

Universidad Mariano Gálvez de Guatemala

Ingeniería en sistemas de información y ciencias de la computación

Ing. Mélvín Cali.



Erick Eduardo Camposeco Guerra 7690-14-14643

Angel Enrique Ibañez Linares 7690-22-1911

Carlos Emanuel Del Cid Corado 7690-22-19541

FECHA: 09 /06/2025

EcoRide

Cliente: Municipalidad de Guatemala (Sector de transporte sostenible).

Proveedor: TechSolutions (Estudiantes universitarios de ingeniera en sistemas).

Identificación del Contrato: *CON-2025-006* (Acuerdo Marco de Desarrollo Ágil).

Fechas clave

Fecha de Inicio: 21 de marzo de 2025

Fecha de Entrega Acordada: 9 de junio 2025.

Fecha Efectiva de Entrega: 9 de junio 2025

Responsables Principales

Rol	Nombre	Contacto
Project Manager	Erick Camposeco	ecamposecog@miumg.edu.gt
Líder Técnico	Ángel Linares	@miumg.edu.gt
Contacto Cliente	Carlos Del Cid	@miumg.edu.gt

Versión del producto

Versión Estable: v1.2.0 (incluye módulos de reserva y login).

Estándar de Versionado: Semántico (SemVer): v1.2.0 = 1 (mayor, cambios importantes), 2 (menor, nuevas funcionalidades), 0 (patch, correcciones).

Nota: La versión v1.0.0 se entregó en fase beta (11 de abril 2025).

Descripción General del Sistema

Objetivo General del Proyecto

El sistema de préstamos de bicicletas busca proporcionar una solución eficiente, automática y gratuita para la gestión de las bicicletas que tiene múltiples terminales que están distribuidas estratégicamente. El propósito principal es permitir a los usuarios poder registrar reservas, utilizar y devolver las bicicletas de forma segura y conveniente al mismo tiempo se ofrece a los técnicos y administradores una serie de herramientas para el monitoreo, mantenimiento y redistribución de las diferentes bicicletas.

Funcionalidades

- Registro e inicio de sesión de usuarios
- Reserva de bicicletas por terminal de origen y destino
- Control del tiempo de uso y penalizaciones por incumplimientos
- Seguimiento de la ubicación mediante GPS
- Reporte de problemas técnicos o físicos
- Monitoreo del estado de cada bicicleta
- Administración centralizada de usuarios, reportes y recursos

Módulos

Usuario

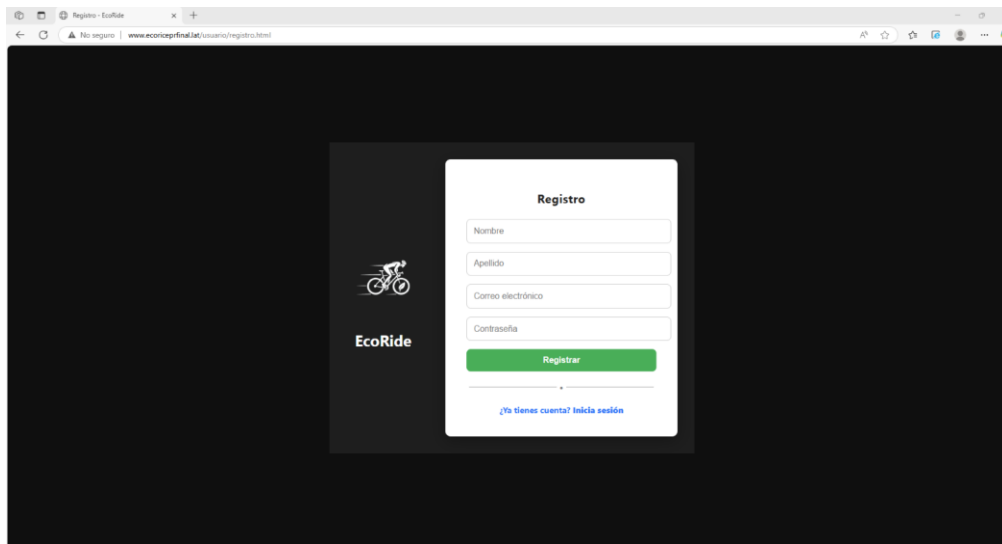
- Inicio de sesión: permite a los usuarios ingresar sus credenciales al sistema.
- Historial: muestra el historial de reservas realizadas por el usuario.
- Devolver: permite al usuario marcar una bicicleta como devuelta.
- Reservar: permite al usuario reservar una bicicleta

Administrador

- Bicicletas: permite al administrador agregar, editar o eliminar bicicletas.
- Usuarios: permite ver y editar usuarios.
- Reportes: permite ver, editar y eliminar los reportes creados por los usuarios.

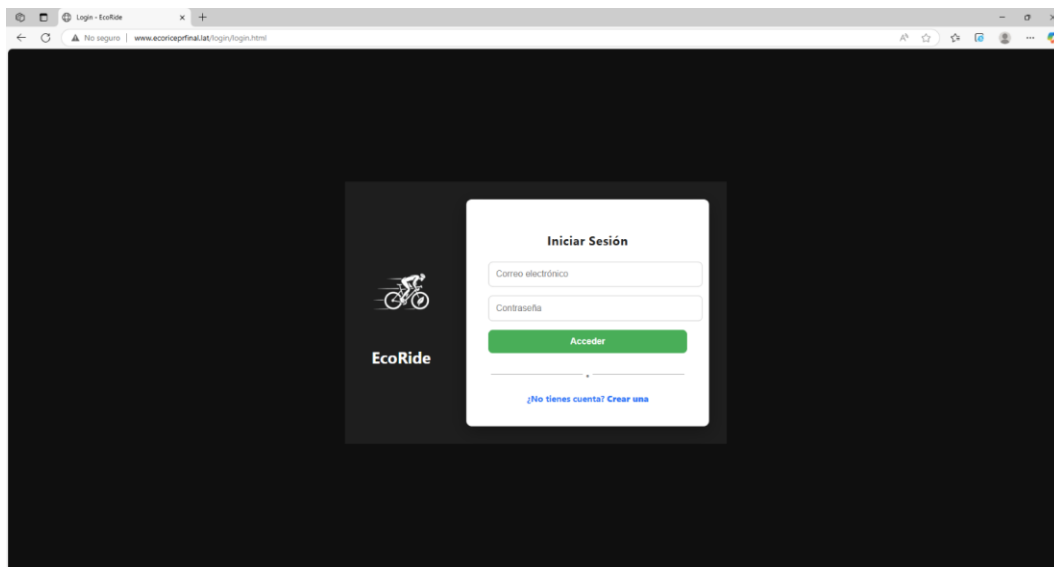
Pantallas

Registro de Usuario



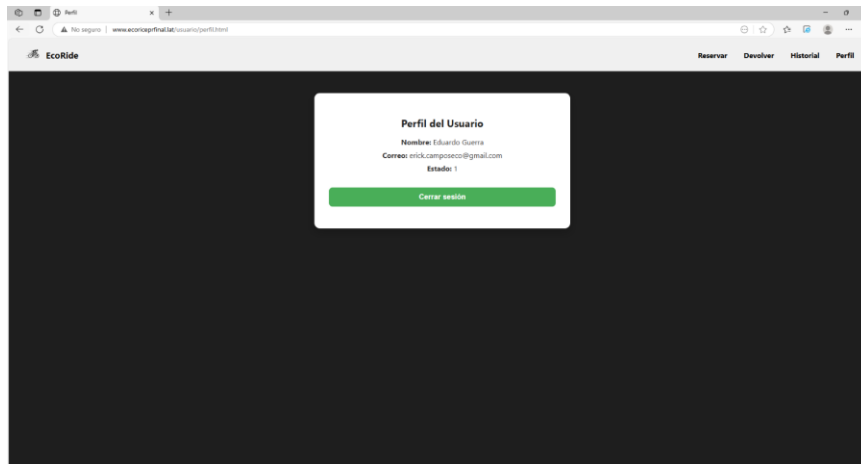
The screenshot shows a web browser window with the title 'Registro - EcoRide'. The address bar shows 'www.ecorideprfinal.lat/usuario/registro.html'. The page has a dark background. On the left, there is a logo for 'EcoRide' featuring a stylized bicycle. On the right, there is a white registration form titled 'Registro'. The form contains four input fields: 'Nombre', 'Apellido', 'Correo electrónico', and 'Contraseña'. Below these fields is a green button labeled 'Registrar'. At the bottom of the form, there is a link that says '¿No tienes cuenta? Inicia sesión'.

Inicio de sesión

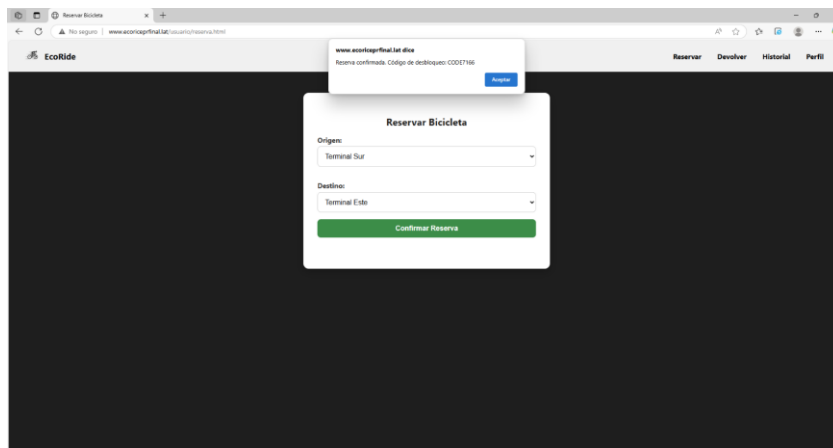
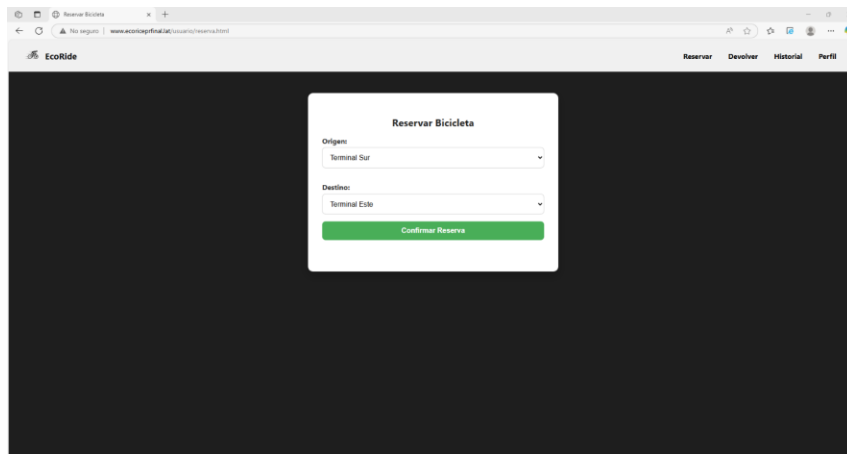


The screenshot shows a web browser window with the title 'Login - EcoRide'. The address bar shows 'www.ecorideprfinal.lat/login/login.html'. The page has a dark background. On the left, there is a logo for 'EcoRide' featuring a stylized bicycle. On the right, there is a white login form titled 'Iniciar Sesión'. The form contains two input fields: 'Correo electrónico' and 'Contraseña'. Below these fields is a green button labeled 'Acceder'. At the bottom of the form, there is a link that says '¿No tienes cuenta? Crear una'.

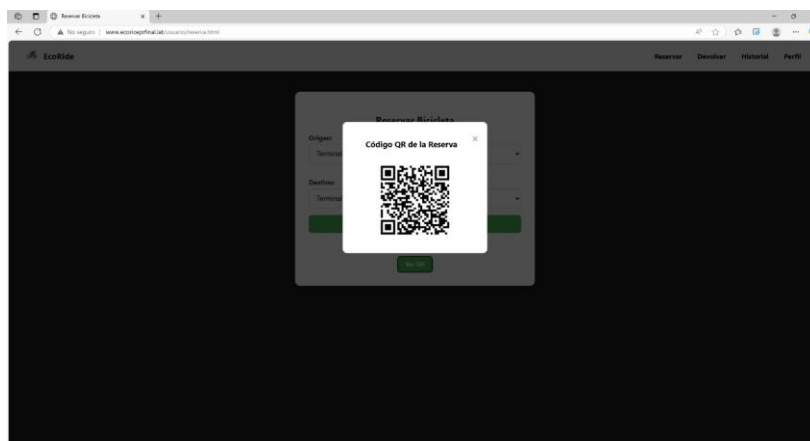
Perfil de usuario



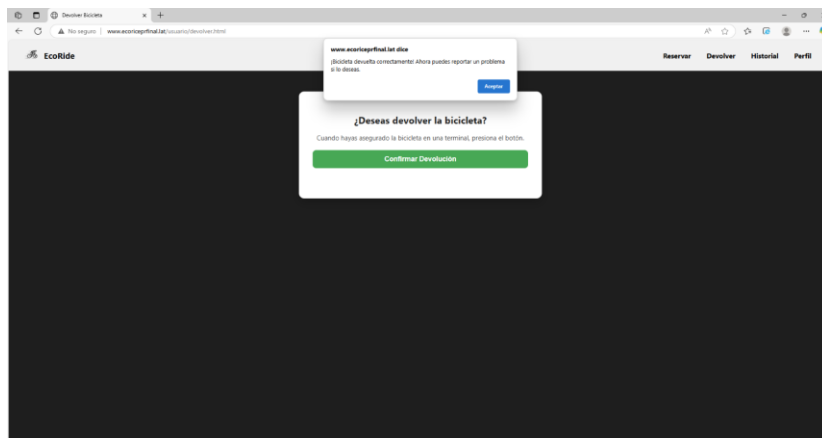
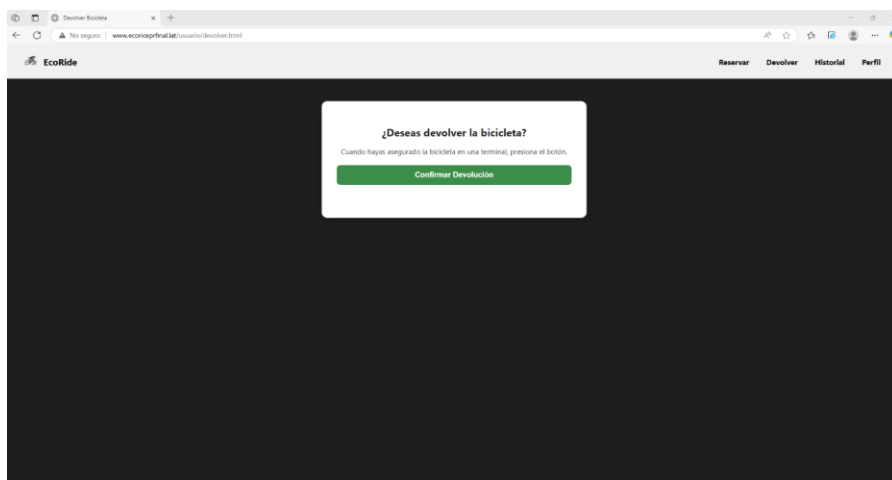
Reservar Bicicleta



Ver QR



Devolver bicicleta

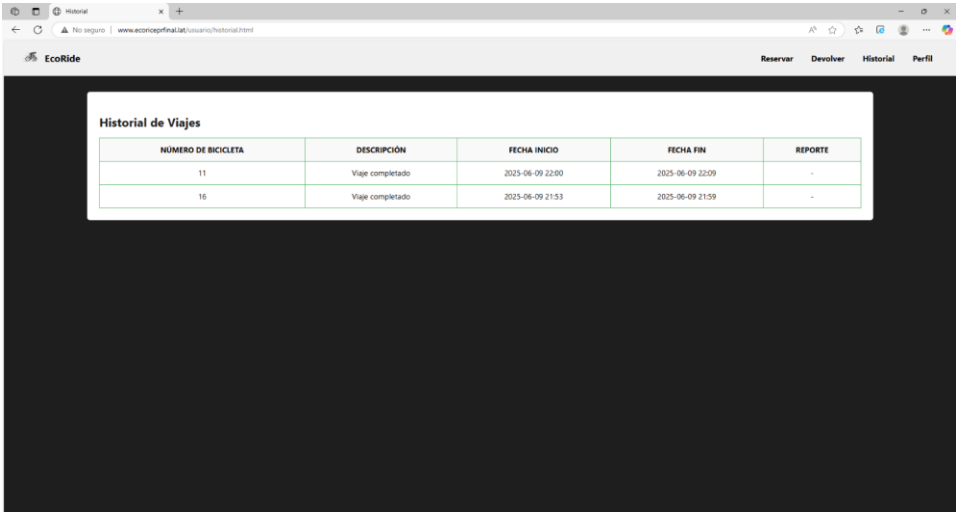


Reportar Problema

The screenshot shows a web browser window with the address bar displaying "www.ecoriceprfinal.lat/usuario/reporta.html". The page header includes the "EcoRide" logo and navigation links: "Reservar", "Devolver", "Historial", and "Perfil". The main content area features a white modal box with the title "Se ha regresado la bicicleta con éxito" and the subtitle "Reporte de problema". Inside the modal, there is a text input field with the placeholder "Describe el problema...". Below the input field are two green buttons: "Enviar" and "Omitir".

This screenshot shows the same "Reportar Problema" form, but with a success message. The text input field now contains "todo ok". Above the modal box, a toast notification from "www.ecoriceprfinal.lat" appears, stating "(Reporte enviado exitosamente)" with an "Aceptar" button. The "Enviar" and "Omitir" buttons remain visible at the bottom of the modal.

Historial



Gestión de bicicletas



14	Modelo C4	activo	Terminal Sur	Editar	Eliminar
15	Modelo C5	activo	Terminal Sur	Editar	Eliminar
16	Modelo D1	activo	Terminal Este	Editar	Eliminar
17	Modelo D2	activo	Terminal Este	Editar	Eliminar
18	Modelo D3	activo	Terminal Este	Editar	Eliminar
19	Modelo D4	activo	Terminal Este	Editar	Eliminar
20	Modelo D5	activo	Terminal Este	Editar	Eliminar
21	Modelo E1	activo	Terminal Oeste	Editar	Eliminar
22	Modelo E2	activo	Terminal Oeste	Editar	Eliminar
23	Modelo E3	activo	Terminal Oeste	Editar	Eliminar
24	Modelo E4	activo	Terminal Oeste	Editar	Eliminar
25	Modelo E5	activo	Terminal Oeste	Editar	Eliminar

Crear nuevo ítem

Modelo

Ubicación

Guardar

Cancelar

Gestión de Usuarios				
ID	Nombre / Correo	Estado de cuenta	Rol	Acciones
6	I a / Ia@gmail.com	1	Usuario	<button>Editar</button>
7	Edras Tatuaca / EdrasT@gmail.com	1	Usuario	<button>Editar</button>
8	Io a / Ioa@gmail.com	1	Usuario	<button>Editar</button>
16	Emely Leon / EL@gmail.com	1	Usuario	<button>Editar</button>
17	Eduardo Guerra / erick.camposeco@gmail.com	1	Usuario	<button>Editar</button>
1	Ana L♦♦pez / ana@example.com	activo	Usuario	<button>Editar</button>
2	Luis G♦♦mez / luis@example.com	activo	Usuario	<button>Editar</button>
3	Angel Linares / angellinares@gmail.com	activo	Administrador	<button>Editar</button>
4	Erick Lorenzo / LEZ@gmail.com	inactivo	Usuario	<button>Editar</button>
5	Melvin Cali / Melvin@ejemplo.com	activo	Administrador	<button>Editar</button>
15	Edras Tatuaca / edras@gmail.com	1	Usuario	<button>Editar</button>
9	Lucas Marcos / LucasM@gmail.com	1	Usuario	<button>Editar</button>

Gestión de reportes

 EcoRide

BicicletasUsuariosReportesSalir

ID	Usuario	Fecha	Descripción	Estado	Acciones
30	Eduardo Guerra	2025-06-09 22:00	todo ok	pendiente	Editar Eliminar
31	Eduardo Guerra	2025-06-09 22:10	todo ok	pendiente	Editar Eliminar
27	Ana López	2025-06-09 05:23	nada	pendiente	Editar Eliminar
29	Edras Tatuaca	2025-06-09 19:46	La bicicleta estaba manchada de popo unu	pendiente	Editar Eliminar
28	Edras Tatuaca	2025-06-09 19:28	Le faltaban las llantas	pendiente	Editar Eliminar
1	W Q	2025-06-06 16:32	Nada	pendiente	Editar Eliminar

Estado de Funcionalidad

- Registro e inicio de sesión de usuarios: completa
- Reserva de bicicletas por terminal de origen y destino: completa
- Control del tiempo de uso y penalizaciones por incumplimientos: descartada
- Seguimiento de la ubicación mediante GPS: descartada
- Reporte de problemas técnicos o físicos: completa
- Monitoreo del estado de cada bicicleta: descartada
- Administración centralizada de usuarios, reportes y recursos: completa

Reglas de Negocio

Autenticación y Roles

1. Solo los usuarios registrados pueden iniciar sesión.
2. Existen dos roles con permisos diferentes:
 - Usuario: puede reservar, devolver y ver su historial.
 - Administrador: puede gestionar bicicletas, usuarios y reportes.

Reservación de Bicicletas

3. Solo se puede reservar una bicicleta si está disponible.
4. Un usuario solo puede tener una bicicleta activa en uso a la vez.
5. Al reservar una bicicleta, se registra la fecha y hora de inicio.

Devolución de Bicicletas

6. Una bicicleta reservada debe ser devuelta antes de poder reservar otra.
7. La devolución actualiza el estado de la bicicleta a “disponible”.
8. Se registra la fecha y hora de devolución para calcular el tiempo de uso.

Gestión de Usuarios

9. Solo los administradores pueden crear, editar o eliminar usuarios.
10. No se pueden registrar usuarios con el mismo correo electrónico o nombre de usuario.

Gestión de Bicicletas

11. Solo los administradores pueden agregar, modificar o eliminar bicicletas.
12. Las bicicletas tienen un estado que puede ser: activo, inactivo o reservada.

Reportes

13. Los administradores pueden ver reportes del uso de bicicletas.
14. Todo registro de uso queda guardado en un historial

Requisitos de Rendimiento

- Tiempo de respuesta menor a 3 segundos en operaciones críticas (reservar, devolver, consultar historial).
- La carga de páginas debe mantenerse por debajo de los 3 segundos incluso con múltiples usuarios activos.

Requisitos de Escalabilidad

- El sistema debe permitir el aumento del número de bicicletas y usuarios sin necesidad de rediseñar la base de datos.
- Debe ser posible agregar nuevas estaciones o ubicaciones para reservar bicicletas on cambios mínimos.

Requisitos de Seguridad

- Autenticación obligatoria para acceder a funcionalidades como reservar o devolver bicicletas.
- Implementación de roles y permisos diferenciados

Requisitos de Usabilidad

- Interfaz intuitiva con botones claramente identificados.
- Textos y botones en español, con instrucciones clara

Repositorio de Código Fuente

https://github.com/AEIL2203/Proyecto_1.git

Rama: máster

Requisitos técnicos cumplidos

Lenguajes de programación utilizados:

Frontend:

- HTML5 (estructura)
- CSS3 (estilos)
- JavaScript (lógica dinámica)

Backend:

- Oracle Database

Frameworks y bibliotecas aplicadas:

Frontend:

- CSS responsivo
- HTML/JS

Backend:

- Node.js

Infraestructura tecnológica:

Servidor Web:

- Nginx
- OracleDB (gestión de usuarios/reservas)

Hosting:

- Servidor VPS/Cloud (IP pública 31.97.134.125)

Contenedores/Servicios:

- No se usó contenedores (ej. Docker) ni servicios cloud específicos (AWS/Azure)

Cumplimiento de estándares:

Codificación:

- HTML/CSS válido (W3C Validator)
- JavaScript en modo estricto (use strict)

Seguridad (OWASP):

- Cifrado: Conexiones HTTPS
- Control de acceso: Autenticación de usuario (pendiente de implementar JWT/OWASP)
- Vulnerabilidades: Riesgo medio en XSS (requiere sanitización de inputs)

Accesibilidad (WCAG):

- Nivel A parcial (mejorable)

Rendimiento:

- Carga optimizada (CSS/JS minificados)

Ambientes de despliegue

Entorno de desarrollo:

Herramientas:

- Editores: VS Code / Sublime Text
- Control de versiones: Git

Configuración:

- Stack local: XAMPP/WAMP

Entorno de pruebas:

- URL: <http://31.97.134.125/test/> (no accesible públicamente)

Configuración:

- Réplica del entorno de producción (datos anonimizados)

Datos:

- Dataset de prueba con reservas simuladas

Entorno de producción:

IP/Servidores:

- IP: 31.97.134.125 (Ubuntu Server 22.04 LTS)

Configuración:

- Apache (v2.4.x) con módulo mod_php
- MySQL (v8.x) en puerto estándar

Esquema de red:

- Firewall básico (iptables), sin balanceadores de carga

Dependencias:

- Servicios externos: No detectados (ej. APIs de pago)

Control de acceso:

- Acceso SSH por clave pública
- Credenciales de DB encriptadas en archivos PHP

Análisis y diseño Refleja

La base conceptual y técnica del sistema desarrollado.

Análisis:

ID	Descripción	Entradas	Salidas	Comportamiento Esperado	Excepciones y Errores
RF01	El sistema debe permitir a los usuarios registrarse e iniciar sesión	Nombre, correo, contraseña	Acceso al sistema, datos del perfil	El usuario se registra o inicia sesión correctamente según su rol	Usuario o contraseña incorrectos
RF02	El usuario podrá reservar una bicicleta desde una terminal de origen	Terminal origen, terminal destino, ID usuario	Confirmación de reserva, código QR	El sistema verifica disponibilidad, genera QR, actualiza estado de bicicleta y terminal	No hay bicicletas disponibles en la terminal seleccionada
RF03	El usuario podrá devolver una bicicleta en una terminal habilitada	ID de usuario (localStorage), QR/código reserva	Confirmación de devolución	Al devolver, se actualiza la reserva, la bicicleta vuelve a estado 'activo' y se suma la disponibilidad	No se encontró bicicleta activa o QR inválido
RF04	El usuario podrá reportar un problema tras devolver la bicicleta	ID bicicleta, ID usuario, descripción	Confirmación de reporte	El sistema almacena el reporte, verifica si la bicicleta debe quedar inactiva según acumulación de reportes	Error al enviar reporte, error de conexión
RF05	El sistema permitirá a los administradores visualizar y	ID usuario, acción a realizar	Estado actualizado del usuario	El administrador podrá cambiar estado entre	Usuario no encontrado o error al actualizar

	suspender usuarios			activo/suspendido	
RF06	El usuario podrá consultar la disponibilidad de bicicletas en terminales	Terminal seleccionada	Lista de bicicletas disponibles por terminal	El sistema muestra en tiempo real cuántas bicicletas están disponibles en la terminal seleccionada	Error de conexión con la base de datos o sin resultados

Requerimientos no funcionales

Categoría	Descripción
Rendimiento	El sistema debe responder en un tiempo máximo de 2 segundos para operaciones básicas (reserva, login y etc...). Debe ser capaz de atender al menos 100 usuarios concurrentes sin degradación perceptible en el rendimiento.
Seguridad	El acceso al sistema estará protegido mediante autenticación con Google a través de Google cloud. Los datos estarán cifrados y no se almacenarán credenciales en el cliente.
Escalabilidad	El sistema debe poder adaptarse al crecimiento de usuarios y terminales sin rediseño. Google cloud permite escalar automáticamente según la demanda.
Usabilidad	La interfaz será intuitiva, accesible y adaptativa. Funcionará correctamente en computadoras, tablets y smartphones.
Mantenibilidad	El código será modular y organizado, facilitando comprensión, depuración y mantenimiento. Se aplicarán buenas prácticas en HTML, CSS y JavaScript.
Compatibilidad	El sistema será compatible con navegadores modernos: Chrome, Firefox, Edge y Safari, y funcionará en Windows, Android e iOS.
Legalidad	El sistema respetará la privacidad del usuario. La información personal será mínima y usada sólo para autenticación.

Casos De Uso

Caso de Uso: Autenticar usuario

Actor: Usuario

Descripción: Permite a un usuario registrarse o iniciar sesión en el sistema.

Precondición: El usuario debe tener una cuenta (si inicia sesión).

Postcondición: El usuario accede a su perfil.

Flujo principal:

- - Usuario ingresa correo y contraseña.
- - El sistema valida las credenciales.

Excepciones:

- - Datos inválidos → mensaje de error.

Caso de Uso: Reserva bicicleta

Actor: Usuario

Descripción: Permite al usuario seleccionar una bicicleta disponible desde una terminal.

Precondición: El usuario debe estar autenticado.

Postcondición: La bicicleta queda reservada con código QR/PIN.

Flujo principal:

- - Selecciona terminal origen.
- - Selecciona bicicleta y confirma.

Excepciones:

- - No hay bicicletas disponibles.

Caso de Uso: Reporte de problema

Actor: Usuario

Descripción: Permite enviar reportes de fallas o incidencias.

Precondición: Usuario autenticado.

Postcondición: Problema registrado.

Flujo principal:

- - Elige opción de reporte.
- - Llena formulario y lo envía.

Excepciones:

- - Fallo al enviar el formulario.

Caso de Uso: Consulta de historial

Actor: Usuario

Descripción: Permite al usuario ver su historial de préstamos.

Precondición: Usuario autenticado.

Postcondición: Historial mostrado.

Flujo principal:

- - Selecciona opción 'Ver historial'.
- - Sistema muestra lista con fechas y terminales.

Excepciones:

- - No hay registros.

Caso de Uso: Gestión usuarios

Actor: Administrador

Descripción: Permite activar, suspender o verificar cuentas.

Precondición: Administrador autenticado.

Postcondición: Usuario actualizado.

Flujo principal:

- - Selecciona opción de gestión.
- - Realiza acción sobre usuario.

Excepciones:

- - Usuario no existe.

Caso de Uso: Gestión bicicletas

Actor: Administrador

Descripción: Permite añadir, editar o eliminar bicicletas.

Precondición: Administrador autenticado.

Postcondición: Información actualizada.

Flujo principal:

- - Abre módulo de bicicletas.
- - Añade, edita o elimina registro.

Excepciones:

- - Bicicleta ya registrada o no existe.

Caso de Uso: Gestionar reportes

Actor: Técnico

Descripción: Revisa y resuelve reportes de fallas.

Precondición: Técnico autenticado.

Postcondición: Reporte resuelto.

Flujo principal:

- - Consulta lista de reportes.
- - Marca como resuelto.

Excepciones:

- - No hay reportes o sistema falla al guardar.

Matriz de Trazabilidad de Requerimientos

ID Requerimiento	Descripción	Casos de Uso Relacionados	Historias de Usuario Relacionadas
RF01	Autenticación mediante Google	Autenticarse	-
RF02	Reserva de bicicletas	Reservar bicicleta	HU1
RF03	Devolución de bicicletas	Devolver bicicleta	-
RF04	Reportar problemas	Reportar problema	HU3
RF05	Visualizar y suspender usuarios	Gestionar usuarios	-
RF06	Consultar disponibilidad en terminales	Reservar bicicleta	HU1
RF07	Ver historial de viajes	Ver historial	HU2
RF08	Visualizar reportes pendientes	Gestionar reportes	HU5
RF09	Cambiar terminal de destino durante trayecto	Modificar destino	-
RF10	Redistribución de bicicletas entre terminales	Redistribuir bicicletas	-
RF11	Realizar mantenimiento a bicicletas	Realizar mantenimiento	HU5
RF12	Visualizar ubicación de bicicletas en tiempo real	-	HU4

Diseño

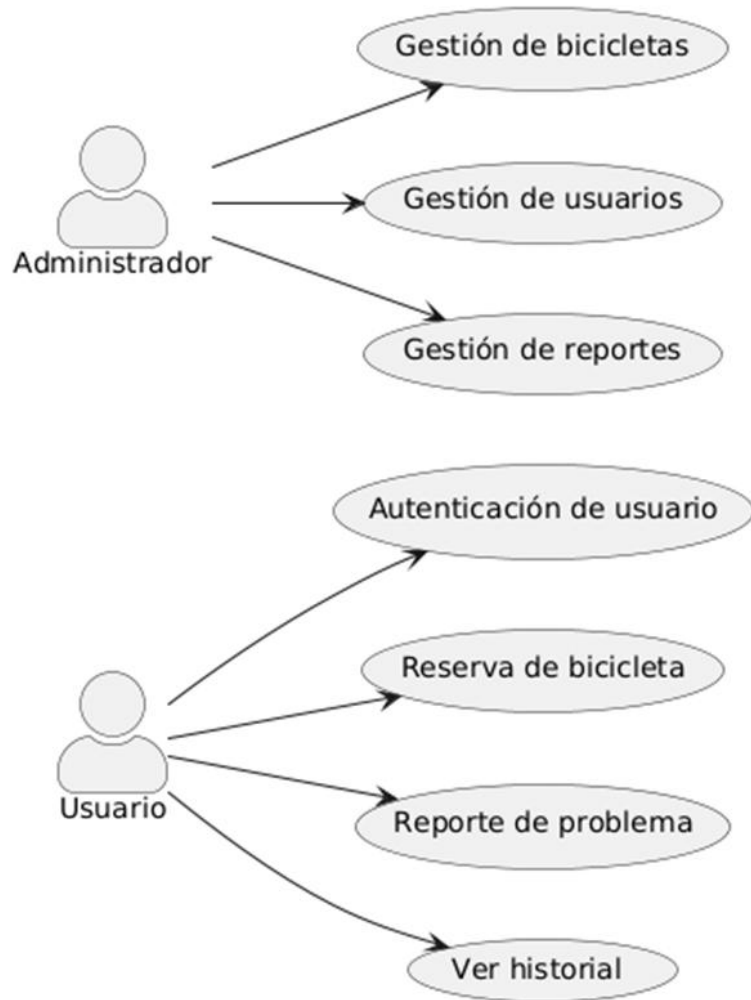
Arquitectura del sistema:

Patrones Aplicados y Justificación Técnica

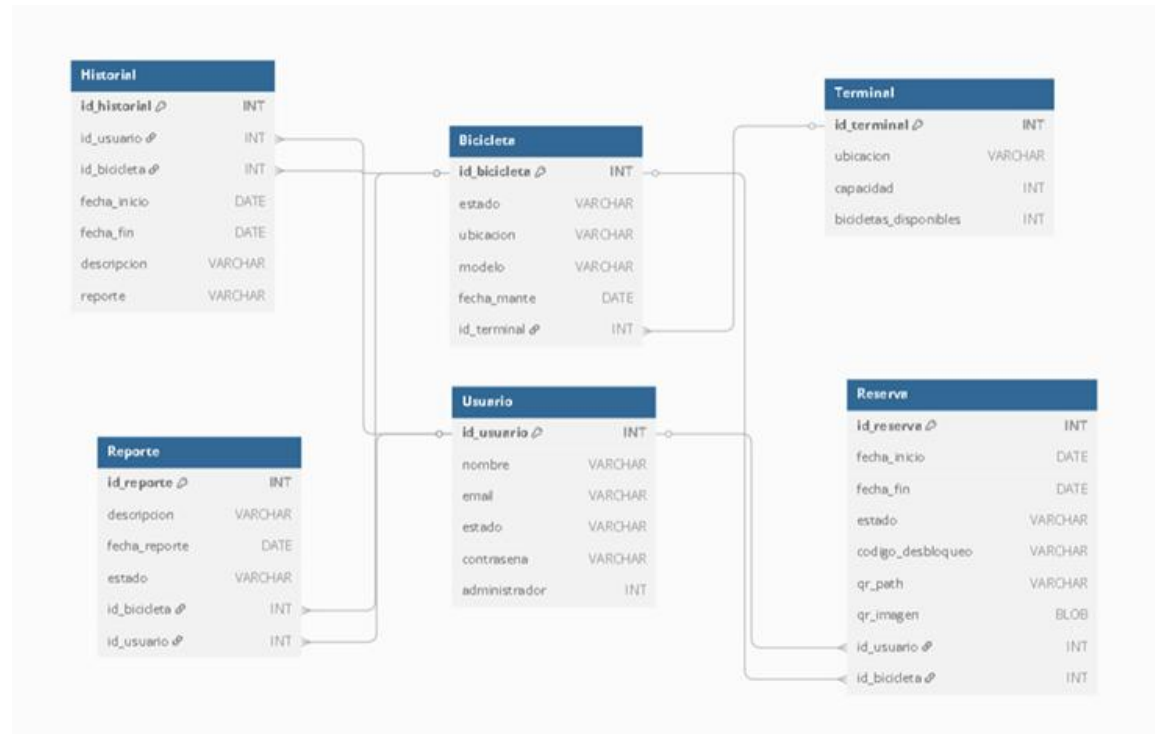
Patrón Aplicado	Justificación Técnica
MVC (Modelo-Vista-Controlador)	Se utilizó como patrón base para estructurar la arquitectura del sistema. Express.js organiza los archivos en `models/`, `views/` y `controllers/`, separando claramente la lógica de presentación, negocio y datos. Esto facilita el mantenimiento y escalabilidad del sistema.
Singleton	Aplicado a la conexión con Oracle Database. Se reutiliza una única instancia de conexión (`oracledb.getConnection(...)`), lo que evita sobrecarga y mejora el uso eficiente de recursos.
Módulo (Modularización)	Uso de `require()` para importar librerías como `express`, `oracledb`, `qrcode`, encapsulando funcionalidad y manteniendo el código limpio, reutilizable y sin variables globales.
Fachada (Facade)	Cada endpoint del backend actúa como una fachada, ocultando la complejidad del acceso a datos y lógica de negocio frente al frontend. Esto simplifica la comunicación entre capas.
Transaction Script	Especialmente en la funcionalidad de reservas: cada endpoint ejecuta pasos secuenciales (verificar usuario, disponibilidad, insertar reserva, actualizar estado, etc.), lo que permite una lógica clara y procedural.
Repository	Encapsulación del acceso a la base de datos dentro de cada endpoint, facilitando cambios y acceso controlado a los datos sin exponer consultas SQL directamente.
Controller	Los endpoints `app.get`, `app.post`, etc., en Express funcionan como controladores RESTful, procesando solicitudes y retornando respuestas adecuadas a cada acción del cliente.

Modelado:

Diagramas UML

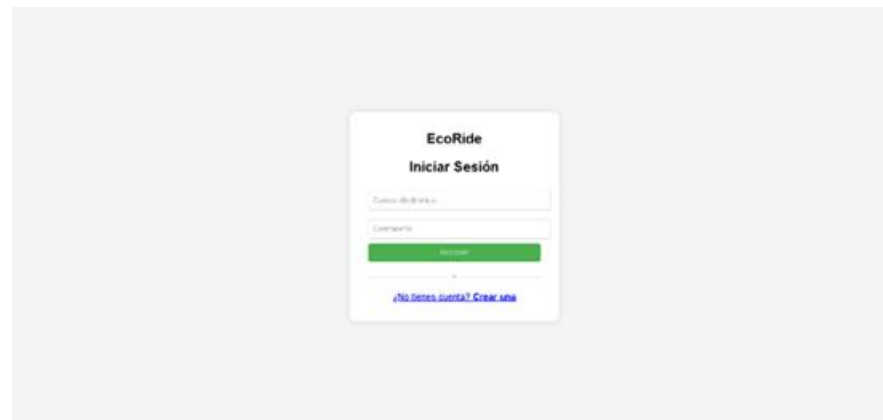


Modelo Entidad Relación



Diseño de interfaces

- Wireframes:



EcoRide

REGISTRO

Nombre

Apellido

Celular

Contraseña

Registrarse

EcoRide

ReservarDevolverHistorialPerfil

Perfil del Usuario

Nombre: Juan Pérez

Correo: juan@example.com

Estado: Activo

Editar Perfil

Reservar Bicicleta

Origen:

Terminal Central

Destino:

Terminal Central

Confirmar Reserva

LogoutEcoRide

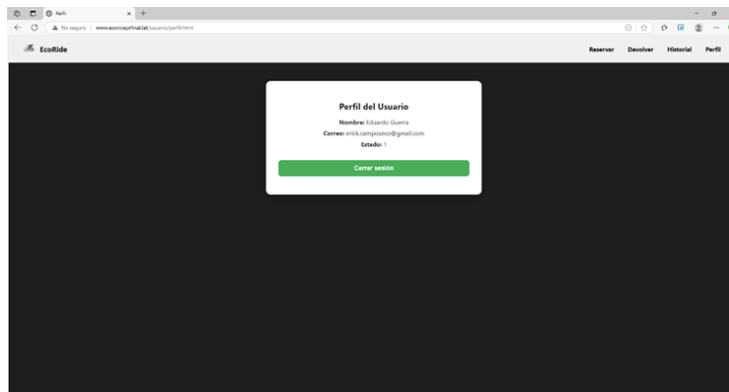
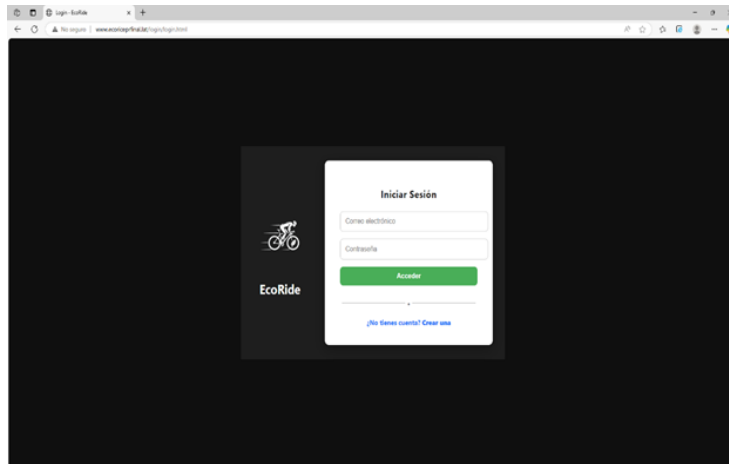
ReservarDevolverHistorialPerfil

¿Deseas devolver la bicicleta?

Quando hayas asegurado la bicicleta en una terminal, presiona el botón.

Confirmar Devolución

Mockups



Principios de Usabilidad Aplicados:

Principio	Aplicación en el Proyecto
Intuición y Simplicidad	Se especifica que la interfaz debe ser intuitiva para que los usuarios puedan reservar y devolver bicicletas fácilmente.
Adaptabilidad / Responsividad	Se requiere que funcione en computadoras, tabletas y smartphones, asegurando acceso desde cualquier dispositivo.
Retroalimentación Inmediata	Se incluyen notificaciones y confirmaciones (como generación de código QR o mensaje de devolución correcta).
Prevención de errores	Validaciones en formularios y respuestas adecuadas para errores como “bicicleta no reconocida” o “usuario no autorizado”.
Consistencia visual	Se propone un CSS uniforme en todas las vistas y mejora del diseño visual con Bootstrap o librerías similares.

Validación de Entrega

Evidencias de Pruebas de Aceptación (UAT)

Durante las pruebas de aceptación se validaron las siguientes funcionalidades con usuarios reales y datos simulados:

- Inicio de sesión de usuarios con autenticación básica (funcional).
- Reserva de bicicletas desde múltiples terminales (funcional).
- Generación y lectura de código QR (funcional).
- Devolución de bicicletas y actualización de estado (funcional).
- Reporte de fallas y visualización en panel de administrador (funcional).
- Gestión de usuarios, bicicletas y reportes (funcional).

Entorno de pruebas: <http://31.97.134.125/test/>

Criterios de Aceptación Cumplidos

Criterio	Resultado
Inicio de sesión y registro	Cumplido
Reserva con confirmación de código QR	Cumplido
Devolución de bicicleta	Cumplido
Reporte de problema	Cumplido
Gestión de usuarios y bicicletas	Cumplido
Acceso por roles diferenciados	Cumplido
Visualización de historial	Cumplido

Observaciones y Conformidades Finales

- El proyecto fue entregado en la fecha establecida (9 de junio de 2025).
- La versión v1.2.0 cumple con los requisitos funcionales establecidos en el contrato.
- Las pruebas de usuario confirmaron que el sistema es estable, usable y cumple con el objetivo general.
- No se detectaron errores críticos durante las pruebas UAT.
- Se recomienda integrar HTTPS y mejorar medidas OWASP en futuras versiones.

Conclusión: El sistema "EcoRide" fue validado y aprobado por el cliente. Se declara entregado y en conformidad con los criterios establecidos.