Course Syllabus
Gyanmanjari Institute of Technology
Semester-6 (B. Tech)

Subject: Android Development Framework–BETCE16330

Type of course: Professional Elective

Prerequisite: Students should have basic knowledge of mobile application development using Java and Kotlin, including concepts of Android components, object-oriented programming, IDE usage, and fundamental database operations.

Rationale:

The rapid growth of cross-platform technologies has created a strong demand for developers who can build applications for multiple operating systems using a single codebase. This course introduces students to Flutter, a modern and industry-preferred framework for cross-platform mobile application development. By learning Flutter and Dart, students will be able to design, develop, and deploy interactive, responsive, and performance-oriented applications for both Android and iOS. The course builds on the foundation of native Android development (Java and Kotlin) covered in earlier semesters and prepares students with advanced skills that are aligned with current industry practices and future career opportunities.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks					Total Marks
				Theory Marks		Practical Marks		CA	
CI	T	P	C	ESE	MSE	V	P	ALA	
3	0	2	4	60	30	10	20	30	150

Legends: CI-ClassRoom Instructions; T – Tutorial; P - Practical; C – Credit; ESE - End Semester Examination; MSE- Mid Semester Examination; V – Viva; CA - Continuous Assessment; ALA- Active Learning Activities.

Course Content:

Sr. No	Course Content	Hrs.	% Weightage
1	**Introduction to Flutter & Dart Basics:** Overview of Flutter framework, advantages of cross-platform development, Flutter architecture, Widgets overview. Setting up Flutter SDK, IDE (Android Studio/VS Code), and Emulator/Device configuration. Dart basics – variables, data types, functions, classes, and collections.	09	20%
2	**Flutter UI Design with Widgets:** Understanding Widget tree, Stateless vs Stateful widgets, Layout widgets (Row, Column, Container, Stack), Input widgets (TextField, Buttons, Checkbox, Switch). Styling widgets, themes, and responsive design for multiple devices.	09	20%
3	**State Management and Navigation:** Concept of state, setState, InheritedWidget, Provider basics. Navigation and routes, passing data between screens, Bottom Navigation, Drawer. Form handling and validation.	09	20%
4	**Data Handling and Integration:** Working with JSON, APIs, and RESTful services. Parsing data, displaying lists with ListView & GridView. Local data storage using SharedPreferences, SQLite, and Hive. Introduction to Firebase integration.	09	20%
5	**Advanced Features & Deployment:** Using animations, custom widgets, gesture detection. Accessing device features (camera, location, sensors) , integrating third-party packages (pub.dev). App testing, debugging, and publishing Flutter app to Play Store & App Store. Basics of version control, GitHub repository management, and collaborative software development.	09	20%

Continuous Assessment:

Sr. No	Active Learning Activities	Marks
1	**Widget Tree Modeling:** Students will study Flutter architecture and Dart language fundamentals and prepare a widget tree diagram of a sample application using draw.io and export it in PDF format. This ALA is to be completed individually. Students will upload the PDF file on the GMIU Web Portal.	10
2	**RESTful API Integration & Repository:** Students will integrate a public REST API into a Flutter application, parse the JSON data, and display it using ListView or GridView. The complete project code must be uploaded on GitHub, and the output description should be included in the README file or GitHub Pages section. This ALA is to be completed individually. Students will upload only the GitHub repository link on the GMIU Web Portal.	10
3	**Persistent Data Management with Documentation:** Students will develop a Flutter application with local data storage using SQLite, Hive, or SharedPreferences, demonstrating storing, retrieving, and updating data. They must upload the project code on GitHub along with proper documentation/manual in PDF format explaining the project workflow. This ALA is to be completed individually. Students will submit the GitHub repository link and PDF documentation on the GMIU Web Portal.	10
	Total	30

Suggested Specification table with Marks (Theory):60

Distribution of Theory Marks (Revised Bloom's Taxonomy)						
Level	**Remembrance (R)**	**Understanding (U)**	**Application (A)**	**Analyze (N)**	**Evaluate (E)**	**Create (C)**
Weightage %	25%	25%	20%	10%	10%	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	Apply Flutter fundamentals and Dart concepts to build cross-platform apps.
CO2	Design user interfaces using Flutter widgets, layouts, themes, and responsive principles.
CO3	Manage state and navigation using setState, Provider, and route management.
CO4	Integrate APIs, JSON data, and local storage (SQLite, Hive, SharedPreferences) for data-driven apps.
CO5	Apply advanced Flutter features (animations, custom widgets, device services) and deploy fully tested applications on Play Store/App Store.

List of Practical

Sr. No	Description	Unit No	Hrs.
1	Set up Flutter SDK, configure IDE (VS Code/Android Studio), and run a sample Flutter project.	01	02
2	Develop a "Hello Flutter" app to understand widget tree structure and Dart basics.	01	02
3	Create UI layouts using basic widgets (Text, Image, Button, Row, Column, Container).	02	02
4	Design interactive forms using TextField, Radio, Checkbox, and Dropdown with validation.	02	02
5	Implement navigation between multiple screens and pass data using routes.	03	02
6	Demonstrate state management using setState and Provider.	03	02
7	Fetch and parse JSON data from a public REST API and display it using ListView.	04	02
8	Store and retrieve data using SharedPreferences for key-value pairs.	04	02
9	Implement local database storage using SQLite or Hive for CRUD operations.	04	04
10	Create animations and custom widgets for interactive UI.	05	02
11	Access device features such as camera and location in a Flutter app.	05	02
12	Implement push notifications using Firebase Cloud Messaging.	05	02
13	Debug a Flutter application using Flutter DevTools and fix runtime issues.	05	02

14	Generate a signed APK/IPA and deploy the Flutter app to Play Store/App Store.	05	02
		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] Alessandro Biessek – Flutter for Beginners: An Introductory Guide to Building Cross-Platform Mobile Applications with Flutter and Dart 2 – Packt Publishing, 2nd Edition, 2020.
[2] Marco L. Napoli – Beginning Flutter: A Hands-On Guide to App Development – Wiley, 2019.
[3] Carmine Zaccagnino – Programming Flutter: Native, Cross-Platform Apps the Easy Way – Pragmatic Bookshelf, 2020.
[4] Thomas Bailey – Flutter Cookbook: Over 100 Proven Techniques and Solutions for App Development with Flutter 2.2 and Dart – Packt Publishing, 2021.
[5] Fu Cheng – Flutter Recipes: Mobile Development Solutions for iOS and Android – Apress, 2019.