

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.](#)

[Describe how you will implement Google Play Services.](#)

[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: Superhuman07

UnSpammer

Description

An awesome app that enables the user to make calls and turn down the calls from spammers in one click and add those who spam you whether it is the telemarketers, advertisers or any other person. UnSpam your phone in just a click!

Intended User

UnSpammer can be used by anyone who likes to 'UnSpam' their phone that is giving the phone the power to automatically disable or turn down the calls from the persons who constantly bugs them.

Features

List the main features of your app. For example:

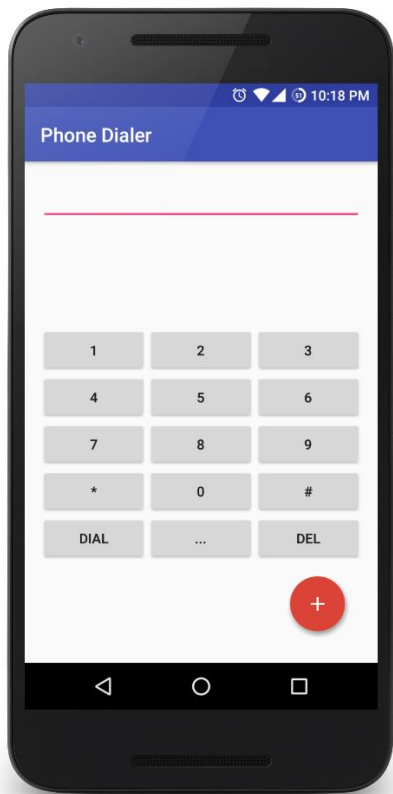
- Can make calls directly through the app\
- Turn down automatically spammers who are blacklisted by the user
- Enable or disable the 'UnSpammer' by just clicking on the switch
- Add or remove the spammers from the blacklist
- Switch the 'UnSpammer' enable or disable using widget

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.

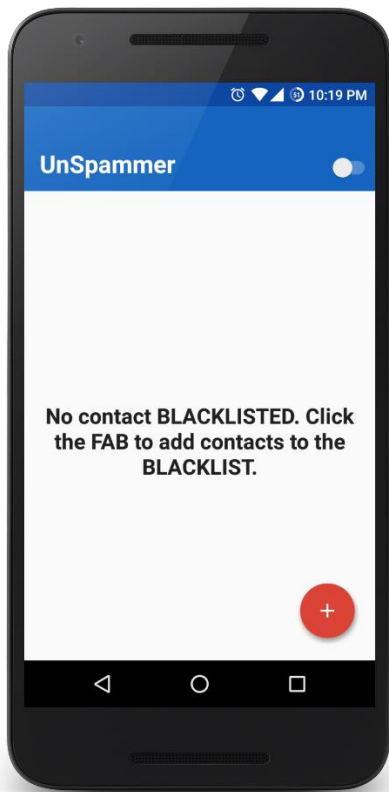
Note – They might be not same as the finished product.

Screen 1



Replace the above image with your own mock [click on the above image, then navigate to
Insert → Image...]
Provide descriptive text for each screen

Screen 2



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

Key Considerations

How will your app handle data persistence?

One content provider for the list of numbers for getting the calls and using them with the loaders.

Second one is for the numbers to be add the users' android account contacts to be added to the database from the contacts app (user must have some contacts).

Describe any corner cases in the UX.

The user should get tutorials well ahead on how to use the app and get persistent change while dialling the number or blacklisting the number, it will be immediately shown by the toast or confirmation dialogs to decrease the app complexity.

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

- Some tutorial library for the user to get to know the app such as showcaseview.
- Some library to pick up all the user contacts.
- Some library for FABs
- Android Support Recycler View

Describe how you will implement Google Play Services.

- Google Analytics – For getting the users' info and which part of the app is crashing.
- Google Admob – For earning money from the ads (a banner ad).

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

Implement the call dialer for the app

- Make dialer UI
- Enable the calling feature that is adding state listener inside the app so that user can directly call and end the blacklist calls automatically.

Task 2: Implement UI for Each Activity and Fragment

Implement material design UI for the dialer and the other activity which will contain the numbers that are getting turn down automatically.

Creating the cardview for the numbers that are going to be turn down.

Getting the number call service from android telephony manger service.

Task 3: Implement the Recycle View to show the blacklists

Using the library to pick contacts use it to add the numbers inside the cardview by using the recycle view and Loaders to show the contacts.

Using FAB library to add or delete the blacklist numbers.

Task 4: Implement widget to enable or disable the 'UnSpammer'

Adding a widget that will enable or disable the app.

Task 5: Implementing the google play services

Adding the google analytics for the information about the users and to get the reports about the crashes.

Adding the banner ad to the app using google admob.

Task 6: Implement the tutorial on how to use the app

Implementing the library which show the tutorial on how to use the app.

Final finishing of UI.

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**"