CNG 492 Senior Project: Implementation

Software Configuration Management (SCM) Evaluation:

Name of the project: CID (Can It Drive)

The Team Members: Daniyal Selani

Mahmoud Hamra

Overview: For this project we will be using GIT for SCM. We will be hosting our project on GitHub and using the inbuilt tools and some plugins to generate and maintain certain SCM objects.

SCM Objects:

- Source and test Code:
  1. The source and test code is contained in multiple files which have dependencies on one another. Every time a new change is committed to the source code through git, a new branch is created for that change. The branch will then be merged to the master version after the validity and functionality of the code is verified. Git can track which member made which change.
  2. The source and test code is stored on git and is accessible to anyone as the repository is public as of now. The version of each new commit and branch can either be updated automatically by git or renamed manually.
- Document/File/Binary:
  1. Documents pertaining to the file can either be uploaded directly to the git repository or be written using a collaborative writing tool such as Google Docs. Google docs can track the changes made by each member
  2. The document can either be stored in a google drive, or more relevantly on the repository itself. The correct document would be uploaded to corresponding project branch
- Build Scripts
  1. Build scripts can be stored and tracked similarly to the source and test code. The dependencies in a build script can be stored in a manifest file. Either as a .json file or .xml file
  2. The build scripts are to be updated and changed with the relevant changes made to the source or test code whenever a new branch is created.
- Version Tree
  1. GitHub provides a tool which generates a version tree automatically based on the commit and branching history for that project.
  2. The version tree should be up to date as long as the branching and merging of the versions is done correctly in accordance to the flow and changes made to the project
- Log /History
  1. Log history can directly be accessed through the github repository settings or be produced by running different scripts. Log history can contain:
     - Who initiated the push
     - Whether it was a force push or not

    - The branch someone pushed to
    - The protocol used to push
    - The originating IP address
2. The log history will automatically be generated everytime it is requested or the script to build it is run