

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 12 Flujo del caso de uso CU3

Nro.	Ejecutor	Paso o Actividad
1	Restaurante	Presiona el botón home
2	Sistema	Abre la página de reservas de ese día
3	Sistema	Imprime las reservas de ese día
4	Restaurante	Presiona sobre una reserva
5	Sistema	Muestra la información de la reserva
6	Restaurante	Presiona asignar mesa
7	Sistema	Abre un modal para ingresar el número de mesa
8	Restaurante	Ingresa el número de mesa
9	Restaurante	Presiona en asignar
10	Sistema	Registra el número de mesa
11	Restaurante	Presiona en buscar
12	Restaurante	Ingresa el nombre del comensal
13	Sistema	Imprime la reserva buscada

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 13 Caso de uso CU4

Caso de Uso	Crear reserva (Restaurante)	**Identificador:** CU4
Actores	Restaurante	
Tipo	Primario	
Referencias	REQ-6	
Precondición		
Postcondición		
Descripción	El restaurante podrá crear un ingresar los datos del comensal y crear una reserva	
Resumen	El restaurante presiona en el botón de crear reserva y llenara los datos del comensal y los datos de la reserva, luego presiona crear reserva y se indica un mensaje de que se creó la reserva	

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 14 Flujo del caso de uso CU4

Nro.	Ejecutor	Paso o Actividad
1	Restaurante	Presiona en el botón de crear reserva
2	Sistema	Abre página para crear la reserva
3	Restaurante	Ingresa los datos del formulario
4	Restaurante	Presiona en crear
5	Sistema	Valida los datos
6	Sistema	Registra la reserva

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 15 Caso de uso CU5

Caso de Uso	Lista de restaurantes	**Identificador:** CU5
Actores	Comensales	
Tipo	Primario	
Referencias	REQ-7, REQ-8	
Precondición		
Postcondición		
Descripción	Los comensales podrán ingresar a la página y ver los diferentes restaurantes registrados en la aplicación y buscar entre estos restaurantes	
Resumen	El comensal abre la página de la aplicación y vera los restaurantes registrados, si aplasta en el botón de buscar, ingresara el nombre del restaurante y se presentara el restaurante buscado	

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 16 Flujo del caso de uso CU5

Nro.	Ejecutor	Paso o Actividad
1	Comensal	Abre la página de la aplicación
2	Sistema	Verifica que restaurantes tienen la opción para aparecer en la lista de restaurantes
3	Sistema	Imprime los restaurantes
4	Comensal	Presiona en buscar
5	Comensal	Ingresa el nombre del restaurante
6	Sistema	Imprime el restaurante buscado

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 17 Caso de uso CU6

Caso de Uso	Crear Reserva (Comensal)	**Identificador:** CU6
Actores	Comensales	
Tipo	Primario	
Referencias	REQ-9	
Precondición		
Postcondición		
Descripción	Los comensales podrán realizar una reserva en un restaurante en especifico	
Resumen	El comensal presiona en un restaurante y se abrirá una página con los datos del restaurante y un formulario, el comensal llenará los datos del formulario y aplastará en crear reserva.	

Fuente: Elaboración en base a pmoinformatica.com (2014)

Tabla 18 Flujo del caso de uso CU6

Nro.	**Ejecutor**	**Paso o Actividad**
1	Comensal	Presiona sobre un restaurante
2	Sistema	Abre la página del restaurante elegido
3	Comensal	Llena los datos de la reserva
4	Comensal	Presiona en crear reserva
5	Sistema	Valida los datos ingresados
6	Sistema	Registra la reserva

Fuente: Elaboración en base a pmoinformatica.com (2014)

4.4. Diseño de las pantallas

En base a los requerimientos y funcionales definidas se presentan las siguientes pantallas:

Figura 2 Diseño de la pantalla home de restaurantes

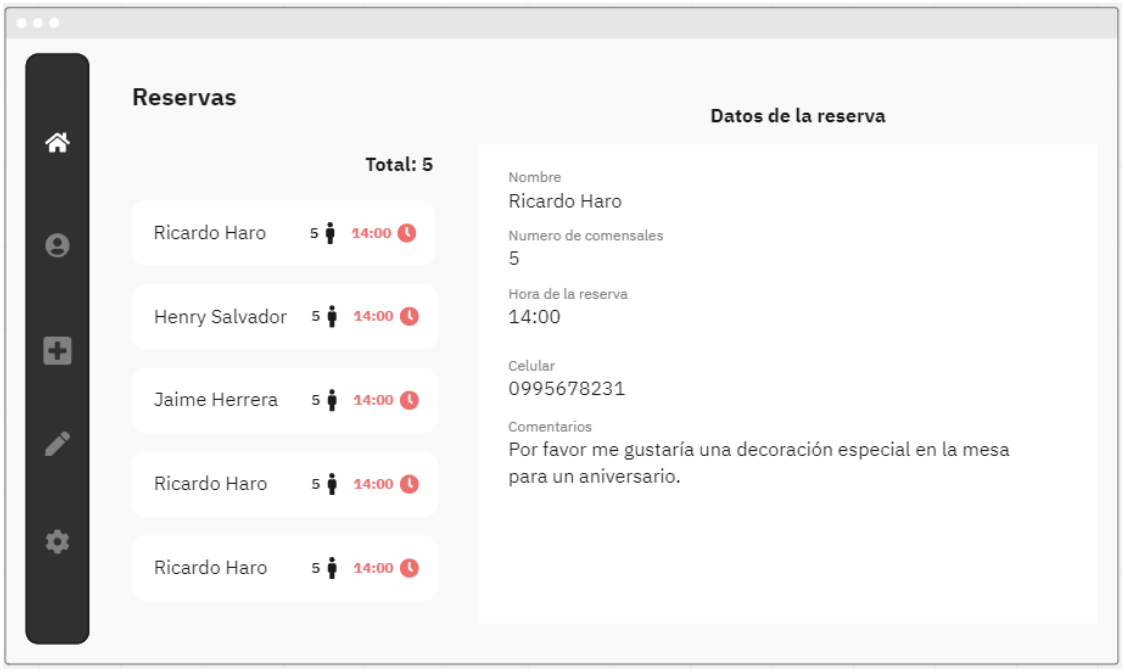


Figura 3 Diseño de la pantalla de reservas

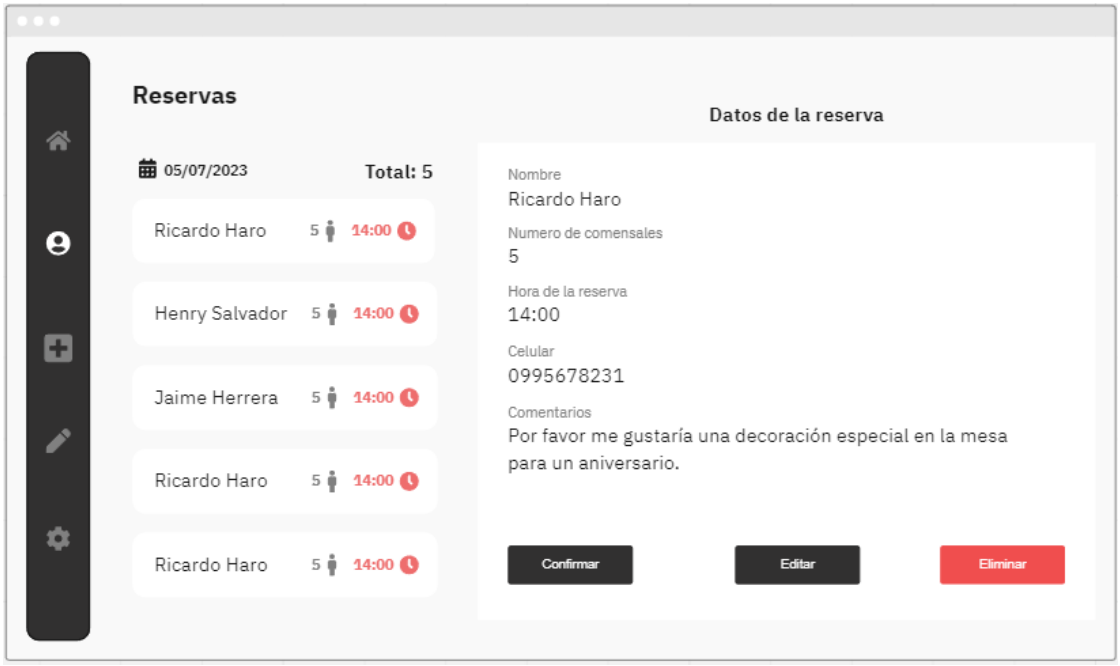


Figura 4 Diseño de la pantalla nueva reserva (restaurantes)

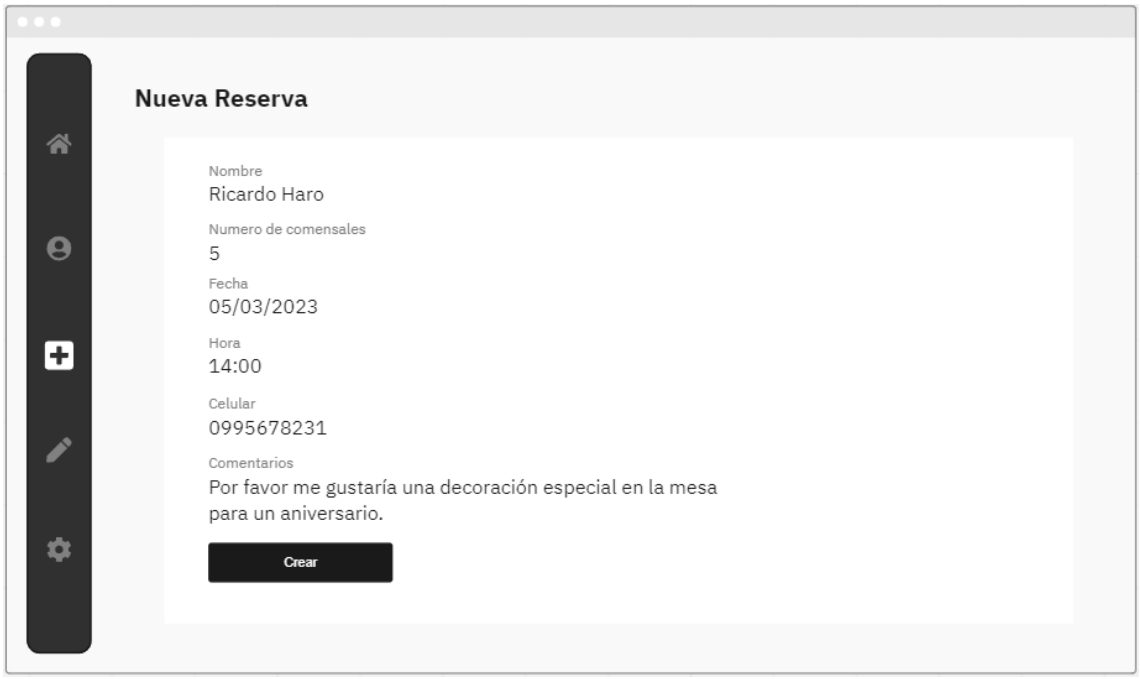


Figura 5 Diseño de la pantalla editar perfil

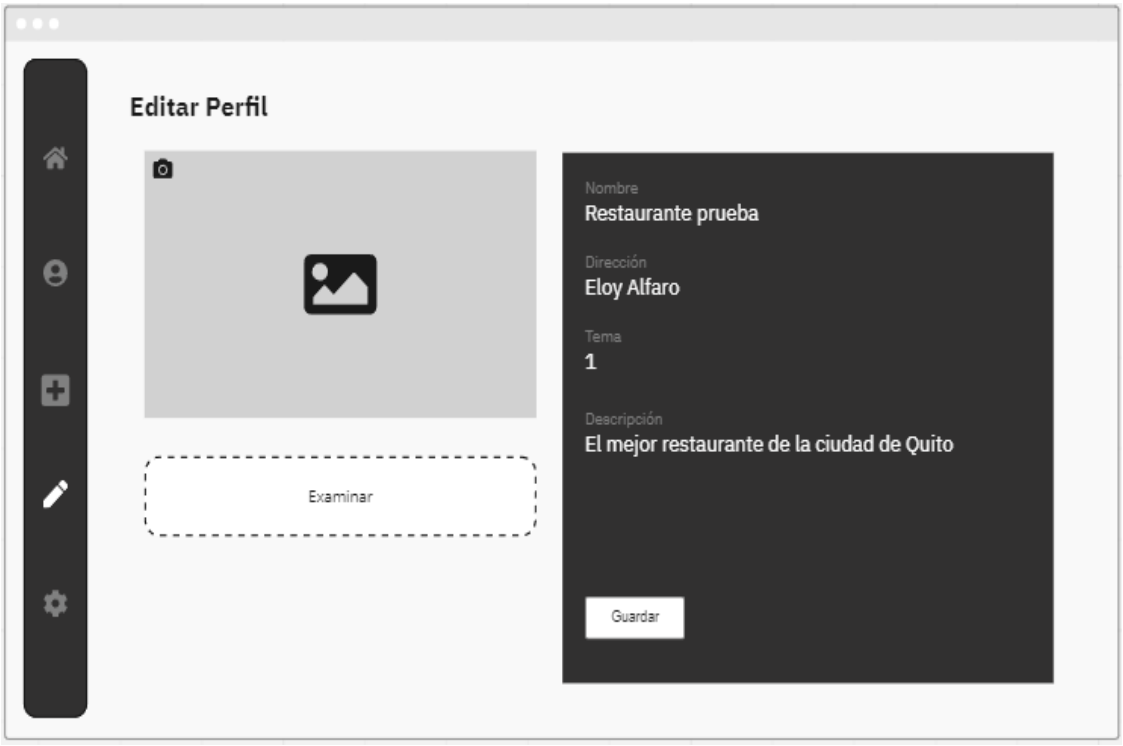


Figura 6 Diseño de la pantalla configuración

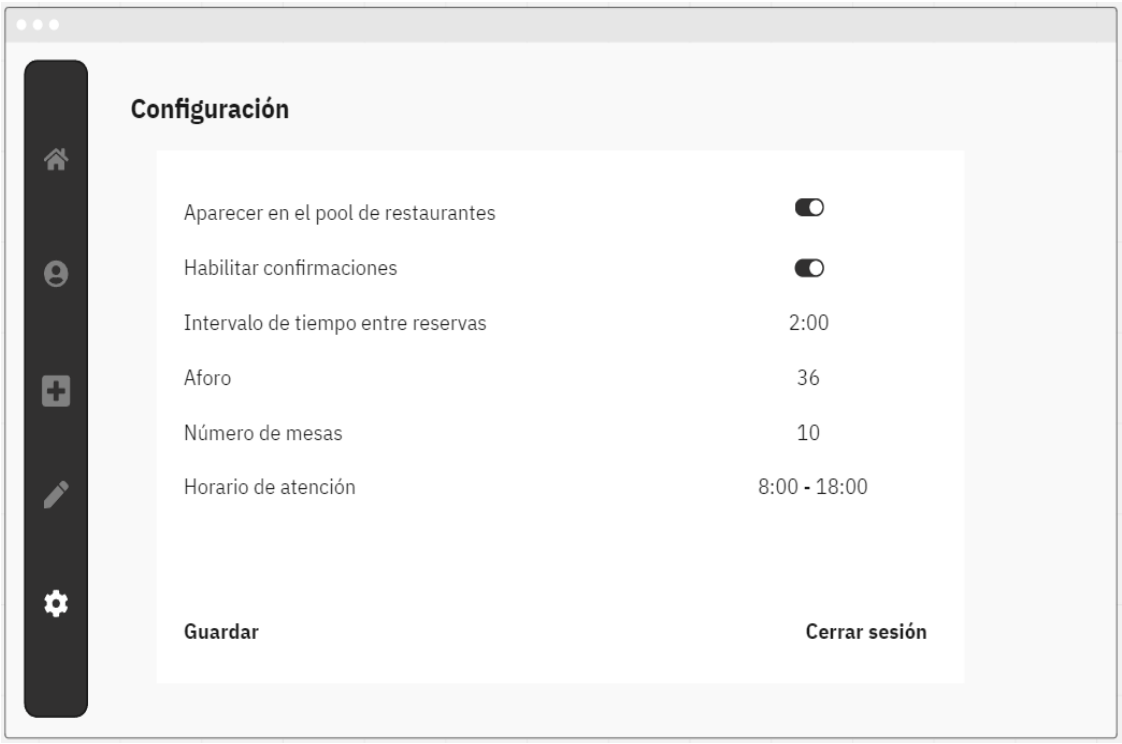


Figura 7 Diseño pantalla home

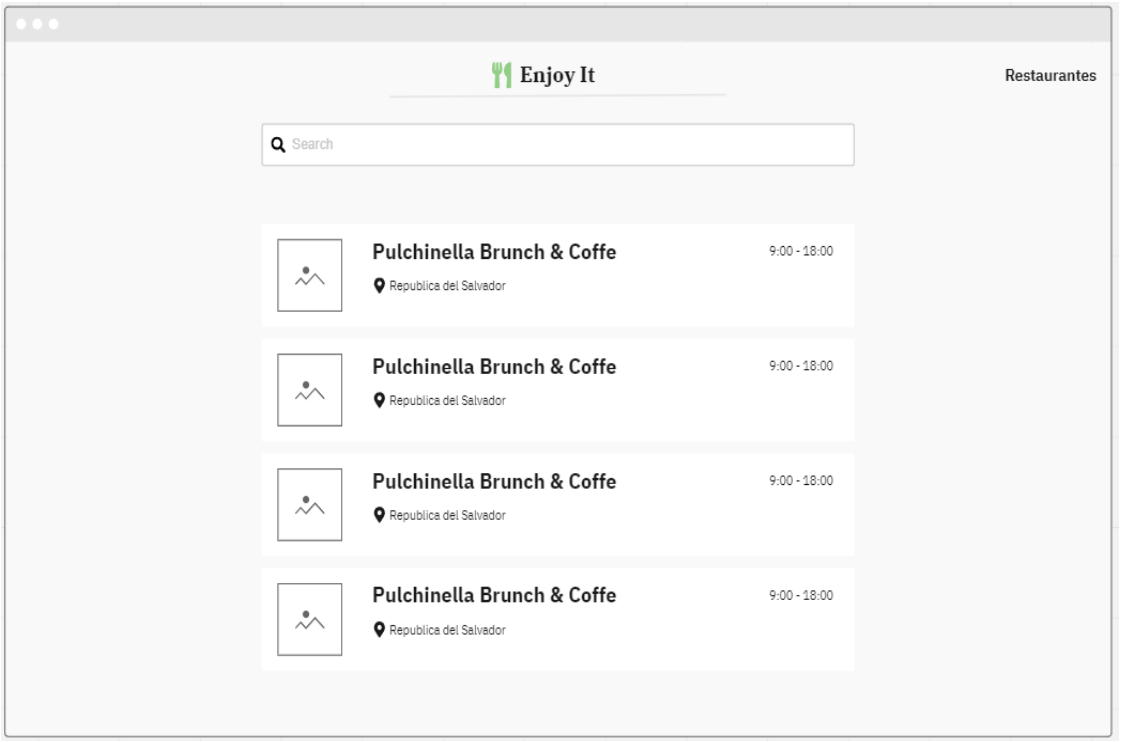


Figura 8 Diseño pantalla restaurante

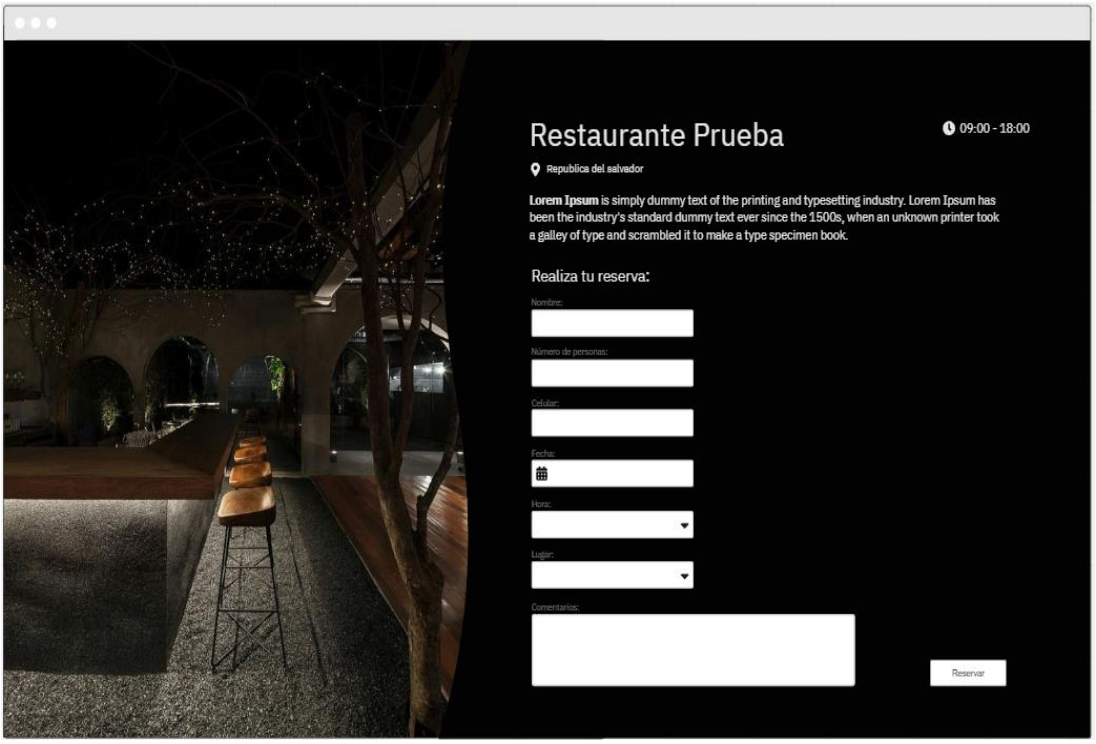
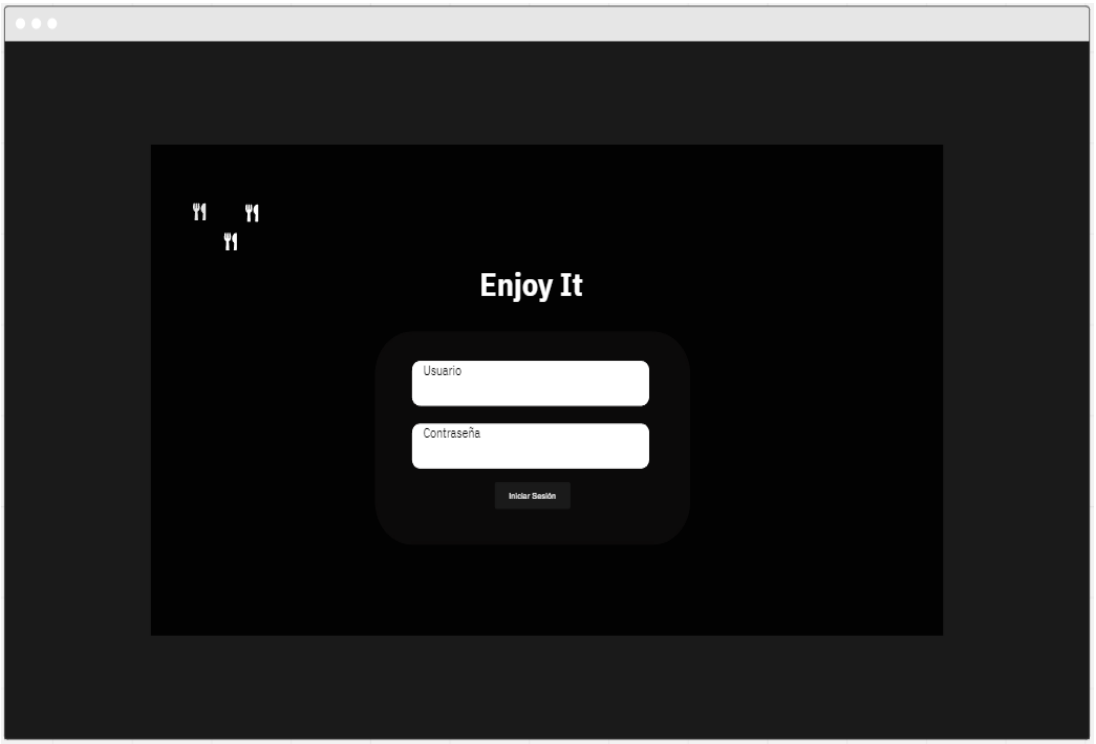


Figura 9 Diseño pantalla Home



5. Implementación de la aplicación

5.1. Pruebas funcionales.

Estas pruebas permiten verificar si la ampliación cumple con los requerimientos y funcionalidades planteadas. Para lo cual se presentan pruebas por cada caso de uso, tomando en cuenta los diferentes escenarios que se pueden encontrar para cada uno de estos.

5.1.1. Caso de uso: Registro del restaurante

Identificador: CU1

Descripción: Registrar los datos del restaurante y crear un usuario y contraseña

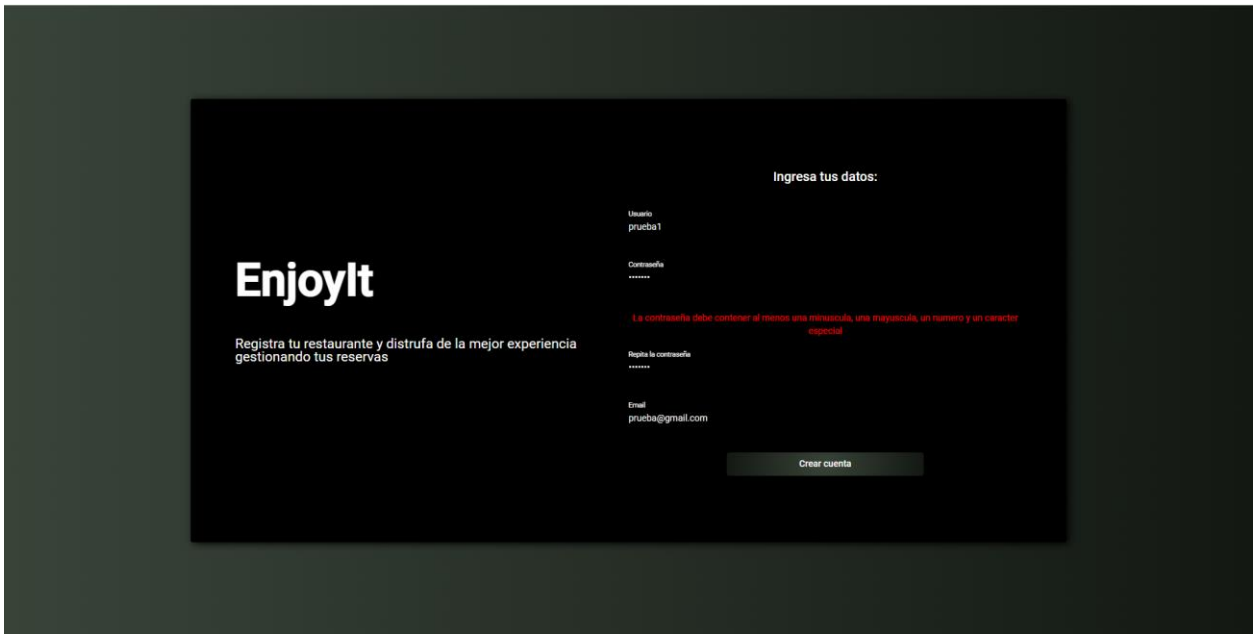
Resultados esperados:

- El nuevo restaurante registra una contraseña poco segura y recibe un mensaje de error
- El nuevo restaurante en la verificación de la contraseña ingresa diferentes contraseñas y recibe un mensaje de error
- El nuevo restaurante crea su usuario y registra exitosamente sus datos
- El nuevo restaurante elije la ubicación en el mapa y obtiene la dirección señalada
- El nuevo restaurante ingresa los datos públicos y se registran con éxito
- El nuevo restaurante configura la aplicación y se guardan con éxito

Resultados reales:

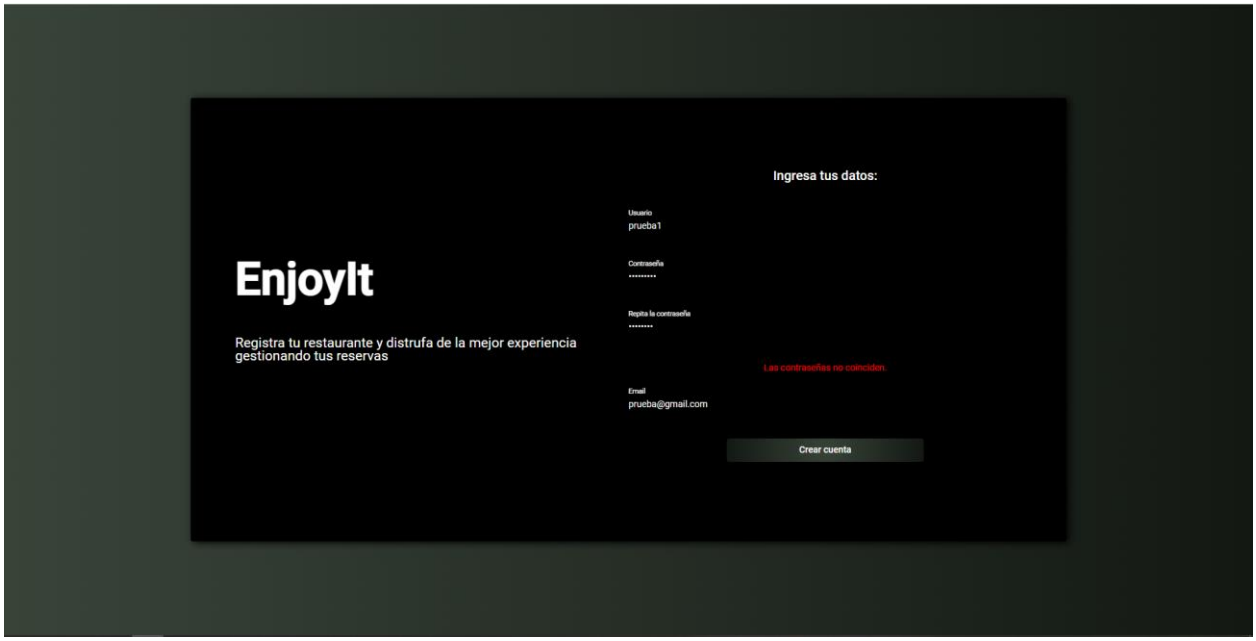
- El nuevo restaurante registra una contraseña poco segura y recibe un mensaje de error

Figura 10 Error contraseña insegura



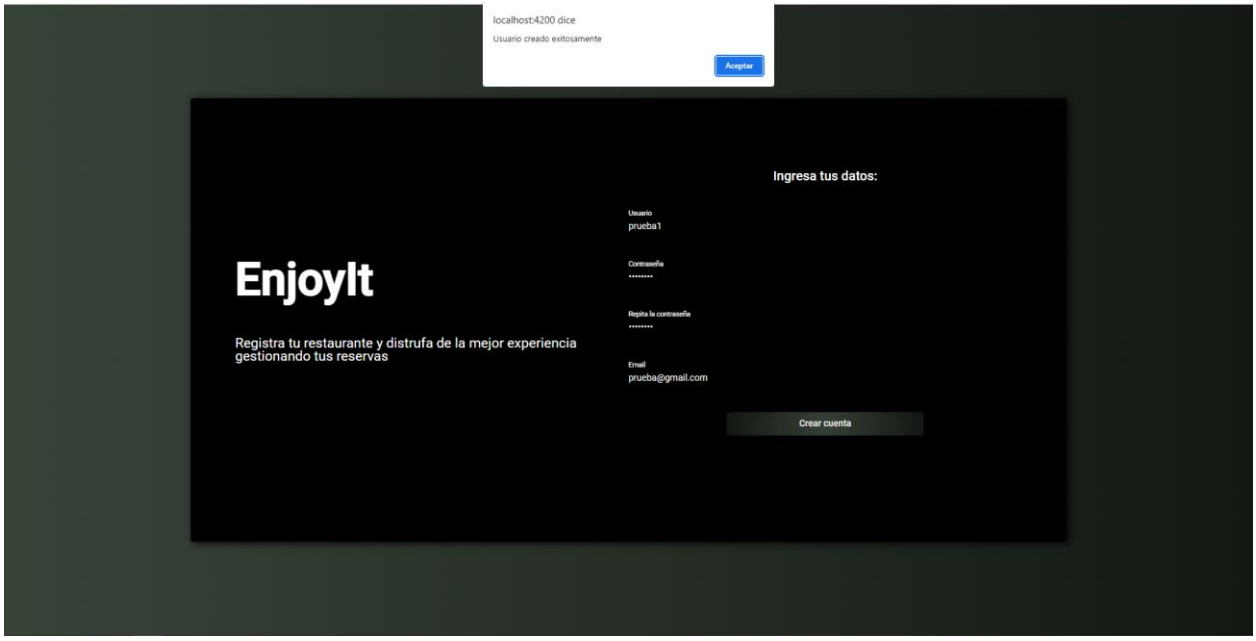
- El nuevo restaurante en la verificación de la contraseña ingresa diferentes contraseñas y recibe un mensaje de error

Figura 11 Error contraseñas no coinciden



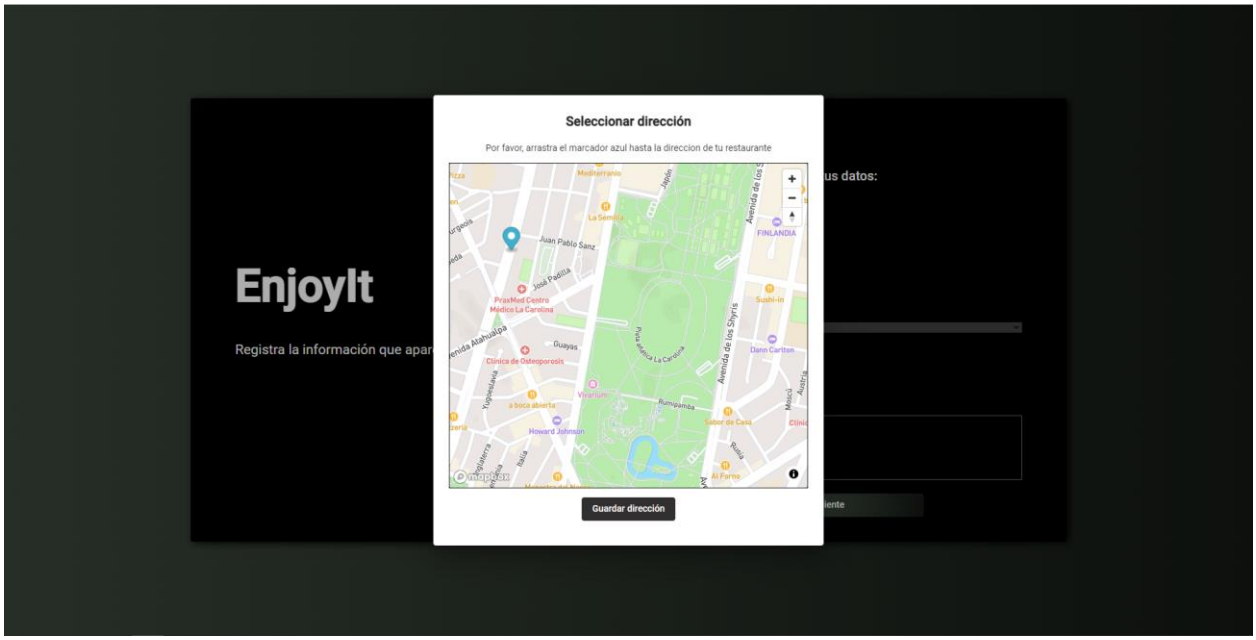
- El nuevo restaurante crea su usuario y registra exitosamente sus datos

Figura 12 Usuario creado con éxito



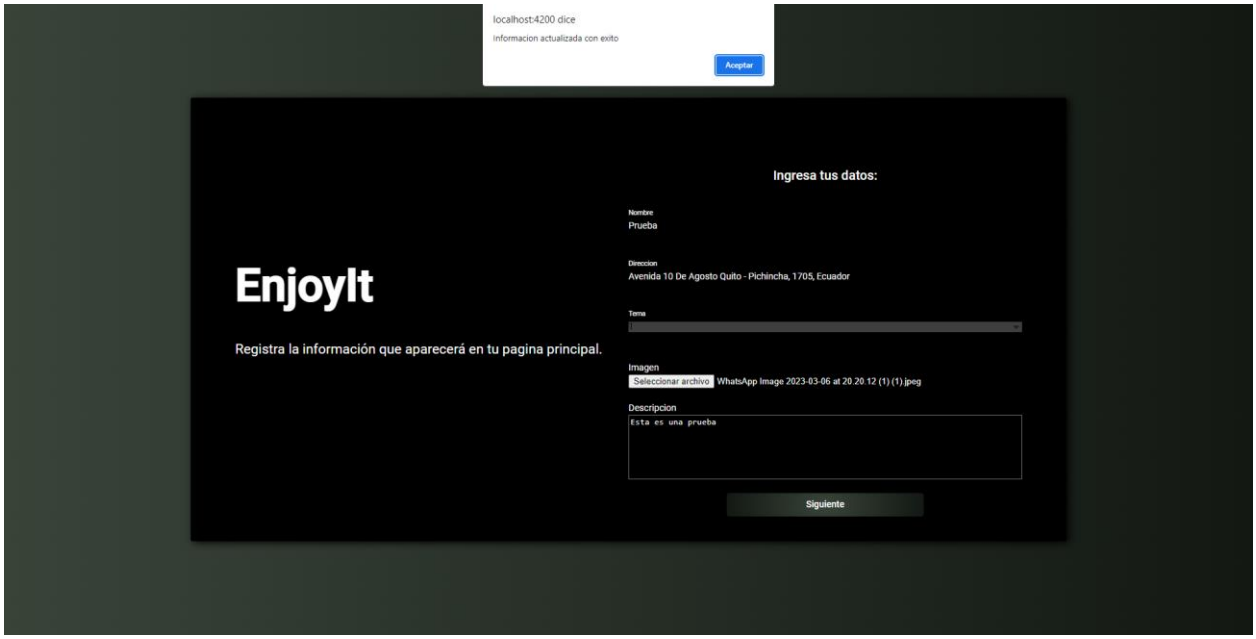
- El nuevo restaurante elije la ubicación en el mapa y obtiene la dirección señalada

Figura 13 Seleccionar ubicación del restaurante



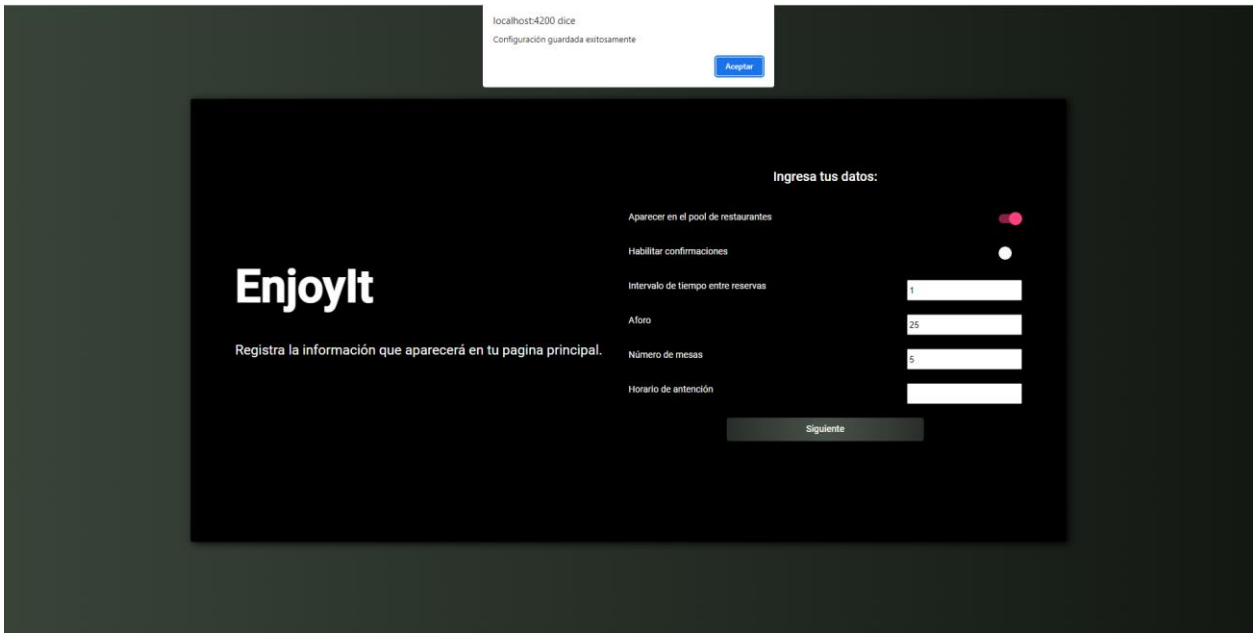
- El nuevo restaurante ingresa los datos públicos y se registran con éxito

Figura 14 Registro de datos públicos exitoso



- El nuevo restaurante configura la aplicación y se guardan con éxito

Figura 15 Configuración del restaurante exitosa



5.1.2. Caso de uso: Login

Identificador: CU2

Descripción: Inicio de sesión y validación de datos del restaurante

Resultados esperados:

- Si el usuario y la contraseña son correctos se redirige a la pantalla home de gestión de las reservas.
- Si el usuario y la contraseña son incorrectos se muestra mensaje de error.

Resultados reales:

- Si el usuario y la contraseña son correctos se redirige a la pantalla home de gestión de las reservas.

Figura 16 Login de restaurantes

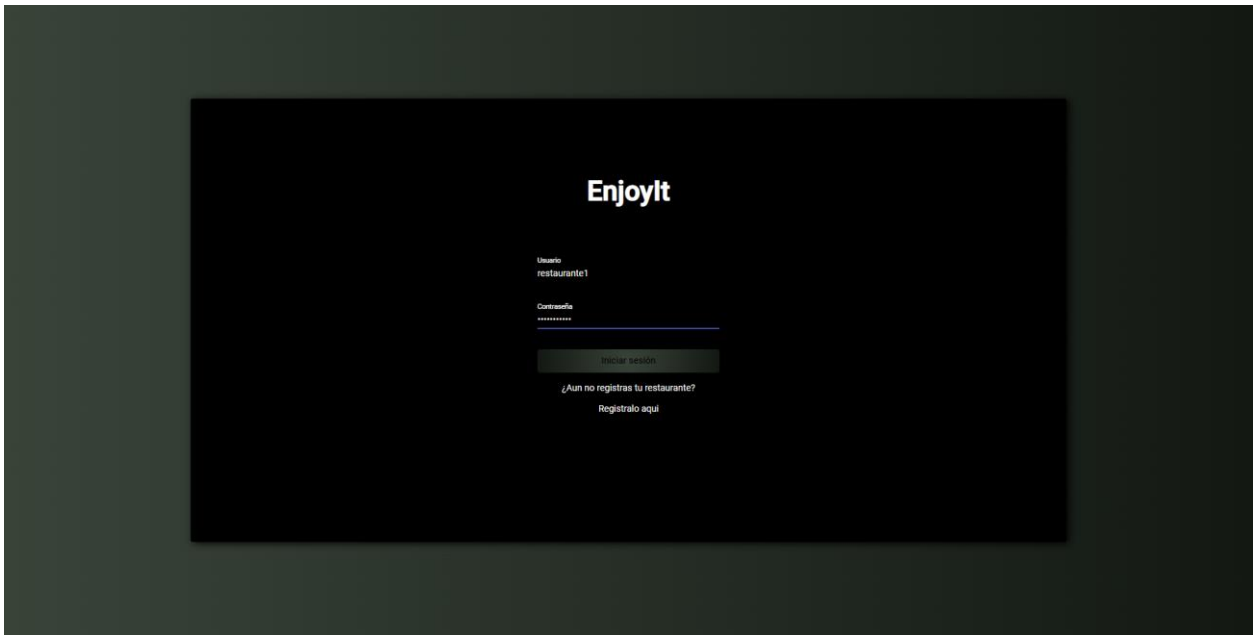
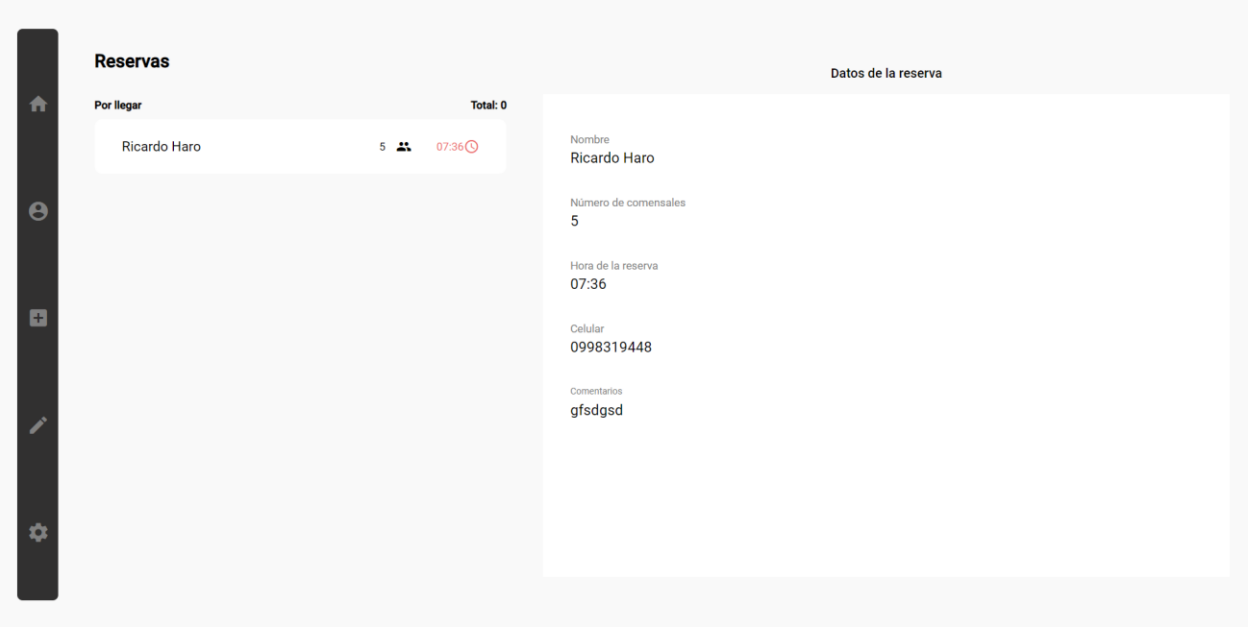
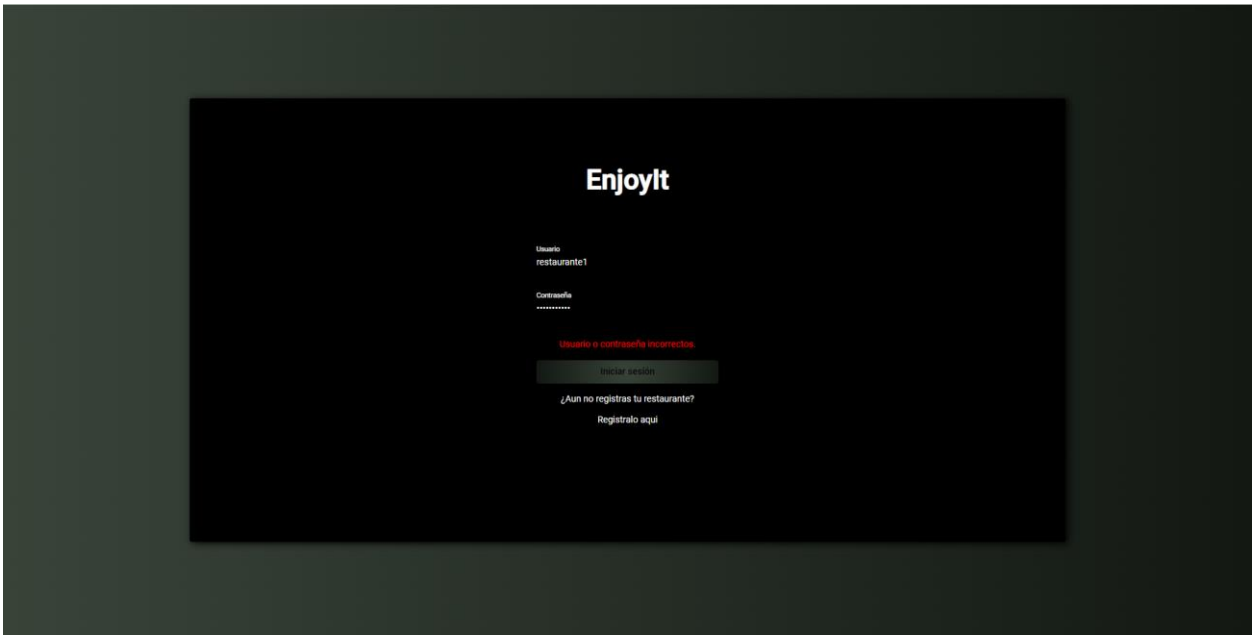


Figura 17 Pantalla home de restaurantes



- Si el usuario y la contraseña son incorrectos se muestra mensaje de error.

Figura 18 Error contraseña o usuario incorrectos



5.1.3. Caso de uso: Ver las reservas

Identificador: CU3

Descripción: Ver las reservas que un restaurante tiene para el mismo, día y para el resto de los días según la fecha.

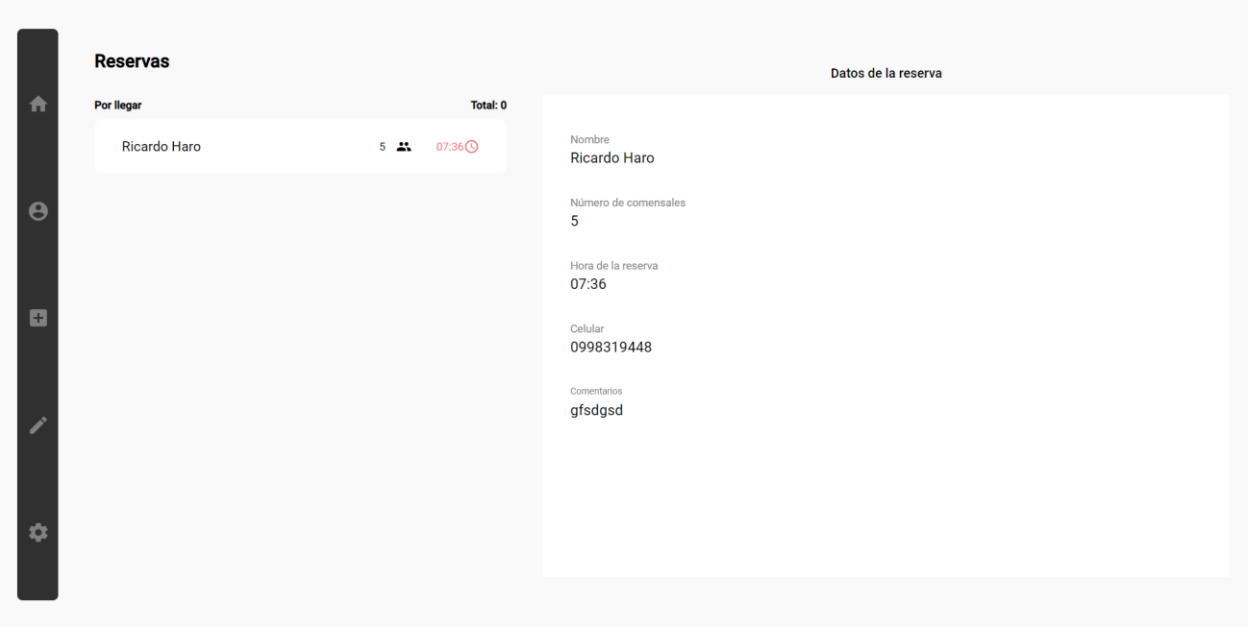
Resultados esperados:

- En la pantalla home se presentarán las reservas para el mismo día en tiempo real, por lo que, si se crea una nueva reserva para ese día, se podrá ver sin necesidad de actualizar la página.
- En la pantalla de reservas se podrá elegir la fecha y cargaran las reservas que se tienen para la fecha indicada.
- En la pantalla de reservas se podrá editar una reserva y se recibirá un mensaje de que se actualizo con éxito.
- En la pantalla de reservas se podrá borrar reservas recibiendo un mensaje de que se borró exitosamente.

Resultados reales:

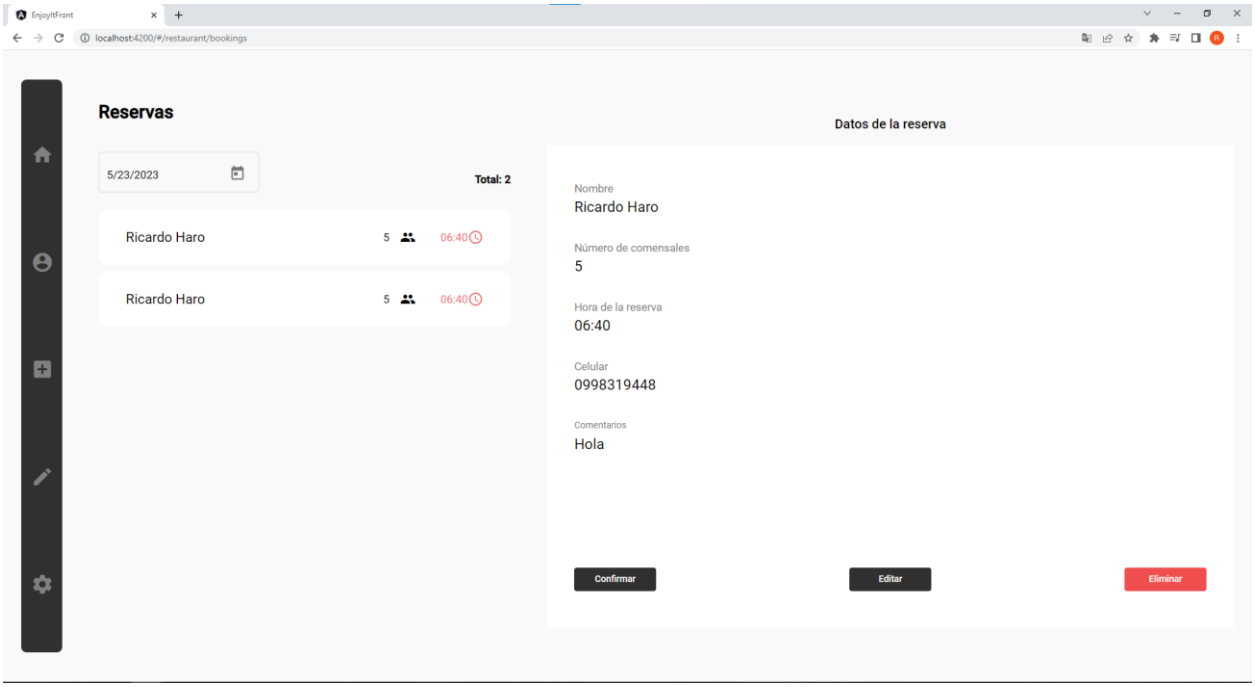
- En la pantalla home se presentan las reservas para el mismo día en tiempo real, si se crea una nueva reserva para ese día, aparece en pantalla sin necesidad de actualizar la página.

Figura 19 Pantalla home gestor de reservas



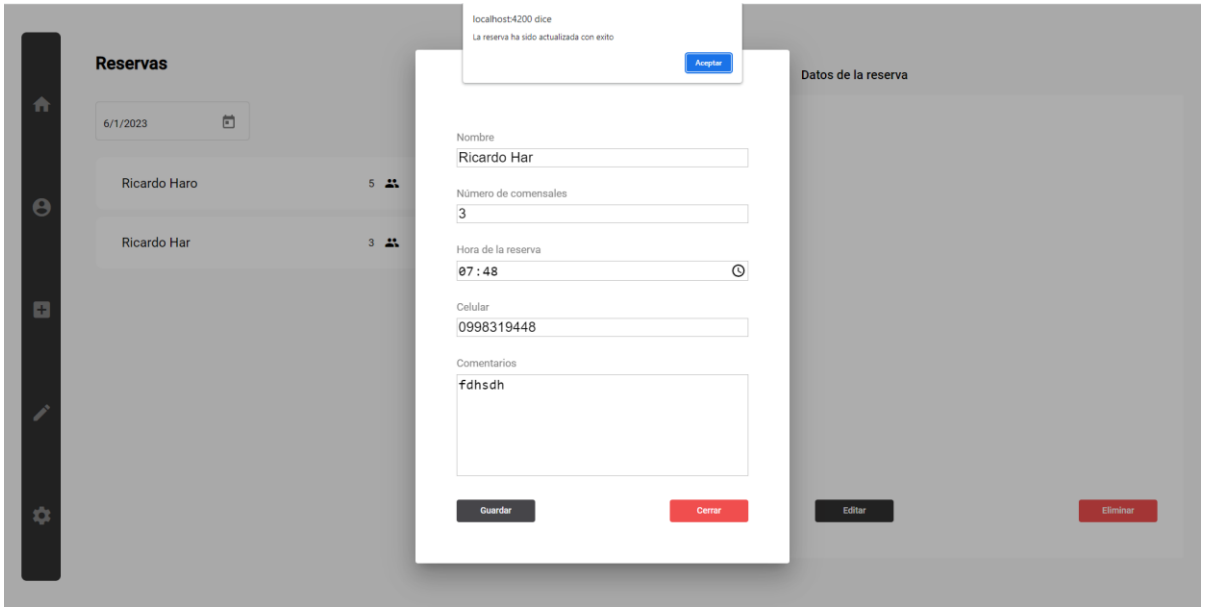
- En la pantalla de reservas se elige la fecha y cargan las reservas que se encuentran para esa fecha.

Figura 20 Pantalla para ver las reservas según la fecha



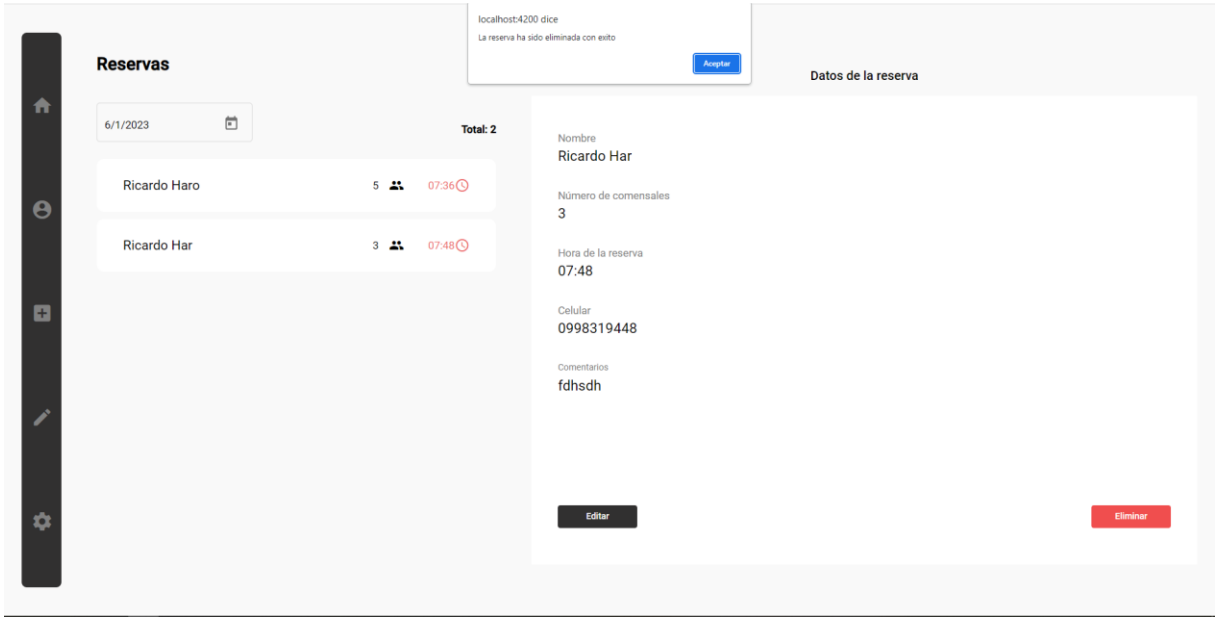
- En la pantalla de reservas se podrá editar una reserva y se recibirá un mensaje de que se actualizo con éxito.

Figura 21 Editar reserva



- En la pantalla de reservas se podrá borrar reservas recibiendo un mensaje de que se borró exitosamente.

Figura 22 Eliminar reserva



5.1.4. Caso de uso: Crear reserva (Restaurante)

Identificador: CU4

Descripción: El restaurante podrá ingresar los datos del comensal y crear una reserva

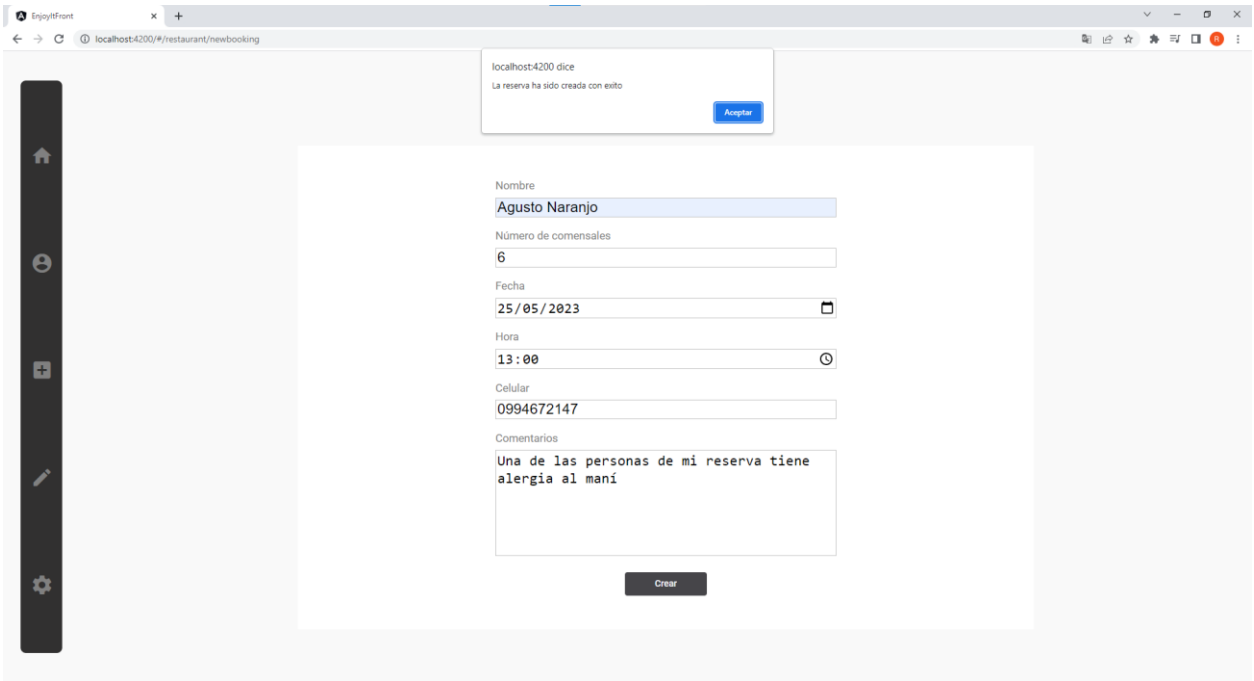
Resultados esperados:

- La reserva se creará y se presentara un mensaje que indique que la reserva ha sido creada
- Si se intenta crear una reserva y ya no hay disponibilidad se presentará un mensaje indicando que no se puede crear.
- Si se encuentra fuera del horario de atención se mostrará un mensaje indicando que no se puede crear la reserva

Resultados reales:

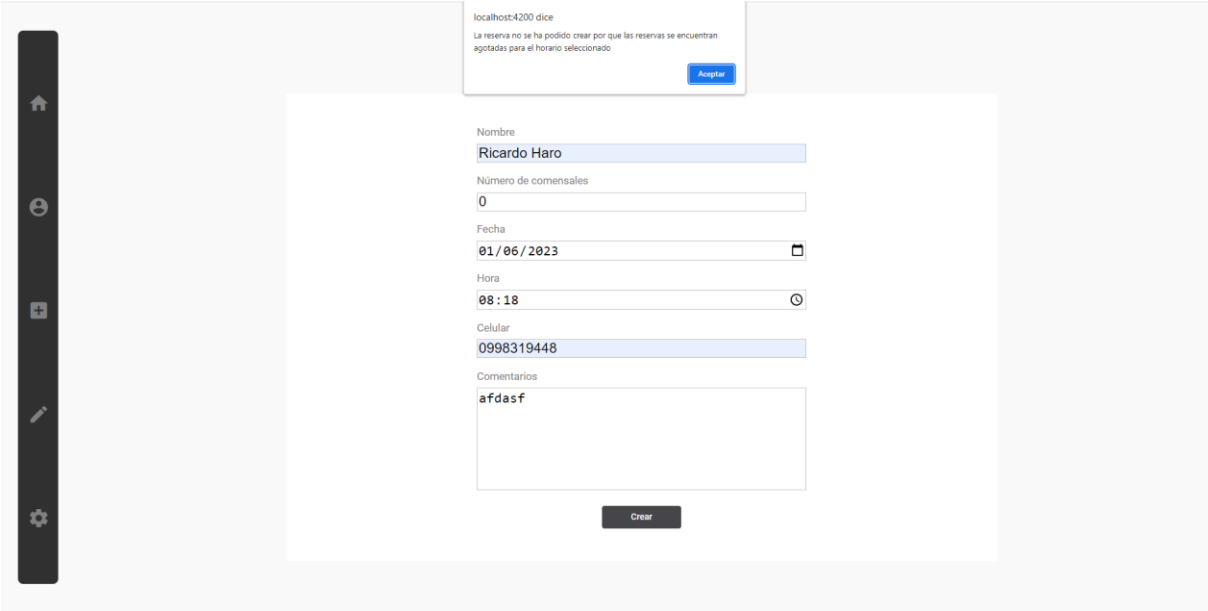
- La reserva se crea y se presenta un mensaje que indica que la reserva fue creada

Figura 23 Reserva creada exitosamente



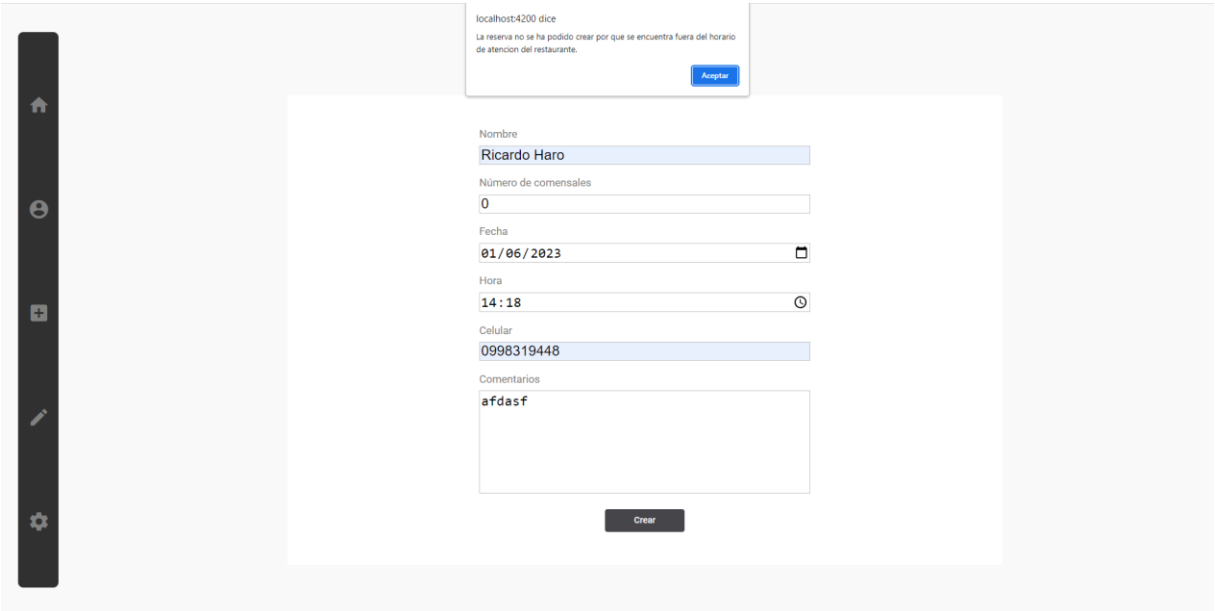
- Si se intenta crear una reserva y ya no hay disponibilidad se presentará un mensaje indicando que no se puede crear.

Figura 24 No hay disponibilidad para el horario elegido



- Si se encuentra fuera del horario de atención se mostrará un mensaje indicando que no se puede crear la reserva

Figura 25 La hora seleccionada esta fuera del horario del restaurante



5.1.5. Caso de uso: Lista de restaurantes

Identificador: CU5

Descripción: Los comensales podrán ingresar a la página y ver los diferentes restaurantes registrados en la aplicación y buscar entre estos restaurantes

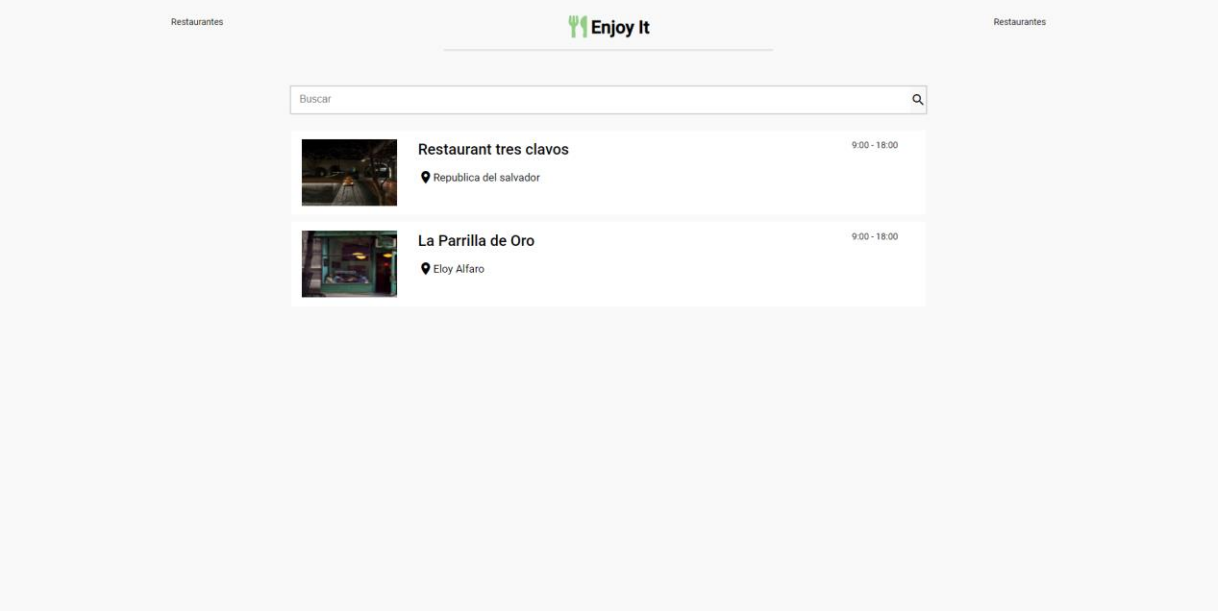
Resultados esperados:

- Se presentan los restaurantes registrados en la aplicación
- Se puede buscar un restaurante por el nombre

Resultados reales:

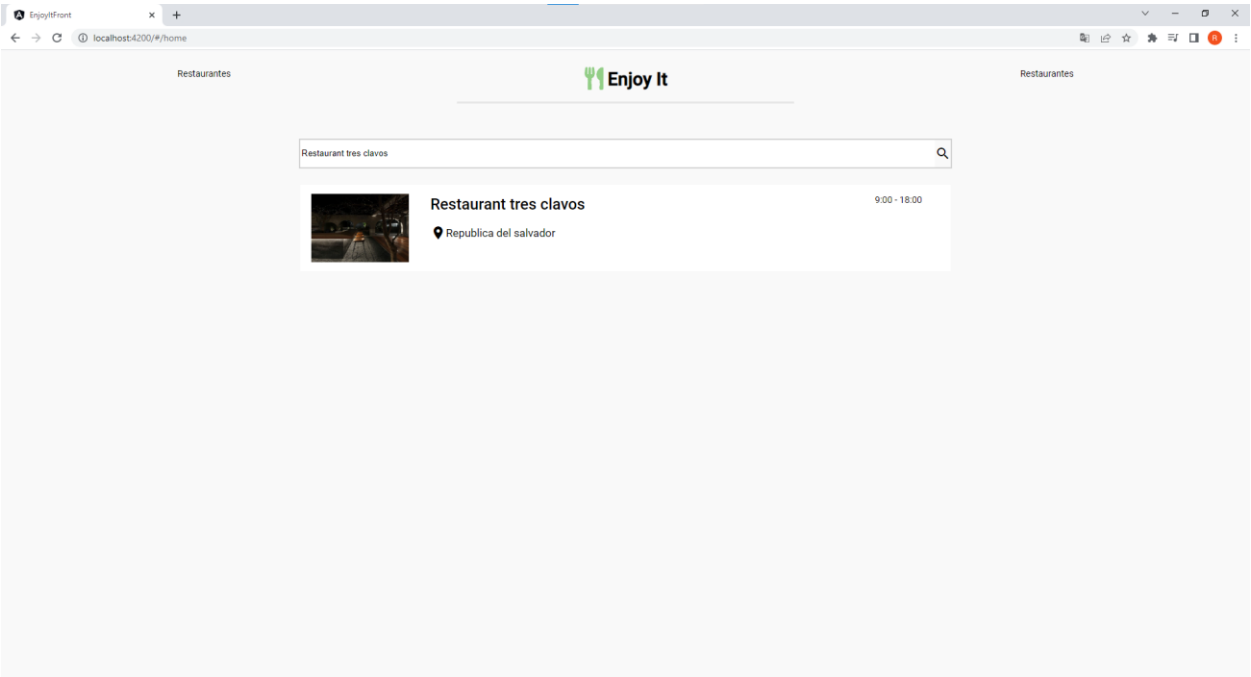
- Se presentan los restaurantes registrados en la aplicación

Figura 26 Pantalla home



- Se puede buscar un restaurante por el nombre

Figura 27 Búsqueda de restaurante



5.1.6. Caso de uso: Crear Reserva (Comensal)

Identificador: CU6

Descripción: Los comensales podrán realizar una reserva en un restaurante en especifico

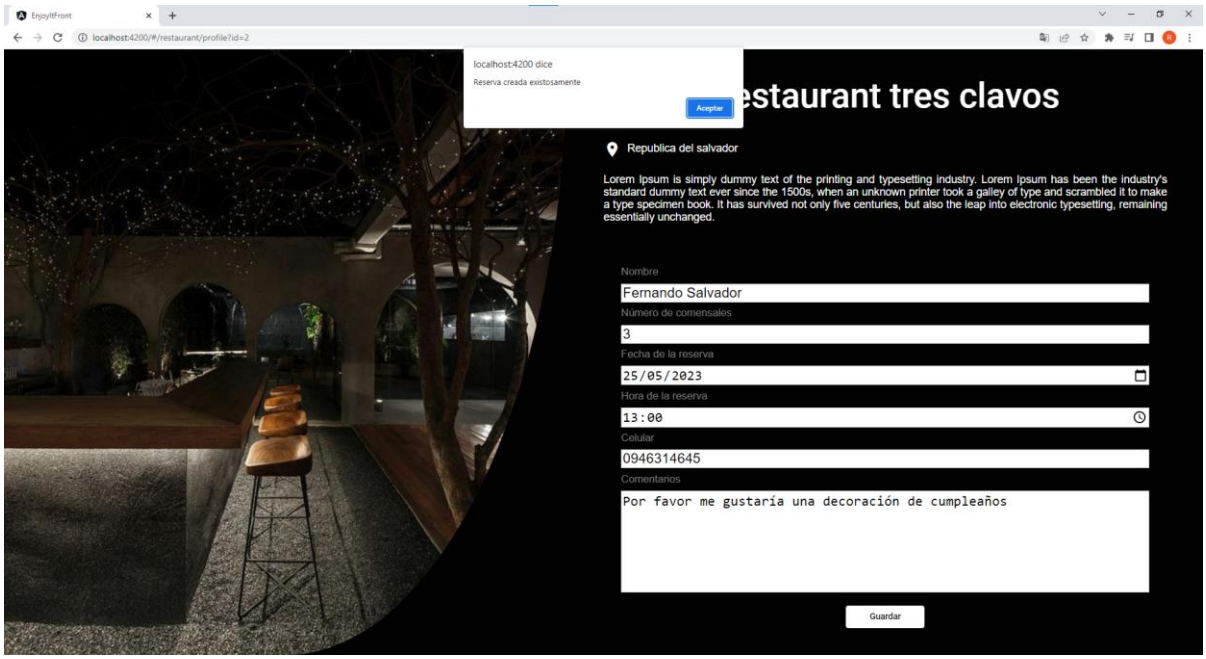
Resultados esperados:

- La reserva se creará y se indicará un mensaje por pantalla, indicando que la reserva fue creada con éxito.
- Si se intenta crear una reserva y ya no hay disponibilidad se presentará un mensaje indicando que no se puede crear.
- Si se encuentra fuera del horario de atención se mostrará un mensaje indicando que no se puede crear la reserva.

Resultados reales:

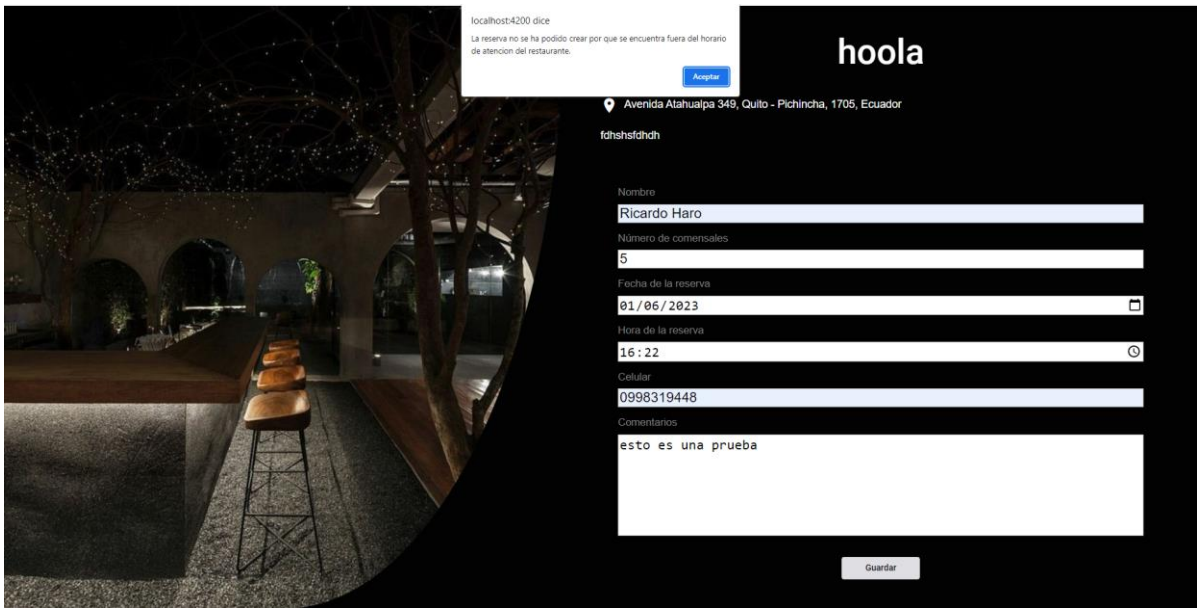
- La reserva se crea y luego se presenta un mensaje en pantalla, indicando que la reserva fue creada con éxito.

Figura 28 Reserva creada con exito por un comensal



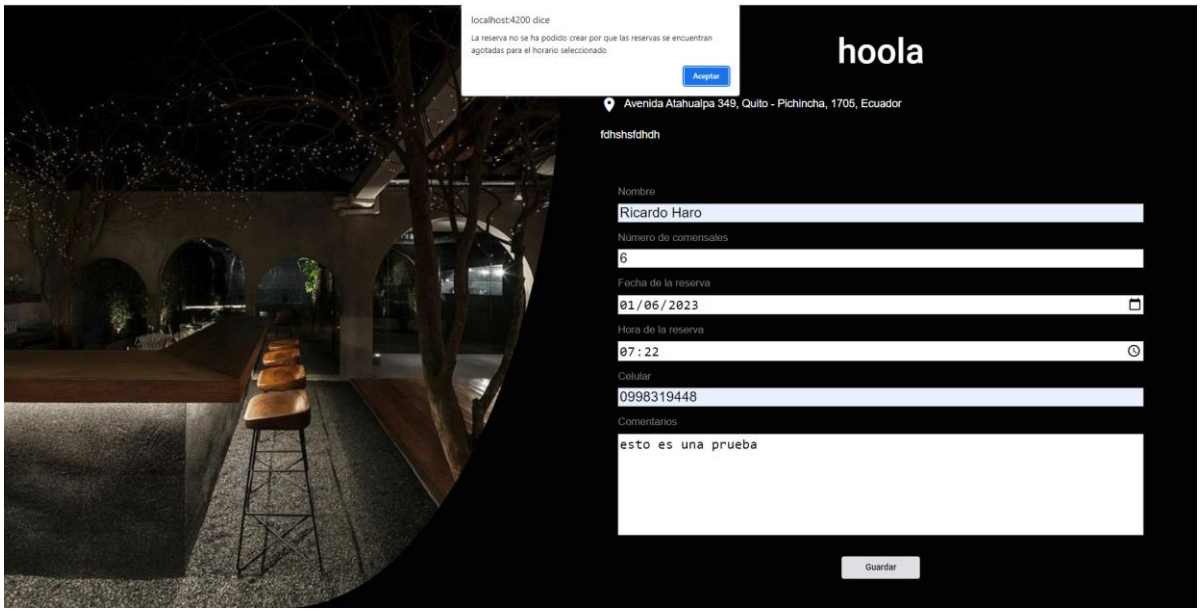
- Si se intenta crear una reserva y ya no hay disponibilidad se presentará un mensaje indicando que no se puede crear.

Figura 29 Error de disponibilidad de reservas



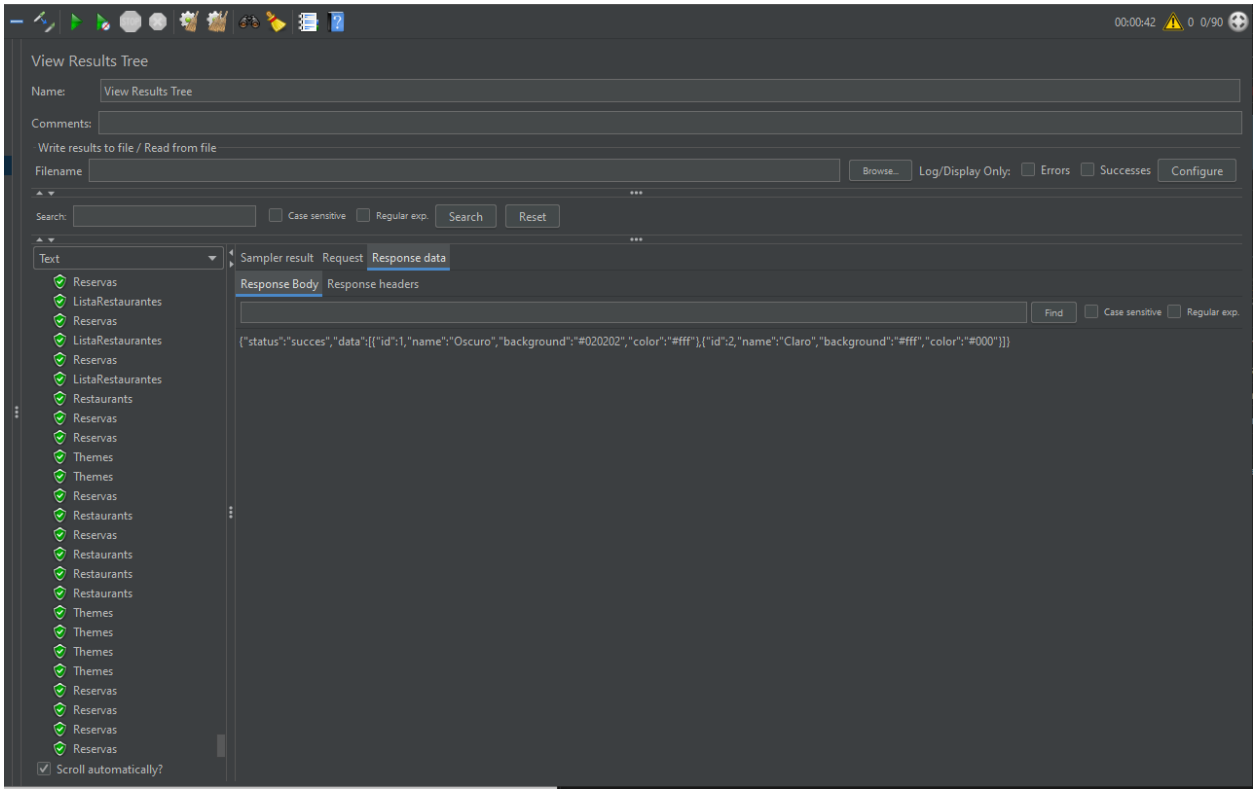
- Si se encuentra fuera del horario de atención se mostrará un mensaje indicando que no se puede crear la reserva

Figura 30 Error hora fuera de horario de atención



Pruebas de rendimiento

Figura 31 Pruebas de rendimiento



CONCLUSIONES Y RECOMENDACIONES

6. Conclusiones:

- El desarrollo de la aplicación web aportó a mis conocimientos no solo técnicos, sino también a la capacidad de investigación sobre la situación actual de un problema que se presenta en la sociedad, y de cómo buscar una solución a este usando herramientas informáticas.
- Durante el desarrollo de la aplicación se presentaron varios problemas. Entre los cuales estuvo la integración de mapas en la aplicación, al no tener conocimientos previos sobre estas herramientas, me llevó a enfrentar conflictos y errores al integrar los estilos de los mapas. Esto me obligo a investigar y buscar soluciones. A través de este proceso, aprendí la importancia de leer la documentación oficial de las herramientas. Este debería ser el primer recurso en ser revisado, ya que es ahí donde se pueden encontrar la mayoría de las respuestas sobre cómo usar las herramientas.
- Durante el proceso de desarrollo se encontró una mejora que se podría implementar en la aplicación. Los restaurantes para poder manejar de mejor manera las reservas que reciben, podrían tener un panel en el que se tengan las mesas del restaurante, y se pueda asignar la reserva a esa mesa. Permitiéndole ver a los restaurantes cual mesa está ocupada por cual reserva.

7. Recomendaciones:

- Se recomienda que los restaurantes actualicen sus datos constantemente, para que los comensales que van a hacer la reserva puedan ver los datos actuales de cada restaurante
- Se recomienda investigar sobre nuevas tecnologías como aplicaciones web progresivas, para permitir a la aplicación implementar nuevas funcionalidades y facilidades para el usuario.

- Se recomienda investigar las necesidades de los restaurantes de los otros sectores de Quito, para poder añadir soluciones a esas necesidades y así tener una aplicación escalable a más restaurantes.

BIBLIOGRAFÍA

(2022, Febrero 28). Obtenido de Angular: https://angular.io/guide/what-is-angular

Amazon Web Services. (s.f.). Obtenido de https://aws.amazon.com/es/what-is/web-application/

Bradygaster. (2022, Septiembre 21). *Microsoft Learn*. Obtenido de https://learn.microsoft.com/es-es/aspnet/signalr/overview/getting-started/introduction-to-signalr

Cordero, Z. R. (2008). *La investigación aplicada: Una forma de conocer las realidades con evidencia científica.*

Cristal, G. (2020). Desarrollo de aplicaciones web usando JavaScript. *(Tesina de Pregrado).* Universidad Politécnica de Sinaloa, Mazatlán.

Drumond, C. (s.f.). *Atlassian.* Obtenido de https://www.atlassian.com/es/agile/project-management

Espasa Saval, J. (2021). Desarrollo de una aplicación web móvil. *(Tesis de Ingeniería).* Universitat Politècnica de València, Valencia.

Hutagikar, V., & Hegde, V. (2020). Analysis of Front-end Frameworks for Web Applications. *International Research Journal of Engineering and Technology (IRJET)*, 4.

Leaflet. (2022, Septiembre 21). Obtenido de https://leafletjs.com/

Macho, G. (2020). Desarrollo de una aplicación segura con .Net Core. *Trabajo Fin de Máster.* Universidad Politécnica de Madrid, Madrid.

Mapbox. (s.f). Obtenido de https://docs.mapbox.com/mapbox-gl-js/guides/

Mazo Jiménez, R. (2022). Plataforma web para la gestión de reservas en restaurantes. *Trabajo Fin de Grado de Ingeniería Informática.* Universidad Politécnica de Madrid, Madrid.

Microsoft. (s.f.). Obtenido de https://support.microsoft.com/es-es/windows/windows-aplicaciones-de-escritorio-y-privacidad-8b3b13bc-d8ff-5460-8423-7d5d5c1f6665

Microsoft. (s.f.). Obtenido de https://dotnet.microsoft.com/es-es/learn/dotnet/what-is-dotnet

Pérez, S., Quispe, J., Mullicundo, F., & Lamas, D. (2021). HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO WEB DESDE. *XXIII Workshop de Investigadores en Ciencias de la Computaci´on*, 4.

React. (s.f). Obtenido de https://es.react.dev/learn/describing-the-ui

redhat. (2022, Mayo 10). Obtenido de https://www.redhat.com/es/topics/automation#importancia-de-la-automatizaci%C3%B3n

Romo Rivero, A. (2020). APLICACIÓN JAILEDTABLE SISTEMA DE CONTROL DE RESERVAS EN RESTAURANTES CON POLÍTICA “ANTI NO SHOW”. *Trabajo Fin de Grado.* UNIVERSIDAD DE JAÉN, Linares.

Shah, K. (. (2014, Julio 24). *Eater.* Obtenido de https://www.eater.com/2014/7/24/6181723/the-origins-of-restaurant-reservations

Thomas , P., Delia, L., Corbalán, L., Cáseres, G., Sosa, J., Tesone, F., . . . Pesado, P. (2020). Enfoques y Tendencias en el Desarrollo de Aplicaciones para Dispositivos. *Centro Asociado a la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires (CIC)*, 7.

Twilio. (s.f.). Obtenido de https://www.twilio.com/docs/glossary/what-is-an-application

Vera Vélez, L. (s.f.). *LA INVESTIGACION CUALITATIVA.*

Villao, W. (2021). Desarrollo de Aplicación web y móvil para gestión de reserva de eventos y mesas del Restaurante Diana y Andrés Banquetes. *Trabajo de titulación.* UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA, LA LIBERTAD.

Yanez Rea, E. (2022). APLICATIVO WEB PARA LA GESTIÓN DE LAS RESERVAS DE UN RESTAURANTE. *Trabajo de Titulación.* Pontificia Universidad Católica del Ecuador, Quito.

ANEXOS

Anexo A:

Pontificia Universidad Católica del Ecuador

Quito, 23 de febrero de 2023

Acta de reunión informativa.

Reunión informativa sobre el proceso de reservas en ciertos restaurantes de la ciudad de Quito.

La presente acta muestra el desarrollo de la reunión llevada a cabo el 23 de febrero, la cual tuvo como orden del día:

1. Contextualización del objetivo del trabajo de integración curricular.
2. Preguntas acerca del manejo actual del proceso.
3. Otros.

Así se obtuvieron datos necesarios para describir la situación actual del proceso de reservas en ciertos restaurantes de la ciudad de Quito.

Shirley Grimaldo
Administradora **Pulcinella Brunch & Caffè**

Ricardo Haro
Estudiante de Ingeniería en sistemas de información