



ISC

OPT 2 RSD

Bases de datos para

INGENIERÍA EN SISTEMAS
COMPUTACIONALES

Tecnológico de estudios superiores de Ecatepec.
Ingeniería en Sistemas Computacionales.

Alumnos:

Hallal Cervantes Héctor David.

Hernández Gracia Bruno.

Monter Blanco Brian Ricardo.

Pérez Sosa Ángel Alexis.

GRUPO: 5801.

Materia: Opt 2 RSD Bases De Datos Para Dispositivos
Móviles.

Profesora: Griselda Cortes Barrera.

Módulo “Mapa de ubicación”.



Índice.

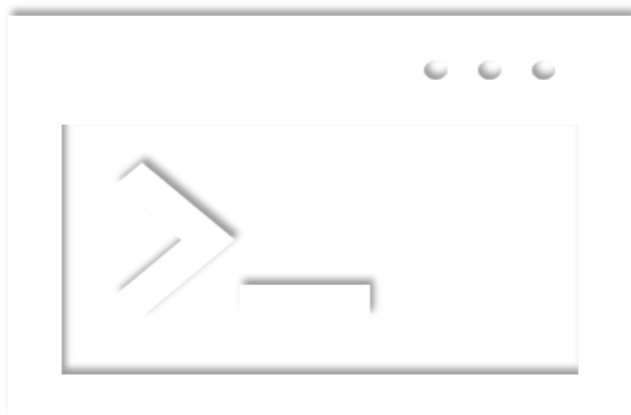
• Portada.....	1
• Índice.....	2
• Introducción.....	3
• Desarrollo	
• Investigación.....	4
• Elementos.....	4
• Configuración del mapa.....	6
• Código del mapa	
• Mapa Regular.....	7
• Mapa Personalizable.....	9
• Mapa Satelital.....	11
• Mapa Lugar en Especifico.....	13
• Mapa Lugar en Especifico en el TESE.....	15
• Mockup.....	18
• Conclusiones.....	19
• Referencias.....	20



Introducción.

En la siguiente información presentada se encontrar algunos pequeños avances sobre mapas de geolocalización como lo son el de ubicación actual y/o de un lugar específico de acuerdo con las coordenadas insertadas a través del código del componente del mapa de Google.

El API de Google para mapas es algo más veterano que Angular, pero siempre es un recurso muy recurrido en determinados tipos de aplicativos. A continuación, aprenderemos a crear y manipular mapas de una manera muy sencilla en nuestros desarrollos web basados en Angular, con el fin de poder exportar dicho modulo y poder hacer uso en nuestro proyecto a desarrollar en un futuro y no tener problemas de compatibilidad o de importación sobre los otros módulos que se encuentren el dicho proyecto.

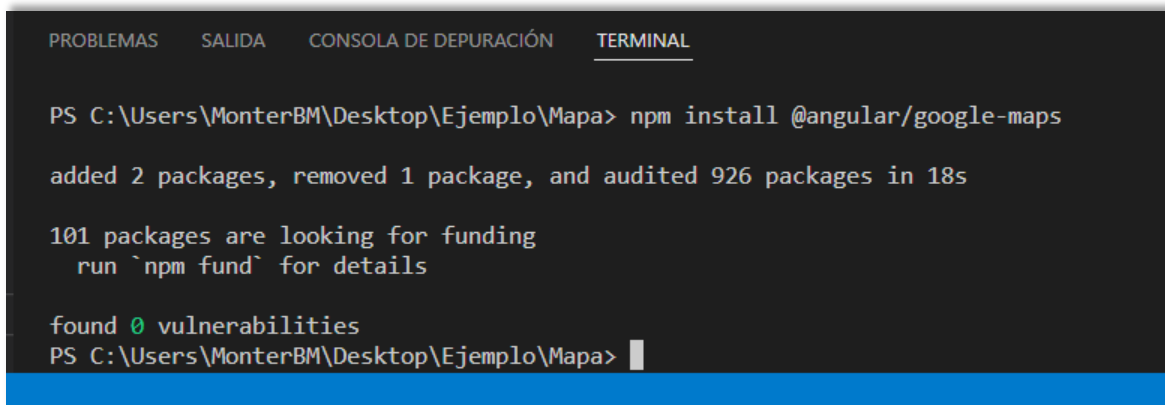


Investigación.

Elementos.

Lo primero que tenemos que hacer es crear un nuevo proyecto en angular e instalar todos los elementos necesarios para poder hacer uso de algunas opciones de Bootstrap, continuamente procederemos a crear nuestro modulo y sus respectivos componentes, los cuales van a incluir a los mapas a realizar en dicha práctica.

Por otro lado, también es indispensable instalar el componente **Google maps** porque no viene por defecto y eso es muy simple se hace con npm vamos a hacer **npm install @angular/google-maps**, le damos a enter y esperamos que instale sin errores.



```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

PS C:\Users\MonterBM\Desktop\Ejemplo\Mapa> npm install @angular/google-maps

added 2 packages, removed 1 package, and audited 926 packages in 18s

101 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\Users\MonterBM\Desktop\Ejemplo\Mapa>
```

Una vez que instala vamos a tener que hacer un par de cosas más antes de empezar a usarlo a diferencia de lo que hacemos a veces con algunos componentes lo primero es crear el api de Google Maps acá en el index.html de nuestra aplicación, lo que se puede hacer es importarlo, porque en realidad el tema del script del API de Google de Google Maps es una script es con un API privada, la cual se paga para tener acceso, lo que pondremos dentro del index es:

```
<script type="text/javascript" src= http://maps.google.com/maps/api/js?sensor=false></script>
```

Y de esta manera lo que va a hacer es traer lo que viene siendo el api, nos van a salir unos mensajes de que se ven en el mapa diciendo esto es sólo una versión de desarrollo.

El siguiente paso es ir al module.ts y ahí lo que hacemos es importarnos el módulo de google maps que nos va a dar acceso a un montón de componentes, el módulo se llama google maps module, y ya lo tendríamos en acceso a todos los componentes que forman parte de este módulo.



```
import { AppRoutingModuleModule } from './app-routing.module';
import { AppComponent } from './app.component';
import { GoogleMapsModule } from '@angular/google-maps';
```

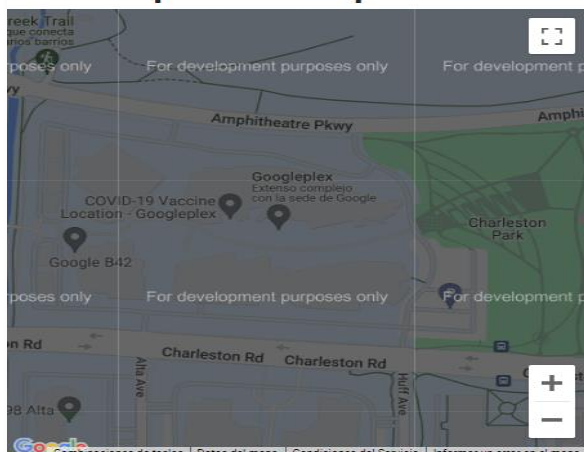
```
@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModuleModule,
    GoogleMapsModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
```

Las GoogleMapsModule exportaciones de tres componentes que podemos utilizar:

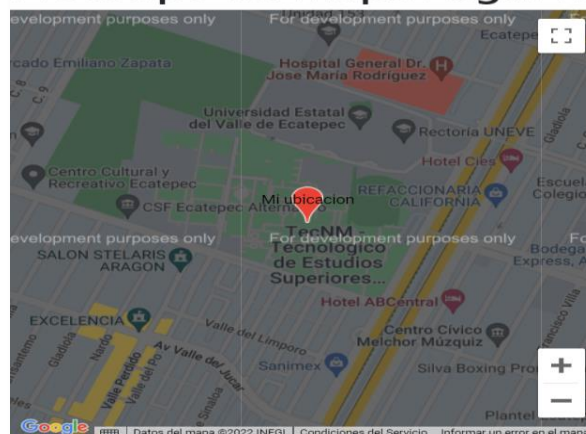
- **GoogleMap:** Este es el envoltorio de Google Maps, disponible a través del google-map-selector
- **MapMarker:** Se usa para agregar marcadores en el mapa, disponible a través del map-marker-selector
- **MapInfoWindow:** La ventana de información de un marcador, disponible a través del map-info-window-selector

Al agregar el componente de Google Maps a una plantilla, ya podemos ver y usar Google Map. El mapa se comportará como un mapa predeterminado con la funcionalidad predeterminada, por ejemplo, puede acercar y alejar, y arrastrar el mapa.

Prototipo de Mapa



Prototipo de Mapa Lugar Especifico



Para hacer un uso completo de la API de Google Maps, también podemos usar las opciones de propiedad y unos parámetros básicos como:

- Width
- Height
- Title
- Position
- Aling

Cuando desarrollamos un sitio web, habitualmente se nos indica que debemos incluir un mapa para que los usuarios sepan ubicar la empresa, los productos, contactos... Google Maps es el servicio de mapas más popular hoy en día que nos proporciona una gran variedad de herramientas y servicios que puedes utilizar a tu antojo.

En esta actividad mostraremos cómo añadir un mapa de Google Maps de manera sencilla en tu sitio web, para después personalizarlo utilizando la librería gMaps.js. Es una librería sencilla y fácil de utilizar, la cual te permite utilizar todo el potencial de Google Maps sin necesidad de escribir líneas y líneas de código.

Configuración del mapa.

Para utilizar Google Maps, debemos importar la librería gMaps.js y la API de Google Maps. Para ello necesitaremos una clave API, por lo que, si ya tienes una, puedes usarla; de lo contrario, puedes crearla en [Google Developer Console](#). Cuando tengas tu clave lista, simplemente reemplaza la parte "YOUR_API_KEY" del script.

```
<script src="http://maps.google.com/maps/api/js?key=YOUR_API_KEY"></script>  
<script src="https://cdnjs.cloudflare.com/ajax/libs/gmaps.js/0.4.25/gmaps.js"></script>
```

Ahora debemos generar el mapa. Creamos un nuevo objeto map y como parámetros configuramos el selector ('#map'), y la latitud y longitud del punto que queremos mostrar en el mapa.

A continuación, tenemos que añadir un marcador en el mapa. Para hacerlo, utilizaremos el método addMarker() con la latitud y longitud donde desees colocarlo. También podemos establecer el zoom inicial en el que se muestra el mapa usando el método setZoom() con un número entero como parámetro. Cuanto más bajo sea el número, más "ampliado" estará el mapa. Además de que lo podemos modificar desde CSS, HTML, App-module.ts, etc.



Código del mapa.

HTML

MAPA REGULAR

```
<!--Google map-->  
<div id="map-container-google-1" class="z-depth-1-half map-container" style="height: 500px">  
  <iframe  
src="https://maps.google.com/maps?q=manhatan&t=&z=13&ie=UTF8&iwloc=&output=embed"  
frameborder="0"  
    style="border:0" allowfullscreen></iframe>  
</div>
```

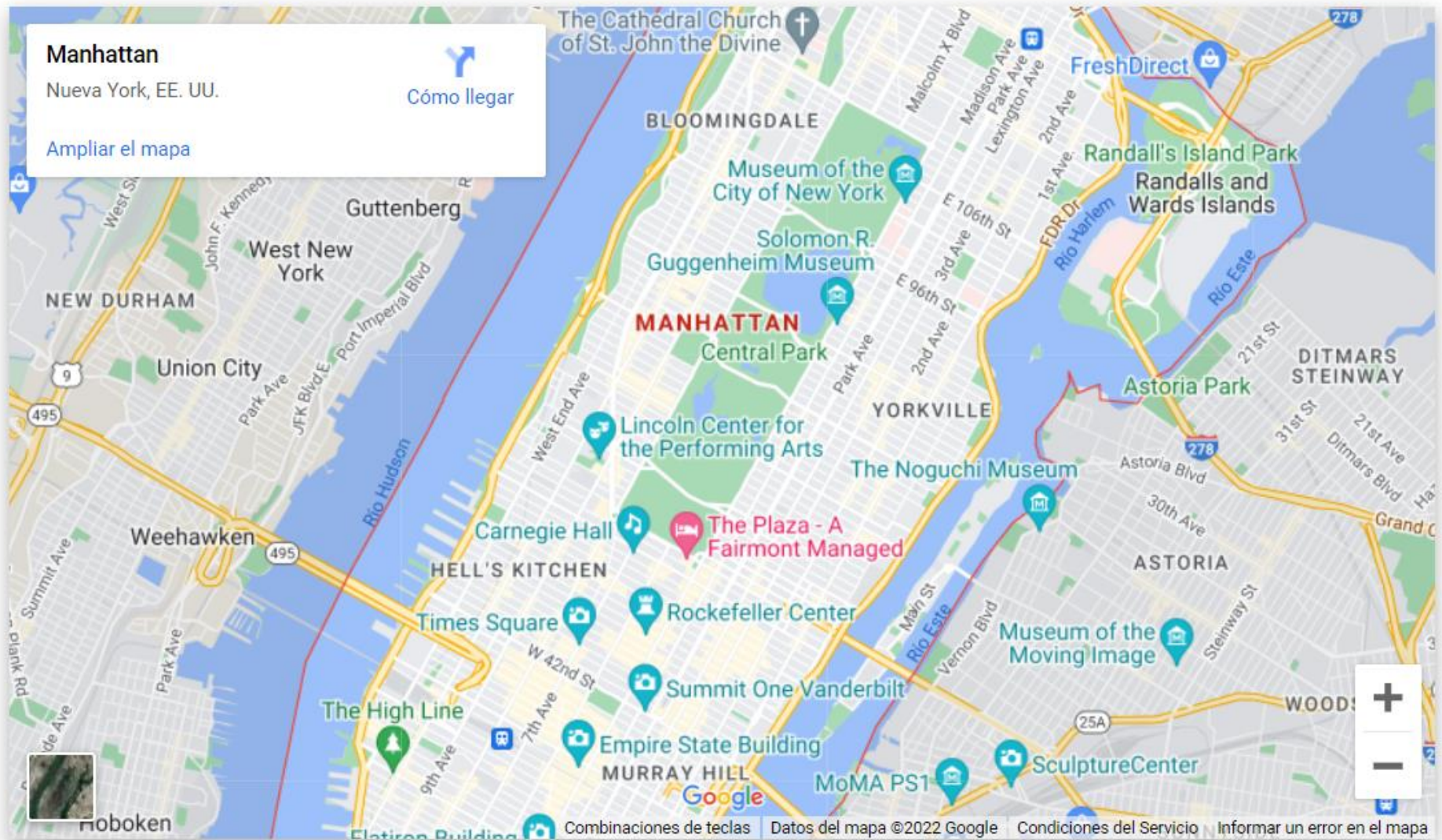
```
<!--Google Maps-->
```

CSS

```
.map-container{  
  overflow:hidden;  
  padding-bottom:56.25%;  
  position:relative;  
  height:0;  
}  
.map-container iframe{  
  left:0;  
  top:0;  
  height:100%;  
  width:100%;  
  position:absolute;  
}
```



Regular map



HTML

MAPA PERSONALIZABLE

```
<!--Google map-->  
<div id="map-container-google-2" class="z-depth-1-half map-container" style="height: 500px">  
  <iframe  
src="https://maps.google.com/maps?q=chicago&t=&z=13&ie=UTF8&iwloc=&output=embed"  
frameborder="0"  
    style="border:0" allowfullscreen></iframe>  
</div>
```

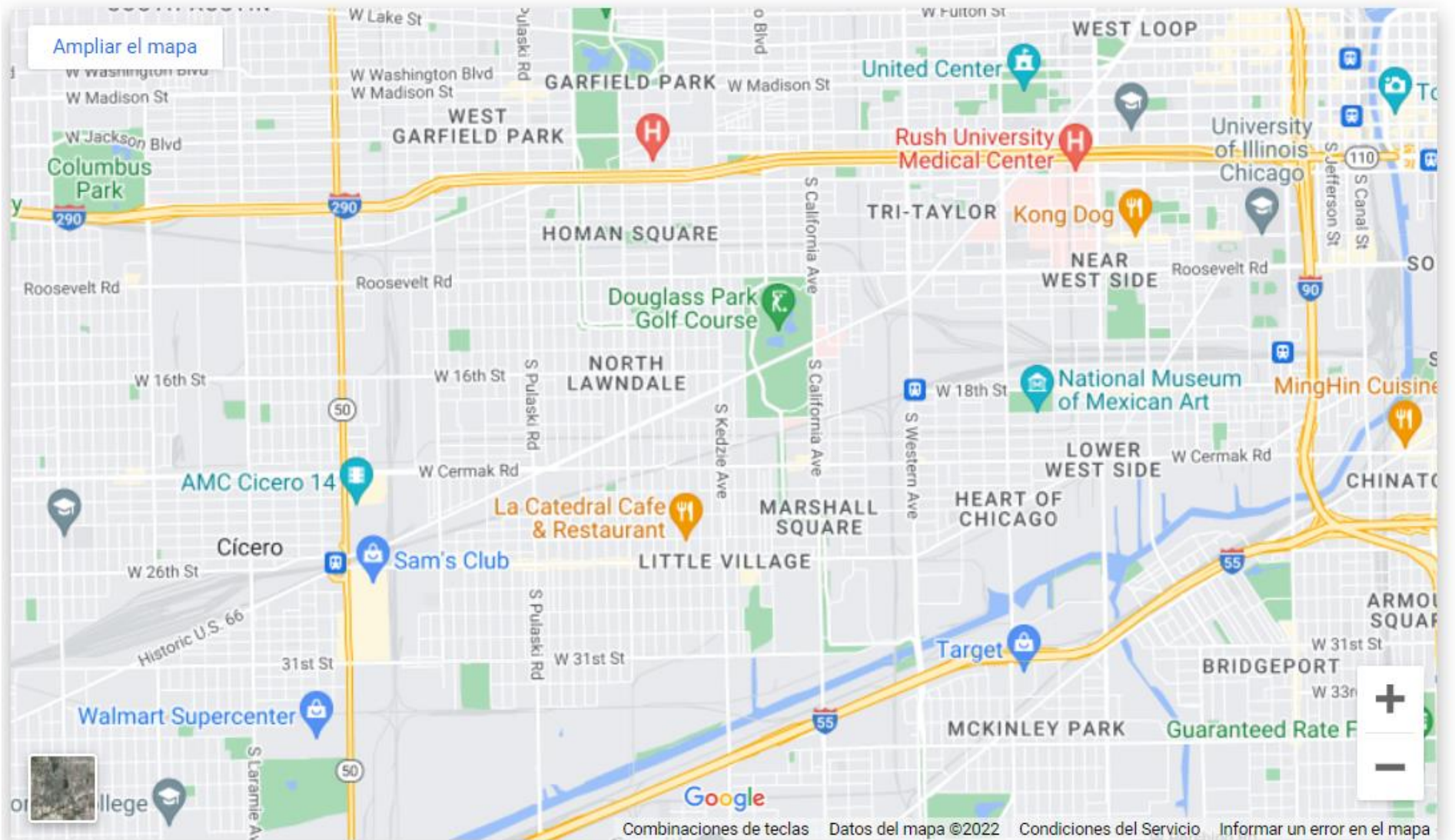
```
<!--Google Maps-->
```

CSS

```
.map-container-2{  
  overflow:hidden;  
  padding-bottom:56.25%;  
  position:relative;  
  height:0;  
}  
.map-container-2 iframe{  
  left:0;  
  top:0;  
  height:100%;  
  width:100%;  
  position:absolute;  
}
```



Custom map



HTML

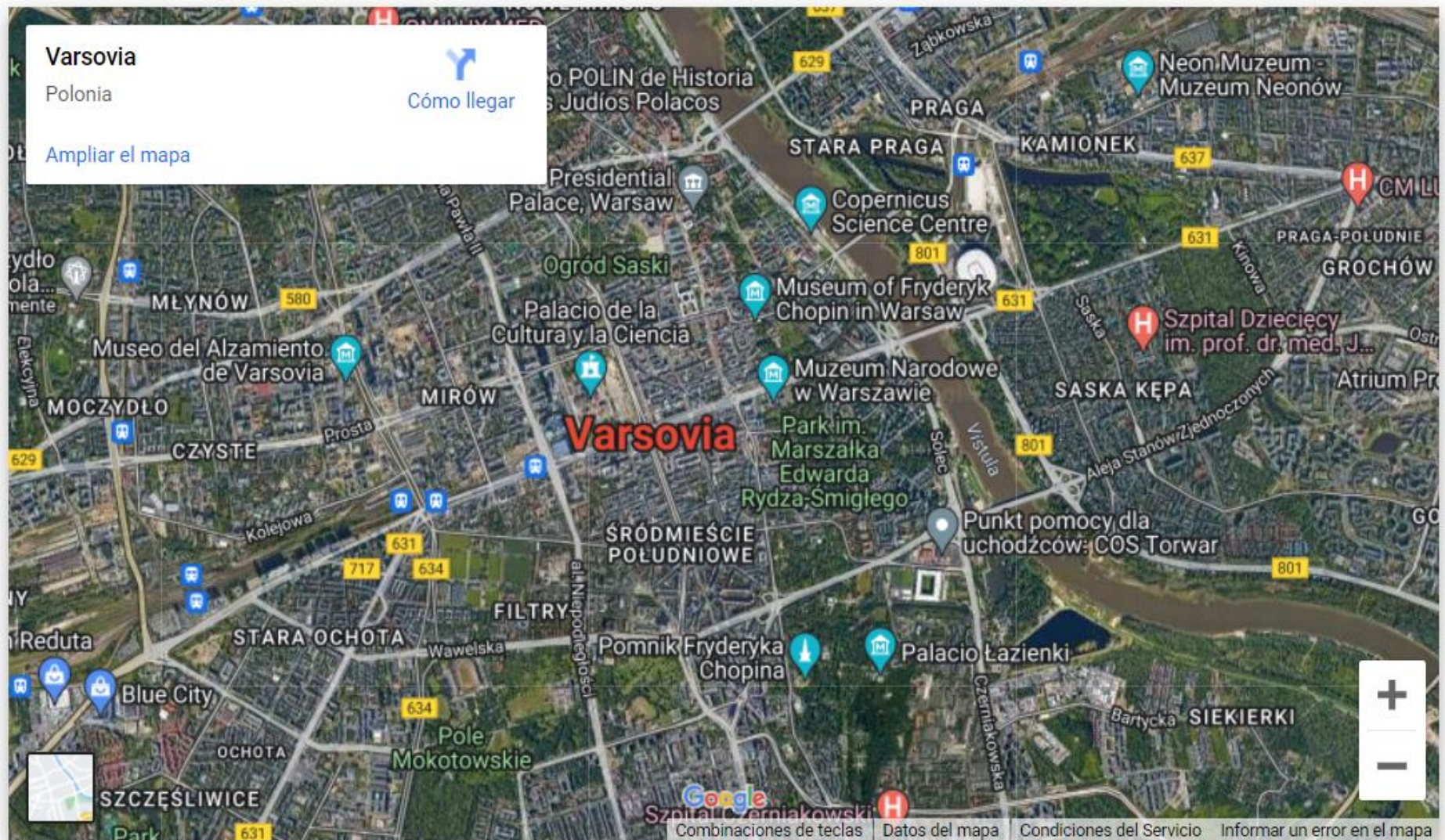
MAPA SATELITAL

```
<!--Google map-->
<div id="map-container-google-3" class="z-depth-1-half map-container-3">
  <iframe
src="https://maps.google.com/maps?q=warsaw&t=k&z=13&ie=UTF8&iwloc=&output=embed"
frameborder="0"
    style="border:0" allowfullscreen></iframe>
</div>
```

CSS

```
.map-container-3{
  overflow:hidden;
  padding-bottom:56.25%;
  position:relative;
  height:0;
}
.map-container-3 iframe{
  left:0;
  top:0;
  height:100%;
  width:100%;
  position:absolute;
}
```


Satellite map



Prototipo

MAPA lugar en Específico

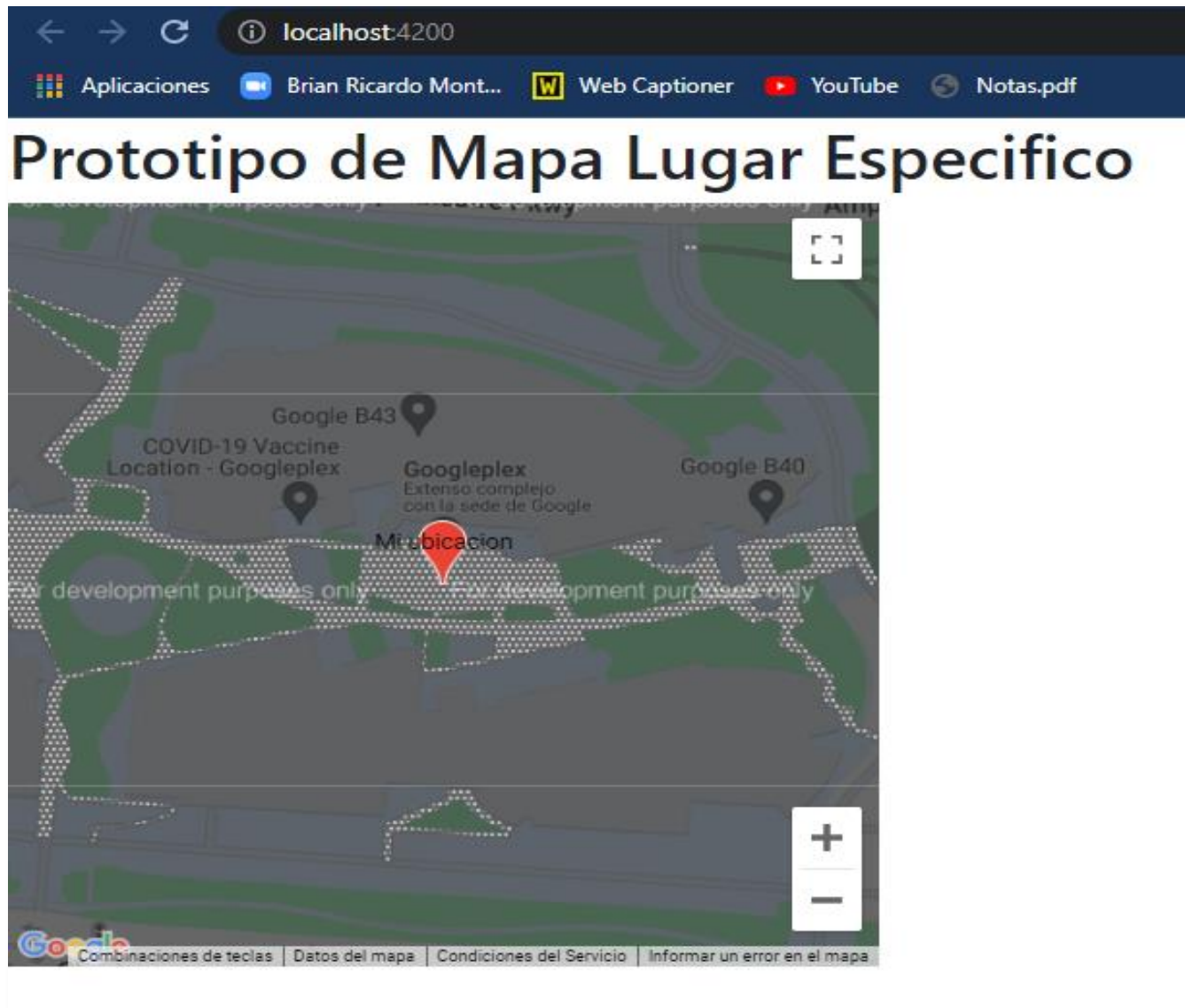
```
<h1>Prototipo de Mapa Lugar Especifico</h1>
```

```
<google-map >  
  <map-marker  
    label="Mi ubicacion"  
    [title]="title" ></map-marker>  
</google-map>
```

App-module.ts

```
export class AppModule {  
  title ='Especifico'  
  position = {  
  
    lat: 19.6044407,  
    lng: -98.9717082  
  
  };  
  
  label = {  
    color: 'red',  
    text: 'Marcador'  
  };  
  
}
```


Mapa.



Prototipo ubicación TESE

MAPA lugar en Específico en el tese.

```
<h1>Prototipo de Mapa Lugar Especifico</h1>
```

```
<google-map [center]="position">
```

```
  <map-marker
```

```
    label="Mi ubicacion"
```

```
    [title]="title"
```

```
    [position]="position"
```

```
  ></map-marker>
```

```
</google-map>
```

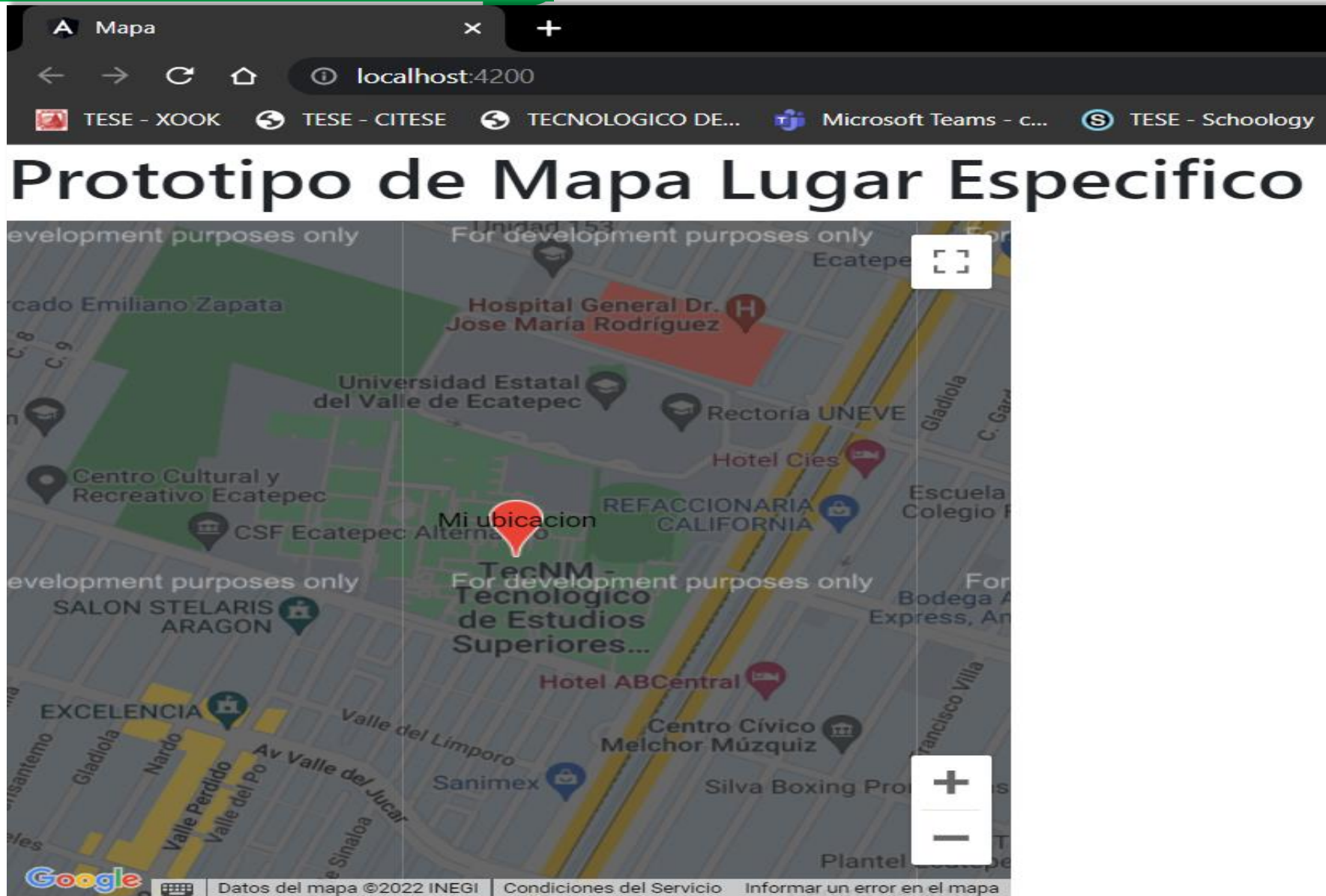
App-module.ts

```
import { Component } from '@angular/core';
```

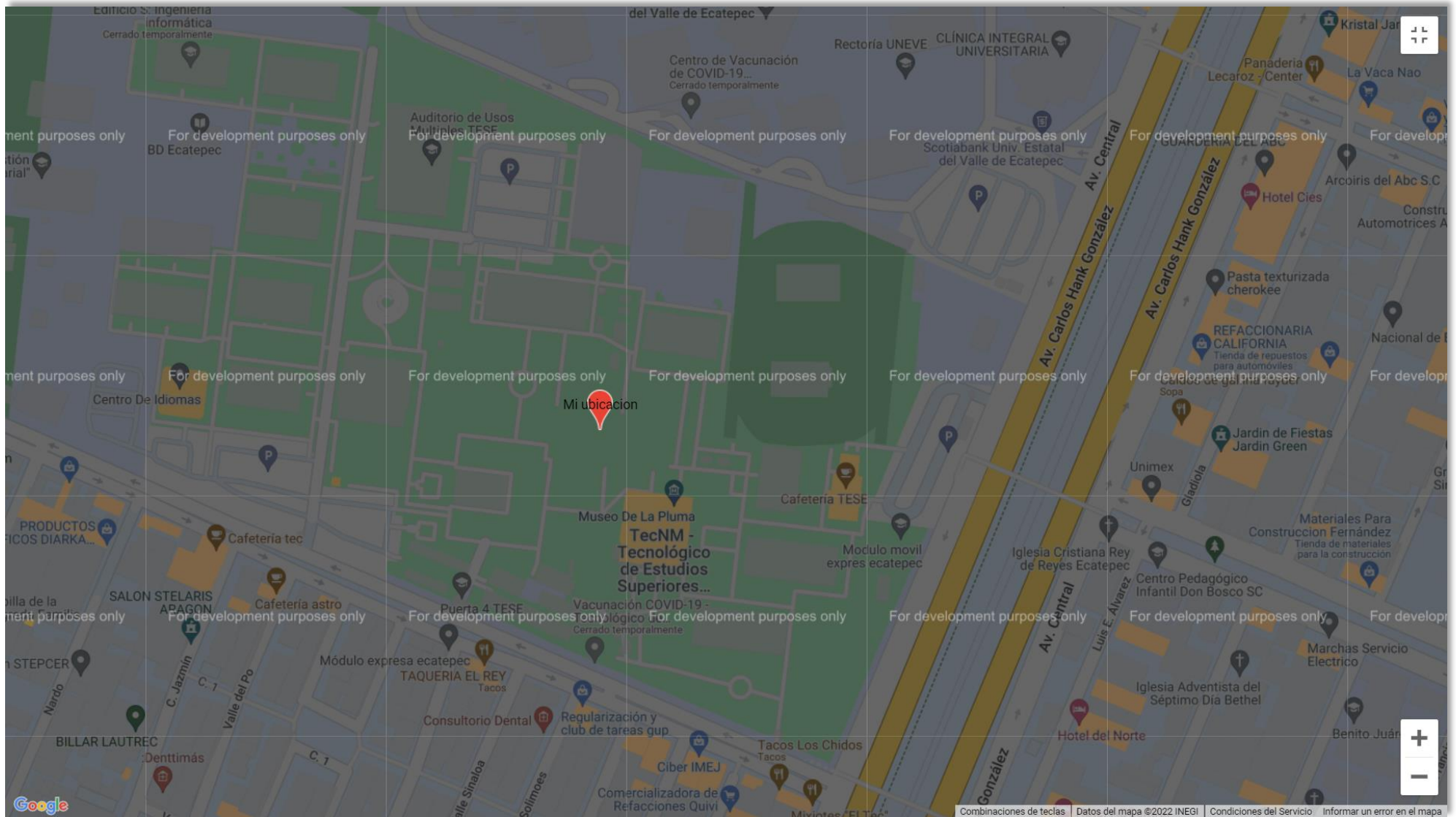
```
@Component({  
  selector: 'app-root',  
  templateUrl: './app.component.html',  
  styleUrls: ['./app.component.css']  
})
```

```
export class AppComponent {  
  title = 'Mapa';  
  position = {lat: 19.510961, lng: -99.040527};  
}
```

Mapa con ubicación en el tese.



Mapa con ubicación en el teso (vista completa).



Mockup.

- En el mockup presentado veremos una vista sobre lo que sería el modulo del mapa ya añadiendo una ubicación fija y una ubicación en específico, a grandes rasgos es como un prototipo, está sujeto a cambios dependiente los requerimientos del proyecto para la primera entrega, además falta aclarar las ideas más claras sobre el proyecto y lo más probable es que cambie el diseño final.



Conclusiones.

Podemos llegar a la conclusión de que hay distintos paquetes que te permiten crear un Mapa de Google en Angular, pero todos son paquetes de terceros, como cuando se implementan algunos códigos de Bootstrap, es similar ya que se hace uso de códigos ya creados con la opción de realizar mejorar o modificaciones pertinentes conforme a los requerimientos de cada proyecto que se nos solicite.

Además de que se tiene que tomar en cuenta algunas variables de latitud y longitud para especificar bien el mapa localizador, sin tener problemas de ubicación ya sea para un mapa de ubicación actual y/o mapa de un lugar específico.

Sin en cambio estamos sujetos a pruebas debido a que aún no se estipula a grandes rasgos los requerimientos, por lo cual es indispensable conocer para implantar el mejor mapa con sus estilos y diseños propios para una buena interactividad con el usuario final.



Referencias.

- <https://mdbootstrap.com/docs/b4/jquery/javascript/google-maps/>
- https://programacion.net/articulo/la_manera_mas_sencilla_de_anadir_google_maps_en_tu_sitio_web_2103
- <https://getbootstrap.com/>
- <https://youtu.be/gdKWeboPvgY>
- <https://youtu.be/fnC5lOaOc5I>

