GitHub Username: shivasurya

ModelExam App For Android

Description

Recently worked with Online Exam Conducting company for Backend Development, So they requested me to build Android app for Online Mock Test for Banking community peoples. The concept is there are several banking exams in India, Modelexam.in provides training for them via Offline and online with mock test with payment options. So, Now my Challenge is to build complete mobile app for attending the mock test via mobile app with payments integrated.

Intended User

This app is intended for Banking community peoples for learning and practicing by taking mock test via our app.
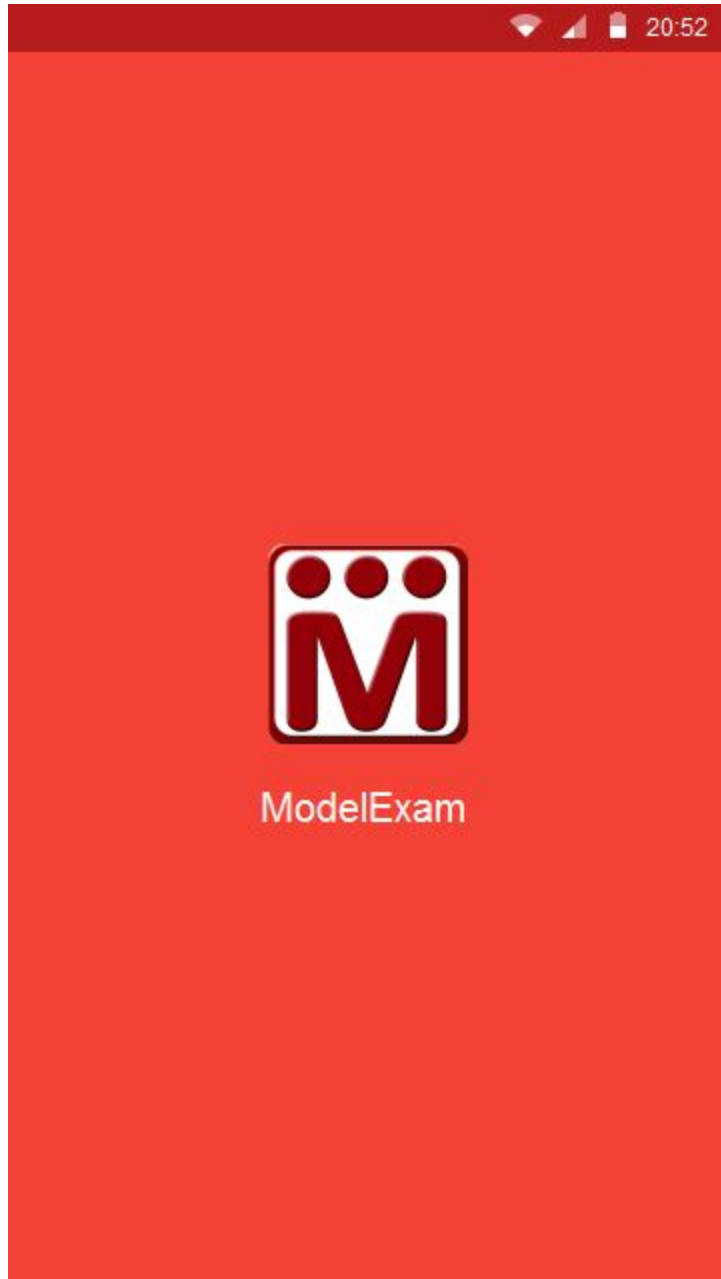
Features

List the main features of your app. For example:
- Take Timed Test
- Payment options within App (Instamojo/ carrier billing)
- Offline Demo exams
- Reviewing Test answers

User Interface Mocks

Screen 1 : SplashScreenActivity

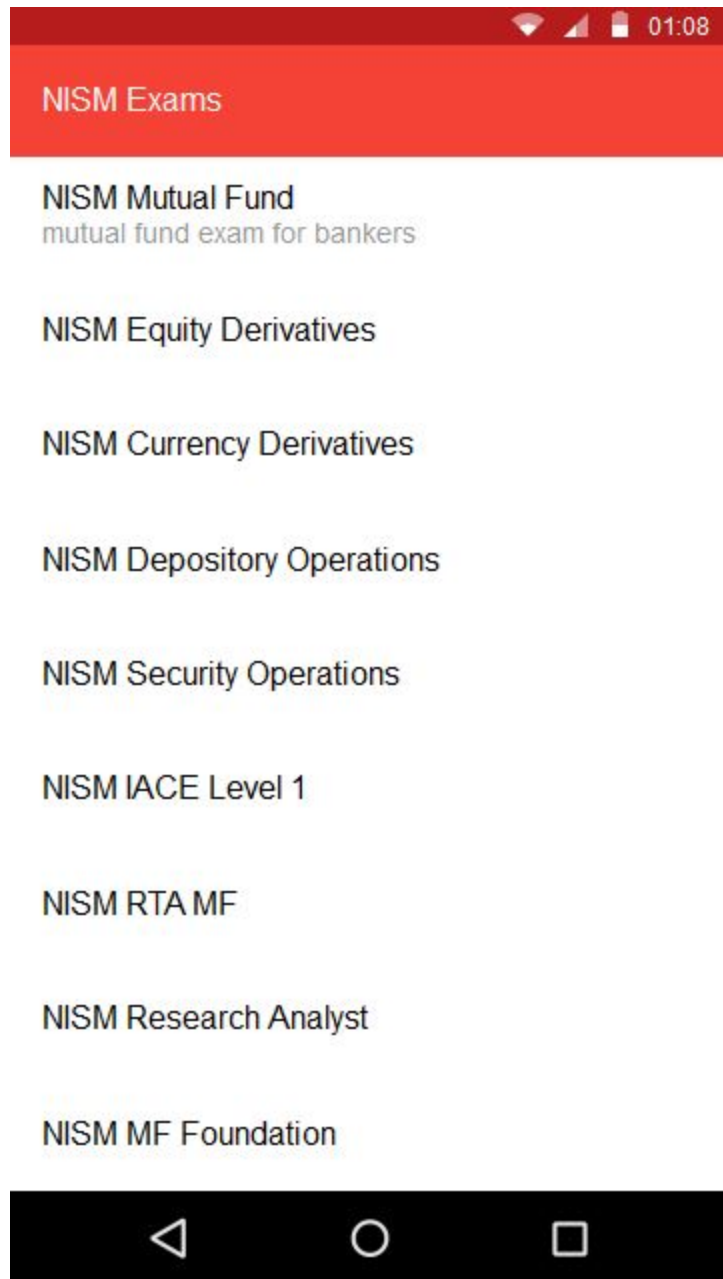
Splash Screen Activity is used to run some initial checks like battery status,location, network availability and Crashlytics,analytics implementation
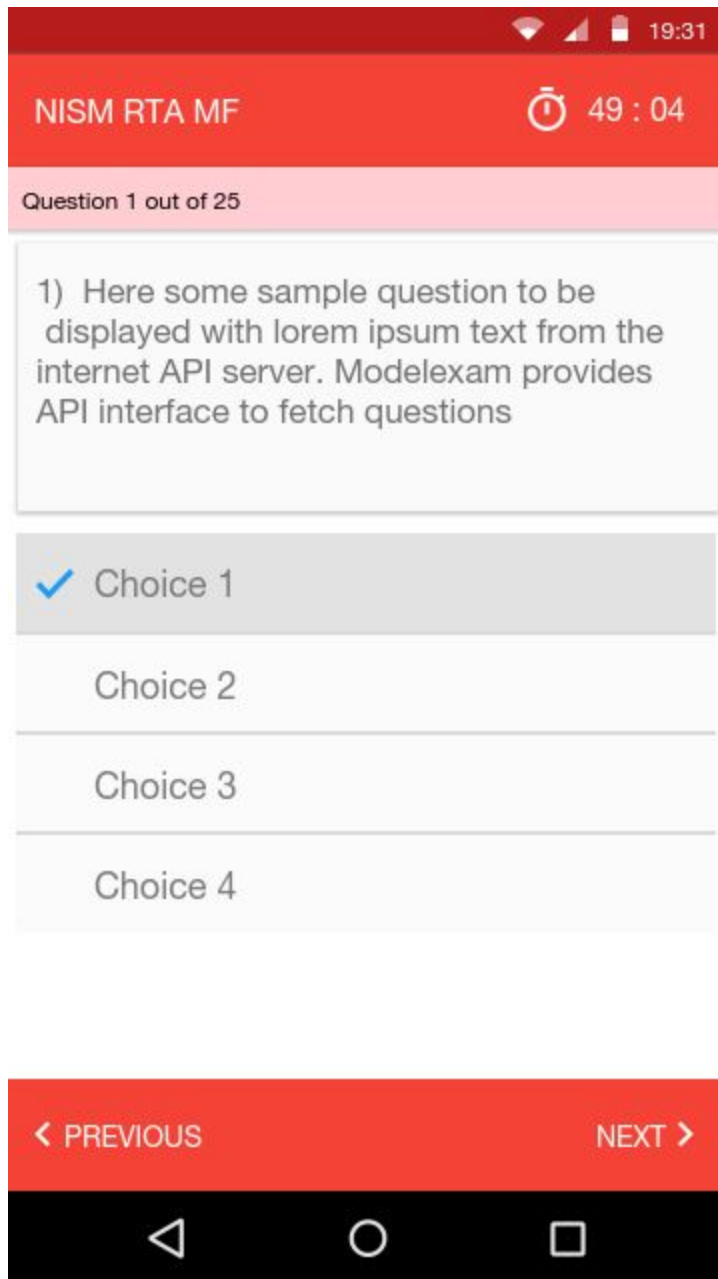
Google Oauth Login Screen

Login with Google implementation for Premium exam takers and maintaining their profile.

Home Screen

Simple Screen to Display all available exams to take test.(It can contain both premium and freemium exams)

Exam view

Simple Mockup on Examview where questions will be loaded dynamically with fragments in activity and maintaining the lifecycle.

Key Considerations

How will your app handle data persistence?

Basically there will be two types of exams, Demo and Premium.
Demo exam will be basically flat files in JSON format in assets directory.(since requested by business owner)
Premium questions will be available in Server provided with API, i'll be downloading it through network calls and build a content provider to support viewing answers even in offline after exam ends.

Describe any corner cases in the UX.

- Since the users of this Application will be 30-40 aged Bank employee peoples, we are trying to reduce the number of clicks to start the exam. Mostly with two taps one can start the exam right away(simple UX)
- While taking test, Timer is provided in Toolbar and as well in notification to alert the user that exam is going on.
- Best UX design will be followed namely Material steppers(official google material design docs) for showing many questions in sequence to the user.

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

1) Retrofit with JSON converters - for accessing API for premium questions online.
2) Fortumo android SDK - for implementing Carrier billing
3) Crashalytics - to monitor the user behaviour and crashes,sessions.
4) Google Analytics - to add google analytics to app to measure behaviour
5) Google Plus Login - Playservice to use google Oauth Login
6) Google Design Support library - to implement material design across All platforms

Describe how you will implement Google Play Services.

1) Google Plus Login - Oauth login playservice will be added to application in order to provide seamless login for users across platforms

I hope this play service will be enough and useful for application smooth and running.

Next Steps: Required Tasks

Task 1: Project Setup

First of all i need to show the design and user experience, Application flow to the End user as well business leads to approve the design.
You may want to list the subtasks. For example:
- Application flow(providing best UX)
- Easy understanding Navigation for end users

Consider now, App flow and UX has been approved by business lead. Moving to Question API to learn and deciding the task, mappings.

Task 2: Implement UI for Each Activity and Fragment

- Designing Material Stepper flow for Question Answer exam dashboard
- Profile activity showing credits in their account(Google+ login)
- Pricing activity
- Listing Exams with freemium and premium exams

Task 3: Implement Google Plus Login

Need to implement Google Oauth login for application where as securely passing the token to API server and validating the client

- Getting Appropriate permissions
- Adding Google plus login code with Activity results handling exceptions
- Transferring the access token to API server to fetch user details
- Showing profile activity with Logout button

Task 4: Designing Content Provider for Question and Answers

Designing questions and answer sqlite tables supported with content provider support
- Design table to hold the records
- Design content provider to handle questions related queries

Task 5: Using Network API calls

Designing Network API calls with handling the questions in JSON format to POJO modal using Retrofit library
- Design modal class for question database
- Adding callbacks and processing code
- Handling network exceptions
- Defining API Call and interface

Task 5: UI components

Utilizing relevant UI components for application to provide best UX

- Adding Recyclerview to list exams
- Cardview + recyclerview for Pricing list
- **App Widget for displaying current exams list or to-do exam list(saving it to preference)**
- Navigation drawer to have seamless UI navigation experience across the app
- Exam view combination of questions and multiple choice questions coordinating with Material Steppers for providing best UX.

Respected,

I've given my full effort on designing and Business Clients expectation. Thanks to Udacity and reviewer. Hoping for constructive feedback and passing this project.