**GEOSTICK**

**Location based Reminder**

**MOBILE PROGRAMMING**

**PROJECT**

**By**

**VINAY DATTA PINNAKA**

**ABSTRACT**

Reminders are must for every human, using sticky notes is most common way of reminding things to do. Smart phones made these reminders easy to set, these smart phones also gives you a convenience to set reminder for a particular time and at particular location. But the applications that are in use for setting reminders organize all these reminders in a list which is not a user friendly way. Populating the location based reminders you make at a particular location on a map is one of the efficient way to organize reminders, Geo stick a Location based reminder system fallows the same. Geo stick has another feature i.e. a new way of sending message, using Geo stick we don’t send a send message we just drop it. If a reminder or a message is set using Geo stick, whenever a person enters into that geo boundary set earlier he gets the notification. Geo stick classifies the location based reminders into three categories and they are I geo stick, We geo stick and You geo stick. A person can identify the geo stick based on the color of the marker which is an easy interpretation. User can set the radius of the geo stick, this allows him to customize the geo boundary for each and every geo stick.

1. **INTRODUCTION**

Geo stick is not only another location based reminder application, it something more. Reminders and messages based on the location are populated on the map instead of showing them in conventional list view.

* 1. Existing applications
* Google location based remainder is a simple app that syncs with google calendar but the drawback is it has default geo radius set. User cannot modify this.
* Life 360 is another application which gives location based reminders but uses the conventional list view to display all the reminders set earlier.
* Geobells is nice app but does not have a feature to send messages to friends based on location.
  1. Proposed Approach

Geo stick classifies the reminders into three categories

* I Geo stick
* We Geo stick
* You Geo stick

**I geo stick** is confined to only single user, he can set a geo stick at a particular location and whenever he is in the close vicinity where a geo stick exists a notification pops out. All the I geo sticks that are set are shown on the map with Pink color markers.

**We geo stick** is global not confined to any one If a person made a We geo stick, It appears on all the Geo stick users map and notification for we geo stick is users choice. A green marker on the map indicates we geo stick.

**You geo stick** is to drop a messages to your friend at a particular location when your friend is in the close vicinity he gets your message as notification. Blue marker indicates a you geo stick.

In Geo stick app users can identify the type of reminder or a message based on the marker color that appears on the map. If user taps the marker he can see the title of geo stick and to view the complete message he need to make another click on the info window of the marker. This way of visualizing the geo sticks on the map is easy for the user to organize and search for the reminder which he already set.

1. **IMPLEM** **ENTATION**

To implement the geo stick application a Mobile backend service system named Firebase is used. Each and every mobile client have their instance in the firebase server whenever a user adds a new geo stick, parameters like title, message, location and type of geo stick are captured by the client sends to server. The data in server is saved in the format shown in the figure. And whenever client requests the data a json object is sent with bundle of all the data that is stored earlier.



* Public
  + Geo stick 1
  + Geo stick 2
* Users
* User 1 details
* User 2 details
* User 1
  + Geo stick user 1
  + Geo stick user 2
* User 2
* User 3

Data Storage in Firebase

Geo sticks on the map

* 1. **Server side implementation**

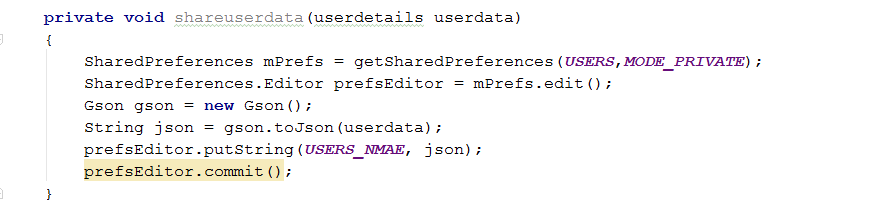
Every mobile client need to be authenticated before it can access data from the server. So for this purpose an email-password login system is used at the client side. When a client is authenticated by the server an authentication token is sent to the client. A unique id is assigned by the server to each and every client. After authentication a mobile client can read and write data to server database using the UID provider in the authentication token.

Data is organized in three categories public, users and each and every single user. All the data corresponding to We Geo stick is stored under public tag, a copy of all the registered user with their username, email id and last updated location is stored under the Users tag. This data is used for populating all the users on the map. Apart from this each user have their own tag under which I geo sticks were stored.

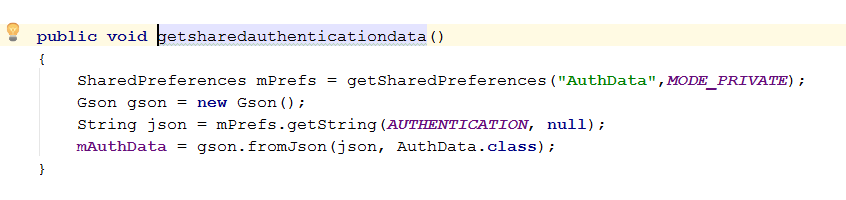
* 1. **Mobile Client side implementation**

When a user login with the registered credentials an authentication token is received and the UID that is obtained with the authentication token is stored in Shared preferences of the mobile. This will allow usage of this UID in different activities of the application. A firebase event listener is implemented which always listens for the addition of data in the database. If data is updated in the server this listener at client side retrieves the data and updates the geosticks on the map. Whenever the new geosticks is obtained notification is set for the corresponding location with the details. A detailed code implementation foe these features is presented in the fallowing section.

* + 1. Shared Preferences



Create a data object and convert this object to json format and put the data in shared preferences with a key



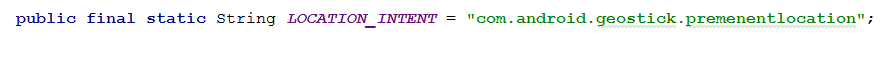
This json object is obtained in another activity class file and assigned to new data object and all the parameters are used for further usage.

* + 1. Firebase Event Listeners

Firebase value event listener is implemented with the fallowing code whenever a new child is added the data snapshot is listened by the client and the objet returned is pushed to an array list of data objects. Latitude and longitude of the new geosticks is obtained from this data object and a marker is selected based on the mode of sharing and added to the map object.

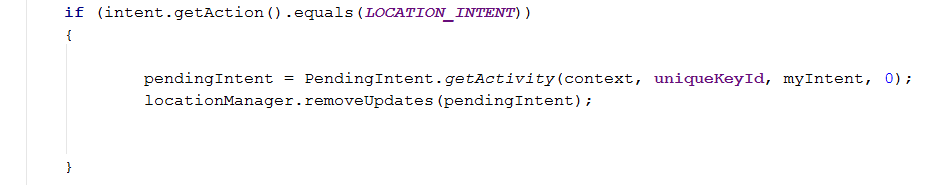


* + 1. Broadcast Services





This code snippet give a brief understanding how a pending intent is created, this pending intent is sent to the location manager which triggers based on the users location. In a class file which extends services this pending intent is received and the corresponding location manager update is removed.

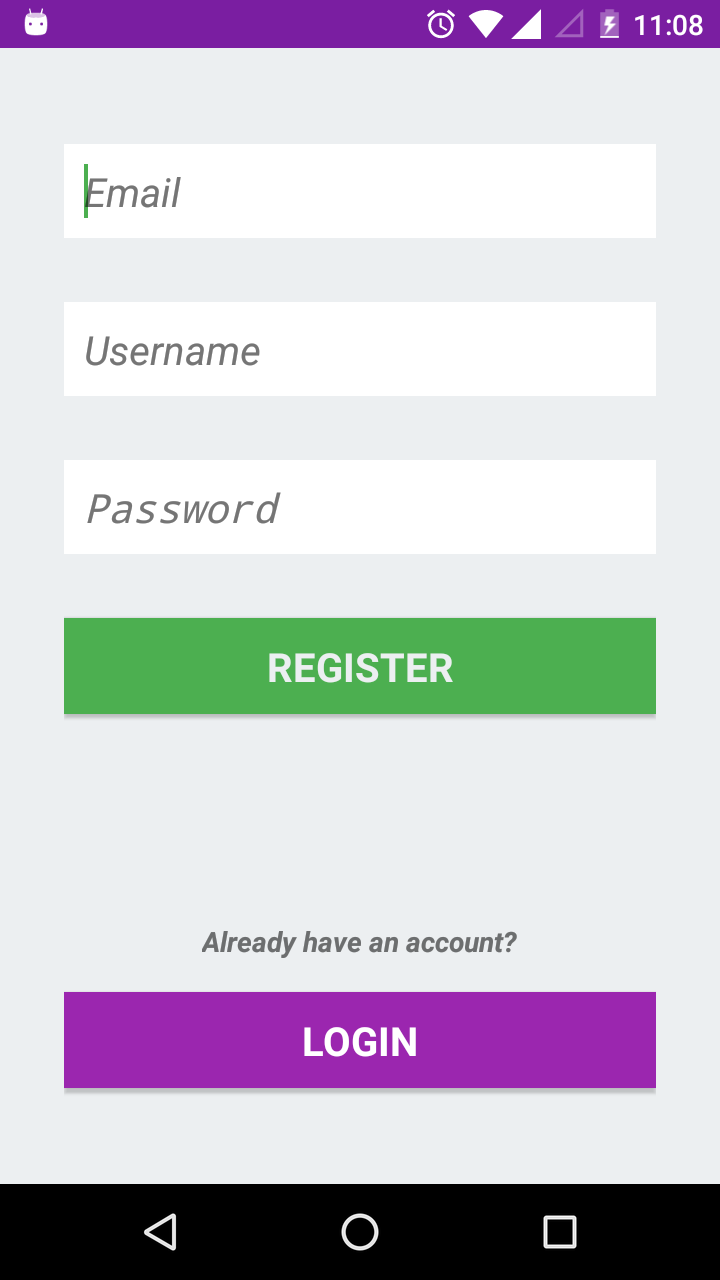
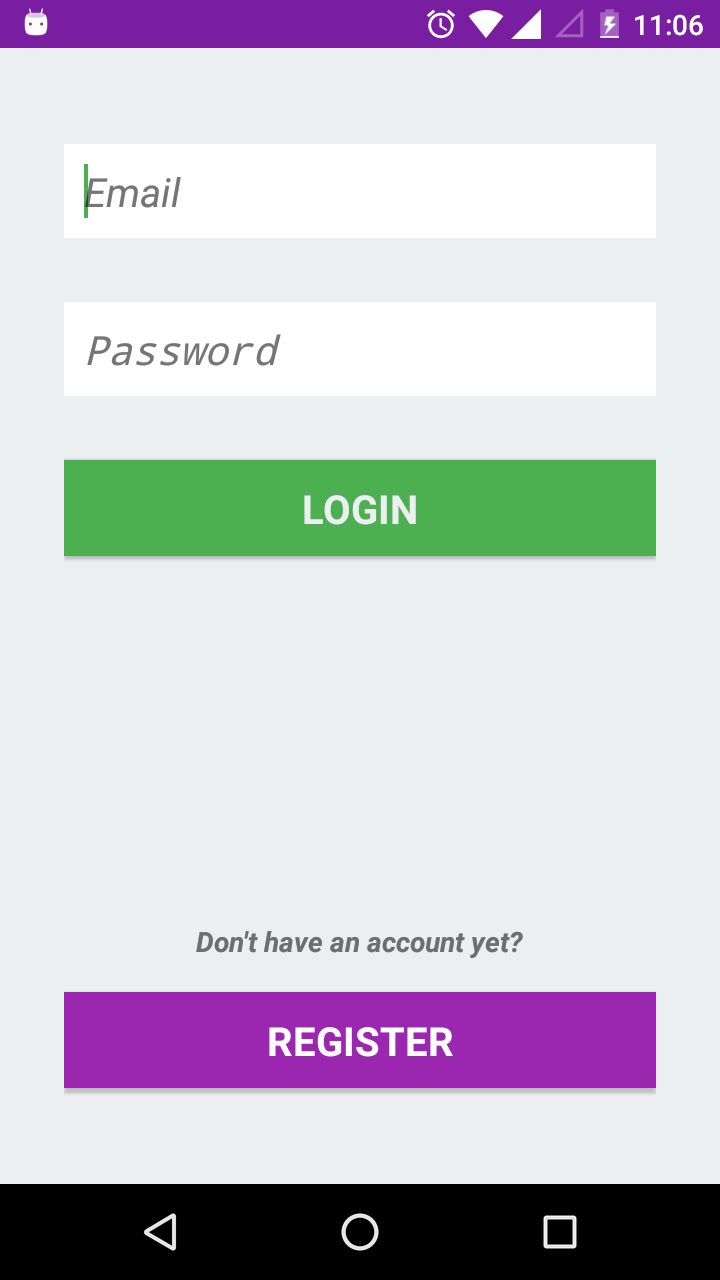


A notification manager is called with the corresponding details from the pending intent which is obtained earlier.

1. USER GUIDE

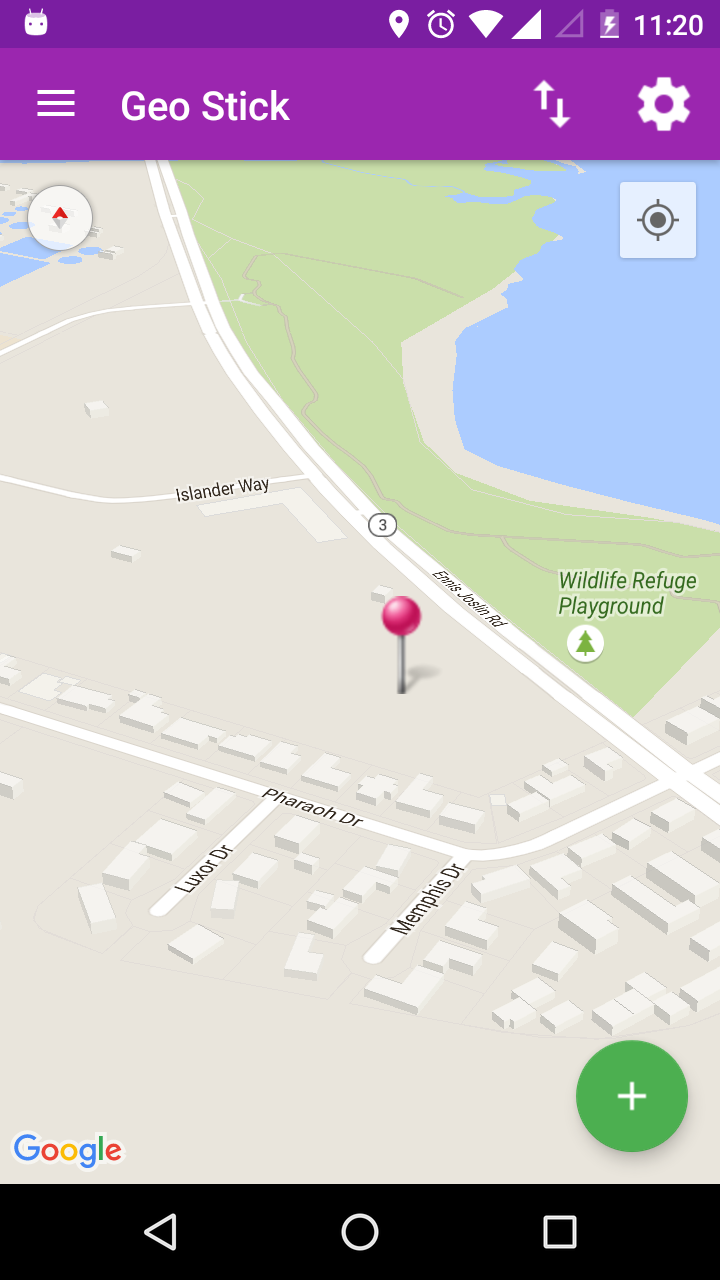
**Step 1**

Register or login to the app with the valid credentials



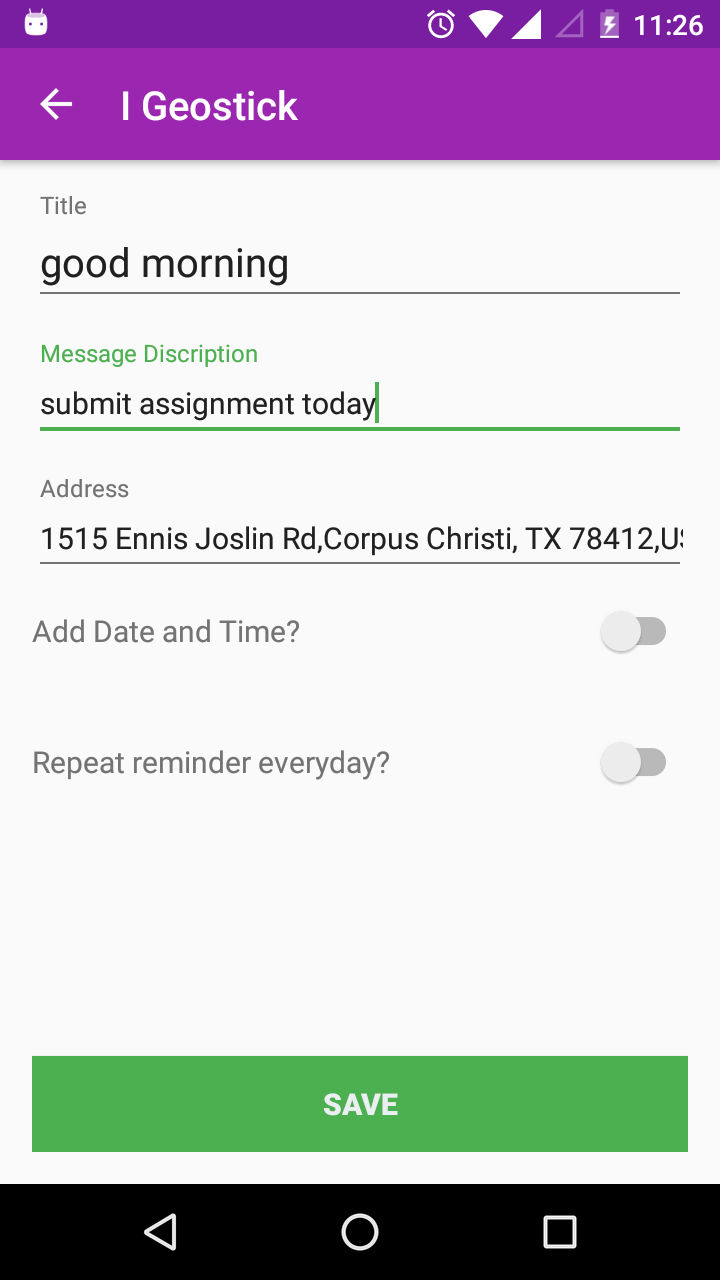
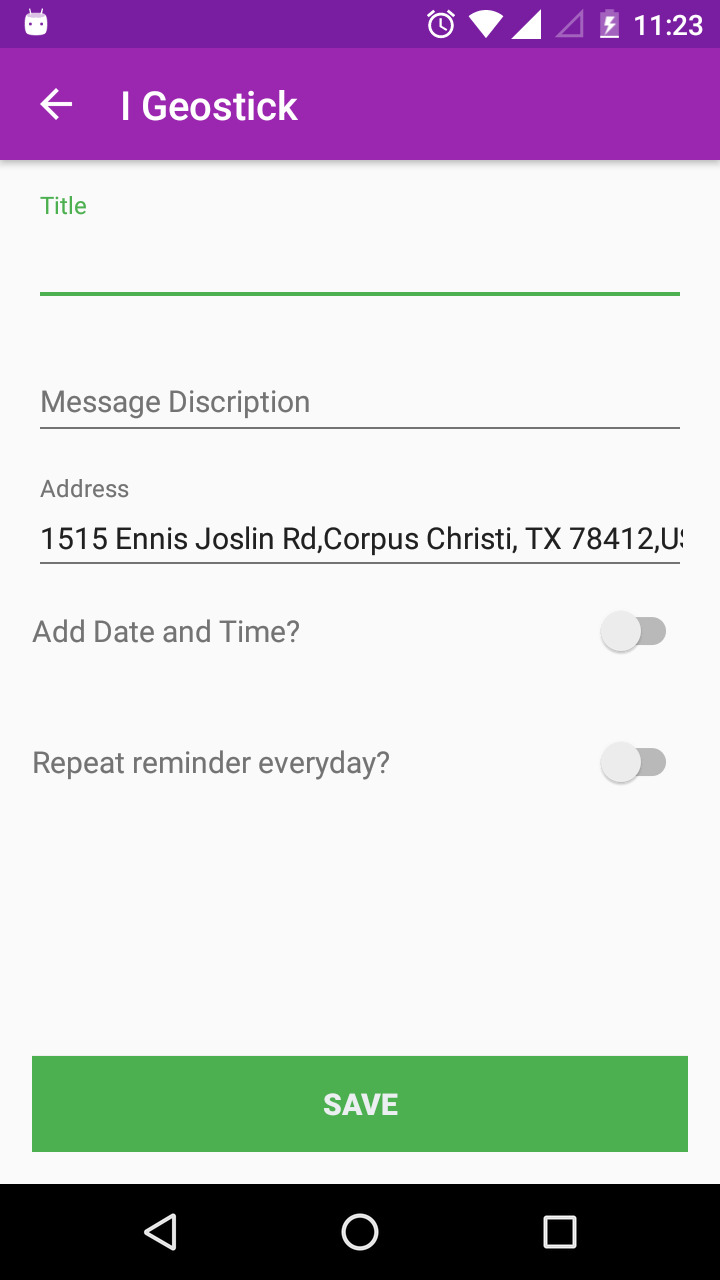
Step 2

After login, activity with map fragment populates out you can add geo sticks to the map by taping anywhere on the map.



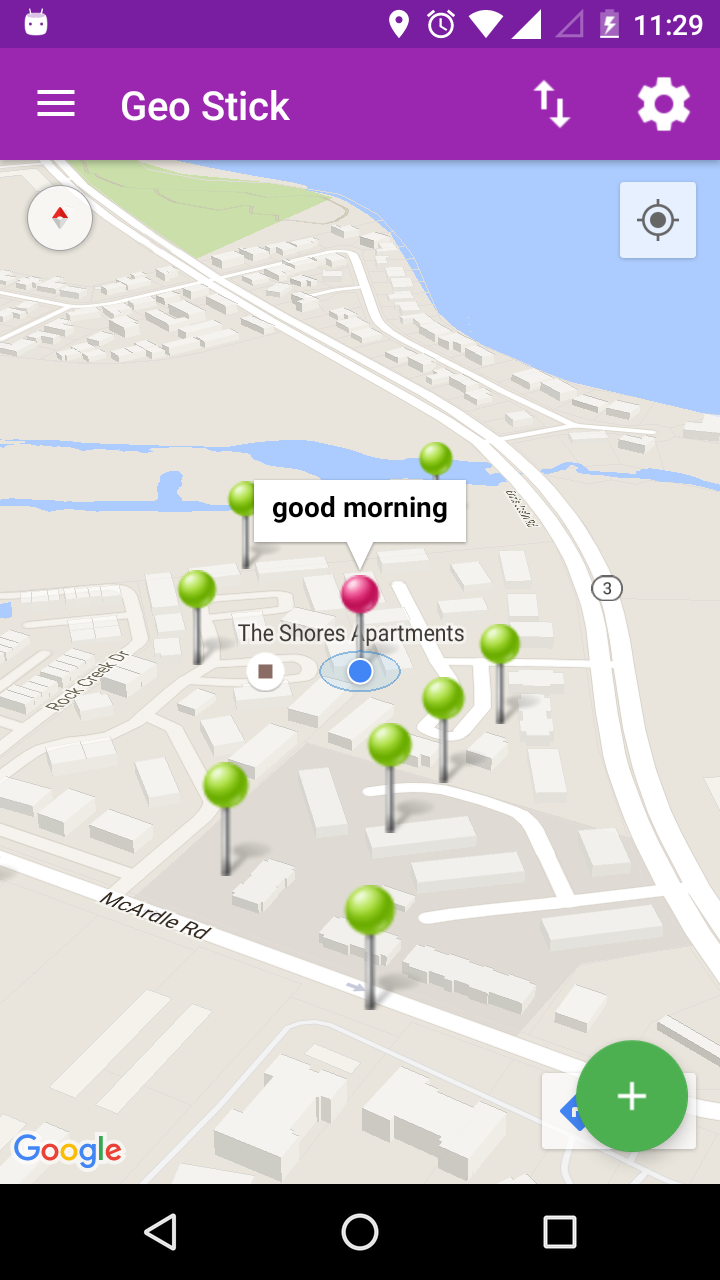
Step 3

Hit the floating action button to create new geosticks a new activity to fill all the details appears



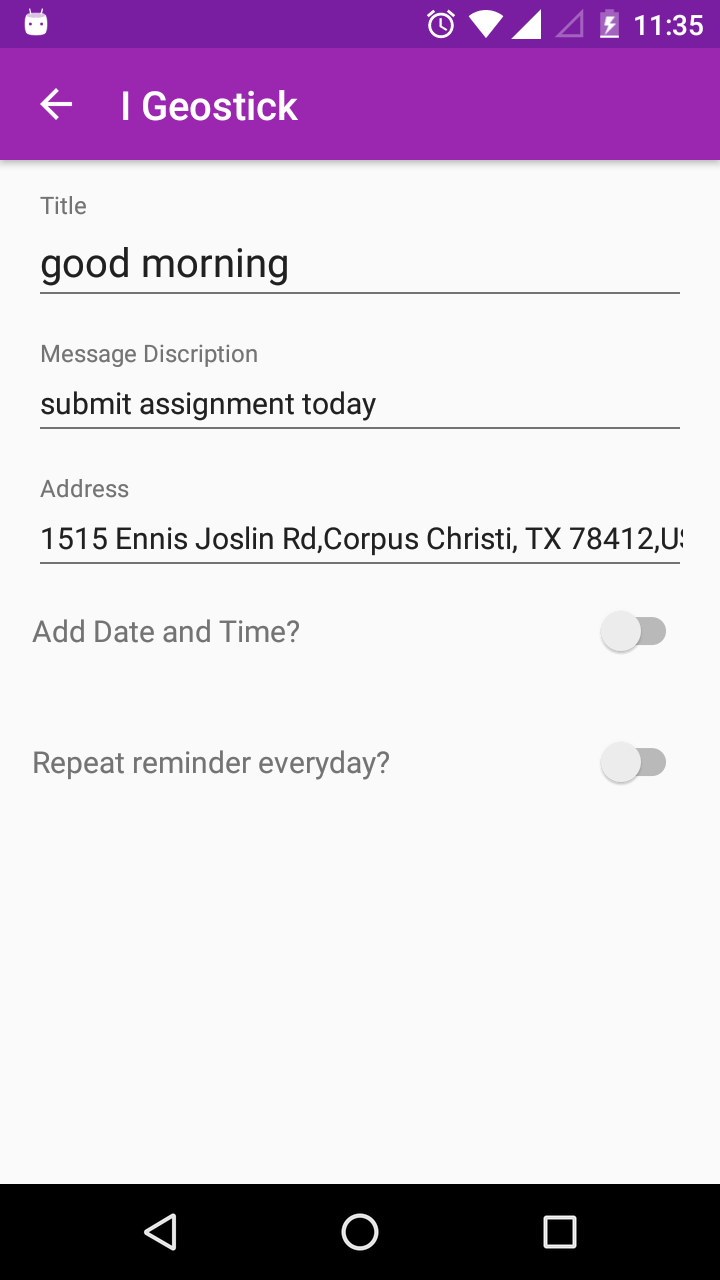
Step 4

After saving the geosticks it is redirected to home activity where a new geosticks appears



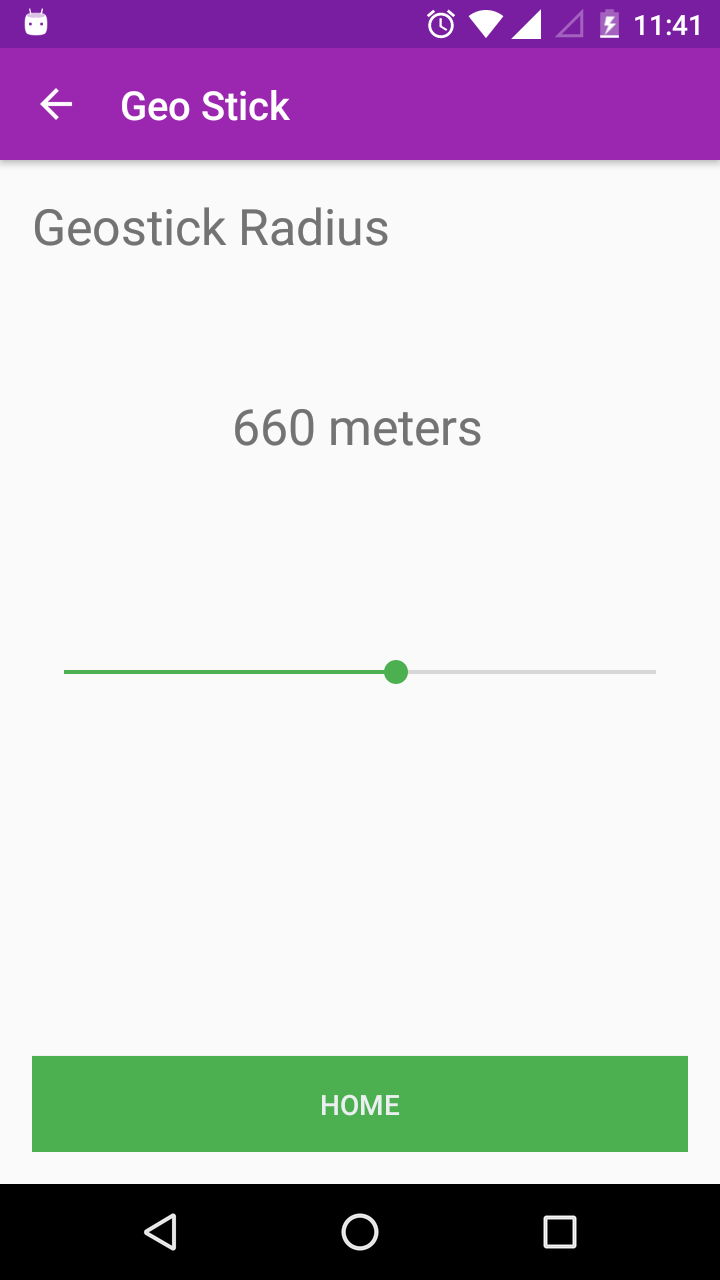
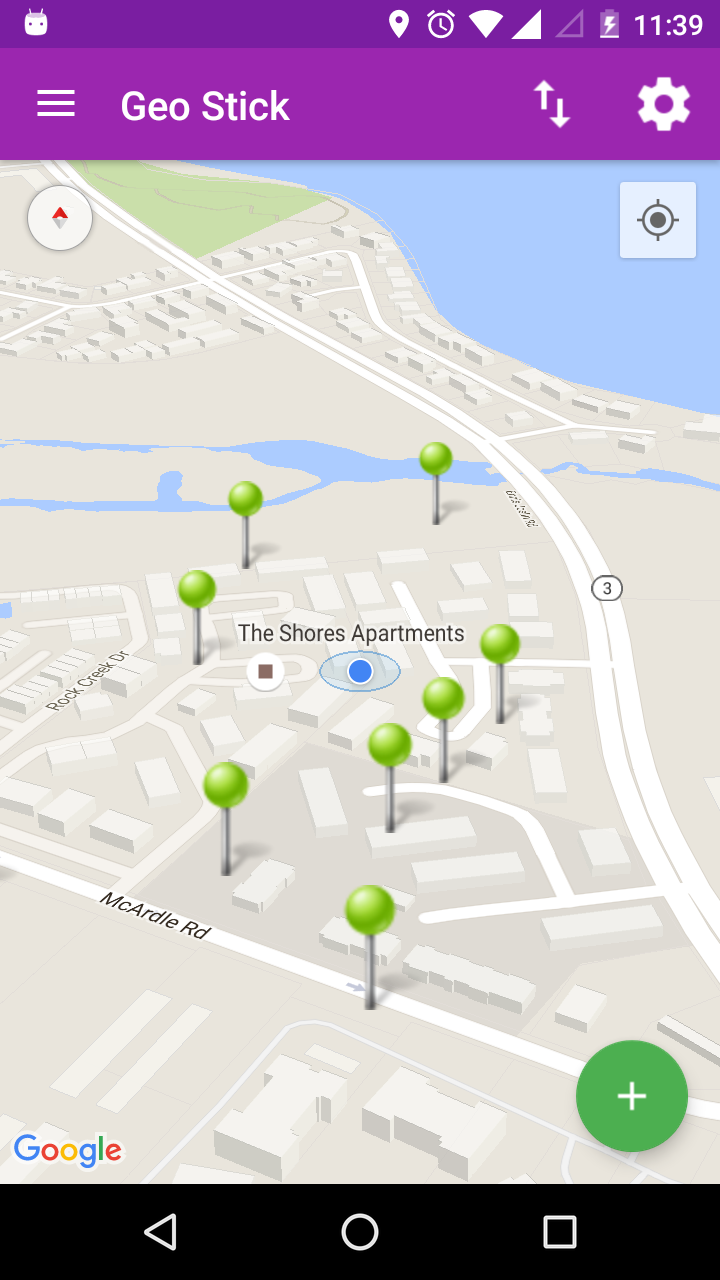
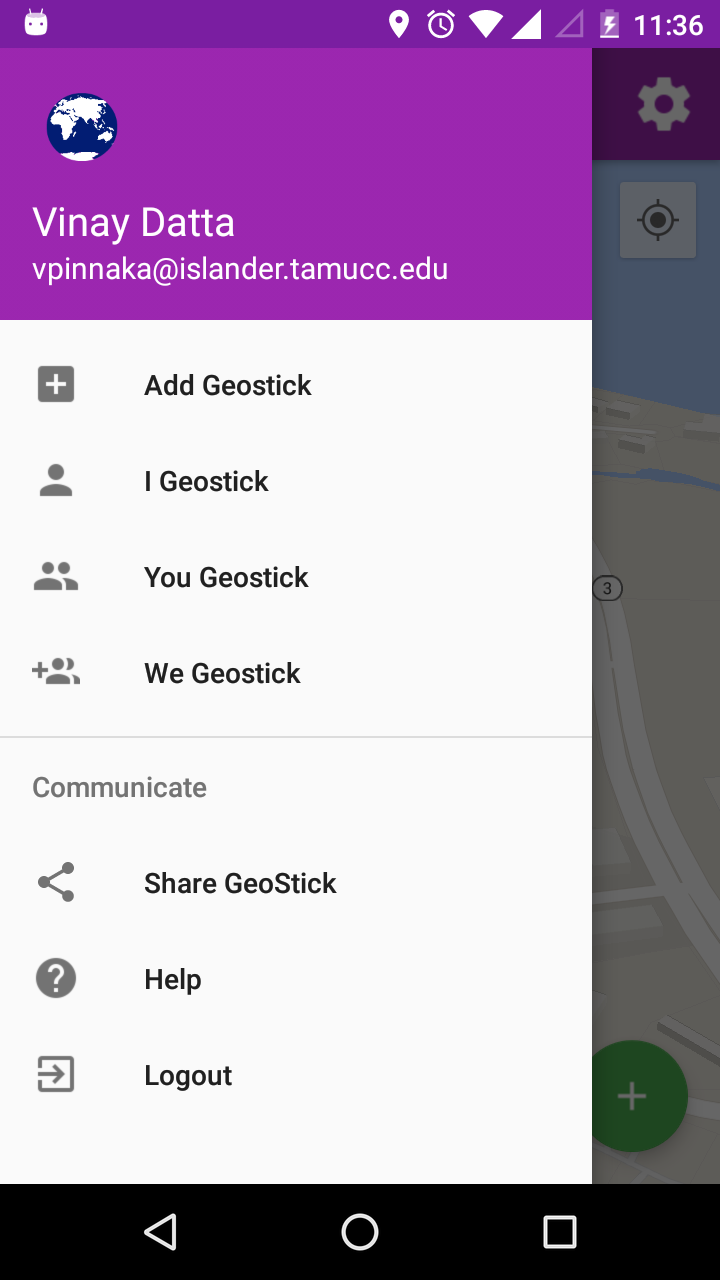
Step 5

Hit the info window to view the complete details of the geosticks



Step 6

User can select the type of geosticks from the navigation drawer and geosticks related to that are shown up for suppose here we geosticks are populated



Step 7

User can set the radius for the gesostick in settings activity top right corner.

1. **CONCLUSION**

Geo stick makes location based reminders more lively on the map and helps the user to organize reminders with ease.