

Hands-On TEST 01 (SEC 003)

DATE: 17th Oct., 2024

Marks/Weight: 100/25%

Time: 2.30 pm to 4.25 pm

Be sure to read the following general instructions carefully:

- This hands-on test **must be completed individually** by all the students. **DO NOT** give your solution to others, otherwise there will be serious consequences. You are not allowed to share, communicate, e-mail, text during the test. See College Academic Policy
- Read the project **naming and submission guidelines** as explained in the following sections
- You should **provide the runtime screenshots** of your solution and **submit the project and screenshots through the Test01 link on eCentennial**.

IDE: Android Studio – Koala Version and Kotlin Jetpack Compose

Android Project Naming rules:

1. You must name your **Android Studio project** according to the following rule:
YourFullName_COMP304_003_Test01
Example: NickAdam_COMP304-003_Test01
2. Use minSdkVersion **24** and targetSdkVersion **34 or higher**
3. **Package name** must be **com.yourfirstname.yourlastname**, for example: com.nick.adam
4. **Main activity** should be named: YourFirstNameActivity, for example: NickActivity.
5. The **second activity** should be named YourLastNameActivity, for example: AdamActivity
6. Must provide your **student name and number as a comment** at the top of each activity

Submission rules:

1. Archive your project in a **zip file** named according to the following rule:
YourFullName_COMP304-003_Test01.zip
Example: NickAdam_COMP304-003_Test01.zip
2. Upload the project through the Hands-On Test 1 link on **eCentennial**.

Exercise #1:

Develop an Android application that demonstrates understanding of basic Android development using Kotlin, Jetpack Compose, Material Design 3, and MVVM architecture.

Description: Create an Android app named "PersonalContacts" that allows users to manage their contacts. The app should include the following features:

1. **Main Activity:**
 - Displays a label (MY PERSONAL CONTACTS) and an image button.
 - When the user clicks on the image button, navigate to the second activity.

2. Second Activity:

- Allows users to add a new contact with fields for name, address, and contact number.
- Uses either checkboxes or radio buttons to select the contact type (e.g., Friend or Family). Use radio buttons if your first name starts with A-N, otherwise use checkboxes.
- Displays a Snackbar with the entered contact details upon clicking the "ADD CONTACT" button.
- Displays a list of added contacts using a LazyColumn.

Requirements:**1. Main Activity (UI and Event Handling):**

- Use Jetpack Compose for the UI components.
- Implement navigation to the second activity.

2. Second Activity (UI, Event Handling, MVVM Architecture):

- Use Jetpack Compose for UI components.
- Implement form fields for entering contact details.
- Implement checkboxes or radio buttons for contact type selection.
- Display the entered contact details in a Snackbar.
- Use ViewModel for managing UI-related data.
- Use LazyColumn to display the list of added contacts.

Evaluation:

Item	Percentage of Total Mark	Details
Functionality:	80%	
- Main Activity (UI and Event Handling)	20%	Ensure the main activity displays correctly and handles navigation.
- Second Activity (UI, Event Handling, MVVM)	30%	Correct implementation of UI, form handling, and ViewModel integration.
- Displaying contact list using LazyColumn	20%	Correct implementation of LazyColumn to display contacts.
- Use of Material Design 3 components	10%	Ensure Material Design 3 principles are applied.
Friendliness:	15%	
- Alignments of UI controls	10%	UI controls should be properly aligned and organized.
- Friendly I/O	5%	The app should provide a user-friendly interface.
Comments, Correct Naming of Variables, Methods, Classes, etc.	5%	Code should be well-documented with appropriate comments and follow proper naming conventions.
Total	100%	