SAN DIEGO GLOBAL KNOWLEDGE UNIVERSITY

Mobile Development Immersive 1

Course Name	Upon completion of this program, the student will be able to:	PLO 1: Apply fundamental and intermediate programming principles to develop and optimize applications using Swift, SwiftUI, and version control techniques.	PLO 2: Implement intuitive and efficient mobile application interfaces using SwiftUI.	PLO 3: Apply programming languages and tools for iOS to create mobile applications.	PLO 4: Demonstrate the use of testing and debugging in mobile applications to ensure functionality, reliability, and performance.	PLO 5: Implement mobile applications on app stores while managing updates effectively.
MDI1 101: Introduction to iOS Development						
Describe the iOS ecosystem and its components, including iOS devices, the App Store, and the operating system's role in mobile development.	X = See key below	RA, WE, RQ andFE				
	IRM	I				
Demonstrate key features of the Xcode IDE, demonstrating the ability to use Interface Builder for basic user interface design.	X = See key below	P, RQ, WE and FE	P, WE and FE			
	IRM	I	I			
MDI1 102: Programming Fundamentals - Introductory						
Apply basic programming concepts, including programming paradigms, variables, data types, control structures, and functions, using Swift	X = See key below	P, RQ, WE and FE				
	IRM	I				
Identify object-oriented programming principles, including classes and objects in Swift.	X = See key below	P, RA, WE, FE				
	IRM	I				
Explain the fundamental concepts of declarative programming and its application in developing user interfaces using SwiftUI.	X = See key below	P, FE, and WE	P, RQ, WE, and FE			
	IRM	I	I			
MDI1 103: Programming Fundamentals - Intermediate						
Use data structures such as arrays and dictionaries to store and access collections of data.	X = See key below	P, WE and FE				
	IRM	R				
Explain the concept of inheritance in object-oriented programming and identify how it supports code reuse.	X = See key below	P, RQ, WE and FE				
	IRM	R				
Identify declarative programming principles to build more complex user interfaces using SwiftUI.	X = See key below	P, RQ, WE and FE	P, FE, and RQ			
	IRM	R	R			
MDI1 104: Principles of UI/UX for iOS						
Apply Apple’s human interface guidelines to design consistent and intuitive user interfaces for iOS applications.	X = See key below	P, WE, RA, and FE	P, WE, and FE	P and FE		
	IRM	I	I	I		
Explain the importance of usability and accessibility in iOS app design and implement best practices in wireframes and prototypes.	X = See key below	P,WE, RQ, and FE	P, WE, RQ, and FE	P, WE, and FE		
	IRM	I	I	I		
Identify appropriate design elements such as typography, color schemes, and iconography to maintain visual consistency in iOS interfaces.	X = See key below	P, WE, RQ and FE	P, WE, RQ and FE	P, WE, and FE		
	IRM	I	I	I		
MDI1 105: Implementation of UI/UX for iOS						
Apply UI/UX design principles using SwiftUI to create visually consistent and interactive iOS applications that adhere to Apple’s Human Interface Guidelines.	X = See key below	P, RA, WE, and FE	P, RQ, WE, and FE	P, WE, and FE		
	IRM	I	I	I		
Use interactions and gestures in SwiftUI, such as buttons, sliders, and navigation, ensuring a seamless user experience.	X = See key below	P, RQ, WE and FE	P, RQ, WE and FE	P, WE, and FE		
	IRM	I	I	I		
MDI1 106: Implementation of Advanced UI/UX for iOS						
Apply advanced animations and transitions in SwiftUI to create smooth, responsive interactions that enhance user experience.	X = See key below	P, WE, and FE	P, WE, and FE	P, and WE and FE		
	IRM	R	R	R		
Identify performance issues related to advanced UI components and implement solutions to ensure smooth animations and layouts.	X = See key below	P, WE, RQ and FE	P, WE, RQ and FE	P, WE, and FE		
	IRM	R	R	R		
Apply best practices for handling large data sets and dynamic content while maintaining UI responsiveness in complex SwiftUI layouts.	X = See key below	P, WE, RQ and FE	P, WE, RQ and FE	P, and WE and FE		
	IRM	R	R	R		
MDI1 107: Testing and Debugging for iOS						
Apply unit and UI testing methodologies in Xcode to ensure that SwiftUI applications function correctly and meet design requirements.	X = See key below		P, WE and FE		P, WE, RQ, and FE	
	IRM		I		I	
Identify common issues in SwiftUI applications using Xcode’s debugging tools, such as breakpoints, console logs, and runtime analysis.	X = See key below		P, WE, RQ andFE		P, WE, and FE	
	IRM		I		I	

Apply UI tests in Xcode to verify user interactions, navigation, and interface elements in SwiftUI applications.	X = See key below		P, WE, RQ andFE		P, WE, RQ, FE	
	IRM		I		I	
MDI1 108: Data Persistence for iOS						
Apply UserDefaults in SwiftUI to persist user settings and small data across app sessions.	X = See key below	P, RA, WE, and FE	P, RA, WE, and FE	P, RA, WE, and FE		
	IRM	I	I	I		
Use Core Data for managing complex data models and performing CRUD operations.	X = See key below	P, FE, RQ, and WE		P and FE		
	IRM	I		I		
Apply storing files locally in SwiftUI apps for handling larger data needs.	X = See key below	P, RQ, WE and FE	P, RQ, WE and FE	P, WE and FE		
	IRM	I	I	I		
MDI1 109: Networking and Web Services for iOS						
Apply URL sessions in Swift to effectively communicate with web services and retrieve remote data.	X = See key below	P, WE, and FE		P, WE, and FE		
	IRM	I		I		
Demonstrate the use of parse and structure JSON data from web APIs using Swift's Codable protocol.	X = See key below	P, RQ, WE, and FE		P, WE, and FE		
	IRM	I		I		
Apply async/await in Swift to handle asynchronous network operations smoothly.	X = See key below	P, RQ, WE, and FE		P, WE, and FE		
	IRM	I		I		
MDI1 110: Mobile Application Device Sensors - Introductory						
Apply Core Motion to capture and respond to accelerometer and gyroscope data, creating motion-sensitive features.	X = See key below	P, RQ, WE and FE		P, FE, and WE		
	IRM	I		I		
Use data from health-related sensors and camera and photo library, such as heart rate monitors, while ensuring user data privacy.	X = See key below	P, RA, RQ, WE, and FE		P, FE, and WE		
	IRM	I		I		
MDI1 111: Mobile Application Device Sensors - Intermediate						
Use Core Motion to track user movements and device orientation accurately within iOS applications.	X = See key below	P, WE, RQ and FE		P, WE, RQ, and FE		
	IRM	R		R		
Apply geofencing and real-time location tracking through Core Location to create location-aware applications.	X = See key below	P, WE, RA, RQ, and FE		P, FE, and WE		
	IRM	R		R		
Use HealthKit for monitoring and displaying health data within iOS applications, maintaining privacy and data integrity.	X = See key below	P, WE, and FE		P, FE, and WE		
	IRM	R		R		
MDI1 112: Mobile Applications for Wearables - Introductory						
Apply responsive design and user-friendly interfaces optimized for smaller screens on wearable devices, emphasizing simplicity and usability.	X = See key below	P, RA, WE and FE	P, RA, WE and FE	P, WE, FE	P, RA, WE and FE	
	IRM	I	I	I	I	
Use real-time heart rate data from wearable devices using HealthKit.	X = See key below	P, RQ, WE and FE		P, WE, FE		
	IRM	I		I		
Identify motion sensors to track user activities, providing real-time feedback in fitness and activity-tracking applications.	X = See key below	P, RQ, WE and FE		P, WE, FE	P, RA, WE and FE	
	IRM	I		I	I	
MDI1 113: Mobile Applications for Wearables - Intermediate						
Apply advanced movement tracking features using Core Motion on wearable devices.	X = See key below	P,WE, RA, and FE		P,WE, and FE	P,WE, RA, and FE	
	IRM	R		R	R	
Use wearable device functionalities for capturing and managing images and media.	X = See key below	P, WE, and FE		P, WE, and FE	P, WE, and FE	
	IRM	R		R	R	
Apply monitoring real-time health data using HealthKit, ensuring responsiveness in real-time applications.	X = See key below	P, RQ, WE and FE		P, WE, and FE	P, RQ, WE and FE	
	IRM	R		R	R	
MDI1 114: IPadOS Development						
Apply Apple pencil support to create interactive features within iPad applications.	X = See key below	P, WE, RQ and FE		P, FE, and WE	P, WE, FE	
	IRM	R		R	R	
Apply multitask capabilities in iPadOS applications by implementing split view, slide over, and multi-windows support that enhance user experience.	X = See key below	P, FE, RA, and WE	P, WE, RQ and FE	P, FE, and WE	P, RQ, WE and FE	
	IRM	R	R	R	R	
MDI1 115: Localization in iOS Application						
Apply localization and internationalization features in iOS apps to support multiple languages and regions.	X = See key below	P, RQ, WE, and FE	P, RQ, WE and FE	P, FE and WE	P, FE and WE	
	IRM	R	R	R	R	

Apply best practices for managing localized strings, images, and resources in iOS development.	X = See key below	P, FE, RA, and WE	P, RA, WE and FE	P, FE and WE	P, FE and WE	
	IRM	R	R	R	R	
Use iOS frameworks to handle different date, time, and number formats across various locales.	X = See key below	P, FE and WE	P, RQ, WE, and FE	P, FE and WE	P, FE and WE	
	IRM	R	R	R	R	
MDI1 116: Deployment and Apple Store Submission						
Apply the process of preparing iOS apps for release, including creating distribution certificates and signing apps.	X = See key below	P, RA, RQ, WE, and FE		P, FE and WE	P, RA, RQ, WE, and FE	P, RA, RQ, WE, and FE
	IRM	I		R	R	I
Use App Store guidelines and policies to ensure compliance during the app submission process.	X = See key below	P, RQ, WE, and FE		P, FE and WE	P, RQ, WE, and FE	P, RQ, WE, and FE
	IRM	I		R	R	I
MDI1 117: iOS Automated Testing						
Apply automated unit, integration, and UI tests using XCTest and XCUITest frameworks.	X = See key below	P, FE, and WE		P, FE and WE	P, FE and WE	P, FE and WE
	IRM	R		R	R	R
Apply test-driven development (TDD) techniques to improve the reliability and maintainability of iOS apps.	X = See key below	P, RQ WE and FE		P, RQ, WE and FE	P, RQ, WE and FE	
	IRM	R		R	R	
Use continuous integration (CI) tools to automate the testing process and ensure consistent code quality.	X = See key below	P, RA, WE and FE		P, RA, WE and FE	P, RA, WE and FE	
	IRM	R		R	R	
MDI1 118: Capstone - Planning and Design						
Identify the objectives and scope of the capstone project.	X = See key below	FE, RA, WE, and P	FE, RA, WE, and P	P, FE, and WE	P, FE, and WE	
	IRM	R	R	R	R	
Describe the target audience and user needs for the application.	X = See key below	FE, RQ, WE, and P	FE, RQ, WE, and P	P, FE, and WE		P, RA, RQ, WE, and FE
	IRM	R	R	R		R
Explain the functional and technical requirements of the proposed Application create wireframes and prototypes.	X = See key below	FE, RQ, WE, and P	FE, RQ, WE, and P	P, FE, and WE	P, FE, and WE	
	IRM	R	R	R	R	
MDI1 119: Capstone - Development Phase 1						
Use Swift programming concepts to develop core application features.	X = See key below	P, RA, RQ, WE, and FE	P, RA, RQ, WE, and FE	P, RA, RQ, WE, and FE		
	IRM	R	R	R		
Demonstrate how to integrate user interface designs into the application using SwiftUI.	X = See key below	P, RA, RQ, WE, and FE	P, RA, RQ, WE, and FE	P, FE, and WE	P, FE, and WE	
	IRM	R	R	R	R	
Implement data persistence methods suitable for the application.	X = See key below	P, RA, RQ, WE, and FE	P, RA, RQ, WE, and FE	P, FE, and WE	P, FE, and WE	P, FE, and WE
	IRM	R	R	R	R	R
MDI1 120: Capstone - Development Phase 2						
Use advanced Swift programming techniques to enhance application features.	X = See key below	P, FE, and WE	P, FE, and WE	P, FE, and WE	P, FE, and WE	P, FE, and WE
	IRM	M	M	M	M	M
Apply methods for optimizing the application’s performance and efficiency.	X = See key below	FE, RA, WE, and P	FE, RQ, WE, and P	P, FE, and WE	FE, RA, WE, and P	FE, RQ, WE, and P
	IRM	M	M	M	M	M
Resolve bugs through comprehensive testing and debugging.	X = See key below	FE, RQ, WE, and P	FE, RQ, WE, and P	P, FE, and WE	FE, RQ, WE, and P	FE, RQ, WE, and P
	IRM	M	M	M	M	M

X = The assessment which measures the stated program objective/outcome.

KEY	
Projects	P
Researched and Critiqued Articles	RA
Review / Discussion Question Responses	RQ
Case Studies	CS
Web Exercises	WE

Final Exam with Essay Questions	FE
Introduced, Reinforced, Mastered	IRM