

# 48433 Software Architecture Spring 2015

## Major Assignment

### Assignment Overview and Conduct

This is a group assignment to be completed by groups of between 4 and 6 students. It is the students' responsibility to find others with whom to form a group.

The assignment requires development and delivery of a written software architecture as described below.

The assignment is worth 25 marks.

The assignment is not expected to exceed 5000 words. It should also contain a number of architecture diagrams.

The expected number of hours to complete this assignment is 30 - 40 hours, equivalent to approximately 6 hours per week over a 6 week period.

You will be able to select a project topic from the case studies provided. If your group wants to develop an architecture for a different problem, please discuss this with the subject coordinator.

The assignment can be completed using any software architecture toolset.

### Related Assessment

Related to this assignment is a peer assessment of each of its sections. This is to be performed each week during the normal tutorial time.

**Participation in a review is required in order to receive marks for your presented work for that section of the assignment.**

The total marks for all reviews will be scaled to 10 marks.

### Subject Objectives

This assignment reflects industry practice in which software architectures are developed in teams. Professional work is normally reviewed or critiqued by your colleagues or peers since they are best qualified ;

1. Understand and describe the factors that affect the architectural context and requirements, including stakeholders and their interests, architectural qualities, business requirements and usage scenarios.
2. Develop and refine multiple views of a software system architecture, based on the conceptual, execution and implementation architecture.
3. Understand the key issues in choosing and implementing architectural patterns, including performance, testing, security, usability, maintainability and reliability qualities.
4. Reason about alternative architectural designs to satisfy a system's quality attributes.

### How/When to submit

The assignment is to be delivered, as a printable document, to a Turnitin assignment box on UTSONline by 9:00 am 9 November 2015.

## **48433 Software Architecture Spring 2015**

An assignment submitted after the due date will be considered late and will be liable to lose 20% per day of the mark gained. Assignments more than one week late will not be marked. If students can demonstrate illness or misadventure then the late penalty may be waived. The conditions for requesting an extension or Special Consideration are described in the subject outline.

### **Academic conduct**

Plagiarism will not be tolerated and is regarded as a serious breach of academic conduct. Further information about academic integrity and avoiding plagiarism can be found in the UTS Guidelines on Good Academic Practice.

As a precaution, you are required to always keep a copy of your work. If it gets misplaced or lost, there is another copy that you can readily submit.

### **Return of marked assignments**

Assignment will normally be marked and returned within 2 weeks of the submission date.

### **Teams**

All assignment work is performed in teams. Teams can consist of 4 - 6 members.

All students are expected to contribute equally to all aspects of the assignment, although within each component there can be allocation of different tasks. The allocation of individual marks will be decided by the team's consensus. Should the team fail to agree on the mark allocation, the tutor will try to find a way to distribute marks that is acceptable to the team.

It is the team's responsibility to ensure that all the tasks are done on time. In other words, the team should have adequate fall-back arrangements in cases of sickness or other types of misadventure the team members might experience during the project.

# 48433 Software Architecture Spring 2015

## Brief overview of the Software Architecture Document

The Software Architecture Document describes the design of the architecture and provides the basis on which further design and implementation work takes place. It should:

- Summarise the system's purpose
- Analyse the system context
- Describe the major stakeholders, their characteristics and their interests in this architecture.
- Describe elaborated customer needs as a set of usage and quality narratives.
- Provide a set of views (conceptual, execution and implementation views) that describe the architecture, including its structure, behaviour, implementation, constraints, and so on.
- Justify all architectural decisions. Refer to the system context, stakeholder input, results of prototypes, and so on.
- Include notes and discussion on issues that will need to be addressed as development proceeds.

The Software Architecture Document's primary purpose is both to capture the reasoning behind the architecture, and to enable further development in the Construction phase. To that end:

- Make sure that the document is no longer or more verbose than needed. Remember that the document will be read (amongst others) by impatient software developers.
- Make sure that the document and your design are realistic. Don't design some fancy infeasible architecture that the software development teams can't or won't build. Everything in the architecture must be feasible, and that feasibility must be supported by reasoning or prototypes.

# 48433 Software Architecture Spring 2015

## Major assignment assessment criteria

Group	
-------	--

Please take note of the following set of assessment criteria. Your assignment will be assessed against them.

Item	Criterion	Max mark	Mark Awarded
1	Appreciation of diverse stakeholders and their needs as documented in a set of stakeholder descriptions and usage narratives (or user stories) that cover a range of key situations.  <b>Comment</b>	5	
2	Identification of stakeholder quality requirements. Appreciation of non-functional requirements as shown in a set of quality narratives.  <b>Comment</b>	5	
3	Identification of conceptual system that provides a feasible solution to the problem.  <b>Comment</b>	5	
5	Refinement of your architecture through developing an execution architecture. Supported by observations of how the execution architecture exposed issues in the conceptual architecture and how they were resolved. Also supported by reasoning about how the execution architecture achieves stakeholder requirements and quality requirements.  <b>Comment</b>	5	
6	Description of the implementation architecture supported by justification for architectural choices and discussion of possible alternative implementations.  <b>Comment</b>	5	
	Total	25	0