

TECHNICAL PROGRAMMING III (TPG316C)

ASSIGNMENT 1

DUE: 20 MARCH 2024 23:59

TYPE: INDIVIDUAL

TOTAL: 100

Learning Goals

This assignment aims to enhance Flutter application development competence by integrating forms, editTexts, buttons, and real-time feedback, implementing sophisticated password validation techniques, and fostering reflective cognition. It aims to demonstrate a comprehensive understanding of Flutter principles and mobile application development skills.

Summary

Your objective is to craft a comprehensive Flutter application centred around your own reflections for Term 1 of TPG316C in 2024. The application comprises a login page, a main page, and two additional pages ("Understood" and "Not Understood"). The main page should feature "*EditTexts*" for expressing both your understanding and areas where challenges persist. Buttons seamlessly navigate to specialized reflection pages.

Introduction

As a TPG316C student, you've likely encountered challenges in fully grasping Flutter concepts like UI design, state management, routes, and forms. Recognizing this, your lecturer encourages you to embark on a self-driven learning journey by creating a personalized app – the "Reflective App". This Android application empowers you to reflect on your understanding of these crucial concepts. Through engaging prompts, you can articulate what you comprehend well and pinpoint areas for improvement. By taking charge of your learning experience, the Reflective App aims to enhance your ability in Flutter development, ensuring a personalized and insightful approach to mastering key principles in mobile application development.

Instructions

1. Login Page

- Develop a login page that validates email and password inputs.
- Passwords must meet specific criteria: **8 characters** and the inclusion of an '@' symbol.
- Upon successful login, smoothly transition to the main page.
- The Application must have an AppBar title as "Reflective App".

2. Main Page

- Design an appealing main page featuring two EditTexts aligned with buttons.
- Capture what you understand and what you find challenging in the respective EditTexts.
- Display previously entered email address prominently on the main page.
- Buttons should automatically lead to specialized reflection pages.

3. Reflection Pages

- Implement two additional pages catering to "understood page" and "not understood page."
- Display entered email address prominently on the reflection pages.

3.1. Understood Page

- Create a dedicated page "Understood" specifically used strategies.
- Inclusion of EditText for students to write strategies used to understand the concepts.
- Button functionality to display a Snackbar to provide students with immediate acknowledgment "e.g.: ***mmbele@cut.ac.za, your strategies are received***".

3.2. Not Understood Page

- Create a dedicated page "Not Understood" specifically for your plan to make improvements based on your reflections.
- Inclusion of EditText for students to write plans to use to improve their understanding.
- Button functionality to display a Snackbar to provide students with immediate acknowledgment "eg: ***mmbele@cut.ac.za, improvement plan acknowledged***".

4. Screenshots of the application which must include information from EditTexts.

- Create a minimum of 5 pages document, which includes screenshots labelled with captions and brief explanations of the pages.
- Ensure that the screenshots include the information entered on the EditTexts, providing a comprehensive view of the application's design and user-generated content.

5. Tasks

- Implement coding best practices and add comments where needed to make your coding more understandable.
- You must make use of Provider and Routes and forms as explained in class.
- Do not forget the necessary packages needed for state management and routes.
- Make use of extracting of widgets and structuring of folders to make your coding easy to follow.

Submission

- Please **do flutter clean** first before submitting your project on ethuto (this will reduce the size of the file), I will do **flutter pub get** on my side.
- Zip your project into a file called **studentnumber_studentname.zip** and upload it via ethuto before **20 MARCH 2024 23:59**.
- The zipped folder must also include the **pdf file** for your screenshots document.