### Evidencia Día 5 Semana 13

### Tabla de Contenidos

- Documentación Agregar
- Documentación Snapsh

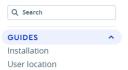
### Agregar Spot

Una de las tareas básicas de Mapistas es poder agregar un punto a un mapa. Sin embargo, para la última versión relase de Mapbox no existen ejemplos oficiales en Kotlin por lo que ha sido muy difícil implementarlo.



All docs > Maps SDK for Android > Guides > Markers and annotations

### Maps SDK for Android



Markers and annotations Map styles

Camera and animations

User interaction

Offline

Migrate to v10

Pricing

EXAMPLES

API REFERENCE

TUTORIALS riangle

TROUBLESHOOTING 🗅

## Markers and annotations

The Mapbox Maps SDK for Android offers several ways to add markers, annotations, and other shapes to a map. This guide helps you choose the best approach for your application based on factors like interaction requirements, number of features, the need for customizing the style of features, and data sources.

### **Annotations**

You can add annotations to the map including point, circle, polyline, and polygon shapes using the Mapbox Maps SDK's Annotations API. Use the Annotations API to create annotation managers based on the type of annotation that you're interested in. Every annotation manager handles a collection of annotations. Once a manager has been created, you can create and add individually styled instances of the corresponding annotation type.

#### Benefits:

- Built-in interaction support like selecting and dragging annotations around the map.
- No external data file necessary.
- Every annotation can be individually styled.
- Every annotation layer can be adjusted to be above or below another layer.
- Same performance benefits as using style layers.

### Limitations:

- Inefficient for adding many features (> 250) to the map.
- No default marker image available.

### Default markers

The Mapbox Maps SDK's Annotations API does not provide a default image for symbol layers. You must provide an image and add it to the style before using the PointAnnotationManager to add it to the map.

### Snapshots

### On this pa

Þ

Annotations
Default ma

Other sha Interactivii Style layers





# Mapbox Tiling Service

Q Search

### GUIDES

Tileset recipes

Tileset sources

Errors

Warnings

Frequently Asked Questions Pricing

EXAMPLES
RECIPE SPECIFICATION
TUTORIALS Å

TROUBLESHOOTING riangle

All docs > Mapbox Tiling Service > Guides

# **Mapbox Tiling Service**

- ✓ Create highly customized vector tilesets
- ✓ Specify IDs for each feature in a tileset
- ✓ Reuse uploaded data across multiple tilesets
- Generate tilesets with multiple layers to optimize for performance



### **Public beta**

Mapbox Tiling Service is in public beta. All features and workflows are subject to potential changes.

Mapbox Tiling Service (MTS) is a tool for creating vector tilesets. With MTS, you use sets of configuration options (tileset recipes) to transform your geospatial data into vector tiles. The resulting tiles are hosted on Mapbox servers for use in your applications

Vector tilesets are helpful if you want to visualize a large amount of data on a map quickly. When creating vector tiles, you can turn gigabytes of raw geospatial data into mere kilobytes, which can be critical for complex data visualizations since most modern browsers support loading about 100 MB of data at a time. And when you use MTS to create vector tiles, you have precise control over how your geospatial data is reduced into tiles.

# Use cases

You can use MTS for many use cases. For example, you could use it to:

- Tile census boundaries to make interactive election visualizations.
- Add hiking paths, trails, or other roadways not included by default in Mapbox Streets to your map.
- Visualize activity data like runs or bike rides on a map.
- Add hotel or real estate properties and their attributes to a map.
- Create multi-layer tilesets (tilesets that contain up to 20 different data layers).

Because MTS is hosted by Mapbox and designed to scale, you can build full, end-to-end data pipelines with it. It's the same

### Reflexion

Se me está poniendo bastante cuesta arriba desarrollar este proyecto en Java debido a la nula documentación de la versión relase del SDK de Mapbox. Me siento bastante a la deriva respecto de las desiciones que hay que ir tomando dentro del proceso de desarrollo. Dudo bastante si fue correcta la opción de utilizar Mapboxs

On this paរូ

Use cases How to use N

