Software Project Management (1 – 20190925)

Mohammed Seyam, PhD
seyam@mans.edu.eg
http://people.cs.vt.edu/seyam

Classroom Policy

- Lecture time: 10:20-12:00
  - Be on time!
  - Once door is closed, DO NOT knock!
    - and, do NOT enter!!
- Mobile phones MUST be on silent/vibration mode during lecture time
  - Recommended: turn your phone completely off!
- Fostering an atmosphere of mutual respect will always be a priority for me.
- I trust you, unless you force me not to!

Classroom Policy

- Office hours
  - Monday: 12:00-1:00
  - Wednesday: 12:00-1:00
- TA and Labs
  - MS Project
  - Assignments
  - Projects

Email Policy

- Feel free to contact me via email
  - Include “Software Project Management 2019” in the subject line
  - Prompt replies can’t be guaranteed!
  - It’s hard to accommodate for same-day appointment requests!
  - Submission-day appointments…?!!

Course Resources

- Primary reference:
  - Introduction to Software Project Management, by Adolfo Villafiorita. CRC Press, 2014
- Secondary reading:
  - An Introduction to Project Management, Fifth edition, by Kathy Schwalbe, Schwalbe Publishing, 2015

Introduction to the Course

Motivations (1/3)

- Many courses in computer science and electronic engineering focus on the technical aspects and on the notations to develop good software (e.g., how to do testing; how to write requirements; how to model systems; UML; Java)

- However, in order to build good software, a well defined and managed process, which organizes the activities in an efficient and controlled way

Motivations (2/3)

- Consider the following:
  - Writing good requirements is no good if you don't have a controlled process, for instance, to accept changes and trace revisions
  - The techniques to do testing are useless if you don't have any time left to do testing (because, for instance, you underestimated the development time and are late with a release)
  - The development of a software system requires to execute, monitor, and control various activities which have little or nothing to do with writing code. Consider, for instance: training users, packaging a product, managing publicity and communication, writing user documentation

Motivations (3/3)

- If you want to deliver on time and within budget a product which has the quality properties agreed upon (be it a software or any other product), you need:
  - A process to define a schedule, a budget, and agree on the (quality) characteristics of a product
  - A list of techniques to define, agree, plan, execute, and monitor: goals, quality, time, and costs

Skills and some goals of the course

- Managing a (software development) project, thus, requires specific competences, skills, and techniques
- Some of the questions you will be able to answer at the end of this course include:
  - How do I **estimate** how long it will take to complete a task?
  - How much am I going to **charge** for a project?
  - How do I keep the team **motivated** and ensure projects are fulfilling and an occasion to learn, grow, and advance in one's career?
  - How do I deal with project **risks**?
  - How do I assess whether the project is on **time**, on **budget**, on **schedule**?
  - How do I control the **quality** of the final output?

Software Project Management

The Project Management techniques are intrinsically multidisciplinary …

… what you will learn in this course is applicable to virtually any other (engineering) domain.

There are however certain characteristics that make the management of software projects particularly interesting.

Software Project Management

- Software project management is interesting and challenging because:
  - The product is intangible
  - The product is uniquely flexible (e.g. different sizes; different constraints)
  - Many software projects are 'one-off'
  - The development process is uniquely flexible
  - Size and complexity are increasing exponentially
  - Human lives might depend on software running as expected (consider the control system of an airplane)- safety critical systems

Questions