# Greensheet

**CS 160: Software Engineering**
**Fall 2016, Section 01**

**San José State University**
**Department of Computer Science**

## Instructor Info

| | | |
|---|---|---|
| **Name** | **Ahmad Yazdankhah** | My name is difficult to pronounce! |
| **Office** | DH 282 | It is a shared office and usually I don't use it! I usually use MH 210. |
| **Email** | ahmad.yazdankhah@sjsu.edu | Please don't use my personal email. |
| **Website** | Under construction! | Our official educational web tools is Canvas at https://sjsu.instructure.com/ |
| **Phone** | (408) 924-5060 | Email is the best way to communicate! |
| **Office Hours** | MW 1915 – 2145 | By appointment please! |

## Class Info

| | **Section 01** |
|---|---|
| **Meeting time** | TR 7:30 – 8:45pm |
| **Classroom** | Duncan Hall 450 |
| **Course Number** | 42044 |

## General Events of Semester

| **Description** | **Day** | **Month** | **Day #** | **Comment** |
|---|---|---|---|---|
| First day of instruction | Wednesday | August | 24 | For TR classes, Thursday August 25th |
| Holiday | Monday | September | 5 | Labor day |
| Last day to drop | Tuesday | September | 6 | |
| Last day to add | Tuesday | September | 13 | |
| Daylight saving time | Sunday | November | 6 | |
| Last day to withdraw | Thursday | November | 17 | |
| Holiday | Wednesday, Thursday & Friday | November | 23, 24, 25 | Thanksgiving |
| Last day of instruction | Monday | December | 12 | For TR classes, last day is Thursday December |

For academic events of this semester, please refer to the course syllabus at the end of this greensheet.

# Course Brief Info

## Catalog Description

Software engineering principles, requirements elicitation and analysis, design, configuration management, quality control, project planning, social and ethical issues. Required team-based software development, including written requirements specification and design documentation, oral presentation, and tool use.

## Prerequisites

| | | |
|---|---|---|
| CS 146 | Advanced Data Structure and Algorithm | Grade C- or better |
| CS 151 or CMPE 135 | Object-Oriented Design | Grade C- or better |
| CS 100W | Technical Writing Workshop | Grade C or better |

The Department of Computer Science strictly enforces prerequisites. If you are not already pre-enrolled, you must come to the first class meeting and pick up an Add-Form from the instructor. If applicable, show the instructor your card that indicates you're a graduating senior. It will be the instructor's and the department decision whether or not to send you an add-code by email.

Any student who does not show up during the first two class meetings may be dropped by the instructor.

## Required Text

For this course, there won't be any official text book.

## Further Readings

Bernd Bruegge and Allen H. Dutoit, "Object-Oriented Software Engineering Using UML, Patterns, and Java," 3rd edition, Prentice Hall, ISBN-13: 978-0-13-606125-0

The references at the end of each lecture note.

# Course Detail Info

## Course Learning Outcomes (CLO)

1. **Software process:** applying the entire software development process.
2. **Requirements engineering:** eliciting software specifications and generating functional specification containing use cases and business rules.
3. **Software design:** applying appropriate architectural design patterns and explaining and defending the design decisions.
4. **Developing a web application:** implementing a browser-based user interface, server-side logic, and a database repository.
5. **Testing and validation**: understanding the software validation process and using issue-tracking tools. Creating and executing test plans.
6. **Experiencing an industry-level software product:** development process hands-on, to the extent possible within a single semester.

## Student Learning Outcomes

Upon successful completion of this course, students should be able to:

- Acquire **critical job skills** that are immediately applicable in the software industry
- Work together in a **team** 4 to 6
- Understand the challenges of **software engineering** and its **methodologies** and **processes**
- Generate a **project schedule** with achievable milestones
- Recognize people and **team management** issues
- Analyze product **requirements** and generate **use cases**
- Write **functional specifications** at a level of detail sufficient for software design
- Make appropriate **architectural** and **design choices**
- **Develop** and **document** a software design at a level of detail sufficient for implementation
- Use **version control** tools
- Automate the **build process** for a software project
- Draw **UML** package, class, and sequence diagrams
- Do oral **presentations** to explain and defend design decisions
- Formulate and execute a **test plan**
- Carry out **code reviews** in a team setting
- Create **Gantt charts** from a work breakdown structure
- Identify **task dependencies** and critical paths
- **Track issues** and measure development progress

## Technologies

Learn to work in a team to build a complete **web application** using the Model-View-Controller (MVC) architectural framework:

- Front-end browser-based user interface
    1. **HTML5**
    2. **Bootstrap**
    3. **JavaScript**
    4. **AngularJS**
- Server-side middleware logic
    5. Web and application server: **Apache Tomcat**
    6. Programming language: **Java**
    7. Web architecture: **RESTful**
- Back-end data repository
    8. Relational database: **MySQL**
    9. Database design tools: **MySQL Workbench**
    10. Database programming language: **SQL**
    11. Connection to database: **JDBC**
- Other tools
    12. Version control: **SVN**
    13. Build tools: **Maven**

This is a challenging course that will demand much of your time and effort throughout the semester.

## Procedure

We will break the class into project teams of 4 to 6 students each. Each team will experience a realistic industry-level software development project by participating in the various development activities and creating **project artifacts**, including:

- Requirements specification
- Functional specification
- Design document
- Test plan
- Project schedule

Each team will also give several **oral presentations** to the class in conjunction with the project artifacts:

- Product overview
- Design review
- Code review
- Product demo

During the oral presentations, the members of other teams will play the roles of project advisors and potential customers for the presenting team. *Class attendance is especially important during oral presentation days.*

Team members will assume various **development team member roles**, such as:

- Project lead
- Chief architect
- User interface developer
- Server logic developer
- Database developer/administrator
- Software quality assurance engineer
- Documentation writer

A key skill to learn is how to make a complete project schedule with achievable milestones. The goal of this course is not to produce a winning product in one semester (although that could be a very lucrative bonus for the team members!) but to learn team-based software engineering methodologies, processes, and tools.

## Term Project

- A major portion of the coursework will be the team projects. Each project will provide opportunities for team members to apply immediately the material taught in the lectures.
- Students will form project teams of 4 to 6 members each. Please choose your team members wisely! Once teams are formed, students may not move from one team to another.
- Each team will choose its own web application to develop, but it must fit the architectural framework and technologies described above.
- Each team must use the assigned project management tools and track its own progress.
- The final web application must be buildable from the command line using Maven script.

- At the end of the semester, all members of a team will each receive the same project score. The project grade will be determined by the overall quality of the final version of the project team's artifacts and by how well the team achieved its goals to create a successful web application.
- Each team member is personally responsible for participating and contributing to the team's work, and for understanding each part of the work for every assignment whether or not s/he worked on that part.
- Programs must be appropriately documented via javadoc comments and should adhere to the coding style posted on the CS Department web page at http://www.cs.sjsu.edu/web_mater/java_code.html.

## Homework assignments

- There would be three individual assignments.
- The rest assignments will be to create the various project artifacts during the semester. As described above, some of the assignments will include oral presentations.
- Every team turns in one copy of each artifact or gives each presentation, and all members of the team will receive the same score for each assignment.

## Postmortem report

- At the end of the semester, each student will turn in an individual postmortem report about their teammates and their experiences. The template will be provided later.

## Examinations and Evaluations

- Every week, there would be a short quiz and there would also be two midterms, and a final exam.
- All examinations will cover from the beginning of the semester.
- All examinations could be partially closed book (concepts) and partially open book (practical topics).

## Grading Information

| | |
|---|---|
| Project | 50% |
| Individual assignments | 5% |
| Quizzes | 20% |
| Midterm | 7% |
| Final | 18% |
| **Total** | **100%** |

## Nominal Grading Scale

| From | To | Grade |
|---|---|---|
| 97 | 100 | A+ |
| 92 | 96.99 | A |
| 90 | 91.99 | A- |
| 88 | 89.99 | B+ |
| 82 | 87.99 | B |
| 80 | 81.99 | B- |
| 78 | 79.99 | C+ |
| 72 | 77.99 | C |
| 70 | 71.99 | C- |
| 68 | 69.99 | D+ |
| 62 | 67.99 | D |
| 60 | 61.99 | D- |
| 0 | 59.99 | F |

To practice time management, late submissions will lose 20% of the assignment's total score and an additional 20% for each 24 hours after the due date.

## Final Grade

Your final grade can be adjusted depending upon:

- Your level and quality of participation in the class activities
- Your team members' assessments (if any) of your performance

## Workload

- Success in this course is based on the expectation that students will spend at least 6 – 10 hours per week for working on the homework, team works, and the programming assignments.
- SJSU classes are designed such that in order to be successful, it is expected that students will spend a minimum of forty-five hours for each unit of credit (normally three hours per unit per week), including preparing for class, participating in course activities, completing assignments, and so on. More details about student workload can be found in University Policy S12-3 at http://www.sjsu.edu/senate/docs/S12-3.pdf.

## Classroom Protocol

- Be on time! Coming late is disruptive.
- My classes are always interactive. So, participate in the class' activities as much as you can.
- Ask good questions and answer to the questions (in class and in the forum) as much as you can and get extra credit for them!
- Your cell phone should be in silent mode and should not be used during the lectures.
- Wireless laptop is required and it should remain closed until I inform you that it is needed for a particular activity.
- Instant messaging, e-mails, texting, tweeting, and any other kind of communications with anyone else during the exams are strictly forbidden.
- Attendance is recommended, but is not mandatory, except for exam dates.
  - NOTE that University policy F69-24 at http://www.sjsu.edu/senate/docs/F69-24.pdf states that "Students should attend all meetings of their classes, not only because they are responsible for material discussed therein, but because active participation is frequently essential to insure maximum benefit for all members of the class. Attendance per se shall not be used as a criterion for grading."

## Consent for Recording of Class and Public Sharing of Instructor's Material

- Common courtesy and professional behavior dictate that you notify someone when you are recording him/her.
- You must obtain the instructor's permission to make audio or video recordings in this class. Such permission allows the recordings to be used for your private, study purposes only.
- The recordings are the intellectual property of the instructor; you have not been given any rights to reproduce or distribute the material.

## University Policies

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' Syllabus Information web page at http://www.sjsu.edu/gup/syllabusinfo/"

# Course Schedule for TR Class Meetings

**Note1**: this is a tentative schedule and is subject to change but with fair notice.

**Note2**: Materials for reading are mentioned in the references at the end of each lecture note.

| # | Day | Topics | Readings | Homework | Exams |
|---|---|---|---|---|---|
| 1 | 08/25 | Greensheet in detail; A big picture of the course; My background; | | 1: about you | |
| 2 | 08/30 | Software engineering overview; forming teams | | 2: a genuine product | |
| 3 | 09/01 | Requirement elicitation overview | | | |
| 4 | 09/06 | Web programming basics | | | Quiz 1 |
| 5 | 09/08 | Presentation 1: product overview | | | |
| 6 | 09/13 | Design overview 1 | | 3: functional spec | Quiz 2 |
| 7 | 09/15 | Design overview 2 | | | |
| 8 | 09/20 | Managing software lifecycle overview | | | Quiz 3 |
| 9 | 09/22 | JavaScript, AngularJS, Bootstrap: overview 1 | | | |
| 10 | 09/27 | Presentation 2: design review | | | |
| 11 | 09/29 | Presentation 2: design review | | | |
| 12 | 10/04 | JavaScript, AngularJS, Bootstrap: overview 2 | | 4: technical spec | Quiz 4 |
| 13 | 10/06 | JavaScript, AngularJS, Bootstrap: overview 3 | | | |
| 14 | 10/11 | Project management overview | | | Quiz 5 |
| 15 | 10/13 | Version control 1 | | | |
| 16 | 10/18 | Midterm | | | Midterm |
| 17 | 10/20 | Midterm solution; Version control 2 | | | |
| 18 | 10/25 | Maven overview | | | Quiz 6 |
| 19 | 10/27 | Test management overview | | 5: test plan | |
| 20 | 11/01 | Advanced web programming topics | | | Quiz 7 |
| 21 | 11/03 | Project management practice: Team meeting | | | |
| 22 | 11/08 | Presentation 3: code review | | | |
| 23 | 11/10 | Presentation 3: code review | | | |
| 24 | 11/15 | Advanced managing software lifecycle | | | Quiz 8 |
| 25 | 11/17 | Advanced project management | | | |
| 26 | 11/22 | Advanced test management topics | | 6: user manual + postmortem | Quiz 9 |
| 27 | 11/24 | Holiday: Thanksgiving | - | - | - |
| 28 | 11/29 | Project management practice: Team meeting | | | Quiz 10 |
| 29 | 12/01 | Presentation 4: product demo | | | |
| 30 | 12/06 | Presentation 4: product demo | | | |
| 31 | 12/08 | Wrapping up the semester; where we are and what would be the next step? Review for final | | Due for all project's artifacts | |

| **Final exam** | **Section 01 (TR 7:30 – 8:45pm)** |
|---|---|
| **Date and Time** | Thursday, December 15 @ 7:45pm – 10:00pm |
| **Venue** | Duncan Hall 450 |