

# Software Design & Architecture

Mei Nagappan

(material adapted from Reid Holmes)

# Lecture Summary

- ▶ Administrative details
- ▶ Expectations
- ▶ Project
- ▶ Assessment



# Dates and Times

**Lectures** in DWE 3522 T/Th @ 1600 - 1720

I will be available for a few minutes before and after.

**Tutorials** will NOT be held this year

**Office Hours** will be by appointment at DC 3349

**TA Office Hours:**

Ivens: Mon 1500 - 1600 (DC 2517)

Aaron: Wed 1100 - 1200 (DC 3334)

Cassiano: By appointment for Android dev questions (DC 3334)



# Directory

- ▶ **Instructor:** Dr. Mei Nagappan (Prof. Mei)

Office: DC 3349 (by appointment)

Email: [mei.nagappan@uwaterloo.ca](mailto:mei.nagappan@uwaterloo.ca)

**TA:** Aaron Sarson

Email: [asarson@uwaterloo.ca](mailto:asarson@uwaterloo.ca)

**TA:** Ivens Portugal

Email: [iptugal@uwaterloo.ca](mailto:iptugal@uwaterloo.ca)

**TA:** Cassiano Monteiro

Email: [cassiano.monteiro@uwaterloo.ca](mailto:cassiano.monteiro@uwaterloo.ca)

IMPORTANT: Please do not leave your messages to the last minute or expect a response time of less than 24h.



# Key Information Source

<https://learn.uwaterloo.ca/d2l/home/362279>

<https://cs.uwaterloo.ca/~m2nagapp/courses/CS446/1181/>

# Slide Availability

Slides are available online

- The course web page will be updated before class with latest set of slides.
- The slides will not be heavy on concrete examples as these will be covered in class.
- In-class activities will not be posted.

The slides **cannot** take the place of the lectures

You will need to attend the architecture and design activity classes to know the material as there will be a discussion on each.



# Textbooks

- ▶ **No** textbooks are required
- ▶ These may be **helpful**:
  - ▶ Software Architecture: Foundations, Theory, and Practice
  - ▶ Essential Software Architecture
    - ▶ Freely available to students in digital form
    - ▶ Design of Design
  - ▶ Mythical Man Month
- ▶ Links are provided on the web page along with slides for SA and ESA



# Intended Learning Outcomes

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

**Critique** an existing architecture or design.

**Differentiate** how various architectural styles and design patterns enhance and 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.





# My Expectations

## Be professional

questions in class, email, interacting with TAs

## Attend lectures

talk to class or team mates if you are away

## Participate

during discussions, activities, group project



# Your Expectations?



# Project

- ▶ Will be completed in teams of three and some cases four
- ▶ Select your own teams
- ▶ One team member must email me and the TAs:
  - ▶ The names of your teammates
  - ▶ The GitHub repo for the project.
  - ▶ Due Noon Jan 11 via email
- ▶ If you do not have a team by Jan 11 or your team is too small, we will sort it out in class
  - ▶ (you will be assigned to a team, so please try to find one yourself/fill up your team)



# Project (Mobile Apps)

- ▶ Goal:
  - ▶ To make something *useful*
  - ▶ To learn something *new*
  - ▶ To leverage current *technology*
  - ▶ To have *fun*
- ▶ Constraints:
  - ▶ Be useful, novel, and leverage technology
  - ▶ Cannot require crowd involvement

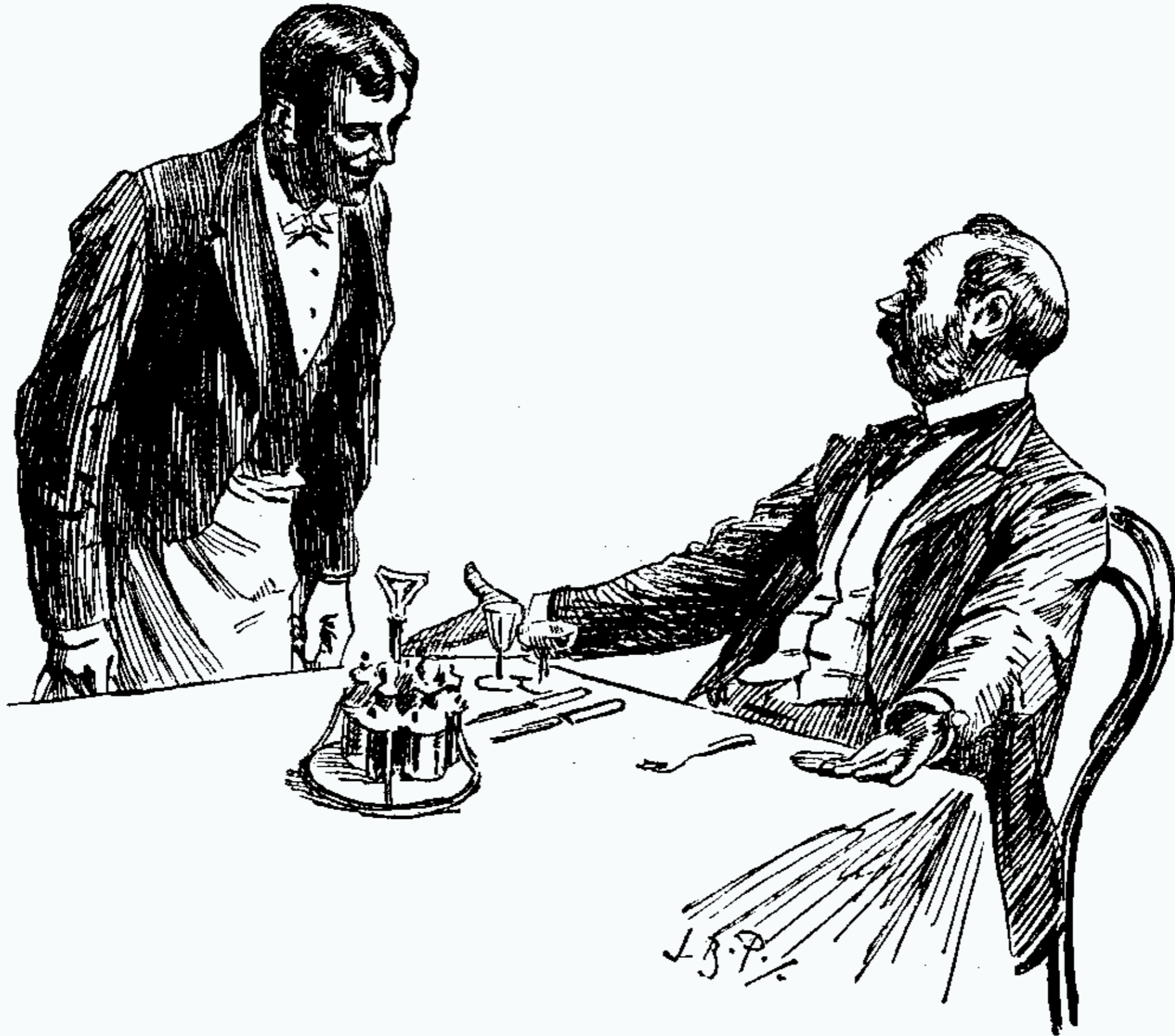


# Project (Mobile Apps)

- ▶ Goal:
  - ▶ To make something *useful*
  - ▶ To learn something *new*
  - ▶ To leverage current *technology*
  - ▶ To have *fun*
- ▶ Constraints:
  - ▶ Be useful, novel, and leverage technology
  - ▶ Cannot require crowd involvement
  - ▶ **MUST** work on Android



# Connoisseur



# *Connoisseur*



# Projects from the Past





# Deliverables

- ▶ Deliverable 0: Team and GitHub repo
- ▶ Deliverable 1: Project proposal (5%)
- ▶ Deliverable 2: Proposal presentations (Pass/Fail)
- ▶ Deliverable 3: Prototype document (5%)
- ▶ Deliverable 3: Prototype demo (Pass/Fail)
- ▶ Deliverable 5: Project arch + design document (10%)
- ▶ Deliverable 5: Project arch + design oral exam (10%)
- ▶ Deliverable 6: Project presentations (5%)
- ▶ Deliverable 6: Participation journal (5%)



# Schedule



# Assessment

- ▶ Project deliverables 40%
  - ▶ + 2% best proposal
  - ▶ +2% best prototype demo
  - ▶ +2% best final demo
  - ▶ +2% accepted to curated app store
- ▶ Arch/Design activity 10%
- ▶ Final Exam 50%
- ▶ Some project deliverables will be pass/fail
- ▶ MUST pass final exam and ALL pass/fail elements



# Project Scaling

- ▶ Project deliverables: 40%  
$$(\text{project} + \text{bonus}) * \text{scale} = \text{final project grade}$$
- ▶ Scale will range between 0.50 and 1.0 (25 points)
  - ▶ 10: completeness (compared to proposal)
  - ▶ 10: utility
  - ▶ 10: polish
  - ▶ 10: difficulty
  - ▶ 10: pivot



# Academic Integrity

collaboration vs. plagiarism

collaboration vs. cheating

This is **important**. The project will have **team** and **individual** components.

