Semester: VII				
Programme: B.Sc. Computer Science (Hons)				
Course: FULL STACK WEB DEVELOPMENT & ANDROID PROGRAMMING				
Paper code: C4CS230741P			**Credits: 6**	
Hours/week: 6				
Category: Core/MDC/SEC/VAC: Core				
Theory / Practical / Composite: Practical				
No of Modules: 2				
Course Overview: A Full Stack and Android development course combines web (front-end/back-end) and mobile technologies to create end-to-end applications. Key areas include UI design (HTML/CSS/React or Android XML), server-side programming (Node.js/Java/Python), database management (SQL/MongoDB), and Android-specific development using Kotlin/Java. The course focuses on building, testing, and deploying mobile apps and web services				
Course Outcome:				
	FULL STACK DEVELOPMENT			
1.	**Recall** fundamental concepts of web development including HTML, CSS, JavaScript, client-server architecture, and HTTP protocols.			
2.	**Interpret** frontend structures, responsive design principles, React components, and API communication workflows.			
3.	**Develop and implement** dynamic web applications using React, Node.js, Express, and perform CRUD operations with databases.			
4.	**Analyze** application architecture, data flow, API interactions, and database integration in full stack systems.			
5.	**Evaluate** security mechanisms (JWT, authentication), performance optimization, and deployment strategies in full stack applications.			
6.	**Design and develop** a complete full stack application integrating frontend, backend, database, and deployment tools.			
	ANDROID DEVELOPMENT			
1.	**Recall** Android architecture components, Kotlin syntax, activity lifecycle stages, and UI elements.			
2.	**Interpret** Android project structure, UI layouts, intents, and data storage mechanisms.			
3.	**Implement** Android applications using Kotlin, UI components, handle user interactions, and integrate REST APIs using Retrofit.			
4.	**Analyze** activity transitions, data flow between components, and compare different storage techniques.			
5.	**Evaluate** design decisions related to UI/UX (Material Design), API integration strategies, and performance considerations in Android applications.			
6.	**Design** a complete Android application integrated with a custom backend API, implementing authentication, data persistence, and real-time data handling.			
Prerequisites:				
SYLLABUS				
UNIT/Module	**CONTENT**	**HOURS or NUMBER OF CLASSES**	**CO Mapping**	**COGNITIVE LEVEL**

FULL STACK DEVELOPMENT				
I.	Internet basics: Client-server model, HTTP/HTTPS; HTML5 & CSS3 refresher (forms, media queries, responsive design); JavaScript basics: Variables, operators, control structures; DOM manipulation & events; Functions, ES6+ features; Fetch API and JSON basics	**5**	**CO1, CO2**	**K1, K2 (Remember/Understand)**
II.	Async programming: Promises, async/await; Error handling & debugging basics; NPM basics (installing and using packages); Introduction to Git & GitHub (version control, pushing projects)	**5**	**CO2, CO3, CO4**	**K2, K3 (Understand/Apply/ Analyse)**
III.	React basics: Components, JSX, props, state; Event handling & forms; React hooks (useState, useEffect)	**8**	**CO3, CO4, CO6**	**K2, K3, K6(Understand / Apply/ Create)**
IV.	Node.js basics & architecture; Modules & NPM; Express.js: Routing, middleware; RESTful API creation (CRUD)	**8**	**CO3, CO4, CO6**	**K2, K3, K6 (Understand / Apply/ Create)**
V.	Introduction: Relational DB (MySQL) vs NoSQL (MongoDB); **MongoDB Basics:** Collections, Documents, BSON format; CRUD operations: Insert, Find, Update, Delete; Using MongoDB Compass and Mongo Shell basics; Connecting Node.js with MongoDB	**5**	**CO5, CO6**	**K4, K5, K6**
VI.	Basic user authentication (login/signup with sessions/JWT); Secure password storage (bcrypt); Hosting basics (Heroku/Render/Vercel)	**8**	**CO4, CO5, CO6**	**K3, K4, K6 (Apply/Analyze/Create)**
ANDROID DEVELOPMENT				
VII.	Android & Kotlin Basics: Android Studio setup; Kotlin basics: Variables, functions, classes; Android project	**4**	**CO1**	**K1, K2 (Remember/Understand)**

	structure (manifest, resources, Gradle).			
VIII.	UI Design in Android: Views, layouts (Linear, Relative, ConstraintLayout); Common widgets: TextView, EditText, Button, ImageView; RecyclerView & Adapters (list-based UI); Material Design guidelines (basic introduction).	**7**	**CO2, CO3**	**K2, K3 (Understand/Apply)**
IX.	Activities & Intents: Activity lifecycle; Explicit vs Implicit Intents; Passing data between activities; Toasts, Snackbar, Dialogs.	**8**	**CO2, CO3, CO4**	**K3, K4 (Apply/Analyze)**
X.	Data Storage: SharedPreferences (simple data storage), SQLite database (structured storage), Room persistence library (ORM).	**8**	**CO3, CO4**	**K3, K4 (Apply/Analyze)**
XI.	Networking & APIs: REST API consumption with Retrofit, JSON parsing Asynchronous tasks in Android, Firebase basics.	**8**	**CO3, CO4, CO5**	**K3, K4, K5 (Apply/Analyze/Evaluate)**
XII.	Full Stack + Android Integration: Connecting Android app with custom backend API (built in Part A), Authentication (JWT with Android); CRUD operations through API calls.	**4**	**CO4, CO5, CO6**	**K4, K5, K6 (Analyze/Evaluate/Create)**

Text Books
1. The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer by Chris Northwood
2. Head First Android Development - David Griffiths and Dawn Griffiths, O'REILLY
Web Resources
1. Full Stack Course on Coursera
2. Android course on Udemy

Evaluation	**Practical** CA: 95 Attendance: 5

Course outcomes (COs) and Cognitive Level Mapping

FULL STACK DEVELOPMENT

COs	CO Description	Cognitive levels
CO1	**Recall** fundamental concepts of web development including HTML, CSS, JavaScript, client-server architecture, and HTTP protocols.	K1 (Remember)
CO2	**Interpret** frontend structures, responsive design principles, React components, and API communication workflows.	K2 (Understand)
CO3	**Develop and implement** dynamic web applications using React, Node.js, Express, and perform CRUD operations with databases.	K3 (Apply)
CO4	**Analyze** application architecture, data flow, API interactions, and database integration in full stack systems.	K4 (Analyze)
CO5	**Evaluate** security mechanisms (JWT, authentication), performance optimization, and deployment strategies in full stack applications.	K5 (Evaluate)
CO6	**Design and develop** a complete full stack application integrating frontend, backend, database, and deployment tools.	K6 (Create)

ANDROID

COs	CO Description	Cognitive levels
CO1	**Recall** Android architecture components, Kotlin syntax, activity lifecycle stages, and UI elements.	K1–K2 (Remember / Understand)
CO2	**Interpret** Android project structure, UI layouts, intents, and data storage mechanisms.	K2 (Understand)
CO3	**Implement** Android applications using Kotlin, UI components, handle user interactions, and integrate REST APIs using Retrofit.	K3 (Apply)
CO4	**Analyze** activity transitions, data flow between components, and compare different storage techniques.	K4 (Analyze)
CO5	**Evaluate** design decisions related to UI/UX (Material Design), API integration strategies, and performance considerations in Android applications.	K5 (Evaluate)
CO6	**Design** a complete Android application integrated with a custom backend API, implementing authentication, data persistence, and real-time data handling.	K6 (Create)