Public Transport Stops

Bartosz Ociepka, Rafał Mazur, Andrzej Lippa

December 2019

Link to the repository:
`https://github.com/bociepka/PublicTransportStops`

1 Concept

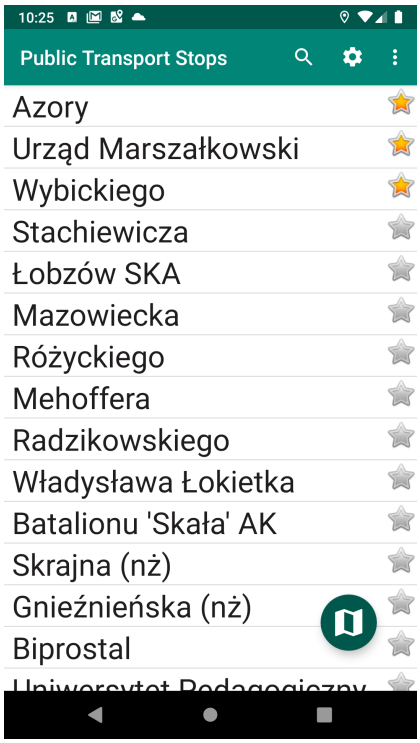
The main idea of our application is to provide information about public transport in Cracow (because there's a good API with that information for Cracow and we could not find one for Tartu). We want to give users access to information similar to one displayed on the displays next to bus and tram stops (Figure 1), but anywhere and anytime they want in their mobile phone

Figure 1: Example of tram stop departures display

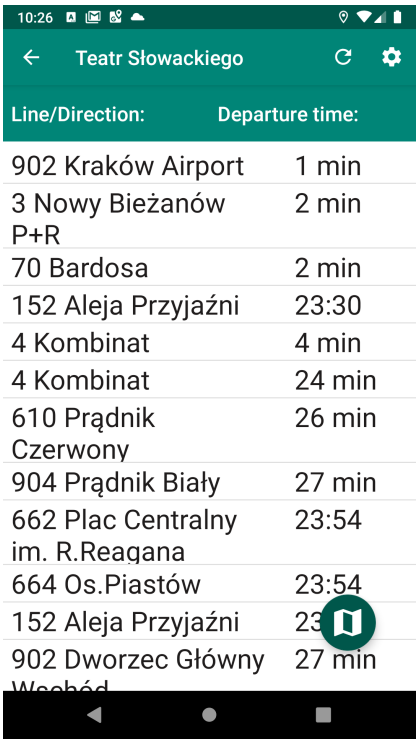
When opening the application user sees list of all the public transport stops in the city, sorted by name or distance from user's location, with user's favourite on the top of the list.

From here, user can eithter tap on a stop to show next departures from that stop or switch to map view, where stops and user location are displayed on the map and user can chose a stop from there.

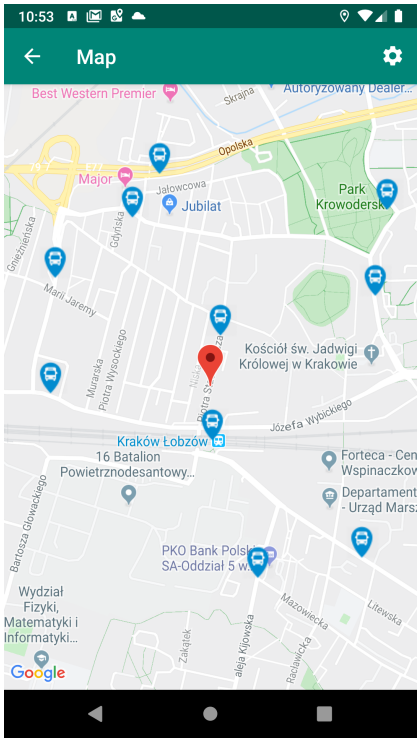
From departures view, user can also go to the map view, where the route from his location to the chosen stop is shown.



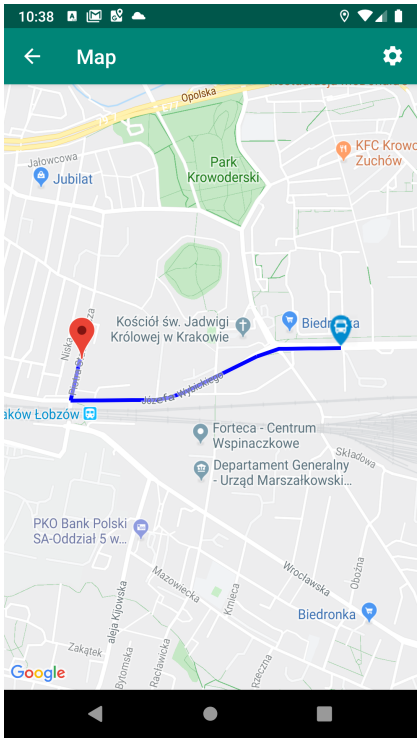
(a) Main Stops View



(b) Departures View



(c) Map Stops View



(d) Map route view

Figure 2: Main screens of the application

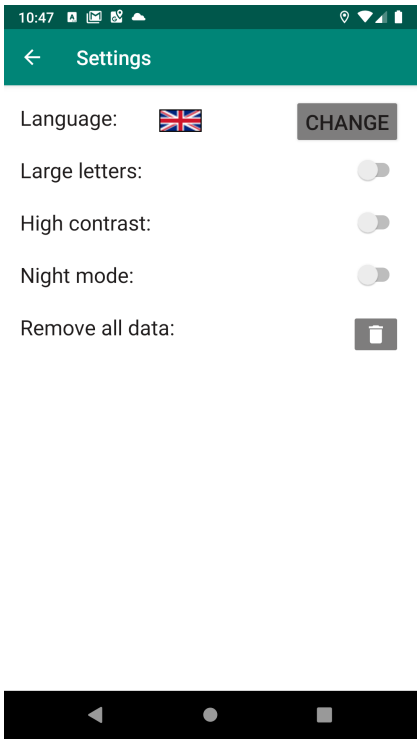
2 Core Blocks

2.1 Multiple screen views

Application has three different views: Stops view, Departures view and Map view

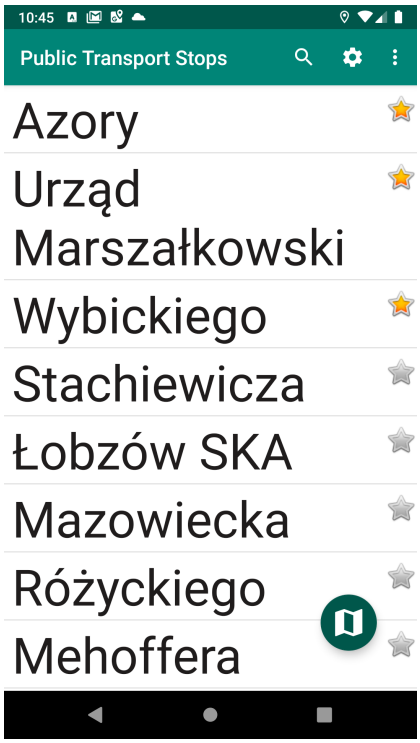
2.2 Settings screen

Application has settings menu, which allows user to change language, switch to high contrast mode or dark mode, change text size to bigger (to help people with poor eyesight use the app) and clear the application data.

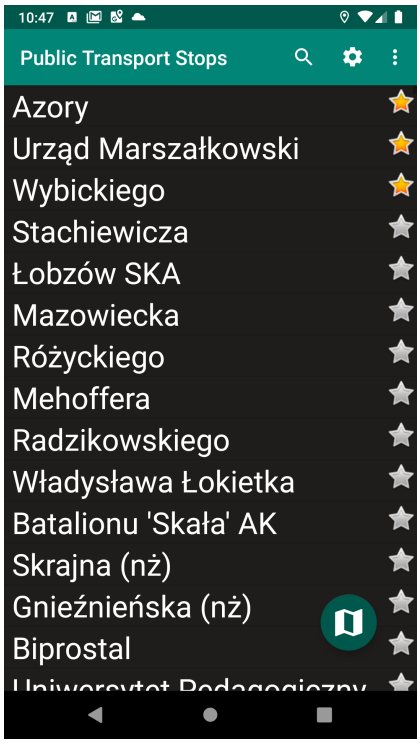


(a) Settings view

(b) High Contrast Mode



(c) Big letters mode



(d) Dark mode

Figure 3: Settings screen and results of using different settings

2.3 Lifecycle and permissions managment

The only dangerous permission required by our app is access to device's location. Regarding lifecycle managment, our application makes sure, that user's flow is not interrupted by rotating the device, sending the app to background etc. For example, filtering query and results are saved when rotating the device, so the user can continue using the app from the same state it was before. Information about stops is kept in Room database so it is stored even when user closes the app.

2.4 Background worker mechanism

Data loading in stops and departures activities is asynchronous - it is managed by Volley library. In addition to that, we use working thread for loading and displaying route in the maps activity.

3 Feature Blocks

3.1 Persistent storage

We are using Room Database to keep table with information about all the stops, so it is stored after closing the app. We also use Shared Preferences to keep user's settings. Data can be cleared in the settings menu.

3.2 Google Maps & GPS

Our application uses Google Maps in two situations:

- Users wants to chose stop from the map instead of the list - the maps is displayed, showing user's current location and markers for all the stops in the city. User can tap on one of the stops to be taken to departures activity showing departures for chosen stop.
- From the departures activity, user can chose to go to the map, where his location, location of chosen stop and route between them is shown (obtained from Directions API)

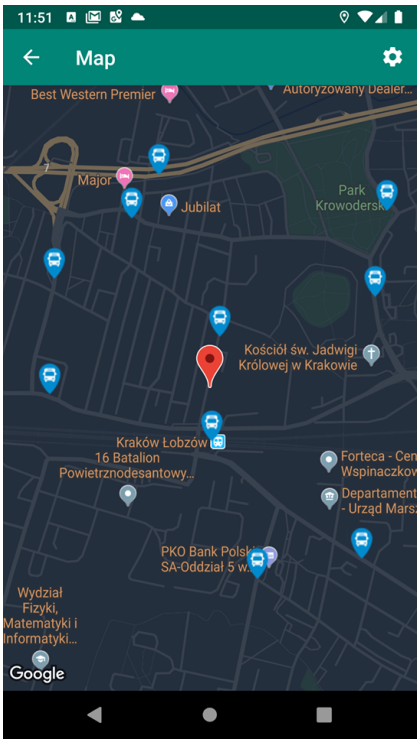


Figure 4: Map view in dark mode

3.3 REST API

We use simple REST API from `krakowpodreka.pl` to get list of all stops in the city (with their location) and later for getting future departures from stop with given ID. For managing the requests we use Volley library[1].

3.4 Language localization

Application is fully localized in 5 languages:

- English
- Polish
- Russian
- German
- Estonian

4 Development Process

4.1 Concept changes

Settings: main concept change was in the colorblind mode. After doing some reasearch in how many types of colorblindness there are, we decided to rename it to High-Contrast mode, because doing one mode for every kind of color vision deficiency would be to complicated. We hope that this palette of colors would make the application easier to read for some people. Second change in settings was not implementing separate change of font color. It was due to the fact that two other motives are already implementing this change.

4.2 Most challenging elements

One of the most challenging elements was dealing with applying changes in settings, which turned out to be more complicated than we assumed and caused some problems with application flow. Another difficult part was dealing with the fact that loading the data is asynchronous.

4.3 Future development

One of the things we definitely want to improve in the application is performance: at the moment stops activity is loading a little bit slowly - we could improve that by dynamicaly loading only part of the stops and not all of them at once. We could also make some UI fixes, to make sure that everything is very clear and good looking in all the themes and both font sizes.

References

[1] https://developer.android.com/training/volley