CSC7426 : Software and Data Engineering

J Paul Gibson, D311

paul.gibson@telecom-sudparis.eu

http://www-public.telecom-sudparis.eu/~gibson/Teaching/CSC7426/

Introduction to Software Engineering

Objectives:

Capacity to identify and describe the software life cycle, data management, roles, artefacts, and activities. Understand the concepts of software "best practices" and when they apply. Be able to adapt a software development process to ones needs and select an appropriate set of best practices that will guide you in completing a software development project.

Keywords: Discipline, Professionalism, Understanding, Fundamentals

Prerequisites: None (except basic programming experience)
Evaluation :

The final mark will be calculated as the maximum of your project work (continuous assessment) and the final exam. For the continuous assessment mark you will be graded out of 10 for each project to which you contributed and the mark will be calculated as the sum of your 2 best projects. Where a project was done in a team, you must also summarise your own personal contribution. The exam is a written paper; you must answer one question only. The paper will be marked our of 20.
Program:

To understand the true nature of software and appreciate that software should be engineered in a disciplined fashion, following professional standards.

To realise that engineering software has similarities to other engineering disciplines, but that there are aspects of engineering software which are unique to that discipline.

Place current (and future) technologies in software engineering into the context of software development techniques and tools that have appeared throughout the history of the discipline.

Provide the ‘big picture’ of software engineering so that students can progress to studying specialist techniques/tools/methods.
Material to Cover

- History of Software Engineering
- The Nature of Software: modelling and abstraction
- Software Process Lifecycle: analysis, requirements, design, implementation, testing, maintenance
- Rigour and formality: Specification, validation, verification and testing
- Software Quality and Software Process Improvement: international standards
- Project Management: roles in software development teams
- Ethics and Software Engineering as a Profession
- Data Engineering Fundamentals
Software and Data Engineering

The material will be uploaded dynamically: the teaching approach is based on PBL and much of the learning will be through interaction/group work during the assigned lecturing time. (Please check the website for updates before every lecture.)

Learning Objectives: Capacity to identify and describe the software life cycle, data management, roles, artifacts, and activities. Understand the concepts of software “best practices” and when they apply. Be able to adapt a software development process to one’s needs and select an appropriate set of best practices that will guide you in completing a software development project.

Questionnaire

Please complete the following quick questionnaire in order to help the lecturer understand your background, interests and motivation:

CSC7426 Questionnaire.

Assessment

The final mark will be calculated as the maximum of your project work (continuous assessment) and the final exam. For the continuous assessment mark you will be graded out of 10 for each project to which you contributed and the mark will be calculated as the sum of your 2 best projects. Where a project was done in a team, you must also summarise your own personal contribution. The exam is a written paper; you must answer one question only. The paper will be marked out of 20.

Books

There are no course texts but the following books have inspired some of the material included in the course and provide additional information that you may find useful.

- Fundamentals of Software Engineering, Ghezzi, Jazayeri and Mandrioli, Prentice Hall
- Software Engineering, Ian Sommerville, Addison Wesley.
- The Mythical Man-Month, Frederick P. Brooks, Jr., Addison-Wesley Longman Publishing Co.

Sessions

Sessions are a mix of problem-based learning, group project work, directed practicals, interactive lectures and traditional lectures. There is no preset format - the lecturer organises the style and content of each session depending on the needs of the class.