
Description
Intended User
Features
User Interface Mocks
  Screen 1
  Screen 2
  Screen 3
  Screen 4
Key Considerations
    Architecture
    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: Create Local Data Base and Content Provider
  Task 4: Load Data to UI and Save Favorite item to database
  Task 5: Add functionality to share Design Pattern UML diagram and Select as Favorite
  Task 6: Create Widget to display Favorite list
  Task 7: Create Google Service to get city name using Location Services

GitHub Username: venugopalraog

Java Design Pattern

Description

This application is intended to users who want to learn design patterns in Java. Helps to understand the Gof design pattern with detailed UML diagram. Allows user to make a design pattern as favorite and share by using gmail.

Provides Desktop Widget to display the list of Design Patterns marked as Favorite.

Also lists out the interview questions so that it helps to understand design patterns in detail.

Intended User

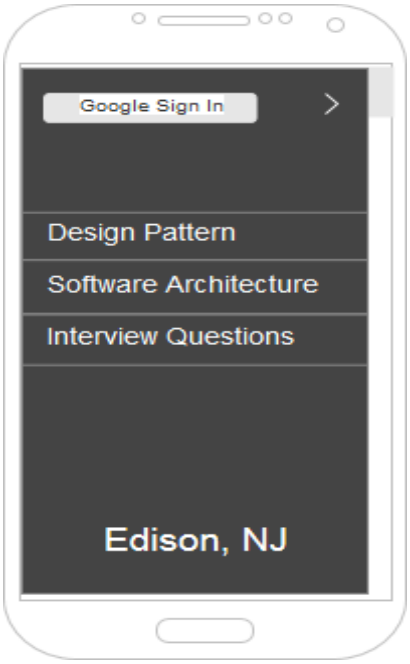
This app is intended for developers interested in learning Design pattern in Java.

Features

- Displays list of Design Pattern based on Category.
- Displays Detailed View for Selected Pattern.
- Allows Dismissing the Design Pattern as read.
- Allows marking Design pattern as favorite.
- Easy to navigate using Navigation Menu
- Screens are implemented in Material Design

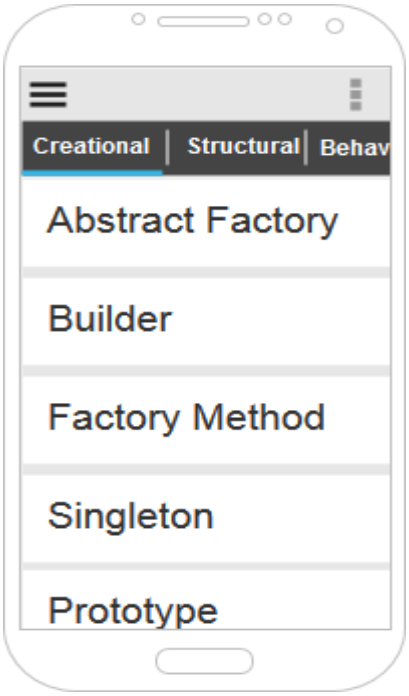
User Interface Mocks

Screen 1



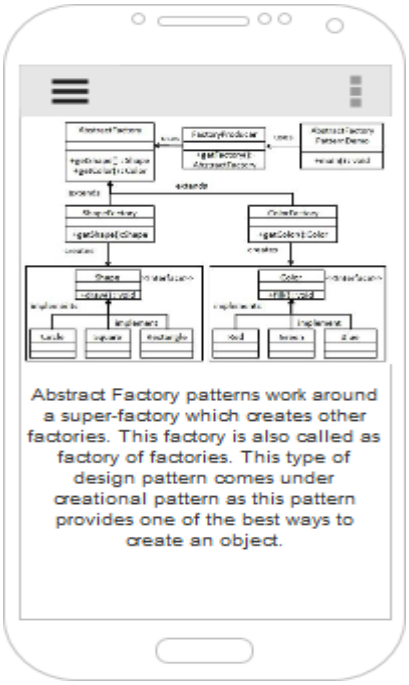
Navigation Menu displays the Design Pattern, Software Architecture and Interview Questions options. User can select any option to open Screen 2 with list of info available.

Screen 2



This screen displays Design Pattern Category and displays the list of Patterns under it. Allows user to dismiss the pattern by swiping the item.

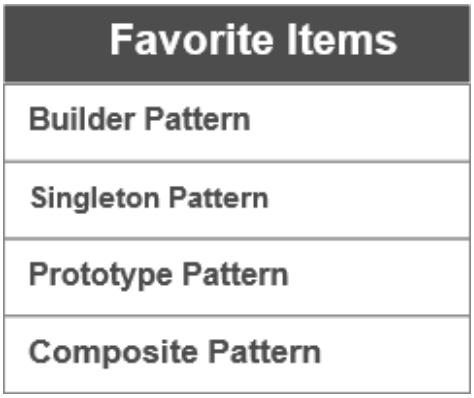
Screen 3



Detailed Screen displays more description about design pattern like flow chart diagram, description and allows user to share flow chart diagram on face book, allows emailing base

implementation of design pattern using Gmail. Share button and favorite icon will be added on top of flow chart image.

Screen 4

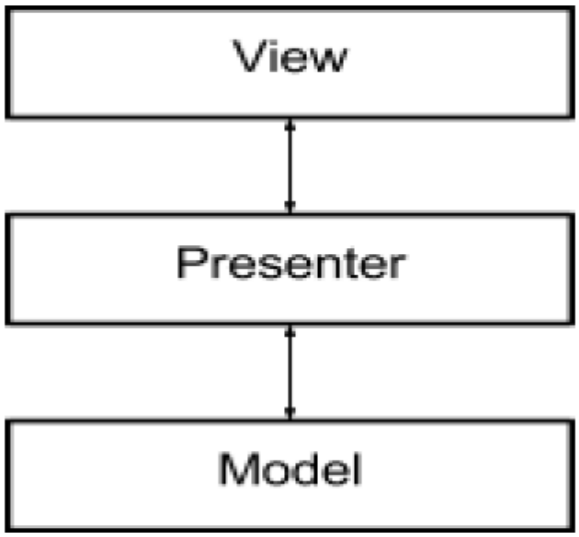


Desktop widget displays the items marked as Favorite. If user clicks on item it will launch the Detail screen.

Key Considerations

Architecture

This application will be based on the Model View Presenter pattern. This style of architecture will make the application easy to maintain and test.



View will handle the display of the data. All the user interaction will be deliver to the Presenter.

Model will provide and store data.

Presenter will coordinate the interaction between the View and the Model.

How will your app handle data persistence?

Data is stored in database and uses Content Provider to retrieve it.

Describe any corner cases in the UX.

When no data is available in database should display error screen.

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

- Butterknife: to bind view with resource id directly
- Dagger 2: For dependency injection. It will make easy to interact between the presenter and the view.
- EventBus 3: To interact between presenter and view. I will use events to notify the view about data changes.

Describe how you will implement Google Play Services.

To login into Google Account, Get Current city Address using Google Location Service and Google Geocoding API.

Next Steps: Required Tasks

Task 1: Project Setup

Design and create database to store required design pattern details and add some data to it.

Task 2: Implement UI for Each Activity and Fragment

Create Android Application design UI for Navigation Menu, Main Activity and Fragments. Add support for bigger screens like Tablet.

Task 3: Create Local Data Base and Content Provider

Design database to store the details required to display in UI. Create Content Provider to access the data.

Task 4: Load Data to UI and Save Favorite item to database

- Implemented Loader in Main Activity to load data from database to UI.
- Create AsyncTask to save the Design Pattern marked as favorite by User to Database.

Task 5: Add functionality to share Design Pattern UML diagram and Select as Favorite

Implement share the design pattern UML and save the favorite design pattern in Database.

Task 6: Create Widget to display Favorite list

Implement Widget to display list of Favorite Design Pattern.

Task 7: Create Google Service to get city name using Location Services

Implement Google play services to retrieve the city name and display it in Navigation Menu.