

**36. iOS Programming Fundamentals (PH055IU)**

Module designation	<i>This course provides students with an introduction to programming on the iOS platform with Swift Programming language including: environment, syntax, data types, variables, tuples, constants, literals, operators, decision making, loops, strings, arrays, sets, functions, classes, properties, methods, OOP concepts, App development methodologies, UI designs.</i>
Semester(s) in which the module is taught	1, 2
Person responsible for the module	MSc. Trương Thị Ngọc Phượng
Language	English
Relation to curriculum	Compulsory
Teaching methods	Lecture, project, practice
Workload (incl. contact hours, self-study hours)	(Estimated) Total workload: 182.5 Contact hours (please specify whether lecture, exercise, laboratory session, etc.): lecture: 37.5, laboratory: 25 Private study including examination preparation, specified in hours: 120
Credit points/ECTS	4 credits (3theory + 1 practice)/6.62 ECTS
Required and recommended prerequisites for joining the module	Previous course: Programming for Engineers (EE057IU)



Module objectives/intended learning outcomes	Upon the successful completion of this course students will be able to:	
	Competency level	Course learning outcome (CLO)
	Knowledge	CLO1. Apply the knowledge of informatics to solve engineering problems. CLO2. Apply the Swift Language to develop iOS applications.
	Skill	CLO3. Implement programs on iOS using the Swift language and app development tools.
Attitude	CLO4. Recognize the legality, professional ethics and responsibilities, and norms of developing and using the software.	



Content	<p><i>The description of the contents should clearly indicate the weighting of the content and the level.</i></p> <p>Weight: lecture session (5 hours)</p> <p>Teaching levels: I (Introduce); T (Teach); U (Utilize)</p>		
	Topic	Weight	Level
	Introduction to Swift Language Architecture of Swift Functions Variable and Simple Types Object Type Flow Control and More	3	I, T
	OOP Concepts & Practices Objects, Properties, Classes, methods. Constructor. Inheritance Polymorphism Abstraction Encapsulation.	3	T
	Xcode Project Anatomy of an Xcode Project Nib Management Documentation Life Cycle of a Project MVC Concepts	2	T, U
	Build the UI UIKit and Interface Builder Build a basic UI Connect the UI to code. Working with View Controllers. Implement custom controls. Define your data model	2	T, U
	Working with Multiple View Controllers and Navigation. TableView Navigation Controller.	2	T, U
	Working with Core Data. Core Data Entities and Attributes. Data saving Data fetching Data deleting.	1	T, U
	Working with Networking Networking services GET request REST & CRUD Decoding, Async, and POST Request Test and publish apps on App Store	2	T, U



Examination forms	Project
Study and examination requirements	<p>Attendance: A minimum attendance of 80 percent is compulsory for the class sessions. Students will be assessed on the basis of their class participation. Questions and comments are strongly encouraged.</p> <p>Assignments/Examination: Students must have more than 50/100 points overall to pass this course.</p>
Reading list	<p>Textbook:</p> <p>[1] Neuburg, Matt, <i>iOS 10 programming fundamentals with Swift : Swift, Xcode, and Cocoa basics</i>, Beijing : O'Reilly, 2017.</p> <p>Reference:</p> <p>[2] Greg Lim, <i>Beginning iOS 13 & Swift App Development: Develop iOS Apps with Xcode 11, Swift 5, Core ML, ARKit and more</i>, independently published</p> <p>[3] <i>Beginning Android</i>, 5th edition, Grant Allen</p> <p>[4] <i>Learning Android Google Maps</i>, Raj Amal W.</p>