Assessment Brief – Version Control Exercises

Notes on Assessment Task:

This brief has been designed around the specific, cumulative evidence you must prepare for and present by your assessment milestone to demonstrate you have achieved competency in the requisite knowledge and skills for this subject. Submissions conforming to this brief will provide the specific evidence listed in the table titled *Assessment and Competency Requirements* within the Subject and Assessment guide for this subject.

You may present additional, or other evidence of competency, but this should be as a result of individual negotiation with your teacher.

For specific competency requirements and evidence needed for assessment submission, always consult the [Subject and Assessment Guide](https://aie.instructure.com/courses/9/pages/subject-guides) for this subject.

Version Control Exercises:

*Version Control Topics*

This subject covers the use of version control across multiple topics. Each topic contains a set of exercises that will guide you through creating a repository and performing common actions, such as committing, pushing, pulling, and branching.

You need to complete all required exercises throughout those topics.

You should be making at least three commits to your project as you work through the exercises.

*Version Control Documentation*

Every tool we use should solve a problem with our work or the way we work together.

Create a brief document that provides the following information:

1. **A link to your GitHub repository**
2. **A link to the merge commit** you created
3. **A link to the pull request** you filed as a part of your Version Control Exercises
4. **A problem statement** describing why we use Git for version control:
   1. **What is the problem we are solving?**
      1. Who is experiencing the problem?
      2. What are our requirements for a possible solution?
   2. **How does this solution (Git) solve this problem?**
      1. What are its advantages?
      2. What are its disadvantages?
5. **A list of resources** you referred to when learning Git (besides AIE resources), if any

Problem Statement

Problem statements summarize key information about the problem and the environment surrounding it. They are often brief, but can provide detail as necessary to describe the problem.

Consider this sample problem statement written for game engines:

*Small development teams may find it expensive to split their resources between building the game engine vs. building the actual game itself. Teams may need to dedicate at least 1-2 full-time employees (FTEs) to developing the engine as requirements change over the product lifecycle, increasing operational costs.*

*Developers may choose to use game engines middlewares like Unity or Unreal Engine in exchange for a small upfront cost in the form of licenses or royalties. Engineers can be instead be dedicated to developing the game instead, requiring less engineers and thus reducing operational costs.*

*Game engine middleware is often maintained by teams with dedicated R&D teams, providing its users with the latest innovations in the games industry at a fraction of the cost. Teams may hire for users experienced in specific game engines, minimizing resources spent on training and ramp-up.*

*If a game engine does not providing the features required for a particular game, its cost savings are reduced. Furthermore, missing features or bugs with the featureset can be difficult to fix if source access is not providing, possibly making the game engine more expensive to work with than a custom built one.*

In a few sentences, the passage of text describe the following:

* **What is the problem?** – The cost of building and maintaining custom game engines
* **Who is experiencing the problem?** – Small or new game development teams
* **What is needed in a solution to this problem?** – Minimizes the game engine development costs

The solution, game engines middlewares like Unity or Unreal Engine, are described as follows:

* **How does Unity or Unreal solve this problem?** – Licenses or royalties for the game engine can be much more affordable than building a custom engine
* **What are its advantages?**
  + Externalized cost of R&D teams
  + New hires may require less training and on-boarding to be effective due to common knowledge of the game engine
* **What are its disadvantages?**
  + Missing or incomplete features can require teams to devote some or all of the cost of developing those features in-house regardless
  + Bugs in the game engine may not be readily diagnosted fixed if source-access is not provided

Submission:

You will need to submit the following:

* A complete copy of your version control repository used to follow the *Version Control Topics*’ tutorials/exercises
  + This includes your full commit history for your repository
  + This includes your Visual Studio project in the repository
  + Be sure to remove any temporary build folders (i.e., the Debug and Release folders). Only project files, source code files, and any resource files used should be included in your submission.
* A copy of your *Version Control Documentation (see above)*

Package the above files in a single compressed archive file (.zip, .7z, or .rar) along with other assessment items when submitting your assessment work for this subject.