GitHub Username: abskaushal

App Name: MobiTest

Description

This app brings teachers and students on common platform to provide online assessment. The teachers can create test having multiple choice questions and then invite the students to take that test. The students can take those tests remotely via internet and then check their performance. Not only teachers even the parents or anyone can use this app to create some assessment in the form of multiple choice questions and then invite the people to take part in that assessment.

Intended User

This app can be used by:-

1) Teachers
2) Students
   Even parents can use this app to assess their children in a more interactive way.

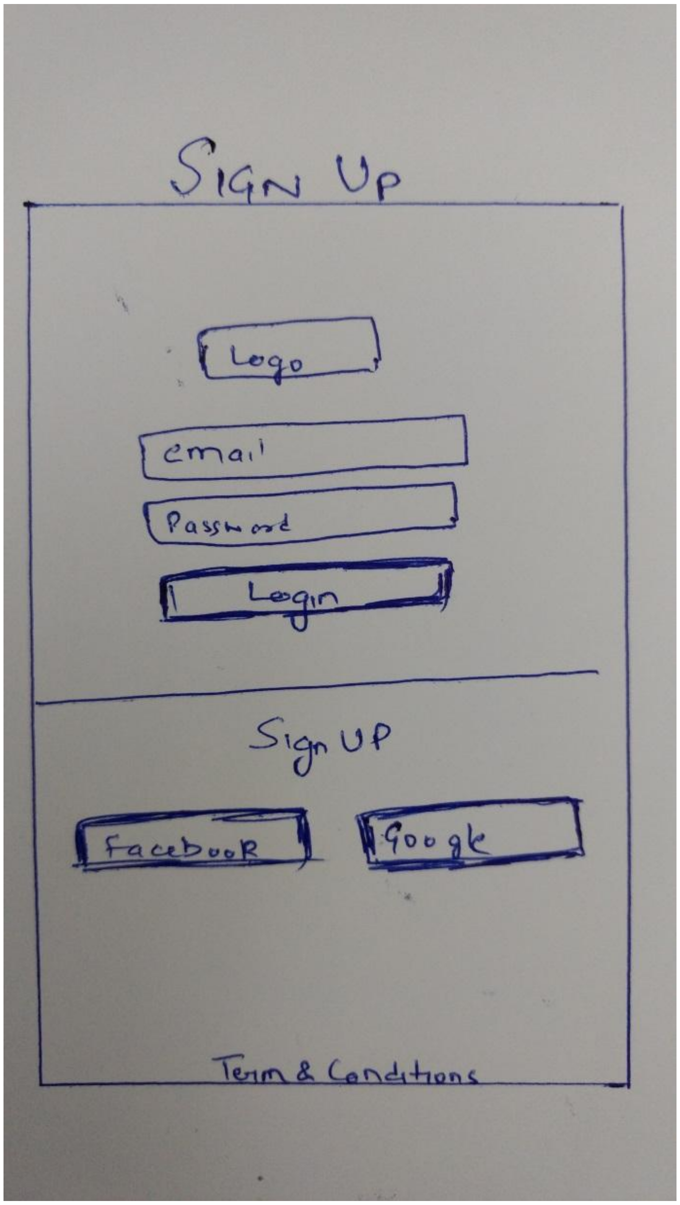
Features

The following will be the main features of the app:

- Account creation via Google and Facebook
- Create test and store them on server
- Invite students/participants by using their email ids
- Students can take those test which are time bound and has to finish within the time limit specified by the teacher.
- Teacher will announce the result and each student can see its own performance by refreshing the app.

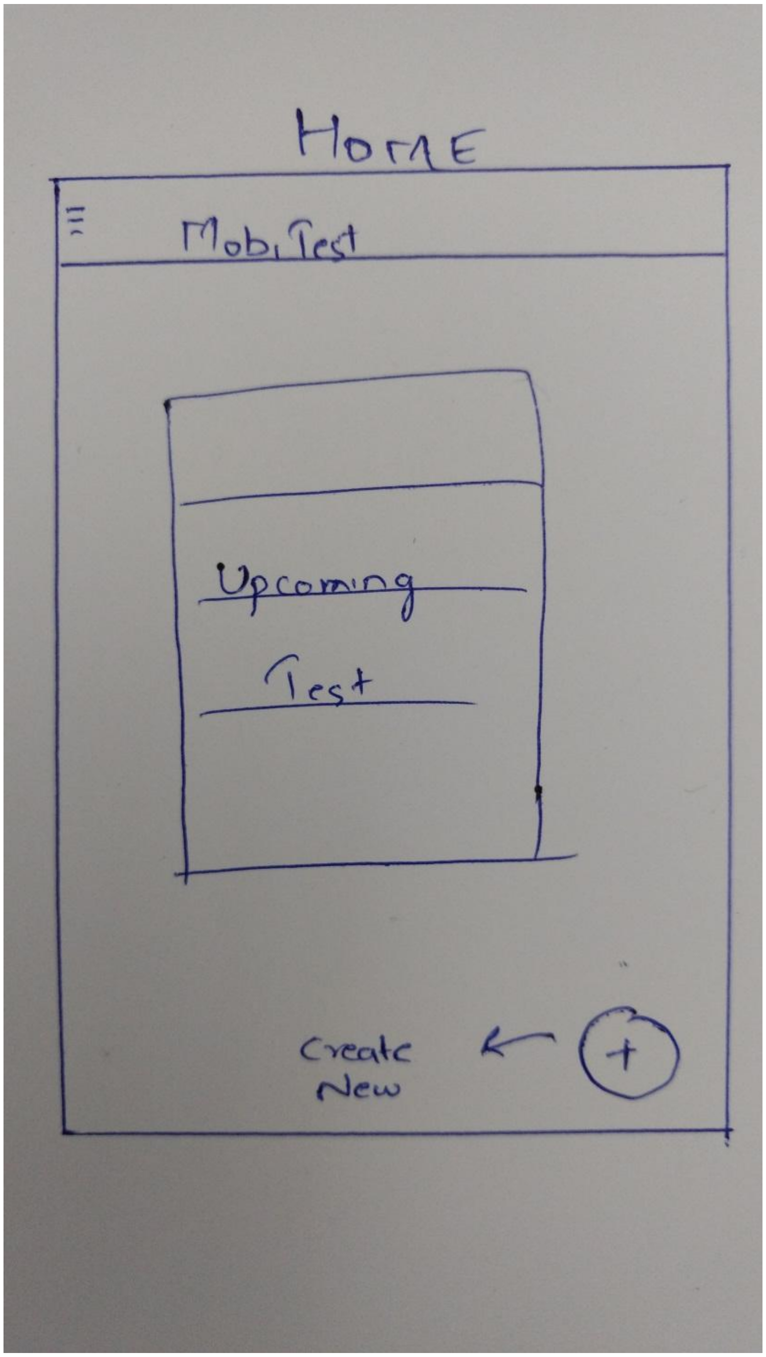
User Interface Mocks

Screen 1: Signup/Login screen



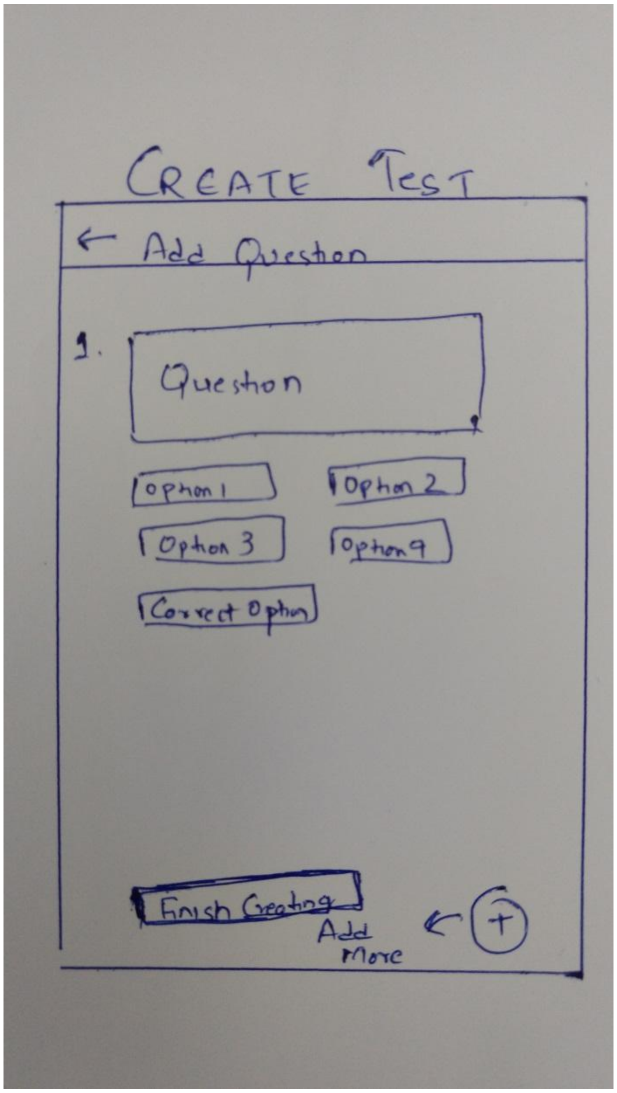
This will be the first screen of the app. User can signup if they don’t have account else they can login via this screen.

Screen 2: Home screen for teachers



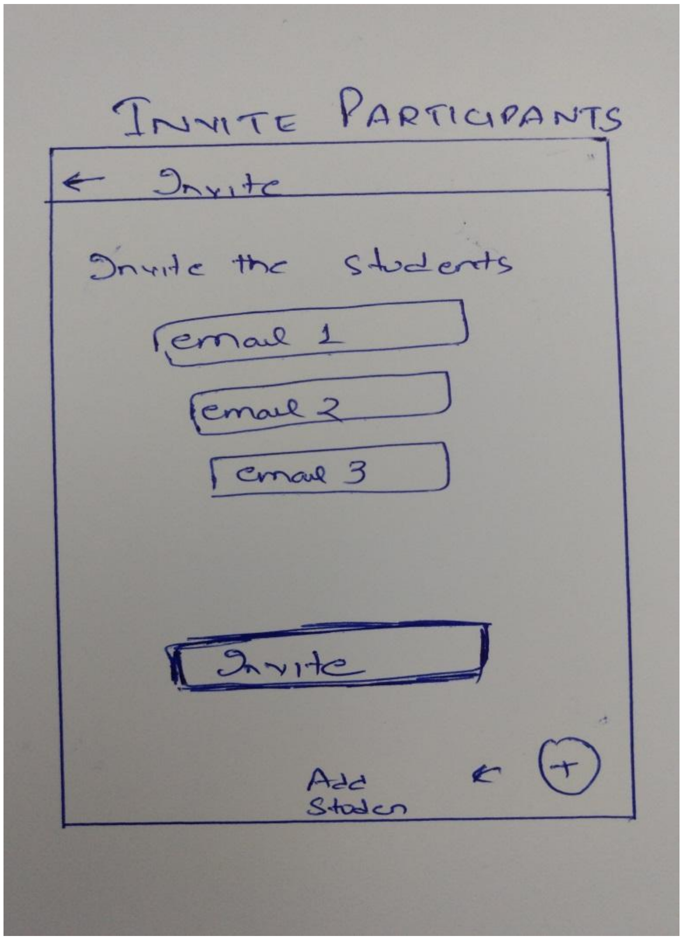
This will be the home screen for Teacher. He will see all future tests which he have created. Also he can use the FAB button at bottom right to create new test.

Screen 3: Create Test screen



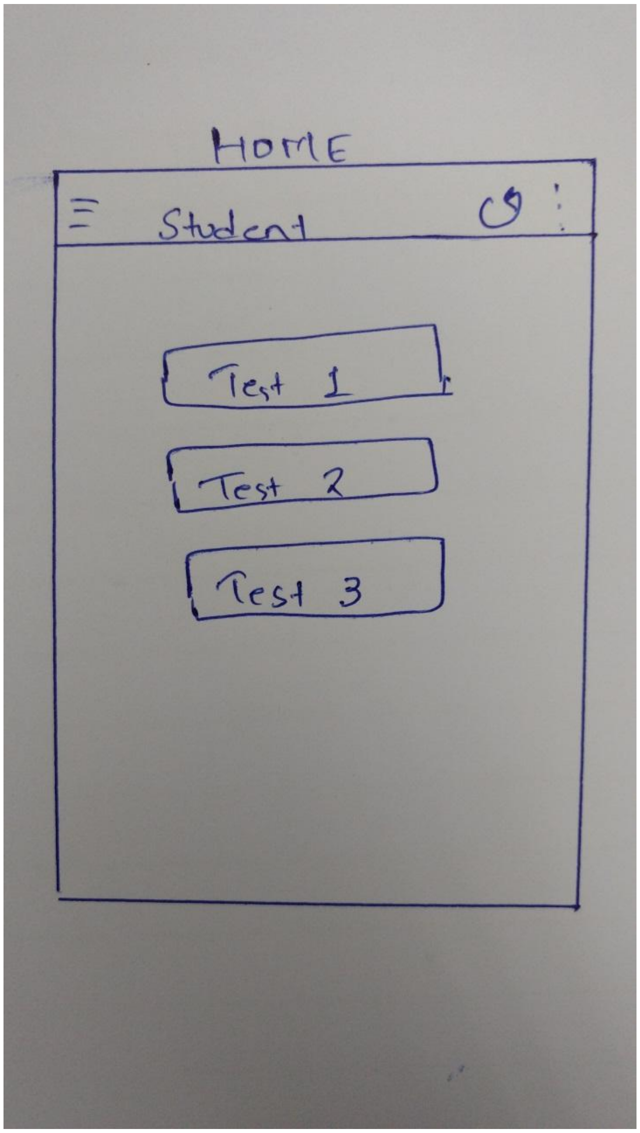
This screen will be open when Teacher presses FAB button to create new test. He will add details about the test(description, duration etc) and also add questions one by one. The FAB button in this screen will add more questions. The Teacher then presses Finish Creating button to create test.

Screen 4: Invite participant screen for Teacher



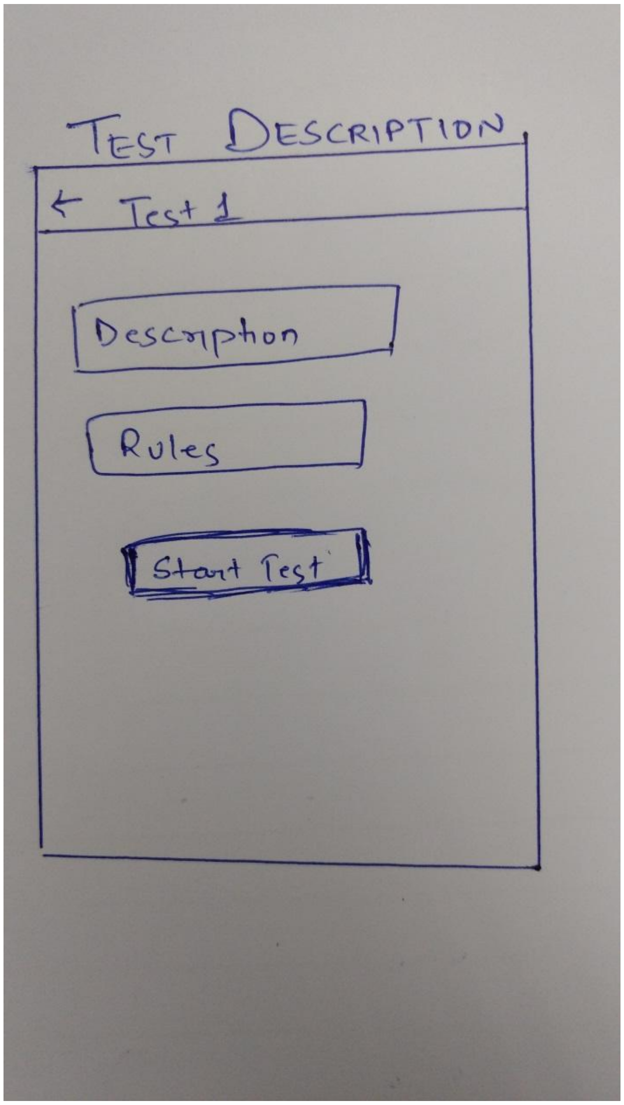
After creating test, teacher will invite participants by using email ids. The FAB button will add edit text one by one to add more email ids. Then he will press Invite button to send invites to the students.

Screen 5: Home screen for participants



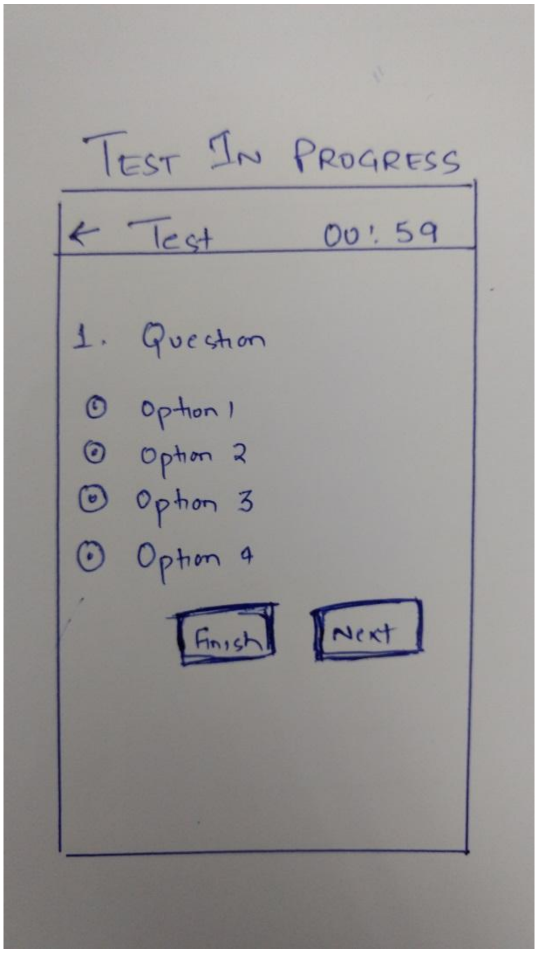
This is the home screen for participants. They will see all upcoming tests for which they are invited. They can click on any test to participate.

Screen 6: Test description screen



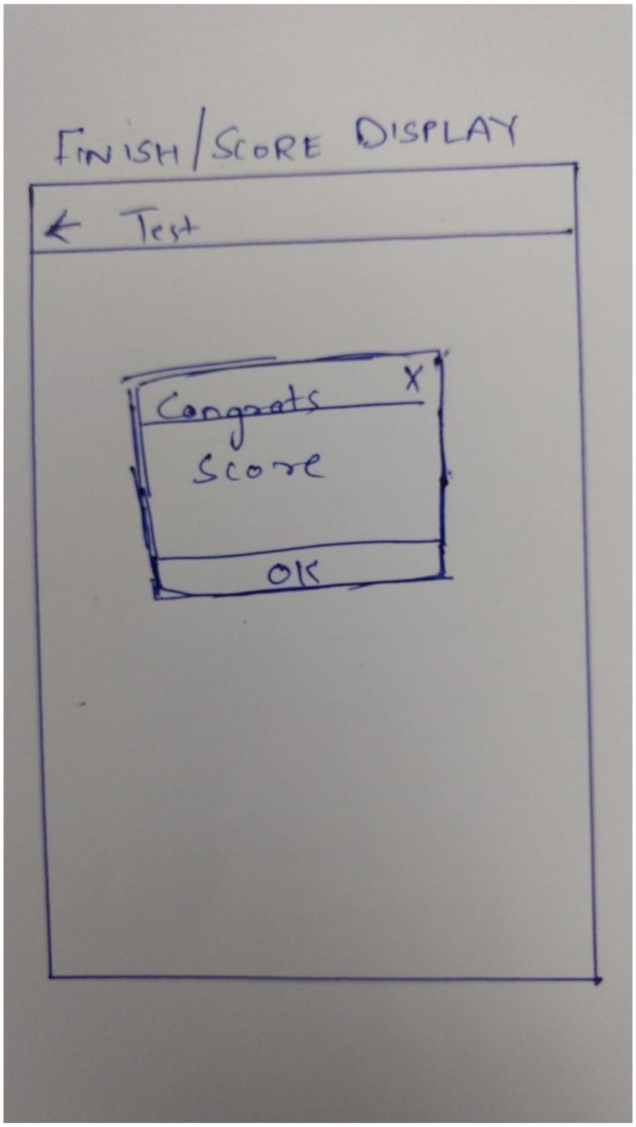
This screen will be opened when the participant click on the particular test. They will see the description and rules of the test. The Start Test button will be clicked to start the test. Test once started cannot be paused.

Screen 7: Test in progress screen



This is the Test in progress screen. The participant will select the appropriate answers and proceed next or they can finish the test.

Screen 8: Score display screen



The participants when clicked on the Finish button will be displayed with their score and it will be updated on the server also so that teacher can see the individuals score.

Key Considerations

How will your app handle data persistence?

There are 2 data storages:-

1) Server: The server will store the following information:-

a) User information like email id, names and type of user (student or teacher).
b) It also stores the test created by teachers for some particular interval of time like 3-5 days within which all the students have to take that test. After that the test data will be auto deleted to save the server storage.

2) Local app: The app will store the following data locally:-
    a. The app will store the user session in the SharedPreferences which will be cleared when user logout.
    b. The test will be fetched for students and stored in Sqlite database. Once student finish the test the score will be displayed and test will be deleted from the database.

Describe any corner cases in the UX.

No corner case in the app till now. I will create layouts for tablet also for better readability.

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

The following libraries will be used:

1) Picasso: For displaying and caching of images
2) Recycler View: For better implementation of GridView
3) Support Library: Will be using Appcompat for backward compatibility.
   More libraries will be added if required.

Describe how you will implement Google Play Services.

I will be using Google signup which will use Google play services. In the future scope I will be adding GCM also for push notifications.

Next Steps: Required Tasks

The following are the high level steps involved in the development activity of this app:-

Task 1: Server Setup

I will setup the server with the required database schema and write the following web services:-

1) Account creation
2) Login
3) Create Test
4) Invite participants
5) Fetch the Test (for Students)
6) End Test (for Teacher)

Task 2: Project Setup

- Create initial project.
- Implement Google and Facebook signup

Task 3: Create Test

The teacher will be able to create test by adding multiple choice questions one by one:-

- Add questions one by one.
- Submit test to the server.
- Invite participants.
- End the test and check the list of participants with their scores.

Task 4: Take Test

The steps are:

- Students refresh their home page and fetch if any invite is there.
- Fetch the questions from server and store in the database.
- Start the test and terminate it either time based or by student and display the score.
- The result will be stored on the server which only teacher can see.

Note: There are lot of features which I have planned for the next release. As I have time constraint so I am implementing only this much in 1st release. The future release will have:-

- Create descriptive questions also which teacher can read and rate.
- Add image support to the questions for pictorial representation.
- Push notifications like notify the students as soon as teacher invites them.
- Students should also be able to see entire score board.