

Universidad Luterana Salvadoreña

**Facultad de Ciencias del Hombre y la naturaleza Licenciatura
en Ciencias de la Computación Cátedra Programación III**



Proyecto: Sistema de Gestión de Parking

Docente:

Lic. Jorge Alberto Coto

Zelaya

Ciclo/Año:

II/2021

No	Apellidos	Nombres	Carnet	Partición
1	Martínez Cubias	Jaaziel Isaac	MC01135738	100%
2	Torres García	Miguel Alejandro	TG01135889	100%
3	Zelaya Moran	Edgar Esaú	ZM01135974	100%
4	Argueta Jovel	Anderson Fernando	AJ01135837	100%

San Salvador, 18 Noviembre del 2021

INDICE

Introducción	4
Objetivos	5
Objetivo General.....	5
Objetivos específicos	5
Justificación.....	6
Marco Teórico	7
Descripción de la Empresa	7
Misión y Visión de la Empresa.....	7
Descripción del proyecto	7
Análisis y requerimientos del sistema	9
Requerimientos del usuario.	11
Herramientas a utilizar para el desarrollo del proyecto	12
Modelado UML.....	15
Diagrama de Caso de Uso	15
Diagrama de Clases	15
Modelo Físico BD	16
Alcances y limitaciones del proyecto	17
Alcances.....	17
Limitaciones	17
Conclusiones	18
Recomendaciones.....	19
Referencias bibliográficas	20
Anexos.....	21
Manual de usuario	21
Diagrama de actividades.....	35

TABLA DE ILUSTRACIONES

Ilustración 1Laragon.....	12
Ilustración 2Xampp	13
Ilustración 3Laravel.....	13
Ilustración 4Phpmyadmin.....	13
Ilustración 5Visual Studio Code.....	14
Ilustración 6Livewire.....	14
Ilustración 7Bootstrap.....	14
Ilustración 8Diagrama de Caso de Uso.....	15
Ilustración 9Diagrama de Clases	16
Ilustración 10Modelo fisico de la BD.....	16
Ilustración 11Bienvenida Laragon.....	22
Ilustración 12Config Laragon.....	22
Ilustración 13Panel de Laragon	23
Ilustración 14Instalador Composer.....	23
Ilustración 15Config Composer.....	24
Ilustración 16Config 2 Composer.....	24
Ilustración 17Listo para instalar	25
Ilustración 18Final instalación.....	25
Ilustración 19Comprobar version	26
Ilustración 20Phpmyadmin en Laragon.....	26
Ilustración 21Colocar la ruta	27
Ilustración 22Iniciar sesión.....	27
Ilustración 23Panel administrativo phpmy..	28
Ilustración 24Crear base de datos	28
Ilustración 25Identificar la BD	29
Ilustración 26Ruta de proyecto	29
Ilustración 27Ejecutar las migraciones.....	29
Ilustración 28Ejecutar los seeders.....	30
Ilustración 29Levantar el servidor	30
Ilustración 30Identificar acceso al sistema	30
Ilustración 31Iniciar sesión.....	31
Ilustración 32Panel de Gestión	31
Ilustración 33Datos de la empresa.....	32
Ilustración 34Confirmación de registro	32
Ilustración 35Gestionar usuario	32
Ilustración 36Gestionar usuario 2	33
Ilustración 37Roles y permisos.....	33
Ilustración 38Roles y permisos 2.....	33
Ilustración 39Asignar roles.....	34
Ilustración 40Cerrar sesión	34
Ilustración 41Eliminacion de usuario	34
Ilustración 42Diagrama de actividades.....	35

Introducción

El presente proyecto denominado Sistema de Gestión de Parqueo (SGP) es muy importante para lo siguiente: dashboard dinámico que cuente con un panel gráfico de todos los estacionamientos físicos, también para llevar tarifas del sistema para el cobro de los recibos. Otra parte muy importante es llevar un reporte de ventas diarias, reporte de ventas por fecha.

El uso adecuado del sistema pretende solucionar el problema, dado que recortará tiempos en relación al servicio que se ofrece, dará un mejor servicio y garantizará un mejor control de los datos.

Se planteado como equipo, que nuestra empresa se llamará Global Parking lo cual sería un Sistema de gestión de Parqueo en pocas palabras seria: SGP en el cual se brinda seguridad y al momento que uno de los clientes nuestros clientes compren tickets se les asignará un boleto que tendrá su respectivo precio por la hora, depende de cuántas horas quiere usar el estacionamiento así se irá elevando el precio. Tomando en cuenta que no solamente se trata de un parqueo de horas, sino también ofrece servicio de parking de mayor duración de días, lo que el cliente necesite. Otro punto a recalcar es que no solamente es un parqueo para transporte común como: camionetas, buses, carros tipo sedan. Sino también para transporte que incluye bicicletas, cuatrimotos o algún otro medio de transporte que sea aceptado por la empresa.

Objetivos

Objetivo General

- Implementar un sistema que gestione el proceso de cobro y disponibilidad de espacios del parqueo privado denominado “Sistema de Gestión de Parqueo (SGP)”.

Objetivos específicos

- Desarrollar el sistema con el fin de automatizar el proceso administrativo del parqueo.
- Mejorar la gestión administrativa de horas del uso del parqueo, para el debido cobro a los usuarios.
- Conocer, sobre el framework Laravel para su implantación en aplicaciones y sistemas.

Justificación

En relación a la justificación del proyecto se pretende realizar un sistema para la empresa Global Parking la cual está enfocada a brindar servicios de parqueo y que el mismo sea una herramienta muy fundamental para ofrecer un mejor servicio a los clientes. El proyecto contará con las funciones más óptimas y específicas para una correcta gestión de parqueos. Para la realización del sistema en general se utilizará el framework Laravel el cual es de código abierto para desarrollar aplicaciones y servicios web con PHP 5, PHP 7 y PHP 8. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". Fue creado en 2011. Es por ello, que se decidió utilizar dicho framework para la realización del sistema aprovechando cada una de las ventajas que nos ofrece Laravel

Marco Teórico

Descripción de la Empresa

Global Parking lo cual sería un Sistema de gestión de Parqueo en palabras más traducidas SGP en la cual vamos a brindar seguridad y al momento que uno de nuestros clientes comprantickets se les asignará un boleto con su respectivo costo, depende de cuánto tiempo quiere usar el estacionamiento así se irá elevando el precio. Nuestra empresa se alojará en una plazade Zacatecoluca en la cual llevará un control de cuántos vehículos entran y cuantos salen, luego del tiempo estipulado tendrán que pagar en la cual nuestro sistema de parqueo será privado.

Misión y Visión de la Empresa

Misión de la empresa

Ofrecer a nuestros clientes una seguridad muy buena estar pendiente de cualquier obra que se haga en la empresa y poder ofrecer nuestro Sistema sin fallas y que genere los tickects vigilar la hora y entrada de los vehículos y motocicletas y sobre todo dar un excelente servicio de nuestro sistema.

Visión de la empresa

Como empresa Global parking tenemos expectativas a futuro sobre agregar nuevas actualizaciones al Sistema dar un mejor mantenimiento y siempre que este generando reportes y facturas como de los clientes que nos visiten y dar siempre un mejor servicio y seguir mejorando día a día.

Descripción del proyecto

El presente proyecto denominado Sistema de Gestión de Parqueo (SGP) es muy importante para la

administración por medio de un dashboard dinámico que cuente con un panel gráfico de las ventas. Podremos identificar fácilmente el incremento o decremento de las ventas que se han realizado. Prácticamente, servirá para llevar un registro esquematizado de las ventas especificadas. A demás tendrá otro gráfico en el que se mostrara la cantidad de ventas por semana identificado cada día de la semana por colores, los cuales son los siguientes: lunes, verde; martes, naranja; miércoles, rojo; jueves, azul; viernes, verde turquesa; sábado y domingo, morado. Siempre en la parte del dashboard se encuentra un apartado de “Rentas” donde podremos ver las entradas y salidas como también tickets extraviados. En las entradas y salidas podremos generar un ticket de visita como también dar salida a un tipo de vehículo. Finalmente en el ticket de visita observaremos datos específicos como: nombre de la empresa, lugar, fecha de entrada con la hora, tarifa por hora. Otra opción importante dentro de “Rentas” encontraremos un apartado de tickets extraviados los cuales se registran los datos como: código, entrada, datos del coche, placa, tipo, y la opción de dar salida.

Otro apartado dentro del dashboard es la “configuración” de la empresa, cajones de estacionamiento, tipos de vehículos, roles y permisos. Los datos que se pueden ingresar dentro de empresa son: nombre, teléfono, email, elegir un logo, dirección. Los de cajones de estacionamiento son: descripción, estatus, tipo, acciones de editar y eliminar. Por supuesto, la opción de gestionar cajones donde se permite agregar nombre del cajón, tipo de vehículo ver su estatus (disponible/ocupado). Dentro de tipos de vehículos se observan los siguientes: descripción, imagen de vehículo, acción de editar y eliminar, se puede editar la descripción y la imagen y por supuesto crear un nuevo tipo, asignándole una descripción e imagen. Por otra parte, dentro de la “configuración” encontramos la opción de roles y permisos los cuales almacena datos como: descripción, usuarios con el rol, acción de editar y eliminar y darle permisos para ciertas vistas de acuerdo a su rol. (admin, empleado) a usuarios registrados.

El apartado de “Tarifas” es otra parte fundamental del dashboard ya que se almacena datos importantes

como: tiempo (fracción, hora, día, semana, mes), descripción, costo, tipo de vehículo, acciones de editar y eliminar. En editar podemos actualizar datos: tiempo, tipo de vehículo, costo, descripción. Podemos también ingresar una nueva tarifa.

Otro apartado es el de “Caja” en la que encontramos opciones de hacer corte y movimientos. En la opción de hacer corte se puede: elegir la fecha, elegir operador. En cuestión de resultados se muestra las ventas, entradas, salidas. La opción de movimientos, guarda los siguientes datos: descripción, tipo (ingreso, gasto, pago de renta), monto, comprobante(archivo), fecha, acciones de editar y eliminar. Podemos agregar un nuevo movimiento con sus respectivos datos.

Los “Reportes” son muy esenciales debido a que en ellos podemos utilizar opciones como: ventas del día, ventas por fecha, rentas próximas a vencer. En reporte de ventas por día, podemos guardar datos como: vehículo, acceso, salida, tiempo, tarifa, importe, usuario, renta, fecha, observando en cantidad, la suma de importes.

Finalmente, el dashboard cuenta también con un apartado de “usuarios” los datos que se guardan son los siguientes: nombre, teléfono, móvil, email, tipo, dirección, acción de editar y eliminar los datos. Podemos editar su nombre, email. Teléfono, tipo (admin, cliente) móvil, password y dirección.

Análisis y requerimientos del sistema

Inicio de sesión al sistema

Al iniciar sesión en el sistema se presentará al usuario un formulario solicitando los siguientes campos:

Correo(registrado en el sistema)

Contraseña(registrada en el sistema)

Cabe recalcar que como es un sistema de gestión de parking no cuenta con un registro de usuario. Es decir, solo los administradores podrán registrar usuarios y delegar los respectivos permisos.

Registro de usuarios al sistema

Para registrar a un usuario nuevo el sistema solicitará lo siguiente: Nombre, teléfono, móvil, email, dirección, password. Para administrador y empleado.

El sistema debe almacenar toda esta información en la base de datos.

Mayor control

Se lleva un mejor control de todos los usuarios con sus respectivos datos y vinculado al sistema de acuerdo a su permiso.

Funciones del sistema

El sistema cuenta con algunas funciones como:

- Entradas y salidas
- Tickets extraviados
- Tipos de vehículos
- Roles y permisos
- Tarifas
- Hacer cortes y movimientos de caja,
- Reportes

- Usuarios.

El sistema permite realizar actualizaciones a diferentes datos muy importantes, se permiten realizar actualizaciones en cualquier momento, de acuerdo a lo que se solicite o necesite realizar en el sistema el usuario.

Por supuesto el sistema cuenta con una distribución inteligente de las secciones, botones, herramientas de tal forma que se cree una experiencia de navegación satisfactoria.

El sistema cuenta con usuarios administradores con el fin de asegurar que los cambios importantes para el funcionamiento del sistema puedan ser correctamente ejecutados.

Al solicitarse cualquier dato debidamente almacenado, el sistema mostrará dichos datos para una fácil visualización y manipulación correcta de los mismos.

Requerimientos del usuario.

Los requerimientos del usuario juegan un papel muy fundamental en el sistema, ya que se describen algunos de los requisitos que el usuario debe tener y que aporte a una correcta gestión del sistema. Algunos de los requerimientos pueden ser:

- El usuario debe ser capaz de buscar cada una de los módulos que conforman el sistema en general.
- El sistema proporciona vistas con los respectivos datos correctamente almacenados para que los usuarios puedan visualizar de una manera ordenada cada uno de los datos. Es decir que el usuario tiene la responsabilidad de gestionarlos de acuerdo a los permisos que éste tenga.
- El usuario debe de velar por la correcta utilización del sistema y orden de los datos

- El usuario debe utilizar la información (de cada los módulos que tenga en sus manos) con la correcta precaución debido a que son datos que están bajo la cobertura de la empresa y cualquier alteración de los datos puede llegar a arrastrar una mala gestión del sistema en general.
- Cualquier inconveniente con sus datos personales registrados en el sistema, debe comunicarlo inmediatamente con los administradores del sistema.

Como parte funcional del sistema y es quien precisamente se encarga de ello: el usuario debe tener equipo informático adecuado para la ejecución del sistema y es importante también que el usuario utilice de manera adecuada las herramientas que puedan llegar a ser proporcionadas por la empresa.

Herramientas a utilizar para el desarrollo del proyecto

Laragon



Ilustración 1Laragon

Laragon: Utilizaremos laragon para la ejecución de nuestro proyecto ya que laragon posee una tecnología en la cual crea fácilmente host virtuales de igual manera crea proyectos nuevos en laravel y trae muchas herramientas más en la cual es muy fácil el uso de nuestras bases de datos y podemos emigrar phpmyadmin en laragon sin ningún problema para llevar un manejo mejor de dichas versiones.

Xampp



Ilustración 2Xampp

XAMPP es un paquete de pila de soluciones de servidor web multiplataforma de código abierto y gratuito desarrollado por Apache Friends, que consta principalmente del servidor HTTP Apache, la base de datos MariaDB e intérpretes para scripts escritos en los lenguajes de programación PHP y Perl.

Laravel 7



Ilustración 3Laravel

Laravel: Utilizaremos el framework laravel en su versión 7 para la creación de nuestro proyecto ya que dicho framework de código abierto para desarrollar aplicaciones y servicios web con PHP, su filosofía es desarrollar código PHP de forma elegante y simple evitando el código espagueti

Phpmyadmin



Ilustración 4Phpmyadmin

Utilizaremos un gestor de bases de datos en la cual nos servirá manejar de una manera más fácil nuestra base de datos y nuestras tablas, phpmyadmin es una herramienta escrita en PHP con la intención de manejar la administración de mysql a través de páginas web, utilizando un navegador web.

Visual Studio Code



Ilustración 5 Visual Studio Code

Visual Studio Code: Utilizaremos un editor de texto en este caso visual studio code para llevar el control de nuestro código fuente de una forma ordenada y para poder llevar el código de una forma más elegante.

Livewire

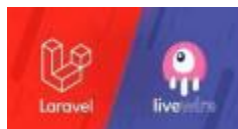


Ilustración 6 Livewire

Livewire: es una biblioteca que se implementará en el proyecto, que facilita la creación de interfaces modernas y dinámicas utilizando Laravel Blade como lenguaje de plantillas. Esta es una gran pila para elegir si deseas crear una aplicación que sea dinámica y reactiva, pero no se sienta cómodo saltando a un marco completo de JavaScript como Vue.js. Al usar Livewire, las rutas de su aplicación responderán con plantillas Blade típicas.

Bootstrap 4



Ilustración 7 Bootstrap

Bootstrap 4: Utilizaremos Bootstrap que es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios web y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como extensiones de JavaScript adicionales.

Modelado UML

Diagrama de Caso de Uso

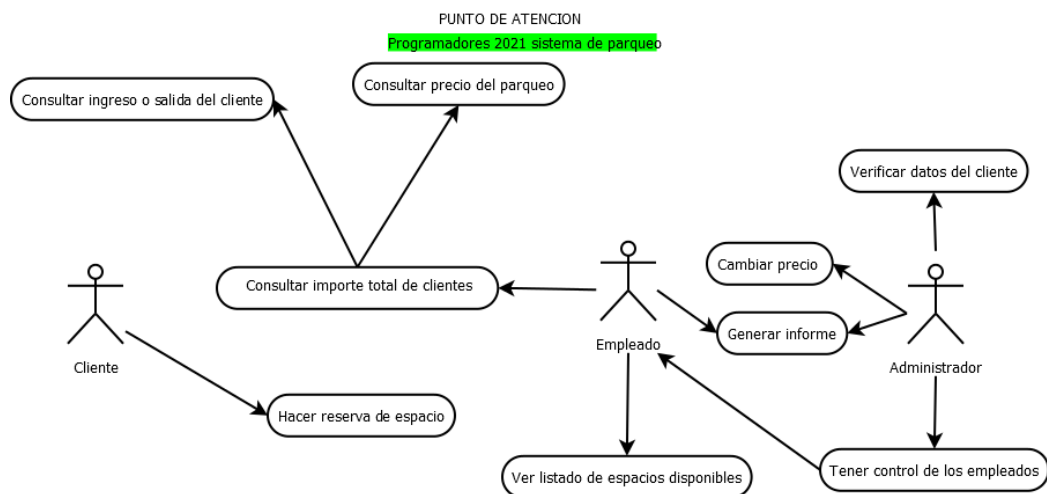


Ilustración 8 Diagrama de Caso de Uso

Diagrama de Clases

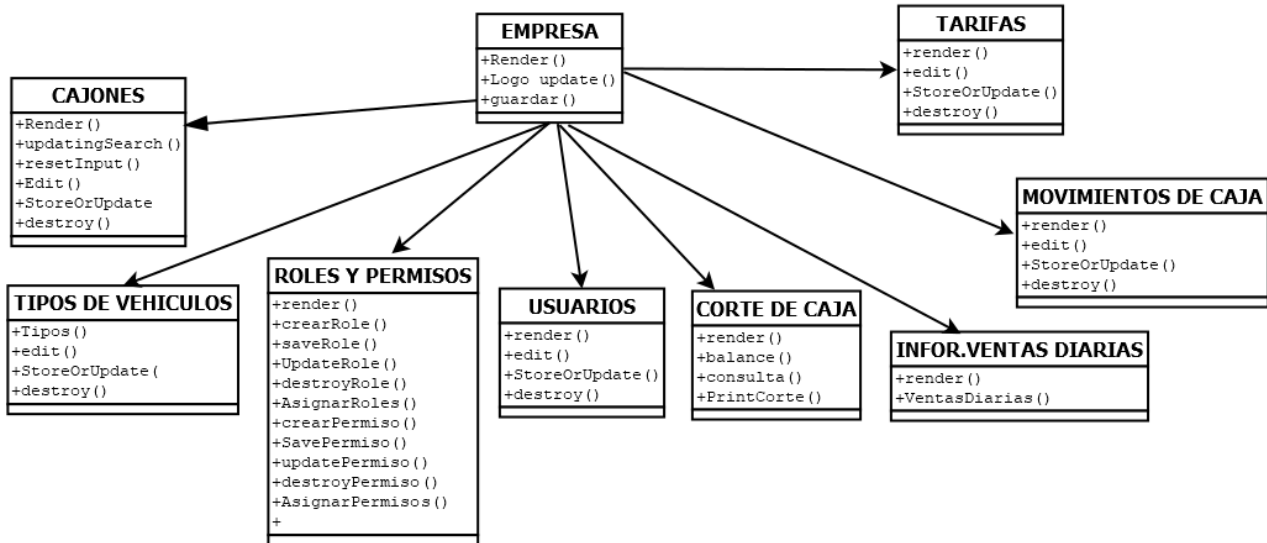


Ilustración 9 Diagrama de Clases

Modelo Físico BD

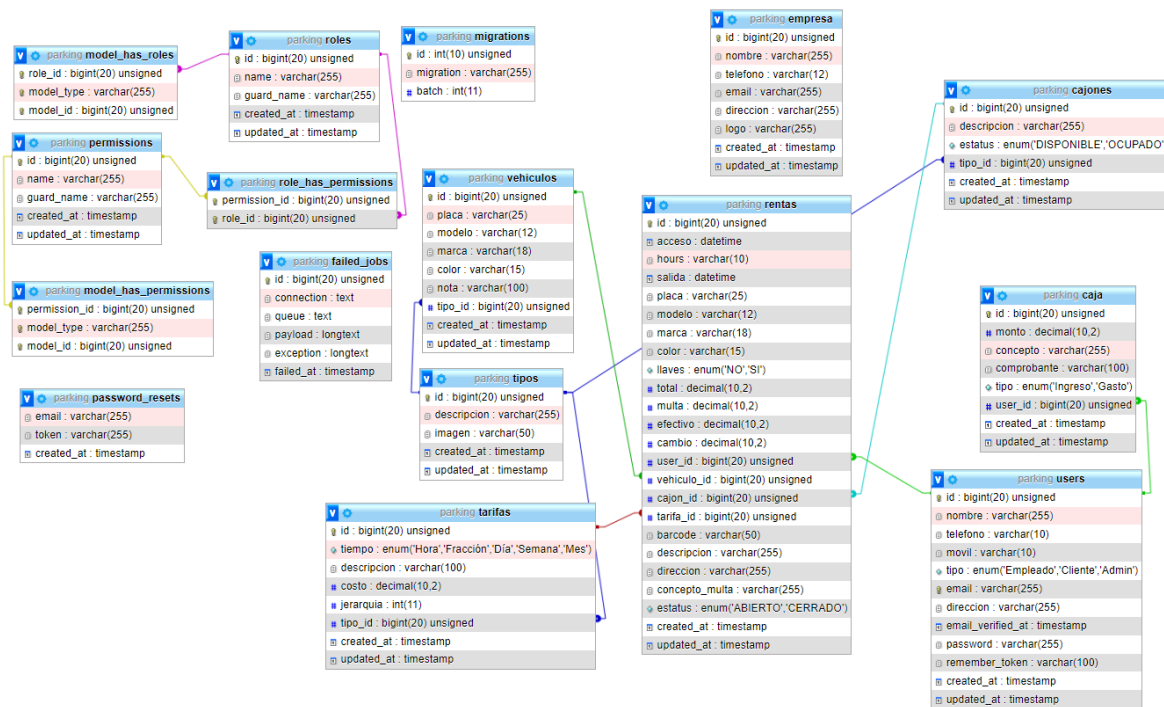


Ilustración 10 Modelo físico de la BD

Alcances y limitaciones del proyecto

Alcances

- El sistema fue realizado con el Framework de Laravel que ayuda al programador a poder realizar sistemas muy completos. Además, tienes una gran comunidad que apoya este Framework.
- Se usaron los complementos de Laravel para un mejor desarrollo del sistema.
- Tiene un área de gráficas para análisis de las operaciones.
- El sistema tiene una mejor seguridad en el envío de datos para evitar algunos ataques
- Generará reportes del de acuerdo a la sección especificada

Limitaciones

- Inicialmente no se cuenta con un hosting para ir visualizando como está quedando el sistema en modo producción. Sino que se utilizara un servidor local.
- Falta de las librerías de comandos para dinamización del proyecto.
- El sistema no genera un reporte de la estructura del parqueo
- Los registros de las ventas son pagos en efectivo, no virtual.
- Falta de registro de ventas realizadas por cada empleado o usuario.

Conclusiones

- Habiendo terminado el proyecto, en conclusión, podemos decir que de acuerdo a los objetivos mencionados al principio se logró implementar satisfactoriamente la instalación y configuración del “Sistema de Gestión de Parqueo (SGP)” esto ofrece muchas ventajas como llevar un mejor control de los diferentes empleados, Administradores, costos, tipos de autos, caja, tickets etc.
- Podemos decir que, en resumen, el problema o la necesidad que resolverá el “Sistema de Gestión de Parqueo (SGP)” es agilizar y optimizar los procesos de gestión, accediendo a información inmediata alojada en el sistema. Al tener un mejor control de ingreso de vehículos y establecer los costos correctos dependiendo el tiempo de estacionamiento gracias a dicho sistema.
- Para la realización del sistema en general se utilizó el framework Laravel el cual es de código abierto para desarrollar aplicaciones y servicios web con PHP 5, PHP 7 y PHP 8. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti".
- Se realizó una documentación donde se reflejan los pasos a ejecutar para la funcionalidad del proyecto, para su mayor comprensión, así como también se mencionan las técnicas de programación aplicadas como: MVC y las herramientas de desarrollo utilizadas como PHP, XAMPP, LARAGON, BOOSTSTRAP, PHPMYADMIN, LIVEWIRE y LARAVEL 7.

Recomendaciones

- El Sistema de Gestión de Parqueo (SGP) requiere de la utilización de los módulos como: Empresa, cajones, Tipos de Vehículos, Tarifa, Movimientos de caja, Roles y Permisos etc. Es necesario que almacene los datos más importantes de los administradores y empleados; para la correcta gestión del Sistema.
- Que los administradores que utilizan el Sistema de Gestión de Parqueo, utilicen material de apoyo o capacitaciones para los empleados para un uso y función más eficiente de dicho sistema.
- Si el Empleado detecta algún inconveniente inusual y parcial en el sistema, es necesario que lo reporte inmediatamente a los administradores.
- Uso del sistema de parking de personas capacitadas, para que no haya futuras fallas del sistema ante su mal uso y el atraso de gestión de los datos.
- Registrar correctamente los datos para prevenir situaciones negativas internas de la empresa.
- Asegurarse que el usuario obtenga su ticket.

Referencias bibliográficas

de Zúñiga, F. G. (2021, 15 junio). *Introducción a Laragon*. Blog de arsys.es.

<https://www.arsys.es/blog/programacion/introduccion-a-laragon/>

García, M. (2021, 8 marzo). *QUE ES XAMPP Y COMO PUEDO USARLO?* Nettix Perú.

<https://www.nettix.com.pe/blog/web-blog/que-es-xampp-y-como-puedo-usarlo>

D. (2020, 12 marzo). *Laravel 7 llega con mejoras en velocidad, componentes y más*.

Ubunlog. <https://ubunlog.com/laravel-7-llega-con-mejoras-en-velocidad-componentes-y-mas/>

A. (2014, 27 septiembre). *¿Qué es Bootstrap y cómo funciona en el diseño web?* Blog,

ARWEB Agencia Digital Costa Rica. <https://www.arweb.com/blog/%C2%BFque-es-bootstrap-y-como-funciona-en-el-diseno-web/>

Wikipedia contributors. (2021, 15 noviembre). *Visual Studio Code*. Wikipedia.

https://en.wikipedia.org/wiki/Visual_Studio_Code

¿Que es phpMyAdmin? (2021, 26 marzo). INC Internet Limitada.

<https://www.inc.cl/blog/hosting/que-es-phpmyadmin>

Enlace de video: <https://www.youtube.com/watch?v=xAhOxcO-A6Q>

Enlace de proyecto: <https://github.com/Esau21/Sistema.git>

Enlace de drive: <https://drive.google.com/file/d/1aB9KT3OQIynivaDEKKp7rHJgKN-cniXU/view>

Anexos

Manual de usuario

Laragon es una suite de desarrollo para PHP que funciona sobre Windows diseñado especialmente para trabajar con Laravel. Similar a otras herramientas como Xampp o Wampp, Laragon nos permite crear un entorno de desarrollo con estas características:

- Cmder (Consola para-Windows)
- Git
- Node.js
- npm
- SSH
- Putty
- PHP 7 / 5.6
- Extensiones de PHP
- xDebug
- Composer
- Apache
- MariaDB/MySQL
- phpMyAdmin
- Soporte para Laravel y Lumen
- Gestion automatica de Virtualhosts.

Puedes descargar el archivo ejecutable desde la web oficial de Laragon: <https://laragon.org/>

Una vez descargado puedes ejecutar el archivo e iniciar la instalación.

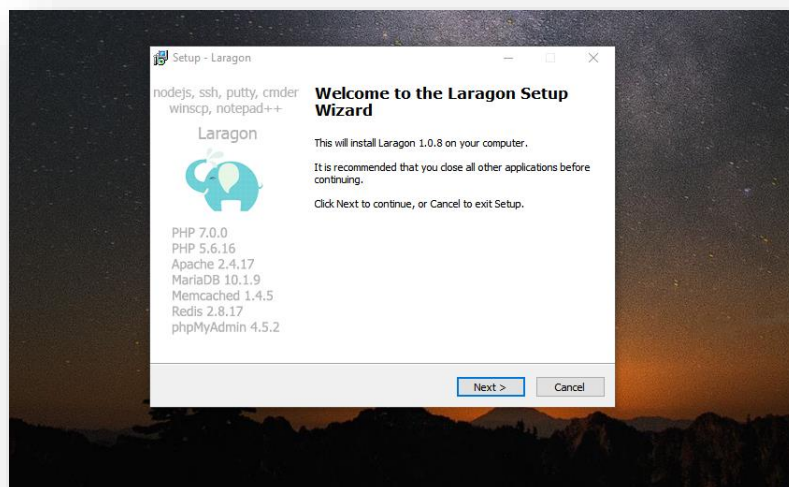


Ilustración 11 Bienvenida Laragon

Laragon posee una utilidad realmente interesante que permite crear automáticamente virtual hosts para cada proyecto, así por ejemplo en lugar de ingresar desde el navegador a un proyecto con localhost/tu-proyecto, puedes usar una url más legible como tu-proyecto.dev. Si deseas activar esta función debes indicarlo durante la instalación.

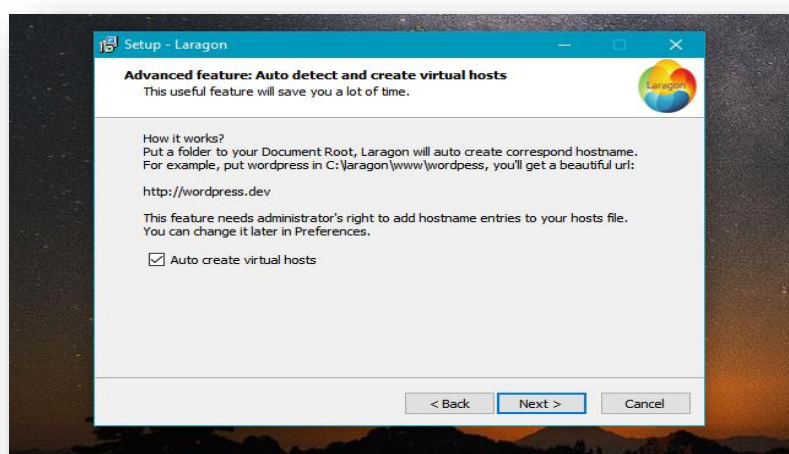


Ilustración 12 Config Laragon

Una vez finalizada la instalación puedes ejecutar Laragon.

Ahora puedes arrancar el servidor haciendo click sobre el botón «Start all».

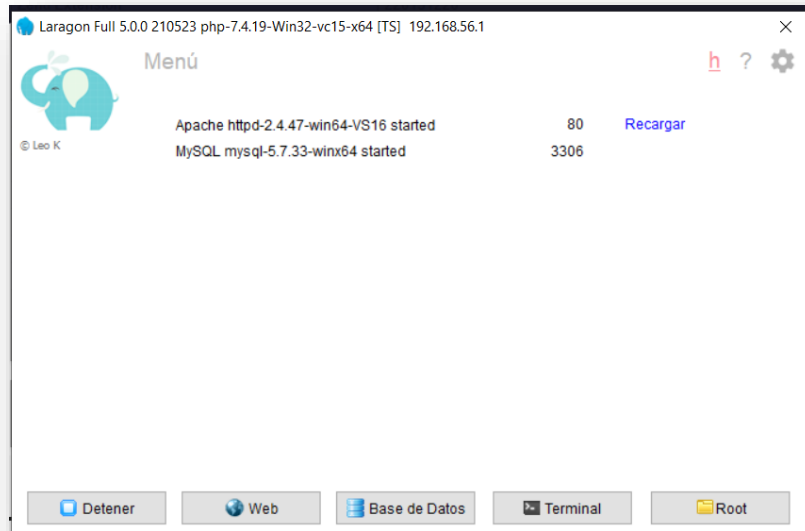


Ilustración 13 Panel de Laragon

Instalar COMPOSER

Instalar composer en sistemas Windows es muy sencillo, para esto te vas a la página oficial y descargas el archivo o Descarga aquí Composer, y elegir la descarga, como se ve en la imagen.

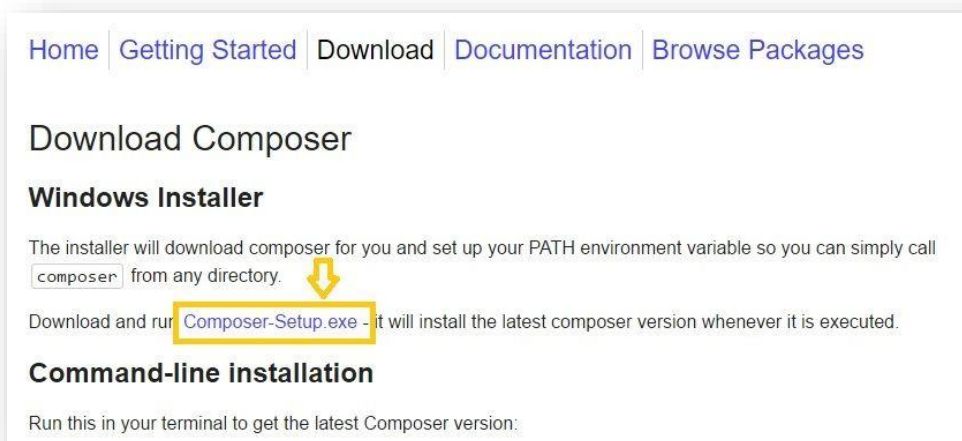


Ilustración 14 Instalador Composer

Posteriormente con un par de clicks lo instalamos:

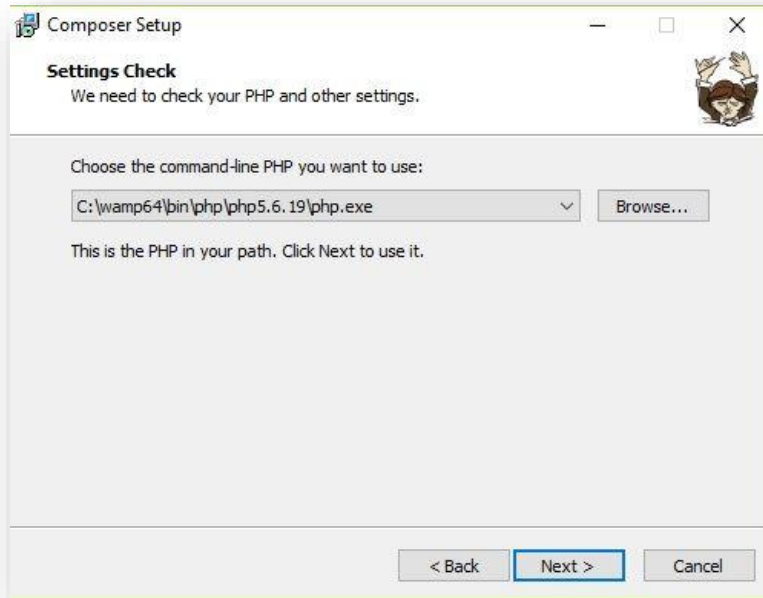


Ilustración 15 Config Composer

En la siguiente imagen dejas los campos en blanco, puesto que no vas utilizar ningún proxy y sólo das Next.

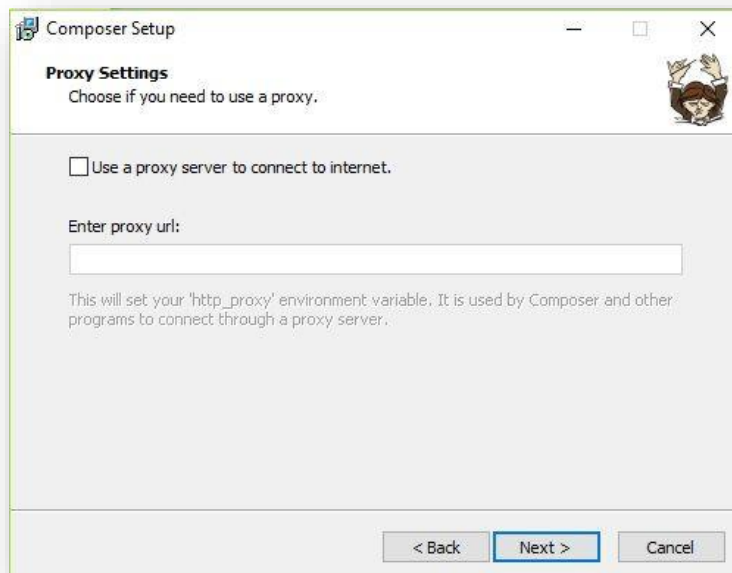


Ilustración 16 Config 2 Composer

En la siguiente pantalla que te va a aparecer le das Install.

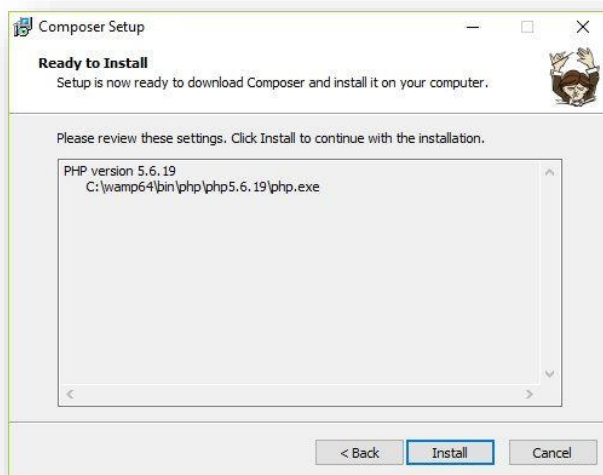


Ilustración 17 Listo para instalar

Y esta es la imagen final que te indicará que todo se instaló correctamente.



Ilustración 18 Final instalación

Finalmente, para verificar que todo está bien instalado nos vamos a la terminal (cmd) y ponemos el comando: `composer -version`. Con lo que nos mostrará lo siguiente pantalla.

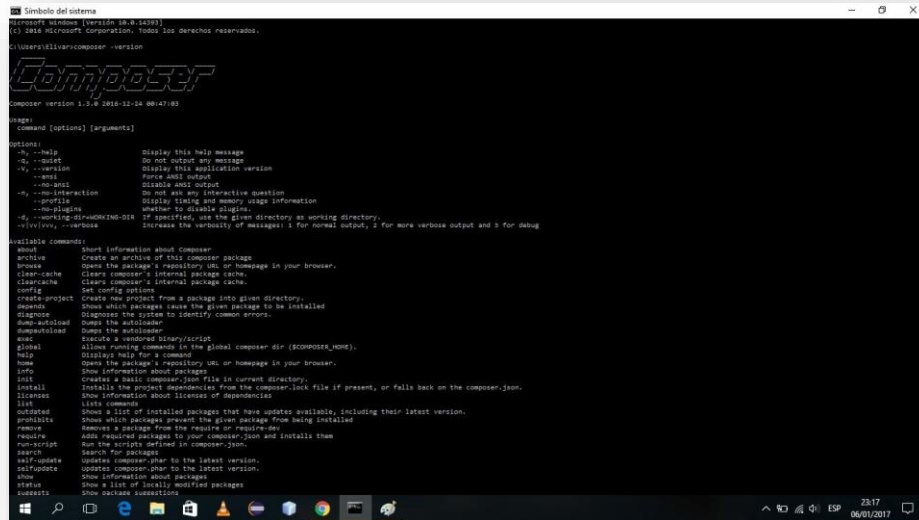


Ilustración 19 Comprobar version

Agregar PHPMYADMIN en Laragon

Descargamos el archivo oficial del sitio:

Una vez realizado eso descomprimes el archivo de Phpmysadmin descargado del sitio y mover la caperta “phpMyAdmin” a la siguiente ruta.

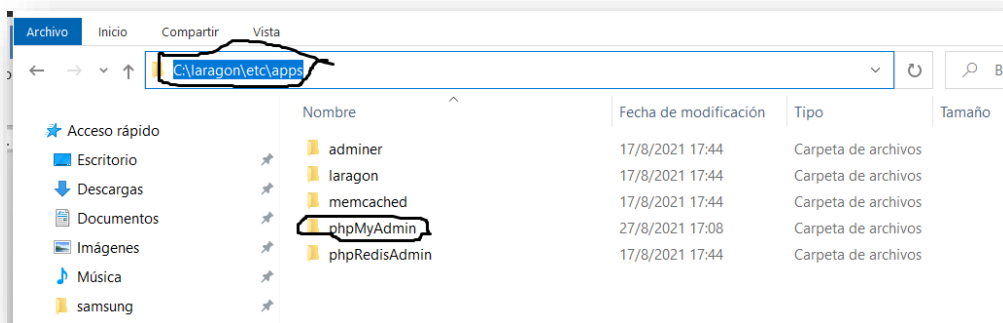


Ilustración 20 Phpmysadmin en Laragon

Ahora con el servidor encendido podremos acceder desde el navegador a la siguiente ruta:

<http://localhost/phpmyadmin/>

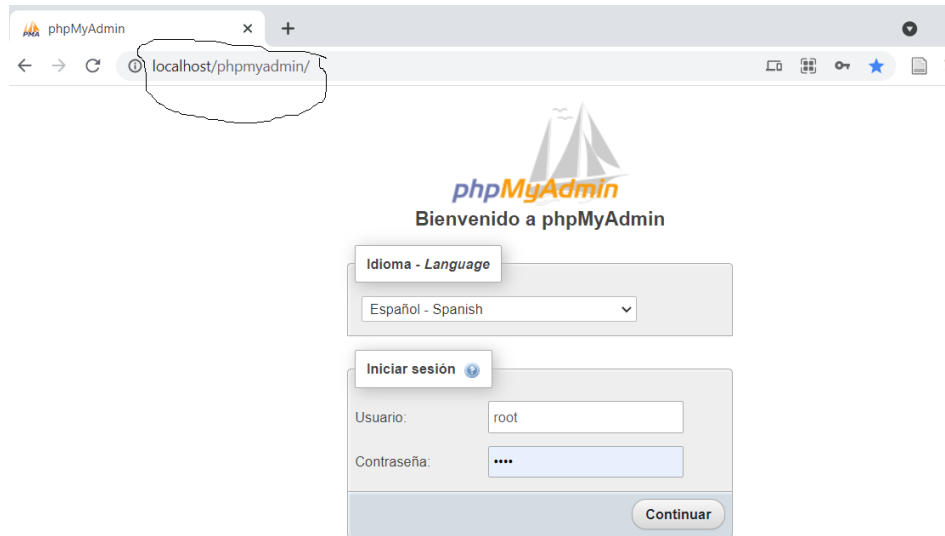


Ilustración 21 Colocar la ruta

Para entrar al gestor de base de datos por defecto el usuario es root y la contraseña es nada.

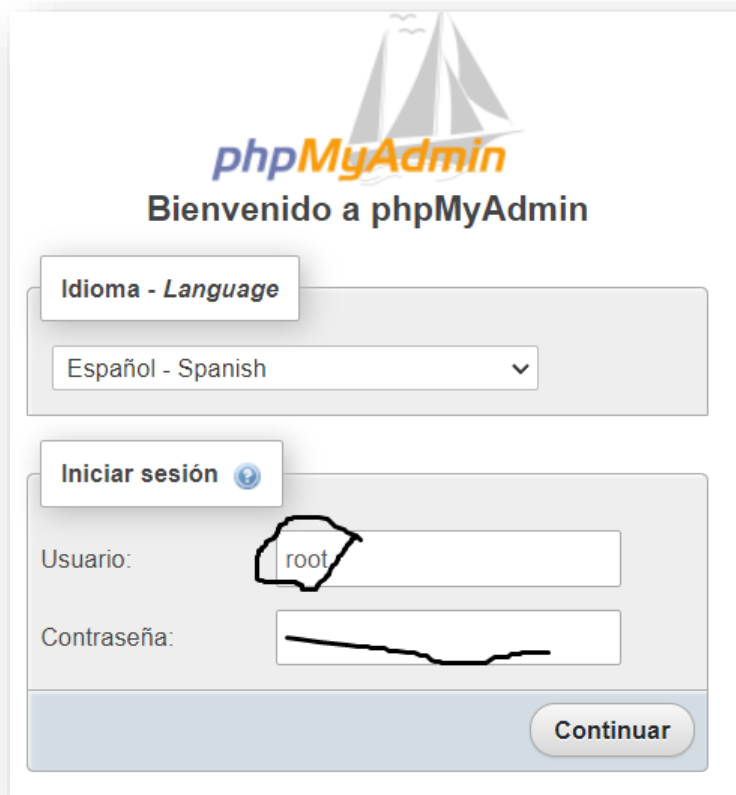


Ilustración 22 Iniciar sesión

Listo entramos a nuestro gestor de base de datos.

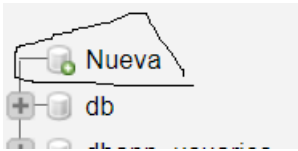


Ilustración 23 Panel administrativo phpmy..

Crear la base de datos del sistema

Una vez terminado de instalar nuestro gestor de base de datos phpmyadmin, ahora crearemos la base de datos parking.

Para eso damos en clic en nueva +.



Luego en el campo de nombre le damos el nombre “parking”.

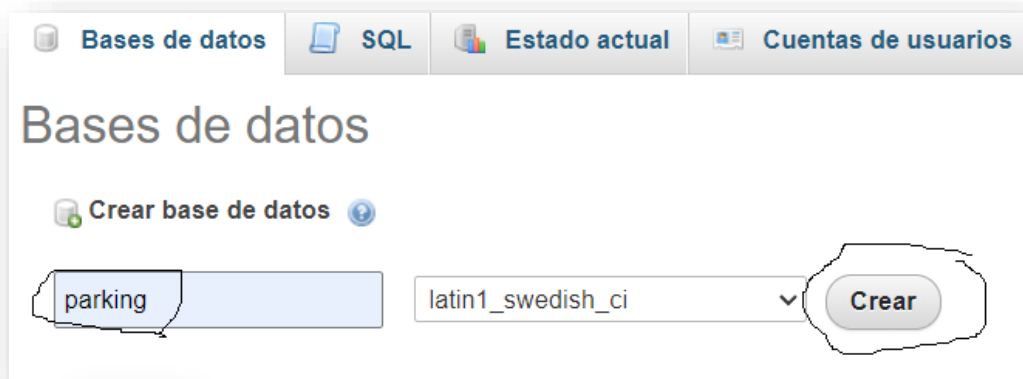


Ilustración 24 Crear base de datos

Listo con eso tendremos la base de datos creada, la cual podemos observar al lado izquierda del gestor.

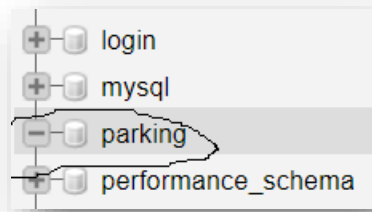


Ilustración 25 Identificar la BD

Creación de las tablas de la base de datos

Abrimos la carpeta del sistema y damos clic derecho en la parte sola de la carpeta para abrir una ventana, luego buscamos la opción de terminal.



Ilustración 26 Ruta de proyecto

Agregamos el siguiente comando para crear las tablas:

`php artisan migrate`

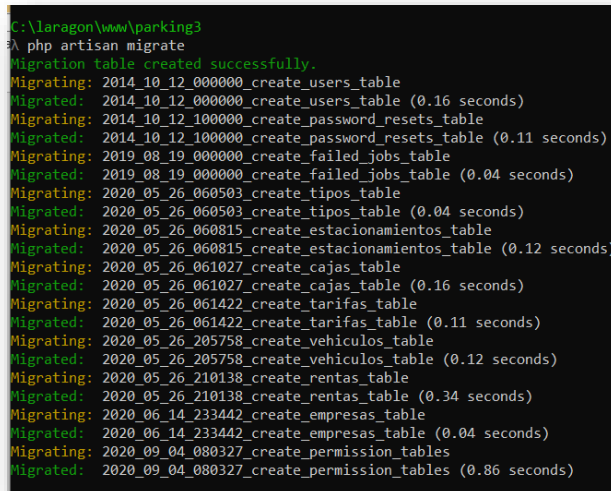


Ilustración 27 Ejecutar las migraciones

Ahora ejecutamos el siguiente comando para ingresar datos importantes al sistema.

```
php artisan db:seed
```

```
C:\laragon\www\parking3
λ php artisan db:seed
Seeding: PermissionsTableSeeder
Seeded: PermissionsTableSeeder (1.53 seconds)
Database seeding completed successfully.
```

Ilustración 28 Ejecutar los seeders

Acceder al sistema agregar información de la empresa y un nuevo admin principal.

Ejecutamos el siguiente comando para iniciar el servidor.

```
php artisan server
```

Listo este iniciado.

```
C:\laragon\www\parking3
λ php artisan serve
Laravel development server started: http://127.0.0.1:8000
[Wed Nov 17 20:31:00 2021] PHP 7.4.19 Development Server (http://127.0.0.1:8000) started
```

Ilustración 29 Levantar el servidor

Ahora vamos al navegador y escribimos 127.0.0.1:8000

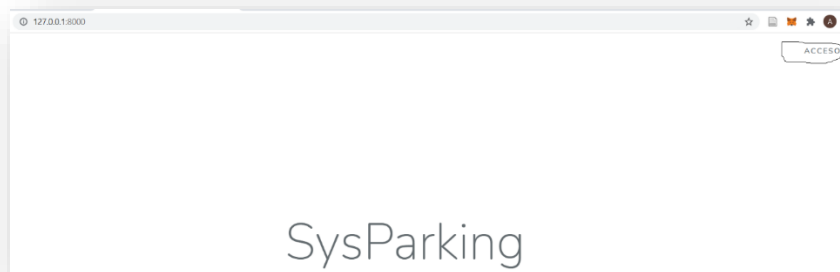


Ilustración 30 Identificar acceso al sistema

Ahora accedemos al login para entrar al sistema, esto está en la esquina superior derecha llamada “acceso”.

Los datos para entra son.

Usuario: admin@gmail.com

Contraseña: 12345678.

Este admin es de prueba y luego creamos uno nuevo para eliminar este.



The screenshot shows a login form with the following elements:

- Title: Acceso al Sistema
- EMAIL field: admin@gmail.com
- CONTRASEÑA field: 12345678
- Ingresar button

Ilustración 31 Iniciar sesión

Ingresamos y estaremos en el panel de administrador.

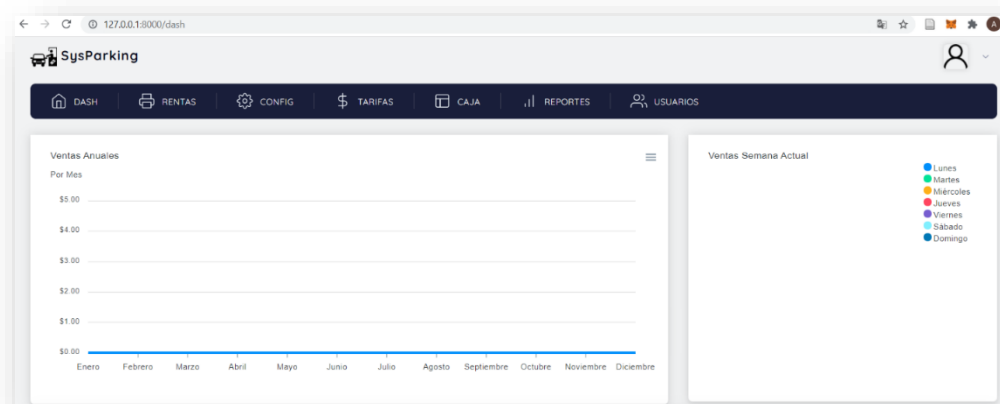
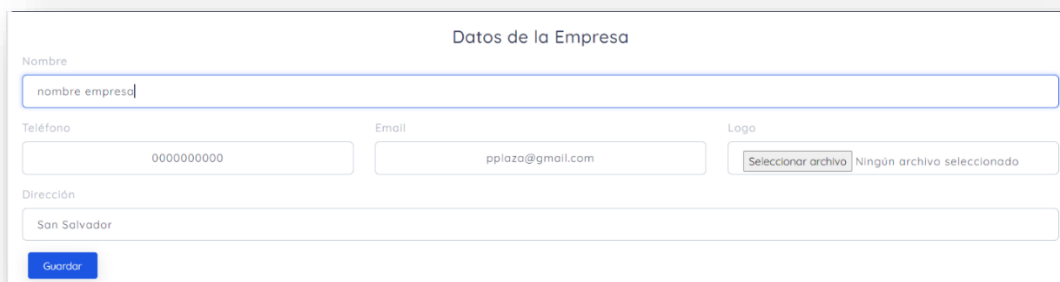


Ilustración 32 Panel de Gestión

Ahora vamos agregar datos de la empresa, esto está en la parte de CONFIG y luego

EMPRESA.



The screenshot shows a web form titled "Datos de la Empresa". It contains several input fields: "Nombre" with the placeholder "nombre empresa"; "Teléfono" with the value "000000000"; "Email" with the value "pploza@gmail.com"; "Logo" with a "Seleccionar archivo" button and the text "Ningún archivo seleccionado"; and "Dirección" with the value "San Salvador". A blue "Guardar" button is located at the bottom left of the form.

Ilustración 33 Datos de la empresa

Rellenamos los campos y aguardamos.

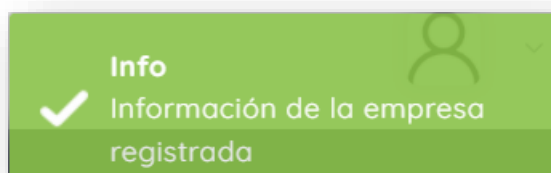


Ilustración 34 Confirmación de registro

Creamos un nuevo administrador en la parte de "USUARIOS".



The screenshot shows a web form titled "Gestionar Usuario". It contains several input fields: "Nombre" with the placeholder "nombre"; "Teléfono" with the placeholder "10 dígitos"; "Movil" with the placeholder "10 dígitos"; "Email" with the value "admin@gmail.com"; "Tipo" with a dropdown menu showing "Elegir"; "Password" with a masked field "*****"; and "Dirección" with the placeholder "dirección...". At the bottom left, there are two buttons: "Regresar" and "Guardar".

Ilustración 35 Gestionar usuario

Ahora ingresamos los datos de este administrador.

Gestionar Usuario

Nombre: Teléfono: Movil:

Email: Tipo: Password:

Dirección:

Ilustración 36 Gestionar usuario 2

Aguardamos y asignamos sus respectivos roles esto es en la pestaña CONFIG y en ROLES Y PERMISOS.

LISTADO DE ROLES

Elegir Usuario:

DESCRIPCIÓN	USUARIOS CON EL ROLE	ACCIONES	
Admin	2		<input type="checkbox"/> Asignar
Empleado	1		<input type="checkbox"/> Asignar

Ilustración 37 Roles y permisos

Elegimos el usuario que creamos reciente en mi caso es JOSE, luego marcamos la casilla ADMIN.

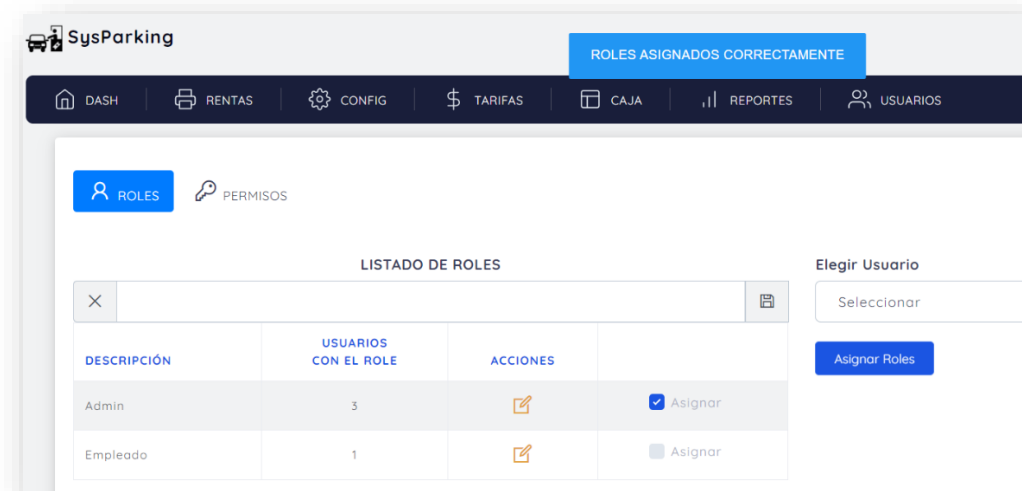
LISTADO DE ROLES

Elegir Usuario:

DESCRIPCIÓN	USUARIOS CON EL ROLE	ACCIONES	
Admin	2		<input checked="" type="checkbox"/> Asignar
Empleado	1		<input type="checkbox"/> Asignar

Ilustración 38 Roles y permisos 2

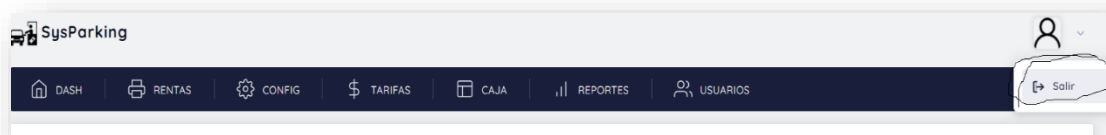
Por último, damos al botón “Asignar Roles”.



*Ilustración 39*Asignar roles

Para terminar, borramos el usuario admin que se crea por defecto al inicia el sistema.

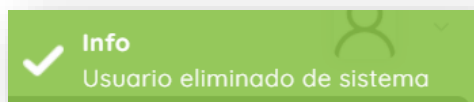
Ahora salimos del sistema para iniciar sesión con el nuevo que creamos.



*Ilustración 40*Cerrar sesión

Nos loguemos con el nuevo admin y su contraseña.

Regresamos la parte de USUARIOS y damos en eliminar, para luego por último aceptar.



*Ilustración 41*Eliminacion de usuario

Con ello terminamos la instalación del sistema.

Diagrama de actividades



Ilustración 42 Diagrama de actividades