

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

**GitHub Username:** [izartxo](#)

## GeoMemo

### Description

This app fixes memory problems. It stores ideas, tasks and/or memos related to specific places and whenever we are close to those spots it activates the “geostored” memo to remind us whatever we wanted or needed to do.

### Intended User

All users are welcome. Anyone wanting to remember something without worrying about forgetting it.

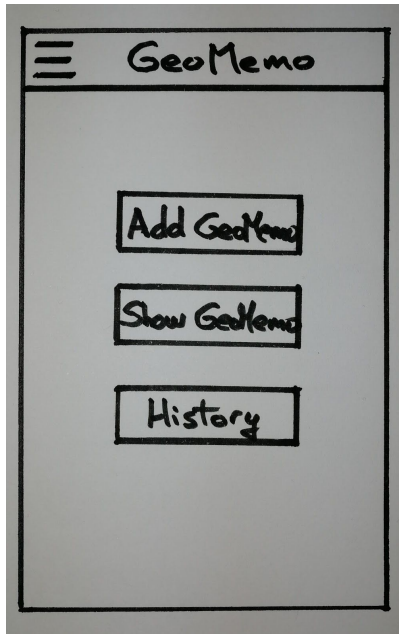
### Features

This are the main features of the app:

- Save memo and geo information
- Geolocalize notifications with memos
- Record past geomemos history/log

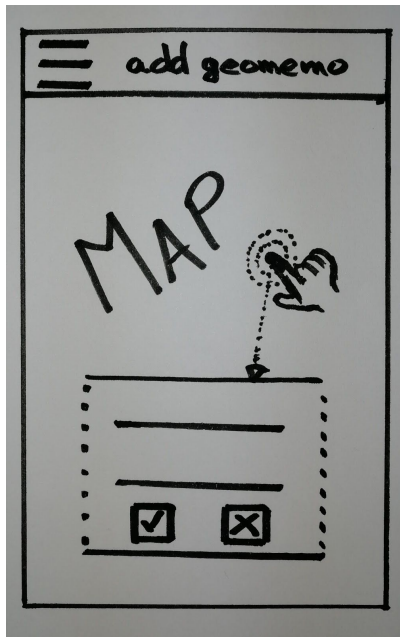
## User Interface Mocks

### Screen 1



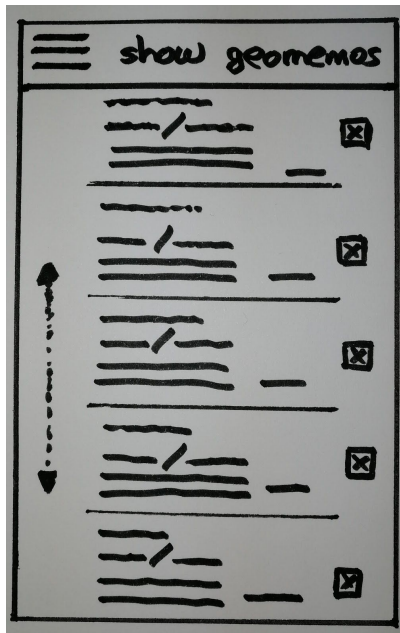
The above shown figure shows a schematic of the main menu of the app. 3 actions can be done with the buttons: (1) create a new geomemo, (2) show all active geomemos, (3) query past (read and deleted) geomemos.

## Screen 2



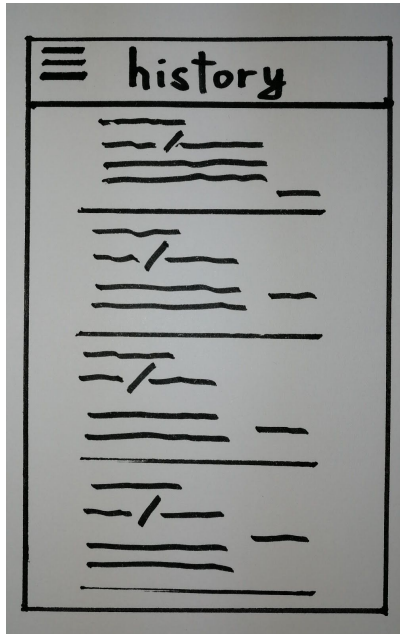
When the “Add Geomemo” button is clicked, the app shows a map where the user can register a geomemo linked to a particular position or place. Once selected the target spot, the user will be able to type the title and a brief description of the memo in a dialog window. This way, when the user approaches the target place a geofence will activate the geomemo.

## Screen 3



After clicking the “Show Geomemo” button, a list of all active GeoMemos will be shown on the screen. Information about the typed message, location and creation time will be provided. There will also be the possibility to deactivate a GeoMemo and move it to the history register.

## Screen 4



Finally, when the “History” button is clicked, a list of all read or deleted GeoMemos will be shown on the screen. There will also be the possibility to deactivate a GeoMemo and move it to the history register. Additionally, it will be possible to clear the history from the toolbar.

## Key Considerations

### How will your app handle data persistence?

Room stores all geofences generated by the user and memos. Widget retrieves data from a content provider.

### Describe any edge or corner cases in the UX.

All screens access to the main menu with home button

### Describe any libraries you’ll be using and share your reasoning for including them.

Butterknife for manage views on coding

**Describe how you will implement Google Play Services or other external services.**

Geofences to activate on place trigger(Location services), Maps to show clicked area and memo(Maps services)

## Next Steps: Required Tasks

### Task 1: Project Setup

Add libraries for Play Services(Geofence, Location, Map)  
Add ButterKnife library and Room requirements

### Task 2: Implement UI for Each Activity and Fragment

- Build UI for MenuActivity, which is the main screen to access the different options. If the app is not in foreground the notifications will alert with memos. The app remains in the background listening for new geomemos.
- Build UI for Map dialog to enter data.
- Build UI for Show GeoMemos on list
- Build UI for Show History GeoMemos on list

### Task 3: Implement Geofences system

- Implement map and register Geofences system.
- Implement a background system to listen all geofences and location requests to provide them to the GeoMemoReceiver.

### Task 4: Implement database structure

- Generate Room architecture to manage all GeoMemo data and activity operations.

### Task 5: Implement notifications

- Implement a notification system to group all received geomemos.

### Task 6: Implement widget

- Create widget and load data from Room with a content provider.