

**Final Assignment**

**Course name:** Mobile Device Programming

**Student Name:** PATWARY MOHAMMAD SAZZAD HOSSAIN

**Student Class:** 19lq

**Student Number:** 1911521212

**The requirements are as follows:**

1. No less than 5 pages (including activity or fragment)
2. Use MVVM pattern
3. Use database to store data

**Activity**

Activities play a crucial role in Android application development as they define the user interface and the actions that can be performed within the app. Each activity is designed to perform a specific task and provides a unique screen with which the user can interact. For example, an email app may have separate activities for composing an email, viewing emails, and searching for emails.

**Fragment**

In contrast, a fragment represents a reusable component of the app's UI. It defines and manages its own layout, has its own lifecycle, and can handle its own input events. However, fragments cannot stand alone and must be hosted by an activity or another fragment. This distinction highlights the fact that a fragment relies on an activity for its existence, representing only a part of the UI. The activity acts as the container for all UI elements, including fragments, and is essential for the creation of a user interface. Without activities, the UI would not exist.

**MVVM pattern**

The view model in MVVM acts as a bridge between the model and the view, exposing the data objects from the model in a way that can be easily managed and presented to the user. It is primarily responsible for handling the view's display logic and may implement a mediator pattern to manage access to the back-end logic. The view model acts as the mediator, organizing access to the business logic around the set of use cases supported by the view. In MVVM, the view model is more model-focused than view-focused, and its main purpose is to facilitate the presentation of data to the user. It allows developers to create dynamic and interactive user interfaces that respond to changes in the model, while also abstracting the underlying business logic.

This results in a more maintainable and scalable codebase, where changes to the view or the model do not affect the other components.

**Database to Store data**

SQLite is a relational database management system, similar to other popular systems such as Oracle, MySQL, and PostgreSQL.

SQLite is known for its small size and public domain license, making it one of the most widely deployed database engines in the world. It is used in a variety of software systems, from embedded systems, browsers, and operating systems. The wide deployment of SQLite is a testament to its reliability and versatility. In the Android environment, SQLite is built-in, and each application's database files are stored in a private, inaccessible disk space. This prevents one application from accessing another application's data, ensuring the privacy and security of user data. Android's implementation of SQLite provides a convenient and efficient way for developers to manage and store data for their applications.

With its small footprint and ability to run efficiently on mobile devices, it is an excellent choice for developing database-driven Android applications.

**Student Management App**













