****

**Universidad Politécnica de Chiapas**

**DSI - PI**

José Gerardo Rodríguez Velasco 16322

Reporte de implementación API Map Matching de MapBox

# Configuración

# HTML 5

# CSS 3

# JS ECMAScript

# MapBox Access Token

# Mapbox GL JS

# Mapbox GL Draw Plugin

# Mapbox Map Matching API

# jQuery

# Sublime Text

# Introducción

# La API Mapbox Map Matching es posible de trazar rutas personalizadas permitiendo diferentes posibilidades de enrutamiento. Ajusta las coordenadas a la red de carreteras de OpenStreetMap para devolver indicaciones paso a paso para las rutas que produce.

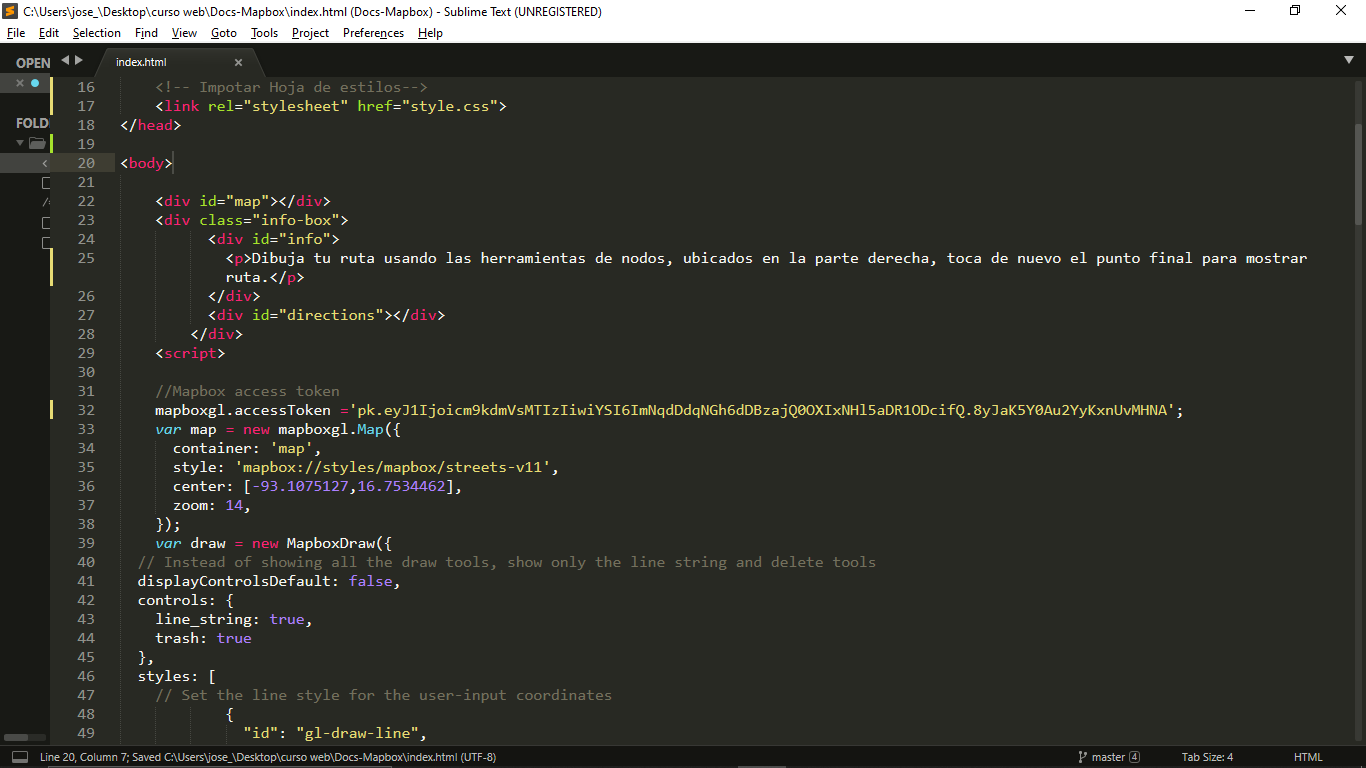
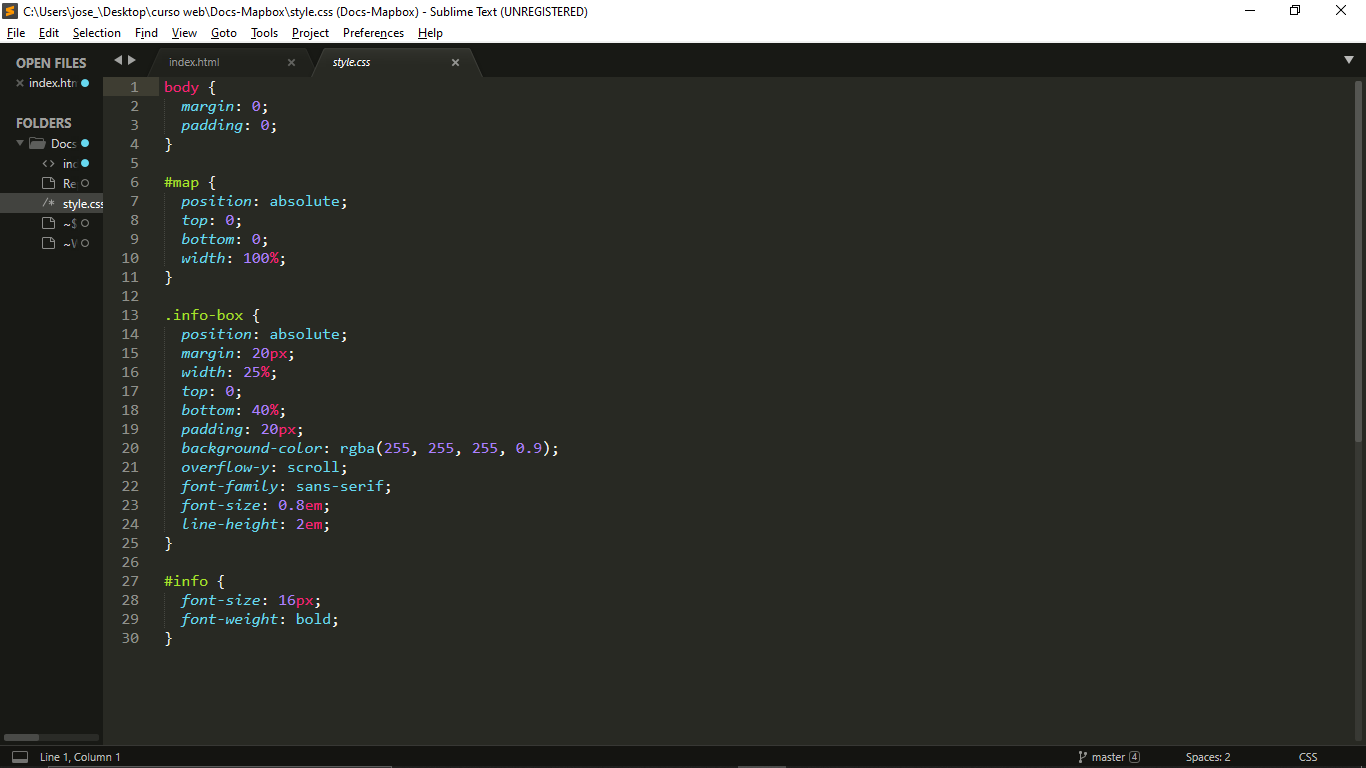
**Objetivos**

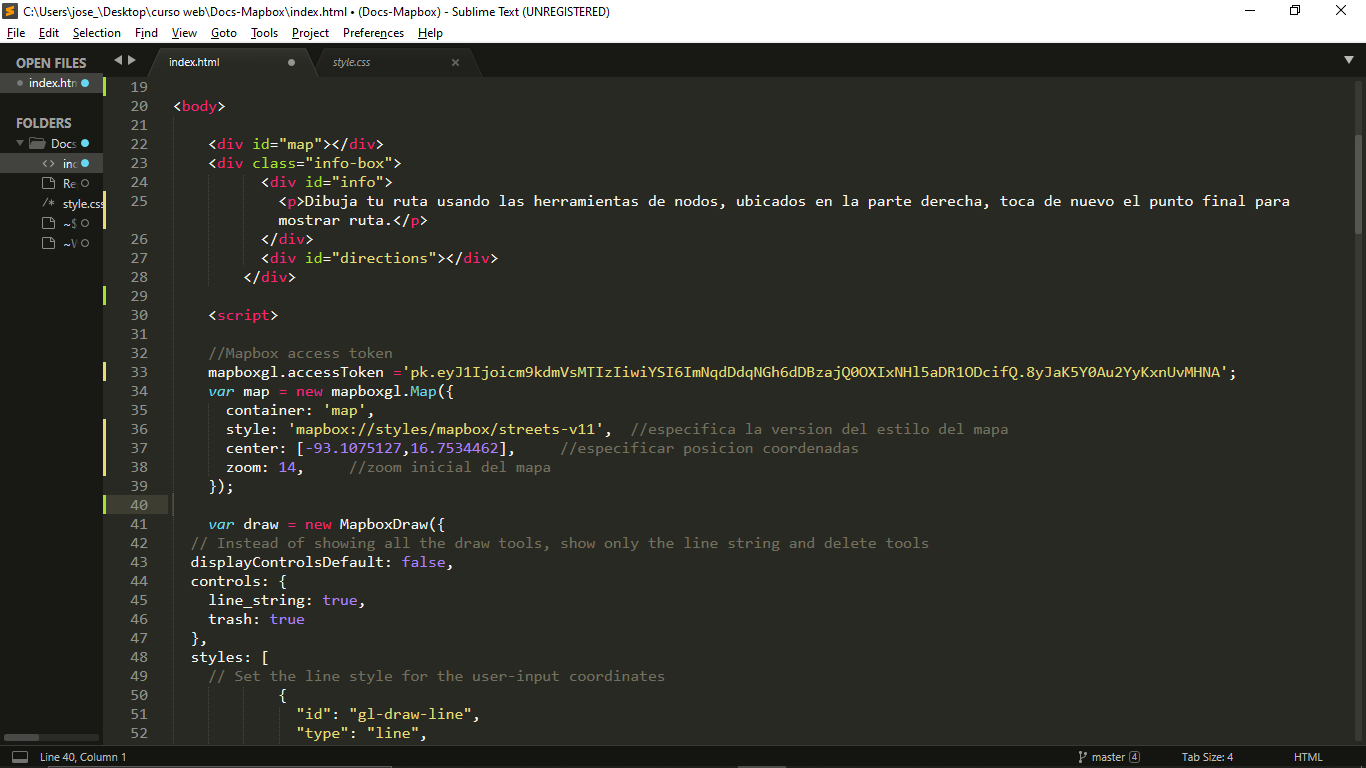
* **Objetivo principal**

Aplicación web que permitirá a sus usuarios crear rutas de entrega.

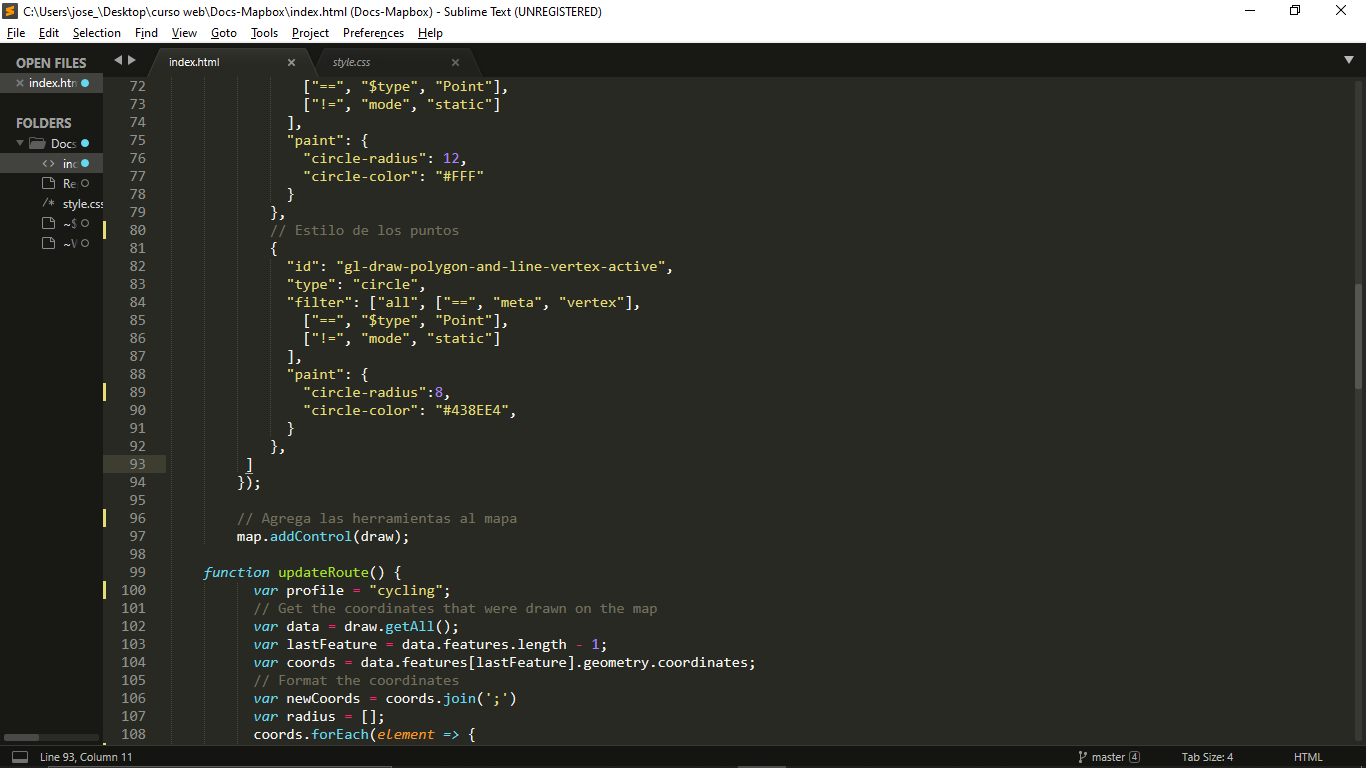
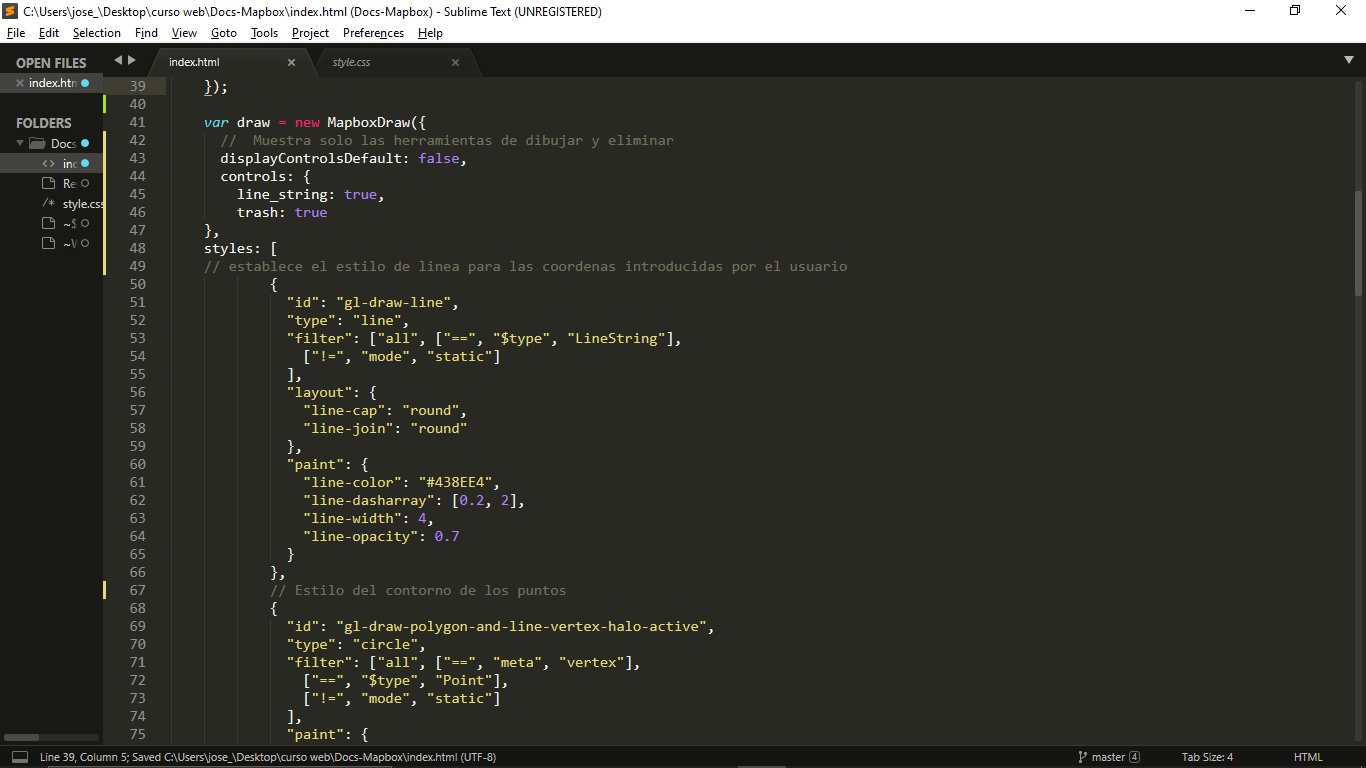
* **Objetivos específicos**
* El usuario marcará la ruta en el mapa utilizando la herramienta de Mapbox GL Draw plugin.
* La aplicación enviará coordenadas a la API Map Matching para generar indicaciones paso a paso para la nueva ruta de entrega.
* La aplicación trazará la ruta con indicaciones para la nueva ruta de entrega.

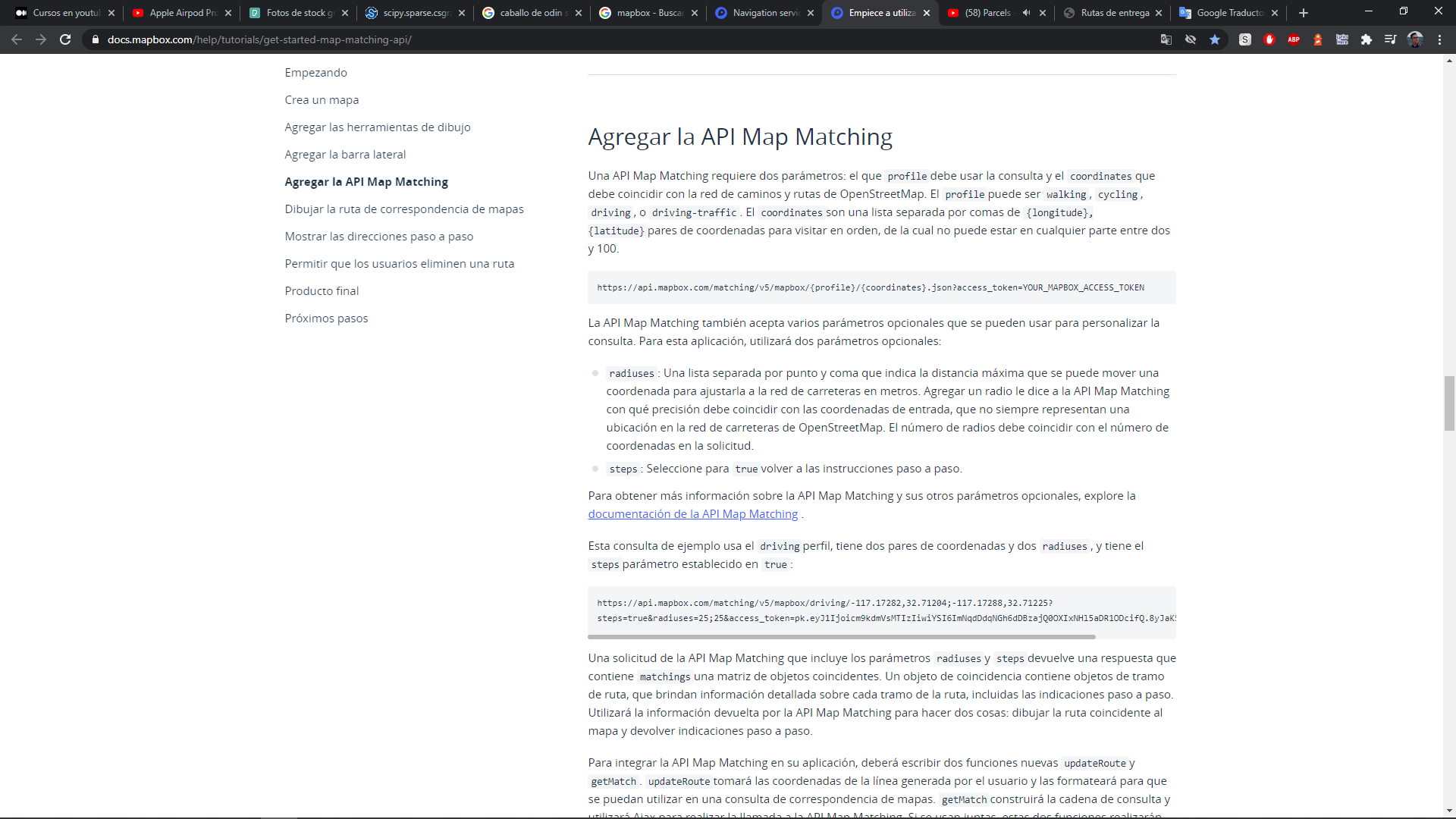
**Implementación**

1. Crear mapa y barra de indicaciones
2. Agregar herramientas de dibujo.
3. Agregar la API de Map Matching.
4. Dibujar la ruta de correspondencia de mapas.
5. Mostrar las direcciones paso a paso.
6. Agregar método eliminar ruta.
7. Implementar scripts en <head>, funcionalidad y estilos de Mapbox, jQuery que permite usar Ajax para analizar peticiones a la API Map Matching.
8. Crear contenedor del mapa y barra de indicaciones
9. Crear estilos para los contendores.
10. Crear script para el Access token, y configuración del mapa.



1. Implementar la herramienta para dibujar líneas y eliminarlas.

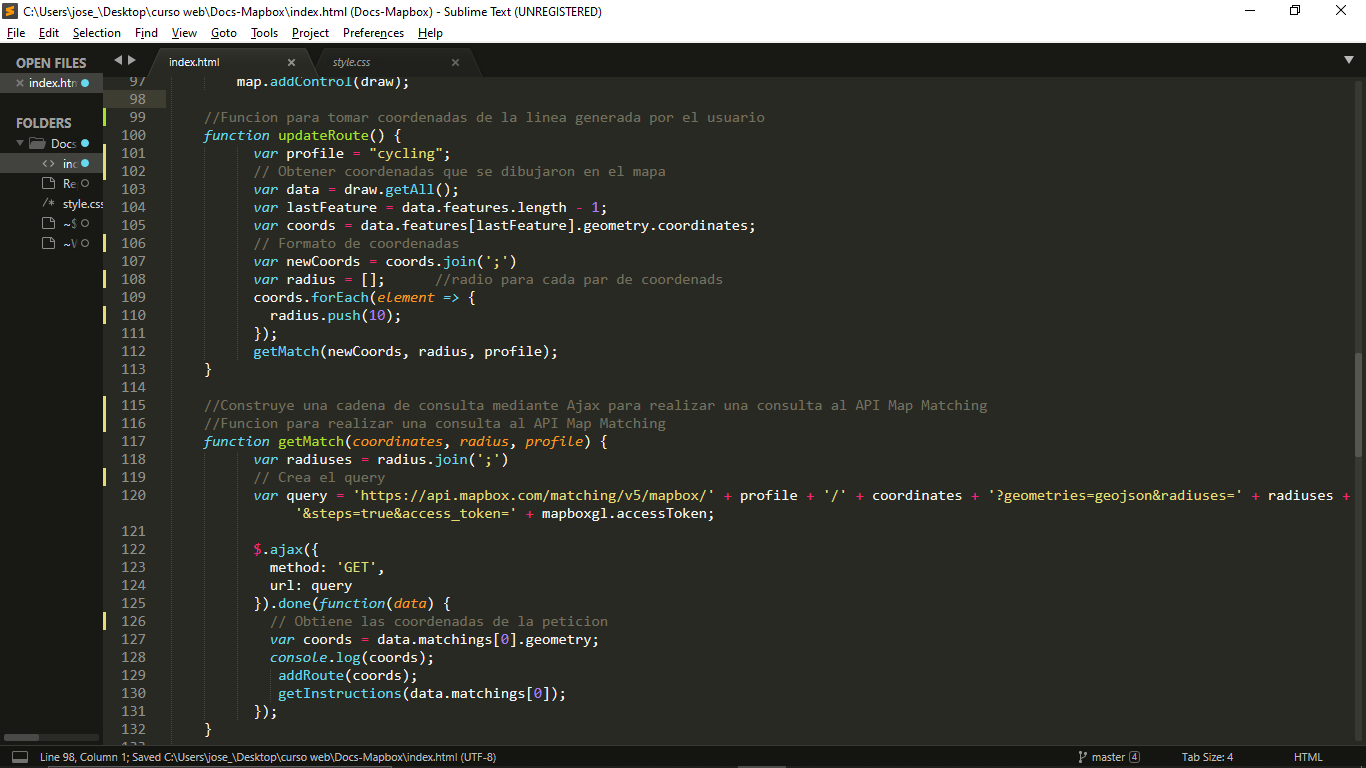


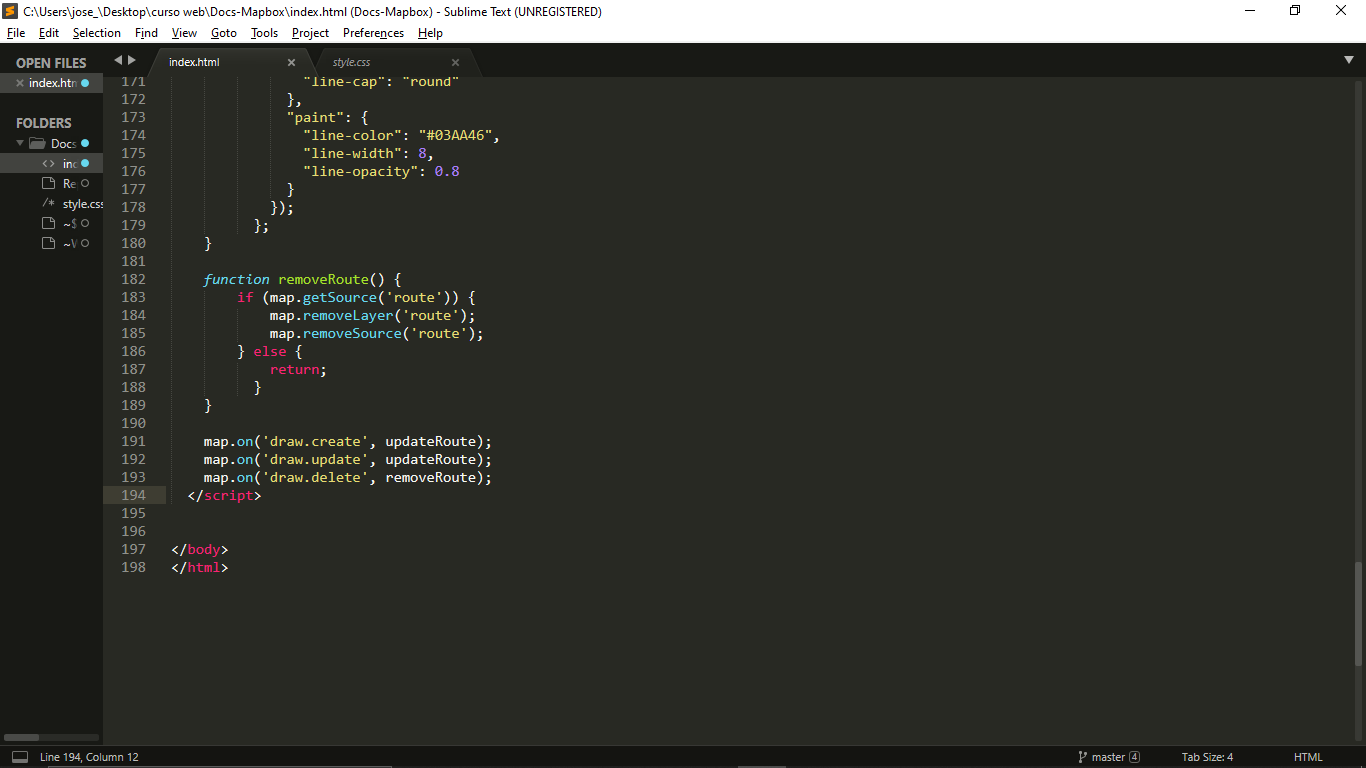
1. La API requiere dos parámetros: **Profile** debe usar la consulta y **coordinates**, debe coincidir con la red de caminos y rutas de OpenStreetMap. Profile puede ser walking, cycling, driving o driving-traffic, Coordinate son pares de coordenadas.

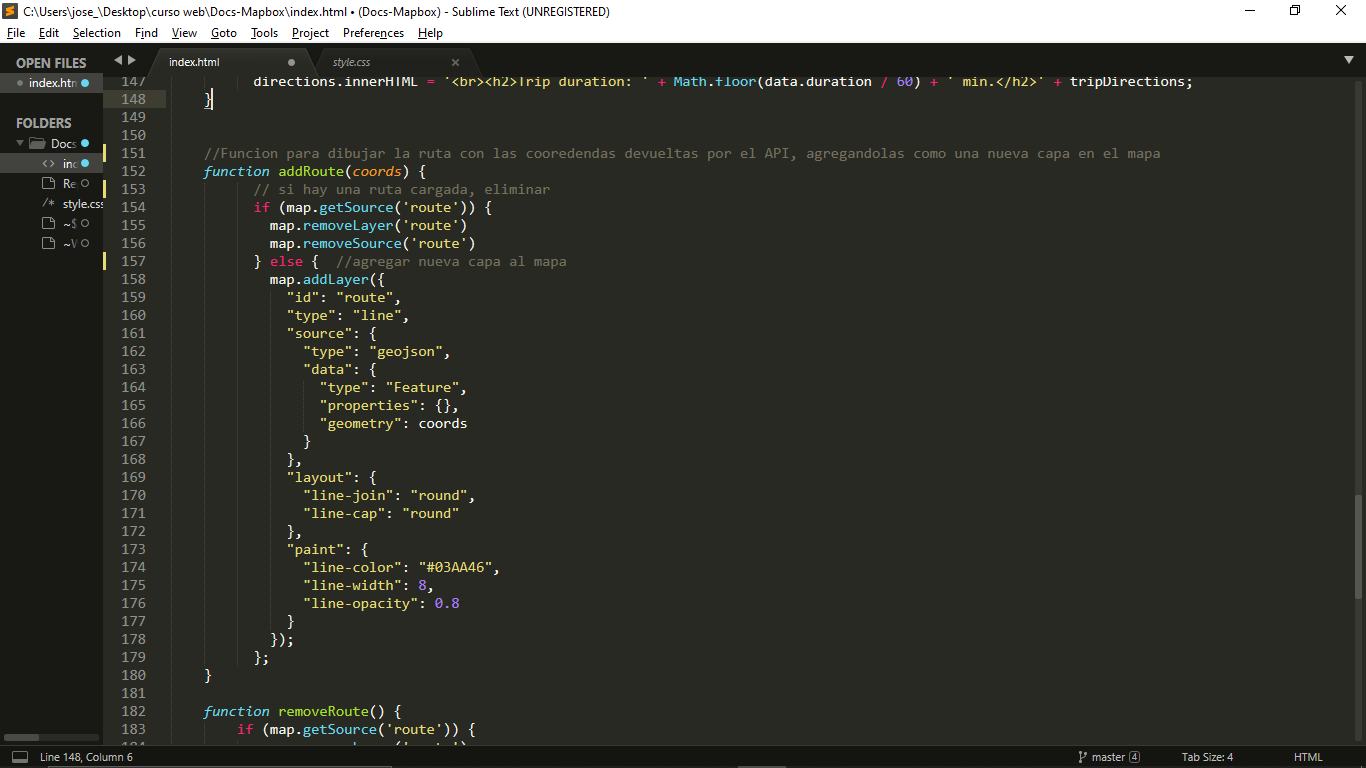
La API Map Matching también acepta otros parámetros opcionales que se utilizan para personalizar la consulta. Para esta aplicación, utilizará dos parámetros opcionales:

* + Radiuses: agregar un radio, para definir la precisión en que debe coincidir con las coordenadas de entrada, no siempre representan una ubicación en la red de carreteras de OpenStreetMap.
  + Steps: Opción predeterminada (True) para devolver las indicaciones paso a paso.

3. Agregar las funciones updateRoute y getMatch.

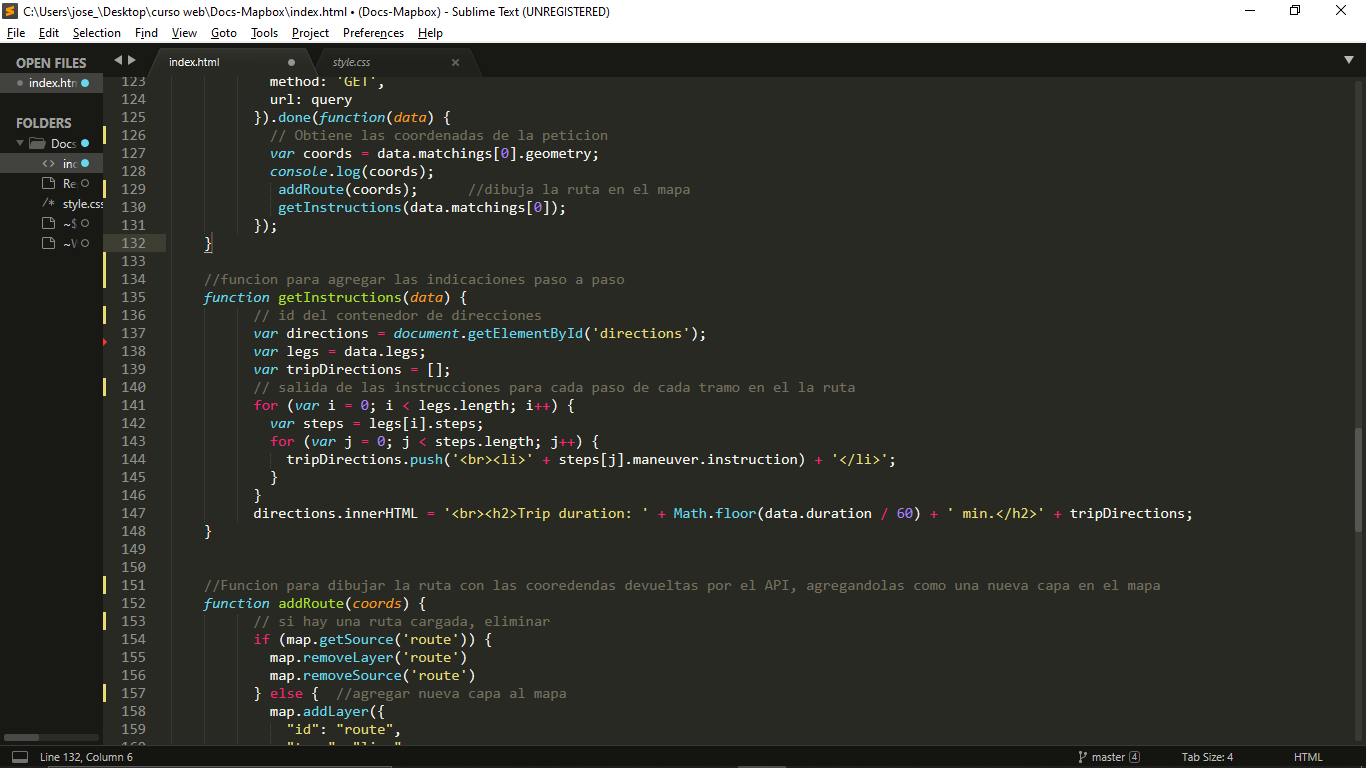
* updateRoute: tomará las coordenadas de la línea generada por el usuario y las formateará para que se puedan utilizar en una consulta de correspondencia de mapas.
* getMatch: construirá la cadena de consulta y utilizará Ajax para realizar la llamada a la API Map Matching.

Para llamar las funciones cuando el usuario dibuja o actualiza una línea en el mapa agregar

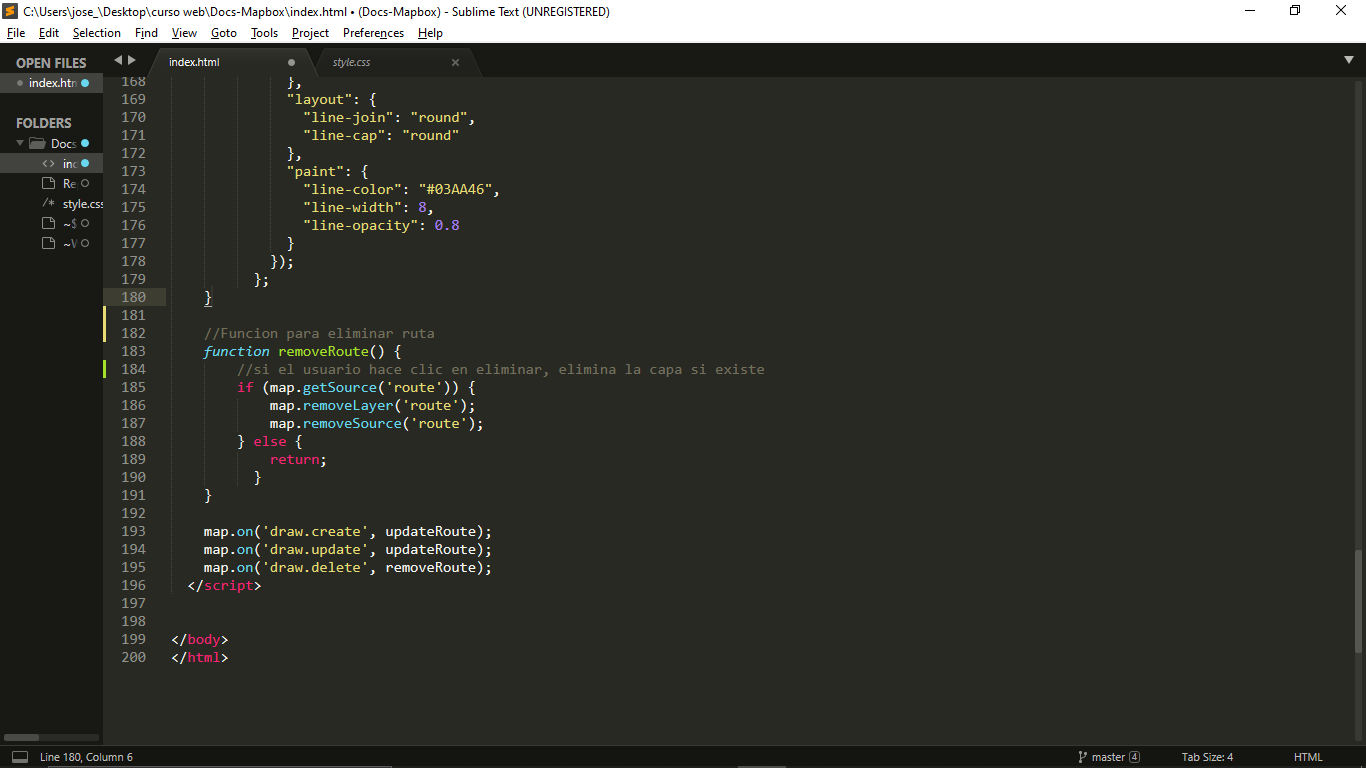
1. Crear función addRoute que toma las coordenadas devueltas por la API para agregarlas al mapa como una nueva capa.

Usar la función addRoute dentro de la función getMatch.

1. Crear función getInstructions para mostrar las indicaciones paso a paso en la barra lateral.

llamar getInstructions dentro de la función getMach para que pueda acceder a los datos devueltos por la API Map Matching.

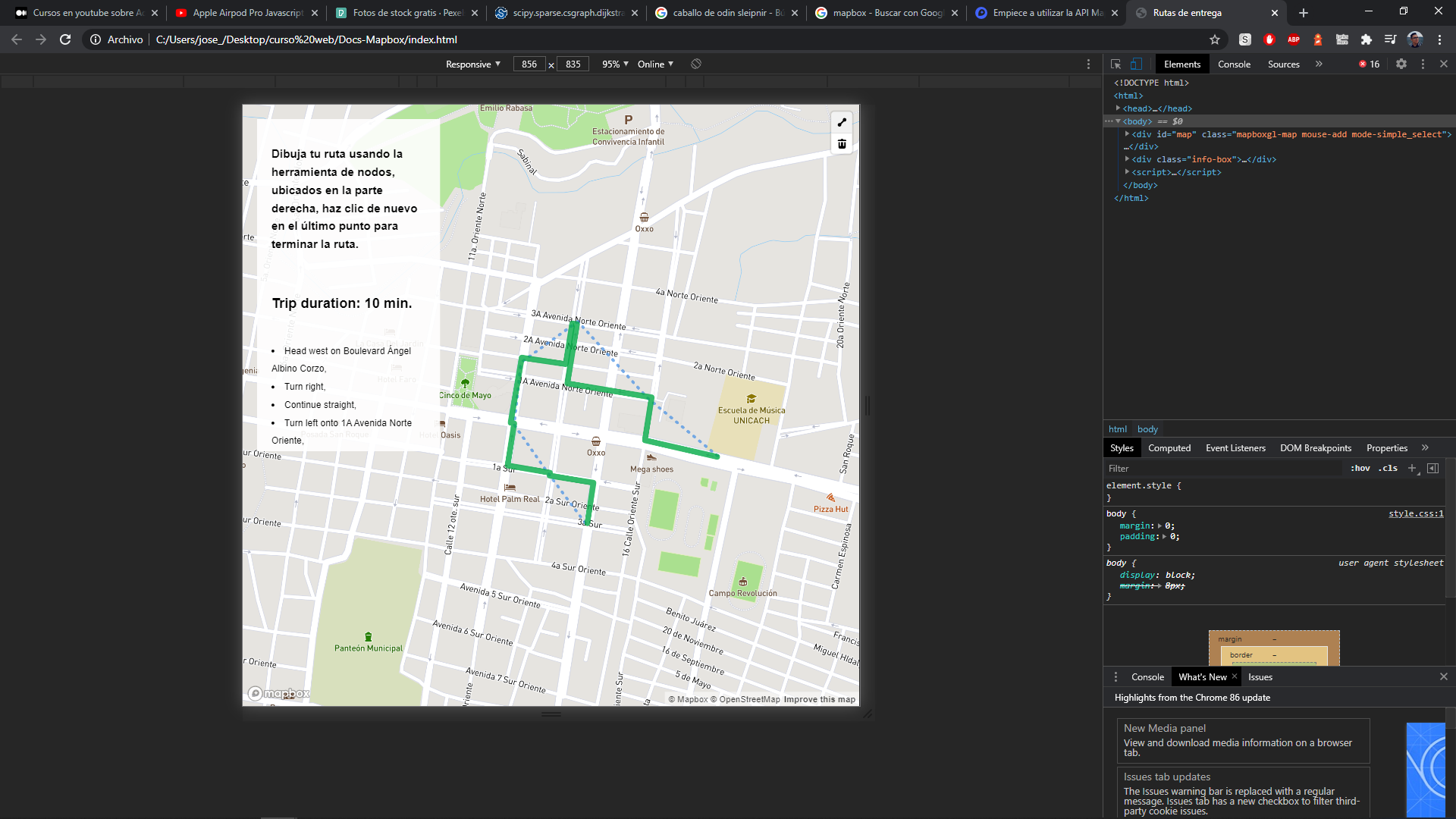
1. Agregar función que permita eliminar una ruta.



**Resultados**



Mapa Ejemplo 1



Mapa Ejemplo 2

# Referencias

*jQuery*. (2020). Obtenido de https://jquery.com/

*Mapbox*. (02 de Septiembre de 2020). Obtenido de Mabbox GL Draw Plugin: https://github.com/mapbox/mapbox-gl-draw

*MapBox Access Token*. (2020). Obtenido de https://account.mapbox.com/

*Mapbox.docs*. (2020). Obtenido de https://docs.mapbox.com/api/navigation/#map-matching

*MaxBox GL JS*. (13 de Agoato de 2020). Obtenido de https://docs.mapbox.com/mapbox-gl-js/api/

*Sublime Text*. (2020). Obtenido de https://www.sublimetext.com/