Uka Tarsadia University

B. Tech.

CSE / CSE (AI&ML) /CSE (CC) / CSE (CS) /CE /CE (SE) /IT

Semester IV

MOBILE APPLICATION DEVELOPMENT USING iOS

IT4039

Effective from July-2024

Syllabus version: 1.0

Subject Code	Subject Title
IT4039	Mobile Application Development using iOS

Teaching Scheme				Examination Scheme			
Hours		Credits		Theory Marks		Practical Marks	Total Marks
Theory	Practical	Theory	Practical	Internal	External	CIE	
3	2	3	1	40	60	50	150

Objectives of the course:

- Develop and integrate user interfaces using SwiftUI, including advanced components, navigation strategies, and data

Course outcomes:

Upon completion of the course, the student shall be able to,

CO1: Describe and apply fundamental SwiftUI views and interactive elements for building user interfaces.

CO2: Design and implement dynamic and advanced UI components in SwiftUI applications.

CO3: Analyze and utilize navigation strategies for managing data-driven user interactions.

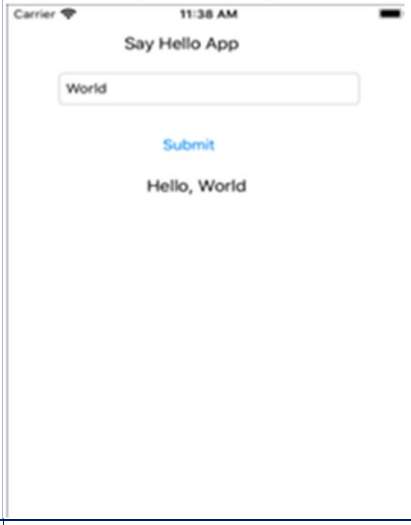
CO4: Integrate SwiftUI with existing frameworks and legacy applications to enhance functionality.

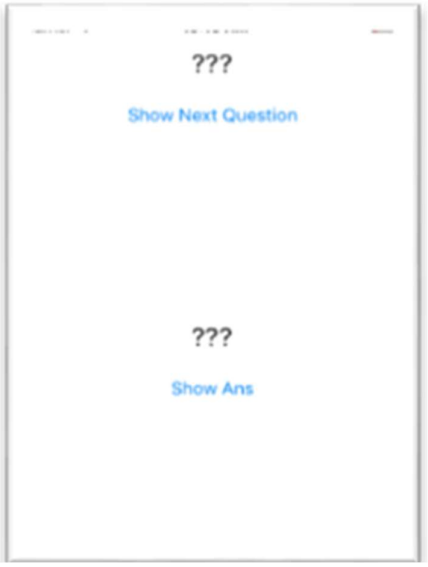
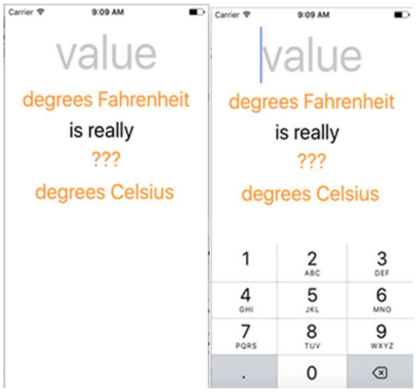
CO5: Manage and synchronize application state across different views and components.

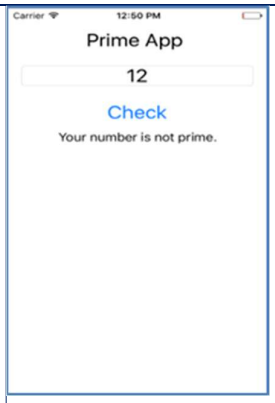
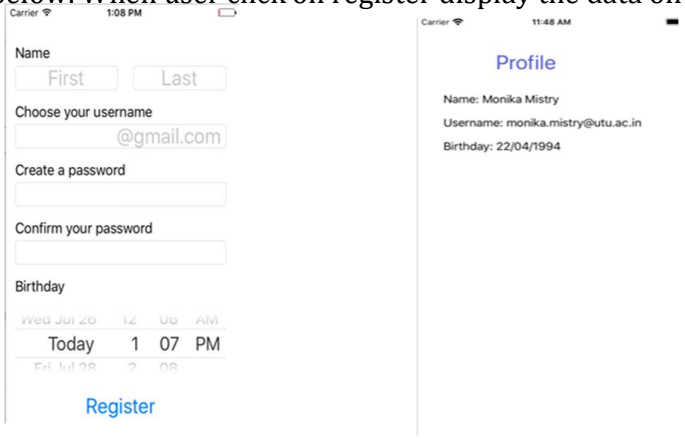
CO6: Implement and manage data persistence and integration solutions in SwiftUI applications.

Sr. No.	Topics	Hours
	Unit – I	
1	**Basic SwiftUI Views and Controls:** iOS Applications, Swift Programming Basics, Laying Out Components, Text, Image, Button, Segmented Control, TextField, Picker, View Modifier, View Builder, Symbol, Integrating UIKit into SwiftUI, Adding SwiftUI to Legacy UIKit App.	06
	Unit – II	
2	**List Views, Scroll Views, and Advanced Components** Scroll Views, Creating List of Static Items, Custom Rows in a List, Adding/Deleting Rows, Editable List View, Editable Collection, Searchable List, LazyHStack and LazyVStack, LazyHGrid and LazyVGrid, Scrolling Programmatically, Expanding Lists, Disclosure Groups, Create SwiftUI Widgets	08

Unit – III		
3	**Navigation:** Using NavigationStack, Typed Data-driven Navigation with NavigationStack, Untyped Data-driven Navigation with NavigationStack, Working with NavigationSplitView, Using TabView, Programmatically switching tabs on a TabView.	08
Unit – IV		
4	**SwiftUI with Data:** Using @State to drive a View's behaviour, Using @Binding to pass a state variable to child views, Core location wrapper @ObservedObject, Preserve model's life cycle using @StateObject, Sharing state object using @EnvironmentObject, Using Observation to manage model data.	08
Unit – V		
5	**Combine and Firebase:** Introduction to Combine, Managing Memory and Validating a Form, Fetching Remote Data and Visualization, Sign in with Apple, Integrating Firebase into SwiftUI, Google Sign in.	08
Unit – VI		
6	**Core Data and Swift Data:** Integrating Core Data with SwiftUI, Showing Core Data Objects with @FetchRequest, Adding Core Data Objects from a SwiftUI View, Filtering Core Data Objects with Predicate, Deleting Core Data Objects, @SectionedFetchRequest, Working with SwiftData.	07

Sr. No.	Mobile Application Development using iOS (Practicals)	Hours
1.	Introduction to iOS and Xcode.	2
2.	Case study of Swift programming language Running Swift Code to print Hello World.	4
3	a) Write a Program to understand Swift while and repeat while Loop by a program to display numbers from 1 to 5. b) Write a Program to understand Nested Loops in Swift by a program to display 7 days of 2weeks.	2
4	Write a program in swift to: a) Check number is even or odd. b) Print prime numbers in given range c) Check the string is palindrome or not.	4
5	a) To print Fibonacci Series in Swift Language. b) To print factorial of a given number in Swift language.	2
6	Create an iOS application to develop "Say Hello App". Use TextField to get user name as input. On tap of button, display user name with Hello in Label 	2
7	Create an iOS Application for Quiz. Create following layout given below and performed following functionality: a) Question and Answers Load from Data Source (Data Source contains String Array). b) Contains two buttons (Show Next Question) and two labels (Show Answer). c) Display next question on tap of Show Next Question button. d) Display answer on tap of Show Answer button.	2

		
8	Create an iOS application for Fahrenheit to Celsius conversion. Create layout as given below. If Text changing in FahrenheitTextField, automatically convert temperature in Celsius and display value change in CelsiusLabel. Note: Layout should be auto (automatically adjust to universal devices). Celsius value should be generated on text changed event. Keyboard should be automatically dismissing on tap of screen. 	4
9	Create an iOS Application demonstrate use of Tab Bar Control in your application. Application contains following functionality: • Two tabs called Map and Prime Number. • On click of Map tag Map Should be open. • On Click of Prime Number, to Check Inputted Number is prime or not. • Map View should be created using programmatically.	4

		
10	Create an iOS Application interface that look like registration view as given below. When user click on register display the data on second view. 	4

Text book:

1. Juan C. Catlan, "SwiftUI Cookbook", Packt Publishing Ltd.

Reference books:

2. Wallace Wang, "Beginning iPhone Development with SwiftUI", Apress.
3. Wallace Wang, "Pro iPhone Development with SwiftUI", Apress.
4. Mukesh Sharma, "iOS Development with SwiftUI", BPB.

Course objectives and course outcomes mapping:

- Develop and integrate user interfaces using SwiftUI, including advanced components: CO1, CO2
- Navigation strategies: CO3
- Data management techniques for creating robust applications: CO4, CO5 and CO6.

Course units and Course outcome mapping:

Unit No.	Unit Name	Course Outcomes					
		CO1	CO2	CO3	CO4	CO5	CO6
1	Basic SwiftUI Views and Controls	✓					
2	List Views, Scroll Views, and Advanced Components		✓				
3	Navigation			✓			

4	The BOM and DOM				✓		
5	Combine and Firebase					✓	
6	Core Data and Swift Data						✓

Programme outcomes:

PO 1: Engineering knowledge: An ability to apply knowledge of mathematics, science, and engineering.

PO 2: Problem analysis: An ability to identify, formulates, and solves engineering problems.

PO 3: Design/development of solutions: An ability to design a system, component, or process to meet desired needs within realistic constraints.

PO 4: Conduct investigations of complex problems: An ability to use the techniques, skills, and modern engineering tools necessary for solving engineering problems.

PO 5: Modern tool usage: The broad education and understanding of new engineering techniques necessary to solve engineering problems.

PO 6: The engineer and society: Achieve professional success with an understanding and appreciation of ethical behavior, social responsibility, and diversity, both as individuals and in team environments.

PO 7: Environment and sustainability: Articulate a comprehensive world view that integrates diverse approaches to sustainability.

PO 8: Ethics: Identify and demonstrate knowledge of ethical values in non-classroom activities, such as service learning, internships, and field work.

PO 9: Individual and team work: An ability to function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give/receive clear instructions.

PO 11: Project management and finance: An ability to demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO 12: Life-long learning: A recognition of the need for, and an ability to engage in life-long learning.

Programme outcomes and Course outcomes mapping:

Programme Outcomes	Course Outcomes					
	CO1	CO2	CO3	CO4	CO5	CO6
PO1		✓	✓	✓	✓	
PO2			✓	✓	✓	
PO3		✓		✓	✓	✓
PO4						
PO5			✓	✓	✓	✓
PO6						
PO7						
PO8						
PO9						
PO10						
PO11				✓		
PO12						