

Becky Leslie (beleslie), Bryce Martz (bmartz), Daniel Houtsma (dhoutsma), David Li (lidav6), Dylan Whitlow (whitldy), Nick Bissiri (nbissiri)

See updated schedule here:

<https://github.com/WheresMyBus/android/wiki/System-Design-Specification#project-schedule>

### Time estimates to complete:

Writing API in Ruby	4 weeks
Writing Controllers	7 days
Writing backend modules	5 days
Unit Testing	5 days
System/Integration Testing	7 days
Frontend testing	5 days
Search a route/neighborhood by catalog	7 days
Search by map	3 days
Find and display bus locations	6 days
Favorite routes and neighborhoods	6 days
View alerts	5 days
Post alerts	5 days
Comment on an alert	2 days
Upvoting/downvoting	5 days

### Features and Cuts

We completed all the major functionalities of our app, but we cut the following features:

- Defining neighborhood perimeters and allowing users to search for neighborhood forums by a map
- Saving coordinates where an alert was posted and allowing users to search for “Alerts near me”
- Allowing users to see which direction a bus was traveling when they looked at bus locations

For the most part, we did not get to implement the above features because we added a feature to our SRS partway through the project: allow users to comment on alerts. On top of this, we underestimated the amount of time it would take to develop the frontend views. We had the least experience in this area as it was relatively unique to Android development.

We decided to cut defining a perimeter of neighborhoods because we could not find data to accurately define these perimeters. Cutting this feature saved roughly 4-5 days of work.

The other two features we cut because we were behind on the major functionality and labeled these as the most unnecessary features we were planning to implement. Together, cutting both these features saved roughly a week and a half.

### **Task Assignments:**

Majority of time spent in bold:

- Becky: **project manager & frontend**
- Bryce: **frontend**
- Daniel: **backend (API) & frontend**
- David: **modules & testing**
- Dylan: **controllers & testing**
- Nick: **testing & frontend**

Bryce was initially going to be part of the testing team, but with only two people working on the frontend, this took most of her time and she was taken off the testing team to focus on the frontend.

Towards the end of the project, more team members were recruited to help finish the frontend and testing. Nick and Daniel helped on the frontend side, and David helped on the testing side.

In general, we didn't spend enough time writing frontend code. None of us had experience in Android development, and even basic tasks took much longer than expected. There were a lot of poor design choices because of inexperience which had to be reworked. In the same vein, we spent too much time figuring out how to display things in the Android views.

There were many methods and variables we initially designed and later realized we did not need or had to cut. The comments for these, along with their implementation for some parts, were discarded. To avoid this, we could have spent less time designing, looking more closely at the APIs we needed to work with - Ruby and OneBusAway - for variables that required serialization and several ids that were Strings, along with unnecessary methods.

Another thing we spent too much time on is build scripts in Linux, as we were asked for it. This was because installing software manually was tedious, compared to Android Studio which installed it automatically.