Developer Document

Major parts and components

Main Window:
Here you will find the map which is a GoogleMap, which is what our whole app is about. The map will show the electric bus' GPS position. Identifying the bus is done with the help of the Wi-Fi on the electric bus which will fetch the correct system id for the bus. The app will then fetch all InfoNodes from a database on a Parse server, and will only show those that aren't unfiltered. These markers will also be sorted by their distance from the next bus stop and will be shown in the list below the map. In the list you can also add them to favorites by pressing the star icon at the far right of the item in the list. More importantly, clicking on a list item will take you to a more detailed view displaying info about the node.

Detailed/Information Window:
Here you can see information like text and pictures for the selected point of interest. More importantly, you will also be able to see sales and offers for the specific marker, e.g. 50% sale in a specific shoe store.

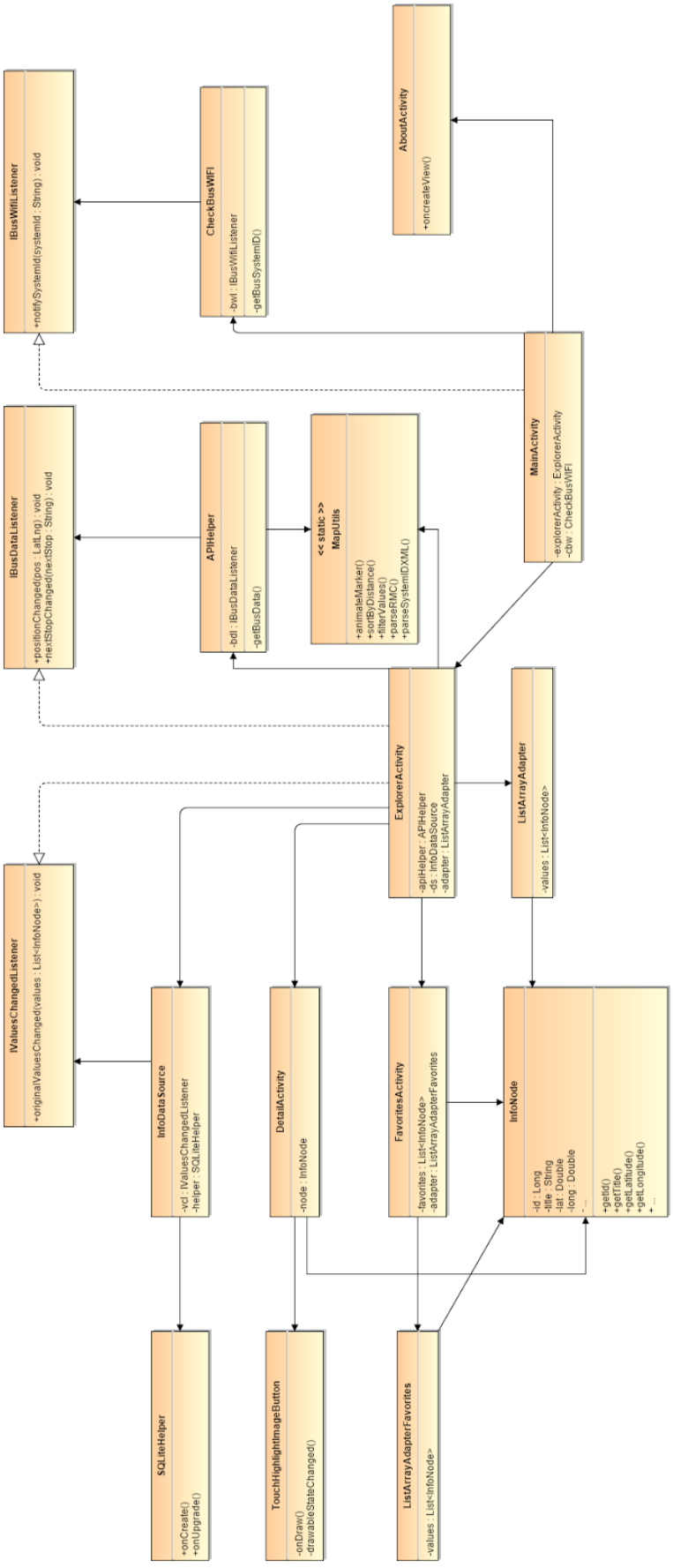
Design decisions

We wanted to select the newest possible API to be able to use the most modern Android features. However, there are still many Android phones that don't use Lollipop or above. So we had to find a compromise between features and audience. After a short discussion we decided to use API level 15 (which corresponds to Android 4.0.3 and above) because it offers a good feature base (such as fragments, the action bar among others) and good compatibility, since 94% of all Android phones use 4.0.3 or higher.

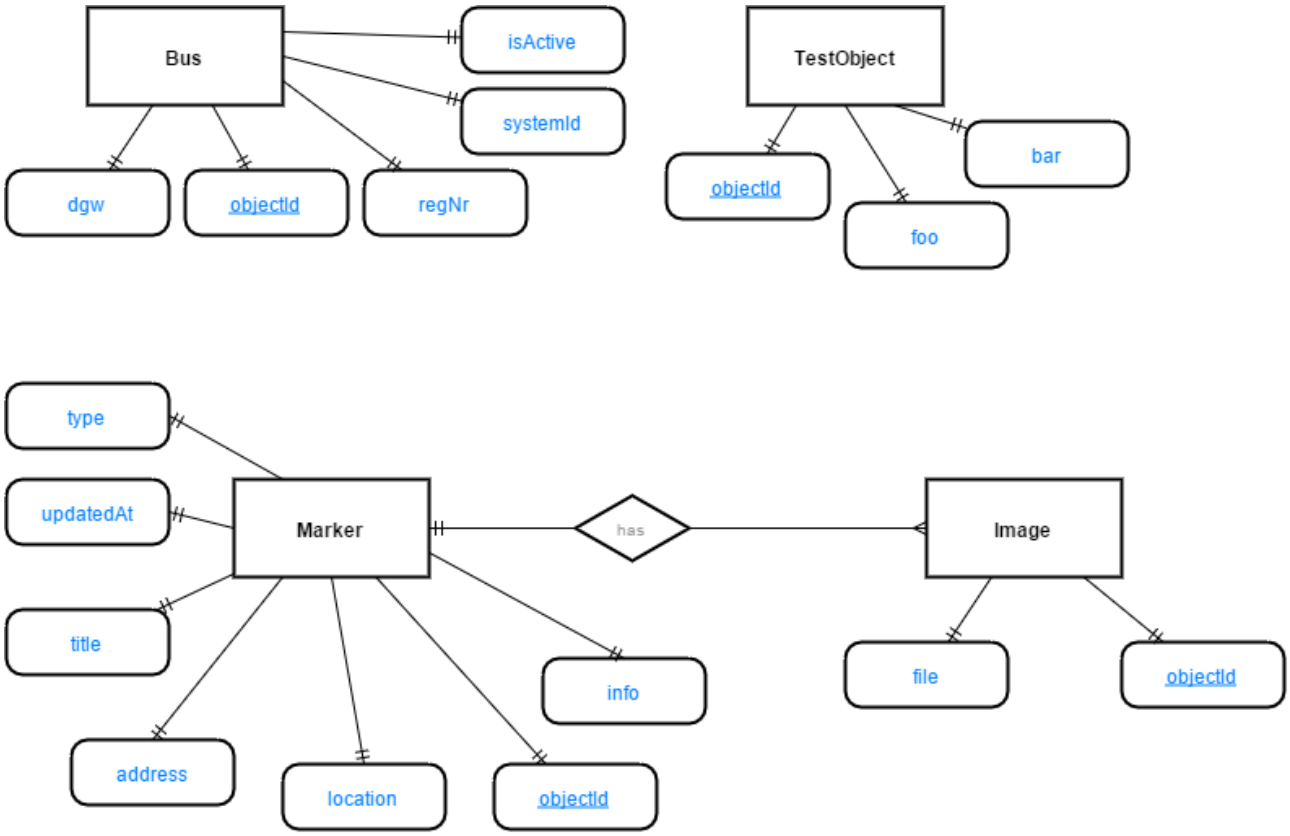
In the beginning of the project we had an activity for each of our main components, but after we decided to use a Navigation Drawer we had to convert all activities except the MainActivity into fragments in order to use with the Navigation Drawer. It led to some code refactoring and slightly different function-calls, but it worked out well in the end.

About halfway through the project our list of classes was growing with no end in sight. That led us to group the different classes into several different packages.

UML Class Diagram (Simplified)



Parse Server Database Structure (ER-Diagram)



The internal database only has the Marker entity.

External dependencies

- **Google Maps:** We use this to show a map with the bus the user is on and markers nearby.
- **Parse:** Is used to save data on our database on the server. Parse API is used to retrieve data.
- **hdodenhof:CircleImageView:** We use this one to get more attractive icons. It will crop the icon within a circle.
- **AppCompat:** We use this like a sort of base class that will make things work properly, like the ActionBar, but we mainly use it to insure backwards compatibility.
- **Support:design:** This dependency is used for our Navigation View we have in our mainView.