GitHub Username: brugienis

Melbourne Public Transport

Description

Problem:

Victoria's public transport network comprises train, tram, bus and coach services. These services are provided by independent operators via contracts managed by Public Transport Victoria.
Public Transport Victoria (PTV) is a statutory authority that acts as a system authority for all public transport and an advocate for public transport users. PTV is a single contact point for customers wanting information on public transport services, fares, tickets and initiatives.

The PTV timetables are not static - they are changing constantly. As a result, it is confusing and challenging to get the most up-to-date information for the users on the go.

Proposed Solution:

Design an app that allows a user to get the get accurate information about transport network.

In order to give users the above information, the app should use PTV Timetable API - the latest version is 2.2.0 and was released on 13 April 2016.

The information user needs should be selected at real time to get the most accurate details.

To get details of the PTV ASPI, use the following link:

https://ptv.vic.gov.au/about-ptv/ptv-data-and-reports/digital-products/ptv-timetable-api/ptv-timetable-api-reference/

Intended User

Anybody living in or visiting Melbourne (Australia, Victoria).

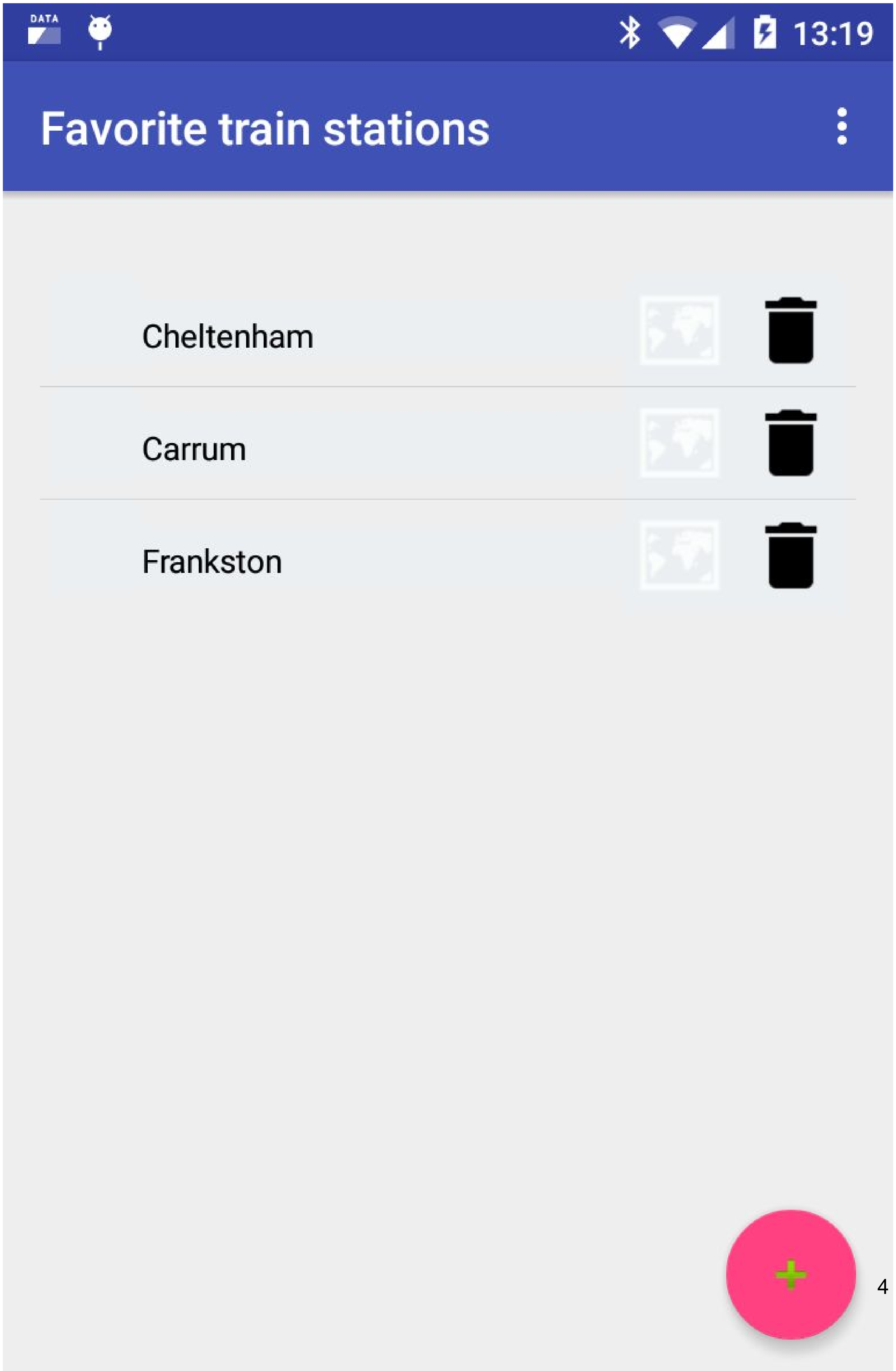
Features

The app shows the following up-to-date information:

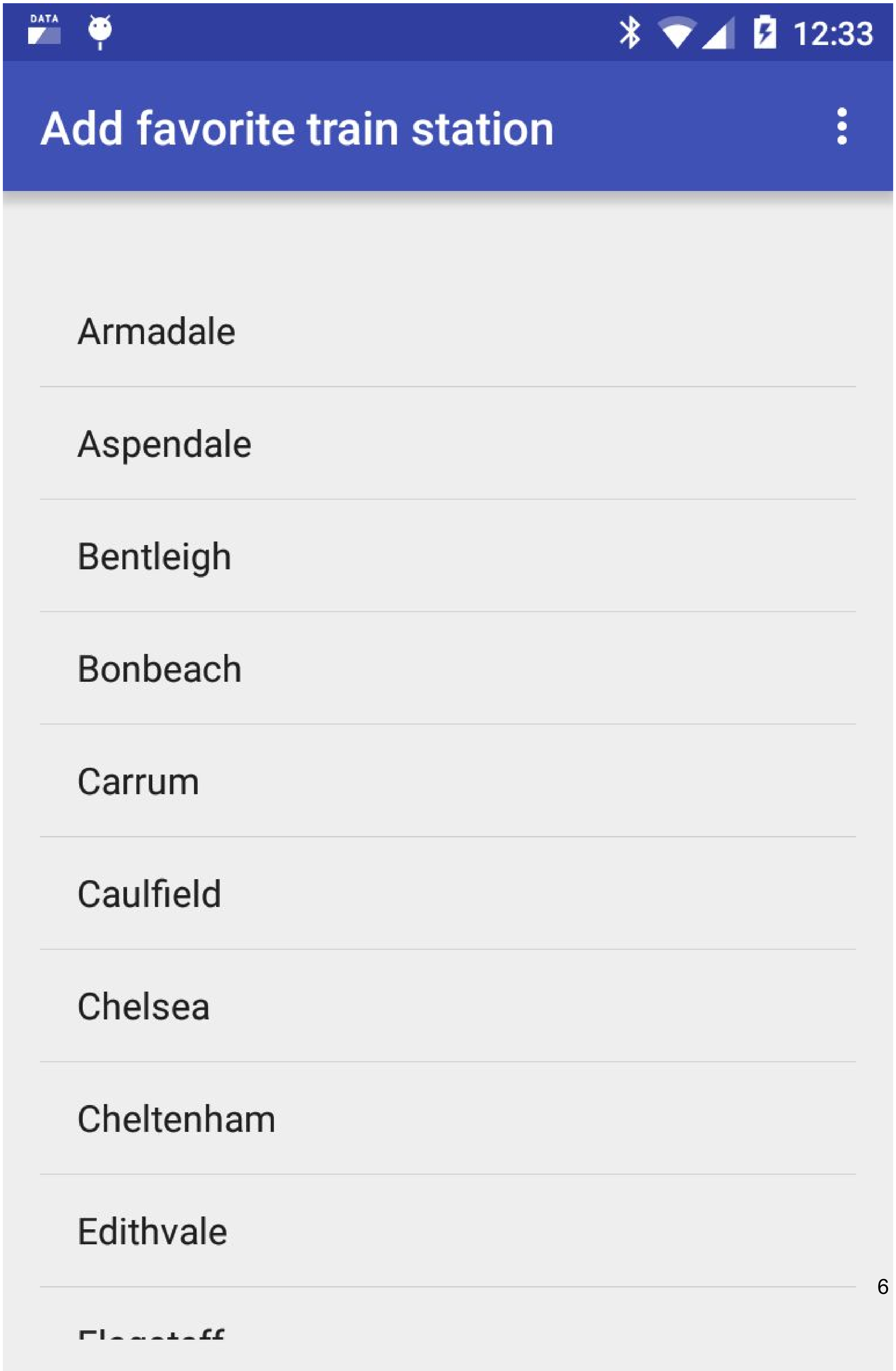
- Current departure times for selected station
- Current disruptions on the train network
- Stops nearby based on the device location
- Stations/stops on the Google Map

All the above is be for trains only except for nearby stops (the current version API does not allow to filter Stops Nearby response on the transport mode, e.g. train, bus, etc.).

User Interface Mocks


13:19
Favorite train stations
Cheltenham
Carrum
Frankston

Favorite stations selected by user. Touching the map image will show station on the Google map. Touching the trash image will remove station.



12:33

Add favorite train station

Armadale

Aspendale

Bentleigh

Bonbeach

Carrum

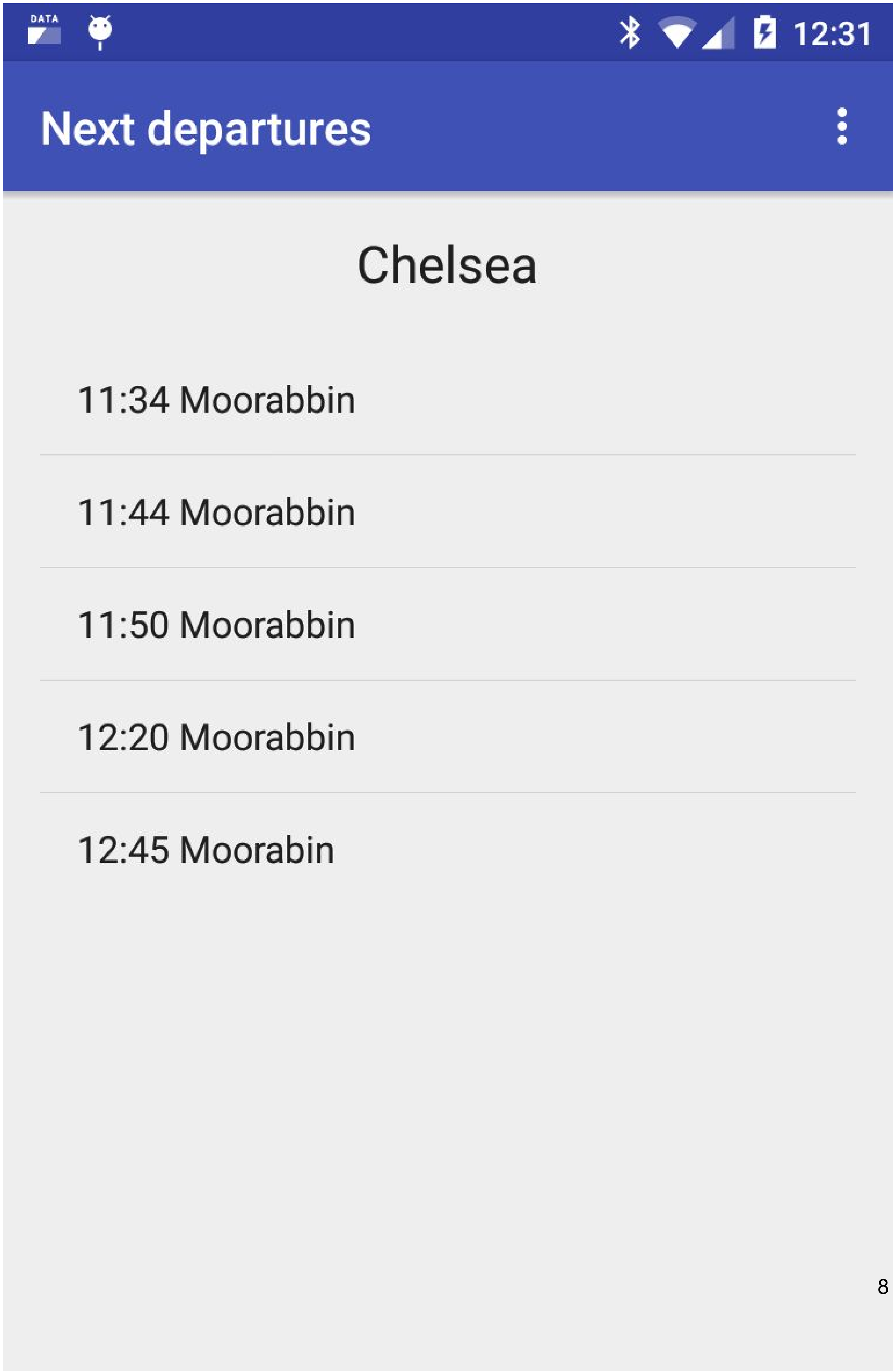
Caulfield

Chelsea

Cheltenham

Edithvale

Users can add favorite stations.



Chelsea

11:34 Moorabbin

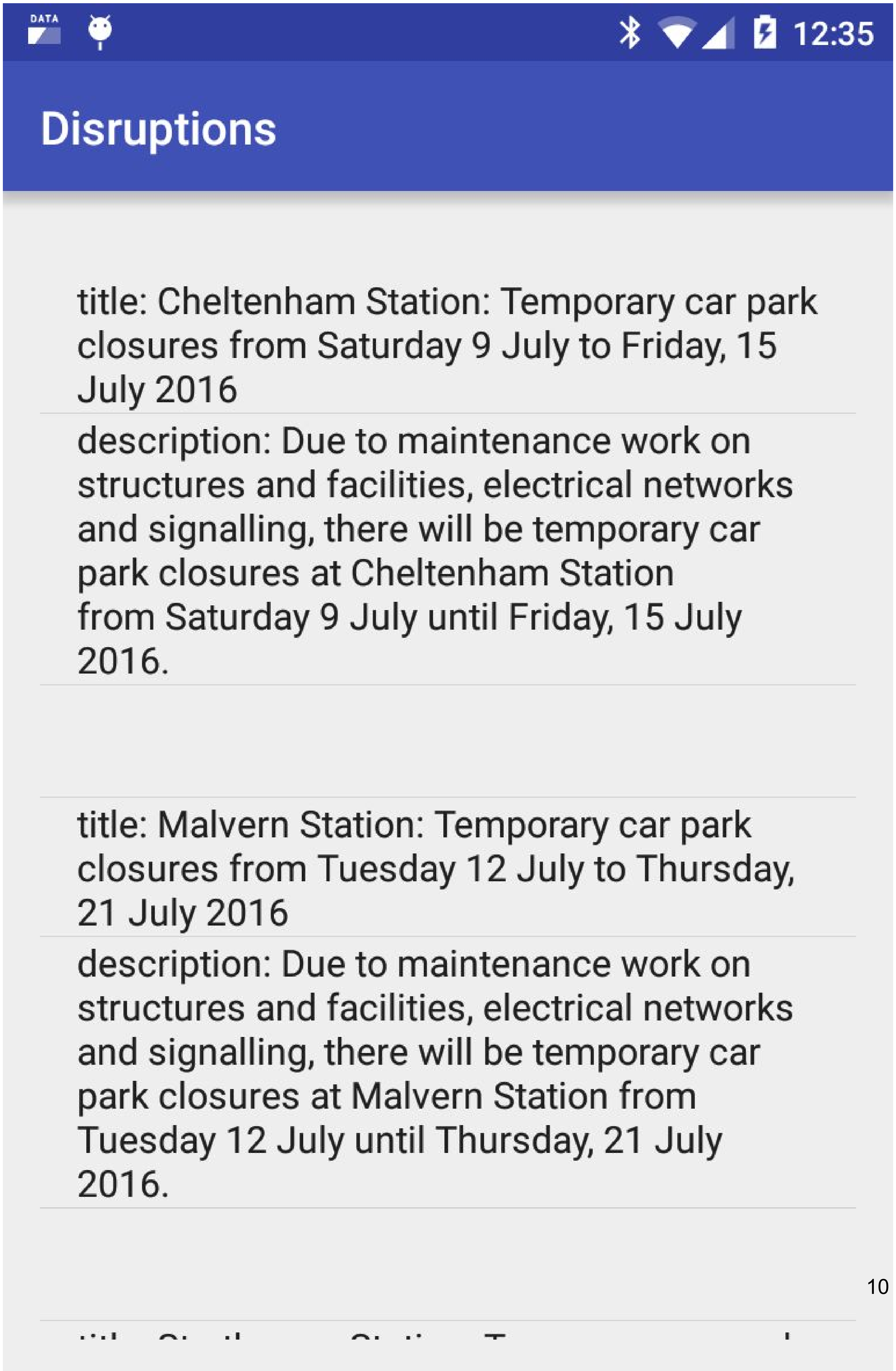
11:44 Moorabbin

11:50 Moorabbin

12:20 Moorabbin

12:45 Moorabin

Shows current next departure times for the selected station.



Disruptions

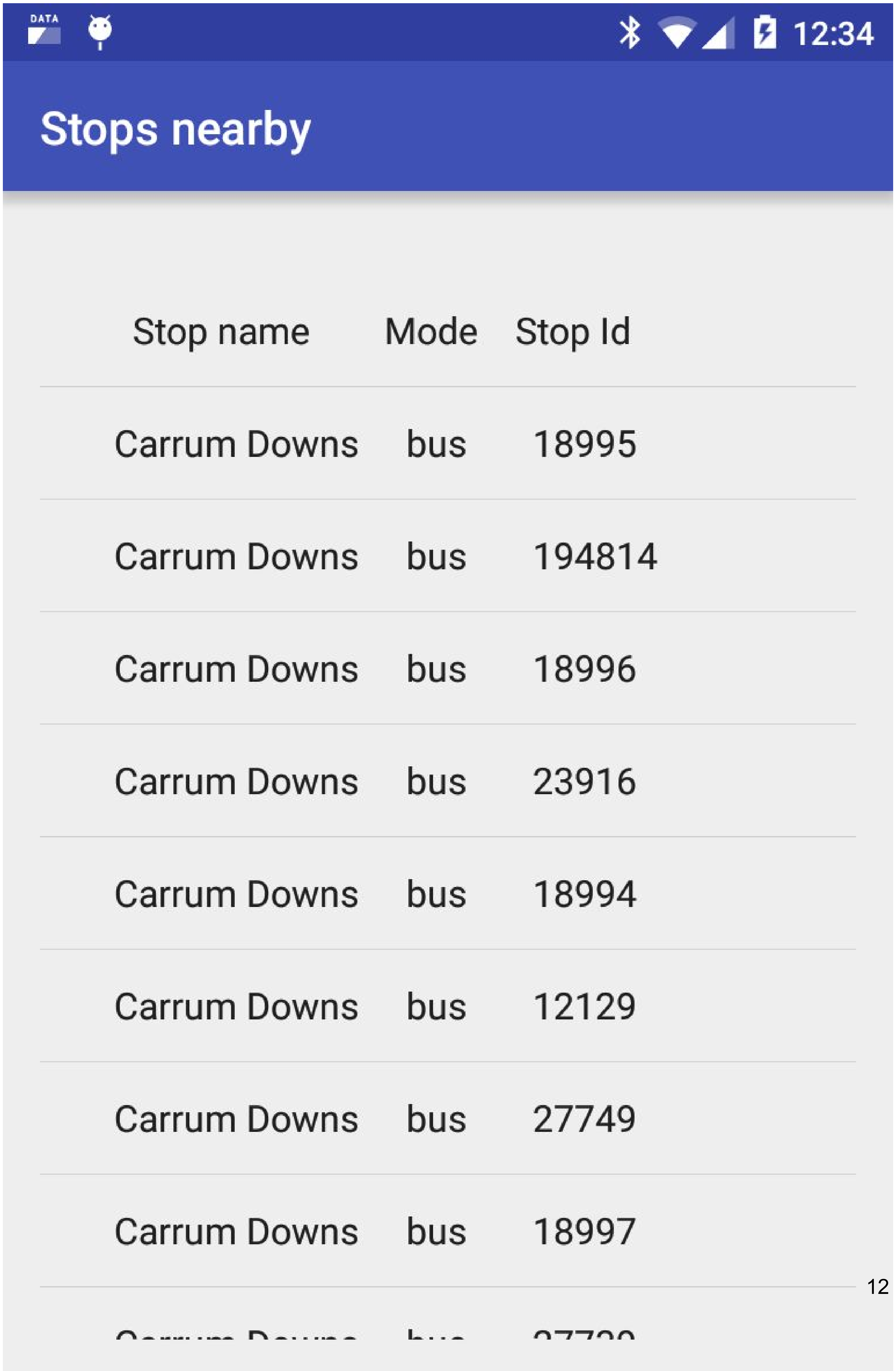
title: Cheltenham Station: Temporary car park closures from Saturday 9 July to Friday, 15 July 2016

description: Due to maintenance work on structures and facilities, electrical networks and signalling, there will be temporary car park closures at Cheltenham Station from Saturday 9 July until Friday, 15 July 2016.

title: Malvern Station: Temporary car park closures from Tuesday 12 July to Thursday, 21 July 2016

description: Due to maintenance work on structures and facilities, electrical networks and signalling, there will be temporary car park closures at Malvern Station from Tuesday 12 July until Thursday, 21 July 2016.

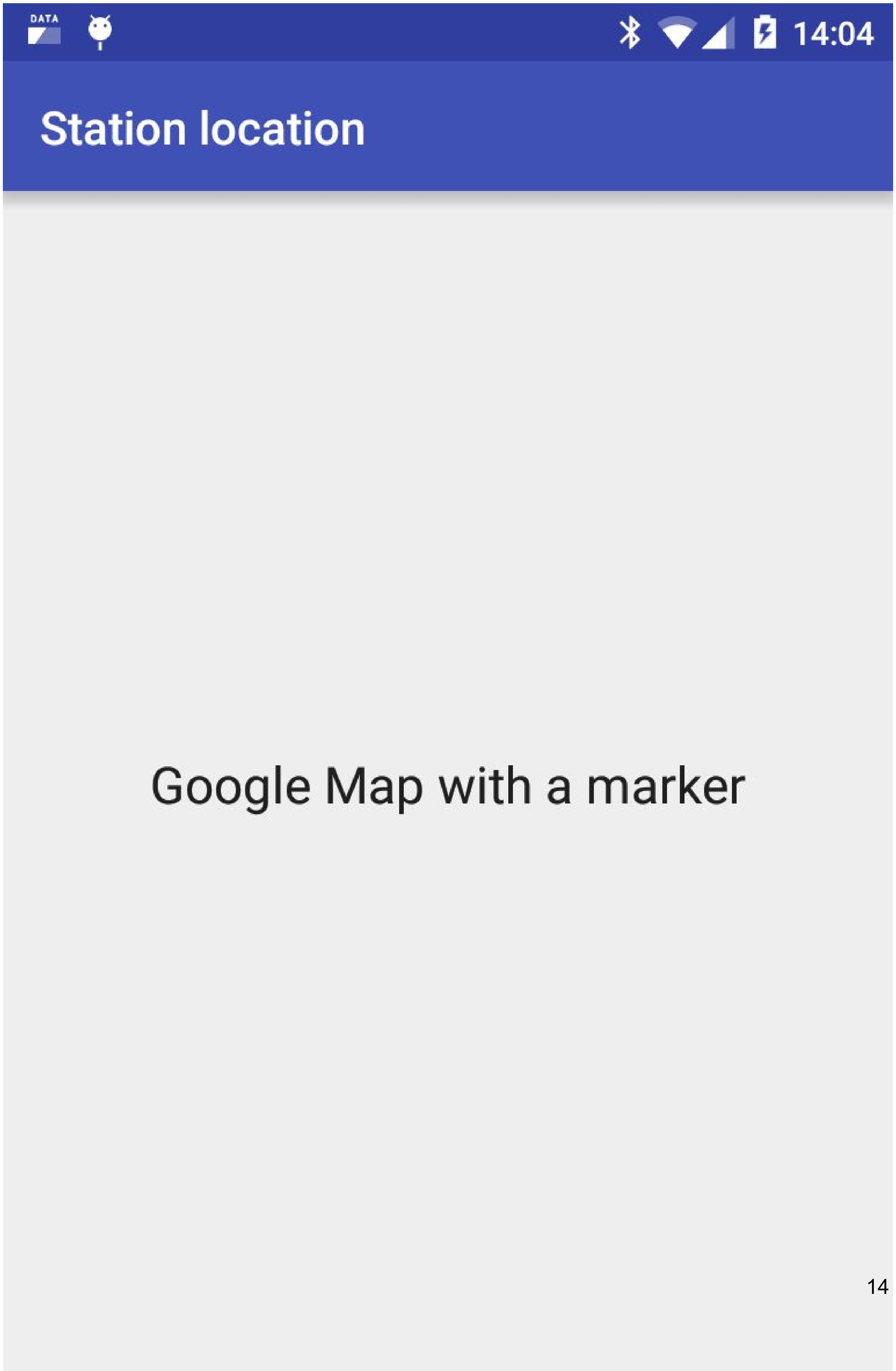
Shows information about disruptions on the network.

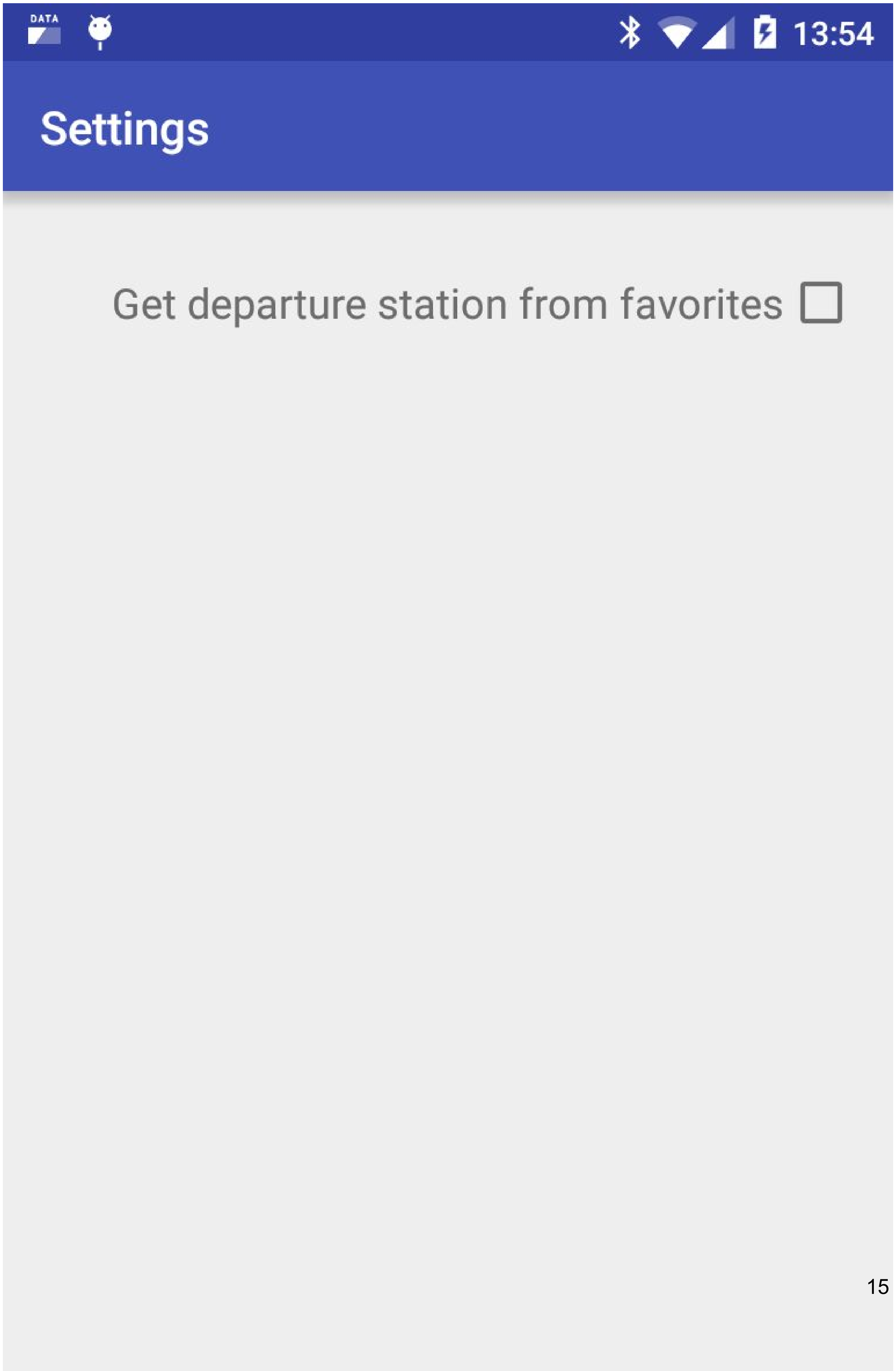


Stops nearby

Stop name	Mode	Stop Id
Carrum Downs	bus	18995
Carrum Downs	bus	194814
Carrum Downs	bus	18996
Carrum Downs	bus	23916
Carrum Downs	bus	18994
Carrum Downs	bus	12129
Carrum Downs	bus	27749
Carrum Downs	bus	18997

Shows up to 30 nearby stations/stops - based on the device’s location. Optionally allow to see the stop on the map and departure times.


14:04
Station location
Google Map with a marker



Get departure station from favorites ☐

Shows settings screen.

Key Considerations

How will your app handle data persistence?

The application will build and use SQLite database to store line’s, station’s details and favorite stations selected by the user (Content Provider)

Describe any corner cases in the UX.

The very first time the user starts the app it will see the Favorite Stations screen. It will allow to add favorite stations. Next time, the app will show the next five departure times for the most recent selected station.

At any time user will be able to see nearby stations/stops or network disruptions information.

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

- Joda-time to convert time from/to local Melbourne and UTC time - the PTV timetable is using UTC time
- Eventbus - to pass data retrieved from Internet (in IntentService) to current activity/fragment
- Design support library to use CoordinatorLayout, AppBarLayoput, e.t.c.

Next Steps: Required Tasks

Task 1: PTV API research

- Spent some time to understand the PTV API

Task 2: Test project Setup

- Build a simple test project that will test different requests available in the PTV API
- Configure project’s build.gradle, e.g. add libraries

Task 3: Melbourne Public Transport project Setup

- Build the new project
- Configure project’s build.gradle, e.g. add libraries

Task 4: Build local DB

- Write code for build local SQLite DB
- Create IntentService that will send requests and process PTV API responses
- Add code to get PTV line’s and station’s details from PTV and populate DB

Task 5: Favorite Stations

- Build UI and write code for Favorite Stations activity/fragment
- Build UI and write code for Add Favorite Station fragment

Task 6: Next Departures

- Build UI and write code for Next Departures fragment

Task 7: Stops Nearby

- Build UI and write code for Stops Nearby fragment

Task 8: Disruptions

- Build UI and write code for Disruptions fragment

Task 9: Stations On Map

- Build UI and write code for Station on Map fragment

Task 10: Settings

- Build UI and write code for Settings activity/fragment