

# Manual técnico

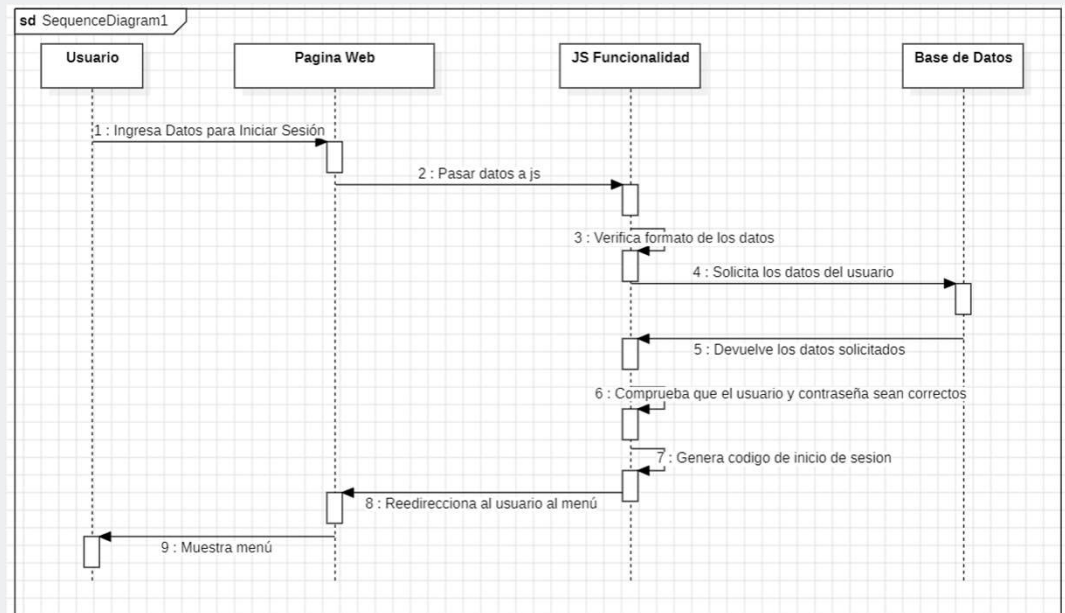


# Patrón de diseños utilizados:

- Singleton: En la aplicación desarrollada se inicializa Firebase utilizando la configuración proporcionada. Esto sigue el patrón Singleton, donde se garantiza que solo haya una instancia de Firebase inicializada con la configuración específica.
- Módulo: Se hicieron un conjunto de funciones relacionadas que funcionan juntas para realizar una tarea específica, permitiendo una mejor organización y estructura del código.
- Patrón de Observador: Esto debido a que se utilizan eventos para manejar acciones como enviar un formulario. Se agregaron oyentes event listeners para escuchar estos eventos y ejecutar funciones específicas cuando ocurren.
- Patrón de Constructor (Factory): Se hizo uso de este patrón para crear y mostrar marcadores para diferentes tipos de ubicaciones (origen, destino, paradas de autobús). Se utiliza una función para crear y configurar los marcadores con diferentes iconos y propiedades.

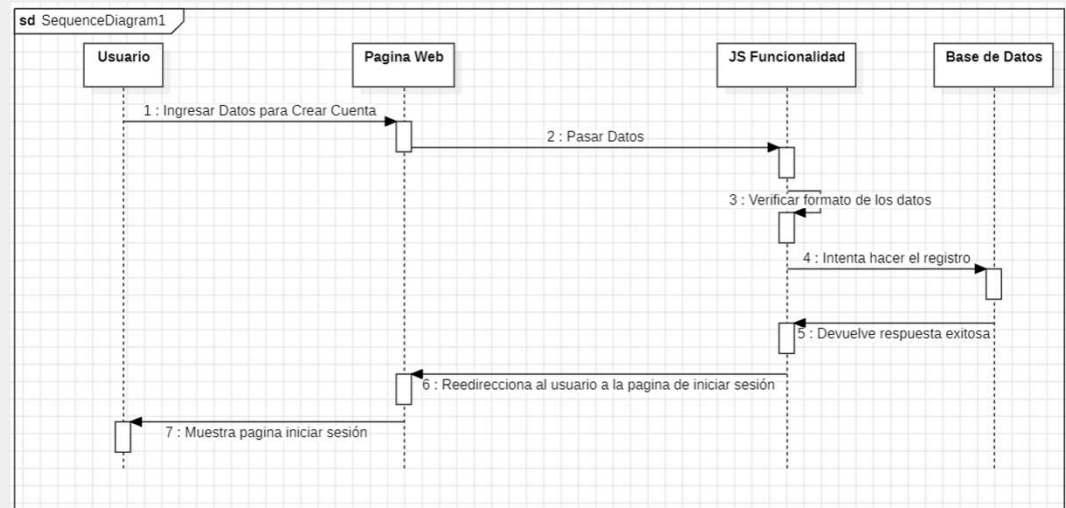
# Diagrama de secuencia para iniciar sesión

Este diagrama de secuencia representa lo que conlleva el iniciar una sesión, donde el usuario ingresa los datos de su cuenta, la página pasa esos datos al JS que le da funcionalidad a la página, verifica el formato de los datos, solicita los datos a la BD, la BD devuelve los datos solicitados, el JS comprueba que los datos coincidan, una vez validados genera un código de acceso y redirecciona la página web al menú donde la página web la encargada de mostrarla correctamente al usuario.



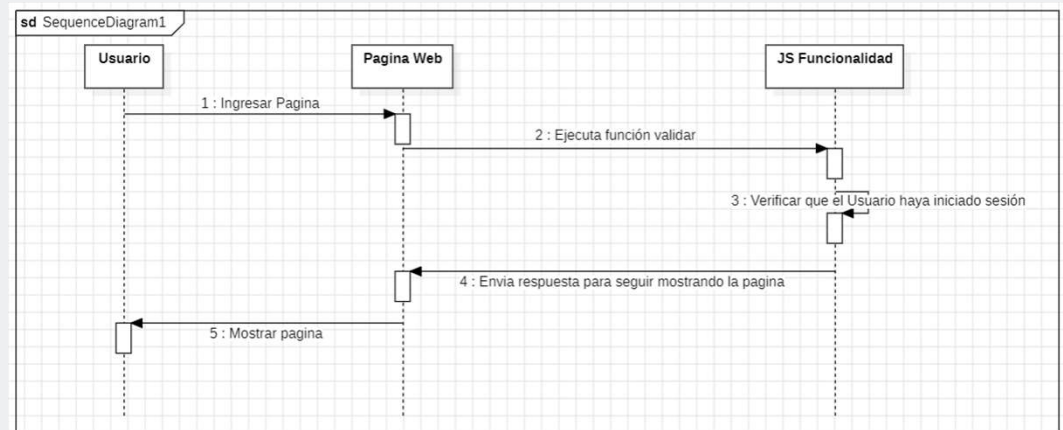
# Diagrama de secuencia para crear cuenta

Este diagrama representa los procesos que se llevan a cabo para poder crear una cuenta, el usuario ingresa los datos, la página web se encarga de pasar esos datos al JS donde se verificara el formato e intentara hacer el registro a la BD, si el registro es exitoso la BD devolverá una respuesta exitosa y el JS redireccionara al usuario a la página para iniciar sesión donde la página web será la encargada de representarlo de manera correcta.



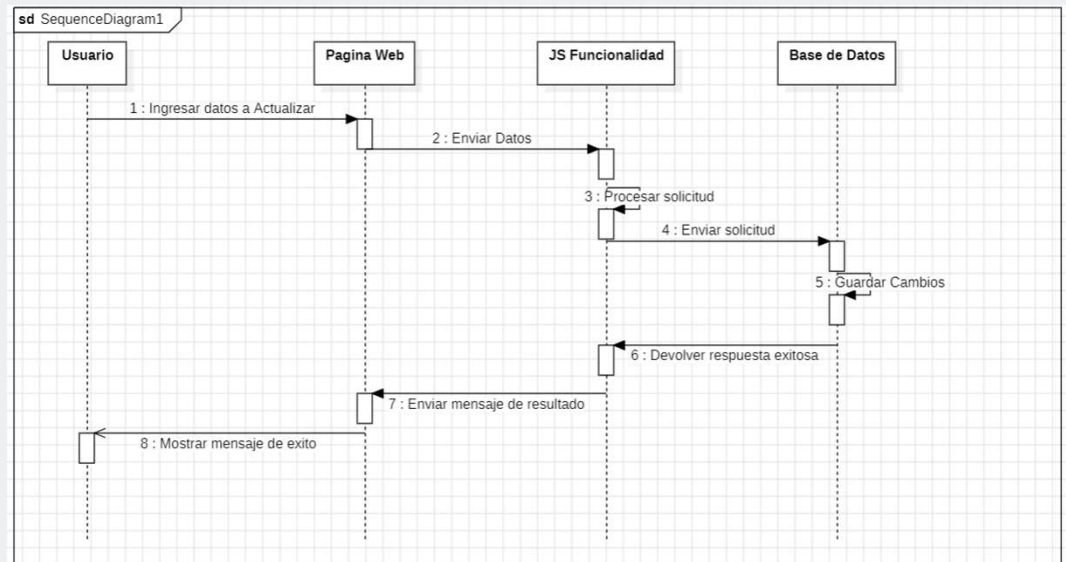
# Diagrama de secuencia para proteger paginas

Este diagrama representa los procesos que se llevan a cabo para proteger las páginas web de que se intente acceder a ellas sin haber iniciado sesión, el usuario ingresa a la página donde la página web ejecutara el JS para verificar que el usuario haya iniciado sesión, una vez se compruebe esto este permitirá a la página web seguir cargando la página web que mostrara al usuario.



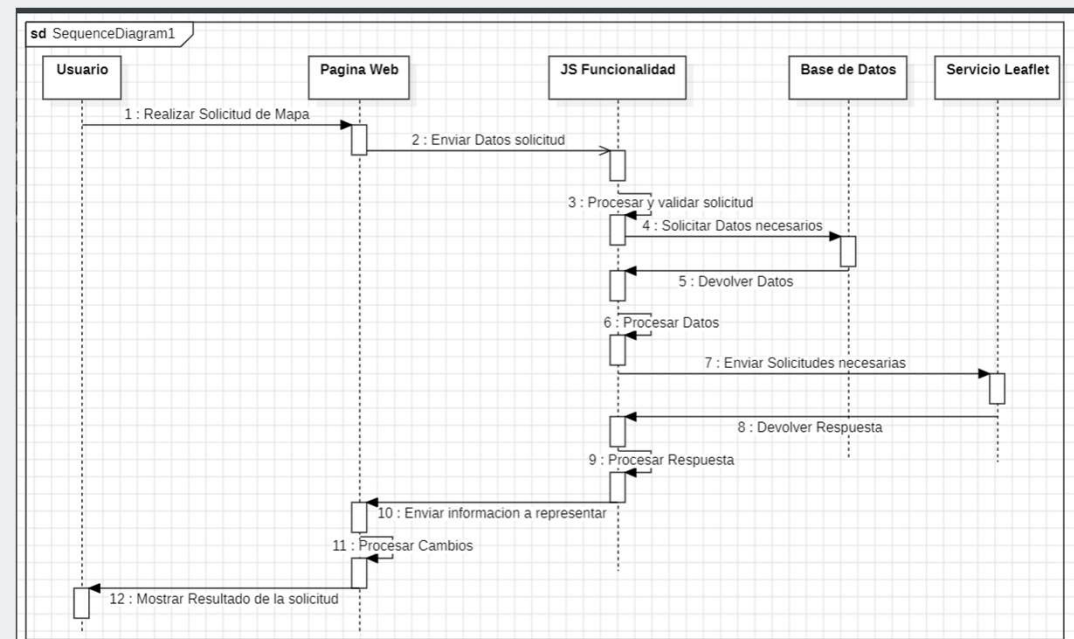
# Diagrama de secuencia para Actualizar datos

Este diagrama representa los procesos que se llevan a cabo para poder actualizar los datos, para ello el usuario ingresa los datos a actualizar, la página web enviara los datos al JS, este procesara la solicitud antes de enviarla a la BD y luego la enviara a la BD, la BD guardara los cambios y enviara una respuesta al JS, el JS enviara un mensaje con el resultado a la página Web y este mostrara un mensaje de éxito.



# Diagrama de secuencia para Funcionalidad del Mapa

Este diagrama representa los procesos que se llevan a cabo para representar correctamente las solicitudes del usuario sobre el mapa, el usuario realiza la solicitud de mapa, la página web envía los datos de la solicitud al JS, este procesara los datos y enviara una solicitud a la BD para obtener los datos necesarios, la BD devolverá los datos, el JS procesara los datos obtenidos de la BD y enviara varias solicitudes al servicio de Leaflet el cual regresara sus respuestas al JS y nuevamente procesara las respuestas obtenidas, después enviara la información a representar a la página web que será encargado de procesar los cambios y mostrar el resultado final de la solicitud del cliente.



# Diseño de la Base de Datos

Actualmente el diseño de la BD es el siguiente, donde cada clase cuenta con los atributos y el tipo de datos necesarios para enviar una respuesta con los datos necesarios para poder ser utilizados de manera correcta por el código actual de la página web.

