

IS 663 Advanced System Analysis & Design – Syllabus, Spring 2017

Part I: Course and Instructor Information	
Semester:	Fall 2017
Course name:	Advanced System Analysis & Design
Course number:	IS 663
Instructor name:	Erick Sanchez Suasnabar
Course location:	CULMLECT 3
Course Meeting time:	Friday 1:00 – 3:55 PM EDT
Office hours & location:	Friday 4:15 to 5:55pm GITC 5601 Please set an appointment with your instructor in advance!
Email:	efs3@njit.edu

Part II: Course Description

1. Course description:

This course develops the skills necessary to analyze, design and manage the development of enterprise-scale information system solutions, incorporating contemporary methods and effective organizational and global project management practices. It focuses on technical business system analysis and design techniques, and covers key software engineering principles, methods and frameworks, including process models, agile and lean principles, project and risk management, estimation, requirements elicitation and analysis, modeling, system and software architecture, design patterns, and quality systems. Students will actively participate in discussions, review selected articles, participate in team exercises and collaborate on projects involving analysis and prototyping of applications addressing real-world problems and integrating current and emerging technologies.

For the latest course information go to <http://njit2.mrooms.net/>
 The information below should help you plan and organize your preparation during the semester.

2. Prerequisite courses and knowledge:

- Prerequisite course: IS 631
- Required background:
 - The students are required to have knowledge of key information systems concept, software development life cycle activities, and project management issues.
 - Good understanding of data modeling techniques and database fundamentals is expected as well.
 - Good understanding of modern trends in business and information analysis, information technology, data modeling, object-oriented principles and agility are a plus
 - Undergraduate software engineering courses provide a good foundation

3. Outcomes expected upon the completion of the course:

- Good understanding of classical and modern lifecycle models, including agile methods
- Hands on analysis and specification skills, using methods such as use cases, scenarios, and user stories
- Good understanding of project management functions, including risk analysis in global and collaborative projects
- Practical knowledge of estimation techniques
- Understanding of architecture and design activities as well as the impact and opportunities provided by modern architectures and infrastructures such as SOA and Cloud computing
- Hands on modeling skills (UML, agile UML)

- Understanding the fundamentals of Quality Assurances and quality frameworks

4. Ways that students will be assessed throughout the course:

- Team project execution and deliverables - content, mastery of methods discussed in class and creativity; team work; research and analysis skills
- Discussions - active participation and moderation of discussions; free sharing of ideas and information related to the discussion topics; systematic progress with paper reading assignments
- Individual assignments - content, understanding of methods discussed in class and their effective user or application to the assignment; research and analysis skills
- Class participation – open contribution to the discussions and exercises, sharing, collaboration
- Final exam – understanding of the course material and demonstrated effective application of the acquired knowledge and skills to solving practical problems

5. Required & Recommended texts:

- *Lecture Notes*
Lecture notes are the basic course material for this class. The notes are made available on Moodle every semester.
- *Text* *Book*
"Software Engineering," Ian Sommerville, Addison-Wesley, England, 10th Edition, 2016.
 - NJIT Bookstore or
 - [Amazon.com – http://www.amazon.com/Software-Engineering-10th-Edition-Sommerville/dp/0133943038/ref=dp_ob_image_bk](http://www.amazon.com/Software-Engineering-10th-Edition-Sommerville/dp/0133943038/ref=dp_ob_image_bk)
- *Articles and Discussion Supporting Materials*
For the list of readings check the Course Outline available on Moodle as well as Moodle Discussions forum.
- *Books Recommended for Extra Reading*
 - "The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition (2nd Edition)," Frederick P. Brooks, 1995.
 - "The Design of Design: Essays from a Computer Scientists," Frederick P. Brooks, 2010.
 - "Scaling Lean & Agile Development: Thinking and Organizational Tools for Large-Scale Scrum," Craig Larman and Bas Vodde, Pearson Education Inc, 2009.
 - "Practices for Scaling Lean & Agile Development: large, Multisite and Offshore Product Development with Large-Scale Scrum," Craig Larman and Bas Vodde, Pearson Education Inc, 2010.
 - "Agile Software Development with Scrum," Ken Schwaber and Mike Beedle, Prentice Hall, 2002.
 - Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives (2nd Edition), Rozanski, Nick, 2011.
 - "Design Patterns / Elements of Reusable Object Oriented Software," Erich Gamma, Richard helm, Ralph Johnson, and Vlissides (known as the "Gang of 4" of "GOF"), 1994.
 - "UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition)," Martin Fowler, 2008.
 - UML Documentation & White Papers: <http://www-01.ibm.com/software/rational/uml/>

6. Required software/hardware:

Free and open software; NJIT supported tools and hosting environments.

7. Web resources:

See Class information on Moodle (<http://njit2.mrooms.net/>)

8. Contacting the instructor (including turn-around time for responses)

Erick Sanchez Suasnabar, Ph.D. Candidate
 Department of Information Systems
 College of Computing Sciences
 New Jersey Institute of Technology

Email: efs3@njit.edu (please use subject – **IS 663** -- for quick response time)

Please always use as a subject "IS-663" plus any details you would like to add.

Office: NJIT, Newark - GITC 5601

Part III: Mapping Learning Outcomes to Course Assessment	
Course Learning Outcome	Measure (i.e., exam, homework, rubric)
Good understanding of classical and modern lifecycle models, including agile methods	In class and online discussions; team project
Hands on analysis and specification skills, using methods such as use cases, scenarios, user stories and Product Backlogs	Team project; final exam
Good understanding of project management functions, including risk analysis in organizational, global and collaborative projects	Team project
Practical knowledge of estimation techniques	Team project; in class exercises
Understanding of architecture and design activities as well as the impact an opportunities provided by modern architectures and infrastructures such as cloud computing and SOA	Individual assignment, based on literature analysis; in class and online discussion
Hands on modeling skills (UML, agile UML)	Final exam
Understanding the fundamentals of Quality Assurances and quality frameworks	In class and online discussion; team project

Part IV: Course Outline (Note: this course outline is preliminary and subject to change)		
Week	Lecture/Activity/Discussion	Reading (preliminary) <i>Check Moodle for additional reading every week</i>
Week 1 Sept 8	<p>Course logistics and introduction Course introduction – topics, objectives, SAD/SE state of the art, gaps, industry focus and ongoing research Project Description</p> <p>Reading: 1st reading assigned</p> <p>Forming Groups for Course Project: teams formed; discussions about topics initiated</p>	<p>Text Book: CH 1 Section 1.2: Software Engineering Ethics</p> <p>Reading 1: "Software Chronic Crisis," W. Wayt Gibbs, Scientific American, September 1994, pp. 86-95.</p> <p>Supporting reading - The Dimensions of Software Engineering Success (see Moodle for details)</p> <p>General SE information: SIGSOFT Web: http://www.sigsoft.org/ Guide to Software Engineering Body of</p>

IS 663 SYLABUS

		<p>Knowledge (SWEBOK Guide) - https://www.computer.org/web/swebok/index</p>
<p>Week 2 Sept 15</p>	<p>Process models: fundamentals</p> <p>Review of traditional process models, comparative analysis</p> <p>Project start:</p> <ol style="list-style-type: none"> 1) Teams work together to select topic and identify project's key contributions 2) Preliminary approval of project topics 3) To Do: Project proposal posted on Moodle and group member list <p>Reading 2 assigned</p>	<p>Text Book: CH 2</p> <p>Reading 2: "No Silver Bullet: Essence and Accidents of Software Engineering," Frederick P. Brooks, Jr, 1987, Brooks87.pdf</p>
<p>Week 3 Sept 22</p>	<p>Iterative – incremental process models: RUP, Agile Models</p> <p>Agile Software Development– (e.g., Scrum)</p> <p>Project:</p> <ol style="list-style-type: none"> 1) Market and competitors research <p>Reading 3 assigned</p> <p>Discussion: Evolution of software development practices; review of week's one reading (paper 1 "Software Chronic Crisis") – Is there really a crisis today?</p>	<p>Text Book: CH 2, 3</p> <p>Reading 3 RUP: A Rational Development Process, Philippe Kruchten, Crosstalk, 9 (7) July 1996, pp.11-16.</p> <p>Key Scrum Concepts: http://www.scrumalliance.org/pages/what_is_scrum</p> <p>Lean Software Development: A Tutorial, IEEE Software, Volume 29, Number 5 (September/October 2012)</p>
<p>Week 4 Sept 29</p>	<p>Requirements Engineering: Concepts, methods and standards</p> <p>Use Cases and User Stores</p> <p>Reading 4 assigned</p> <p>Discussion: "No Silver Bullet," Fred Brooks</p>	<p>Text Book: CH 4</p> <p>Reading 4: Scrum Guides: http://www.scrum.org/Scrum-Guides</p> <p>Alistair Cockburn's Use Cases Reference Page: http://alistair.cockburn.us/Use+cases</p> <p>Writing Effective Use Cases (see book extract- Chapter 1)</p> <p>UC Template</p>
<p>Week 5 Oct 6</p>	<p>Agile estimation and product Backlogs</p> <p>Project meetings- Progress review</p> <p>Discussion: RUP/SCRUM</p>	<p>Product Backlogs (PBLs) and user stories (US) tutorials and examples, Mike Cohn, Scrum Alliance: http://www.mountaingoatsoftware.com/agile/user-stories</p> <p>http://www.scrumalliance.org/community/spotlight/mike-cohn/january-2014/keeping-the-user-in-user-stories</p>
<p>Week 6 Oct 13</p>	<p>Use Cases and User Stores</p> <p>Nonfunctional requirements</p> <p>Project meetings Progress review</p>	<p>"Structuring Use Cases with Goals" & Use Case Fundamentals, Alistair Cockburn</p> <p>Text Book: CH 4</p> <p>Text Book: CH 22, 23</p>
<p>Week 7 Oct 20</p>	<p>Project Use Cases review (all teams, in class discussion)</p>	<p>"Structuring Use Cases with Goals" & Use Case Fundamentals, Alistair Cockburn</p> <p>Text Book: CH 4</p> <p>Text Book: CH 22, 23</p>

IS 663 SYLABUS

<p>Week 8 Oct 27</p>	<p>Specifying the data aspects of RA (informal and BNF)</p> <p>Project Management: Planning, Risk Management, Estimation</p> <p>Project meetings</p>	<p>SRS templates You can download the 1998 template from here: http://ieeexplore.ieee.org/document/720574/</p> <p>To download 2011 Requirements Engineering standard use any of the following direct links:</p> <p>http://ieeexplore.ieee.org/document/6146379/</p> <p>http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6146379</p>
<p>Week 9 Nov 3</p>	<p>Model Driven Engineering: Introduction to Object Oriented Analysis & Design Concepts; Static View and Diagrams</p> <p>Exercises: Simple OOA Models explained</p> <p>Project meetings</p>	<p>Text Book: CH 5, 7</p> <p>Introduction to Agile Modeling, Scott W. Ambler, 2005-2014 http://www.agilemodeling.com/essays/introductionToAM.htm</p>
<p>Week 10 Nov 10</p>	<p>Object Oriented Analysis - Dynamic View and Diagrams UML Summary</p> <p>Exercises: Modeling; CRC Cards</p> <p>Project due date!</p>	<p>Text Book: CH 5, 6, 7</p> <p>Extra Reading: Design Patterns (Dr. Dennis Mancl, DMTS, Alcatel-Lucent) – one more talk by the same presenters</p> <p>Project presentation guidelines and requirements will be provided on Moodle</p>
<p>Week 11 Nov 17</p>	<p>Course Project Presentations!</p> <p>Start of Individual Assignment 2</p>	<p>Online discussion: Current Best Practices (discussion on Moodle)</p> <p>Every student is expected to find a paper, survey or a classification discussing one or several current best practices and to provide an outline and references on Moodle. All students are expected to comment on at least 2 postings by other students.</p> <p>Forward Reading: "The Case for Cloud Computing," R. Grossman, IEEE Software, March/April, 2009. (see Moodle)</p>
<p>Week 12 Nov 22</p>	<p>Architecture and Design basics</p> <p>Trends in Software Technology: DevOps, Cloud, etc.</p> <p>Online discussion: SOA, Cloud Computing, DevOps (see Moodle)</p> <p>Discussion: Project retrospectives, team work, best current practices</p> <p>Assignment 2 due date</p>	<p>Text Book: CH 6, 7, 15, 18</p> <p>" Foundations for the Study of Software Architecture" , A. Wolf and D. Perry. Oct. 1992, ACM SIGSOFT Software Engineering Notes, 17:4 (The most cited software engineering paper for over a decade. The paper received the ACM SigSoft Retrospective Impact Award.)</p> <p>"Realizing Service-Centric Software Systems," O.Nano, A. Zisman: IEEE Software, Nov./Dec. 2007 (Vol. 24, No. 6) pp. 28-30. http://www.computer.org/portal/web/software</p> <p>"Service oriented architectures: approaches, technologies and research issues," Mike P. Papazoglou · Willem-Jan van den Heuvel, The VLDB Journal (2007) 16:389–415</p> <p>Service Oriented Architecture: Concepts, Technology and Design (BPD), Thomas Erl: http://xml.coverpages.org/ErlThomas-SOA2-Ch16-</p>

IS 663 SYLABUS

		<p>BPEL.pdf</p> <p>“Effects of Service-Oriented Architecture on Software Development Lifecycle Activities,” G. Lewis, E. Morris, S. Simanta and L. Wrage (Software Engineering Institute, USA), <i>Softw. Process Improve. Pract.</i> 2008, p. 135 – 144 (see Moodle)</p> <p>See also IEEE Computer, March 2011 (an issue dedicated to Cloud Computing)</p>
<p>Week 13 Dec 1</p>	<p>Quality and Improvement Frameworks (e.g., ISO 9000, CMMI, PSP, TSP, AIM, Six Sigma, etc.)</p> <p>Discussion/Tentative: CMMI – is it still applicable? Course wrap-up and Q&A, Exam Preparation</p>	<p>Text Book: CH 24</p> <p>Reading (Examples)</p>
<p>Week 14 Dec 8</p>	<p>Wrap Up Course</p>	

Part V: Assignment Weighting (How Your Final Grade is Being Calculated?)	
Assessment Item	Percentage of final grade
Team Project	40%
Individual Assignment	10%
Final Exam (Comprehensive, closed book)	50%
Participation Points	Accumulated separately for active participation in class discussions, Moodle-based discussions, moderation, etc.

Grade policy (approximate): A (95%-100%), B+ (85%-94.9%), B (75%-84.9%)

Part VI: Delivery Mechanism

The following delivery mechanisms will be utilized:

- Face-to-face lectures
- Moodle: <http://moodle.njit.edu>
- NJIT on iTunes U: <http://itunes.njit.edu>
- Online resources (other than iTunes):
- Other (see below):

Part VII: Plagiarism and Academic Integrity

The approved “[University Code on Academic Integrity](#)” is currently in effect for all courses. Should a student fail a course due to a violation of academic integrity, they will be assigned the grade of “XF” rather than the “F” and this designation will remain permanently on their transcript.

All students are encouraged to look over the [University Code on Academic Integrity](#) and understand this document. Students are expected to uphold the integrity of this institution by reporting any violation of academic integrity to the [Office of the Dean of Students](#). The identity of the student filing the report will be kept anonymous.

NJIT will continue to educate top tier students that are academically sound and are self-disciplined to uphold expected standards of professional integrity. **Academic dishonesty will not be tolerated at this institution.**

Part VIII: Getting Help - General

The IST Helpdesk is the central hub for all information related to computing technologies at NJIT. This includes being the first point of contact for those with computing questions or problems.

There are three ways to contact the Helpdesk:

1. Call 973-596-2900. Monday - Friday 8 am - 7 pm.
2. Go to Student Mall Room 48. Monday - Friday 8 am - 7 pm
3. Log a Help Desk Service Request online - <https://ist.njit.edu/support/contactus.php>.

Part IX: Getting Help - Moodle

In addition to the Helpdesk NJIT has a number of resources available to help you learn/use Moodle. Please be aware of the following:

1. Getting Started Using Moodle (Student Course): <http://njit.mrooms.net/course/view.php?id=6204>
2. Student Moodle Tutorials: <http://moodle.njit.edu/tutorials/students/index.php>
3. Student Moodle FAQs: <http://moodle.njit.edu/tutorials/students/faq.php>