



Versión 1, Ciclo 3 - 2022

Sprint 1

Báez Andrés, Cruz Leonardo, Bernal Juan, Valencia Jhonatan, y Perez Joaquín

afbaeza@unal.edu.co

cruzeto76@gmail.com

bernal.felipe98@gmail.com

jeivymusica@gmail.com

joaquin.perez-1520@outlook.com

i. Repositorio GitHub:

https://github.com/luis-leonardo-cruz/RETO_CICLO_3.git

ii. Gestión de configuración:

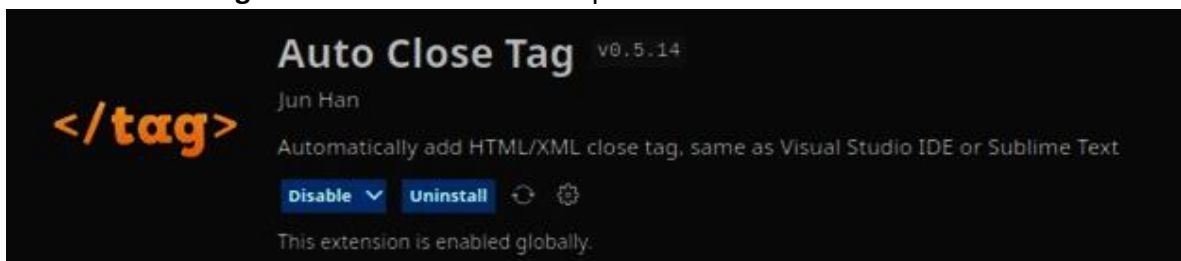
I. Instalación del editor de código

- Selección del editor de código: **Visual Studio Code**.



II. Instalación de extensiones

- **Auto Close Tag:** Cierre automático de etiquetas HTML.



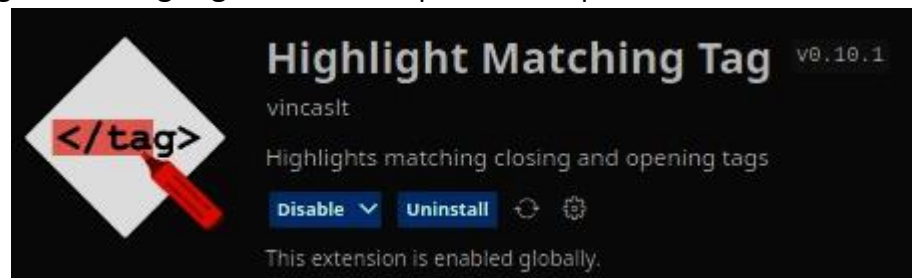
- **Auto Rename Tag:** Renombrado automático de etiquetas HTML.



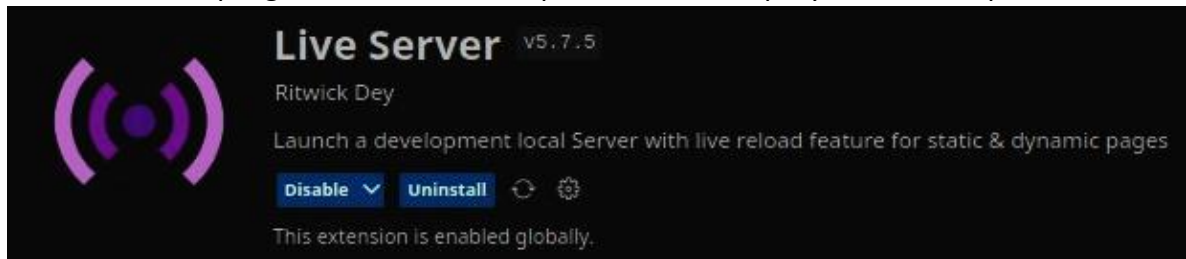
Versión 1, Ciclo 3 - 2022



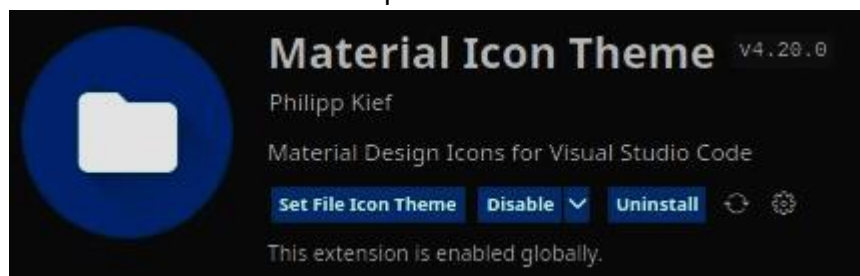
- **Highlight Matching Tag:** Resaltado de pares de etiquetas HTML.



- **Live Server:** Despliegue de servidor local para visualizar el proyecto en tiempo real.



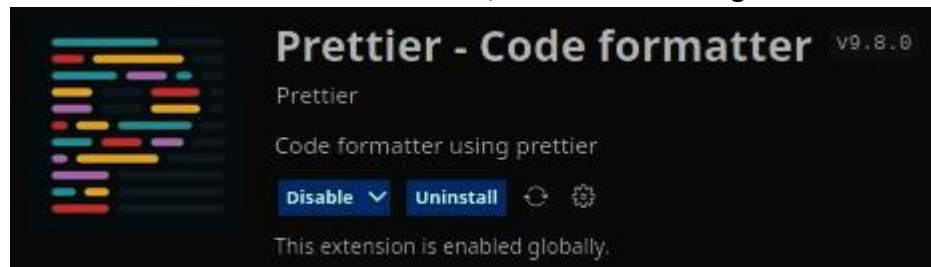
- **Material Icon Theme:** Tema Material para íconos.



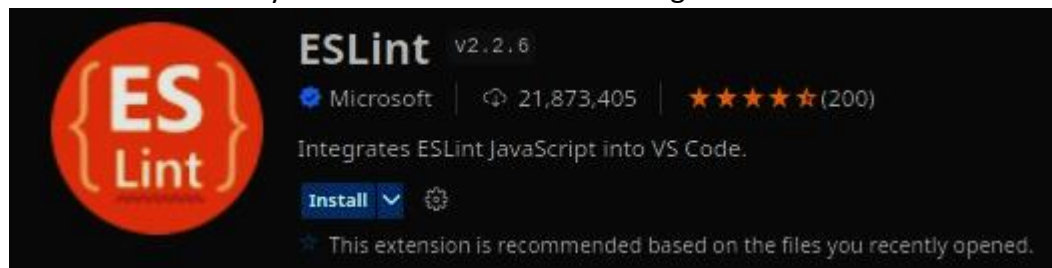
- **Path Intellisense:** Autocompletado de rutas/nombres de archivos.



- **Prettier - Code formatter:** “Embellecedor” /Formateo de código.



- **ESLint:** Identificación y corrección de errores en código JS.



- **JavaScript (ES6) code snippets:** Recomendaciones de bloques de código JS.



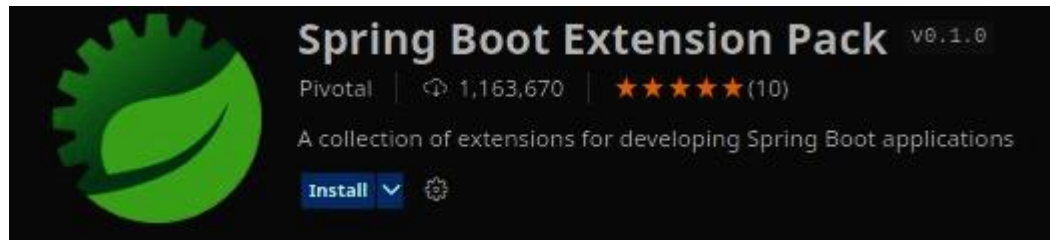
- **Extension Pack for Java:** Paquete de extensiones para trabajo con Java.



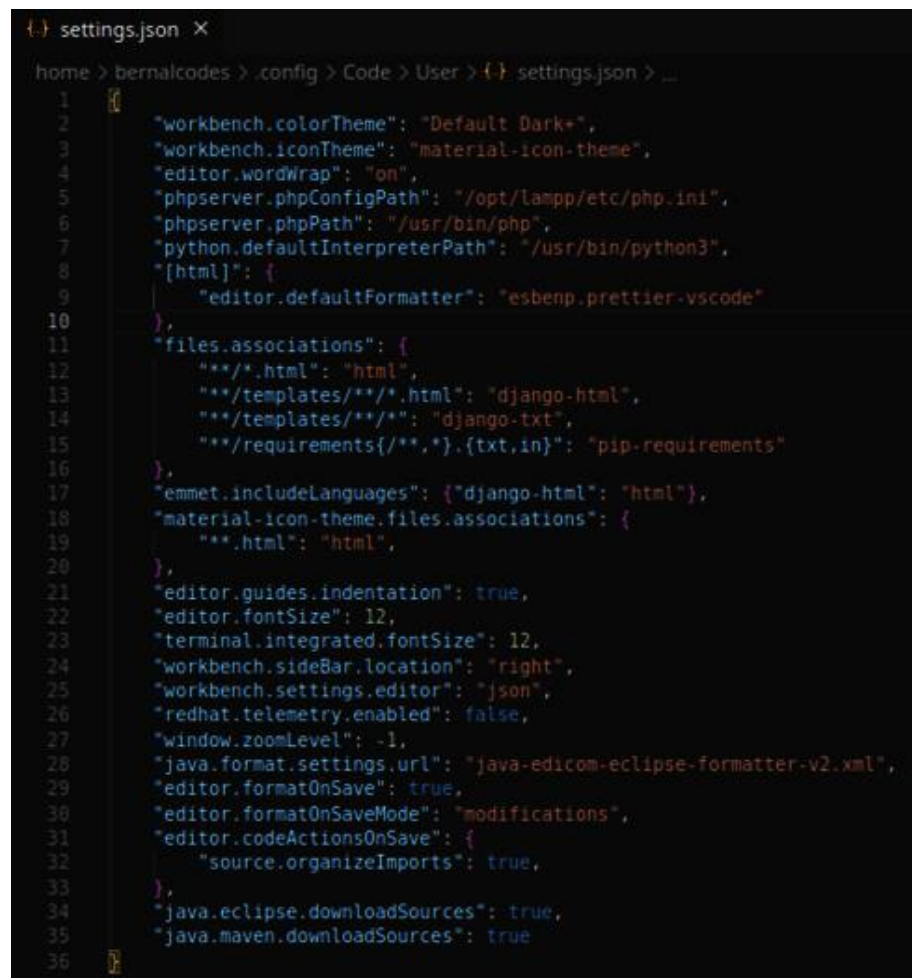
Versión 1, Ciclo 3 - 2022



- **Spring Boot Extension Pack:** Paquete de extensiones para trabajo con Spring Boot.



- **Configuración del IDE**

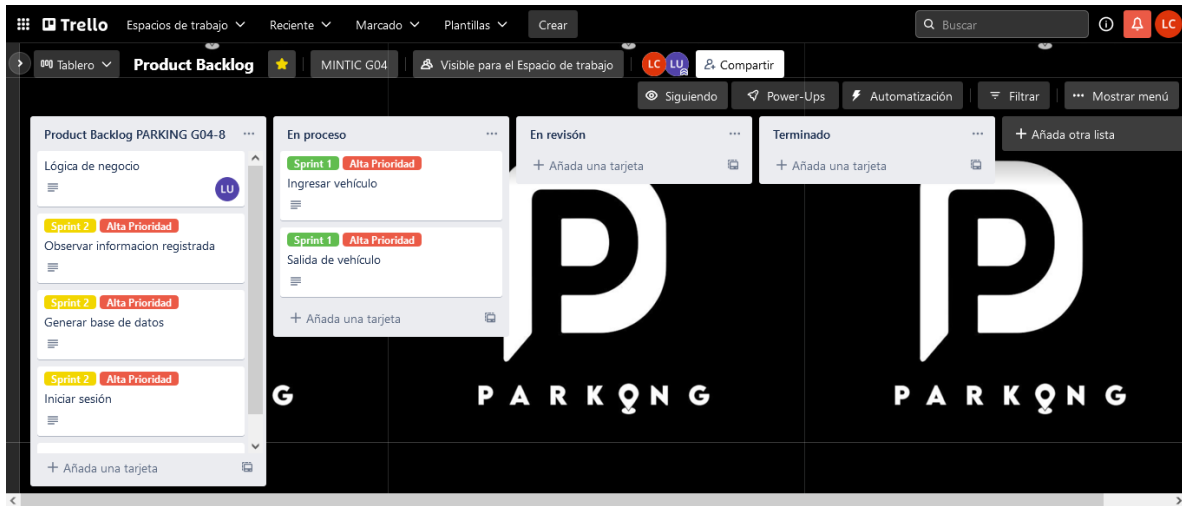




Versión 1, Ciclo 3 - 2022

iii. Historias de usuario (Sprint 1)

<https://trello.com/b/vR3VkJMoo/product-backlog>





Versión 1, Ciclo 3 - 2022



iv. Presentación MVC

I. Introducción

Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.¹

- El Modelo que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.¹
 - La Vista, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste.¹
 - El Controlador, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.¹
- a. El modelo es el responsable de:
- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.¹
 - Define las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor".¹
 - Lleva un registro de las vistas y controladores del sistema.
 - Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo (por ejemplo, un fichero por lotes que actualiza los datos, un temporizador que desencadena una inserción, etc.).¹

¹ Universidad de Alicante, «Modelo vista controlador (MVC)», >> 2022. [En línea]. Available: <https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelovista-controlador-mvc.html>.



Versión 1, Ciclo 3 - 2022

- b.** El controlador es responsable de:
 - Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).¹
 - Contiene reglas de gestión de eventos, del tipo " SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método "Actualizar ()". Una petición al modelo puede ser "Obtener tiempo de entrega (nueva orden de venta)".¹
- c.** Las vistas son responsables de:
 - Recibir datos del modelo y los muestra al usuario.¹
 - Tienen un registro de su controlador asociado (normal- mente porque además lo instancia).¹
 - Pueden dar el servicio de. Actualización ()", para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).¹
- d.** El flujo que sigue el control generalmente es el siguiente:
 - El usuario interactúa con la interfaz de usuario de alguna forma (por ejemplo, el usuario pulsa un botón, enlace, etc.).¹
 - El controlador recibe (por parte de los objetos de la interfaz-vista) la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.¹
 - El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario (por ejemplo, el controlador actualiza el carro de la compra del usuario). Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.¹
 - El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz para el usuario donde se refleja los cambios en el modelo (por ejemplo, produce un listado del contenido del carro de la compra). El modelo no debe tener conocimiento directo sobre la vista. Sin embargo, se podría utilizar el patrón Observador para proveer cierta indirección entre el modelo y la vista, permitiendo al modelo notificar a los interesados de cualquier cambio. Un objeto vista puede registrarse con el modelo y esperar a los cambios, pero aun así el modelo en sí mismo sigue sin saber nada de la vista.¹
 - La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.¹

II. Interfases gráficas de usuario

a. Menú principal

El menú principal de nuestro aplicativo contar 'a con un menú de navegación entre pantallas las cuales estarán disponibles dependiendo el rol de usuario con el cual se inicie sesión en el aplicativo. A esta pantalla se accede después de que el usuario hace el registro respectivo e inicia sesión.

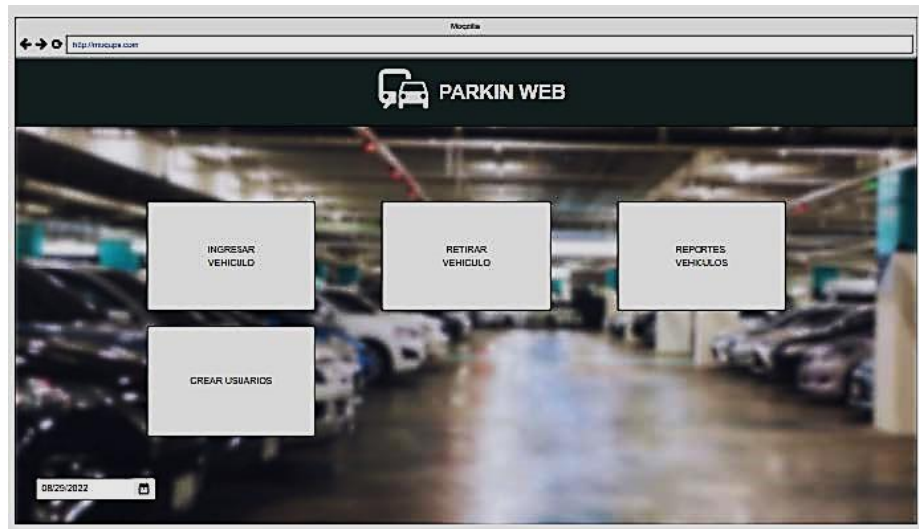


ILUSTRACIÓN 1. PANTALLA PRINCIPAL

b. Crear Usuarios

Para que un empleado pueda tener acceso al aplicativo es necesario que primero llene un registro de datos para crear un usuario con el rol de Empleado, el cual podrá tener acceso al menú de registrar la entrada del vehículo, registrar la salida del vehículo y revisar el historial de ingreso al establecimiento de su vehículo.

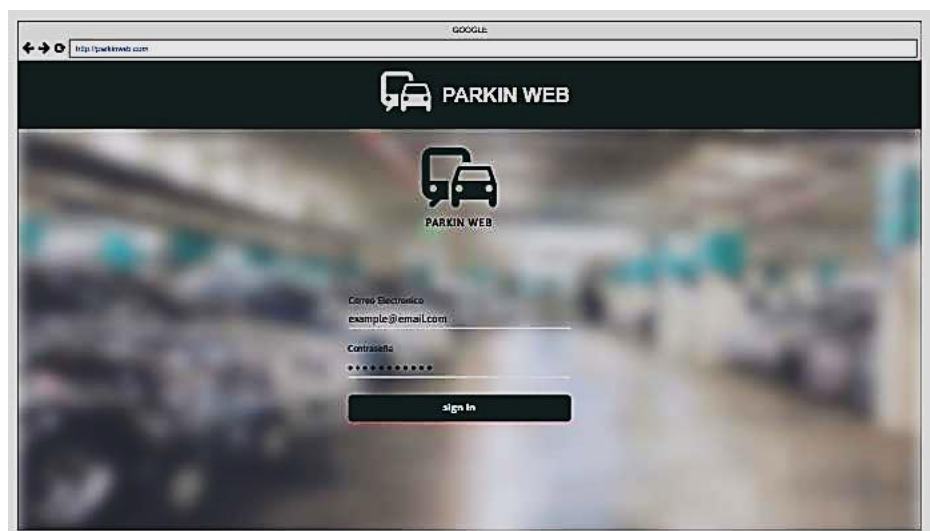
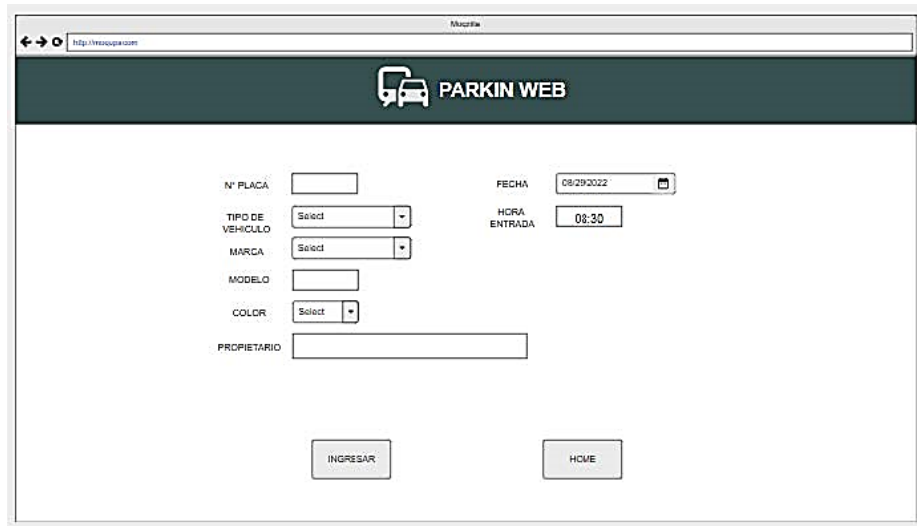


ILUSTRACIÓN 2. PANTALLA DE REGISTRO

c. Registrar ingreso del vehículo

Esta pantalla está disponible tanto para el empleado como para el administrador, el empleado tendrá dos opciones de vista para registrar al cliente, si desea comprar un cupo por un tiempo prolongado que puede ser mínimo 1 mes, o comprar el servicio por horas. Para el cupo por tiempo prolongado el empleado necesita solicitarle al cliente más datos, como número de teléfono, correo, dirección de residencia, placa y tipo del vehículo, y si solo es un servicio por horas solo necesita solicitar la placa y el tipo de vehículo.



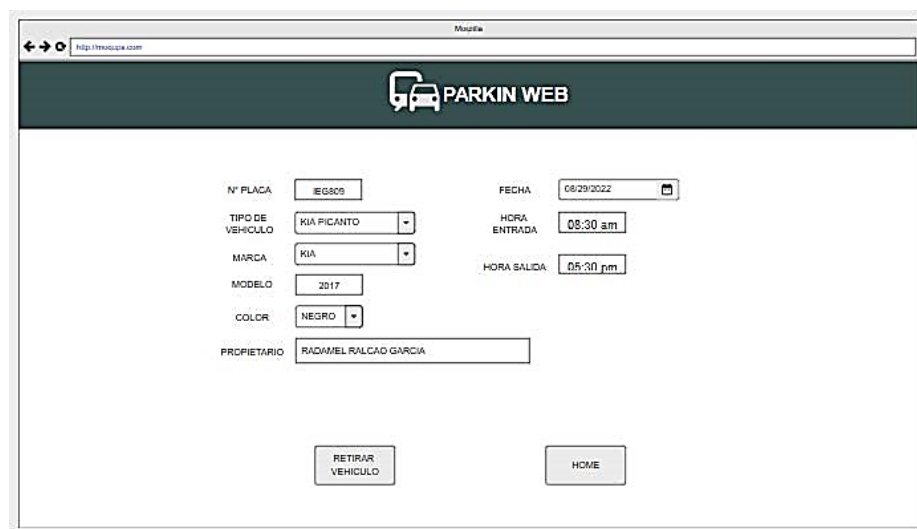
The screenshot shows a web browser window with the URL <http://misiontic.com>. The page header features the 'PARKIN WEB' logo. The main form contains the following fields and controls:

- N° PLACA:** A text input field.
- FECHA:** A date picker set to 08/29/2022.
- TIPO DE VEHICULO:** A dropdown menu with 'Select' as the current value.
- MARCA:** A dropdown menu with 'Select' as the current value.
- MODELO:** A text input field.
- COLOR:** A dropdown menu with 'Select' as the current value.
- PROPIETARIO:** A text input field.
- HORA ENTRADA:** A time picker set to 08:30.
- Buttons:** 'INGRESAR' and 'HOME' at the bottom.

ILUSTRACIÓN 3. PANTALLA REGISTRAR INGRESO DEL VEHÍCULO

d. Registrar Salida de vehículo

El registro de salida vehículo el empleado solo necesita ingresar el número de placa del vehículo y este le indicara al cliente el numero el tiempo del servicio y el dinero a cancelar



The screenshot shows the same web browser window as Illustration 3, but with the following data entered into the form:

- N° PLACA:** IE8809
- FECHA:** 08/29/2022
- TIPO DE VEHICULO:** KIA PICANTO
- MARCA:** KIA
- MODELO:** 2017
- COLOR:** NEGRO
- PROPIETARIO:** RADAMEL RALCAO GARCIA
- HORA ENTRADA:** 08:30 am
- HORA SALIDA:** 05:30 pm
- Buttons:** 'RETIRAR VEHICULO' and 'HOME' at the bottom.

ILUSTRACIÓN 4. PANTALLA SALIDA DEL VEHÍCULO



Versión 1, Ciclo 3 - 2022

e. Reporte de Vehículos

Esta página estará disponible tanto para los administradores como para el empleado y cliente, El administrador podrá ver en su totalidad el reporte de Vehículos ingresados al estacionamiento con gráficos estadísticos y reportes que podrán ser filtrados por fecha, y el cliente tendrá acceso al Historial de ingresos de su vehículo al estacionamiento, con datos como tiempo del servicio, fecha y valor de este.



N°PLACA	TIPO DE VEHICULO	MARCA	MODELO	PROPIETARIO	FECHA	HORA ENTRADA	HORA SALIDA
INT48D	MOTO	YAMAHA	2017	ANDREA CASTAÑO	29/08/2022	08:30AM	10:30PM
FWW48D	MOTO	HONDA	2007	CARLOS JIMENEZ	29/08/2022	10:12AM	04:30PM
LKP605	CARRO	MAZDA	2018	JUAN DAVID VALENCIA	29/08/2022	12:12PM	07:30PM

Buttons: IMPRIMIR REPORTE, HOME

ILUSTRACIÓN 5. PANTALLA REPORTES DE VEHÍCULO

III. Logo del Aplicativo



ILUSTRACIÓN 6. LOGO DEL APLICATIVO

IV. Navegación entre Interfases

La navegación entre pantallas de nuestro aplicativo tendrá la estructura mostrada en la siguiente imagen.

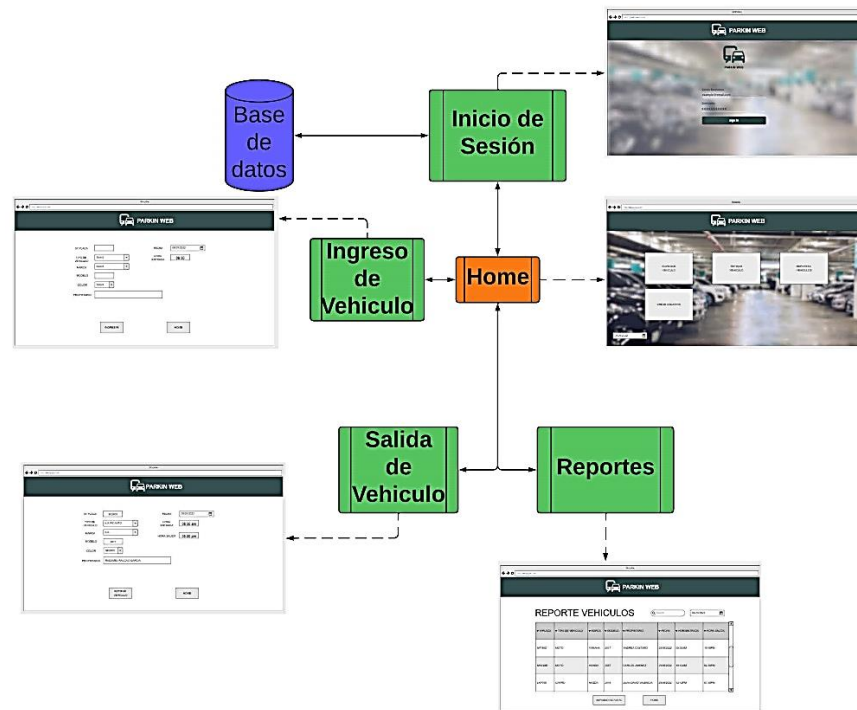


ILUSTRACIÓN 7. NAVEGACIÓN ENTRE PANTALLAS

v. Informe de retrospectiva

REUNIÓN SEMANAL DE SCRUM

ACTA No. 01

28/08/2022 – 03/09/2022

SPRINT No. 1 SEMANA No. 1

ORDEN DEL DÍA

1. Asignación de funciones
2. Seguimiento a tareas
3. Dificultades presentadas
4. Varios

DESARROLLO DEL ORDEN DEL DÍA



Versión 1, Ciclo 3 - 2022

1. Luego de creación de grupo de WhatsApp, se plantean ideas de proyectos y funciones específicas para cada uno de los integrantes del equipo, definiendo como proyecto desarrollar una plataforma para la administración de un parqueadero público.

Las funciones asignadas dentro del equipo quedaron distribuidas de la siguiente manera:

- GESTOR DE PROYECTO, Leonardo Cruz
 - GESTOR BASE DE DATOS, Joaquín Perez
 - DESARROLLADOR BACKEND, Juan Felipe Bernal
 - DESARROLLADOR FRONTEND, Jhonatan Valencia
 - TESTER, Andres Felipe Báez
2. De acuerdo con las funciones asignadas a cada uno de los integrantes del equipo y según los requerimientos del sprint 1, se asignan los siguientes compromisos:
 - a. Leonardo Cruz, generación del repositorio de GITHUB.
 - b. Juan Felipe Bernal, gestión de configuración.
 - c. Historias de usuario (Trello). Todo el equipo.
 - d. Andres Felipe Báez, presentación MVC.
 - e. Jhonatan Valencia, MOCKUP.
 - f. Joaquín Perez, preliminar de usuarios para generación de base de datos.
 3. Dificultades presentadas:
 - Desarrollo de historias de usuario utilizando la aplicación TRELLO. Dudas en los procedimientos para realizar la actividad. Desconocimiento del aplicativo.
 4. Varios:
 - Durante la primera reunión del equipo se definieron días y horarios para la realización de las reuniones de equipo, las cuales tienen como objetivo la presentación de dudas, compartir avances, entre otros. Los horarios definidos son:
 - Martes, al terminar la clase de programación.
 - Jueves, 7:00 pm.
 - Sábado, 9:00 am.
 - En todas las reuniones hubo participación de todo el equipo.
 - Se genera Drive para compartir documentos, archivos, entre otros con el fin de facilitar el acceso por parte de todos los integrantes del equipo.
 - Se plantea lluvia de ideas para el nombre del proyecto y su respectivo logo.

COMPROMISO	RESPONSABLE	FECHA
Configuración de repositorio (GITHUB)	Leonardo Cruz	1/09/2022
Gestión de configuración	Juan Felipe Bernal	1/09/2022



Versión 1, Ciclo 3 - 2022

Presentación base MVC.	Andres Felipe Báez	3/09/2022
Generar MOCKUP	Jhonatan Valencia	3/09/2022
Preliminar de usuarios (BD)	Joaquín Perez	3/09/2022
Historias de usuario en TRELLO	Todo el equipo	3/09/2022

ASISTENTES

GESTOR DE PROYECTO, Leonardo Cruz

GESTOR BASE DE DATOS, Joaquín Perez

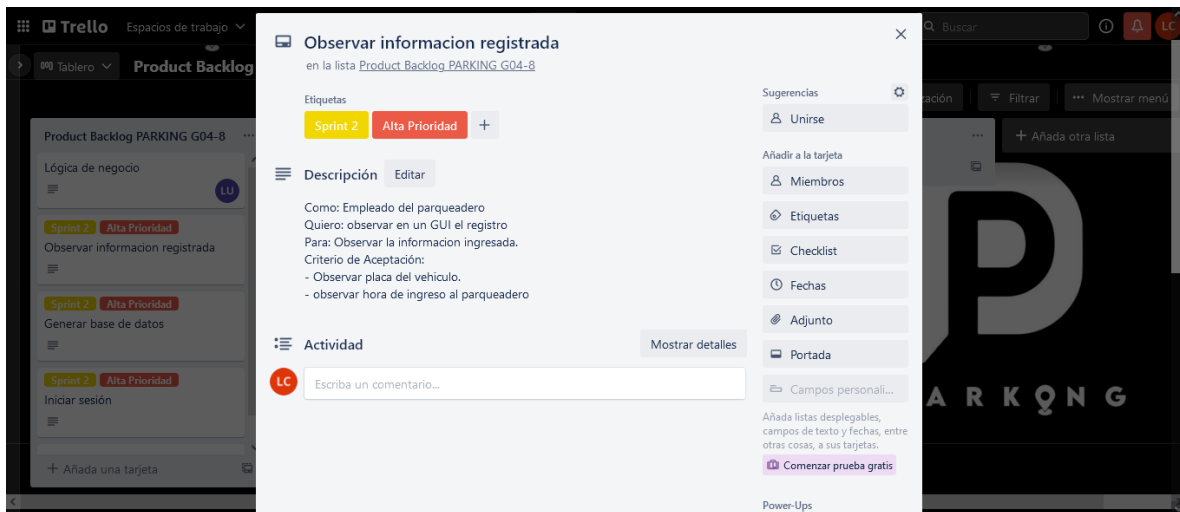
DESARROLLADOR BACKEND, Juan Felipe Bernal

DESARROLLADOR FRONTEND, Jhonatan Valencia

TESTER, Andres Felipe Báez

vi. Historias de usuario a desarrollar en el sprint 2.

<https://trello.com/b/vR3VkJMoo/product-backlog>





Versión 1, Ciclo 3 - 2022

Generar base de datos
en la lista [Product Backlog PARKING GO4-8](#)

Etiquetas
Sprint 2 **Alta Prioridad** +

Descripción Editar

Como: Empleado del parqueadero
Quiero: Almacenar la información del uso del parqueadero.
Para: Revisar historicos del uso del parqueadero y generar reportes
Criterio de Aceptación:
- Crear contenedor de base datos.

Actividad [Mostrar detalles](#)

LC Escribe un comentario...

Sugerencias

- Unirse
- Miembros
- Etiquetas
- Checklist
- Fechas
- Adjunto
- Portada
- Campos personal...

Añade listas desplegables, campos de texto y fechas, entre otras cosas, a sus tarjetas.

[Comenzar prueba gratis](#)

Power-Ups

Iniciar sesión
en la lista [Product Backlog PARKING GO4-8](#)

Etiquetas
Sprint 2 **Alta Prioridad** +

Descripción Editar

Como: Empleado del parqueadero
Quiero: Ingresar al aplicativo
Para: Acceder a las funciones del aplicativo
Criterio de Aceptación:
- El usuario debe proporcionar email.
- El usuario debe proporcionar contraseña.
- El usuario accede a las funciones del aplicativo de acuerdo con su rol

Actividad [Mostrar detalles](#)

LC Escribe un comentario...

Sugerencias

- Unirse
- Miembros
- Etiquetas
- Checklist
- Fechas
- Adjunto
- Portada
- Campos personal...

Añade listas desplegables, campos de texto y fechas, entre otras cosas, a sus tarjetas.

[Comenzar prueba gratis](#)

Power-Ups