**COURSE TITLE : MOBILE PROGRAMMING**

**COURSE CODE : IT 227**

**CREDIT HOURS : 60**

**CREDIT UNITS : 4**

**LEVEL OF COURSE: YEAR II SEMESTER II**

**Course Description:**

This course involves a careful examination of mobile device programming. Emphases are on developing applications as a community that run on the Android platform. Students planning to enroll in this course should have background in computing and be familiar with Java. This course will also give students insight to today's common procedures for getting their mobile application work academically published.

The course introduces mobile application development for the Android platform. Android is a software stack for mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language.

**Course Objectives:**

By the end of the course, students should:

1. Be competent in programming in an event-based model used in application development for mobile devices
2. Be able to write and deploy a content based application using a mobile computing software framework
3. Be able to describe the differences in software development between mobile device programming and server application programming, and develop applications that incorporate both programming methods to efficiently construct a single application task.

**Learning Outcomes:**

Students will learn skills for creating and deploying Android applications, with particular emphasis on software engineering topics including software architecture, software process, usability, and deployment.

**Course Content:**

1. Chapter one: Introduction **2 hours**

* About Android
* Smartphone future

1. Chapter two: preparing the environment **3 hours**

* Installing the SDK
* Creating Android Emulator
* Installing Android Studio
* Installing Android Development Tools
* Choosing which Android version to use

1. Chapter three: android architecture **3 hours**

* Android Stack
* Android applications structure

1. Chapter four: Hello World Example **4 hours**

* Creating a project
* Working with the AndroidManifest.xml
* Using the log system
* Activities

1. Chapter five: UI Architecture **4 hours**

* Application context
* Intents
* Activity life cycle
* Supporting multiple screen sizes

1. Chapter 6: User Interface Widgets **3 hours**

* Text controls
* Button controls
* Toggle buttons
* Images

1. Chapter 7: Notification and Toast **3 hours**

* Parameters on Intents
* Pending intents
* Status bar notifications
* Toast notifications

1. Chapter 8: Menus **2 hours**

* Localization
* Options menu
* Context menu

1. Chapter 9::Dialogs **2 hours**

* Alert dialog
* Custom dialog
* Dialog as Activity

1. Chapter 10: Lists **3 hours**

* Using string arrays
* Creating lists
* Custom lists

1. Chapter 11: Location and Maps **4 hours**

* Google maps
* Using GPS to find current location

1. Chapter 12: Working with data storage **5 hours**

* Shared preferences
* Preferences activity
* Files access
* SQLite database

1. Chapter 13: Animation **4 hours**

* View animation
* Drawable animation

1. Chapter 14: Content providers **5 hours**

* Content provider introduction
* Query providers

1. Chapter 15: Network Communication **5 hours**

* Web Services
* HTTP Client
* XML and JSON

1. Chapter 16: Services **3 hours**

* Service lifecycle
* Foreground service

1. Chapter 17: Publishing Your App **5 hours**

* Preparing for publishing
* Signing and preparing the graphics
* Publishing to the Android Market

**Mode of Delivery:**

1. Lectures
2. Presentations
3. Tutorials
4. Individual Projects
5. Group work for projects

**Course Assessment:**

Exam: 60%

Project: 20%

Test: 10%

Presentations: 10%

**References:**

Bieman, J.M. and Murdock, V., (2001) Finding code on the World Wide Web: a preliminary investigation, Proceedings First IEEE International Workshop on Source Code Analysis and Manipulation.

Dykes, L. (2005) *XML For Dummies, 4th Edition*. Wiley. p. 20. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-7645-8845-1](http://en.wikipedia.org/wiki/Special:BookSources/0-7645-8845-1). *...it's a markup language, not a programming language.*

Duffy T. J. (2012) Programming with Mobile Applications: Android (TM), ios, and Windows Phone 7.

Ettinger, J. (2004) *Jacquard's Web*, Oxford University Press

Kelsey, R. et al (2006) William Clinger and Jonathan Rees (February 1998). ["Section 7.2 Formal semantics"](http://www.schemers.org/Documents/Standards/R5RS/HTML/r5rs-Z-H-10.html#%_sec_7.2). *Revised5 Report on the Algorithmic Language Scheme*. Retrieved 9 June.

Milner, R.M. et al. (1997). *The Definition of Standard ML (Revised)*. MIT Press. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-262-63181-4](http://en.wikipedia.org/wiki/Special:BookSources/0-262-63181-4).

O'Reilly M. (2013) [Programming the Mobile Web (Second Edition) by [Maximiliano Firtman](http://www.amazon.com/Maximiliano-Firtman/e/B003XEKKN2/ref=sr_ntt_srch_lnk_1?qid=1426855193&sr=1-1)](http://www.amazon.com/Programming-Mobile-Web-Maximiliano-Firtman/dp/1449334970/ref=sr_1_1?s=books&ie=UTF8&qid=1426855193&sr=1-1&keywords=mobile+programming) Apr 5, Published.

Powell, T. (2003). *HTML & XHTML: the complete reference*. McGraw-Hill. p. 25. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-07-222942-X](http://en.wikipedia.org/wiki/Special:BookSources/0-07-222942-X). *HTML is not a programming language.*

[Rojas, R.](http://en.wikipedia.org/wiki/Ra%C3%BAl_Rojas) et al. (2000). "Plankalkül: The First High-Level Programming Language and its Implementation". Institut für Informatik, Freie Universität Berlin, Technical Report B-3/2000. [(full text)](http://www.zib.de/zuse/Inhalt/Programme/Plankalkuel/Plankalkuel-Report/Plankalkuel-Report.htm)

Scott, M. (2006). *Programming Language Pragmatics*. [Morgan Kaufmann](http://en.wikipedia.org/wiki/Morgan_Kaufmann). p. 802. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-12-633951-1](http://en.wikipedia.org/wiki/Special:BookSources/0-12-633951-1). *XSLT, though highly specialized to the transformation of XML, is a Turing-complete programming language.*