



Gyanmanjari
Innovative University

Course Syllabus
Gyanmanjari College of Computer Application
Semester-6 (BCA)

Subject: Flutter for Mobile Application Architecture and Development - BCAMA10326

Type of course: Major Core

Prerequisite: Basic knowledge of Programming

Rationale:

Mobile applications have become an essential component of businesses across industries in today's technology-driven world. Organizations increasingly seek cross-platform solutions to reach wider audiences with reduced development effort. Flutter, an open-source framework by Google, enables developers to create high-performance, native-like mobile applications for both Android and iOS from a single codebase using the Dart language.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks					Total Marks
CI	T	P	C	SEE		CCE			
				Theory	Practical	MSE	LWA	ALA	
3	0	2	4	75	25	30	20	50	200

Legends: CI-Class Room Instructions; T – Tutorial; P - Practical; C – Credit; SEE - Semester End Evaluation; MSE- Mid Semester Examination; LWA - Lab Work Assessment; V – Viva voce; CCE-Continuous and Comprehensive Evaluation; ALA- Active Learning Activities.

3 Credits * 25 Marks = 75 Marks (each credit carries 25 Marks) Theory

1 Credits * 25 Marks = 25 Marks (each credit carries 25 Marks) Practical

SEE 100 Marks will be converted in to 50 Marks

CCE 100 Marks will be converted in to 50 Marks

It is compulsory to pass in each individual component.



Course Content:

Sr. No	Course content	Hrs	% Weightage
1	Introduction to Mobile Development & Flutter Basics, Defining Mobile Applications: Native, Hybrid, Cross -Platform, Introduction to Flutter Framework and Dart Language, Flutter architecture, Flutter SDK installation and IDE setup, introduction to pub. dev, first 'Hello World' Flutter App and project structure.	9	20%
2	Dart Programming Essentials Dart Basics: Variables, Data Types, Operators, Control Flow: if, switch, loops, Functions and Parameters, Classes, Objects, Constructors, OOP in Dart: Inheritance, Interfaces, Mixins, Async Programming: Future, async/await, Streams.	10	20%
3	Flutter UI Development Widget tree concept (Stateless vs Stateful Widgets), MaterialApp, Scaffold, Layout Widgets: Row, Column, Stack, Container, Expanded, ListView, GridView, Input Widgets: TextField, Button, Radio, Checkbox, Switch, Slider, Dropdown, Gesture handling: tap, long press, swipe, Styling: Themes, Colors, Fonts, Icons, Images.	10	20%
4	Navigation, State Management, Navigation: push, pop, named routes, Passing data between screens, State Management basics: setState, InheritedWidget, Provider, Menus: Drawer, BottomNavigationBar, PopupMenuButton.	8	20%
5	Data Handling and Deployment Shared Preferences for local storage, Accessing : Camera & Bluetooth & Sensors, Permissions, File storage, Firebase introduction, Building & Releasing Flutter apps.	8	20%

Continuous Assessment:

Sr. No	Active Learning Activities	Marks
1	Real-World Application Analysis: Students will select any Flutter-based mobile application from the Play Store or App Store and conduct an in-depth analysis. The report should cover the app's main features, reasons for choosing Flutter as the development platform, and the advantages observed in terms of performance, UI, and development efficiency. The completed analysis must be uploaded to the GMIU Web Portal.	10
2	UI/UX Prototyping Challenge: Faculty Will Assign students to develop interactive prototypes of mobile app concepts using prototyping tools like Figma, Adobe XD, or Sketch and submit design on GMIU Web Portal.	10



3	UI/UX Design: Students will implement the mobile app prototype created in the UI/UX Prototyping Challenge (ALA 2) using Flutter. They will convert the static design into an interactive and functional mobile interface, ensuring that the implemented UI closely matches the original prototype. This activity aims to help students understand the process of translating design concepts into real app interfaces while learning about Flutter widgets, layout structures, and responsive design principles. The completed Flutter project must be submitted through the GMIU Web Portal.	10
4	Comparative Language Dart vs Java/Kotlin: Students will prepare a detailed comparative report analyzing Dart, JavaScript, Java, and Kotlin programming languages. The report should highlight each language's key features, advantages, limitations, and use cases in mobile app development. Submissions are to be made on the GMIU Web Portal.	10
5	Comparative Platform Study: Students have to will visit companies in group(Maximum 4 students) involved in mobile application development to observe real-world practices and gather insights. They will conduct a comparative study of Flutter, React Native, and Native Android platforms, focusing on what can and cannot be achieved on each in terms of performance, UI capabilities, native integration, and scalability. Based on their observations and research, students will prepare a detailed comparative report and submit it through the GMIU Web Portal.	10
Total		50

Suggested Specification table with Marks (Theory):75

Distribution of Theory Marks (Revised Bloom's Taxonomy)						
Level	Remembrance (R)	Understanding (U)	Application (A)	Analyze (N)	Evaluate (E)	Create (C)
Weightage	30%	25%	20%	10%	5%	10%

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Course Outcome:

After learning the course the students should be able to:	
CO1	Understand mobile development evolution and fundamentals of Flutter.
CO2	Use Dart programming, Flutter architecture, lifecycle, and development tools.



CO3	Design and implement user-friendly UIs using Flutter widgets and layouts.
CO4	Apply navigation, gestures, themes, and multimedia in Flutter applications.
CO5	Implement local/remote data storage and publish a Flutter app.

List of Practical

Sr. No	Descriptions	Unit No	Hrs
1	Installing "Android Studio IDE" and setup "Flutter SDK".	1	2
2	(i) Create First application. That will display "Hello World" in the middle of the screen also change the text color and background color of the text. (ii) Create a Counter App using StatefulWidget & setState().	2	2
3	Write a Dart program demonstrating variables, functions, and OOP concepts (class, inheritance).	2	2
4	Build a Login Form with username/password validation. Show error with Snackbar & navigate on success.	3	4
5	Demonstrate input widgets: TextField, Radio, Checkbox, Switch, Dropdown, Slider.	3	2
6	Design layouts using Row, Column, Stack, ListView, GridView.	3	2
7	Create an app with multiple screens using Navigator and pass data between them.	4	2
8	Implement ImagePicker to pick an image from gallery/camera.	4	2
9	Create an application to demonstrate the use of SharedPreferences	5	2
10	Fetch JSON data from an API using http package and display it in a ListView.	5	4
11	Integrate Firebase Authentication.	5	2
12	Create an application to demonstrate the CRUD operation with SQLite	5	4
		Total	30

Instructional Method:

The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.

From the content 10% topics are suggested for flipped mode instruction.

Students will use supplementary resources such as online videos, NPTEL/SWAYAM videos, e-courses, Virtual Laboratory

The internal evaluation will be done on the basis of Active Learning Assignment

Practical/Viva examination will be conducted at the end of semester for evaluation of performance of students in laboratory.



Reference Books:

- [1] Flutter Complete Reference – 2023 – Alberto Miola – Independently Published.
- [2] Beginning Flutter: A Hands-On Guide to App Development – 2019 – Marco L. Napoli – Wrox Press.
- [3] Flutter Cookbook: 100+ Step-by-Step Recipes for Building Cross-Platform, Professional-Grade Apps with Flutter 3.10.x and Dart 3.x – 2023 – Simone Alessandria – Packt Publishing
- [4] Flutter Apprentice: Learn to Build Cross-Platform Apps – 2022 – Michael Katz, Kevin D. Moore, Vincent Ngo – Kodeco / RayWenderlich
- [5] Flutter and Dart: Up and Running – 2023 – Deepti Chopra & Roopal Khurana – BPB Publications

