Software Design & Architecture

Introduction

Pengyu Nie

Acknowledgements: slides adapted from previous versions by Mei Nagappan and Shane McIntosh, which are adapted from previous versions by Zhen Ming Jiang, Ahmed E. Hassan, Reid Holmes.

Agenda

- Administrative details
- Expectations
- Assessment & Project

Administrative Details

Key Information Source

- Course website: https://pengyunie.github.io/cs446-1251/
  - Lecture slides
  - Project milestone requirements
- Learn: https://learn.uwaterloo.ca/d2l/home/1097952
  - Announcements (should also sent via email)
  - Project submissions
  - Grades
- Piazza: https://piazza.com/uwaterloo.ca/winter2025/cs446ece452cs646
  - Q&A
  - Access code announced in Learn

Scavenger Hunt

- Course website: https://pengyunie.github.io/cs446-1251/
- Search the course webpage to answer these questions:
  - How many members on each project team?
  - Does each team member get the same project grade?
  - If not, how is the grade weighted for each team member?
  - Is there a mid-term exam?
  - Is there a final exam?
- Work in groups with students seating next to you

Dates and Times

- Lectures: 10am-11:20am | 1pm-2:20pm, Mon & Wed
- Classroom: MC 1056 | QNC 2502
- Check the syllabus for lecture topics and project deadlines
  - Report deadlines are usually on Fridays at 11:59pm Eastern Time
  - Demos/presentations are in class
- Announcements via Learn (archived) and emails (as long as you didn't filter emails from Learn)
  - Reminders of project deadlines
  - Any change in lecture topics / project deadlines

Course Staff

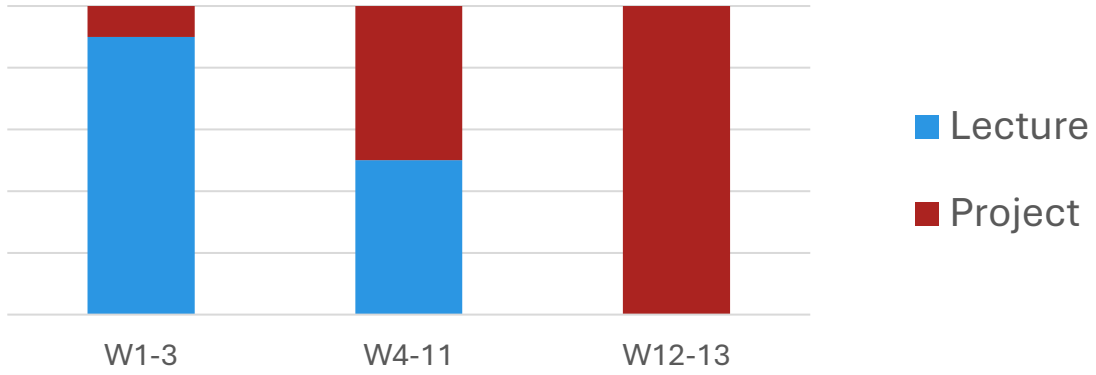
- TA: Saarang Agarwal <saarang.agarwal@>
- TA: Bihui Jin <bihui.jin@>
- TA: Daniel Phan <daniel.phan@>
- TA: Amber Wang <jiale.wang@>
  - TAs are reachable by email or Piazza
  - Each project team will be assigned with one TA throughout the term
- Instructor: Pengyu Nie <pynie@>
  - Reachable in class, by email or Piazza
  - Additional office hours by appointment

Communication Rules

- Piazza
  - Technical questions, logistic questions, generic project-related questions
  - Use folders to categorize the question
  - Feel free to try answering others' questions
  - First two weeks: post to find teammates
- Email
  - More private questions (e.g., personal or specific to your project team)
  - Prefix your email title with [CS446] or [ECE452] or [CS646]
  - CC all team members (stakeholders) if project related
- Important: Please do NOT leave your questions to the last minute. We do NOT guarantee a response time of less than 24h.

Course Delivery

- Lectures are in person; materials uploaded to the course website
  - Slides uploaded before the class
  - More materials (coding demos, links to related resources) may be added after the class
- Attendance is recommended, not only for lectures but also for meeting and working with your teammates



Textbook

- No textbook required
- These may be helpful:
  - Software Architecture: Foundations, Theory, and Practice
  - Essential Software Architecture
  - Design of Design
  - Mythical Man Month
  - Head First Design Patterns

Expectations

What do you hope to take away?

- Discuss with your classmates:
  - What topics do you hope to cover?
  - What skills do you hope to develop?
- Note down and post your ideas
- We will discuss the top responses when we reconvene

Intended Learning Outcomes

By the end of the course, you should be able to:

- Differentiate how various architectural styles and design patterns *enhance* or *degrade* a system's functional and non-functional properties
- Generate and justify an architecture and/or design given a collection of requirements
- Produce and present *concise* and *unambiguous* architecture and design descriptions
- Create and implement an architecture and design, *refining* it into a complete system

Course Outcomes

This is a project-based learning course. The main takeaways through the course project will be:

- The mobile app that you build along with its associated atifacts (e.g., architecture and design documents)
- Experience to work in a team to build a software app from the ground up
- An opportunity to practice pitching a software idea and presenting/defending what you built

Expectations

- What you can expect from me
  - I intend to be as considerate and understanding as possible
- What I expect from you
  - Keep us posted; if there is a technical/logistic/any problem, let us know sooner rather than later
  - If you have to miss any deadline for any reason, let us know as soon as you can; we will try to find a solution
  - If you are struggling to keep up or have any issues, please reach out to us as soon as possible

Expectations (cont.)

- What I expect from you … in your projects
  - Collaborate with teammates!
  - Be responsive to your teammates and do your part in the project
  - Class time is a good time to sync up; there is 10% attendance grade to encourage you to come to class to meet with your teammates
  - Submit deliverables of all forms on time

Assessment & Project

Breakdown

- Details can always be found on the course website

Assessment	Value
Project	50%
Attendance	10%
Final Exam	40%

Break down into milestones (next slide)

1% each week for weeks 2-6 and 8-12 // week 7 is the reading week
Attend either Mon or Wen class and sign the attendance sheet to get the mark

2 hours; to be scheduled during the exam week

Project

- You will build an Android app in teams of six
- Goals
  - To make something useful
  - To learn about architecture styles and design patterns though application in a development setting
  - To leverage current technology – Android
  - To have fun along the way!

Project Restrictions

- Hard restrictions
  - The app should be implemented as a Native Android app (i.e., not built using an app builder or a framework like React/Node.js or HTML5)
  - The app must be useful, taking advantage of being on a mobile platform
    - NO simple database CRUD apps that do not make sense in a mobile context
  - The app should use at least 2 architectural styles and 2 non-trivial design patterns that will be discussed in class
- Soft restrictions
  - Games are not recommended
  - Apps requiring a complex server infrastructure are not recommended
  - Apps requiring crowd buy-in (i.e., requiring large amount of users to contribute content to be viably useful) are not recommended
  - *You are responsible for setting up any infrastructure/user for demoing the app*

Breakdown for Project

- Project grade = (base + bonus) * (scale / 100)
- Base grade: assigned per team, sum of all milestones

Date	Milestone	Value
Jan 17	P0: Team Formation	-
Jan 24	P1: Project Setup	2%
Jan 31	P2: Project Proposal	3%
Feb 12	P3: Iteration 1 Demo	5%
Mar 05	P4: Iteration 2 Demo	5%
Mar 19	P5: Iteration 3 Demo	5%
Mar 31 / Apr 02	P6: Final Presentation	20%
Apr 04	P7: Final Report	10%

Breakdown for Project (cont.)

- Bonus grade: assigned per team across the term

Bonus Item	Value
P2: Best (per section) project proposal	2%
P3-P5: Best (per section) progress at each iteration	2%
P6: Best (per section) final presentation	2%
P7: Prepared a short video (3-5 min) to demo your app	2%
P7: Submitted to the Google Play store	2%

Project Scaling Factor

- Scale is a factor ranging between 0-100

Component	Scale Weight
Completeness	10
Utility	10
Polish	10
Difficulty	20
Individual effort	50

Assigned per team

Assigned per person (next slide)

Individual Effort

- Assigned at the end of the term based on the following factors
- Commit history on GitHub
  - Commit code under your own username!
  - If you do pair-programming, commit your code in turn
- Time log table in your GitHub repository `timelog.csv`

```
date,Pengyu,Saarang,Bihui,Daniel,Amber,task
2024/01/06,1,0,0,0,0,set up github repository
2024/01/07,0.5,0.5,0.5,0.5,0.5,project meeting
2024/01/08,0,1,0,0,0,write requirement doc
2024/01/08,0,0,1,0,0,implement welcome screen
```

Academic Integrity

- You collaborate (with teammates and classmates) to complete your project
  - DO: ask questions on Piazza and answer other students' questions
  - DO: talk to other teams if you want
  - DO: use online resources (e.g., StackOverflow) and Gen AI (e.g., ChatGPT) to help you learn, code, etc.
- However, collaboration ≠ plaglarism/cheating
  - DO NOT: claim someone else's work (including Gen AI) as yours
  - DO: add citations/acknowledgements when you receive help from other teams, online resources, Gen AI, etc.

Action Items

- Review course website
- Review requirements of P0 and P1
- Start looking for teammates
- Download and install Android Studio (may take some time)

Welcome to the class! I hope you enjoy it!