

How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: **“Capstone_Stage1”**
3. Replace the text in green

Submission Instructions

1. After you’ve completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it **“Capstone Project”**
3. Add this document to your repo. Make sure it’s named **“Capstone_Stage1.pdf”**

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you’ll be using and share your reasoning for including them.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: Your GitHub username here

FiveTA

Description

Makes it easy for one or more people to share and control an ETA (Estimated Time of Arrival) for any kind of meeting.

Want to meet up with your friend in 20 minutes? Send up a ETA.
Something came up and you are going to be 5 minutes late!?

It's a really minor inconvenience and you don't want all the hassle of texting or calling your friend to tell him you are going to be 5 minutes late?

No problem!, just add 5 minutes to the ETA and your friend will know that you are running late.

Want to schedule an informal meetup with one or more friends?

Tony's place at 7:00 pm? Make a group ETA and see if everyone is going to be on time!

Problem:

You know how people are almost never on time? Well, most of the time that can't be changed, people are always gonna be late, there is so many factors that can affect the ultimate goal of being on time.

Meet FiveTA, with this app you will be able to create a shared ETA that both parties will be able to see and control your version of the ETA, this means that if you are running 5 or more minutes late, you just add them to your ETA and your friend will know.

Intended User

People who have friends that are always late, and everyone in general.

Features

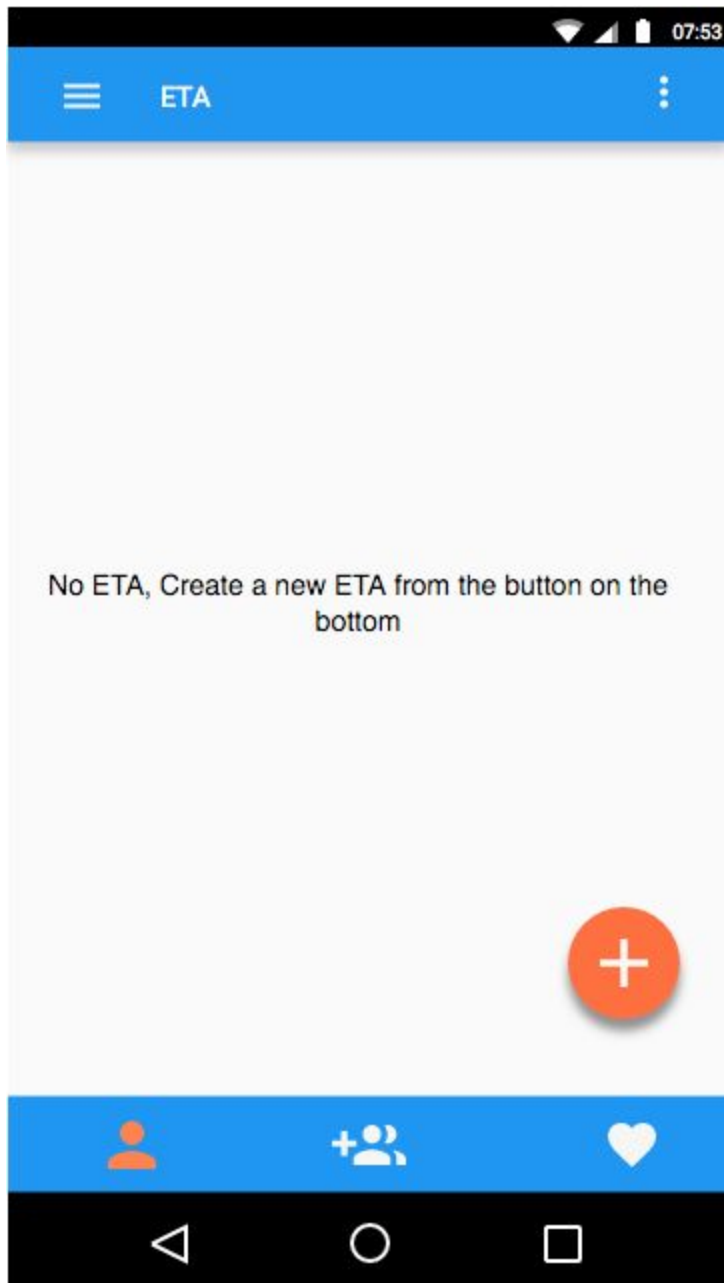
List the main features of your app. For example:

- Saves information about your favorite ETAs
- Shares ETA with your friends
- Connects to Facebook to see your friends

User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Main screen when empty



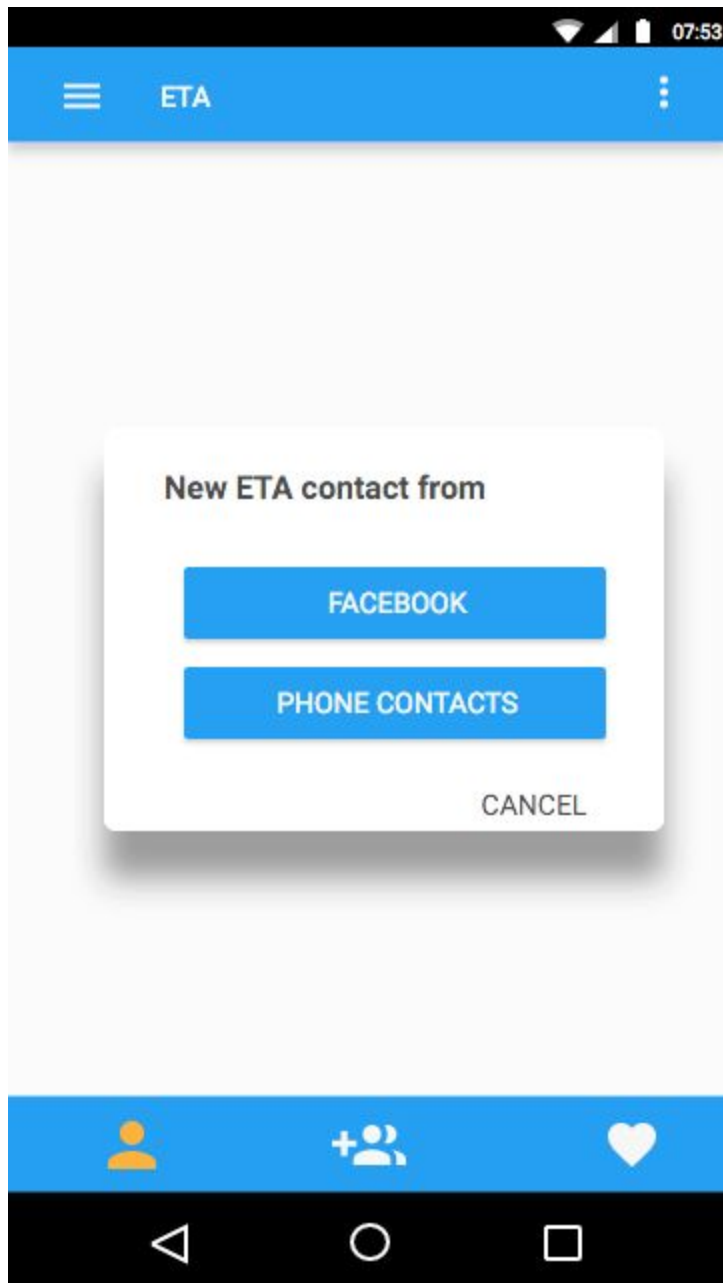
The app will ask you to create a new ETA with the floating action button, as you can see it has 3 tabs.

First tab is Personal or 1on1 ETAs.

Second tab is Group ETAs.

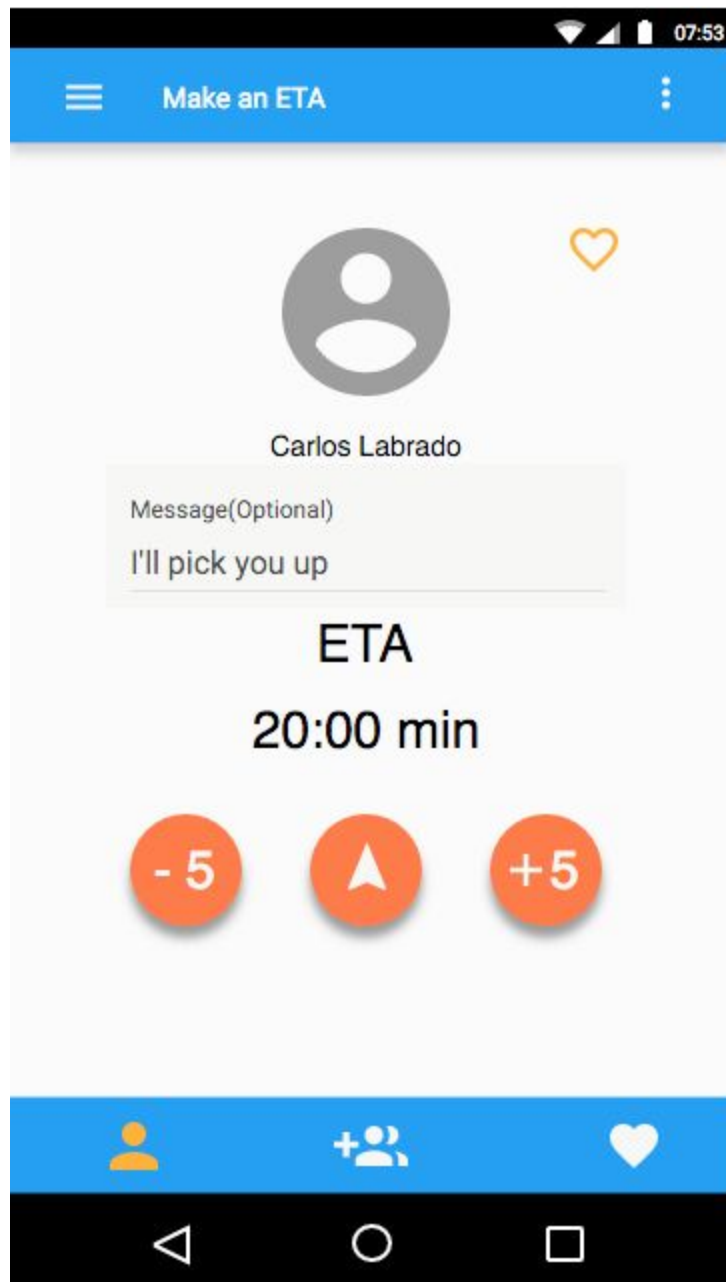
Third is favorite ETAs, personal and group mixed.

Contact selector dialog



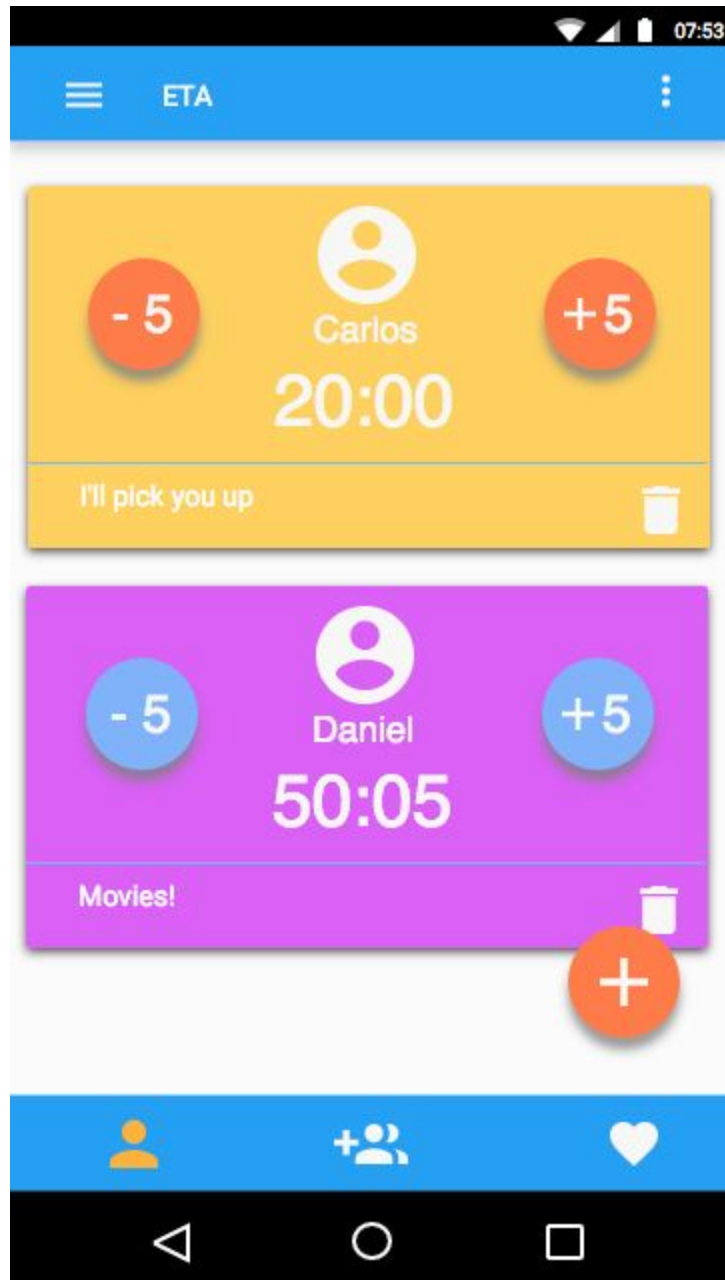
The follow up screen shows a dialog from where you want to look up for your contact.

Create a Personal ETA



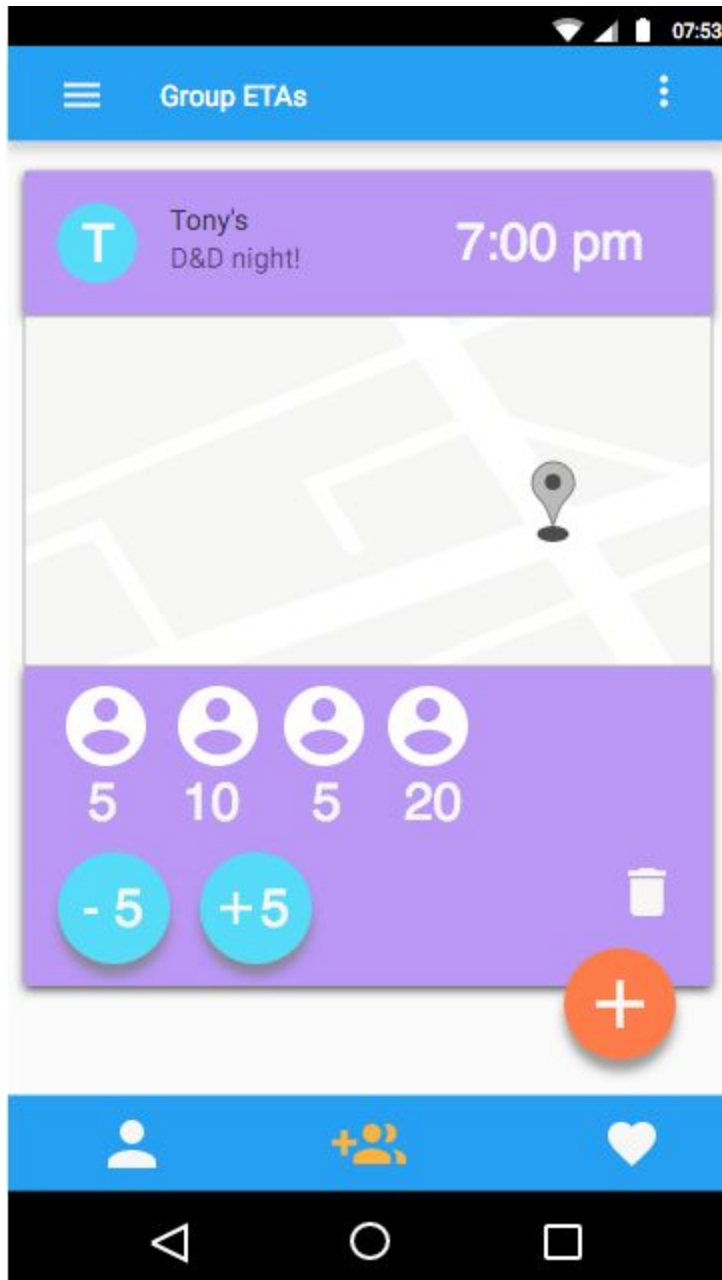
Shows the contact you are going to make an ETA with, a 20 min default timer and buttons to decrease 5 min, start the ETA or add 5 minutes. And the possibility to add an optional message to the ETA. You can also click on the “Heart” and save it to favorites.

Personal ETAs



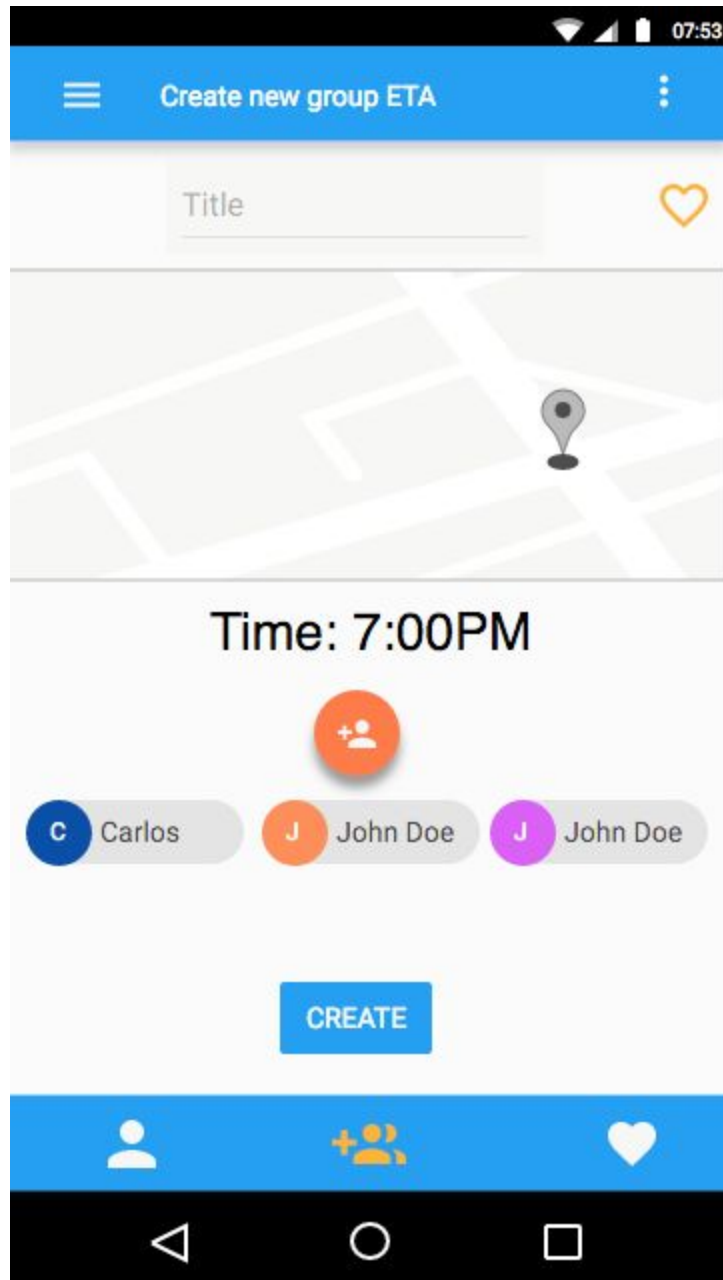
Personal ETA main screen with a few ETAs created and simple controls for decreasing or increasing the timer on each one. Also the "Delete" icon to erase that ETA.

Group ETA screen



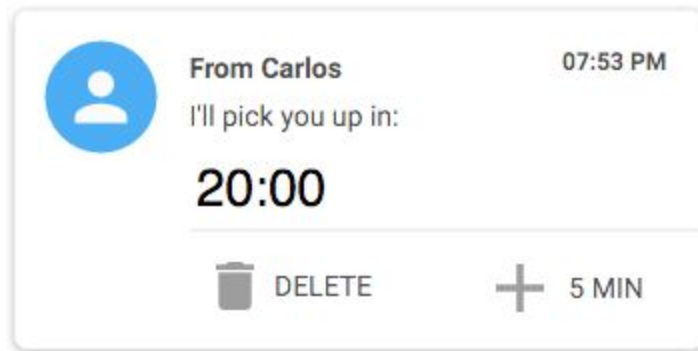
You can see a group ETA, the main difference with personal ETA (apart from being for more than one person) is that the ETA isn't a countdown is more a fixed hour and the numbers below each face is the extra time that each person needs.

Create a group ETA Screen



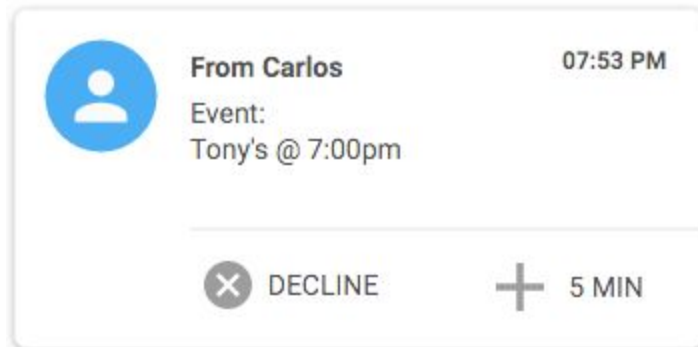
In here you can see the information needed to create an ETA, clicking on the map will prompt you with a map to place a marker, clicking on the time will popup a time picker and clicking on the fab icon will prompt you with the contact selector. If you click the heart it will be saved to favorites.

Personal ETA Notification



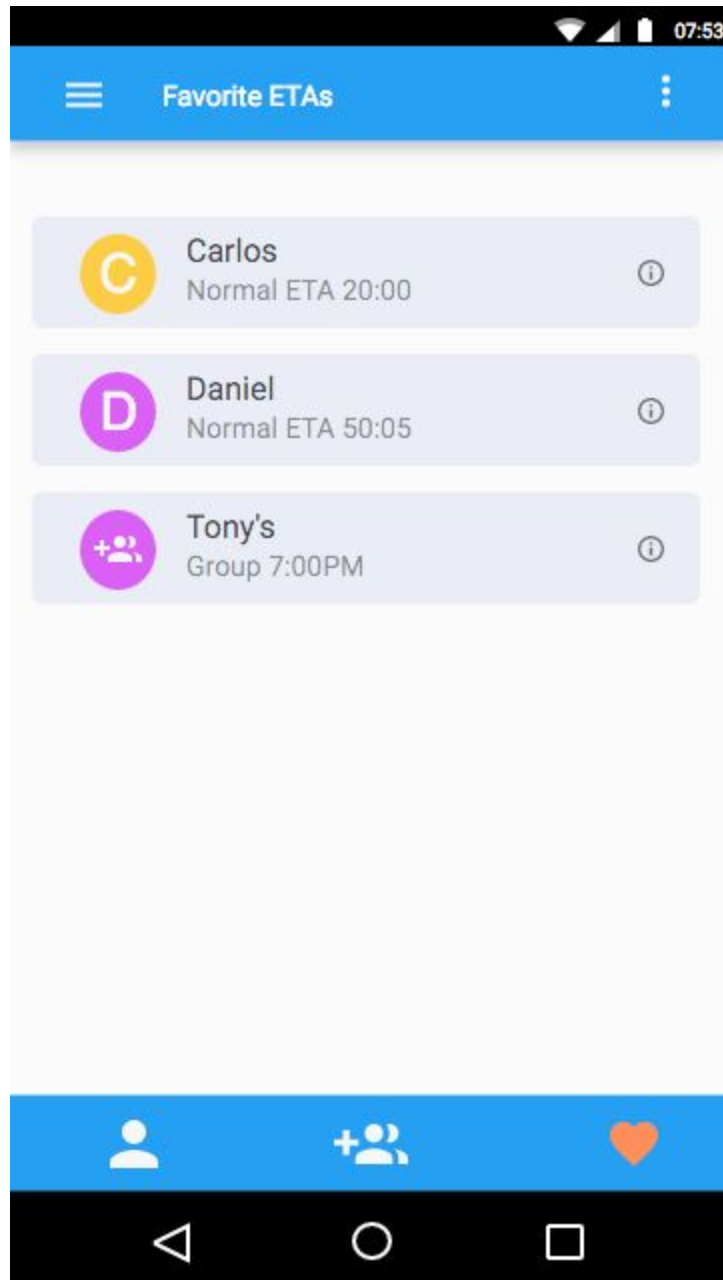
When a personal ETA is created this notification will appear on each person Android device. Both will be able to increase the time.

Group ETA Notification.



When a group ETA is created this notification will appear on each person Android device. You can add 5 minutes to your own time.

Favorites screen



You can see your saved ETAs in here, personal and group are mixed. When you click on one it will take you to the detail and you will be able to start it again from there.

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

For the backend we will use the Google Cloud Storage and the Datastore.
We will use the content provider from the Phone app to get the contacts.
We will create our own content provider to save the “favorites” ETAs.

Describe any corner cases in the UX.

The navigation will be done with the bottom bar, if the user is on the app, goes to another tab and presses the back button, it won't take him to the last tab, it will get him out of the app.

Not included in the screenshots is the drawer, which will have only a settings option.

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

Glide to handle the loading and caching of images.

Firebase for login.

Enhanced event bus to make easier the communication between activities and fragments

ButterKnife to manage the injection of views

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

- Configure libraries
 - Firebase, Facebook
- Get the keys from Google Cloud Services
- Create the backend project

Task 2: Implement UI for Each Activity and Fragment

- Create the splash screen
 - A 500ms screen to show the colors and logo
- Create the login Activity
- Build UI for MainActivity
 - First set up the drawer

- Set up the toolbars
- Build the Tabbed fragment
 - Set up that fragment with 3 items
- Build the Personal ETA fragment
 - Create the legend for when no ETAs are found.
 - Create the fab button
 - Create the Contact chooser dialog
 - Create the recyclerView
 - Create the adapter
 - Create the Item layout
- Build the Personal ETA Detail Fragment
- Build the Group ETA Fragment
 - Create the recyclerView
 - Create the adapter
 - Create the item layout
- Build the Group ETA Detail Fragment
- Build the Favorite ETA fragment
 - Create the recyclerView
 - Create the adapter
 - Create the item layout

Task 3: Implement the Login

- Use the firebase login functionality

Task 4: Implement Google Map

- Create the Google map picker Activity to be launched from the Group ETA Detail Fragment
- Implement the Snapshot method for the map so we only get a preview in the detail.

Task 5: Create Single and group ETA notifications

- Create layout
- Implement the logic on a service

Task 6: Implement the facebook functionality for getting the contact info

- Create the needed code on the MainActivity

Task 7: Create the backend

- Create all the references to firebase
- Create the model of classes that will hold the information

Task 7: Create the content provider for saving the favorites

- Create the content provider

Task 8: Handle the screen changes

- On screen change make sure the data is not lost

Task 9: Create a Widget of the single ETA

- Create the layout

Task 10: Accessibility check

- Include content description for all images and buttons.

Add as many tasks as you need to complete your app.

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"