Construction of User Interfaces (SE/ComS 319)

Ali Jannesari

Department of Computer Science

Iowa State University, Spring 2022

ADMINISTRATION AND CLASS INFORMATION

Outline

- Administrative stuff
  - Who I am, TAs, general information, etc.
  - Course organization and syllabus
  - Schedule and structure of this class

About me

- Instructor: Ali Jannesari (Assist. Prof. at CS department)
- My background: PhD at KIT (Germany), RWTH Aachen, Bosch Research Center, TU Darmstadt, UC Berkeley and ISU
- Research: Software engineering, systems, parallelism, deep learning/machine learning.
- Teaching: I enjoy interactive classes and discussions
- Contact me
  - jannesar@iastate.edu
  - Location: https://iastate.webex.com/meet/jannesar
  - Office hours: Tue 4:00-5:00PM (online)
  - Emails to me regarding the class must include "319" in subject line.

TAs

- TAs for Coms 319:
  - Waqwoya Abebe, wmabebe@iastate.edu
  - Quazi Mahmud mahmud@iastate.edu
  - Arushi Sharma arushi17@iastate.edu
  - Arijit Bhattacharjee arbhatt9@iastate.edu
- TAs office hours
  - Mon, Wed 12:00 - 1:00pm starting from week 2

Computer Science Help Center

- Help Room for your Homework:

Computer Science Help Center

Hours: January 24-May 6, 2022

MWF 11:00a-6:00p and TR 5:00-10:00p

Courses Covered:
Com S 127, 227, 228,
230, 311, 319, 321, 327, 331, 342

Visit the Virtual Help Room on Piazza

piazza.com/iastate/spring2022/comshelproomtutoring

Questions? E-mail cs_helproom@iastate.edu

Visit the In Person Help Room in 1200 Communications

Need help with your Computer Science homework?
Worried about an upcoming assignment?
Need help to develop useful study skills?

Your background?

- How good are you in programming?
  - Java, C/C++, Python, Web programming: HTML, JavaScript, PHP, MySQL?
- How many programs have you written?
  - Which programing languages do you use?, which IDEs?
  - How big are your programs? 100 LOC? 1k? 5k?
- How familiar are you with OO and UML?
- Are you familiar with agile software development methods (e.g. XP, Scrum, etc.)?
- Internships?
- What are your goals after graduation ?
- How will this course help you achieve your goals?

General information

- https://canvas.iastate.edu/courses/89949
- Gives you access to the course material, etc.
- Class schedule
  - Lecture: **Synchronous Online**
    - Monday, Wednesday, Friday.
      - Section 2: 9:55 -10:45 am
      - Section 1: 11:00 - 11:50 am
  - Lab activities and project meetings:
    - Friday classes (**First session: 1/28**)
    - In case you don't have a laptop, please contact SSG (IT Support) to borrow a laptop for this term.
- Email communication must start with "319:" in the subject line
- Prerequisite:
  - COM S 228 (Pre-Req Waiver Form)
  - Knowledge of programming (Java)

Course description

- Overview of user interface design. Evaluation and testing of user interfaces. Review of principles of object orientation, object-oriented design and analysis using UML in the context of user interface design. Design of windows, menus and commands. Developing Web and Windows-based user-interfaces. Event-driven programming. Introduction to Frameworks and APIs for the construction of user interfaces.

Course learning objectives

- Be familiar with web user Interfaces and event-driven programming (client/server, JavaScript, Node.js, frameworks and APIs.)
- Be familiar with user interface design, web and windows-based user-interfaces
- Be familiar with OO analysis and UML
- Be familiar with software design principles and architectural styles for UI applications.
- Be introduced to test-driven development and software testing

Course overview (tentative schedule)

Week (Dates)	Lecture	Homework	Lab (Friday lecture hours)
Week 01 **17 Jan - 21 Jan**	Administration & Class Information Introduction Basics, process vs. threads, client/server programs: no java threads	Formation of Teams (teams preassigned) Homework 1: HTML	No lab
Week 02 **24 Jan - 28 Jan**	Design of User Interfaces	Team Project Proposal along with wireframe	Lab Activity 1 – Prototyping using AdobeXD
Week 03 **31 Jan - 04 Feb**	Web programming using JavaScript	Homework 2: JavaScript	LAB Activity 2 – JavaScript
Week 04 **07 Feb - 11 Feb**	Web programming using JavaScript		LAB Activity 3 – JavaScript
Week 05 **14 Feb - 18 Feb**	Event-driven programming, user interfaces	Homework 3: Node.js	LAB Activity 4 – Node.js
Week 06 **21 Feb - 25 Feb**	Event-driven programming, Node.js		LAB Activity 5 – Setting up Web Server & MySQL on VM, Node.js with MySQL
Week 07 **28 Feb - 04 Mar**	Testing User Interfaces	Homework 4: UI testing	LAB Activity 6 – Web Unit testing, Web UI testing
Week 08 **07 Mar - 11 Mar**	Introduction to Test-Driven Development (TDD)	Team project proposal high fidelity prototype using AdobeXD	LAB Activity 7 – React framework
Week 09 **14 Mar - 18 Mar**	**SPRING BREAK**		
Week 10 **21 Mar - 25 Mar**	Review of Object Orientation System modeling and UML	Homework 5: React	LAB Group Activity 01 - Setting up 319 Team Project
Week 11 **28 Mar – 01 Mar**	Frameworks and APIs for user interfaces		LAB Group Activity 02 - Acceptance Testing for 1st Iteration
Week 12 **04 Apr - 08 Apr**	Data Visualization		LAB Group Activity 03 – Acceptance Testing for 1st Release & 1st Demo
Week 13 **11 Apr - 15 Apr**	User Experience		LAB Group Activity 04 – Acceptance Testing for 2nd Iteration
Week 14 **18 Apr – 22 Apr**	Additional Topics: Architectural styles for UI applications		LAB Activity 05 – Acceptance Testing for 2nd Release & 2nd Demo
Week 15 **25 Apr - 29 Apr**	Additional Topics		LAB Group Activity 06 – Acceptance Testing: Final Release & Final Demo
Week 16 **02 May - 06 May**	Additional Topics		Report & Video of Top 6 Teams Voting for Top project
Week 17 **09 May - 13 May**	**FINAL EXAM**		

Course outcome (ABET outcome)

- **1.** An ability to analyze a complex computing problem, and to apply principles of computing and other relevant disciplines to identify solutions.
- **2.** An ability to design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
- **6.** An ability to apply **computer science theory** and **software development fundamentals** to produce computing-based solutions.

What you need to do?

- Lab activities **10%**
- Quizzes **5%**
- Homework assignments **25%**
- A team project (portfolio) **40%**
  - 5% proposal; 10% final presentation/demonstration; 20% project content
  - 5% attendance and project meetings with TA and team
- Final exam **20%**

In case you are going to drop this course, do it soon please.

Quizzes

- ~10 minutes @ every few weeks (Web-based using Canvas during lecture)
- Close-book, close-note
- Cover lectures and lab activities
- 1 point for each quiz, 5% of your overall grades!

Assignments

- Homework assignments
  - Problems for you to do (individual)
  - Code and explanations (comment your code)
  - **Mandatory**
  - TBA (including report format and submission check list)
  - Selected solutions for code walk, explanation and demonstration during lab sessions (TA hours)

Group project (Portfolio)

- Do it in team (team size later!)
- Grading of group project (**40% of your grade**):
  - Individual performance assessed
- We will look for all of the below:
  - Evidence of vigorous interactions with materials (questions, insights)
  - Exploration of new and complex issues (examples, explanations)
  - Evidence of working at higher levels of blooms taxonomy: analysis, evaluation, synthesis.
  - Evidence of teamwork
- These requirements will be explained later as well.
- **Build your team as early as possible!**

Final Exam

- Final exam
  - During the **exam week**
  - 20% of your final grades

- Exams will cover material from class, labs, assignments and projects

Miscellaneous

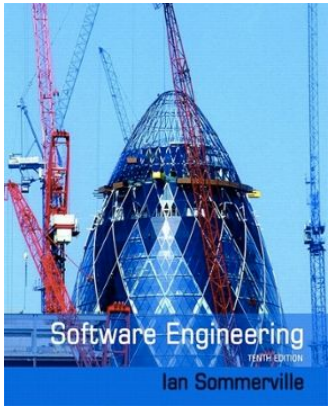
- A lecture day to be assigned for lab activities @lecture room
  - Lab activities and project meetings: Most **Friday classes (TBA);** 10% of your grade!
  - In case you don’t have a laptop, please contact SSG (IT Support) to borrow a laptop for this term
- Start early, look for online materials, tutorials on web programming, JavaScript, Node.js, and Test-driven development
- In case you copy/paste any code available in public domain you need to cite it in your source code!
- Slides will be available in PDF on Canvas
  - Only for the purpose of this class. **Redistribution not permitted!**

Philosophy

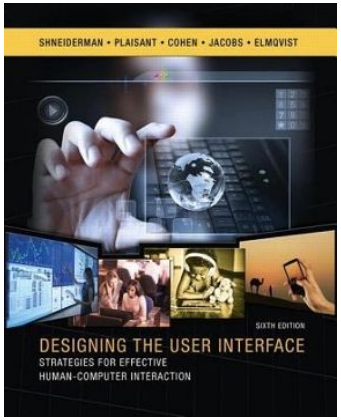
- Interactive style – physical presence strongly recommended but not monitored
- No full coverage of programming standard – rather in-depth study of key concepts
- Sound track not always mirrored on slides – please take notes or rely on books for reference

Literature & Textbooks

- Most of the materials will be based on free online textbooks/tutorials/articles.
- Optional textbooks for your references:
  - Software Engineering, Ian Sommerville, Pearson; 10th edition.



  - Designing the User Interface: Strategies for Effective Human-Computer Interaction, Shneiderman, 6th edition.



Questions?

- Thank you!