

CSE-360

**Introduction to Software
Engineering Course**

Project Booklet

Spring 2012

Version 1.0

Hessam S. Sarjoughian

CSE-360 – Introduction to Software Engineering

Table of Content

1. Instructions and Guidelines for Course Project.....	3
2. Team Formation Guidelines	4
3. Preliminary Deliverable Cover Page	6
4. Team Member Peer Evaluation (Preliminary Deliverable)	7
5. Individual Progress Statement* (Preliminary Deliverable)	8
6. Preliminary Presentation: Peer Team Evaluation	9
7. Final Deliverable Cover Page	11
8. Team Member Peer Evaluation (Final Deliverable).....	12
9. Individual Progress Statement* (Final Deliverable)	13
10. Final Presentation: Peer Team Evaluation	14
11. Preliminary Report Content Evaluation	16
12. Final Report Content Evaluation	17
13. Guidelines for Preliminary Report*	18
14. Guidelines for Final Report*	19
15. Guidelines for Preparing Reports	20
16. Sample computerized teammate evaluation sheet	21

CSE-360 – Introduction to Software Engineering

1. Instructions and Guidelines for Course Project

Purpose: The aim of the course project is for a team of students to individually and collaboratively apply the course concepts, techniques, and approaches project in creating a software product. Each team consists of 4 (or if necessary 3) members. Each team must have a leader throughout the semester unless there is a justification for a change. Teams are approved by the instructor.

As in professional settings, success of a software project depends on factors including individual effort and competency as well as team's ability to communicate and cooperate, be creative, have discipline, and manage efficiently available resources and time. Hence, project's emphasis is on the analysis, design, coding, and testing that are documented as well as team activities and software prototype and demonstration.

Grading: Evaluation of all deliverables – reports, presentations, and software – is based on their correctness, completeness, and quality. The deliverables are due at the start of the class on their due dates. There is a 20% reduction for late submissions for 1 to 24 hours after the due date/time. **Individual grades are adjusted based on team member peer evaluations.**

Every project report must include a table listing contribution percentages for every team member. Each deliverable must be submitted in hardcopy and its softcopy submitted to Blackboard. It is important for the reports, software (source and compiled code necessary for testing as well as a self-contained executable software), and any supporting items to be clearly identifiable.

	Parts	Points
Preliminary deliverable	Project schedule (10) Software Requirements Specification (30) Use case diagrams (20) Class diagrams (30) Initial demo (20) User's guide (10) Risk management plan (20) Presentation (20) Table stating each team member's contribution (10) Peer evaluation* (10) Report quality (20)	200
Final deliverable	Sequence diagrams (20) State diagrams (20) Test plan (20) Test cases and results (30) Implementation (40) User's guide (10) Presentation (20) Table stating each team member's contribution (10) Peer evaluation* (10) Report quality (20)	200

* Graded individually

2. Team Formation Guidelines

Umbrella Activities (contribution from all team members is required)

- project schedule
- meeting planning
- risk management
- user's guide
- deliverables
- presentations
- demonstrations
- Review, quality management, ...

Framework Activities

Analysis:

- use-case, class, sequence diagrams
- ERD, DFD, CRC, CFD, ...

Design:

- class, sequence, state machine diagrams
- decision tables

Implementation:

- Coding, algorithms, and code organization
- development environment

Testing:

- test plan and test cases
- code coverage, test results, and evaluation

Every team has analysts, designers, coders, and testers as well as a team leader.

Preliminary Deliverable

CSE-360 – Introduction to Software Engineering

3. Preliminary Deliverable Cover Page

Title of Your Project

Submission date: ??/??/??

* Names are given as examples

Team ID	Members		Role				
	First Name	Last Name	Leader	Analyst	Designer	Coder	Tester
??	Mary	Marian			√		√
	John	Johnson		√			√

Parts	Maximum Points	Received points
Project schedule	10	
Software Requirements Specification	30	
Use case diagrams	20	
Class diagrams	30	
Initial demo	20	
User's guide	10	
Risk management plan	20	
Presentation	20	
Table stating each team member's contribution	10	
Peer evaluation*	10	
Report quality	20	
Total	200	

* Graded individually

CSE-360 – Introduction to Software Engineering

4. Team Member Peer Evaluation (Preliminary Deliverable)

Title of Your Project

Submission date: ??/ ??/ ??

Each team member can evaluate any number of his/her teammates. Should a team member decides to do so, s/he needs to use one form (given below) for each team member and submit the filled out form(s) to the TA in a sealed envelope.

Scale	Minimal (1)	Below average (3)	Average (5)	Above average (7)	High (10)
-------	-------------	-------------------	-------------	-------------------	-----------

Team # _____ Name of team member being evaluated: _____						
	1	3	5	7	10	Explanation / Comments
Participation						
Contribution						
Role fulfillment						

To be submitted to blackboard.

CSE-360 – Introduction to Software Engineering

5. Individual Progress Statement* (Preliminary Deliverable)

Title of Your Project

Submission date: ??/??/??

Team ??		Date	Reporting Period
Member Name			
Time Spent	On Meetings	Individual Tasks	
Tasks Completed			
Task in Progress			
Tasks carried over to next reporting period			
Tasks planned for next reporting period			

* Must be included in the Preliminary Deliverable.

CSE-360 – Introduction to Software Engineering

6. Preliminary Presentation: Peer Team Evaluation

Submission date: ??/??/??

Date: _____ Evaluator Name: _____ Signature: _____

Lowest score: 1, Highest score: 10 (See Sample computerized scoring sheet on Page 20)

Team ID	Comments

Final Deliverable

CSE-360 – Introduction to Software Engineering

7. Final Deliverable Cover Page

Title of Your Project

Submission date: ??/ ??/ ??

* Names are given as examples

Team ID	Members		Role				
	First Name	Last Name	Leader	Analyst	Designer	Coder	Tester
??	Mary	Marian			√		√
	John	Johnson		√			√

Parts	Maximum Points	Received Points
Sequence diagrams	20	
State diagrams	20	
Test plan	20	
Test cases (scenarios) and results	30	
Implementation	30	
User's guide	10	
Table stating each team member's contribution	10	
Peer evaluation and self-report *	10	
Report quality and conclusions	20	
Presentation, final demo, video	30	
Total	200	

* Graded individually

CSE-360 – Introduction to Software Engineering

8. Team Member Peer Evaluation (Final Deliverable)

Title of Your Project

Submission date: ??/ ??/ ??

Each team member can evaluate any number of his/her teammates. Should a team member decides to do so, s/he needs to use one form (given below) for each team member and submit the filled out form(s) to the TA in a sealed envelope.

Scale	Minimal (1)	Below average (3)	Average (5)	Above average (7)	High (10)
-------	-------------	-------------------	-------------	-------------------	-----------

Team # _____ Name of team member being evaluated: _____						
	1	3	5	7	10	Explanation / Comments
Participation						
Contribution						
Role fulfillment						

To be submitted to blackboard.

CSE-360 – Introduction to Software Engineering

9. Individual Progress Statement* (Final Deliverable)

Title of Your Project

Submission date: ??/??/??

Team ??		Date	Reporting Period
Member Name			
Time Spent	On Meetings	Individual Tasks	
Tasks Completed			
Task in Progress			
Comments and observations on course project			
Recommendations			

* Must be included in the Preliminary Deliverable.

CSE-360 – Introduction to Software Engineering

10. Final Presentation: Peer Team Evaluation

Submission date: ??/??/??

Date: _____ Evaluator Name: _____ Signature: _____

Lowest score: 1, Highest score: 10 (See sample computerized scoring sheet on Page 20)

Team ID	Comments

Report Grading and Preparation Guidelines

CSE-360 – Introduction to Software Engineering

11. Preliminary Report Content Evaluation

Project description and schedule	Choice of software engineering process model; Gantt chart content (e.g., tasks, dates, and milestones); buffer time allocation; identification of stakeholders.
Software Requirements Specification	Description of the project; requirements' categorization as must have, should have, and nice-to-have; system, sub-system, and architecture requirements; functional and non-functional requirements; analysis models (e.g., CRC, ERD, DFD).
Use-case diagrams	Use-cases and actors with their descriptions; use-case diagrams; common use-scenarios.
Class diagrams	Analysis and design models.
User's Guide	User's needs in simple terms with sketches (hand written/drawn); basic software system operations.
Initial demonstration	Demonstrating must-have user requirements.
Risk Management	Identification of risk categories.
Presentation	50% of grade is from peer evaluation. Evaluation must be objective and supported.
Table stating each team member's contribution	Description of each member's role and contribution.
Peer evaluation	Evaluations with supporting comments are submitted for all team members.
Report Quality	Organization, grammar, readability, tables, figures, listings, and completeness.

CSE-360 – Introduction to Software Engineering

12. Final Report Content Evaluation

Sequence diagrams	Models defining interactions among software components. Important input/output data exchanges and functions are specified.
State diagrams	Models defining internal dynamics of major software components. Behaviors among collaborating components may also need to be specified.
Test plan	Test plan has been consistent with requirements and other software engineering process phases.
Test cases and results	Test cases with measurements. Tests conducted and results. Discussion of results.
Implementation	Code structure and quality (package hierarchy) with sample code. Tools and programming languages, software package tree hierarchy, development environment, OS, and supporting run-time environment.
User's Guide	Focuses on user's needs in simple terms for developed UI; describes basic user operations and software responses.
Prototype demo and video	Demonstrating user requirements.
Presentation	50% is from peer evaluation. Evaluation must be objective and supported; Code of ethics is applied.
Table stating each team member's contribution	Description of each member's role and contribution. The contributions of each team member must be clearly stated as major contributor, contributor, or support.
Peer evaluation	Evaluations with supporting comments are submitted for all team members. Evaluations are cumulative.
Report Quality	Organization, grammar, readability, tables, figures, listings, and completeness.

13. Guidelines for Preliminary Report*

1. Overview

- Problem description
- Scope of the project

2. Team formation, software engineering process, and other umbrella activities

- Team organization and team member roles/responsibilities (Table stating each team member's contribution; peer evaluation)
- Users, customer, and other stakeholders
- Software engineering process - framework and umbrella activities
- Risk assessment

3. Requirements

- User and customer needs includes user interface
- Data – input and output
- Techniques and tools
- Measurement and success criteria

4. Analysis modeling

- Top-level system functionality (architecture) including modes of operation and use
- Sub-system functionality
- Detailed analysis models focusing on overall software behavior. It is necessary to have models of important parts of the overall software. Each report can have class diagrams, sequence diagrams, state machine diagrams, decision tables, event table, or data and control flow diagram.

4. Design modeling

- Top-level system functionality (architecture) including modes of operation and use
- Sub-system functionality
- Design specifications for major and high-risk parts of the software system
- Decision logics that are necessary for navigation (control) of different parts of the software system
- Design models (e.g., class and state machine diagrams) for basic parts of the software system

5. Prototype

- Parts of software system that are impractical to analyze and design without prototyping.

6. References

Appendix

- Presentation (see Guidelines for preparing reports)

*** To be submitted in hardcopy and also to Blackboard (further instructions will be made available in due course)**

14. Guidelines for Final Report*

1-6. Updated preliminary report

- Substantive changes to the preliminary report are marked with keyword (REVISED).
- New materials to the preliminary report are marked with (NEW)
 - Table stating each team member's contribution
 - Peer evaluation
 - User's Guide

7. Testing

- Test plan, tested components, test scenarios, and test results.

8. Software

- Source code, third-party APIs, Readme file, operational/execution environments.
- Sample video (4-8 minutes)

9. Conclusions

- Observations on the project with lessons learned, critiques, and recommendations.

References

Appendix

- Presentation (see Guidelines for preparing reports)

*** To be submitted in hardcopy form. Final report, software, and video are to be submitted to Blackboard (further instructions will be made available in due course)**

15. Guidelines for Preparing Reports

A report should include as much relevant information as possible with essential items included in its main body and all support materials (for example, source code) in an appendix. The report needs to be divided into two parts: main and appendix.

The following provides some suggestions for organizing the products of your software engineering project into a nice report.

Main body: For analysis, design, and testing, important pieces are distinct use case, class, and sequence diagrams, architecture, test plan and test cases, and major code snippets. These go into the main body of the report. Complete specifications (such as System Requirements Specification, class diagrams, and data) may be included in the main body, but preferable to appear in the appendix.

Gantt chart and analysis requirements also fall into the Main category. It is necessary to include the matrix of your team (team members, their responsibilities, and self/team assessment of each team member contribution in the project including active participation in group meetings). Other information such as comments, suggestions, and observations about various aspects of the team (challenges and approaches in improving team-working) can also be included in the main body.

Appendix: Pseudo code with snippets for low level components or parts of software and all source code are suitable for the appendix. Background materials and items that are not your own or are strong derived from existing materials also belong to this category.

Final report size: The main body of a report should be about 20 – 30 pages depending on font size, the size and the number of figures and tables, line spacing, etc. Note that not every figure and table should be included in the main body. The report should include a table of content, page numbers, and figure/table numbers. The appendix should be divided to parts in order to increase the readability of the report. Project report need to accurately represent your individual and team efforts.

16. Sample computerized teammate evaluation sheet

Grading: A: 5 (Excellent) B:4 (very good) C:3 (Good) D:2 (average) E:1 (Poor)

1. Problem Description(10%) 2. Design(40%) 3. Prototype(20%) 4. Presentation(20%) 5. Time Control(10%)

Entries 1-5: Team1

Entries 6-10: Team2

Entries 11-15: Team3

Entries 16-20: Team4

Entries 21-25: Team5

Entries 26-30: Team6

Entries 31-35: Team7

