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**”

GitHub Username: Your GitHub username here

Crafty

Description

Crafty consumers with art and craft and otherwise buy the art and craft material online. It is like ecommerce app for art and craft. Get timely delivery of the items.

Intended User

This is for all the consumers who have artist in them.

Features

List the main features of your app. For example:

- Allow users to register
- Allow users to login
- Allow users to view the items on the portal
- Allow users to buy item on the portal

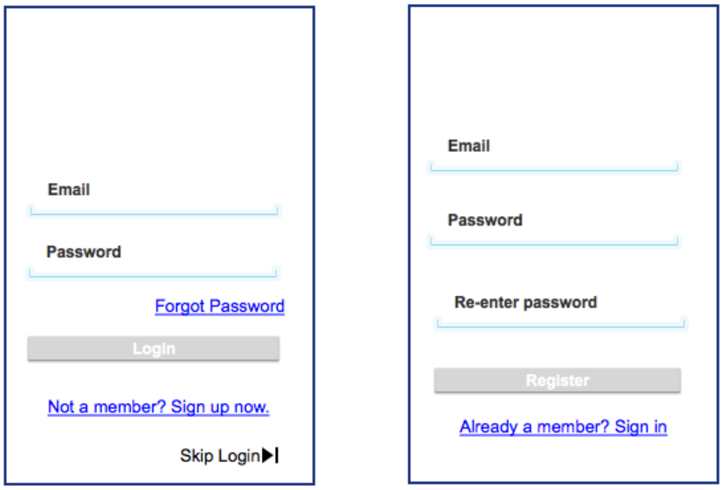
Stretched goal features:

- Allow users enquire about the product
- Allow users to give feedback about the product
- Allow users to watch video related to the art and craft techniques like quilling, bottle cutting etc.

User Interface Mocks

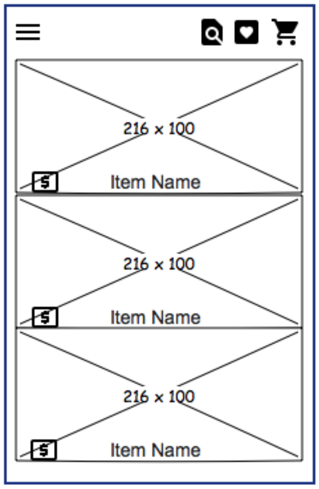
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1 - User Login



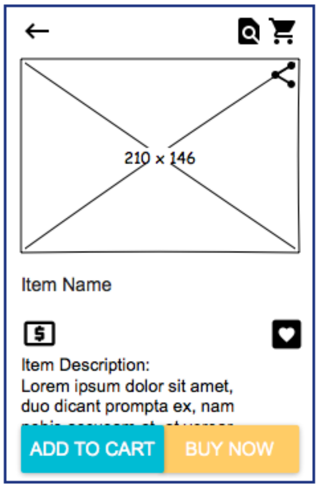
In the above screen user is allowed to enter the site either as logged in user or skip the login. If he want to login and is a new user, he can register also.

Screen 2 - Catalog



This is the page where all the items related to craft will be visible. Now when user clicks any of the item, he will see the item detail page.

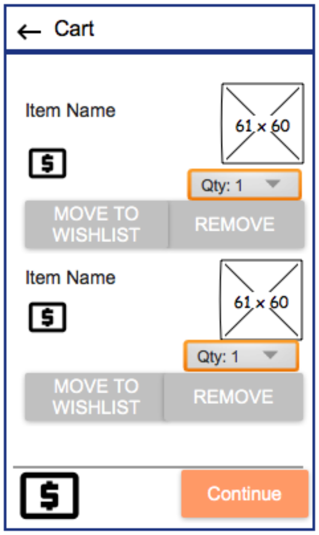
Screen 3 - Item Detail



Item Description:
Lorem ipsum dolor sit amet,
duo dicant prompta ex, nam

This is the page where user can view the item detail and price. He can either add it to cart or buy now to add to cart and go to cart screen.

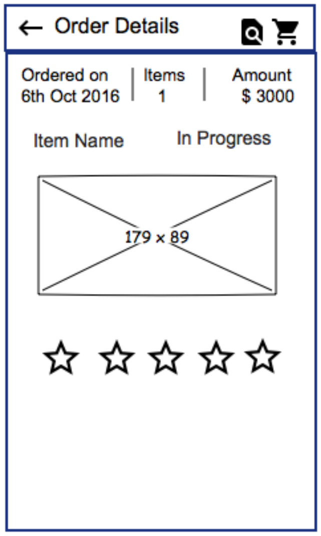
Screen 4 - Cart Screen



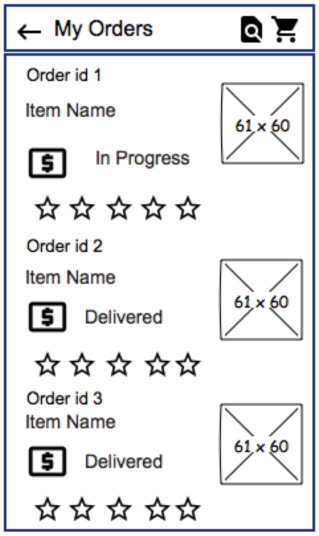
This is the cart, where user can see item name, image, item price and also can change the quantity. He can scroll the page to see all the cart items. The continue and $ sign will be kind over the top of the item name, when user scrolls, will not hide with the scroll.

When the order is complete, the user is moved to order screen.

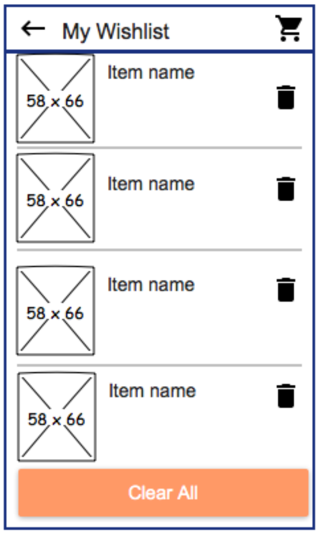
Screen 5 - Order details



Screen 6 - My Orders

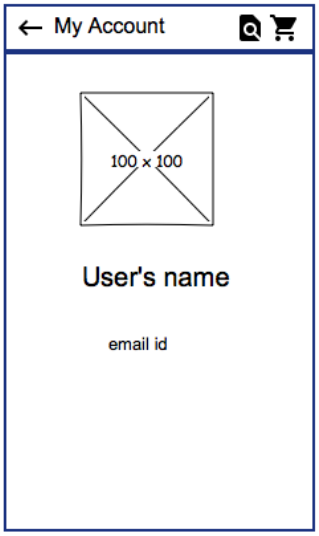


Screen 7 - My Wishlist



User can view his wishlist and clear them also. All the items are clickable and will take user to detail screen.

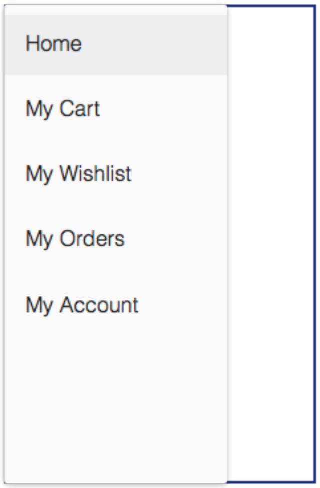
Screen 8 - My Account



User can view his account detail with which he logged in.

Screen 9 - My Account

This is misnomer as it is not really a screen. But the menu will look like this.



Key Considerations

How will your app handle data persistence?

I will connect to existing content provider or sqlite for storing data. Most of the other things will be taken care off by the api itself. For viewing the catalog, I will use on demand request, and update the UI, using intent service. But, as a stretched goal, to make app more efficient, I will switch to SyncAdapter. For the search operation, I will use get the data from API only once, or on a per request basis using an IntentService to do so.

Describe any corner cases in the UX.

As of now, there shouldn't be any corner cases. I have to take care of back button, when people try to migrate between various activities.

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

I will be using Picasso or Glide to handle the loading and caching of images. Also, I will use okhttp or volley for http requests. Also, I may use retrofit for rest calls.

Describe how you will implement Google Play Services.

I will try to use google location api and google+ services to share the information to the friends.

Using the Widgets.

I will create the widget for the showing the wish-list to the users.

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

I have to setup the projects, including the libraries.

- Create a new project and update the gradle build file to include okhttp, volley, retrofit, picasso and butterknife

Task 2: Implement UI for Each Activity and Fragment

Here are my UI activities:

- Build UI for Login Activity
- Build UI for register Activity
- Build UI for viewing the catalog
- Build UI for showing the item detail
- Build UI for viewing the cart
- Build UI for placing the order
- Build UI for viewing all the orders

- Build UI for viewing all the wishlists
- Buld UI for viewing the user account

Task 3: Build the Backend

Here are my tasks:

- Create the rest based web-service for registering the user
- Create the rest based web-service for authorizing the user
- Create the rest based web-service for get the catalog
- Create the rest based web-service for searching the item in the catalog
- Create the rest based web-service for CRUD operations on order
- Create the rest based web-service for CRUD operations on wishlist
- Create the rest based web-service for editing the user information for logged in user

Task 4: Setup the DB

We have to build the models for persisting the information from the web services. Here are the tables I may need:

- Userlogin
- Users
- Items
- Orders
- Wishes

Task 5: Fixing the build variants

Here are my tasks:

- Create build variant for debug and release version
- Code cleanup
- Trying to meet the stretched goals

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**”