


QFO-AP-FI-002	اسم النموذج: خطة تدريس مادة دراسية <b>Course Syllabus</b>	جامعة فيلادلفيا  Philadelphia University
رقم الاصدار: 2 Revision 2	الجهة المصدرة: كلية تكنولوجيا المعلومات <b>Faculty of Information Technology</b>	
التاريخ: 2018/11/10	الجهة المدققة عمادة التطوير والجودة	
عدد صفحات النموذج: 6		

**Department of Web Engineering  
First Semester, 2019/2020**

<u>Course Syllabus</u>	
<b>Course Title: Web System Analysis and Design</b>	<b>Course code: 0780320</b>
<b>Course Level: 3</b>	<b>Course prerequisite (s) and/or co-prerequisite (s):</b> 0780221
<b>Lecture Time: 12:10-1:00</b>	<b>Credit hours: 3</b>

<u>Academic Staff Specifics</u>				
Name	Rank	Office Number and Location	Office Hours	E-mail Address
<b>Dr. Amro Al-Said Ahmad</b>	Assistant Professor	<b>Room 308 IT Building</b>	STT: 10-12 MW: 10-12	<a href="mailto:asaid@philadelphia.edu.jo">asaid@philadelphia.edu.jo</a>

<http://www.philadelphia.edu.jo/academics/asaid/>

**Course module description:**

This course will provide students with knowledge of the techniques, concepts, and processes to undertake the Web System Analysis and Design, once the requirements activities have been completed. Topics include the design concepts, analysis, and modeling in the context of web applications development.

**Course module objectives:**

This course aims to provide students with:

- The ability to creatively build knowledge on design for web environments using range of approaches, and to determine which approaches are suitable for the issue in hand.
- Valuable skills required for analyze, model, and design in order to develop quality web applications.

## Course/ module components

### Books (title, author (s), publisher, year of publication)

**Title:** Web Engineering: A practitioner approach

**Author:** R. S. Pressman, D. Lowe

**Editor:** Mac Graw-Hill

**Year of edition:** 2017 (India), 2009

**Title:** Web Engineering: Modelling And Implementing Web Applications.

**Author:** Daniel Schwabe, Gustavo Rossi, Luis Olsina.

**Editor:** Springer.

**Year of edition:** 2008, 2015(2<sup>nd</sup> edition)

**Title:** Web Engineering for Workflow-based Applications: Models, Systems and Methodologies.

**Author:** Freudenstein, Patrick

**Editor:** KIT Scientific Publishing

**Year of edition:** 2009

**Title:** Model-Driven Software Engineering in Practice: Second Edition.

**Author:** Marco Brambilla, Jordi Cabot, and Manuel Wimmer.

**Editor:** Morgan & Claypool Publishers

**Year of edition:** 2017

**Support material (s) (vcs, acs, etc):** Slides

### Teaching methods

*Lectures, tutorials, laboratory sessions*

Duration: 16 weeks, 48 hours in total. Lectures (+ Exams): 33 hours, Tutorials + case studies, and Lab sessions: 15 hours

### Learning outcomes

A student completing this module unit should be able to:

#### • Knowledge and understanding

1. Place the design, analysis, and modelling in the software development lifecycle. (A2)
2. Recognize the importance of the Web System Analysis and Design. (A1, A3)
3. Explain the place, importance and benefits of design in the different phases of software development lifecycle (A1).
4. Develop an object based solution from use cases to all stages of modeling, through the use of appropriate UML web-specific notations. (A2, A3)
5. Demonstrate a comprehensive labeling of the different approaches in the web systems development lifecycle.(A2)
6. Recognize and describe of web-specific quality and the impact of those quality on web systems development. (A5)
7. Evaluate critically and assess the appropriateness of in-house and outsource development, design patterns, architectures and methods for developing a design solution to a web-based business needs. (A2)

#### • Cognitive skills (thinking and analysis).

8. Build and design software models. (B2)
9. Find appropriate design models. (B2)
10. Select and use appropriate design approaches and models (B3)
11. Build a web UML models for web applications (B1)

- **Practical skills**

12. Document and report the analysis, design, and model of medium-size web applications. (C1)
13. Practice web modeling language tools (C2)
14. Practice modeling and design approaches using the right tools (C2)

- **Transferable skills**

15. Communicate effectively with non –specialist as well computer scientist (D5)
16. Work as a part of a team (D3)
17. Solve problems (D1)
18. Manage time, tasks, resources (D2)
19. Prepare and deliver coherent and structured verbal and written technical report (D4)

**Learning Outcomes Achievements:**

**Development**

are developed through lectures, and tutorials.  
are developed through projects.

**Assessment:**

are assessed by quizzes and examinations.  
are assessed by projects

**Assessment instruments**

Quizzes.  
Mini-projects  
Final examination: 40 marks

<b><u>Allocation of Marks</u></b>	
<b>Assessment Instruments</b>	<b>Mark</b>
First examination	<b>20</b>
Second examination	<b>20</b>
Final examination: 40 marks	<b>40</b>
Quizzes, Projects	<b>20</b>
<b>Total</b>	<b>100</b>

**Documentation and academic honesty**

Submit your homework covered with a sheet containing your name, number, course title and number, and type and number of the home work (e.g. tutorial, assignment, and project).

Any completed homework must be handed in to my office (room IT 602) by 15:00 on the due date. After the deadline “zero” will be awarded. You must keep a duplicate copy of your work because it may be needed while the original is being marked.

You should hand in with your assignments:

- 1- A printed listing of your test programs (if any).
- 2- A brief report to explain your findings.
- 3- Your solution of questions.

For the research report, you are required to write a report similar to a research paper. It should include:

- **Abstract:** It describes the main synopsis of your paper.
- **Introduction:** It provides background information necessary to understand the research and getting readers interested in your subject. The introduction is where you put your problem in context and is likely where the bulk of your sources will appear.

- **Methods (Algorithms and Implementation):** Describe your methods here. Summarize the algorithms generally, highlight features relevant to your project, and refer readers to your references for further details.
- **Results and Discussion (Benchmarking and Analysis):** This section is the most important part of your paper. It is here that you demonstrate the work you have accomplished on this project and explain its significance. The quality of your analysis will impact your final grade more than any other component on the paper. You should therefore plan to spend the bulk of your project time not just gathering data, but determining what it ultimately means and deciding how best to showcase these findings.
- **Conclusion:** The conclusion should give your reader the points to “take home” from your paper. It should state clearly what your results demonstrate about the problem you were tackling in the paper. It should also generalize your findings, putting them into a useful context that can be built upon. All generalizations should be supported by your data, however; the discussion should prove these points, so that when the reader gets to the conclusion, the statements are logical and seem self-evident.
- **Bibliography:** Refer to any reference that you used in your assignment. Citations in the body of the paper should refer to a bibliography at the end of the paper.

### Protection by Copyrights

1. Coursework, laboratory exercises, reports, and essays submitted for assessment must be your own work, unless in the case of group projects a joint effort is expected and is indicated as such.
2. Use of quotations or data from the work of others is entirely acceptable, and is often very valuable provided that the source of the quotation or data is given. Failure to provide a source or put quotation marks around material that is taken from elsewhere gives the appearance that the comments are ostensibly your own. When quoting word-for-word from the work of another person quotation marks or indenting (setting the quotation in from the margin) must be used and the source of the quoted material must be acknowledged.
3. Sources of quotations used should be listed in full in a bibliography at the end of your piece of work.

### Avoiding Plagiarism.

1. Unacknowledged direct copying from the work of another person, or the close paraphrasing of somebody else's work, is called plagiarism and is a serious offence, equated with cheating in examinations. This applies to copying both from other students' work and from published sources such as books, reports or journal articles.
2. Paraphrasing, when the original statement is still identifiable and has no acknowledgement, is plagiarism. A close paraphrase of another person's work must have an acknowledgement to the source. It is not acceptable for you to put together unacknowledged passages from the same or from different sources linking these together with a few words or sentences of your own and changing a few words from the original text: this is regarded as over-dependence on other sources, which is a form of plagiarism.
3. Direct quotations from an earlier piece of your own work, if not attributed, suggest that your work is original, when in fact it is not. The direct copying of one's own writings qualifies as plagiarism if the fact that the work has been or is to be presented elsewhere is not acknowledged.
4. Plagiarism is a serious offence and will always result in imposition of a penalty. In deciding upon the penalty the Department will take into account factors such as the year of study, the extent and proportion of the work that has been plagiarized, and the apparent intent of the student. The penalties that can be imposed range from a minimum of a zero mark for the work (without allowing resubmission) through caution to disciplinary measures (such as suspension or expulsion).

### Course/module academic calendar

Week	Basic and support material to be covered	Homeworks
(1)	<b>Requirement Engineering for Web applications:</b> <ul style="list-style-type: none"> <li>▪ Specifics in Web Engineering</li> <li>▪ Top distinguishing characteristics for web-applications</li> <li>▪ Specification</li> </ul>	

	<ul style="list-style-type: none"> <li>▪ Validation</li> <li>▪ Example</li> </ul> <p><b>Model-Driven Engineering for Web Engineering</b></p>	
(2)	<p><b>Modeling Web Applications (I):</b> Fundamentals of modeling Overview on UML/ WebML/Modeling tools Tutorial</p> <p><b>Modeling Web Applications (II):</b> Behavior/Interaction Model (1): class/object, data type, and class diagrams. Tutorial</p>	<b>Quiz</b>
(3)	<p><b>Modeling Web Applications (III):</b> Behavior/Interaction Model (2): associations, composition/aggregation, and inheritance. Tutorial</p> <p><b>Modeling Web Applications (IV):</b> Structural modeling (1): activity diagrams, and component diagram. Tutorial</p>	<b>Teamforming</b>
(4)	<b>Agile for Web Engineering/SCRUM, Assignment Introduction, and Team Management</b>	
(5)	<p><b>Modeling Web Applications (V):</b> Specific-web modeling (1): levels, aspects, and phases. Tutorial</p>	<b>Quiz</b>
(6) <b>First Exam</b>	<p><b>Modeling Web Applications (V):</b> Specific-web modeling (2): content, hypertext, access, and presentation modeling. Tutorial</p> <p style="text-align: center;"><i>First Exam</i></p>	
(7)	<p><b>Web Application Design (I)</b></p> <ul style="list-style-type: none"> <li>▪ Design concepts</li> <li>▪ Design goals</li> <li>▪ Workflow design</li> </ul>	
(8)	<p><b>Web Application Design (II)</b></p> <ul style="list-style-type: none"> <li>▪ Data Design</li> <li>▪ Navigation design/Interaction design</li> <li>▪ Presentation design</li> </ul>	
(9)	<p><b>Web Application Design (III)</b></p> <ul style="list-style-type: none"> <li>▪ Architecture design</li> <li>▪ Functional Design</li> </ul>	<b>Quiz3</b>
(10)	<p><b>Web Application Architecture (II)</b></p> <ul style="list-style-type: none"> <li>▪ Components of generic web applications architecture</li> <li>▪ Data/database/web document/multimedia architectures</li> </ul>	

<b>(11)</b> <b>Second Exam</b>	<b>Web Application Architecture (I)</b> <ul style="list-style-type: none"> <li>▪ Fundamentals</li> <li>▪ Developing and categorizing architecture</li> <li>▪ Specific of web application architecture</li> </ul>	
<b>(12)</b>	<b>Design Patterns (I)</b> <ul style="list-style-type: none"> <li>▪ Patterns: understanding the concept</li> <li>▪ Web application patterns</li> </ul>	
<b>(13)</b>	<b>Assignment Follow-up for Teams progress.</b>	<b>Quiz 4</b>
<b>(14)</b>	<b>Design Patterns (II)</b> <ul style="list-style-type: none"> <li>▪ Pattern repositories</li> <li>▪ Example patterns</li> </ul>	<b>Team reports handout</b>
<b>(15)</b>	<b>Team Demos</b>	<b>Individual report handout</b>
<b>(16)</b>	<b>Individual Report Presentation</b>  <b>Revision</b>	

### **Expected workload**

On average students need to spend 3 hours of study and preparation for each 50-minute lecture/tutorial.

### **Attendance policy**

Absence from lectures and/or tutorials shall not exceed 15%. Students who exceed the 15% limit without a medical or emergency excuse acceptable to and approved by the Dean of the relevant college/faculty shall not be allowed to take the final examination and shall receive a mark of zero for the course. If the excuse is approved by the Dean, the student shall be considered to have withdrawn from the course.