Philadelphia University

Faculty of Information Technology

Department of Web Engineering

Course Catalogue

2017/2018

CONTENT

CHAPTER 1

INTRODUCTION TO CURRICULUM DESIGN

This catalogue describes the approach that has been used to design and organize the curriculum of the Web Engineering PROGRAMME and a brief description of its modules.

Web Engineering is the application of systematic, disciplined and quantifiable approaches to development, operation, and maintenance of Web-based applications. It is both a pro-active approach and a growing collection of theoretical and empirical research in Web application development.
This department is multidisciplinary and encompasses contributions from diverse areas: Software engineering and design, hypermedia/hypertext engineering, requirements engineering, human-computer interaction, user interface, information engineering, information indexing and retrieval, testing, modelling and simulation, project management, and graphic design and presentation.
In particular, Web engineering focuses on the methodologies, techniques and tools that are the foundation of Web application development and which support their design, development, evolution, and evaluation. The Bachelor of Engineering degree is designed to offer students fundamental knowledge and skills in Web engineering.

These modules are offered at the Department of Web Engineering, Faculty of Information Technology in Philadelphia University, to obtain the four years Bachelor of Science degree in Web Engineering (WE).
The information given in this catalogue is extracted for the PROGRAMME Specifications for the Degree PROGRAMME. These specifications are published separately.

1.1 Fundamental Concepts

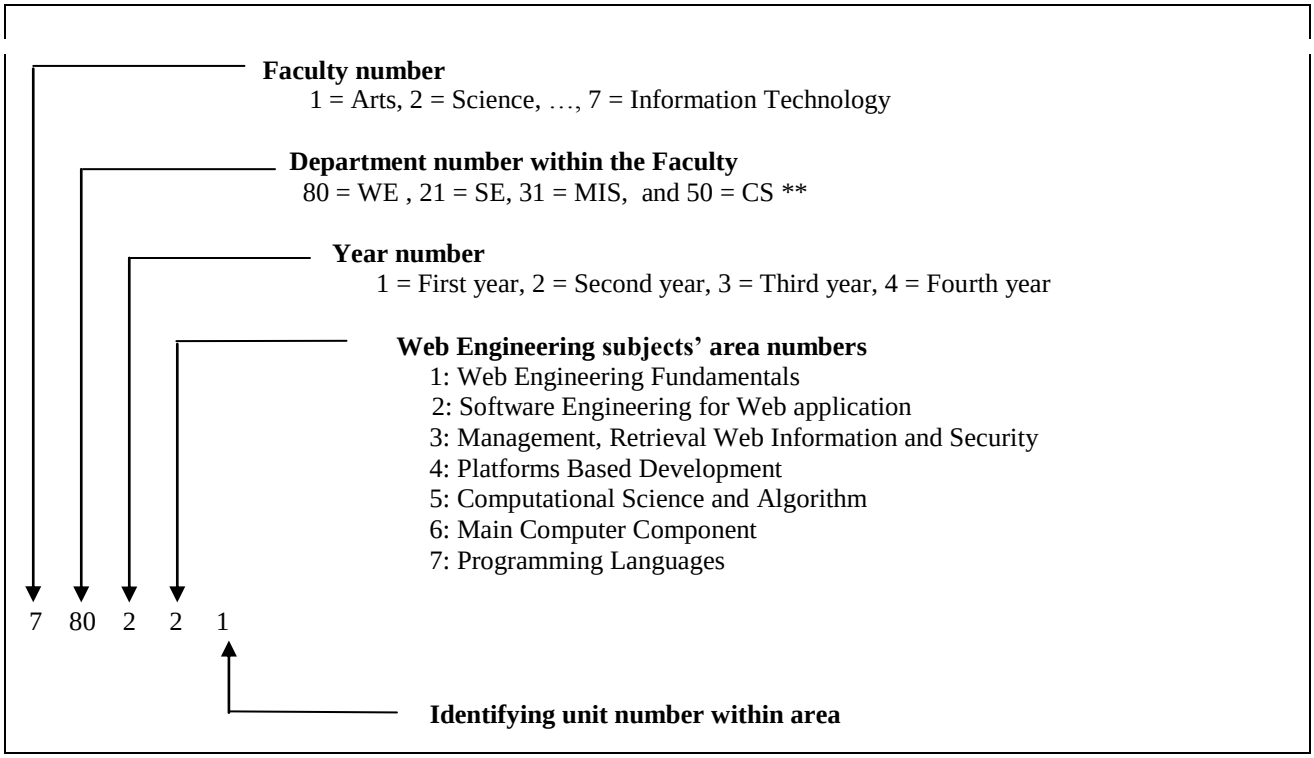
The most important concepts for understanding the module descriptions are as follows:

- **The Web Engineering Body of Knowledge.** The modules described in this Catalogue are defined in relation to a general taxonomy of that portion of Web Engineering appropriate for an undergraduate curriculum. That taxonomy represents the body of knowledge for Web Engineering. The highest level of the hierarchy is the **area**, which represents a particular disciplinary sub-field. The areas are broken down into smaller divisions called **units**, which represent individual thematic modules within an area. Each unit is further subdivided into a set of **topics**, which are the lowest level of the hierarchy.
- **Core and Elective Units**. Given the expanding scope of the computing discipline. The CS 2013 and SE2014 IEEE/ACM report defines a minimal set of core units for which there is a broad consensus that the material is essential to anyone obtaining an undergraduate degree in information technology disciplines. Because the core is defined as minimal, the core alone cannot constitute a complete undergraduate curriculum. Therefore, the undergraduate PROGRAMME include units from the body of knowledge of Web Engineering. These units could be mandatory or electives to be chosen according to the needs of the individual student.
- **Credit Hours.** To give a sense of the time required to cover a particular unit, a time metric should be chosen. The system of study at Philadelphia University is based on the credit hours. The basic measure unit of the curriculum is 3 credit hours module (or course unit). A module which delivers at least 3 hours per week of lectures or tutorial time, is worth 3 credit hours. Some modules may also provide an extra 1-hour per week for laboratory, but the module is still classified as 3 credit hours. In general, over a 16 weeks semester, a typical module provides minimum 45 hours of contact time. The final 3 hours of the semester is used for the examinations. The contact time corresponds to the in-class time required to present the material in a traditional lecture oriented format. Note that this time does not include the instructor's preparation time or the time students spend outside of class. As a general guideline, the time required outside of class is twice the time of the in class time. Thus, a unit that is listed as requiring 3 credit hours will typically entails a total of 9 hours (3 in class and 6 outside). It is also important to keep in mind that the time associated with each unit represents the minimum number of hours required for adequate coverage, and that it is always appropriate to spend more time than the listed minimum.

1.2 Format of the Module Coding Adopted

Each module in the Web Engineering PROGRAMME is identified by a code and a title. For example, "780221 Requirements Engineering for Web Applications" represents a module offered by Faculty of Information Technology, Department of Web Engineering in the second year, in the area of software engineering for Web application, and the module title is Requirements Engineering for Web Applications. Figure (1-1) illustrates the scheme of module coding and numbering, where the Requirements Engineering for Web Applications module is presented as an example.

Figure (1-1) Module Coding and Numbering Scheme



**** WE= Web Engineering , SE=Software Engineering, MIS=Management Information Systems, CS=Computer Science,**

CHAPTER 2

CURRICULUM DESIGN, ORGANISATION, AND CONTENT

2.1 Outlines of the Degree PROGRAMME

Within the general area of Web Engineering (WE), the modules recognize several major subject themes. This represents fundamental material on Programming, algorithms, Web engineering and software engineering, the structure and operation of computer systems including a high-level view of processing, memory, data communication and input/output devices, plus operating systems and user interfaces. This includes the theoretical foundations of computing, including Programming languages and formal analysis of algorithms and machines. Details of each module are set out in Chapter (3).

2.2 Requirements for the Degree PROGRAMME

The Web Engineering PROGRAMME is covered with different requirements. For obtaining their degrees, students must complete 44 modules; 43 are 3-credit hours courses, one 1-credit hour course, and one 2-credit hours course for a total of 132 credit hours summarized as follows:

-	9 modules (University requirements)	(27 credit hours)	(20 %)
-	8 modules (Faculty requirements)	(24 credit hours)	(18 %)
-	15 modules (Departmental Compulsories)	(42 credit hours)	(32 %)
-	3 modules (Departmental Electives)	(9 credit hours)	(7 %)
-	10 modules (Supportive modules)	(30 credit hours)	(23 %)

The Faculty requirements and University requirements include some computer-oriented modules that account to the Department requirements. (See Chapter (3), Table (3-1) for the titles of these modules).

2.3 Design, Organization, and Content of Curriculum

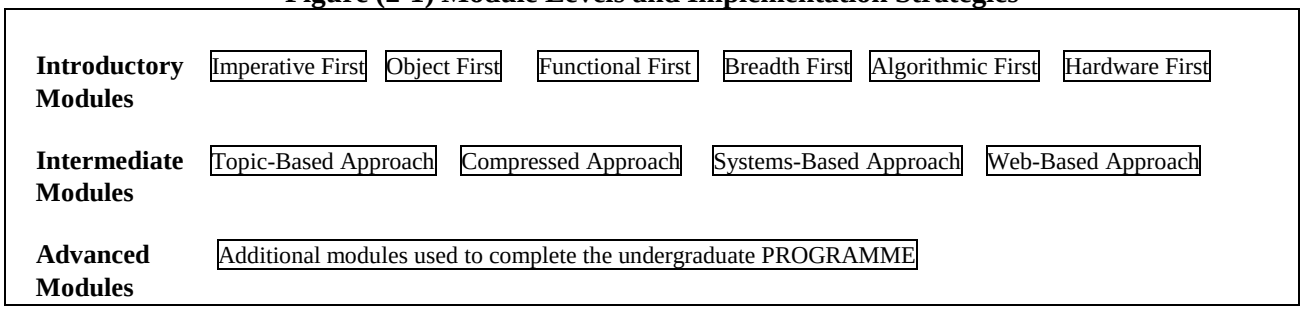
The modules are organized into three levels according to the year at which they occur in the curriculum are either Introductory, Intermediate, and advanced modules.

Modules designated as **Introductory** are offered in the first and second years of the Department curriculum. Modules listed as **Intermediate** are usually offered in the second or third year and build a foundation for further study in the field. Modules designated as **Advanced** tend to be taken in later years (third and fourth) and focus on those topics that require significant preparation in the earlier coursework. For these modules, the Department wishes to orient such modules to its own areas of expertise**.**

While these distinctions are easy to understand in their own right, it is important to recognize that there is no necessary relationship between the notions of core and elective - which apply to units in the body of knowledge - and the level of the module. The introductory and intermediate modules concentrate on core material, and the advanced modules include some core material and elective modules.

The point of organizing the modules into three levels: **Introductory**, **Intermediate**, and **Advanced** is to provide natural boundaries for defining implementation strategies. The CC2001 report defined many strategies. Figure (2-1) shows these strategies and their relationship in the curriculum.

Figure (2-1) Module Levels and Implementation Strategies



- For Introductory Modules, the Department adopted the **Imperative-First (or Procedural-First) strategy**. The imperative language is C. Then C# is adopted to introduce Object Oriented concepts.
- For Intermediate Modules, the Department adopted **Topic-Based strategy** to preparing for specific areas.
- Some Advanced Modules are selected to attend the departmental objectives and the areas of expertise.

The Web Engineering PROGRAMME is organized to cover specific knowledge areas. Table (2-1) shows the areas covered by the specialization Modules (including those computer-oriented modules taken from the Faculty and University requirements) and the number of modules in each of them. Note that the ratios in Table (2-1) are calculated according to the total number of modules (i.e. 44).

Table (2-1) Knowledge Areas and Number of Modules

		Compulsory Modules		Elective Modules		
	Area	**No.**	**(No./45) %**	**No.**	**(No./48) %**	**Modules**
1-	Computer Science & Algorithms	**3**	6.67%	**0**	0%	**3**
2-	Programming	**5**	11.11%	**0**	0%	**5**
3-	Information Science & Applications	**5**	11.11%	**2**	4.17%	**7**
4-	Internet Technologies	**16**	35.56%	**3**	6.25%	**19**
5-	Practical Training	**1**	2.22%	**0**	0%	**1**
6-	Research Project	**2**	4.44%	**0**	0%	**2**
7-	Statistics, Numerical Analysis, & Linear Algebra	**2**	4.44%	**0**	0%	**2**
	Total	**31**	**68.89%**	**Any 3**	**10.42%**	**34 (79.31%)**

- ***The Study Plan.*** The whole modules of the curriculum offered by the Department of Web Engineering are shown in Appendix A of this Catalogue.

- ***The Guidance Plan***. The Department guides students in their registration and selection of modules during the four years. The Department organizes a guidance plan that is shown in Table (2-2), where UR, FR, DR, and SR indicate University Requirements, Faculty Requirements, Department Requirements, and Supportive Requirements, respectively.

Table (2-2) Guidance Plan for the Web Engineering Curriculum

Year	First Semester				Second Semester			
	Module Number	Module Title	Prereq	Type	Module Number	Module Title	Prereq	Type
(1)	0114101	Arabic Language Skills (1)		Uni	0130102	English Language Skills (2)	0130101	Uni
	0130101	English Language Skills (1)		Uni	0111101	National Education		Uni
		University Elective 1		Uni		**University Elective 2**		Uni
	0750113	Programming Fundamentals (1)		Fac	0750120	Discrete Mathematics	0750099	Sup
	0731110	Introduction to Information Systems and Technology		Fac	0750114	Programming Fundamentals (2)	0750113	Fac
	0780110	Introduction to Internet and Web Technology		Fac	0780111	Web Engineering Fundamentals	0780110	Dept
	Semester Total		*18 Hours*		*Semester Total*		*18 Hours*	
(2)		**University Elective 3**		Uni	0250231	Introduction to Statistics and Probabilities	0750120	Sup
	0721223	Object-Oriented Programming	0750114	Fac	0721224	Data Structures	0721223	Sup
	0750272	Numerical Analysis	0750114	Sup	0750215	Visual Programming	0721223	Fac
	0731213	Introduction to Web Programming	0750114	Fac	0731221	Database Fundamentals	0721223	Sup
	0780220	Fundamentals of e-Government	0780111	Dept	0780221	Requirements Engineering for Web Applications	0780111	Dept
	0780230	Web Documents	0780111	Dept	0721240	Computing Ethics	0731110	Fac
	Semester Total		*18 Hours*		*Semester Total*		*18 Hours*	
(3)		**University Elective 4**		Uni		Department Elective 1		Dept
	0750323	Algorithms	0721224	Sup	0731340	Fundamentals of Computer Networks	0721224	Sup
	0750335	Operating Systems	0721224	Sup	0780323	Web Applications Usability	0780320	Dept
	0780320	Web System Analysis & Design	0780221	Dept	0780324	Web Services	0780340	Dept
	0780321	Web Process and Project Management	0780221	Dept	0780341	Web Client Side Technologies	0780230	Dept
	0780340	Web Server side Technologies	0780230	Dept				
	Semester Total		*18 Hours*		*Semester Total*		*15 Hours*	
(4)		Department Elective 2		Dept		Department Elective 3		Dept
	0780420	e-Commerce System Engineering	0780323	Dept	0780431	Web Security	0731340	Dept
	0750464	Information and Data Retrieval	0731221	Sup	0780481	Project (2)	0780480	Dept
	0780480	Project (1)	90 Hrs	Dept	0780423	Quality Assurance and Testing of Web Applications	0780320	Dept
	0780470	Practical Training	90 Hrs	Dept	0731423	Data Mining	0750464	Sup
	0111100	Military Sciences		Uni				
	Semester Total		*13 Hours*		*Semester Total*		*14 Hours*	

(Uni) University Req. (Fac) Faculty Req. (Dept) Dept. Req. (Sup) Supplementary Req.

CHAPTER 3

FULL DESCRIPTION OF MODULES

This chapter presents the full description of the Department modules and those modules from the Faculty and University requirements that are computer-oriented modules.

3.1 Module Descriptor

The Department organized a format for the module descriptor that includes much information on the module. This sub-section presents the components of the adopted module descriptor that are shown in Figure (3-1). The University Quality Assurance Catalogue explains in details the components of the module descriptor.

Figure (3-1) Components of the Module Description

Module Number, Module Title
Providing Department:
Module Coordinator(s):
Year:
Credit:
Prerequisites: Required modules or background
Aims:
Teaching Methods:
Learning Outcomes:
Assessment of Learning Outcomes:
Contribution to PROGRAMME Learning Outcome:
Syllabus: Bulleted list providing an outline of the topics covered.
Modes of Assessment:
Textbook and Supporting Materials:
Instructor:

3.2 Introductory Modules

Table (3-1) presents the Introductory (Level 1) modules whose full descriptions are given below.

Table (3-1) Introductory Modules in the Department of Web Engineering

Module Number	Module Title	Prerequisite
0250104	Discrete Mathematics	None
0731110	Introduction to Information Systems and Technology	None
0750113	Programming Fundamentals (1)	None
0750114	Programming Fundamentals (2)	0750113
0780110	Introduction to Internet and Web Technology	None
0780111	Web Engineering Fundamentals	0780110

250104, Discrete Mathematics

3 hours per week, 3 credit hours, Prerequisite: None

Aims:

This module will introduce the student to the basic language and ideas of discrete mathematics that occur in all branches of information technology. It will also begin the process of training the student to argue

correctly, both informally and formally, about these structures. The student will begin to learn the use of abstract analysis to solve concrete problems.

Teaching Methods: 32 hours Lectures (2 per week) + 16 hours Tutorials (1 per week)

Synopses: **Arithmetic:** The standard discrete number systems and the arithmetical operations on them with their properties; **Sets and Functions:** Standard set and function notation and terminology. Boolean operations on sets. Injective and surjective functions. Composition of functions; **Logic:** The connectives (or, and, not, implies, if and only if), Formulae of propositional logic, Truth tables, Tautologies and logical equivalence, Normal forms, The quantifiers (for all, there exists); **Binary Relations:** Definitions and examples, Properties of relations, Digraphs and representations of relations, Equivalence relations and Partitions, Combining relations and closure operators, Order relations, **Recurrence Relations:** Construction an solutions; **Induction:** The principle of mathematical induction, with many examples. Structural induction; **Combinatory:** Inclusion Exclusion principle, Binomial coefficients and permutations, Pascal's triangle. Summing series involving binomial coefficients.

Modes of Assessment:
Two 1-hour midterm exams (15% each); Coursework (15%); Tutorial Contribution (5%); Final (unseen) 2-hour examination (50%)

Textbooks and Supporting Material:

1- TRUSS, J.K. Discrete Mathematics for Computer Scientists. (ISBN 0-201-175-649) 2nd Edition, Addison Wesley 1998.

There is not a book, which covers exactly the material in this module. The above book covers a large part of the module but also contains additional material, some of which is covered in later modules.

There are many books on discrete mathematics, which have useful features. For example

1- MATTSON, H.F. Discrete Mathematics with Applications (ISBN 0-471-599-662), Wiley 1993.
2- GARNIER, R. and TAYLOR, J. Discrete Mathematics for New Technology (ISBN 0-750-301-35X) Institute of Physics Publishing 1992.
3- ECCLES, P.J, An Introduction to Mathematical Reasoning, (ISBN 0-521-59718-8) C.U.P. 1997.

……..

731110, Introduction to Information Systems Technology

3 hours per week, 3 credit hours, Prerequisite: None

Aims: This module aims to provide students with some concepts of information systems and some applications in business and management systems. This is a major introductory course presents problems in business environment and solutions with computer-based tools. It focuses on systems and information systems concepts and techniques. Students will learn the most effective ways to use information systems. Case studies are examined to highlight new technology and applications like multimedia.

Teaching Methods: 20 hours Lectures (1-2 hours per week) + 25 hours Class workshop and labs/E-Learning (1-2 per week) + 3 hours Workshops

Modes of Assessment: Two midterm exams (15% each); Homework (10%); Workshop Contribution (10%); 2-hours Final Exam (50%).

Textbooks and reference books:

1- Ralph M. Stair, George W. Reynolds, Fundamentals of Information Systems. Course Technology; 4th edition, 2007
2- Gerald M. Weinberg, an Introduction to General Systems Thinking, Silver Anniversary Edition, 2001.

3- The Analysis, Design, and Implementation of Information Systems, Henry C. Lucas, Jr, 4th ed. McGraw-Hill, 1992.
4- Leonard M. Jessup and Josef S. Valacich, Information Systems Foundations, 1999, Que E&T
5- James A. O'Brien, Introduction to Information Systems: Essentials for the e-Business Enterprise. 11th ed. 2003, McGraw-Hill Higher Education
6- David Kroenke, Management Information System, 1999.

……..

750113 Programming Fundamentals (1)

Course Hours: 3 hours per week, 3 credit hours (total of 48 hours)
Prerequisite: None

Aims:
This module aims to introduce computer programming and emphasis in problem solving on the fundamentals of structured design using the principles of Top Down problem solving strategy (divide and conquer). This includes development, testing, implementation, documentation. The module also aims to explore the logic of programming via the algorithm concepts and implement them in programming structures including functions, arrays, strings, and pointers.

Teaching Methods Duration: 16 weeks, 80 hours in total
Lectures: 32 hours (2 hours per week)
Tutorials: 16 hours (1 per week)
Laboratories: 32 hours, 2 per week

Synopsis:
Problem solving strategies, algorithmic language to describe such problem solving, introduces the principles of procedural programming, data types, control structures, data structures and functions, data representation on the machine level. Various problems are considered to be solved using C-like procedural programming language.

Assessment:
Two 1-hour midterm exams (15% each); lab (30%); one 2-hours Final Examination (40%)

Textbook:
- P. Deitel & H. Deitel, C++ How to program, Pearson Education Limited, 2013.
- Guttag, John. Introduction to Computation and Programming Using Python. Spring 2013 edition. MIT Press, 2013. ISBN: 9780262519632. – MIT

……

750114 Programming Fundamentals (2)

Course Hours: 3 hours per week, 3 credit hours (total of 48 hours)
Prerequisite: 750113

Aims:
This module aims to introduce computer programming and emphasis in problem solving on the fundamentals of structured design using the principles of Top Down problem solving strategy (divide and conquer). This includes development, testing, implementation, documentation. The module also aims to explore the logic of programming via the algorithm concepts and implement them in programming structures including functions, arrays, strings, and pointers .

Teaching Methods: Duration: 16 weeks, 80 hours in total
Lectures: 32 hours (2 hours per week),
Tutorials: 16 hours (1 per week)
Laboratories: 32 hours, 2 per week

Synopsis:
Functions definition, Parameters definition and passing , One dimensional array, Two dimensional array, use of main operations of a sequential file: open, reset, rewrite, read, write, eof, Introduction to Class and object, Generics, components reuse, component programming Various problems are considered to be solved using C-like procedural programming language.

Assessment:
Two 1-hour midterm exams (15% each); lab (30%); One 2-hours Final Examination (40%)

Textbook
D.S. Malik, Thomson, C++ Programming: From Problem Analysis to Program Design, Sixth Edition, Course Technology, 2011

……

780110,: Introduction to Internet and Web Technology

3 hours per week, 3 credit hours, Prerequisite: None

Internet, Internet services, internet applications and tools. World Wide Web, WWW applications and tools, mark-up language, e-mail, web browsers, Web-based research and information resources, World Wide Web Consortium (W3C), creating web pages, search engines, FTP.

- **Books (title , author (s), publisher, year of publication)**
  - Web Technology Theory And Practice Paperback – 2012 by M Srinivasan (Author)
  - Web Programming And Internet Technologies: An E-Commerce Approach 1 Pap/Cdr Edition by Porter Scobey (Author), Pawan Lingras (Author) 2012

0780111, Web Engineering Fundamentals

3 hours per week, 3 credit hours, prerequisite: 750113+0731110

Overview of requirements, analysis modelling, design modelling, testing.
Internet basics for web applications. Technologies and tools for developing web applications: mark-up languages, styling, data description and transformation, client and server side Programming. Web services. Advances in web engineering.

Textbook:
R. Pressman, Web Engineering: A Practitioner's Approach, McGraw-Hill Higher Education, 2008.
http://highered.mcgraw-hill.com/sites/0073523291/

3.3 Intermediate Modules

The Intermediate (Level 2) modules are listed in Table (3-2) and their full descriptions are given below.

Table (3-2) Intermediate Modules in the Department of Web Engineering

Module Number	Module Title	Prerequisite
0250231	Introduction to Statistics and Probabilities	0750120
0721223	Object-Oriented Programming	0750114
0721224	Data Structures	0721223
0721240	Computing Ethics	0731110
0731213	Introduction to Web Programming	0750114
0731221	Database Fundamentals	0721223
0750215	Visual Programming	0721223
0750272	Numerical Analysis	0750114
0780220	Fundamentals of e-Government	0780111
0780221	Requirements Engineering for Web Applications	0780111
0780230	Web Documents	0780111

250231, Introduction to Probability and Statistics

3 hours per week, 3 credit hours, Prerequisite: None

Aims: This module aims to help students grasp basic statistical techniques and concepts, and to present real-life opportunities for applying them.

Teaching Method: 30 hours Lectures (2 per week) + 15 hours Tutorials (1 per week)

Synopsis: Descriptive statistics and probability distribution; Sampling distribution Estimation for the mean, variance and proportions; Testing for the mean, variance and proportions; Regression and correlation; One-way analysis of variance.

Assessment: Two 1-hour midterm exams (15% each); Assignments/Quizzes (10%); Tutorial Contribution (10%); 2-hours Final Exam (50%).

Textbooks:

1- D.C. Montgomery and .G.C. Runger, Applied Statistics and Probability For Engineers, 2nd Edition, Wiley, 2002
2- William, Probability and Statistics in Engineering and Management, Wiley, 2002

……

721223, Object-Oriented Programming

Course Hours: 4 hours per week (1 hour for practice) , 3 credit hours
Prerequisite: 750114

Teaching Method: 30 hours lectures (2 hours per week) + 15 hours Tutorials (1 per week) + 15 hours Laboratory (1 per week)

Aims: The module aims to develop an understanding of the principles of the object-oriented paradigm; to provide familiarity with approaches to object-oriented modelling and design; to provide a familiarity with the syntax, class hierarchy, environment and simple application construction for an object-oriented Programming language. The module emphasizes developing fundamental Programming skills in the context of a language that supports the object-oriented
Synopsis: Introduction: Classes, Objects, Methods, and Properties, A method deeper Look: static Methods, static Variables, Scope of Declarations, and Method Overloading, Declaring and Creating Arrays, Array of Object, and Generic Collection, Classes and Objects A Deeper Look: Data Abstraction and Encapsulation, Controlling Access to Members, static Class Members, Referring to the Current Object's, Overloaded, Constructors, and Composition, Inheritance : Base Classes and Derived Classes, protected Members, Relationship between Base Classes and Derived Classes, Constructors in Derived Classes, and Class object, Polymorphism: Polymorphic Behaviour, Abstract Classes and Methods, Using Interfaces, Inheritance : Base Classes and Derived Classes, protected Members, Relationship between Base Classes and Derived Classes, Constructors in Derived Classes, and Class object, Polymorphism: Polymorphic Behaviour, Abstract Classes and Methods, Using Interfaces, Exception Handling.

Assessment: Two midterm exams (20% each); Laboratory (20%); Tutorial contribution (20%); Final exam (40%).

Textbooks:

1- Deitel & Deitel, C# 2010 for Programmers, Publisher: Prentice Hall, 2011, ISBN-10: 0132618206 | ISBN-13: 9780132618205.
2- Purdum, Jack, " Beginning C# 3.0 : an introduction to object oriented Programming ", Wiley Publishing, Inc., 2007.
3- Deitel , Paul J. , Deitel, Harvey M., "C# 2008 for Programmers", 3rd ed., Prentice Hall, 2009 (ISBN: 978-0-13-714415-0).
4- Palmer, Grant, Barker, Jacquie, "Beginning C# 2008 objects: from concepts to code", Berkeley: Apress, 2008. (ISBN: 978-1-4302-1088-7).
5- Marshall , Donis, " Programming Microsoft Visual C# 2008 : the language", Redmond, Washington: Microsoft Press, 2008. (ISBN: 978-0-7356-2540-2)

721224, Data Structures

Course Hours: 4 hours per week (1 hour for practice), 3 credit hours

Prerequisite: **721223**

Teaching Method: 30 hours lectures (2 hours per week) + 15 hours Tutorials (1 per week) + 15 hours Laboratory (1 per week)

Aims: This is a **Programming-intensive** module where students learn the fundamentals of designing data structures for use in complex PROGRAMMEs. Data structures module is an essential area of study for computer scientists and for anyone who will ever undertake any serious Programming task. This module deals with the fundamentals of organizing and manipulating data efficiently using clean conceptual models. Students study many of the important conceptual data types, their realization through implementation, and analysis of their efficiency. Implementations in

this module are carried out in the Java Programming language, but the principles are more generally applicable to most modern Programming environments.

Topics include recursion, the underlying philosophy of object-oriented Programming, fundamental data structures (including stacks, queues, linked lists, hash tables, trees, and graphs), and the basics of algorithmic analysis.

Synopsis: Data Design and Implementation, Abstract Data Types, Lists, Stacks and Queues, Linked Lists, Programming with Recursion, Binary Search Trees, Priority Queues and Heaps, Graphs, Introduction to Algorithms analysis, Sorting and Search Algorithms.

Assessment: Two 1-hour midterm exams (20% each); Assignments (20%); 2-hours final exam (40%).

Textbooks:

1- Daniel T. Joyce & Chip Weems,, Object-Oriented Data Structures Using Java, Third Edition, Nell Dale, Jones and Bartlett Publishers Inc., 2012, ISBN-13: 9781449613549.
2- Kaur, Samarjeet, " Data structure: complete course book", New Delhi: Deep & Deep Publications, 2006. (ISBN: 81-7629-774-7)
3- Main, Michael, " Data structures & other objects using Java: using J2SE 5.0", Boston: Pearson Addison Wesley, 2006. (ISBN: 0-321-36412-0)
4- Malik, D. S, " Data structures using C++ ", Boston, MA: Course Technology, 2010. (ISBN: 978-1-4390-4023-2)

……..

721240, Computing Ethics

Course Hours: 3 hours per week, 3 credit hours (total of 48 hours)

Prerequisite: 731110

***Level*: 2**

Aims:
This module aims to give students an informed awareness of the principal issues of professional ethics and responsibility (ergonomics and ethics) in the analysis, design, implementation and use of computers, information systems and Information Technology (IT) products. This will help students in recognition of ethical problems when they occur. Also it will enable students to deal effectively with ethical, social and professional issues now and in their future careers.

Teaching Methods: 36 hours Lectures (2-3 per week) + 9 hours Projects (class work) (average 1 per week) + 3 hours Seminars (1 per month)

Synopsis*:* Introduction to Ethics; Professional and Professionalism; Code of Ethics and Social Issues; Computer/IT professionals; Computer Security; Privacy and Internet Issues; Information Systems and Ethics; Associations of IT professionals; Ethics and the Internet; Ethical Challenges of e-Business; Ethical Challenges of e-Business; Continuous Professional Development; Intellectual Property Rights; Jordanian Codes for Intellectual Property Rights; Seminars and Project Discussion.

Modes of Assessment:

Two 1-hour midterm exams (20% each); Assessment by individual essay (10%); Workshop Assessment (10%); 2 hour written final Exam (40%)

Textbooks and Supporting Material:

1. Deborah G. Johnson, Computer Ethics. 3ed Edition, Englewood Cliffs, N.J., Prentice Hall, 2001.
2. Gorge Reynoids, Ethics in Information Technology, Thomason, 2003.
3. Sara Baase, A Gift of Fire: Social, Legal and Ethical Issues for Computer and the Internet, 2nd ed., 2003.
4. Tavani H. T. and Hoboken N. J., Ethics and Technology, John Wiley, 3rd ed, 2004.
5. مجموعة تشريعات الملكية الفكرية الأردنية

Website(s):
ACM, IEEE and BCS Web Sites.
www.cyberethics.cbi.msststе.edu
www.aitp.org
www.acm.org
www.prenhall.com
www.jcs.rg.jo

……..

731213, Introduction to Web Programming

3 hours per week, 3 credit hours, prerequisite: **750114**

Aims:
This module aims to give students an introduction and general concepts of the Internet and Intranet technology, the World Wide Web, TCP/IP and Web design languages (HTML, CSS, JavaScript, and ASP). It also involves the necessary background that student needs to develop different tasks of Programming aspects concerning the foregoing objectives. Sufficient study levels are supposed to be studied and learned by the students within the course for the sake of applying the different fields of education, learning, economical, E-Business and other approaches.

Teaching Methods: 32 hours Lectures (2 per week) + 8 hours Tutorials (1 per 2 weeks) + 24 hours Laboratory (1-2 per week)

Synopsis*:* Internet and Intranet Technology: Concepts, protocols, Services, and architecture, TCP/IP Architecture and Protocols (Client & Server), DNS, Internet Service Providers (ISP), Internet Services: USENET News, E-Mail, FTP, and Telnet; The Web: Basic Concepts, WWW and Web Servers, Links: Hyperlinks & Hypermedia, Web pages and home pages, Browsers & Search Engines; Introduction to Mark-up Languages; Editing HTML, HTML Tags: Headers, HTML Tags: Text Styling and Formatting, and linking; HTML Tags: Images and Image maps; Basic HTML Lists and Tables; Basic HTML Forms and Frames; Frames and Cascading Style Sheets; Cascading Style Sheets and Introduction to Client Scripting; Simple JavaScript PROGRAMMEs; JavaScript: Control Structures, if, if/else, While, for, and switch. JavaScript: Break and Continue statements; JavaScript: Functions, Arrays.

Modes of Assessment: Two 1-hour midterm exams (15% each); Lab work (15%); Tutorial contribution (5%); 2-hours Final Exam (50%).

Textbooks and reference books:

1- Deitel & Deitel, Internet and World Wide Web How to PROGRAMME, Prentice Hall, 3rd Edition, 2004.
2- Douglas Comer, Computer Networks & Internets, Prentice Hall, 2003

3- Brian Salter, A simple guide to HTML, Prentice Hall, 2002.
4- Ellenn Behoriam, "HTML and XHTML: Creating Web Pages", Prentice Hall, 2002
5- Gary Rebholz, "How to Use HTML & XHTML", Sams, 2001
6- Ellie Quigley, "JavaScript by Examples", Prentice Hall, 2004
7- Tom Negrino, "JavaScript for the World Wide Web: Visual Quick Start Guide", Student Edition, 5/E, Peachpit Press, 2004
8- Susan Anderson-Freed, "Weaving a Website: Programming in HTML, Java Script, Perl and Java", Prentice Hall, 2002

Website(s):

1. www.w3schools.com
2. www.webteacher.org.
3. www.microsoft.com.
4. www.whatis.com.
5. www.idocs.org.
6. www.w3.org.
7. www.webdeveloper.com
8. www.javascriptmall.com
9. www.javascripts.com/toc.cfm
10. www.Deitel.com

……

731221, Database Fundamentals

3 hours per week, 3 credit hours, prerequisite: 721223

Aims: This module aims to provide students with an overview of database management system architectures and environments, an understanding of basic database design and implementation techniques, and practical experience of designing and building a relational database. The other aim of this module is to make students able to discuss/explain the importance of data, the difference between file management and databases. In addition, it enables students to apply conceptual design methodologies for a database and learn about architectures and environment of database management system (in particular the Ansi-Sparc model). This module requires a practical work, which is assessed by producing individual and group small projects.

Teaching Method: 30 hours lectures (2 hours per week) + 15 hours Laboratory (1 per week).

Synopsis: General introduction and database systems. Architectures: Ansi-Sparc model of databases, components of a database management system, DBMS functions schemas, levels of abstraction and mappings, role of the data dictionary, client-server systems, PC based systems, database servers, distributed systems. General database design: Design framework, mappings between abstractions, integrity, compromises, data vs functional design, non-functional considerations e.g. performance, volumes, user interface etc., security. Conceptual design: Requirement for conceptual design, Extended Entity Relationship model, object-oriented design. Logical design: The relational model, normalization, relational algebra, SQL, mapping conceptual design to relational, integrity, views, embedded SQL, PL/SQL, triggers. Relational databases: Mapping conceptual schema to a relational schema; entity and referential integrity; relational algebra and relational calculus. Database query languages: Overview of database languages; SQL; query optimization. Relational database design: Database design; functional dependency; normal forms; multivalued dependency; join dependency; representation theory. Physical design: Clustering, indexes, performance considerations. Transaction processing: Transactions, Concurrency techniques (locking, 2-phase locking, serialisability), recovery (rollback and commit, 2-phase commit), Transaction Processing Management Systems. Introduction to distributed databases: Distributed data storage; distributed query processing; distributed transaction model; concurrency control; homogeneous and heterogeneous solutions; client-server. Physical database design: Storage and file structure; indexed files; hashed files; signature files; b-trees; files with dense index; files with variable length records; database efficiency and tuning.

Assessment: Two 1-hour midterm exams (15% each) + Lab work and Assignments (20%) + 2-hours Final Exam (50%).

Textbooks:

1- Elmasri R. and Navanthe S. B., Fundamentals of Database Systems, 3rd edition, (ISBN 0-201542633), Addison Wesley, 1999.

2- C. J. Date, An Introduction to Database Systems,

……..

750215, Visual Programming

3 hours per week, 3 credit hours, prerequisite: 721220

Aims: This module aims to provide students capabilities to design and implement the applications using visual Programming through Microsoft Visual Studio .Net and VC# to develop different types of applications using .Net platform.

Teaching Methods: 32 hours Lectures (2 per week) + 12 hours Tutorials (on average 1 per week) + 16 hours Laboratory (1 per week) + 4 hours Seminar

Synopsis: **Introducing the Microsoft .NET Platform:** .NET Platform, .NET and Windows DNA, .NET Architecture Hierarchy, .NET Platform features, Multilanguage Development, Platform and Processor Independence, .NET Components, Common Type System CTS, Common Language Specification CLS , .NET Base Class Library (BCL); **Visual Studio.NET IDE:** Visual Studio.NET, Components of VS.NET, Design Window, Code Window, Server Explorer, Toolbox, Docking Windows, Properties Explorer, Solution Explorer, Object Browser, Dynamic Help, Task List Explorer, Features of VS.NET, XML Editor, Creating a Project, Add Reference, Build the Project, Debugging a Project; **Introducing C# Programming:** Data Types, Value Types, Reference Types, Control Structures (if, if-else, switch, for, while, do while, break, continue, return, goto), Understanding Properties and Indexers Accessing Lists (Array) with Indexers, Events, Exception Handling, Using OOP (Object, Class, Constructor/destructor, Inheritance, Polymorphism, Encapsulation); **Windows Forms:** Windows Forms, Adding Controls, Adding an Event Handler, Adding Controls at Runtime, Attaching an Event Handler at Runtime, Writing a Simple Text Editor, Creating a Menu, Adding a New Form, Creating a Multiple Document Interface, Creating a Dialog Form, Using Form Inheritance, Adding a *TabControl,*, Anchoring Controls, Changing the Startup Form, Connecting the Dialog, Using the ListView and TreeView, Controls, Building an *ImageList*, Adding a *ListView*, Using the Details View, Attaching a Context Menu, Adding a *TreeView*, Implementing Drag and Drop, Creating Controls, Creating a User Control, Adding a Property, Adding Functionality, Writing a Custom Control, Testing the Control, Enhancing the Control, Sub classing Controls; **Graphics and Multimedia:** Graphics Contexts and Graphics Objects, Color Control, Font Control, Drawing Lines, Rectangles and Ovals, Drawing Arcs, Drawing Polygons and Polylines, Advanced Graphics Capabilities, Introduction to Multimedia, Loading Displaying and Scaling Images, Animating a Series of Images, Windows Media Player, Microsoft Agent; **ADO.NET:** ADO.NET Architecture, Understanding the *Connection*Object, Building the *Connection* String, Understanding the *Command*Object, Understanding *DataReaders*, Understanding *DataSets* and *DataAdapters*, DataTable, *DataColumn*, *DataRow*, Differences between *DataReader* Model and *DataSet* Model, Understanding the *DataView*Object ,Working with System.Data.OleDb, Using *DataReaders*, Using *DataSets*, Working with SQL.NET, Using Stored Procedures, Working with Odbc.NET, Using DSN Connection; **Multithreading:** Thread States: Life Cycle of a Thread, Thread Priorities and Thread Scheduling, Thread Synchronization and Class Monitor, Producer/Consumer Relationship without Thread, Synchronization, Producer/Consumer Relationship with Thread Synchronization, Producer/Consumer Relationship: Circular Buffer; **Networking:** Introduction, Establishing a Simple Server (Using Stream Sockets), Establishing a Simple Client (Using Stream Sockets), Client/Server Interaction with Stream-Socket Connections, Connectionless Client/Server Interaction with Datagrams, one Server multi-Clients system; **ASP.NET:** Introducing the ASP.NET Architecture, ASP.NET Server Controls, Working with User, Controls, Custom Controls, Understanding the Web.config File, Using the Global.asax Page,

Modes of Assessment: Two 1-hour midterm exams (15% each); Assignment 15%; Tutorial Contribution (5%); 2-hours Final Exam (50%: 35% Written Exam + 15% Practical Exam)

Textbooks and reference books:

1- H. M. Deitel & J. Deitel, " C# How to Program", Prentice Hall, 2001
2- A.Turtschi et.al. " Mastering Visual C# .Net", Sybex 2002
3- Eric Gunnerson, "A PROGRAMMEr's Introduction to C#", Apress 2000
4- Anders Hejlsberg et.al. " C# Language Reference", Microsoft Corporation 2000
5- Erric Buttow et al. "C#, your visual blueprint for building .Net application", Hungry Minds 2002
6- Charles Carroll, "Programming C#", O'Reily & Associates 2000
7- Karh Watson "Beginning C#" Wrox Press 2001.

……..

750272, Numerical Analysis

Course Hours: 3 hours per week, 3 credit hours (total of 48 hours) Prerequisite: 250101 + 750114

Aims:

The aim of the module is to give students a clear understanding and deep knowledge how the typical of "real life" mathematical, physical, or engineering problems are to be solved in the modern setting. As opposed to tendency in lower-level mathematical courses to teach recipes for "exact" solving particular problems fitting into very special form, this module provides the idea of approximate solving wide variety of applied standard problems on a computer by numerical methods.

Teaching Methods:

Lectures: 36 hours,
Tutorials: 12 hours

Synopsis:

Mathematical Preliminaries: Computer arithmetic, round-off error, source of errors, Solution of equations in one variable: Bisection method, fixed point method, false position method, Secant method, Newton-Raphson method, Interpolation and polynomial approximation, Introduction to interpolation, Direct methods for solving linear systems of equations, Iterative methods for solving linear systems, Iterative methods for solving nonlinear systems,Curve fitting techniques.

Assessment:

Two 1-hour midterm exams (20% each); Assignments (20%); One 2-hours Final Examination (40%)

Text book:

Richard L. Johnson and Douglas J. Faires, Numerical Analysis, 9th Edition, Brooks/ Cole 2010.

……..

0780220, Fundamentals of e-Government

3 hours per week, 3 credit hours, prerequisite: 780111

Aims:

This course aims to introduce students to the fundamentals components of an e-government and how to best design them.

Synopsis:

E-government is a way for governments to use the most innovative information and communication technologies, particularly web-based Internet applications, to provide citizens and businesses with more convenient access to government information and services, to improve the quality of the services and to provide greater opportunities to participate in democratic institutions and processes.

Topics include: The Information Society and E-Government, The Concept and Strategy of E-Government, E-Government Business Models, User-driven E-Government, E-Government Services, Multichannel E-Service Delivery, Success Factors of E-Government, E-Government Implementation, E-Government Case Studies

Textbook:

e-Government, Strategy Process Instruments. Bernd W. Wirtz & Peter Daiser, 2015

……

0780221, Requirements Engineering For Web Applications

3 hours per week, 3 credit hours, prerequisite: 780111

Categories of Requirements: Web Site, Business, non-functional, functional, technical; Requirements Engineering, activities, (A) elicitation, main steps, stakeholders, communication, techniques, challenges (B) description, artefacts, adequacy, user stories, itemized requirements, template (C) validation, verification, requirements for requirements (D) management

Textbooks:

Agile Software Requirements: Lean Requirements Practices for Teams, PROGRAMMEs, and the Enterprise (Agile Software Development Series) Hardcover – January 6, 2011 Dean

Engineering Web Applications (Data-Centric Systems and Applications) Hardcover – August 5, 2009 by Sven Casteleyn (Author), Florian Daniel (Author) Peter Dolog (Author) Peter Dolog (Author), Maristella Matera (Author)

……

0780230, Web Documents

3 hours per week, 3 credit hours, prerequisite: 780111

Mark-up Languages ;Hypertext, Rendering, Mark-ups (A) SGML, Element, Tag, DTD (B) HTML, Tags, kinds of mark-ups, history, HTML 4.01, XHTML, HTML5 (C) XML, elements, attributes, entities, well-forkedness, names spaces, validity, DTD, constituents, CDATA, PCDATA, XSD, constituents, types Cascading Style Sheets; CSS, history, cascading, author/user styles XML Technologies; (A) DOM, objects, tree, API; (B) XSL-T, transformation, pattern matching, usage.

Textbooks:

Beginning XML Paperback – July 10, 2012, by Joe Fawcett (Author), Danny Ayers (Author),

XML, DTDs, Schemas: The Personal Trainer Paperback – April 30, 2014by William Stanek

3.4 Advanced Modules

In this sub-section, the full descriptions of Level 3 modules are presented. Table (3-3) shows these modules and their descriptions are given below.

Table (3-3) Advanced Modules

Module Number	Module Title	Prerequisite
0731340	Fundamentals of Computer Networks	0721224
0750323	Algorithms	0721224
0750335	Operating Systems	0721224
0780320	Web System Analysis and Design	0780221
0780321	Web Process and Project Management	0780221
0780323	Web Applications Usability	0780320
0780324	Web Services	0780340
0780340	Web Server Side Technologies	0780230
0780341	Web Client side Technologies	0780230
0731423	Data Mining	0750464
0750464	Information and Data Retrieval	0731221
0780420	e-Commerce System Engineering	0780323
0780423	Quality Assurance and Testing of Web Applications	0780320
0780431	Web Security	0731340
0780470	Practical Training	90 Hrs
0780480	Project (1)	90 Hrs
0780481	Project (2)	0780480

0731340, Fundamentals of Computer Networks

Credit: 3 credit hours, Prerequisite: 721224

Aims:

This module is the first module of the curriculum related to the computer network field. Its aim is to provide students with a broad coverage of the basic computer networking concepts of the four layers of ISO, circuit switch, packet switch, etc.

The module, however, does not focus on a detailed study or cover the technologies. The concepts given in this module will be deeply handled in the next level module (750441).

Teaching Methods:

32 hours Lectures (2 per week (including two 1-hour midterm exams)) + 16 hours
Tutorial (1 per week) + 16 hours Laboratory

Synopsis:

Introduction; Network Model; Data and Signal; Digital signal; Analog signal; Switching; Error Detection and Control; Error Detection and Control; Data Link Control; Multiple Access; Network Layer: Logical Addressing; Network Layer: Delivery, Forwarding, and Routing; Network Layer: Delivery, Forwarding, and Routing; Process-to process Delivery; Congestion Control and Quality of service.

Modes of Assessment:
Two 1-hour midterm exams (20% each); Assignments (20%); One 2-hours Final Examination (40%)

Textbooks and Supporting Material:
1- Behrouz A. Forouzan, Data Communications and Networking, McGraw Hill Higher Education, Fourth Edition, 2007
2- Andrew S. Tanenbaum, Computer Networks, Prentice Hall, Last Edition.
Website(s): www.mhhe.com/forouzan

……..

0750323, Algorithms

Credit: 3 credit hours, Prerequisite: 721224

Aims:
The aim of this module is to learn how to develop efficient algorithms for simple computational tasks and reasoning about the correctness of them. Through the complexity measures, different range of behaviours of algorithms and the notion of tractable and intractable problems will be understood. The module introduces formal techniques to support the design and analysis of algorithms, focusing on both the underlying mathematical theory and practical considerations of efficiency. Topics include asymptotic complexity bounds, techniques of analysis, and algorithmic strategies.

Teaching Methods:
38 hours Lectures (2 per week (including two 1-hour midterm exams)) + 10 hours
Tutorials (average 1 hour per week)

Synopsis:
Introduction, Algorithm definition, Algorithm Analysis; Mathematical Induction; Summation Techniques; Recurrence Relations; Design & Analysis of Algorithms: Divide and conquer, Greedy Algorithm, Dynamic Programming, Backtracking, Branch-Bound; Lower Bound Theory; Sorting and Searching; NP-Complete Problems: Basic Concepts, NP-Hard & NP-Complete Problem

Modes of Assessment:
Two 1-hour midterm exams (15% each); Tutorial contributions (5%), Coursework (15%); Final written Examination (50%)

Textbooks and Supporting Material:
1- Jon Kleinberg, Eva Tardos, Algorithm design, Boston: Pearson Education Limited, 2014.
2- Alwan, Raad F., Design and Analysis of Algorithms, Dar Majdalawi Publication & Distribution, Amman, 2010.
3- Sara Baase, Computer Algorithms: Introduction to Design and Analysis, Third Edition, Addison-Wesley, 2000.
4- Udi Manber, Introduction to Algorithms: a Creative Approach, Addison-Wesley, 1997.
5- T. Cormen, et.al., Introduction to Algorithms, 1999.
6- R. Sedgewick, Algorithms in C++, 2002.

……..

750335, Operating Systems

3 hours per week, 3 credit hours, prerequisite: **750232**

Aims:
The aims of this module are to introduce the basic principles of computer systems organization and operation; to show how hardware is controlled by PROGRAMME at the hardware/software interface; to outline the basic OS resource management functions: memory, file, device (I/O), process management, and OS security/protection. Two concrete examples of operating systems are used to illustrate how principles and techniques are deployed in practice.

Teaching Method: 40 hours Lectures (2-3 per week) + 8 hours Tutorials (1 each fortnight)

Synopsis*:* Operating System overview; Operating System Structures: System components, Operating system services, System calls, System structures, Virtual machine; Processes: Process concept, Process scheduling, Operation on process, Cooperative process, Inter process communication; Threads: Thread overview, Benefits, User and kernel threads, Multithreading model, Solaris 2 threads; CPU Scheduling: Basic concept, Scheduling criteria, Scheduling algorithm, Thread scheduling, Algorithm evaluation; Process synchronization and mutual exclusion: Critical section problem, Two task solution, Synchronization hardware, Semaphore, Classical synchronization problem; Deadlock and starvation: System model, Deadlock characterization, Method for handling deadlock, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock; Memory management: Background, Swapping, Paging, Virtual memory, Background, Demand paging, Page replacement, Allocation of frame, Thrashing; File system implementation and management: File concept, Access method, Directory structure, Protection, File system structure, Allocation method, Free space management, Directory implementation, Efficiency and performance, I/O management and disk scheduling, Application I/O interface, Kernel I/O subsystem, I/O request handling, Disk structure, Disk scheduling, Disk management, Swap space management, Disk reliability, Stable storage implementation

Modes of Assessment:
Two 1-hour midterm exams (15% each); Assignments (10%); Lab work (5%); Tutorial contribution (5%); 2-hours Final Examination (50%)

Textbooks and Supporting Material:
1- A. Silberschatz and Peter Galvin, Applied Operating Systems Concepts, First edition, John Wiley & sons, Inc., 2000
2- J. Bacon, Concurrent Systems: Database and Distributed Systems, 2nd Edition, (ISBN 0-201-177-676), Addison Wesley, 1998.
3-A. S. Tanenbaum, Modern Operating Systems, Prentice Hall, 1992

……..

0780320, Web System Analysis and Design

3 hours per week, 3 credit hours, prerequisite: 780221

Model driven methodology for Web Domain; Analysis Models, Design Models, UI Models ; Web Modelling Languages: UWE, Web ML, IFML; Dimensions for Web Modelling ; Structural versus Behavioural Modelling; Modelling Requirements; Use case diagrams, stereotypes, navigation, scenarios; Modelling Content; Content, domain model, behaviour, document life cycle
Modelling Hypertext, Navigation, navigation structure model, access model, link types, navigation patterns; Modelling Representation; Presentation modelling, layout, presentation structure, presentation elements, behaviour of a presentation
Customization of Models; Context, personalization, mobility, annotations, aspect, weaving

Textbooks:

Web Engineering: Modelling and Implementing Web Applications - Gustavo Rossi, Oscar Pastor, Daniel chwabe, L. Olsina, Human-Computer Interaction Series (Paperback), 2010

Interaction Flow Modeling Language: Model-Driven UI Engineering of Web and Mobile Apps with IFML, Marco Brambilla & Piero Fraternali, Elsevier, 2014

……

0780321, Web Process and Project Management

3 hours per week, 3 credit hours, prerequisite: 0780221

Web project management method, management in commercial Web sites, organize and put together a team, develop goals, manage schedules and budgets, overcome pitfalls, maintain, evaluate and evolve a commercial Web presence. Manage service applications

Textbooks:

Pro Web Project Management (Expert's Voice in Web Development) Paperback – November 29, 2011 by Justin Emond (Author), Chris Steins (Author)

Web Project Management: Delivering Successful Commercial Web Sites Paperback – October 17, 2000

Interactive Project Management: Pixels, People, and Process (Voices That Matter) Paperback – April 22, 2012

0780323, Web Applications Usability

3 hours per week, 3 credit hours, prerequisite: 0780320

Methodology design process, Create Usable; Interfaces, traditional accessibility issues, Information architecture, website design, display the results, Requirement analysis Mock-ups/Prototypes, Conceptual design. Evaluations techniques.

Textbooks:

Designed for Use: Create Usable Interfaces for Applications and the Web Paperback – June 28, 2011

Web Application Design Handbook: Best Practices for Web-Based Software (Interactive Technologies) Paperback – July 7, 2004 by Susan Fowler (Author), Victor Stanwick (Author)

0780324, Web Services

3 hours per week, 3 credit hours, prerequisite: 0780340

Mechanism of client-server architecture; Internet architecture; development of applications for the Internet.

.NET technologies; develop web-based database applications. Web services; Universal Description, Discovery and Integration (UDDI), Simple Object Access Protocol (SOAP) and Web Services Description Language (WSDL).

Textbook
Web Services: Concepts, Architectures and Applications
Gustavo Alonso, Fabio Casati, Harumi Kuno, and Vijay Machiraju
ISBN: 3-540-44008-9

……

0780340, Web Server Side Technologies

3 hours per week, 3 credit hours, prerequisite: 780230

Web Server, Apache, Modules Server Side Include; Include, Pseudo Comment, Commands Common Gateway Interface; CGI, CGI architecture, CGI execution, CGI interface, environment variables Server-Side Scripting; (A) PHP, Embedding, language elements, example, usage of databases (B) Servlets, API, container, Lifecycle, Example (C) Java Server Pages, Example

Textbooks:

Dynamic Web Programming and HTML5Paperback – November 21, 2012 by Paul S. Wang (Author)

Web Programming And Internet Technologies: An E-Commerce Approach Paperback – February 16, 2012 by Porter Scobey (Author), Pawan Lingras (Author)

0780341, Web Client Side Technologies

3 hours per week, 3 credit hours, prerequisite: 780230

Helper PROGRAMMEs, plug-ins Java Applets; Applet, compilation, JVM, embedding, API JavaScript; Scripting languages, history, properties, embedding, language elements, boxes, events, objects, JSON, functions, AJAX

Textbooks:
Professional JavaScript for Web Developers Paperback – January 14, 2009 by Nicholas C. Zakas (Author)

Writing for Interaction: Crafting the Information Experience for Web and Software Apps Paperback – April 19, 2013 by Linda Newman Lior (Author)

……

731423, Data Mining
3 hours per week (48 hours in total), 3 credit hours, prerequisite: 731221

Teaching Method:
30 hours lectures (2 hours per week) + 10 hours seminars (1-2 hours per 2 weeks) + 5 hours tutorials (1 per 2 weeks).

Aims:
The main goal is to provide the student with an understanding of the concepts and elements of data warehousing and data mining both from a business and technology prospective, including hands-on experience with a sample of tools used in decision support environments. At the conclusion of this course the students will be able to: Explain the purpose for developing a data warehouse, including the differences between operational and decision support systems. Describe and use the dimensional modelling technique for designing a data warehouse. Describe the architecture of a data warehouse. Understand the project planning aspect of building a data warehouse. Use OLAP analysis with contemporary analysis and visualization tools. Understand and explain the purpose of data mining. Understand the knowledge discovery process. Understand several different data mining techniques such as market basket analysis, clustering, genetic algorithms, as well as which kinds of problems these techniques are applicable to.

Synopsis:
This course covers the fundamentals of data warehousing architecture and the issues involved in planning, designing, building, populating and maintaining a successful data warehouse. The course introduces students to data mining and how it relates to data warehousing. Specific topics covered include the logical design of a data warehouse, the data staging area and extract-transform-load processing, the use of multi-dimensional analysis using OLAP techniques, and coverage of the knowledge discovery process including common data mining modelling techniques. This course covers the fundamentals of data warehousing architecture and the issues involved in planning, designing, building, populating and maintaining a successful data warehouse. The course introduces students to data mining and how it relates to data warehousing. Specific topics covered include the logical design of a data warehouse, the data staging area and extract-transform-load processing, the use of multi-dimensional analysis using OLAP techniques, and coverage of the knowledge discovery process including common data mining modelling techniques.

Assessment:
Two 1-hour unit tests (20% each) + Assignments (20%) + 2-hours final exam (40%).

Textbook:
Modern Data Warehousing, Mining, and Visualization, by George M. Marakas, 2003, Prentice-Hall.

References:
The Data Warehouse Toolkit, by Ralph Kimball and M. Ross, 2002, Wiley

---

0750464, Information and Data Retrieval
Credit: 3 credit hours Prerequisite: 731221

Aims:
Information Retrieval (IR) is a really hot subject these days. All thanks to the World Wide Web, Web Search, and our friends at Google, Yahoo!, MSN, and all the other search engines that have come and gone! But, there's a lot that happens between the typing of 2-3 keywords in a small box at

the User Interface, and receiving the results. In the next ten weeks we'll look at issues surrounding information retrieval systems. We will examine information system design and evaluation issues, and look under the hood of the search engines to pick at what's going on and why.

Teaching Methods:
36 hours Lectures (2-3 hours per week) + 8 hours Seminars (1 per 2 weeks) + 4 hours Laboratory (1 per 3 week)

Synopsis:
Introduction to IR. Overview of the components of an IRS, Queries, Documents, Indexing, Theories & Models in IR (Retrieval Techniques) for text, hypermedia, web, Exploring Google search, Queries and Information Needs, Document Analysis, Structure of documents, parsing, stemming, morphological analysis, tokenization, Retrieval Techniques, Exact Match vs. Partial Match Weighted Ranked Retrieval, Vector Space & Probabilistic retrieval models, Relevance Feedback, Retrieval Techniques, Retrieval Techniques, Exact Match vs. Partial Match Weighted Ranked Retrieval, Vector Space & Probabilistic retrieval models, Relevance Feedback, Query Processing for IR ,Query Formulation, Expansion, Refinement, Web search. Hypertext, Internet Search Engines, Link Analysis, Evaluation of IRS, Performance Measures. Relevance, User centered evaluation of IR systems, Evaluation of IRS, Evaluating Exploratory Search Systems, Web search Crawlers, Web graph. Log Analysis, evaluation revisited, Faceted Search.

Mode of Assessment:
Two 1-hour midterm exams (15% each); Coursework (15%); Tutorial contribution (5%); Final Examination: written (unseen) exam (35%) + Lab Exam (15%)

Textbook and Supporting Material:
Title: Information retrieval models : foundations and relationships Author(s)/Editor(s): Poelleke, Thomas Publisher: Morgan & Claypool, 2013

0780420, e-Commerce System Engineering
3 hours per week, 3 credit hours, prerequisite: 0780323

Aims:
The course aims to teach the student the proper methodology to build highly efficient e-commerce systems.

Synopsis:
This course covers principles, techniques, architectures, and enabling technologies for the development of the different components and layers of complex e-commerce systems (presentation and personalization layer, business logic, message exchange). It discusses: (1) e-commerce transaction models, system architectures and functions, (2) enterprise applications development using J2EEE, (3) Web services and business process modelling, (4) security, transaction, payment protocols for enterprise applications, (5) e-catalogues, (6) inter-enterprise message exchange, and (6) personalization. The lecture materials will be complemented by several assignments and labs

Textbooks:
Electronic Commerce 12th Edition, Gary Schneider. 2016.

0780423, Quality Assurance and Testing of Web Applications

3 hours per week, 3 credit hours, prerequisite: 0780320

Aims:

The course aims to teach the student on proper approaches to test and produce high quality web applications.

Synopsis:

Quality assurance, testing, test, test suite, test case, test level, coverage, people, test-driven development Test Approaches; Conventional Testing, specifics of WAs, sources of defects Test Methods and Techniques; Dimensions of testing, main test techniques, functional testing, deriving test cases, unit tests, link testing, usability testing, load & stress testing, security testing, target platform testing, test automation

Textbooks:

Quality Assurance of Web Design: Web Usability Paperback – September 27, 2012 by G. Sreedhar (Author), A. Anandaraja Chari (Author)

Security Strategies In Web Applications And Social networking (Information Systems Security & Assurance) Paperback– September 8, 2010 by Mike Harwood (Author)

0780431, Web Security

3 hours per week, 3 credit hours, prerequisite: 0750444

Aspects of security, specifics of WAs
• Web security model
Browser security model including same-origin policy; Client-server trust boundaries, e.g., cannot rely on secure execution in the client
• Session management, authentication
Single sign-on; HTTPS and certificates
• Application vulnerabilities and defences
SQL injection; XSS; CSRF;
• Client-side security
• Cookies security policy; HTTP security extensions, e.g. HSTS; Plugins, extensions, and web apps; Web user tracking
• Server-side security tools, e.g. Web Application Firewalls (WAFs) and fuzzers

Textbooks:

Security for Web Services and Service-Oriented Architectures

Bertino, E., Martino, L., Paci, F., Squicciarini, A. 2010

0780470, Practical Training

3 hours per week, 3 credit hours, prerequisite: 90h

This module consists of realizing a supervised training in an industrial organization, or using distance online training.

Students may apply, in the real world, what they have learned during the first three years of their study in the University.

0780480, Project (1)

1 hours per week, 1 credit hours, prerequisite: 90h

Overview of projects and project assessment, real project definition, analysis, design, , writing, and presenting; Expected deliver: SRS (Software Requirement Specification) document, executable prototype, software architecture of their project.

Textbooks:

Projects in Computing and Information Systems: A Student's Guide (2nd Edition) Paperback – April 29, 2009 by Christian Dawson (Author)

Software Requirements (3rd Edition) (Developer Best Practices) Paperback – August 25, 2013 by Karl Wiegers (Author), Joy Beatty

Interaction Flow Modeling Language: Model-Driven UI Engineering of Web and Mobile Apps with IFML, Marco Brambilla & Piero Fraternali, Elsevier, 2014

0780481, Project (2)

2 hours per week, 2 credit hours, prerequisite: 0780480

Overview of projects and project assessment, real project definition, analysis, design, , writing, and presenting.

Textbooks:

Projects in Computing and Information Systems: A Student's Guide (2nd Edition) Paperback – April 29, 2009 by Christian (Author)

Writing for Computer Science Paperback – April 27, 2004, by Justin Zobel (Author)

3.5 Elective Modules

Each student should select 2 modules out of a list of 4 modules according to his/her interest. The Department has a list of elective modules, which can be updated according to the staff expertise and the most recent trends in the field of SE. The current list of such modules is shown in Table (3-4), where some modules are marked with (R) to indicate that these modules are research-oriented according to the staff expertise.

Table (3-4) Elective Modules in Computer Science Department

Module Number	Module Title	Prerequisites
0780344	Mobile Web Applications	0770323
0780346	Web Server Administration	0780340
0780430	Semantic Web	0780324
0780432	Special Topics in Web Engineering	90 Hrs
0780445	Cloud Computing based Development	0780324

0780344, Web Mobile Applications

3 hours per week, 3 credit hours, prerequisite: 0780320

Mobile Phones, Tablets, Web Access, Small Screen, Cross-Device Web, Mobile Web Apps, Android, iOS, Transform, Desktop Web Applications.

Textbooks:
Build Mobile Websites and Apps for Smart Devices. 1st Edition. 2011. By Earle Castledine, Myles Eftos, Max Wheeler. ISBN-13: 978-0987090843. ISBN-10: 0987090844

Oracle Application Express for Mobile Web Applications. 1st Edition. 2013. By Roel Hartman, Christian Rokitta, David Peake. ISBN-13: 978-1430249474

077346, Web Server Administration

3 hours per week, 3 credit hours, prerequisite: 0780340

Basics and Practical aspects of Web Server platforms: Web Server platform: functionality, Components. Web Server installation, configuration, Programming, maintenance, monitoring, and security. Use of current commercial Web Server platforms: Apache, IIS.

Textbooks:
Web Server Administration (Web Warrior) Steve Silva, Course technology CENGAGE Learning, 2008

0780430, Semantic Web

3 hours per week, 3 credit hours, prerequisite: 0780420

Semantic Web, aspects of the Web. Development semantic modelling, information sharing, cooperation, and collaboration, make up languages of Semantic Web RDF, RDFS, SPARQL, OWL, inferencing in the Semantic Web

Textbooks:
Semantic Web for the Working Ontologist, Second Edition: Effective Modeling in RDFS and OWL Paperback – May 20, 2011 by Dean Allemang (Author), James Hendler (Author)

The Semantic Web Explained: The Technology and Mathematics behind Web 3.0 Paperback – October 27, 2014 by Péter Szeredi (Author), Gergely Lukácsy (Author), Tamás Benkő (Author)

A Semantic Web Primer (Cooperative Information Systems) Hardcover – August 24, 2012 , by Grigoris Antoniou (Author), Paul Groth (Author), Frank van van Harmelen (Author), Rinke Hoekstra (Author)

0780432, Special Topics in Web Engineering

3 hours per week, 3 credit hours, prerequisite: 90h

Relevant and recent topics are selected each semester.

Textbooks:
Depends on the topic

0780445, Cloud Computing based Development

3 hours per week, 3 credit hours, prerequisite: 780420

Cloud Computing: concepts, architecture, current available technologies Infrastructure as a service, Business Process as a Service, Platform as a Service, Software as a Service Cloud computing based Application development and Programming

Textbooks:
1. Cloud computing - A Hands-On Approach, A. Bahga and V. Madisetti, University Press, 2014

2. Cloud Computing for Programmers: Software Development in the Age of Cloud, D. Casal, 2014, ISBN-13: 978-1484903124

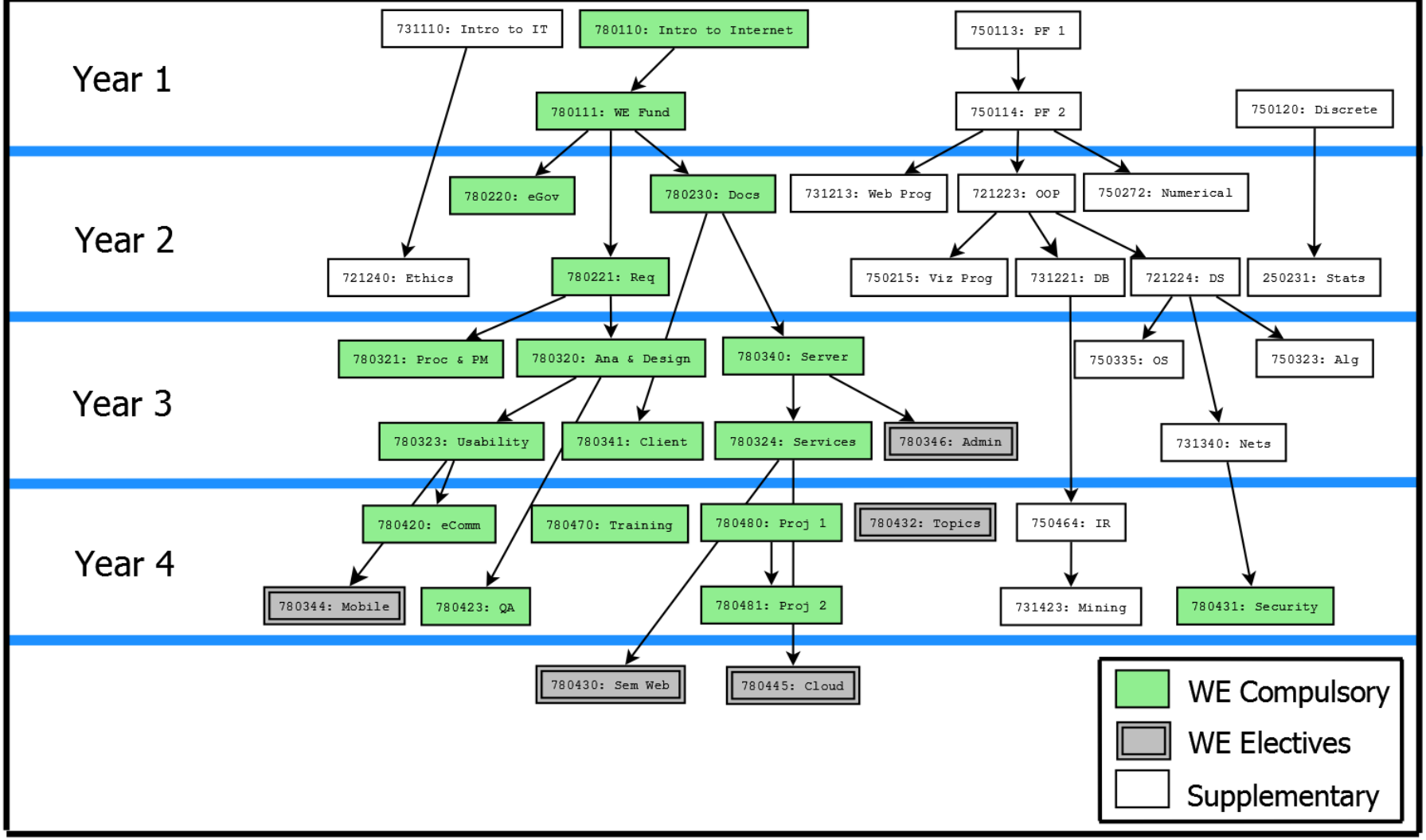
APPENDIX A

THE PREREQUISITE RELATIONSHIPS BETWEEN MODULES

Philadelphia University

The way to the future

Department of Web Engineering
Tree Plan



APPENDIX B

STUDY PLAN

OF

WEB ENGINEERING PROGRAMME

Philadelphia University
(Private Accredited University)

Faculty of Information Technology
Web Engineering Department

Study Plan for Bachelor Degree in Web Engineering (132 Credit Hours)

First: University Requirements (27 Credit Hours)

1- University Compulsory: (15 Credit Hours)

Module No.	Module Name	Credit Hours	Prereq.	Mark
0111100	Military Sciences **	3	----	
0111101	National Education	3	----	
0114101	Arabic Language Skills (1)	3	0114099	
0130101	English Language Skills (1)	3	0130099	
0130102	English Language Skills (2)	3	0130101	

** Compulsory for Jordanian students and elective for Non-Jordanians

2- University Electives: (12 credit hours)

(Each student studies (12) credit hours from the following fields one module from each field as minimum and two modules from one field as maximum)

Module No.	Module Name	Credit Hours	Prereq.	Mark
a. Humanity Sciences Field (3 - 6) Credit Hours				
0114102	Arabic Language Skills (2)	3	0114101	
0130103	English Language Skills (3)	3	0130102	
0140101	French Language Skills (1)	3	---	
0140104	Foreign Language (Italian 1)	3	---	
0140105	Foreign Language (Italian 2)	3	0140104	
0140106	Foreign Language (Hebrew 1)	3		
0140109	Chinese language Skills (1)	3		
0140110	Chinese language Skills (2)	3	0140109	
b. Social and Economical Sciences Field (3 - 6) Credit Hours				
0111111	Introduction to Sociology	3	---	
0111112	Introduction to Psychology	3	---	
0111133	Culture and Civilization (1)	3	----	
0111142	Communication and Society	3	---	
0115255	Culture of Development	3	---	
0420140	Human Rights	3	---	
0420143	Legal Culture	3	---	
c. Science, Technology, Agriculture, & Health Field (3-6) Credit Hours				
0371111	Project Management Skills	3	---	
0731101	Social Networking Skills	3	---	
0731111	Computer Skills	3	---	
0910101	Health Promotion of Individuals and the community	3	---	
0910105	Principles of Nursing and First Aid	3	---	

Second: Faculty Requirements (24 Credit Hours)

Module No.	Module Name	Credit Hours	Prereq.	Mark
0721223	Object-Oriented Programming *	3	0750114	
0721240	Computing Ethics	3	0731110	
0731110	Introduction to Information Systems and Technology	3	----	
0731213	Introduction to Web Programming	3	0750114	
0750113	Programming Fundamentals (1) *	3	----	
0750114	Programming Fundamentals (2) *	3	0750113	
0750215	Visual Programming *	3	0721223	
0780110	Introduction to Internet and Web Technology	3	----	

Third: Major Requirements (81 Credit Hours)

a- Compulsory Modules (42 Credit Hours)

Module No.	Course Name	Credit Hours	Prereq.	Mark
0780111	Web Engineering Fundamentals	3	0780110	
0780220	Fundamentals of e-Government	3	0780111	
0780221	Requirements Engineering for Web Applications	3	0780111	
0780230	Web Documents *	3	0780111	
0780320	Web System Analysis and Design	3	0780221	
0780321	Web Process and Project Management	3	0780221	
0780323	Web Applications Usability *	3	0780320	
0780324	Web Services	3	0780340	
0780340	Web Server Side Technologies *	3	0780230	
0780341	Web Client side Technologies *	3	0780230	
0780420	e-Commerce System Engineering	3	0780323	
0780423	Quality Assurance and Testing of Web Applications	3	0780320	
0780431	Web Security	3	0731340	
0780470	Practical Training	0	90 Hours	
0780480	Project (1) *	1	90 Hours	
0780481	Project (2) *	2	0780480	

***All major modules include at least 25% Practical work, Tutorial, Lab., and Assignment**

b- Supplementary Compulsary Requirements (30 Credit Hours)

Module No.	Module Name	Credit Hours	Prereq.	Mark
0250231	Introduction to Statistics and Probabilities	3	0750120	
0721224	Data Structures *	3	0721223	
0731221	Database Fundamentals *	3	0721223	
0731340	Fundamentals of Computer Networks *	3	0721224	
0731423	Data Mining	3	0750464	
0750120	Discrete Mathematics	3	0750099	
0750272	Numerical Analysis	3	0750114	
0750323	Algorithms	3	0721224	
0750335	Operating Systems	3	0721224	
0750464	Information and Data Retrieval	3	0731221	

c- Elective Modules (9 Credit Hours)

Module No.	Module Name	Credit Hours	Prereq.	Mark
0780344	Mobile Web Applications	3	0780323	
0780346	Web Server Administration	3	0780340	
0780430	Semantic Web	3	0780324	
0780432	Special Topics in Web Engineering	3	90 Hours	
0780445	Cloud Computing based Development	3	0780324	

All students must apply for level exam in Arabic and English languages and Computer skills