CS 319 Object-Oriented Software Engineering

0 – Introduction

Eray Tüzün, Phd

e-mail: eraytuzun@cs.bilkent.edu.tr
About Me

Education

- Bilkent University, Computer Science, B.S.
- CASE Western Reserve University, Computer Science, M.S.
- Middle East Technical University, Information Systems, PhD

Research interests

- Software Productivity / Software Analytics
- Application Lifecycle Management
- Agile Methodologies (Scrum)
- Software Product Line Engineering

Work

- HHMI
- Microsoft
- HAVELSAN

@tuzuneray
@eraytuzun
About You

- https://pollev.com/erayt350
“It is not the strongest of the species that survive, nor the most intelligent, but the one most responsive to change.

Charles Darwin
My assumptions and expectations

Proficient in any programming language, but you have limited experience in analysis or design of a system

CS 201

Wants to be a Software Engineer /wants to learn

Internship experience? (poll)

For Team project
  Being a committed team member
Feedback / Class Participation

Share your thoughts
Ask questions – wave your hand forcefully – (just unmute your mic and talk) to get my attention
If there is something you do not understand -- ASK!
    There are no stupid questions!!!!
If you have a relevant comment, experience, anecdote -- SPEAK!
    Participation will make the class better!!!!
During Lecture...

- Please open up your webcams
- Try to concentrate
- Participate (Ask questions / Interact)
- No audio & video recording
Regulations due to Pandemic

• As per university regulations, masks are compulsory
• Keep social distancing whenever possible
• Do not approach the dais before/after class to ask questions
  – After classes, I will be available outside to take questions
  – All out-of-classroom meetings through Zoom only
• Failure to wear masks or violating regulations any other way will result in being removed from the class and additional disciplinary action
• Major Objective: keeping everyone healthy during the semester
Logistics

- Github (Code)
- Google docs (Documentation)
- Visual Paradigm (UML models)
- Zoom (for classes and term project group meetings)
- Polleverywhere (for polls)
- Slack for peer learning and group formation
  - Make sure you are already logged in ...
Your Expectations

What are your expectations of the course?
What do you want to learn?

...

Please go to https://pollev.com/erayt350
And express your opinion...
Granularity of Software

Trivial: <1 month, 1 programmer, 500 LOC
   Ex: Intro programming assignments

Very small: <3 months, 1 programmer, 2000 LOC,
   Ex: Course project

Small: <1 year, 3 programmers, 50K LOC,
   Ex: Mobile App

Medium: 3 years, 10s of programmers, 100K LOC
   Ex: Optimizing compiler

Large: 5 years, 100s of programmers, 1M LOC,
   Ex: MS Word, Excel, Linux, Windows

Very large: 10 years, 1000s of programmers, 10M LOC
   Ex: Air traffic control, Telecommunications, space shuttle
Programming != Software Engineering
Analogy with Bridge building

- Over a stream – easy, one person job
- Over river nile ... ? (The techniques do not scale)
Large-Scale Software-Engineering

Project involves a team of people – need to manage process, people and artefacts

System takes a long-time to build – need to plan

Systems are complex – need powerful tools, methods and technologies

Need to reuse code/designs/process

- > 300 Engineers
- > 10.000.000 LOC
- > 10 years!

- > $100.000.000
- > 10.000.000 LOC
Effort, Software Size, & Complexity
Course Objectives

• Learn basics of the software engineering (SE) process life cycle.

• Learn what the object-oriented (OO) approach to software development is, through OO principles and design patterns.

• Learn UML (Unified Modeling Language) that is part of most CASE (Computer Aided Software Engineering) tools and the benefits of visual modelling / diagramming.

• Practice the application of principles of object-oriented software development through the course group project.

• Develop teamwork and communication skills through the course group project.
What will you really gain from this course?

YourCV ++
- UML
- Git
- Design Patterns
- Analytical Thinking
- Requirements Analysis and Design
- Visual Paradigm
- Slack

Programming vs Software Engineering (Programmer - -> Software Engineer)

Building software from scratch

CV writing tips

Technical presentation skills
Outline

Intro to SE (Chapter 1)
Modeling w/ UML (Chapter 2)
Project Organization and Communication (Chapter 3 Sections 3.1 - 3.3)
Requirements Elicitation (Chapter 4)
Requirements Analysis (Chapter 5)
System Design (Chapters 6 & 7)
Object Design (Chapters 8 & 9)
Mapping Models to Code (Chapter 10)
Testing (Chapter 11)
New Activities (Responding to change)

- CV Development workshop
- Git Tutorial and Lab
- Design Patterns Tutorial Lab
- Slack channel to ease up team-formation and peer learning
- Peer Review of other groups

Please provide feedback throughout the semester related to teaching style
Contact Information

Eray Tüzün

– email: eraytuzun@cs.bilkent.edu.tr
– Office hours: by arrangement (Zoom)
– Use Slack channels first!
– Talk to your TA first!
– Ask me any questions in the class / after the class

Class webpage:
http://www.cs.bilkent.edu.tr/~eraytuzun/teaching/cs319/
## Contact Information (TAs)

<table>
<thead>
<tr>
<th>TA</th>
<th>Topics</th>
<th>Office Hours</th>
<th>Contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cevat Aykan Sevinç</td>
<td>Mentorship</td>
<td>-</td>
<td>DM on Slack or <a href="mailto:cevataykansevinc@gmail.com">cevataykansevinc@gmail.com</a>, GitHub</td>
</tr>
<tr>
<td>Elgun Jabrayilzade</td>
<td>Git Lab, Design Reports, Slack Accounts, Implementation</td>
<td>EA 527 by appointment</td>
<td>DM on Slack or <a href="mailto:elgun@bilkent.edu.tr">elgun@bilkent.edu.tr</a>, GitHub</td>
</tr>
<tr>
<td>Emre Sülün</td>
<td>Design Patterns Lab</td>
<td>EA 527 by appointment</td>
<td>DM on Slack or <a href="mailto:emre.sulun@bilkent.edu.tr">emre.sulun@bilkent.edu.tr</a>, GitHub</td>
</tr>
<tr>
<td>Erdem Tuna</td>
<td>Git Lab, Requirements Reports, User Manuals, Visual Paradigm Accounts</td>
<td>EA 527 by appointment</td>
<td>DM on Slack or <a href="mailto:erdem.tuna@bilkent.edu.tr">erdem.tuna@bilkent.edu.tr</a>, GitHub</td>
</tr>
<tr>
<td>Muhammad Umair Ahmed</td>
<td>Git Lab, Requirements Reports, Final Reports, User Manuals</td>
<td>EA 527 by appointment</td>
<td>DM on Slack or <a href="mailto:umair.ahmed@bilkent.edu.tr">umair.ahmed@bilkent.edu.tr</a>, GitHub</td>
</tr>
</tbody>
</table>
## Weekly Schedule

<table>
<thead>
<tr>
<th>Hours</th>
<th>Monday</th>
<th>Tuesday</th>
<th>Wednesday</th>
<th>Thursday</th>
<th>Friday</th>
</tr>
</thead>
<tbody>
<tr>
<td>08:30 - 09:20</td>
<td>CS 319 - 001 B-204 Face-to-face Lecture (Spare Hour)</td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
<td>CS 319 - 003 B-204 Face-to-face Lecture</td>
<td>CS 319 - 003 B-204 Face-to-face Lecture</td>
<td>CS 319 - 003 B-204 Face-to-face Lecture</td>
</tr>
<tr>
<td>09:30 - 10:20</td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
<td></td>
<td>CS 319 - 003 B-204 Face-to-face Lecture</td>
<td></td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
</tr>
<tr>
<td>10:30 - 11:20</td>
<td></td>
<td></td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
<td></td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
</tr>
<tr>
<td>11:30 - 12:20</td>
<td></td>
<td></td>
<td></td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
<td></td>
</tr>
<tr>
<td>12:30 - 13:20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
</tr>
<tr>
<td>13:30 - 14:20</td>
<td></td>
<td>CS 319 - 003 B-204 Face-to-face Lecture (Spare Hour)</td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
<td></td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
</tr>
<tr>
<td>14:30 - 15:20</td>
<td></td>
<td>CS 319 - 003 B-204 Face-to-face Lecture</td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
<td></td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
</tr>
<tr>
<td>15:30 - 16:20</td>
<td>CS 590 - 001 EA-502 Online Lecture</td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
</tr>
<tr>
<td>16:30 - 17:20</td>
<td>CS 590 - 001 EA-502 Online Lecture</td>
<td>CS 319 - 002 B-204 Face-to-face Lecture (Spare Hour)</td>
<td></td>
<td>CS 319 - 002 B-204 Face-to-face Lecture</td>
<td>CS 319 - 001 B-204 Face-to-face Lecture</td>
</tr>
</tbody>
</table>
Textbooks


Recommended - Textbook: Developing Software with UML, Object-Oriented Analysis and D, Bernd Oestereich, 1999, Addison-Wesley

Recommended - Textbook: Object-Oriented Analysis and Design with Applications, 2nd e, G. Booch, 1994, Benjamin/Cummings


Required - Textbook: Object-Oriented Software Engineering, Using UML, Patterns, and Java, Bernd Bruegge and Allen H. Dutoit, 2010/3rd, Pearson
Grading (Tentative)

Grading Criteria:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attendance/Quiz/Assignment</td>
<td>20</td>
</tr>
<tr>
<td>Project</td>
<td>40</td>
</tr>
<tr>
<td>Midterm [closed book &amp; notes] (TBD)</td>
<td>15</td>
</tr>
<tr>
<td>Final [closed book &amp; notes] (TBD)</td>
<td>25</td>
</tr>
</tbody>
</table>

Those students who fail to get a minimum of 30 (out of 75) points from the weighted average of the total grades (attendance/quiz/assignment, project, midterm exam) before the final exam will get the grade FZ. For instance, A/Q/A: 5/10, P: 20/100, M: 40/100 (0.2 * 50 + 0.4 * 20 + 0.15 * 40 = 24) fails, whereas, A/Q/A: 8/10, P: 30/100, M: 40/100 (0.2 * 80 + 0.4 * 30 + 0.15 * 40 = 34) will take the final exam.

* 20% of the course grade will be based on virtual labs (e.g. Design pattern, Git Lab ), and possible activities given during online lecture hours.
All individual assignments must represent your own work.

No collaboration is permitted during the quizzes, the final examination, the individual labs, and the assignments. Collaboration among team members is permitted for the term project.

Plagiarism is to take and use as one’s own, or copy without acknowledgement, the works of another person. The provider of such material can be ruled equally culpable.
Term Project – 2020

* 5 teams per each game – First come first serve
Term Project 2021

• Classroom helper
  – Group formation
  – Peer review
    • Term project team member assessment
    • Reviewing the other’s group work
  – You are expected to extend the requirements after the requirement analysis
  – Best group’s software will be used next semester.

• Tools
  – Documentation: Google docs
  – Source control: Github
Term Project 2022

• Pandemy Manager for University
  – Managing HES codes/ Vaccine & PCR status
  – Keeping track of quarantine status of students
  – ....
  – You are expected to extend the requirements after the requirement analysis

• Tools
  – Documentation: Google docs
  – Source control: Github
Term Project 2022

• Student Club Manager
  – Managing activities
  – Managing users
  – Announcements, events ...
  – ...
  – You are expected to extend the requirements after the requirement analysis

• Tools
  – Documentation: Google docs
  – Source control: Github
Term Project

Every Group Project

Does 99% of the work

Says he's going to help but he's not

Has no idea what's going on the whole time

Disappears at the very beginning and doesn't show up again till the very end

In school you have ever done
## Tentative Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sep 22 - Sep 25</td>
<td>Course Overview</td>
</tr>
<tr>
<td>Sept 27 - Oct 1</td>
<td>Chapter 1, Group Formation (via PeerPanda.net)</td>
</tr>
<tr>
<td>Oct 4 - Oct 8</td>
<td>Chapter 1, Chapter 2, CV Workshop</td>
</tr>
<tr>
<td>Oct 11 - Oct 15</td>
<td>Chapter 2, Git (Tutorial), 1 page Project Scope</td>
</tr>
<tr>
<td>Oct 18 - Oct 22</td>
<td>Chapter 3, Git (Lab)</td>
</tr>
<tr>
<td>Oct 25 - Oct 29</td>
<td>Chapter 4, Iteration 1 Analysis Report (to TA and peers)</td>
</tr>
<tr>
<td>Nov 1 - Nov 5</td>
<td>Chapter 4, Peer Reviews (Iteration 1)</td>
</tr>
<tr>
<td>Nov 8 - Nov 12</td>
<td>Chapter 5, Chapter 6</td>
</tr>
<tr>
<td>Nov 15 - Nov 19</td>
<td>Chapter 6, Midterm</td>
</tr>
<tr>
<td>Nov 22 - Nov 26</td>
<td>Chapter 7, Iteration 1 Design Report (to TA and peers)</td>
</tr>
<tr>
<td>Nov 29 - Dec 3</td>
<td>Peer Reviews, Iteration 1 Demos</td>
</tr>
<tr>
<td>Dec 6 - Dec 10</td>
<td>Chapter 8, Design Patterns (Tutorial)</td>
</tr>
<tr>
<td>Dec 13 - Dec 17</td>
<td>Chapter 9, Design Patterns (Lab), Iteration 2 Analysis and Design Reports</td>
</tr>
<tr>
<td>Dec 20 - Dec 24</td>
<td>Chapter 10</td>
</tr>
<tr>
<td>Dec 27 - Dec 31</td>
<td>Final Report, Iteration 2 Demos</td>
</tr>
</tbody>
</table>
Online Activity (Combined for three sections)

• Peer Review and Team Formation Software – Spring 2021 Peer Panda Members
• How to Survive CS 319 – Cevat Aykan Sevinc
• Options
  – Monday 19-20:30