FriendsTrackerWorldwide (FTW) – Android Application Smartphones

Project Documentation
in
Reutlingen University
Alteburgstraße 150
72762 Reutlingen

developed by
Philipp Bahnmüller
Felix Rosa
Kevin Schrötter

Advisor
Natividad Martinez

Inhalt

Abstract

This paper gives an overview of the FriendsTrackerWorldwide Android App and its functionality. It will illustrate the purpose of the app and explain the user interface, user input handling, the feedback system and activities as well as basic functions of FriendsTrackerWorldwide (FTW).

1 Purpose

FTW is designed to use a smartphones GPS-location to sent it to a MongoDB database and draw the users‘ position in a MapView from osmdroid, an OpenStreetMap tool for android. The user can then add and remove his own Markers on the map provided by a description that he can write himself to describe the marker.

A friend management system using friendrequests that need to be confirmed by the counterpart is included, which supports adding other users of the app to the account to see the friends positions together with their markers, if they decided to add some to their account.

Furthermore a user is able to display a list of every marker on the map, except his own position. The items of the list are sorted by their distance to the users’ own position. When clicking on an item of the list, the map will center the selected marker so that it is possible to directly see where its location is.

This enables to see all markers and their distance to users own position, so that it is possible to track those locations.

2 Requirements

There are a few things to consider when running the app.

2.1 Device

FTW is designed for smartphones that have an internet connection and GPS enabled

2.2 Operating System and Version

FTW requires an Android operating system.

The Minimum Build Version is:

- Android Build Version 23
- Android 6.0 (Marshmallow)

2.3 Permissions

To run the app properly, you need to allow internet access and GPS-location when installing the application. In detail:

- NETWORK_STATE
- WIFI_STATE
- COARSE_LOCATION
- INTERNET
- WRITE_EXTERNAL_STORAGE

3 Activities

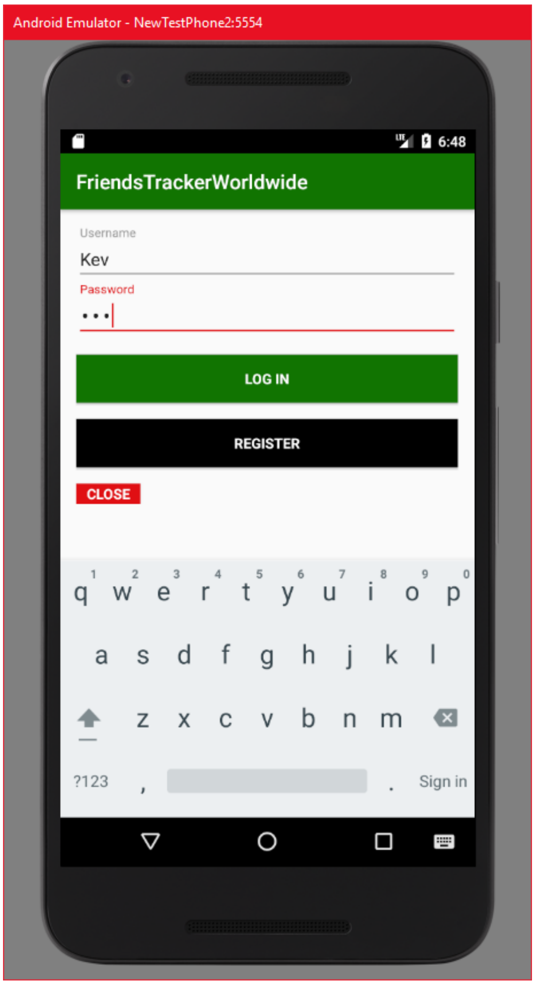
FriendsTrackerWorldwide uses 7 activites for its computing.

- LoginActivity
- MainActivity
- FriendActivity
- AddFriendActivity
- NotificationActivity
- RequestActivity
- LegendActivity

3.1 LoginActivity

This activity is used for user registration and user login.

3.1.1 Design



Android Emulator - NewTestPhone2:5554

FriendsTrackerWorldwide

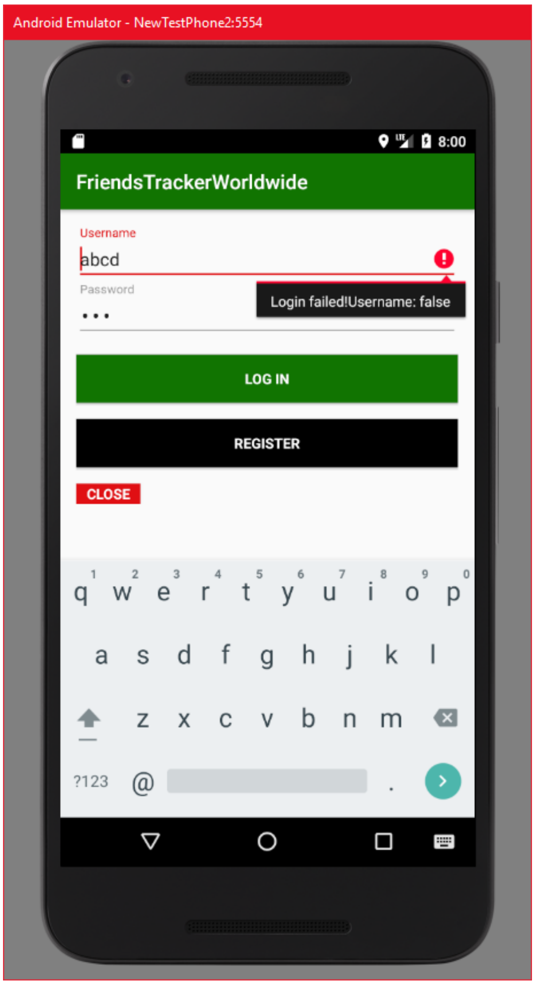
Username: Kev

Password: •••

3.1.2 Interface interaction

3.1.2.1 Username field and password field

These fields will show error messages, when the entered input does not match the restrictions or the registration/login attempts fail:



Android Emulator - NewTestPhone2:5554

FriendsTrackerWorldwide

Username: abcd

Password: •••

Login failed!Username: false

3.1.2.2 "LOG IN" button

By clicking on the login button, the app will start the login process calling ServerAdress + api/user/loginUser.

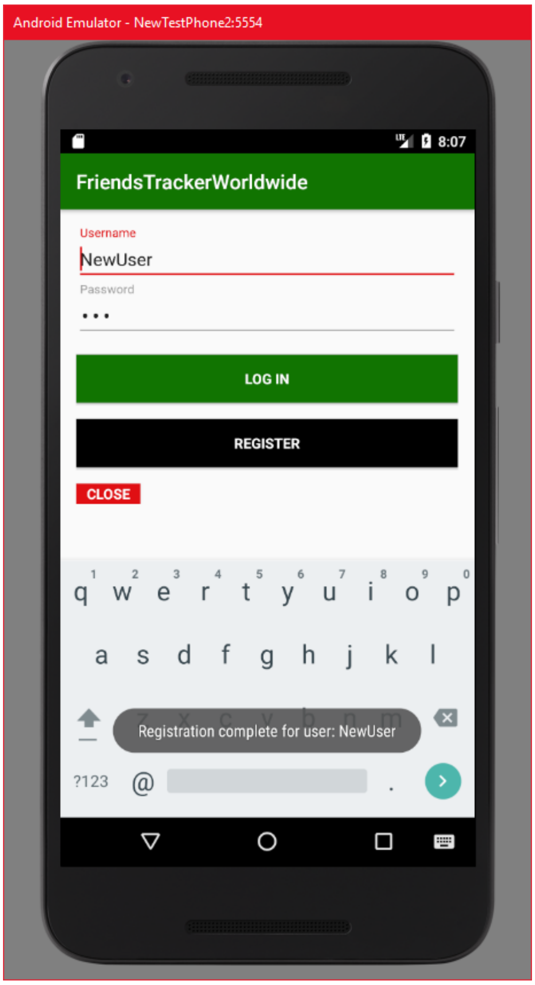
If the attempt is successful, the MainActivity is called and the initialization process starts.

If the login attempt fails, an error message as shown in 3.1.2.1 will be displayed.

3.1.2.3 "REGISTER" button

Clicking on register will attempt a registration in the database for the user calling ServerAdress + api/user/addUser.

If the attempt is successful, the user gets a Toast displaying that the registration was successful and the user can proceed to log in:



Android Emulator - NewTestPhone2:5554

FriendsTrackerWorldwide

Username: NewUser

Password: •••

Registration complete for user: NewUser

A failed attempt displays an error message as shown in 3.1.2.1.

3.1.2.4 "CLOSE" button

The "CLOSE" button will close the application.

3.1.3 Activity Calls

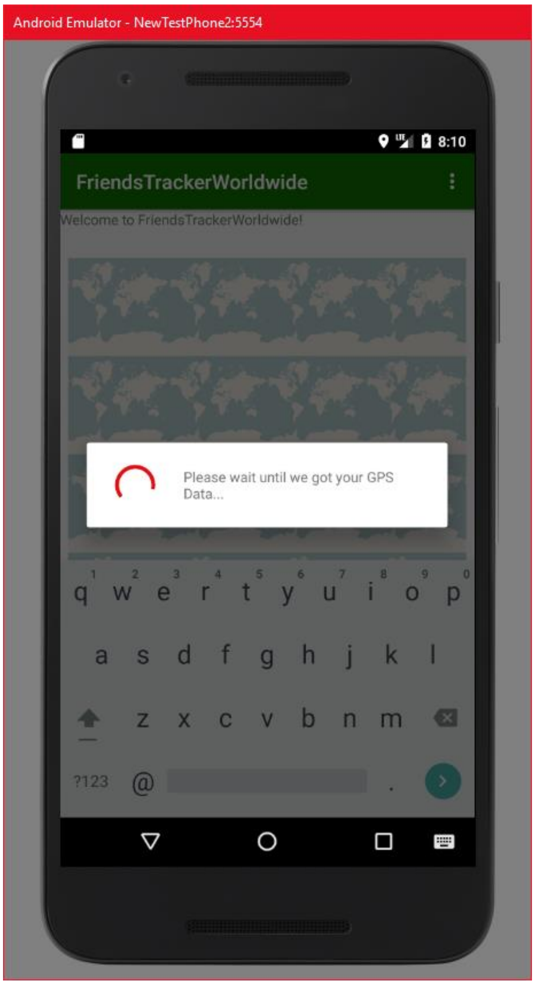
- MainActivity

3.2 MainActivity

MainActivity is the core of the app. It handles interaction with the map and invications of the other activities.

3.3 Initialization

The activity will start by waiting for the device to send GPS-data:



Android Emulator - NewTestPhone2:5554

8:10

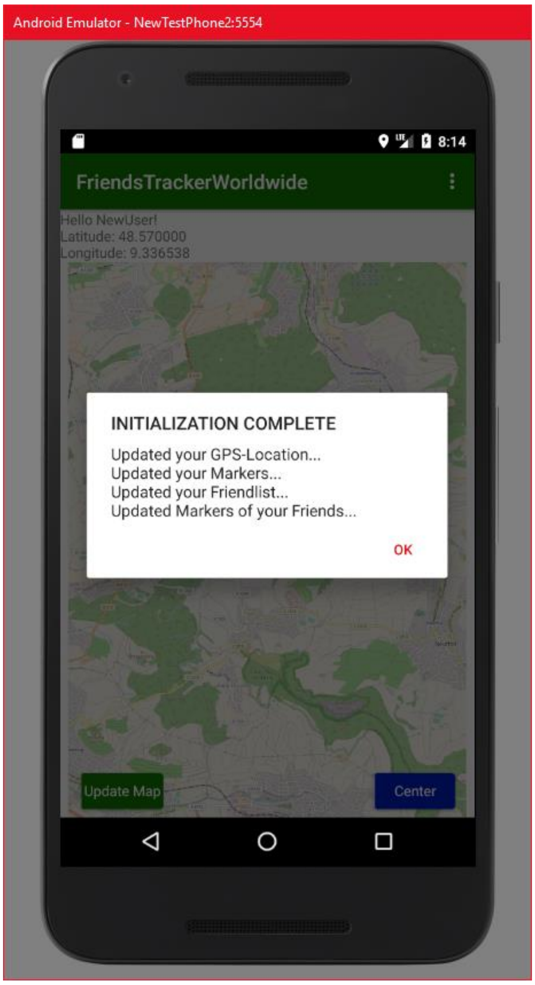
FriendsTrackerWorldwide

Welcome to FriendsTrackerWorldwide!

Please wait until we got your GPS Data...

As soon as the app got the GPS-location of the smartphone, it will start to fulfill all required tasks such as saving location information in the database, receiving friends, friend markers, and the users marks from the database. Also it checks the amount of friends and open requests to either show or hide the “Requests” button.

On a successful initialization, a dialog will be displayed showing that the initialization is complete:



Android Emulator - NewTestPhone2:5554

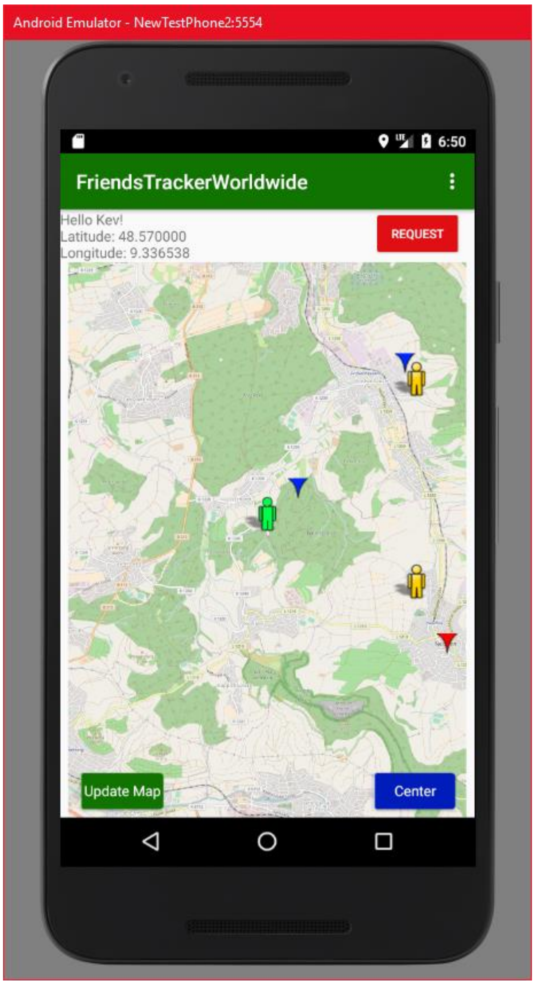
Hello NewUser!
Latitude: 48.570000
Longitude: 9.336538

INITIALIZATION COMPLETE

Updated your GPS-Location...
Updated your Markers...
Updated your Friendlist...
Updated Markers of your Friends...

After clicking “OK” the app will then load the initial design of the MainActivity.

3.3.1 Design



Hello Kev!
Latitude: 48.570000
Longitude: 9.336538

3.3.2 Interface interaction

This activity offers the following user interactions:

3.3.2.1 Location Textfield

This field shows the geo coordinats of the users current location.

3.3.2.2 "Map Interaction"

The user can interact with the map in different ways:

3.3.2.2.1 "Swiping on the map"

By swiping on the screen, the user can move the map. Also this will display the ZoomIn and ZoomOut buttons.

3.3.2.2.2 "Single Tapping Map"

Single tapping a point in the map will display the ZoomIn and ZoomOut buttons.

3.3.2.2.3 "ZoomIn" button

The ZoomIn button will zoom in the map.

3.3.2.2.4 “ZoomOut” button

The ZoomOut button will zoom out of the map.

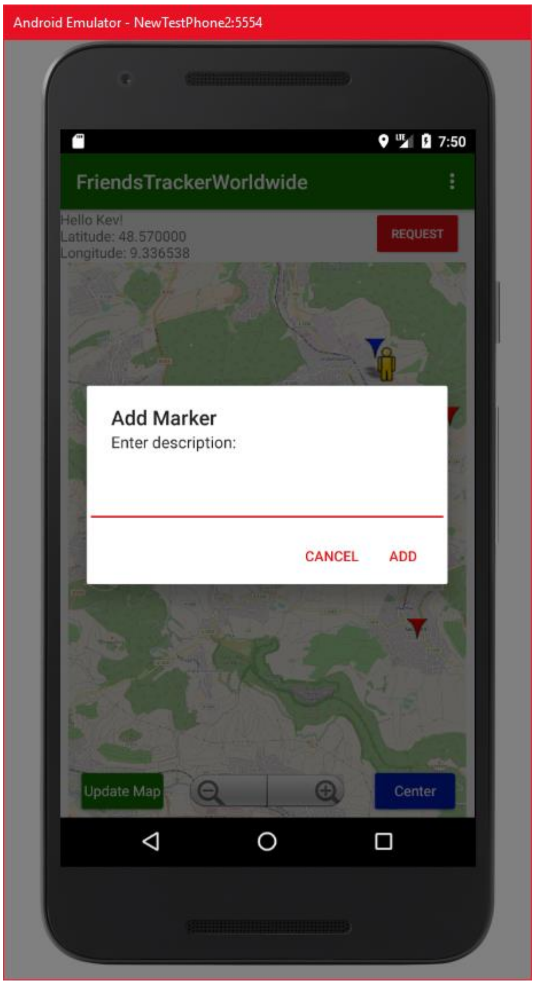
3.3.2.2.5 “Double Tapping Map”

Double tapping the map will zoom in.

3.3.2.2.6 “Long Tapping Map”

This will create a personal marker.

A Dialog field will pop up to enter a marker description:



Android Emulator - NewTestPhone2:5554

FriendsTrackerWorldwide

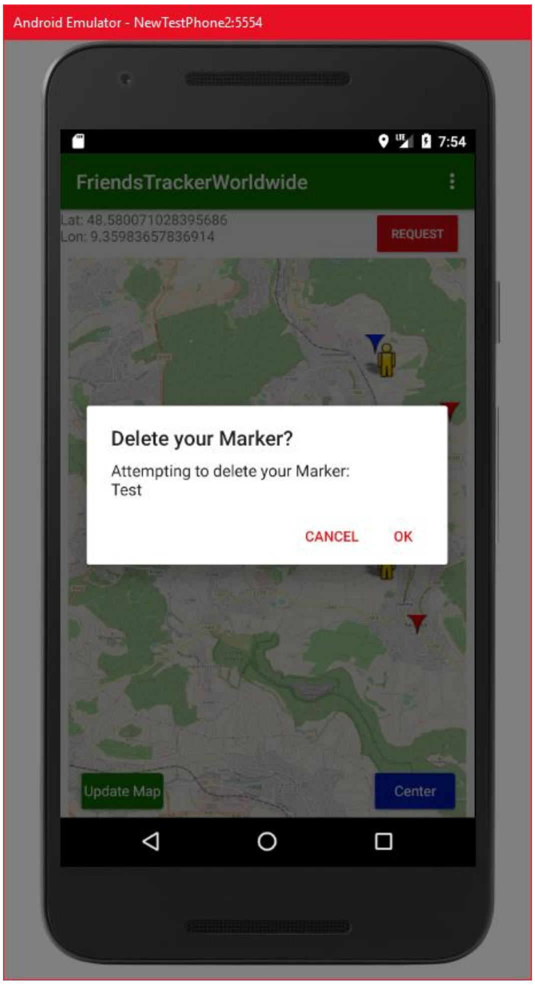
Hello Kev!
Latitude: 48.570000
Longitude: 9.336538

Add Marker
Enter description:

Clicking on “CANCEL” will return to the map, clicking on “ADD” will then create a new personal marker on the map and send its data to the database using ServerAdress + api/marker/addMarker and update the own marker information in the database and locally.

3.3.2.2.7 “Long Tapping personal Marker”

This will call a dialog field asking if one really wants to delete the marker:



Android Emulator - NewTestPhone2:5554

FriendsTrackerWorldwide

Lat: 48.580071028395686
Lon: 9.35983657836914

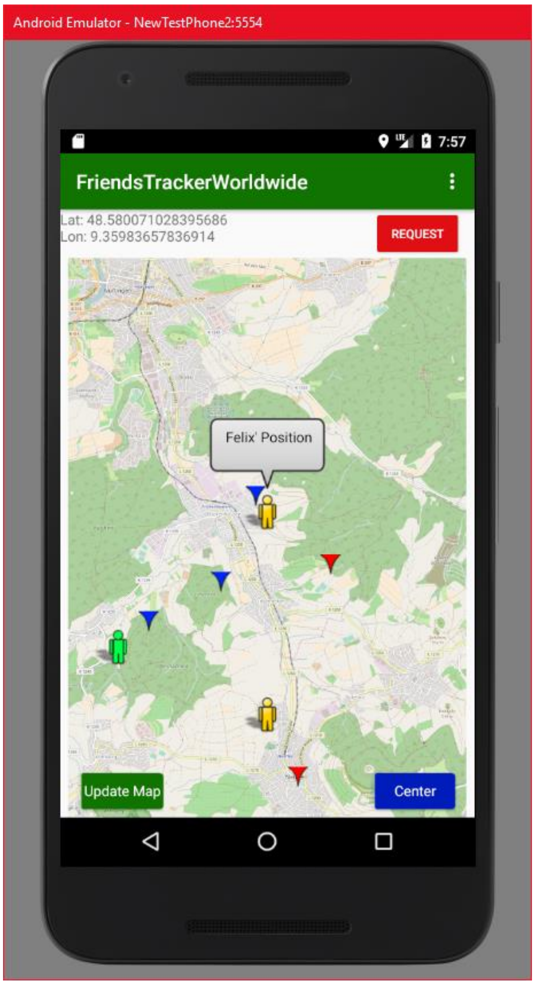
Delete your Marker?

Attempting to delete your Marker:
Test

Clicking on “CANCEL” will return to the normal map, clicking “OK” will delete the marker from the map and as well delete it from the database using ServerAdress + api/marker/deleteMarker.

3.3.2.2.8 “Single Tapping Markers or People”

By single tapping either the users own position, the users markers, friend position or friends markers the marker will display a bubble containing the markers description while centering the map at the tapped element:



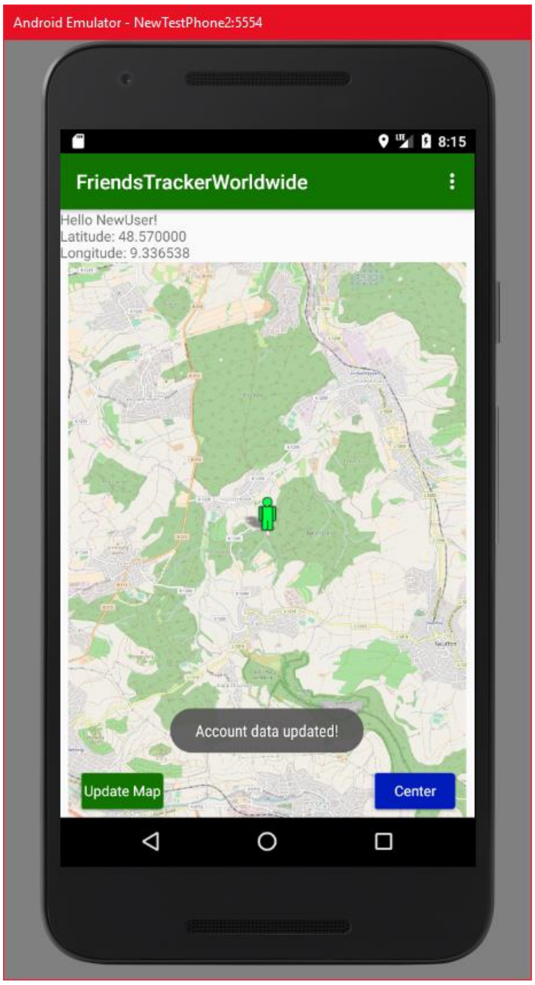
Another click on the bubble will make it disappear again.

3.3.2.3 "Update Map" Button

The "Update Map" button will update all information in the database including:

- Send user location calling ServerAdress + api/user/updateUser
- Update Friends and Friend Position calling ServerAdress + api/user/getFriendsLocation/:username
- Update Friends Marker calling ServerAdress + api/marker/getFriendsMarker/:username
- Update own markers calling ServerAdress + api/marker/getMyMarker/:username

When the update was successful, the app will show a Toast showing that the data has been updated:



Android Emulator - NewTestPhone2:5554

8:15

FriendsTrackerWorldwide

Hello NewUser!
Latitude: 48.570000
Longitude: 9.336538

Account data updated!

If it fails, another Toast will show that something went wrong, displayed at the same position as the successful Toast.

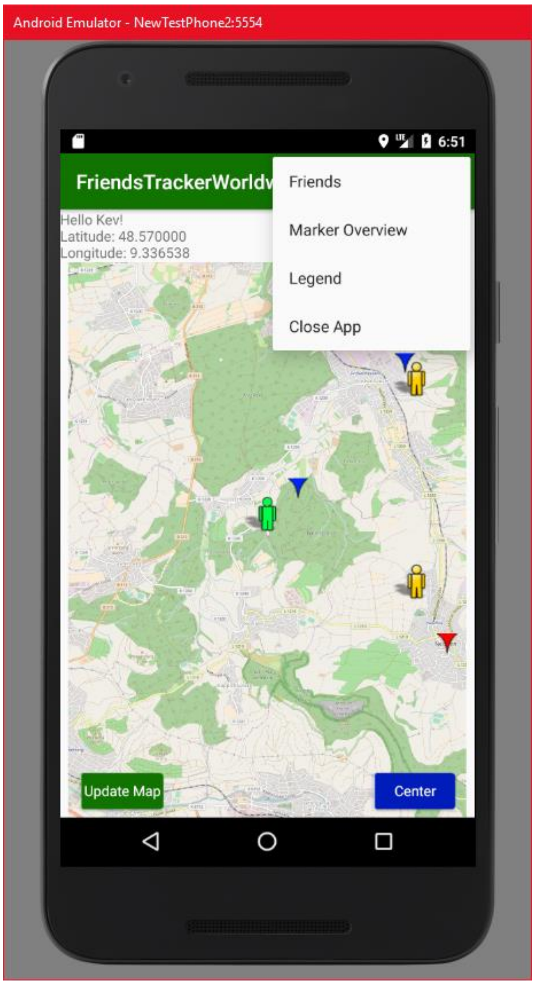
3.3.2.4 “Center” Button

By clicking the “Center” Button, the map will focus on the users position.

3.3.2.5 “Requests” Button

This button opens the RequestActivity. The Button only appears, when the user got requests. If not, the button will simply be hidden.

3.3.2.6 *Menu Bar*



Hello Kev!
Latitude: 48.570000
Longitude: 9.336538

The MenuBar on the top right corner is used to open the Friends (FriendsActivity), Marker Overview (NotificationActivity), Legend (LegendActivity) and closes the app to return to the login screen.

3.3.2.6.1 “Friends” Button

This button will open the FriendActivity.

3.3.2.6.2 “Marker Overview” Button

By clicking on the “Marker Overview” button, the app will calculate the distances from users location to users markers, friends positions and friends markers and then call the NotificationsActivity.

For updating information, the following FriendsTrackerWorldwide API methods are called:

- ServerAdress + api/users/getFriendsLocation/:username
- ServerAdress + api/marker/getMyMarker/:username
- ServerAdress + api/marker/getFriendsMarker

3.3.2.6.3 “Legend” Button

By clicking the “Legend” Button, the app will open the LegendActivity.

3.3.2.6.4 “Close App” Button

This will close the MainActivity and return to the LoginActivity.

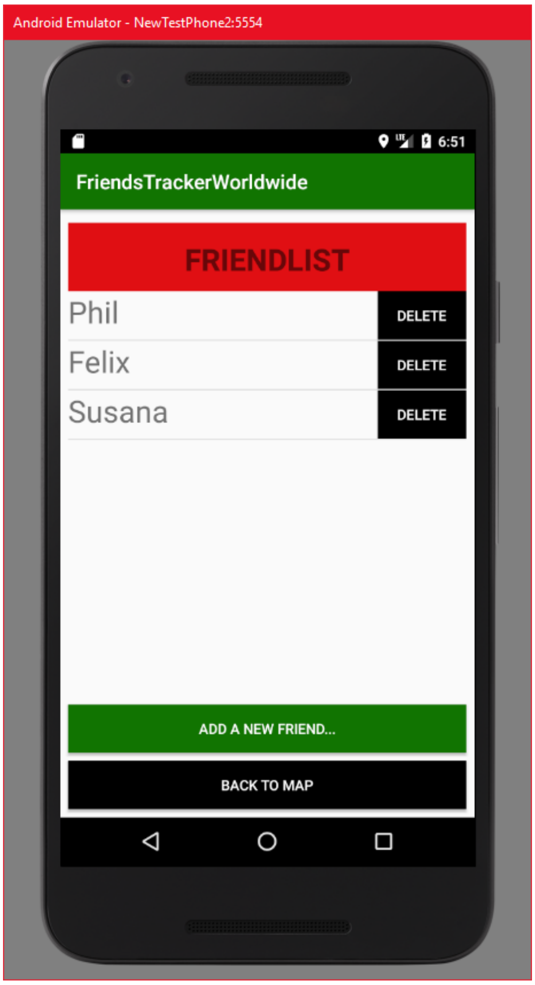
3.3.3 Activity Calls

- FriendActivity
- NotificationActivity
- RequestActivity
- LegendActivity
- LoginActivity

3.4 FriendActivity

The FriendActivity manages displaying the friendlist, sending friend requests to other users and deleting friends from the friendlist.

3.4.1 Design



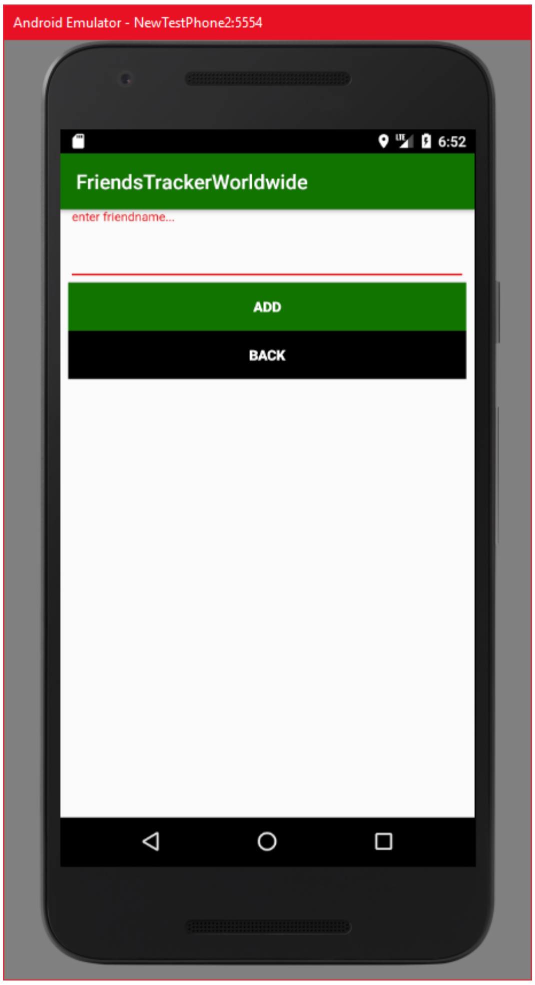
3.4.2 Activity Calls

- AddFriendActivity
- MainActivity

3.5 AddFriendActivity

This activity communicates with the database in the background to send friend requests and then updates the friendlist in FriendActivity and the requestlist in RequestActivity.

3.5.1 Design



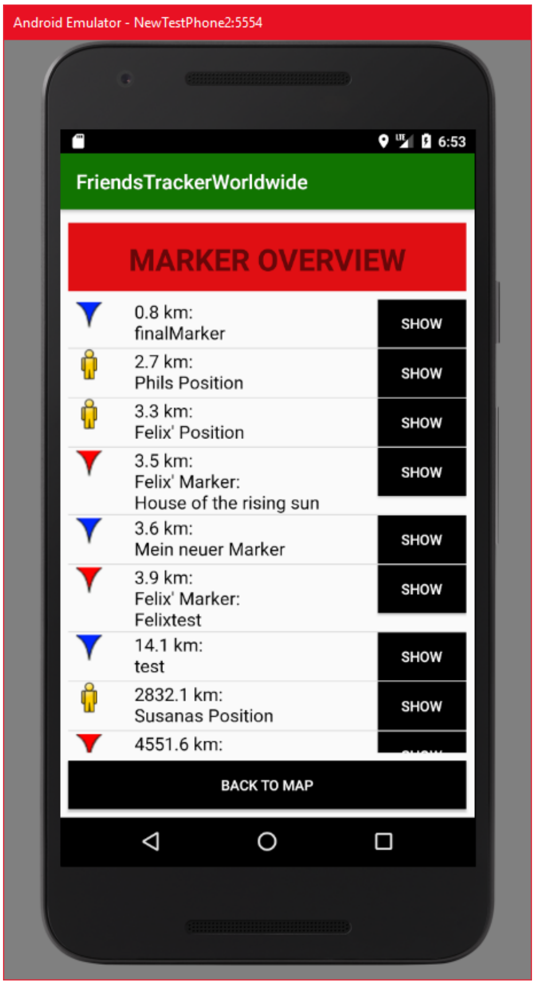
3.5.2 Activity Calls

- FriendActivity

3.6 NotificationActivity

NotificationActivity enables displaying all markers in a listView sorted by distance using a custom ArrayListAdapter.

3.6.1 Design



0.8 km:
finalMarker

2.7 km:
Phils Position

3.3 km:
Felix' Position

3.5 km:
Felix' Marker:
House of the rising sun

3.6 km:
Mein neuer Marker

3.9 km:
Felix' Marker:
Felixtest

14.1 km:
test

2832.1 km:
Susanas Position

4551.6 km:

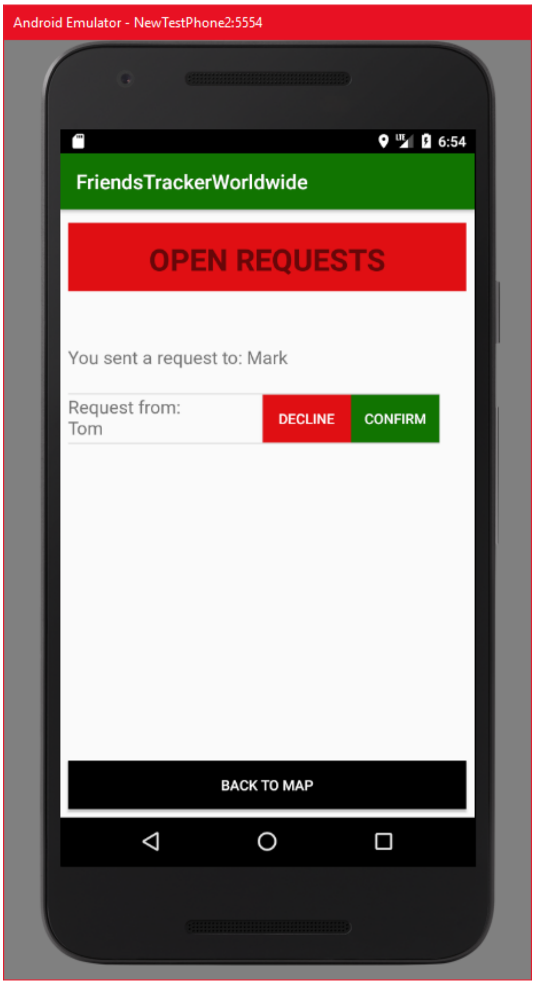
3.6.2 Activity Calls

- MainActivity

3.7 RequestActivity

This activity displays all open friend requests, which are either received or sent by the user, in a list using a custom ArrayListAdapter.

3.7.1 Design



You sent a request to: Mark

Request from:
Tom

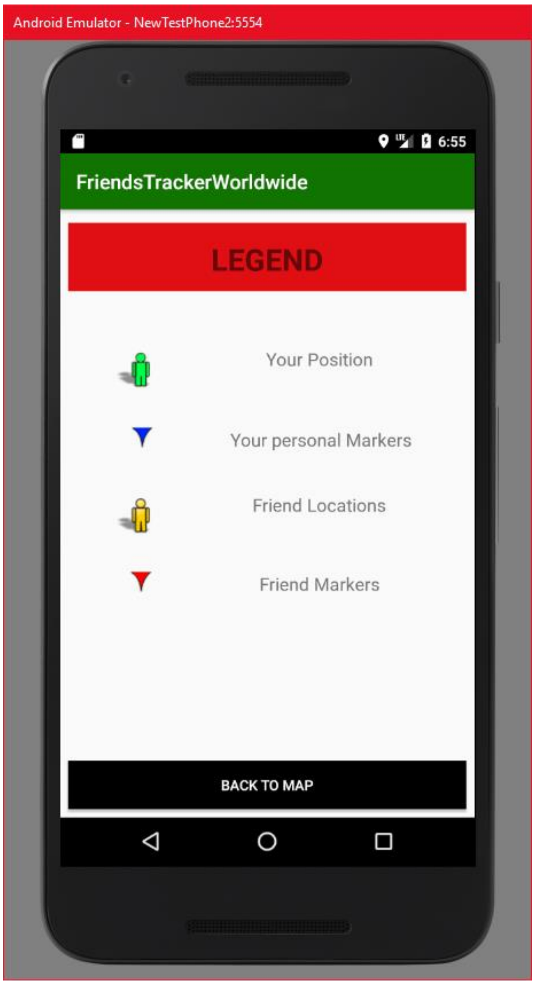
3.7.2 Activity Calls

- MainActivity

3.8 LegendActivity

This activity simply displays a legend of the symbols used on the mapView.

3.8.1 Design



3.8.2 Activity Calls

- MainActivity

4 Update Intervall

On the initial load when calling the MainActivity after login, the app will update the users location together with friends, requests, and friend markers by communicating with the database.

4.1 Users location

After the initial load the users location will locally update every 30 seconds by retrieving GPS data from the device, but this will not update the location in the database! The users location will only be sent to the database when manually pressing the “Update Map” button in the main activity.

4.2 Friends and friend markers

Friends and the friendlist as well as the friends markers will be updated locally and in the database everytime the user changes friend information. The changes will be applied when:

- removing a friend
- sending a friend request
- confirming a friend request
- denying a friend request
- pressing “Update Map” button

4.3 Personal marker

The personal markers will be updated locally and in the database everytime the user changes his own marker information. The changes will be applied when:

- adding a marker
- removing a marker
- pressing “Update Map” button

4.4 Marker Overview

The marker overview will always be updated when:

- user locally changes position
- friends get changed
- friend markers get changed
- personal markers get changed