DOCUMENT VERSION CONTROL (MVC PROJECT)

This is an **individual project**. You must be able to explain any part of the project.

SCOPE

BASIC FEATURES (REQUIRED)

You need to develop a Version Control website for Documents. The documents can be placed in a public or private repository. Private documents can only be viewed by approved users (by choice of the uploader). Logs must be kept monitoring changes made to the files. For example, the file Report.csv was edited on Monday the 6th of August by Boba Fett. The choice is up to you as to how detailed you are going to log the changes. However, you must at least log the name of the user that changed the file, the date the file was changed as well as the size of the updated file.

There must be a register and login system beforehand.

This version control application must support at least two of the following file extensions

- .doc
- .docx
- .xlxs
- .csv
- .txt

USER STORY

The user needs to register an account. Once the account is setup, they will login in with their details. (Please provide validation checks on the registration page).

Once logged in, the user may choose to upload a file or view an existing file.

UPLOAD A FILE

The user browses to a folder on their local drive (or cloud drive for extra points) and uploads a file to their repository. The repository, for the sake of simplicity, can be created on your local drive or on an online server*. The file may have permissions. If it is public, then all users on the application can view and edit the file. If private*, the user needs to specify who can view and edit the file.

* These are bonus features that will get you extra marks, but will not deduct from your mark if they are missing.

VIEW AN EXISTING FILE

All files on the repository can be downloaded. A log of the file must be visible on the system.

ADVANCED (EXTRA FEATURES):

Not including any of these features will not subtract marks from your project, but they will add bonus points to your final mark.

- The repository is stored on an online server
- Users can communicate on the platform in real-time
- Use compression to manage uploading large files
- Encrypt all passwords

TECHNICAL ASPECTS.

This is a web application, not a desktop project.

1. You must build it using the MVC architecture.
2. SOLID principles must applied.
3. Manage Exceptions.
4. Use Source Control.
5. Use a static code analytic tool (e.g SonarQube) to inspect your code quality.
6. Refactor your code!

Although the emphasis of this project is writing code, a brief description of the Models, Controllers and Views created in your codebase must be documented as well as your database design.