

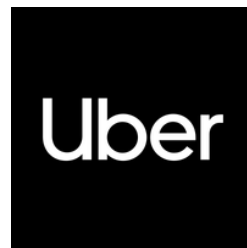
# **CSCI3100 Tutorial 1: Project Introduction**

January 18, 2021

# Objective

- Practice what you are learning in this CSCI3100 Software Engineering course by **designing, implementing, testing,** and **documenting** a **modern Web based client-server application.**

# Modern Application

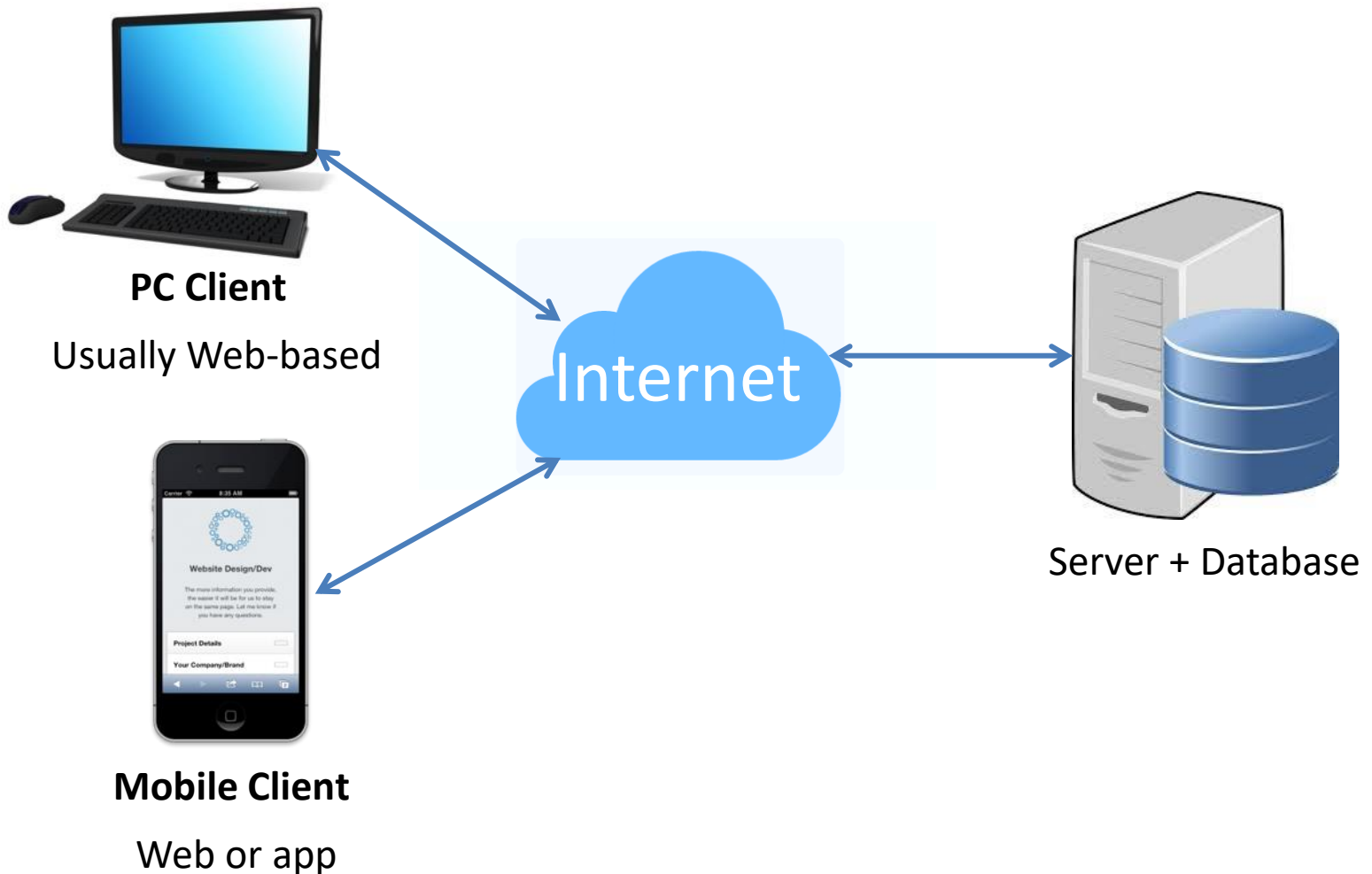


The Government of the Hong Kong Special Administrative Region



恒生銀行  
HANG SENG BANK

# Common Architecture



# Important Stats

- Project accounts for **40%** of the course grade.
- There are **4** phases in the project.

Phase Deliverables	Weightings
Project Design Document	10%
Initial Code	10%
Final Code and Demo	50%
Final Report and Commented Code	30%

# Project Topic

- You define your own project topic.
- In specification, we define “basic functions” and “advanced functions” for your project.
- Basic functions: functions that your project must have.
- Advanced functions: functions that are optional for your reference.
- You are recommended to enhance your project with other fancy functions.

# Phase 0: Forming Project Team

- **5 students** for each group. A team with  $<5$  students may be assigned partners randomly.
- All students in a group work on the same project for the entire project duration.
- **No joint work** over any technical aspects of the project is allowed between any two teams.
- Each project group will be assigned to **one** tutorial section for the relevant project phase presentations.
- Deadline: **Jan 22 (Fri.)**

# Phase 1: Project Design Document

- **Duration:** 5 Weeks (19 Feb 23:59:59 pm)
- **Grade Weighting:** 10%
- **Submit** a **project preliminary design** document to provide **high-level descriptions** on functionalities, features, and architecture design of your application.
- **Project background, architecture diagram, brief descriptions of the system components** and **UML diagrams** should be provided.



# Phase 1: Project Design Document

- **Feedbacks** will be provided on your project preliminary design. Students should **reconsider** and possibly **revise** the project goals.
- More details are in **Appendix 1** (e.g., outline of the document).

# Phase 2: Initial Code

- **Duration:** 4 weeks (23:59:59 19 Mar)
- **Weighting:** 10%
- Submit your initial code. The initial code may consist of ALL **major class definitions, interfaces** and **member function prototypes**.
- The implementation is **NOT** the primary concern.
- Evaluate your **project design** and make sure your project is in **right direction**.

# Phase 3: Completed Code and Demo

- **Duration:** 4 weeks
  - Code submission: 23:59:59 14 Apr
  - Demo Day: **15 & 16 Apr (two days)**
- **Weighting:** 50%
- Submit your **completed code** of the project, which should be self-contained and working properly.
- You need to make a **demonstration** after the submission of the final code. (15 min for each group)
- Signup schedule for demonstration will be announced in the **course website**.

# Phase 4: Final Report and Commented Code

- **Duration:** 2 weeks (23:59:59 30 Apr)
- **Weighting:** 30% (Important!)
- Submit a **final report** and **commented code** of your project.
- In addition to detailed project description, the final report should also show what **software engineering techniques** you have applied in the project, and what **lessons you have learned**.

# Phase 4: Final Report and Commented Code

- More details are in [Appendix 1](#) (e.g., outline of the document, requirement for each Section).

# Grading Criteria

- **Demo:** Market-based: more **impressive** systems receive higher grades.
- **Reports:** based upon the **technical content** and the **clarity of the presentation**.
- **Final code:** based upon the **modular structure, comments, and cleanliness**.
- The overall **quality** and **functionality** of the project is the key scaling factor for all aspects.

# Grading Criteria

- Project grade will be based for the **whole team** and will **NOT** be assigned **individually** to members.
- However, complaints about **free-riders** will be considered during project development and will be verified in Demo Day.

# Submission (report)

- Each project group should submit the softcopy of the report and the **VeriGuide** recipient to **Blackboard** before the deadlines.
- 1 **Project Initial Design** softcopy (phase 1, 5-10 pages) + 1 **Final Report** softcopy (phase 4, 30 or more pages)
- File names **(Important!)** :
  - “Group\*\* Project Initial Design Report”
  - “Group\*\* Project Initial Design Report VeriGuide”
  - “Group\*\* Final Report”
  - “Group\*\* Final Report VeriGuide”(replace \*\* with your **group ID**) (without quotes)



# Submission (code)

- **ALL** project stuff (source code, images, databases files, etc.) should be conducted by the **version control system (Git)**.
- You **MUST** submit your project to GitHub and **faithfully** record your **coding activities**.

# Submission (code)

- You are required to create a GitHub repository and submit the **HTTPS URL** or **SSH URL** of your repository to the Google form <https://forms.gle/8W8GYhDwEyctBT716> before 23:59:59, 17 March (two days before Phase 2 due).
- We will pull the latest code on the due date of **Phase 2, Phase 3, and Phase 4.**

# Submission (code)

- We will **NOT** accept submissions via other approaches.
- We will **NOT** help you debug your code.
- A detailed guide for code submission is in **Appendix 2.**

# Requirement: Technical

- Frontend: Web based access.

The server-side program is recommended to be built on Node.js. PHP, or Django is also acceptable.

Why  ?

- ① High-Performance
- ② Easy to modify and maintain



# Requirement: Technical

- Backend: Database.

SQL database (e.g., MySQL, or Sqlite), or NoSQL database (e.g., MongoDB, or Redis) **MUST** be employed for **storing data**.



# Requirement: Programming

- Encouraged to employ various **well-known** technologies and tools (e.g., AJAX, Vue.js).



# Requirement: Programming

- Please note that designing **static HTML web pages** is not programming.
- Project tutorials will cover **related techniques** and **tools**, such as JavaScript, AJAX, CCS3, HTML5, Node.js, Bootstrap, etc.

# Requirement: Documentation

- One key purpose of this course is that you learn how to do **modular design** of software and how to **document the design** using symbolic representations, i.e., UML diagrams.
- The templates are available in the appendix of the project specification.





# Tutorial Schedule

Week	Date	Tutorial	Topics	Task
1	1/11~1/13		<a href="#">Tutorial policies, schedule, and session assignment</a> ↓	Read tutorial procedure
2	1/18~1/20	P	<u>PJ1: CSCI3100 Project introduction, requirement, and demonstration</u>	Project assigned (1/18), HW1 assigned
3	1/25~1/27	P	<u>PJ2: Client side technologies (HTML5, JavaScript)</u>	
4	2/1~2/3	H	HW1 demo (Lecture Topic 1-3)	HW2 assigned
5	2/8~2/10	P	<u>PJ3: Server side technologies (Node.js, AWS, Database)</u>	HW1 due (2/7)
6	2/15~2/17		No Class/Tutorial	Lunar New Year vacation Project design document due (2/19)
7	2/22~2/24	H	HW2 demo (Topic 4 Software Spec – Data Flow Diagram, FSM)	HW3 assigned
8	3/1~3/3	P	<u>PJ4: Advanced technologies I (Vue.js, Angular)</u>	HW2 due (2/28)
9	3/8~3/10	H	HW3 demo (Topic 4 Software Spec – Petri Net, ER Diagram, and Logic Specification)	HW4 assigned
10	3/15~3/17	P	<u>PJ5: Advanced technologies II (Android)</u>	HW3 due (3/14) Initial code due (3/19)
11	3/22~3/24	H	HW4 demo (Topic 5 Software Design – TDN, GDN, and Refinement)	HW5 assigned
12	3/29~3/31		No Class/Tutorial	HW4 due (3/28) Reading Week & Easter Holiday
13	4/5~4/7		No Class/Tutorial	Reading Week & Easter Holiday
14	4/12~4/14	H	HW5 demo (Topic 5 Software Design – UML, Programming Technique)	Completed Code due (4/14) Demo day (4/15 and 4/16) HW6 assigned
15	4/19~4/21	H	HW6 demo (Topic 6 Software Verification)	HW5 due (4/18)

# Demo:

## Selected Previous CSCI3100 Projects