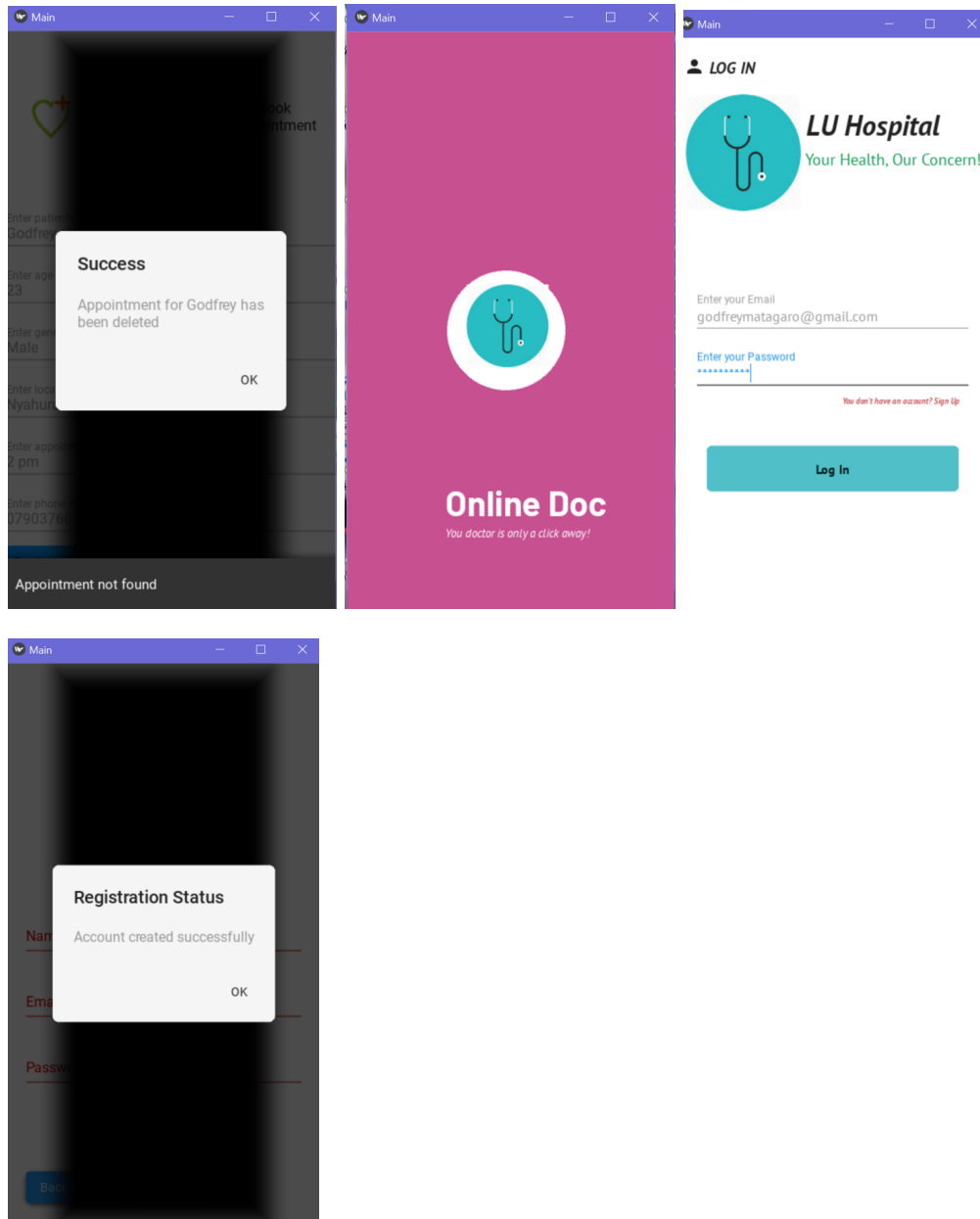


# **Title: Designing a Doctor's Appointment App with Enhanced Authentication and Database Integration**

## **Introduction:**

In this project, I designed a doctor's appointment app using the KivyMD framework. The app allows patients to register, log in, and book or delete appointments with their doctors. To ensure security, the app includes enhanced authentication that verifies user data against a MySQL database, and password hashing using a Python hashing algorithm. In addition, sensitive data can be stored in a separate configuration file to avoid exposing it in the code. The app also includes snackbar classes to alert users of incorrect login credentials, successful appointment bookings, and appointment deletions. The font used throughout the app is PT Sans regular 400. Separate KV language files to help with separating the app's script logic from UI design. The app is scalable and can be expanded in the future to include additional functionalities such as doctor presence and telemedicine. This can be shown in the snippets below.



## Design Process:

The first step in designing the doctor's appointment app was to create a mockup of the app's screens and user flow. I used the KivyMD framework to create the app's UI, and designed a loading screen, a login screen, a registration screen, and an appointments screen. The login screen included a text button that users would click to access the registration screen. I also designed two appointment buttons on the appointments screen, one for booking and one for deleting appointments.

Next, I implemented enhanced authentication in the login script to verify user data against a MySQL database. The database connection details were stored in a separate database.py file, and sensitive data could be stored in a separate database.ini file. I also added password hashing to both the registration and login scripts using a Python hashing algorithm to ensure that user passwords were not exposed in plain text.

To provide feedback to users, I added snackbar classes to the app's login and appointments screens. The snackbar classes would alert users if they entered incorrect login credentials, if they successfully booked an appointment, or if an appointment was deleted.

### **Future Developments:**

In the future, the doctor's appointment app can be expanded to include additional functionalities such as doctor presence and telemedicine. The app could allow doctors to set their availability and schedule appointments with patients, and patients could have the option to book telemedicine appointments if they cannot physically visit the doctor's office. These features would make the app more convenient and accessible for users, and could be implemented using the existing database and authentication systems.

### **Conclusion:**

In conclusion, the doctor's appointment app is a secure and user-friendly tool for patients to manage their appointments with doctors. The app uses enhanced authentication to verify user data against a MySQL database, and password hashing to protect user passwords. The app also includes snackbar classes to provide feedback to users, and is scalable for future developments such as doctor presence and telemedicine.