SE4010 – Current Trends in SE

4th Year – Software Engineering Specialization

Faculty of Computing

SLIIT

2023 – Practical – Integrating Firebase with Flutter

Introduction:

In this lab you will modify the app that you created on the previous practical. Update the previous app to connect with Firebase. This app will allow the user to add and delete tasks, as well as mark them as complete. The tasks will be stored in Firebase Firestore, a NoSQL database.

Objectives:

- To gain familiarity with Flutter and its core concepts.
- To learn how to integrate Firebase with a Flutter app.
- To understand how to store data in Firebase Firestore and retrieve it in a Flutter app.
- To understand how to use Firebase Authentication to authenticate users in a Flutter app.

Requirements:

- A computer with the Flutter SDK installed. You can follow the instructions in the official Flutter website to install the SDK.
- A code editor of your choice, such as Visual Studio Code.
- Basic knowledge of Dart programming language.
- A Firebase account. You can sign up for a free account on the Firebase website.

Steps:

1. Set up Firebase in your Flutter project by following the instructions in the official Firebase documentation.
2. Define a model class for the to-do tasks, which will include properties such as task name and completion status.
3. Modify Existing or Create a stateful widget for the home screen of the app, which will display the list of to-do tasks. Retrieve the tasks from Firebase Firestore.
4. Implement a form to allow the user to add new tasks to the list. Save the tasks to Firebase Firestore.
5. Add the ability to mark tasks as complete and delete tasks from the list if not implemented yet. Update the tasks in Firebase Firestore accordingly.
6. Implement Firebase Authentication to allow users to sign up and log in to the app.

7. Style the app to make it look attractive and easy to use.
8. Test the app to ensure it is working as expected.

Tips:

- Make use of the official Flutter and Firebase documentation and tutorials for reference.
- Use comments in your code to explain what different parts of the code are doing.
- Start with the basic functionality and add more features as you go along.
- Make sure to test the app thoroughly to catch any bugs.