CITS3007 group project – phase 1

Version:	0.1
Date:	04 Apr 2025

1. Background

Your team work as software developers for **Enjjin Media**, the game publisher behind such well-known hit titles as *The Least of Us* and *Underwatch*. The company is now venturing into the MMORPG (Massively Multiplayer Online Role-Playing Game) space with its latest project:

Oblivionaire Online (OO)

An ambitious new online multiplayer game, *Oblivionaire Online* thrusts players into a post-apocalyptic world in which obscenely rich plutocrats have brought about the end of civilised society. Players take on the roles of neo-feudal barons, financial warlords and venture capitalists, vying for control over the shattered remains of civilisation: piloting extravagant mecha-suits, defending their fortified enclaves, and wielding grotesque amounts of resources while the rest of the world burns. Enjjin Media's marketing department predict that this sort of escapist fantasy is exactly the kind of thing that today's consumers will flock to.

1.1 Your tasks

Enjjin Media has tasked your team with developing a critical subsystem for *OO* – the **Access Control System (ACS)**. This system will be responsible for authenticating players, managing account security, and enforcing access policies for both users and Enjjin's operations staff.

The ACS has the following responsibilities:

- **Player authentication** – Ensuring that only registered players can log in.
- **Role-based access control (RBAC)** – Managing in-game privileges, from ordinary wasteland wanderers to elite plutocratic overlords with access to game-altering financial instruments
- **Session management** – Handling login sessions, preventing unauthorized access, and dealing with connection timeouts or suspicious activity.
- **Admin and operations access** – Providing secure tools for Enjjin's staff to manage accounts, enforce bans, and troubleshoot authentication issues.

2. Project rules and deadlines

2.1 Total marks

- This project contributes **30%** towards your final mark this semester, and must be completed as a group. Groups are allocated by the Unit Coordinator, and consist of 4-6 people.
- The project is marked out of 60.

2.2 Project phases, submission dates, and peer evaluation

- This document describes phase 1 (planning) of the CITS3007 group project, which is worth 10 marks and is due on Wednesday 16 April (11:59 pm) 2025 (week 8).

  All students in a group are awarded the same mark for phase 1.

  Specifications for phase 2 will not be released until all groups have submitted deliverables for phase 1 (or have forfeited all marks for phase 1).

- Phase 2 (implementation and report) is due in **week 11** and is worth 40 marks.

- Phase 3 (demo/presentation) is delivered in **weeks 11-12** and is worth 10 marks.

In phase 3 (final demonstration/presentation for the project) you'll be expected to reflect on your plans from phase 1, what challenges you faced and how you dealt with them.

Each group member will be asked to evaluate their own and other group members' contributions to the project for phases 2 and 3.

Once these member evaluations are done, each group member will receive a Group Contribution Factor (GCF). The final mark for phases 2 and 3 is $raw\_mark \times GCF$.

The minimum possible GCF is 0, and the maximum GCF is 1.2 (but no student can be awarded more than 100% for phases 2 and 3).

2.3 Academic conduct

You are expected to have read and understood the University Academic Integrity Policy. In accordance with this policy, you may discuss with other students the general principles required to understand this project, but the work you submit must be the result of your own group's effort.

2.4 Penalties for late submission

For phases 1 and 2, you must submit your project before the submission deadline. The standard penalties for late submission apply.

2.5 Revision and effort tracking

All members of a group are expected to contribute equal effort to the project.

We recommend you maintain:

- a **private** Git repository (hosted on a service such as GitHub or GitLab) containing the code for your project, and
- a shared spreadsheet (e.g. on MS Teams or Google Docs) tracking tasks allocated to and completed by members each week.

Neither the repository nor spreadsheet are submitted for assessment, but the Unit Coordinator may ask to review them.

Tips:

- In addition to code, you might also choose to use your Git repository to track the content of reports, presentations, or even your task spreadsheet. But if you do, you're best using a plain text format (such as Markdown, LaTeX or CSV) – version control systems do a poor job of tracking changes to binary formats such as `.pptx` (PowerPoint) and `.xslx` (Excel).

3. Phase 1 – planning

3.1 Phase 1 deliverable

The phase 1 deliverable is a **report**: its purpose is to establish a clear plan for collaboration, tooling, and quality considerations ahead of the implementation phase (phase 2).

3.2 Report content requirements

Your report should address the following:

1. Team Communication & Responsibilities

   - Describe how the group will communicate throughout the project:
     - Frequency of meetings (e.g., weekly; fortnightly; as needed)
     - Meeting format (e.g., face-to-face; online via video calls or chat platforms)
     - Preferred communication tools (e.g., Discord, Slack, MS Teams, email)
   - Define how responsibilities will be allocated in phase 2:
     - Who will be responsible for which tasks?
   - How will the group ensure accountability and track progress?

2. Version Control Strategy

   - Specify where the project's source code will be hosted (e.g., GitHub, GitLab, Bitbucket).
   - How will you handle merging members' contributions?

- If you're familiar with version control branching strategies, will you adopt any particular strategy or workflow? (e.g., feature branches, main/dev workflow)
- Identify any version control policies (e.g., commit message conventions, review/approval process before merging).

3. Development Tools

   List the common tools the group will use for implementation:

   - Code editor or IDE (e.g., VS Code, JetBrains, Vim)
   - Any additional tools for collaboration or efficiency (e.g., linters, debugging tools, CI/CD services).

   Explain why you made these choices.

4. Key Secure Coding Practices for Phase 2

   Identify three security-related tools or practices covered in the unit that will be most critical during phase 2. For each, explain:

   - why it is relevant to the project
   - how it will be applied during development, and
   - how the group will ensure it is effectively used.

5. Risk Management & Quality Assurance

   - Outline potential risks to the project and how they will be mitigated. (You may wish to think about resourcing risks – e.g. member illness, service outage – as well as technical and operational risks.)
   - Describe how code quality will be maintained:
     - Will the group follow a specific coding standard?
     - Will peer reviews, automated testing, or static analysis tools be used?

6. Optional – group name

   - Nominate a name for your group – less than 20 characters, may be multiple words. Pronounceable, workplace friendly, and no embedded HTML tags or SQL commands, please. ☺

At the end of the project, you'll have a chance to reflect on how useful (or not) these plans were, and to what degree you needed to adapt or amend them.

- ***Tip:*** It's easy to be over-optimistic in your plans. You might want to investigate the strategy of conducting a *pre-mortem* for the project to counteract this tendency.

You'll need to create your phase 1 report based on the information in section 1.1 – full requirements won't be available until phase 2. The code you submit in phase 2 will need to comply with the guidelines for **"Long answer" questions requiring code** in the standard CITS3007 rubric framework.

3.3 Length guidelines and assessment

There is no maximum or minimum length for the report.

Assessment is based on:

- whether the project addresses the content requirements (5 marks), and
- report clarity, coherency, and strength of justifications given for choices made (5 marks).

3.4 Report submission

- Submit your reports for phase 1 via Moodle (one submission per group) by the due date.
- Ensure that the name and student number of all group members are included in the submission.
- Ensure that the group name and number are included in the submission.

3.5 Report format

See the CITS3007 website FAQ, "What are the formatting expectations for project reports?" for report format guidelines.

Briefly: you may submit a PDF, or a Markdown file.

4. Annexes

4.1 Group organisation tips

It's up to each group to decide what roles are important and which group members will perform them.

But besides coding, a few tasks or areas of responsibility you might want to consider are:

- Project management – Coordinating team efforts and checking that tasks are carried
- Design – Investigating how design principles (like the Principle of Least Privilege) will be incorporated
- Threat modelling – Identifying threats to important resources, and investigating how to address them
- Testing – Developing tests for the code your group writes
- Documentation – Writing reports or presentations