|  |  |  |
| --- | --- | --- |
| Course: | **Mobile Application Development** | Date: 25/03/2025 |
| Course Code: | CSL 341 | Session: IV |
| Faculty’s Name: | Muhammad Ahtesham Noor | Max Marks: 20 |
| Time Allowed: | 1.5 Hours | Total Pages: (2) |

**INSTRUCTIONS:**

1. This is closed book exam. Communication devices and any written material is strictly prohibited.
2. All questions are compulsory.

Student’s Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Enroll No:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(USE CAPITAL LETTERS)

**Tools Required**

1. Android Studio/VS Code with Flutter & Dart plugins.
2. Flutter SDK (latest stable version).
3. Emulator/Physical Device (Android/iOS).

**General Instructions**

1. Complete all experiments in sequence.
2. Submit source code, screenshots, and a lab report.
3. Viva voce will assess conceptual understanding.

**Experiment 1: Setup and Basic App Creation (3)**

**Objective:** Install Flutter and create a simple "Hello World" app.

**Tasks:**

1. Install Flutter SDK and set up Android Studio/VS Code.
2. Create a new Flutter project named lab1\_hello\_world.
3. Replace the default code in main.dart to display "Hello, [Your Name]" in the center of the screen.
4. Run the app on an emulator and capture a screenshot.

**Expected Outcome:**

* A working app with centered text.
* Understanding of the MaterialApp, Scaffold, and Center widgets.

**Pre-Lab Question**

* What is the role of the pubspec.yaml file?

**Experiment 2: Dart Programming Basics (3)**

**Objective:** Demonstrate understanding of Dart syntax and core concepts.

**Tasks:**

1. Write a Dart function to check if a number is prime.
2. Create a list of integers and use map() to square each element.
3. Implement a simple class Student with properties name and rollNo.

**Expected Outcome:**

* A Dart script with working functions and classes.

**Sample Code Snippet:**

dart

Copy

bool isPrime(int n) {

// Your code here

}

**Experiment 3: Widgets and Layouts (3)**

**Objective:** Build a UI using Flutter widgets and layouts.

**Tasks:**

1. Create a Column with three Text widgets showing your name, enrollment number, and course.
2. Add a Row with two IconButton widgets (e.g., "Like" and "Share").
3. Use Container with padding and margin to style the UI.

**Expected Outcome:**

* A responsive UI with proper alignment and styling.

**Pre-Lab Question**

* What is the difference between Row and Column widgets?

**Experiment 4: Gestures and State Management (3)**

**Objective:** Implement interactivity and state management using setState.

**Tasks:**

1. Create a counter app with a Text widget showing the count.
2. Add a FloatingActionButton to increment the counter.
3. Add a GestureDetector to reset the counter on a long press.

**Expected Outcome:**

* A counter app with working increment and reset functionality.

**Code Hint:**

dart

Copy

int \_counter = 0;

void \_incrementCounter() => setState(() => \_counter++);

**Experiment 5: Animation (3)**

**Objective:** Create a simple animation.

**Tasks:**

1. Animate a Container to change its color and size over 3 seconds.
2. Use AnimationController and Tween classes.

**Expected Outcome:**

* A smooth color/size transition animation.

**Experiment 6: Android-Specific Code (5)**

**Objective:** Use platform channels to write Android-native code.

**Tasks:**

1. Create a method channel to fetch the Android device’s battery level.
2. Display the battery level in a Text widget.

**Code Reference:**

dart

Copy

// Method channel setup

const platform = MethodChannel('battery\_channel');

Future<void> \_getBatteryLevel() async {

// Invoke Android-specific code

}

**Submission Checklist**

1. Source code for all experiments.
2. Screenshots of app outputs.
3. Lab report explaining challenges faced and solutions.

**Best of Luck**