COMP5241 Project Overview

Form a team of 4–5 students to develop a web-based system that empowers universities lecturers in HK to create, generate, deliver, and manage interactive learning activities inside/outside classes.

Here are some initial ideas of the core requirements:

- Teachers can create a variety of learning activities (e.g., polls, quizzes, word clouds, and short-answer questions, mini-games) for students to complete synchronously in class
- PolyU Teacher can create courses and import their students (which can be linked to the student IDs).
- Integrate GenAI to platform features. E.g.
  - Create learning activities (e.g. based on the instructor's teaching contents, web resources, etc) and let instructors review and refine AI outputs;
  - Implementation of learning activities (E.g., automatically group similar answers from students)
- Leaderboard, Dashboard and reports of learning activities for instructors and students
- Admin features and dashboards
- Responsive UI Design (e.g., display properly in mobile devices)
- You should deploy a version of your app in some cloud platforms (e.g. Vercel) which can be accessed publicly with URL.

Here are some initial ideas about the nice to have features

- The system should be scalable to support a large class of students to participate in learning activity at the same time.
- The platform can support instructors/students from other universities
- The students can complete the learning activities asynchronously before/after the lessons.
- Integration with LMS (e.g. Canva, Moodle) such that students are not required to login to separate system, or the uploaded teacher materials in LMS can be accessible to the platform.
- Integration with external API to enhance the platform features
- Your platform can be deployed locally using Docker.

You should study the existing platform such as ClassPoint, Mentimeter. etc , their limitations, and explore how your team may develop a platform with improved/more innovative features and better user experience. Just like real-world software projects, the initial requirements will often be vague initially and may change during the development process. Please feel free to contact me for clarification of requirements. The project is also open to any new ideas about the features of the application.

Example: Mentimeter poll
https://www.youtube.com/watch?v=DKHWDI-6Pn8

Group Formation

Students should form a group of 4-5 members in blackboard by the end of week 3 (i.e. 20/9/2024 (SAT)). Students without a group will be randomly assigned a group after the deadline.

The GitHub classroom invitation link will be available soon. Team leader is required to accept the invitation and create the project repository for the team in our course project's GitHub organization. The repository should be private and only accessible by the team members and the instructor. The team leader will be the admin of the repository and should the other team members to the repository for collaboration. During the project,

- Each team member should use the GitHub repository (e.g. issues, project board, etc) to manage and collaborate on your project.
- Your project source code and other related files (e.g. documentation) should be uploaded to your project repository in the course project's GitHub organization.
- All students should in the group should actively participate during the software development (E.g. coding, testing, writing documentation).

The project meeting will be held during the lectures in week 7 or week 8. Each group will have around 5 minutes to discuss your project progress. You should prepare a 1–2-page summary of your progress, division of work, and your system design/prototype for discussion. We may also clarify the requirements in the meeting.

Final Submission

The presentation will be held in the week 12 and week 13. Each group will have around 8-10 minutes (to be confirmed) to present your project. The presentation slides should cover the following aspects:

- An overview of your project, features and system architecture.
- The demo and features of your prototype/system (record a video of your demo prototype)
- Sharing, findings and reflection on the project. Examples:
  - Development Process
    - Briefly introduce your team's technical/programming background and discuss your development process and examples of how your group collaborates. Discuss how each team member contributing to the project and the challenges encountered (if any)
    - How do you design the UI? How did you improve the UI and UX of the platform?
    - Which feature in the project was the most challenging to implement and how do you solve the problem?
  - Use of GenAI in software development
    - Which parts of the code are generated by AI and which parts are manually coded? How is your team collaborating while using GenAI to complete the software development tasks? What works well/not so well?
    - Have you used GenAI to support the software testing and perform other tasks in software development? Have you done any refactoring? What are the good practices when using GenAI for software development? Give examples to showcase how your team are using GenAI to assist in your project development and implementation.
  - Overall reflection: Reflect on what you have learnt/the findings from the project, what you have done well/not done so well, what the challenges and solution, etc.
- Appendix: You may include other information/screenshots that are relevant to your project.

You should submit your presentation slides and a final report with the following sections

- Overview of your project, system design and features
- Discuss your software development process (requirement, design, development, testing, deployment, etc) and how your team collaborates in your project
- User manual for admin, instructor and students and links to demo videos of the features

Evaluation Criteria

- Project discussion meeting (5%)
- Final presentation and deliverables
  - System features and prototype (30%):
  - Presentation (30%)
  - Sharing, findings, and reflection (35%)

Rubrics (Tentative)

Criteria	Good	Satisfactory	Needs Improvement
Project Discussion Meeting (5%)	Clear progress summary with well-defined prototype and division of work. Team shows active participation and strong grasp of requirements.	Progress shared but limited details on prototype or task division. Participation evident but uneven among members.	Lacks clarity on progress or work division. Prototype/design unclear. Some members appear disengaged.
System Features & Prototype (35%)	Features are **innovative/tailored** to needs of the stakeholders. Clear articulation of motivations, differentiation from existing tools (e.g., ClassPoint, Mentimeter). **GenAI features** are well integrated. Prototype is functional, responsive, and deployed with good UI/UX.	Core features implemented but reflect more **basic or imitative functions**. Motivation and differentiation partly explained. Prototype deployed but with limited interactivity or UI/UX polish.	Features very **basic or similar** to existing tools. Lack of motivation or justification. Prototype incomplete, not responsive, or not deployed.
Presentation (25%)	**Clear, engaging, professional delivery** with confident pronunciation, varied tone, and smooth demo. Slides are well-structured with good visuals to enhance understanding.	Presentation understandable but less polished. Delivery may include minor clarity/pronunciation issues; slides/content somewhat repetitive or cluttered. Audience engagement limited.	Presentation **unclear, monotonous, or poorly structured**. Slides confusing with missing sections. Demo ineffective or absent. Audience disengaged.
Sharing, Findings & Reflection (35%)	Provides **deep, critical reflections**: clear discussion of development process, team collaboration, GenAI usage (successes & challenges). Insights are thoughtful and specific. Clear takeaways about learning process.	Reflections present but **general or brief**. Development process and GenAI usage described superficially. Limited insights about design choices, challenges, or lessons learned.	Reflection **minimal or missing**. No real insights beyond surface-level description. Lacks discussion of GenAI usage, challenges, or lessons learned.

FAQ

1. What programming languages should we use?

   Ans: In the project, you are welcome to use any programming languages, frameworks, and libraries, etc. You can also GenAI for your project (e.g. idea brainstorming, designing, evaluating alternatives, development and coding, testing, etc).

2. Can I work on the project individually?

   Ans: According to the COMP5241 subject description form, the course's Intended Learning Outcome (ILO) d is about "Work and collaborate effectively in teams possibly with multi-disciplinary backgrounds as a leader or a member" and "Group project will fulfill the need to assess intended learning outcome d". . Therefore, we prefer that students can complete the project in group so that the course can fullfill the requirements specified in the subject description form. If there are genuine difficulties/justification that you want to complete it individually, please email me for discussion.

References:

- https://www.classpoint.io/get-started
- https://www.mentimeter.com/features
- https://docs.vibe-coding-framework.com/