Design Credit Project Report

Submitted by : - Aditya Gadhavi (B19EE004)
                  Ajay Kumar Meena (B19EE005)

Mentor : - Dr. Sumit Kalra

Table of Content -

Goal and Motivation : Many exams are now administered online because to COVID and the increased usage of technology in recent years. Our project's aim is to develop an application that can work even in areas with poor internet connectivity.

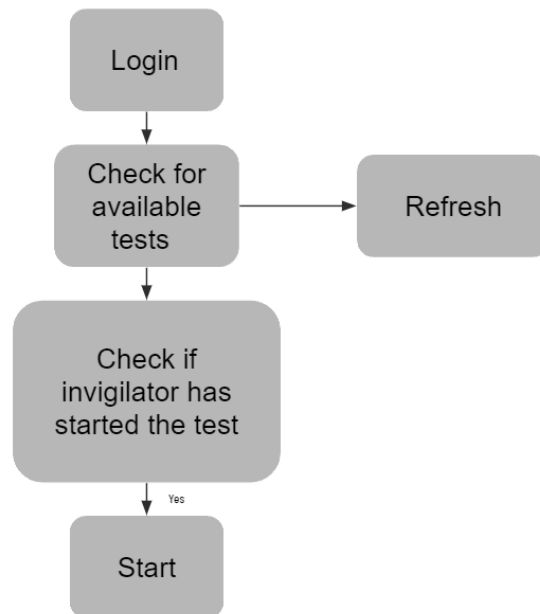**Workflow and Functionalities :** The four components of the application's workflow are-

- **User -** Both the invigilator and the person giving the exam are considered as users. They can interact and modify the application and we created separate user interfaces for both.

- **Lowering the FPS -** We select one picture buffer for one instance from an array of frames before uploading it to the database. This is accomplished using the 'Timer' library in java where we are setting the time for which one picture is extracted. Currently we have set it to 0.2 frames per second.

- **Image compression -** For saving the data which is consumed for one frame upload, we are also compressing the image size for selected pictures. This is done using the **ImageReader** class. Right now, we can reduce an image's size by more than 99%.

- **Database -** Finally the extracted pictures are numbered in sequence and uploaded to the database. There are two types of database which fulfill separate tasks -
    - Real Time - Here the exam data is stored. Through it the invigilator can add and modify exams.

    - Storage– The image buffers are saved in this database.

**Code flow :** For three different functions, we have developed three different classes. -

- **Activity -** This class is where most of the main processes of code are executed. This can further be divided into two parts -

    Main - The processes such as logging in and starting the test are included in this. The flow for main activity is like -

```
      ┌──────────┐
      │  Login   │
      └──────────┘
            │
            ▼
    ┌──────────────┐          ┌──────────┐
    │  Check for   │          │          │
    │  available   │ ───────▶ │ Refresh  │
    │    tests     │          │          │
    └──────────────┘          └──────────┘
            │
            ▼
    ┌──────────────┐
    │   Check if   │
    │ invigilator  │
    │ has started  │
    │   the test   │
    └──────────────┘
            │ Yes
            ▼
      ┌──────────┐
      │  Start   │
      └──────────┘
```

    Test - All other further processes regarding tests are included in this.

- **Camera Handler -** All processes which related to cameras such as getting images and compressing them are included in this. This data is connected to activity through an interface. If we want to understand its processing then a basic diagram can be shows as -

OnSurfaceTextureAvailable
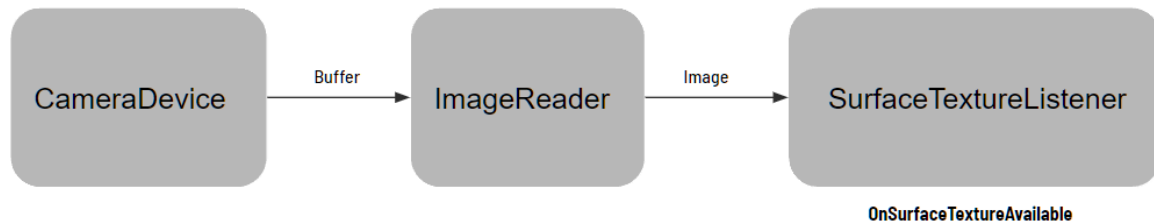
- **Repository -** Processes related to database such as uploading pictures to database or fetching test data from real time database are included in this. There are three main processes -

> **Get Exam Data -** This is to fetch test data from the real time database.

> **Frame Upload -** The extracted frames are uploaded to the local database.

> **Warning -** This functionality is included for invigilators where they can warn someone if they find suspicious activity.

This data is once again given to **Activity** by use of interface.

In case if the student is disconnected from the network, the firebase storage request persists as long as the application is open. Once the connection is restored the files will be uploaded in their correct index and the invigilator will be notified.

If the student tries to close the application in the middle of examination, an alert is sent to the invigilator notifying them of the same.

Finally, this is the flow of how various components are interacting with each other -