



ESCUELA POLITÉCNICA NACIONAL

INGENIERÍA DE SOFTWARE

2024-A

PROYECTO DE I-BIM

APLICACIONES MÓVILES

TEMA: Gestión de Citas y Servicios Automotrices

INTEGRANTES: Guingla Joel y Delgado Joel

Especificaciones de la Aplicación

Para el desarrollo de la aplicación de gestión de citas y servicios en un taller automotriz, se empleará la metodología Scrum.

- Escala de estimado, Reuniones de diseño y Documentación

Escala del estimado con **Scrum Poker** (basado en la serie de Fibonacci):

- **1 o 2:** Tareas pequeñas y simples, con poco esfuerzo y tiempo requerido.
- **3 o 5:** Tareas de complejidad y esfuerzo moderados.
- **8 o 13:** Tareas más grandes y complejas que requieren más tiempo y recursos.
- **21 o más:** Tareas muy grandes que pueden necesitar ser divididas en subtareas más pequeñas debido a su complejidad y el gran esfuerzo requerido.

Para las reuniones de diseño, habrá:

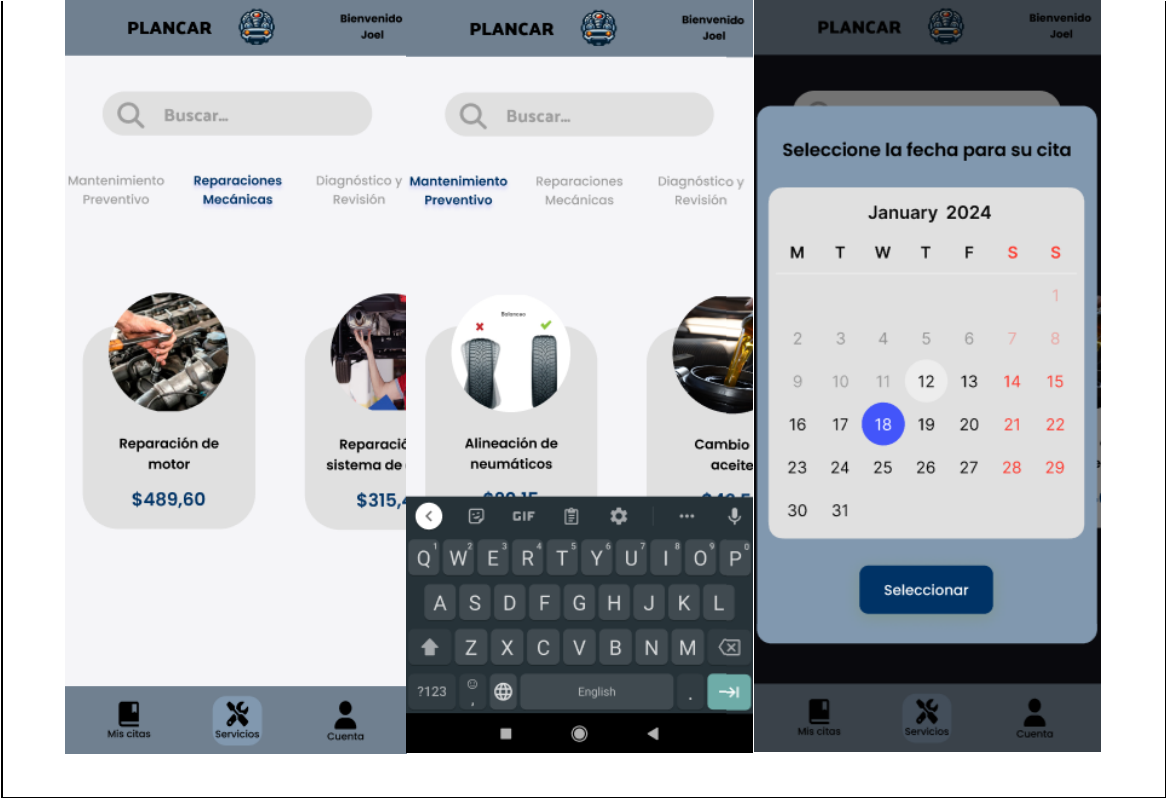
- Discusión sobre el flujo del usuario y la interfaz.
- Validación de los requisitos del usuario cumplidos con la aplicación.
- Discusión sobre manejo de excepciones.

Para la documentación, se considerará:

- Comentarios en el código para explicar cambios y decisiones de diseño.
- Nombres de variables en inglés, estructurados de manera clara y descriptiva.
- Todos los nombres de clases y métodos seguirán el formato camelCase.

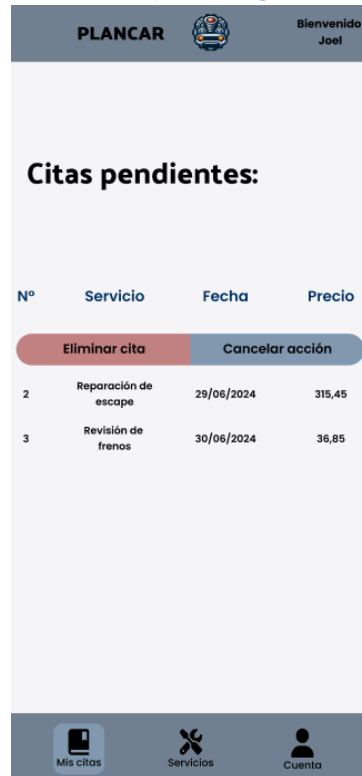
- Product Backlog

Historia de Usuario 1
Título: Agregar nueva cita
ID: US001
Yo, como cliente, Quiero poder agendar una cita, Para recibir un servicio específico en una fecha determinada.
Criterios de aceptación: <ul style="list-style-type: none">- El sistema debe permitir seleccionar un tipo de servicio y ver su precio antes de agendar.- Debe permitir seleccionar la fecha de la cita.
Estimado: 8 (Scrum Poker)
Prototipo en Figma:



Historia de Usuario 2
Título: Cancelar cita
ID: US002
Yo, como cliente, Quiero poder cancelar una cita programa, Para evitar cargos o planificar según cambios en mi disponibilidad.
Criterios de aceptación: - Debe permitir cancelar la cita a través de la interfaz de usuario.
Estimado: 2 (Scrum Poker)

Prototipo en Figma:



Historia de Usuario 3

Título: Ver detalles de citas

ID: US003

Yo, como cliente, Quiero ver los detalles de mis citas programadas, Para estar informado sobre la fecha, el servicio reservado y el precio total.

Criterios de aceptación:

- Debe mostrar claramente la fecha de la cita, el tipo de servicio seleccionado y el precio total.
- Debe permitir acceder a esta información desde la interfaz principal de la aplicación.

Estimado: 3 (Scrum Poker)



- **Definition of Done (DoD)**

El DoD incluirá los siguientes criterios:

- **Revisión de Código:** Cada pieza de código será revisada por al menos un desarrollador para asegurar la calidad y la adhesión a las normas de codificación.
- **Aprobación del Stakeholder:** Los stakeholders clave harán la revisión y aprobación de la funcionalidad completa del aplicativo.
- **Funcionalidad Verificada:** Se verificará que la funcionalidad cumple con los criterios de aceptación especificados en las historias de usuario.

- **Definition of Ready (DoR)**

Incluirá los siguientes puntos:

- **Criterios de Aceptación Claros:** Cada historia de usuario tiene criterios de aceptación bien definidos y entendidos por todos los miembros del equipo.
- **Dependencias Resueltas:** Todas las dependencias externas o bloqueadores han sido resueltos antes de que la historia sea incluida en el sprint.
- **Viabilidad Técnica:** La viabilidad técnica de la historia ha sido evaluada y confirmada.

Diseño de la Base de Datos

- **Estructura de la Base de Datos**

La implementación se realizará utilizando PostgreSQL. La base de datos está simplificada para enfocarse en la interacción entre clientes y las citas que agendan. Este consta de dos tablas principales:

- **Cliente:** Almacena información sobre los clientes del servicio, incluyendo su nombre, teléfono y correo electrónico.
- **Cita:** Contiene todos los detalles relacionados con las citas programadas, incluyendo la relación con clientes, la fecha de la cita, el tipo de servicio y su precio.

- Modelo E-R y Cardinalidades

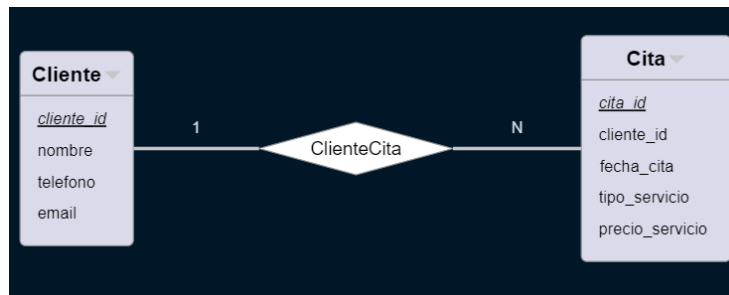


Figura 1. Modelo E-R del aplicativo

• Clientes y Citas

Cardinalidad: Uno a Muchos

Descripción: Un cliente puede tener múltiples citas, pero cada cita está asociada a un único cliente. Esto se refleja en la relación donde la clave primaria de clientes (cliente_id) es una clave foránea en citas.

- Implementación en SQL

```
-- Creación de la tabla cliente
CREATE TABLE IF NOT EXISTS cliente (
  cliente_id SERIAL PRIMARY KEY,
  nombre VARCHAR(255),
  telefono VARCHAR(50),
  email VARCHAR(100)
);

-- Creación de la tabla cita
CREATE TABLE IF NOT EXISTS cita (
  cita_id SERIAL PRIMARY KEY,
  cliente_id INT,
  fecha_cita DATE,
  tipo_servicio VARCHAR(255),
  precio_servicio DECIMAL(10, 2),
  FOREIGN KEY (cliente_id) REFERENCES cliente(cliente_id)
);
```

A continuación, se presentará la inserción de datos en PostgreSQL de cada una de las tablas anteriores:

	cliente_id [PK] integer	nombre character varying (255)	telefono character varying (50)	email character varying (100)
1	1	Juan Martínez	0998765432	juan.martinez@example.com
2	2	Ana López	0996543218	ana.lopez@example.net
3	3	Carlos Pérez	0981234567	carlos.perez@example.org

Figura 2. Tabla de clientes

	cita_id [PK] integer	cliente_id integer	fecha_cita date	tipo_servicio character varying (255)	precio_servicio numeric (10,2)
1	1	1	2024-07-15	Cambio de aceite	45.00
2	2	2	2024-07-18	Revisión general	75.50
3	3	3	2024-07-20	Alineación y balanceo	60.00

Figura 3. Tabla de citas

Anexos

- 1. Enlace al prototipo completo en Figma:
<https://www.figma.com/proto/TbkzbdIGX4kcW8CDFT3ljZ/PLANCAR?node-id=7-3&t=1pPy2Klp8M4JT9GD-1&scaling=scale-down&content-scaling=fixed&page-id=0%3A1&starting-point-node-id=7%3A3>