**VCS Management Report**

I have selected GitHub for Musoplan Project as the version control management system since GitHub meets all the criteria mentioned in the organisational VCS requirements document for Globex Corporate.

As per the project development requirement, created a GitHub repository in the GitHub web account.

To work with the project in the local computer location, the repository link has copied from GitHub account and cloned it into the local project folder.

For that, I did the following steps:

* Cloned the GitHub repository link from the website.
* Open the local project folder and open the Git bash.
* Using Git command, ‘git clone <copied repository link> enter’
* Now, the local repository has created.
* Uploaded the Meeting minutes text file and VCS documentation.
* Created a branch under Main Project branch using git command:

‘git checkout -b <second branch name>’

* Added Project Design report md file in the newly created second branch using git command:

‘git add <ProjectDesign.md>’

* Added Project Coding JavaScript file in the newly created branch second branch using git command:

‘git add <ProjectCoding.js>’

* Committed and updated the change by using git command for each file:

‘git commit -m “msg” enter

* Updated files to the second branch using git command:

git push origin <second branch name>

* Checked the status of the files in the repository to check whether all the files have successfully deployed and to check for any untracked files to be committed.

Using git command: git status.

* To check the entire branch contents and activities performed,

and used git command: git log.

* Merged the completed documents to the main branch by using git commands below:

git checkout main

git merge <second branch name>

**Compliance Check with the VCS requirement document of Globex.**

The Musoplan project must be done using a distributed VCS that is familiar and has good online supporting ecosystems.

All development in the VCS must take place on feature branches, so the main branch is exclusively initialisation and then merges, Hence created secondary branches test and code runs.

Created a new branch under main branch to coordinate project files and activities throughout the design and coding phase.

All commit messages must follow the industry standard format.

Applied the format as per company standards

Commit scope should be kept to a single change. Always made sure that git commit has done after every file update.

Applied the scope as single changes,

for e.g.: git commit -m “Uploading completed MusoPlanCode.js file”

git commit -m “Uploading VCSReport.md file”

VCS must be configured so each commit contains the author's name and email address.

This has done and verified at the setting up and installation stage of GitHub and repository.

All repositories must start with just a README.md file and no source code, do not initialise a repository with code already present.

Verified the project folder with README.md file has created at the initial stage itself.

**Issues encountered while using GIT**

Few of the issues occurred during the Git repository development and usage are below:

* Error shown: Main ahead of commits.

Solution: Checked the git status and log.

Found the uncommitted files and committed it.

* Error shown: Can’t find the file or file path

Solution: Change directory to the correct folder and ran commands again.