

# **PROYECTO DE DESARROLLO DE APLICACIONES WEB**

**WILSON SOTO**  
(docente)

**EDUWIN ROJAS SUÁREZ**  
**EVELIN MIRANDA BERNAL**  
**JAVIER DARIO URREGO BLANCO**  
(estudiantes)

**UNIVERSIDAD POLITÉCNICO GRANCOLOMBIANO**  
**TECNOLOGÍA EN DESARROLLO DE SOFTWARE**  
**DESARROLLO WEB**  
**SEPTIEMBRE 06, 2022**

## 1. Identificación de requisitos representados en historias de usuario.

### **Historia de usuario 1 – Crear Página de autenticación**

Como: Cliente

Quiero: ingresar mi usuario y contraseña en la página inicio

Para: poder tener acceso a la información almacenada en la aplicación Web.

#### **Criterios de aceptación**

- Utilizar la biblioteca Bootstrap.
- Debe tener la opción de ingresar el usuario.
- Debe tener la opción de ingresar la contraseña.
- Debe tener la opción de iniciar sesión.
- Solo se permitirá acceder a los usuarios registrados.
- Validar que la contraseña sea correcta.

#### **Tareas**

- Crear un proyecto en Angular llamado “TransportePoli”
- Configurar la biblioteca Bootstrap en el proyecto “TransportePoli”
- Crear un componente llamado “login”
- Dentro del componente “login” incluir un formulario llamado “loginForm”
- Crear método para invocar el servicio de usuario y contraseña y realizar las respectivas validaciones de usuario.

### **Historia de usuario 2 – Crear servicios para el Registro de Rutas.**

Como: Cliente

Quiero: tener la opción de generar una ruta

Para poder: ingresar, consultar, editar y eliminar la información de los buses.

#### **Criterios de aceptación**

- Debe permitir consultar una ruta registrados en la base de datos.
- Debe permitir agregar una ruta a la base de datos.
- Debe permitir editar una ruta que ya se encuentre registrado en la base de datos

#### **Tareas**

- Crear un proyecto en Spring Boot usando la capa de negocio REST llamado RutaPoli, 3 package con los siguientes nombres:
  - co.edu.poli.rutaspoli.controller
  - co.edu.poli.rutaspoli.model
  - co.edu.poli.rutaspoli.repository
- Crear en MySQL un nuevo schema de nombre rutasPoli.
- En el package co.edu.poli.rutaspoli.model crear la clase Route.
- En la clase Route invocar la anotación @Entity, en la siguiente línea de código invocar la anotación @Table(name="ruta")
- En la clase Route invocar la anotación @Id en la siguiente línea de código se debe incluir el parámetro "numero\_del\_bus" de tipo de dato String, seguido por el parámetro "fecha" de tipo String, el parámetro "hora" de tipo String, el parámetro "sillas\_vacias" de tipo int
- En la clase Route generar dos constructores, uno debe ir vacío y el otro deberá contener todos los parámetros de la clase Route.
- En la clase Route generar los métodos Getters y Setters.
- En el package co.edu.poli.rutaspoli.repository crear una interfase llamada RouteRepository
- La interface RouteRepository debe extender de JpaRepository<Route, String>
- En el package co.edu.poli.rutaspoli.controller crear la clase RouteController.
- Crear método getRoute() para invocar la anotación GetMapping para obtener la información de una ruta en la aplicación en la clase RouteController.
- Crear método saveRoute() para invocar la PostMapping para insertar rutas en la aplicación en la clase BusController.
- Crear método updateRoute() para invocar la anotación PutMapping para actualizar un buses en la aplicación en la clase RouteController.
- Crear método deleteRoute() para invocar la anotación DELETE para eliminar buses en la clase RouteController.

### **Historia de usuario 3 – Crear servicios para el Registro de Conductor.**

Como: Cliente

Quiero: crear un conductor

Para poder: ingresar, consultar y editar la información de los conductores.

#### **Criterios de aceptación**

- Debe permitir consultar un conductor(es) registrados en la base de datos.
- Debe permitir agregar un conductor(es) a la base de datos.

- Debe permitir editar un conductor que ya se encuentre registrado en la base de datos.

## Tareas

- En el package `co.edu.poli.rutaspoli.model` crear una clase llamada `Driver`
- En la clase `Driver` invocar la anotación `@Entity`, en la siguiente línea de código invocar la anotación `@Table(name="Conductor")`
- En la clase `Driver` invocar la anotación `@Id` en la siguiente línea de código se debe incluir el parámetro "identificación" de tipo de dato `String`, seguido por el parámetro "nombre" de tipo `String` y el parámetro "fecha\_ingreso"
- En la clase `Driver` generar dos constructores, uno debe ir vacío y el otro deberá contener todos los parámetros de la clase `Driver`.
- En la clase `Driver` generar los métodos `Getters` y `Setters`.
- En el package `co.edu.poli.rutaspoli.repository` crear una interfase llamada `DriverRepository`
- La interface `DriverRepository` debe extender de `JpaRepository<Driver, String>`
- En el package `co.edu.poli.rutaspoli.controller` crear la clase `DriverController`.
- Crear método `getDriver()` para invocar la anotación `GetMapping` para obtener la información de una ruta en la aplicación en la clase `DriverController`.
- Crear método `saveDriver()` para invocar la `PostMapping` para insertar rutas en la aplicación en la clase `DriverController`.
- Crear método `updateDriver()` para invocar la anotación `PutMapping` para actualizar un buses en la aplicación en la clase `DriverController`.

## Historia de usuario 4 – Crear página Registro de Rutas

Como: Cliente

Quiero: ver una página llamada "Rutas"

Para: poder ingresar y consultar la información de la ruta realizada por los buses del Politécnico Grancolombiano.

### Criterios de aceptación.

- Debe tener una lista desplegable donde se pueda seleccionar un conductor del Politécnico Grancolombiano
- Debe permitir el ingreso del número de sillas vacías del bus.
- Debe tener una tabla con cuatro columnas "Número de bus", "Conductor", "Sillas vacías" y "Fecha y hora".
- Debe tener un botón para guardar el registro de la ruta.

- Se deben visualizar las rutas ingresadas

### **Tareas**

- Crear un componente llamado “rutas”.
- Crear un formulario llamado “formularioRuta” dentro del componente “rutas”.
- Dentro del componente “rutas” añadir el elemento HTML <table> para poder visualizar las rutas que se vayan registrando.
- Crear método para invocar el servicio de guardar los datos de la ruta al ser registrada

## **Historia de usuario 5– Crear Página de registro de conductores**

Como: Cliente

Quiero: registrar a los conductores que trabajan en el Politecnico Grancolombiano

Para: poder tener control sobre las rutas y la dinámica e transporte del politecnico grancolombiano.

### **Criterios de aceptación**

- Utilizar la biblioteca Bootstrap.
- Debe tener la opción de ingresar el nombre del conductor.
- Debe tener la opción de ingresar la identificación del conductor
- Debe tener la opción de ingresar la fecha de ingreso del conductor.
- Debe tener la opción de registrar.

### **Tareas**

- Crear un componente llamado “conductores”
- Dentro del componente “conductores” incluir un formulario llamado “conductoresForm”
- Crear método para invocar el servicio de guardar los datos del conductor al ser registrados

## **2. Definición de mockups.**

### **Login**

Email address

Password

☒ Remember me [Forgot password?](#)

SIGN IN

## Registro de conductores

INGRESAR CONDUCTOR

Nombre

Identificacion

Fecha de ingreso a la universidad

Guardar

## Registro de bus

← → ↻ localhost:4200

TRANSPORTE POLITECNICO GRANCOLOMBIANO.

REGISTRO DE BUSES

BUSES REGISTRADOS

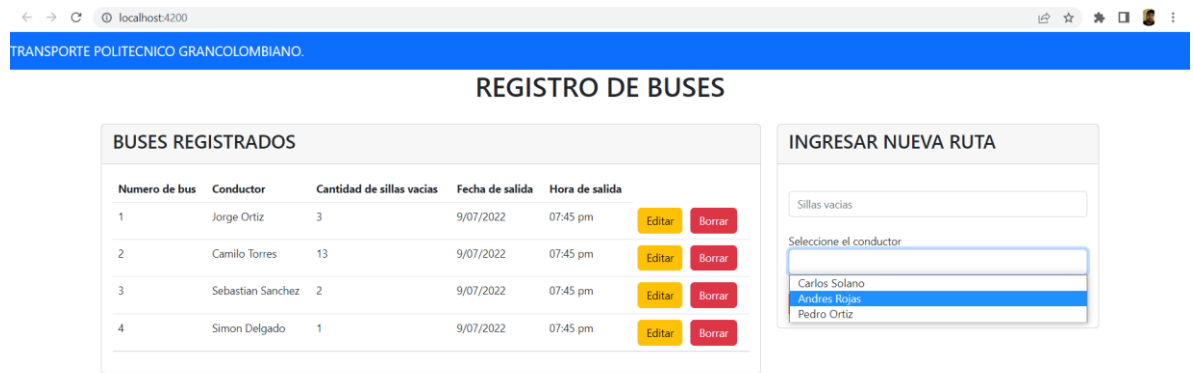
Numero de bus	Conductor	Cantidad de sillas vacias	Fecha de salida	Hora de salida		
1	Jorge Ortiz	3	9/07/2022	07:45 pm	Editar	Borrar
2	Camilo Torres	13	9/07/2022	07:45 pm	Editar	Borrar
3	Sebastian Sanchez	2	9/07/2022	07:45 pm	Editar	Borrar
4	Simon Delgado	1	9/07/2022	07:45 pm	Editar	Borrar

INGRESAR NUEVA RUTA

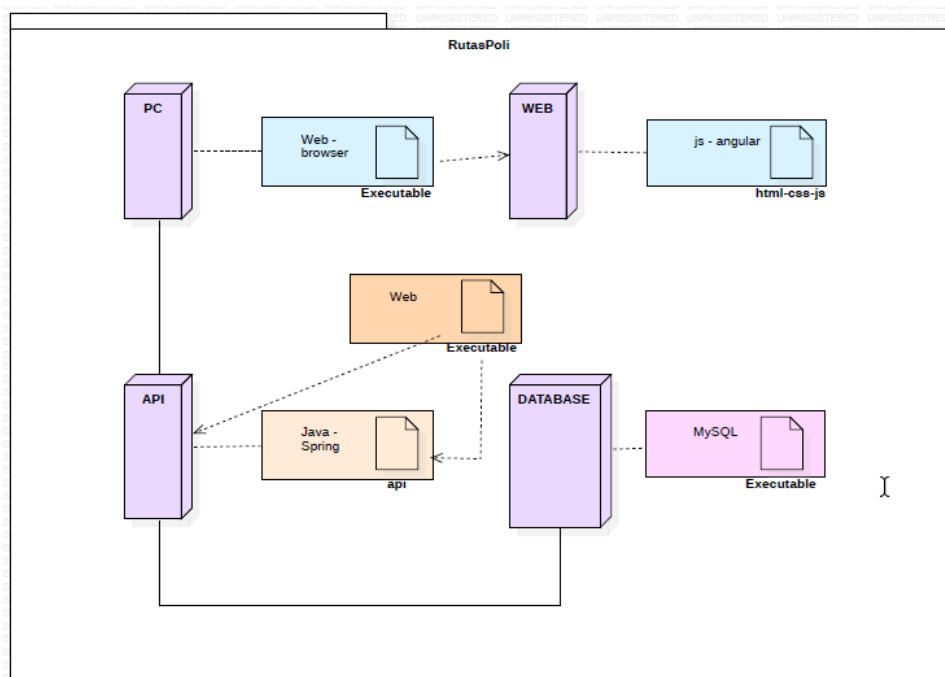
Sillas vacias

Seleccione el conductor

Guardar



### 3. Diagrama de despliegue.



### 4. Identificación de las herramientas seleccionadas para cada capa con una breve descripción.

**Presentación:** Angular, es un framework opensource desarrollado por Google para facilitar la creación y programación de aplicaciones web de una sola página. Separando el frontend del backend en la aplicación, ofrece plantillas de diseño como Bootstrap que facilita la construcción de toda la parte visual (frontend) que verá el usuario, además evita escribir código repetitivo y mantiene todo más

ordenado gracias a su patrón MVC (Modelo-Vista-Controlador) asegurando los desarrollos con rapidez, a la vez que posibilita modificaciones y actualizaciones.

**Negocio:** Spring Boot, es una tecnología que nos permite crear aplicaciones, con , proporcionando una serie de dependencias que se pueden añadir a los proyectos dependiendo de lo que necesitemos minimizando la necesidad de configuración al desarrollar permitiendo desplegar nuestros servicios, esto nos podemos olvidar de la arquitectura y enfocarnos únicamente en desarrollo.

**Persistencia:** MySQL, es el sistema de gestión de bases de datos relacional más extendido en la actualidad al estar basada en código abierto, trabaja con bases de datos relacionales, es decir, utiliza tablas múltiples que se interconectan entre sí para almacenar la información y organizarla correctamente. Además, basa su funcionamiento en un modelo cliente y servidor. Es decir, clientes y servidores se comunican entre sí de manera diferenciada para un mejor rendimiento. Además, presenta facilidad a la hora de configurarlo, es compatible con la mayoría de las plataformas informáticas.

## Referencias

*Qué es MySQL: Características y ventajas.* (2019, septiembre 24). Openwebinars.net.

<https://openwebinars.net/blog/que-es-mysql/>

Gonzalez, M. V. (2021, marzo 17). *¿Qué es Spring Boot?* Codmind. <https://blog.codmind.com/que-es-spring-boot/>

Devs, Q. (2019, septiembre 16). *¿Qué es Angular y para qué sirve?* Quality Devs.

<https://www.qualitydevs.com/2019/09/16/que-es-angular-y-para-que-sirve/>