College of Engineering, Construction and Living Sciences
Bachelor of Information Technology
ID607001: Introductory Application Development Concepts
Level 6, Credits 15

Practical

Assessment Overview

In this **individual** assessment, you will design and develop an application that enables a user to upload and download a game build, i.e., .exe file to and from **Google Drive**. In addition, marks will be allocated for code quality and best practices, documentation and Git usage.

Learning Outcome

At the successful completion of this course, learners will be able to:

1. Design and build secure applications with dynamic database functionality following an appropriate software development methodology.

Assessments

Assessment	Weighting	Due Date	Learning Outcome
Practical	20%	21-06-2024 (Friday at 4.59 PM)	1
Project	80%	21-06-2024 (Friday at 4.59 PM)	1

Conditions of Assessment

You will complete this assessment during your learner-managed time. However, there will be time during class to discuss the requirements and your progress on this assessment. This assessment will need to be completed by **Friday, 21 June 2024** at **4.59 PM**.

Pass Criteria

This assessment is criterion-referenced (CRA) with a cumulative pass mark of **50%** across all assessments in **ID607001: Introductory Application Development Concepts**.

Submission

You **must** submit all application files via **GitHub Classroom**. Here is the URL to the repository you will use for your submission – https://classroom.github.com/a/wlzE5yYo. If you do not have not one, create a **.gitignore** and add the ignored files in this resource - https://raw.githubusercontent.com/github/gitignore/main/Node.gitignore. Create a branch called **practical**. The latest application files in the **practical** branch will be used to mark against the **Functionality** criterion. Please test before you submit. Partial marks **will not** be given for incomplete functionality. Late submissions will incur a **10% penalty per day**, rolling over at **5:00 PM**.

Authenticity

All parts of your submitted assessment **must** be completely your work. Do your best to complete this assessment without using an **AI generative tool**. You need to demonstrate to the course lecturer that you can meet the learning outcome for this assessment.

However, if you get stuck, you can use an **AI generative tool** to help you get unstuck, permitting you to acknowledge that you have used it. In the assessment's repository **README.md** file, please include what prompt(s) you provided to the **AI generative tool** and how you used the response(s) to help you with your work. It also applies to code snippets retrieved from **StackOverflow** and **GitHub**.

Failure to do this may result in a mark of **zero** for this assessment.

Policy on Submissions, Extensions, Resubmissions and Resits

The school's process concerning submissions, extensions, resubmissions and resits complies with **Otago Polytechnic | Te Pūkenga** policies. Learners can view policies on the **Otago Polytechnic | Te Pūkenga** website located at https://www.op.ac.nz/about-us/governance-and-management/policies.

Extensions

Familiarise yourself with the assessment due date. Extensions will **only** be granted if you are unable to complete the assessment by the due date because of **unforeseen circumstances outside your control**. The length of the extension granted will depend on the circumstances and **must** be negotiated with the course lecturer before the assessment due date. A medical certificate or support letter may be needed. Extensions will not be granted for poor time management or pressure of other assessments.

Resits

Resits and reassessments are not applicable in **ID607001: Introductory Application Development Concepts**.

Instructions

Functionality - Learning Outcome 1 (50%)

- **Backend:**
  - Developed using **Node.js**.
  - Can run in development without modification.
  - Use the **Google APIs Node.js Client** library to interact with the **Google Drive API**.
  - Implement endpoints for uploading and downloading game builds to and from **Google Drive**.
  - Ensure proper authentication is implemented using **OAuth 2.0**. Store the **client id**, **client secret**, **redirect uri** and **refresh token** in a **.env** file.
- **Frontend:**
  - Create an **HTML** file with a form to upload a game build to **Google Drive**. Ensure proper security measures are in place, i.e., validating the file type and size.
  - Display a list of game builds stored in **Google Drive** in a **HTML** table. The table should include the game build's name, size and a download button.
  - UI is visually attractive with a coherent graphical theme and style.

Code Quality and Best Practices - Learning Outcome 1 (45%)

- A **Node.js .gitignore** file is used.
- Environment variables' key is stored in the **.env.example** file.
- Appropriate naming of files, variables, functions and resource groups.
  - Resource groups are named with a plural noun instead of a noun or verb, i.e., **/api/items** not **/api/item**.
- Idiomatic use of control flow, data structures and in-built functions.
- Efficient algorithmic approach.
- Sufficient modularity.
- Code is formatted.
- No dead or unused code.

Documentation and Git Usage - Learning Outcome 1 (5%)

- A **GitHub** project board or issues to help you organise and prioritise your development work. The course lecturer needs to see consistent use of the **GitHub** project board or issues for the duration of the assessment.
- Provide the following in your repository **README.md** file:
  - How do you setup the environment, i.e., after the repository is cloned?
  - How do you run your application locally?
- Use of **Markdown**, i.e., headings, bold text, code blocks, etc.
- Correct spelling and grammar.
- Your **Git commit messages** should:
  - Reflect the context of each functional requirement change.
  - Be formatted using an appropriate naming convention style.

Additional Information

- You need to show the course lecturer the initial **GitHub** project board before you start your development work. Following this, you need to show the course lecturer your **GitHub** project board at the end of each week.
- **Do not** rewrite your **Git** history. It is important that the course lecturer can see how you worked on your assessment over time.