

How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [Select All → Copy → Paste into new document]
 2. Name your document file: “**Capstone_Stage1**”
 3. Replace the text in green
-



[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: aaroncrutchfield

Natasha's Hair Studio

Description

Natasha's Hair Studio provides a mobile friendly experience for current and prospective clients to interact with Natasha's business. The app brings together the different aspects of her business and allows users to interact with them in one place on their mobile device.

Natasha's goal is to provide a level of service that exceeds her clients' expectations. Her mobile app is a platform to help further increase her level of service to them.

Intended User

Current and prospective clients.

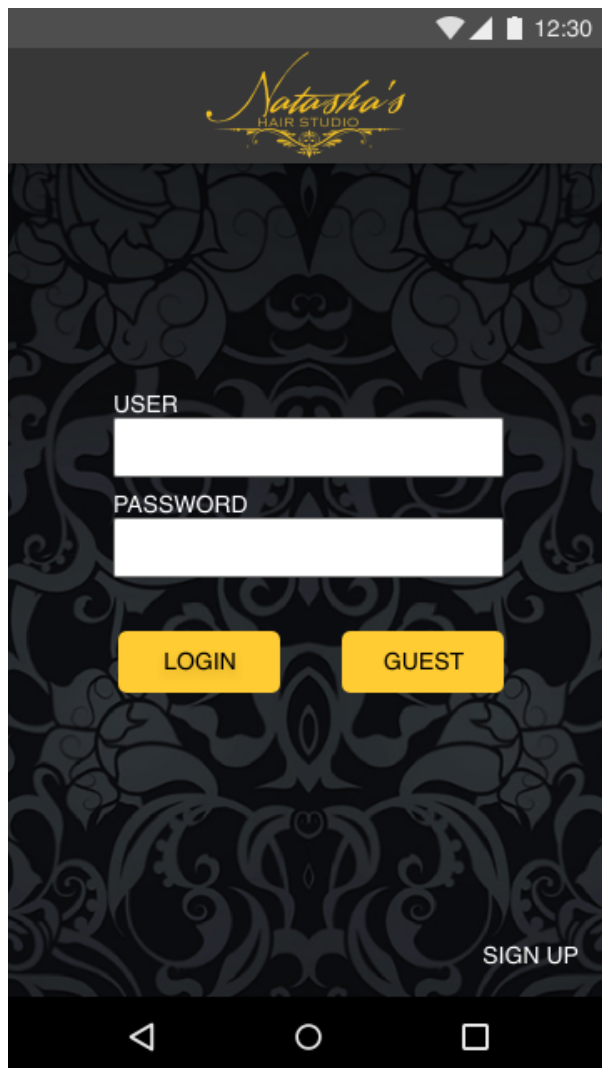
Features

- App is written solely in Java
- View online catalog of hair extensions
- Book appointments for salon service
- Write and view reviews of services
- Follow Natasha's Hair Studio on social media

User Interface Mocks

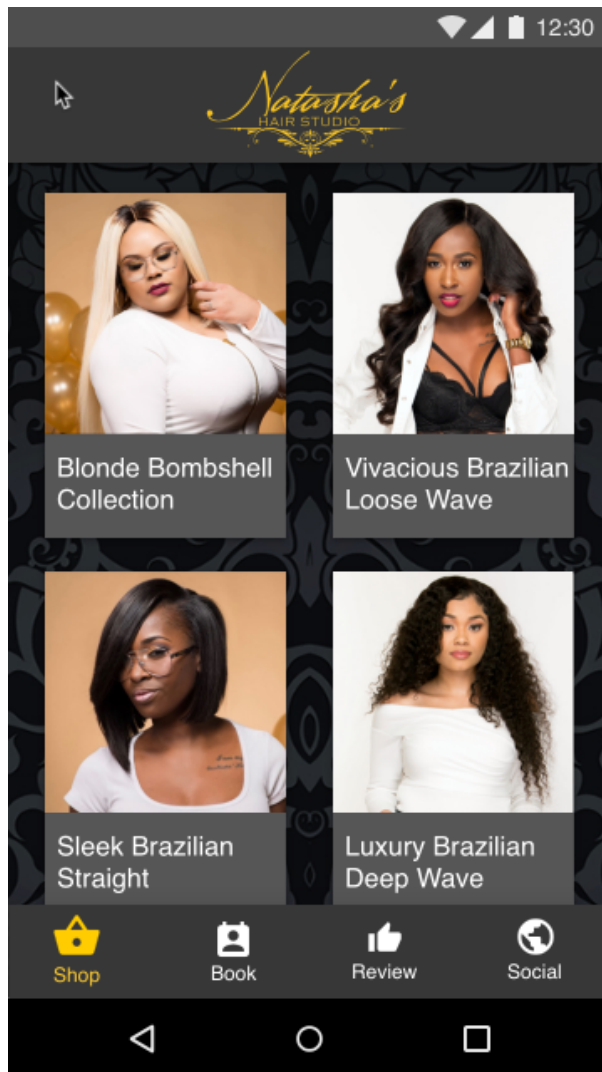
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1



The login screen allows clients to create a personal account. This is mainly to allow them to write reviews. If they wish to continue as a guest to use the other features, they may.

Screen 2



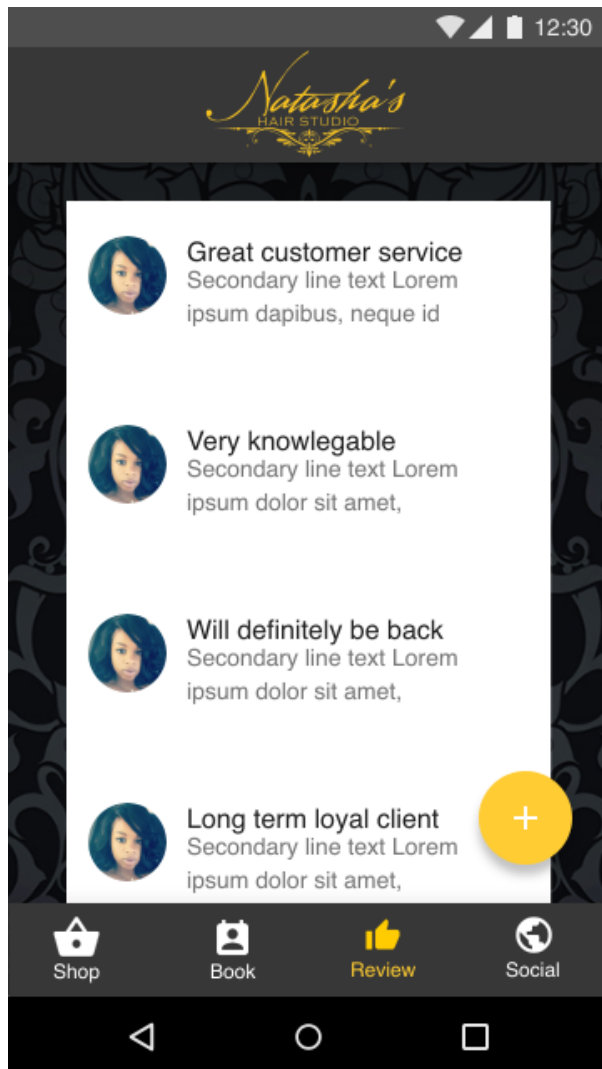
The Shop page will be for displaying the online catalog of hair extensions available for purchase.

Screen 3



The Booking page will allow clients to book appointments using a web-view of Natasha's Hair Studio's Acuity Scheduling site

Screen 4



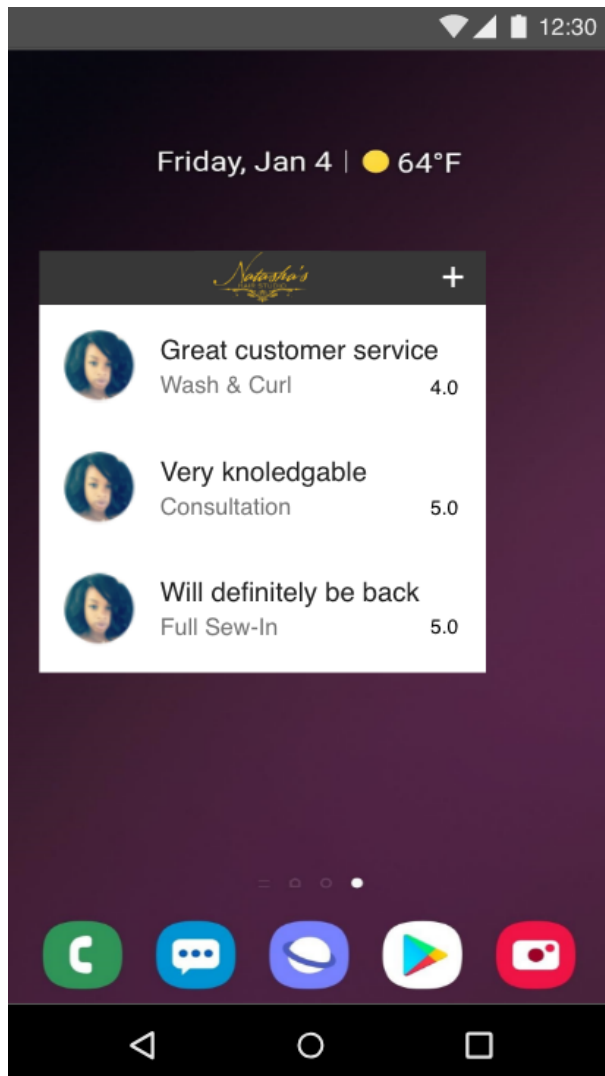
The Review page is for writing and viewing reviews of services from the salon.

Screen 5



The Social page provides links to follow the salon's social media page. Clients often post pictures of themselves with the new hair style that they received.

Screen 6



The App Widget also allows writing and viewing reviews of services from the salon.

Key Considerations

How will your app handle data persistence?

Firebase Firestore will be used to persist the data for the store information as well as the reviews that are posted.

Describe any edge or corner cases in the UX.

Guest users will not be able to write reviews

Users can only delete/modify their own reviews

Without a network connection, users can't book appointments or view her social media.

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

Android Studio

- 3.3.1

Gradle

- 4.10.1

[Glide](#) – Loading images

- 4.8.0

[Firebase Firestore](#) – Persisting data for the users and their reviews

- 16.1.0

[Firebase Authentication](#) – Convenient login using Google account

- 16.1.0

[Firebase Storage](#) – Storing pictures for in-app use

- 16.0.5

[Firebase UI](#) – Convenient to populate data into UI

- Authentication – 4.1.0
- Firestore – 4.3.1
- Storage – 4.3.1

Describe how you will implement Google Play Services or other external services.

Google Sign-in will be utilized to authenticate users in Firebase Authentication

Technical Requirements

- RTL layout switching enabled on all layouts
- All strings kept in strings.xml file
- Use IntentService to update App Widget

Next Steps: Required Tasks

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

Task 1: Project Setup

Add libraries and configure external API to communicate with the app.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Activities
 - LoginActivity
 - MainActivity
 - ProductCollectionActivity
 - ProductDetailsActivity

- Build UI for Fragments
 - ShopFragment
 - BookAppointmentFragment
 - ReviewFragment
 - Social Fragment
- Setup BottomNavigationView

Task 3: Add Firebase Authentication

- Use Firebase assistant to add Firebase Authentication to project
- Enable Google sign in method in the Firebase Console
- Add logic for signing in
- Handle potential errors from signing in
- Add logic for signing out

Task 4: Add Firebase Firestore

- Use Firebase assistant to add Firebase Firestore to project
- Enable test mode for data access
- Initialize an instance of Firestore
- Add logic for adding reviews
- Add logic for editing reviews
- Add logic for deleting reviews
- Disable test mode and secure the data

Task 5: Wire up Store functionality

- Create item layouts for ProductCollections and Products
- Handle clicks on ProductCollections to view Products
- Handle spinner selection of product options to show the co-related price
- Handle pressing the buy button to have PayPal handle to purchase
 - Button disabled if mandatory product options not selected

Task 6: Wire up Booking webview

- Add a WebView to the layout file
- Set the Acuity Scheduling URL for the view to load
- Handle page navigation by setting a WebViewClient

Task 7: Wire up Reviews functionality

- Create item layout file for reviews
- Handle swipes on items for editing or deleting reviews
 - Handle case if user doesn't own that review
- FAB click will prompt user to add a review
 - Handle case if user isn't logged in

Task 8: Wire up Social functionality

- Launch Facebook page when Facebook imageView is selected
 - Show in web browser if Facebook app isn't installed
- Launch Instagram page when Instagram imageView is selected
 - Show in web browser if Instagram app isn't installed

Task 9: Wire up Widget functionality

- Create the widget layout
- Get the reviews from Firestore
- Populate data into the widget layout
 - Using
- Update the widget each time there is a new review using IntentService

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

Submission Instructions

- After you've completed all the sections, download this document as a PDF [File → Download as PDF]
 - Make sure the PDF is named "**Capstone_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"