QFO–AP–FI–MO02	اسم النموذج: Course Syllabus	جامعة فيلادلفيا
رقم الاصدار : 1 (Revision)	الجهة المصدرة: كلية تكنولوجيا المعلومات	
التاريخ: 2017/11/05		Philadelphia University
عدد صفحات النموذج:	الجهة المدققة: عمادة التطوير والجودة	

Course Syllabus	
Course Title: *Software Process Perspectives*	Course code: 750782
Course Level: *Master*	Course prerequisite(s) and/or corequisite(s): None
Lecture Time:	Credit hours: 3

Academic Staff Specifics				
Name	Rank	Office Location	Office Hours	E-mail Address

Course Description

This course is intended for students already accepted to study the MSc in Computer Science in the Faculty of Information Technology. Every software development organization needs to be focused on the delivery of quality. The software engineering discipline responds by calling for a managed process for the construction and testing of software, and for the improvement of that process. This course presents the necessary concepts within the frameworks provided by three important international standards. The practical aspect should be emphasized.

Course Objectives

At the end of the course, students will be able to
- define and criticize the concepts of process and supporting paradigms in the context of software development;
- evaluate development activities against an accepted, standardized lifecycle model;
- recognize or define frameworks by which the delivery of software can be made the focus of an organization or project;
- determine the nature of compliance with, or location within, standards and models such as the Capability Maturity Model.

Course Components

Textbooks

Münch, J., Armbrust, O., Kowalczyk, M., Soto, M. Software Process Definition and Management. Springer, 2012.

In addition to the above, the students will be provided with handouts by the lecturer.

Teaching Methods

Duration: 15 weeks, 45 hours in total. Lectures: 15 hours, 1 per week. Tutorial (case study in classroom): 21 hours, 2 per week. Seminar: 3 (15 mn at the end of each lecture). Laboratories: 15 hours in total, 1-hour per week (free lab). Exams: 6 hours (3h for the mid and 3 for final exam). The last week is reserved to practical works examination.

Learning Outcomes

A. *Knowledge and understanding*
A1. Be prepared for some of the demands of, and skills required for, work in IT and IT-related industries
A2. Have been introduced to the skills and knowledge necessary to software processes.

B. *Intellectual skills (thinking and analysis)*
B1. Understand A wide range of principles, methodologies, and tools available to the software processes developer and practitioner. All these directions informed by research.
B2. Understand the professional and ethical responsibilities of the practicing computer professional including understanding the need for quality.
B3. Understand the application of computing in a business context

C. *Practical skills*
C1. Solve a wide range of problems related to the software design, Analysis, development, maintenance; and use.
C2. Design, analysis and implementation of different kinds of software.
C3. Plan and undertake a major individual project, and prepare and deliver coherent and structured verbal and written technical reports

D. *Transferrable skills*
D1. Have skills and knowledge necessary to undertake the project
D2. Be able to display an integrated approach to the deployment of communication skills,
D3. Use IT skills and display mature computer literacy;
D4. Strike the balance between self-reliance and seeking help when necessary in new situations, and display personal responsibility by working to multiple deadlines in complex activities.

Learning Outcomes Achievement
A2, B3, C1, and C2 are assessed by quizzes and examinations
A1, B1, B2, C1, C2, C3, D1, D2, D3, and D4 are assessed by practical and research works

Assessment Instruments

Allocation of Marks	
Assessment Instruments	**Mark**
Midterm examination	**30%**
Final Exam (written unseen exam)	**40 %**
Reports, research projects	**30%**
Total	**100%**

* Make-up exams will be offered for valid reasons only with consent of the Dean. Make-up exams may be different from regular exams in content and format.

Practical Submissions
The assignments that have work to be assessed will be given to the students in separate documents including the due date and appropriate reading material.

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. assignment, and project).
Any completed homework must be handed in the class 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:
- A brief report to explain your findings.
- Your solution of given problem

For the research report, you are required to write a report similar to a scientific 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 definition, summary of contribution, related work, and is likely where the bulk of your sources will appear.
- *Methods (Algorithms and Implementation):* Describe your methods here. Summarize the algorithms (if any) generally, highlight features relevant to your project, and refer readers to your references for further details. Information from sources must be rephrased in own words, "copy-and-paste" from documents, found for example on the Internet, is NOT allowed. It is allowed to use short quotations, or figures, from other documents,

but then the source MUST be clearly stated in the reference list (please check copy rights). Papers not fulfilling these rules will be failed.
- *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 Copyright**
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	HW
(1)	*Chapter 1. Software Engineering conventional processes overview* Prescriptive process, Evolutionary process, and Agile process	
(2)	*Chapter 1 (continued)* *Tutorials based on relevant research works papers*	PW on conventional SPs
(3)	*Chapter 2. Software Processes Modeling Concepts* Methodology concepts: Methodology, Method, Technique, and tool Coordination concepts and theory: coordination rules, knowledge and goal *Tutorials based on relevant research works papers*	
(4)	*Chapter 2 (continued)* *Tutorials based on relevant research works papers (analysis, design, coding, testing techniques)*	PW/RW on Modeling concepts
(5)	*Chapter 3. Software Process Meta modeling and Management* Software Process variability management: variability modeling and process generation Software Process Design Pattern, Framework, and Instance	
(6)	*Chapter 3 (continued).* *Tutorials based on relevant research works papers*	PW on SPMM

(7)	*Chapter 4. Software Process Evolution* Static and dynamic self adaptive software processes	
(8)	**Mid Exam**	
(9)	*Tutorials based on relevant research works papers*	
(10)	*Chapter 5. Bio-Inspired Software Processes* Genetic concepts, Bio-inspired variability modeling, bio-inspired processes modeling	Pw/RW on variability
(11)	*Tutorials based on relevant research works papers*	
(12)	*Chapter 6. Software Processes Quality*	
(13)	*Tutorials based on relevant research works papers*	
(14)	**Homework Exam**	
(15)	**Written Final Exam**	

Expected workload

On average students need to spend 2 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.

Module References (sample)

Students will be expected to give the same attention to these references as given to the Module textbook(s):

Research papers (provided in the course along with course chapters in tutorials)

Software Process modeling
1. Fadila Atil and Said Ghoul. A New Textual Description for Software Process Modeling, Journal of Information Technology JIT, 2006
2. Fadila Atil and Said Ghoul. Software Process Modelling Using Role and Coordination, Journal of Computer Science 2(4): 333-336, 2006
3. Fadila Atil and Said Ghoul. Role based Software Process Modeling, ISPS'2005, May 9-11, 2005, Algiers.
3. Fadila Atil and Said Ghoul. A New approach for software process modeling, WSEAS Transactions on Information Science and applications, 1(1), july 2004
4. Fadila Atil and Said Ghoul. Modelling software process using roles, 17th International Conference on Software & Systems Engineering and their Applications, Paris - November 30 & December 1-2, 2004
5. Said Ghoul. Object-based softwre process modellig. *Proc of 1st UK Colloquium on Object Technology & System Reengineering, COTSR'98 (UK), 1998*
6. Said Ghoul. Software Process between research and industrial reality. *Conference on Advanced Technologies of Information Processing, Philadelphia University*, Amman, July, 1997
7. Said Ghoul. Doctorate Es. Science (Post doctoral researches), in Software Engineering, Annaba University, Algeria, 1995. Topic: *Methodologies and Structures Aspects in Software Processes*.

Software Process variability modeling
1. Sumaya Mehdi Ibraheem. Feature-based Variability Modeling in Software Evolution. CS MSc thesis, Philadelphia University, 2016
2. Ola Younes Said Ghoul. Systems Versioning: A Features-Based Meta-modeling Approach. ONLINE SPECIAL JOURNAL ISSUES**.** published in International Science Index Vol:8 No:06, 2014 at www.waset.org/Publications.
3. Ola Younes Said Ghoul . Systems Variability Modeling: A Textual Model Mixing Class and Feature Concepts. International Journal of Computer Science & Information Technology (IJCSIT) Vol 5, No 5, October 2013, http://airccse.org/journal/ijcsit2013_curr.html

Software Process meta modeling

1. A. Soltan and S. Ghoul. Modeling Variability in Algorithms Design Methods - Divide and Conquer Case, International Journal of Software Engineering and Its Applications Vol. 9, No. 2 (2015), pp. 47-58. http://www.sersc.org/journals/IJSEIA/vol9_no2_2015.php

Software Process evolution modeling
1. E. Naffar and S. Ghoul. A Genetic Methodology for Object Evolution. International Journal Of Software Engineering and Its Applications, ITSEIA, Vol.8, No. 3, 2014, http://www.sersc.org/journals/IJSEIA/vol8_no3_2014.php

Bio-inspired software process modeling
1. S. Ghoul. Software Engineering Perspective: Bio-inspired Approach, The International Arab Conference on Information Technology, ACIT 2015

Website(s):

http://ecourse.philadelphia.edu.jo/login/index.php   // Prof. Said
http://www.springer.com/us/book/9783642242908   // Textbook