Software Engineering Tools Lab
Assignment No-5

PRN: 2020BTECS00033
Name: Prathamesh Raje
Batch: T6

Title - Understanding version control using VSS and Managing code using SVN

Q1. What is version control system and why it is important?

Version control systems are a category of software tools that helps in recording changes made to files by keeping a track of modifications done in the code.

- Enhances the project development speed by providing efficient collaboration,
- Leverages the productivity, expedites product delivery, and skills of the employees through better communication and assistance,
- Reduce possibilities of errors and conflicts meanwhile project development through traceability to every small change,
- Employees or contributors of the project can contribute from anywhere irrespective of the different geographical locations through this VCS,
- For each different contributor to the project, a different working copy is maintained and not merged to the main file unless the working copy is validated. The most popular example is Git, Helix core, Microsoft TFS,
- Helps in recovery in case of any disaster or contingent situation,
- Informs us about Who, What, When, Why changes have been made.

When you work on a development team, you may be touching similar parts of the code throughout a project. As a result, changes made in one part of the source can be incompatible with those made by another developer working at the same time.
Version control helps solve these kinds of problems and provides:

1. A complete history of every file, which enables you to go back to previous versions to analyse the source of bugs and fix problems in older versions.
2. The ability to work on independent streams of changes, which allows you to merge that work back together and verify that your changes conflict.

3. The ability to trace each change with a message describing the purpose and intent of the change and connect it to project management and bug tracking software.

Q 2. Illustrate different types of version control system with example.

There are two types of version control: centralized and distributed.

Centralized version control

With centralized version control systems, you have a single “central” copy of your project on a server and commit your changes to this central copy. You pull the files that you need, but you never have a full copy of your project locally. Some of the most common version control systems are centralized, including Subversion (SVN) and Perforce.

Distributed version control

With distributed version control systems (DVCS), you don't rely on a central server to store all the versions of a project’s files. Instead, you clone a copy of a repository locally so that you have the full history of the project. Two common distributed version control systems are Git and Mercurial.

- Visual VCS: TFS, Perforce
- Cloud-based VCS: Bitbucket, Github

Q 3. Perform below operations using CVS

a. cvs checkout
b. cvs update
c. cvs add
d. cvs remove
e. cvs commit

Q 4. Differentiate Between The Git & SVN Repository?

Git is a distributed VCS, while SVN is a centralized VCS. Git allows multiple people to work on the same codebase at the same time without a central repository. In contrast, SVN has a central repository that all changes are checked into. Git is generally faster than SVN, as it can perform most operations locally, while SVN requires communication with the central server. Git is better suited for large projects with many contributors, while SVN is better suited for smaller projects with a few contributors.

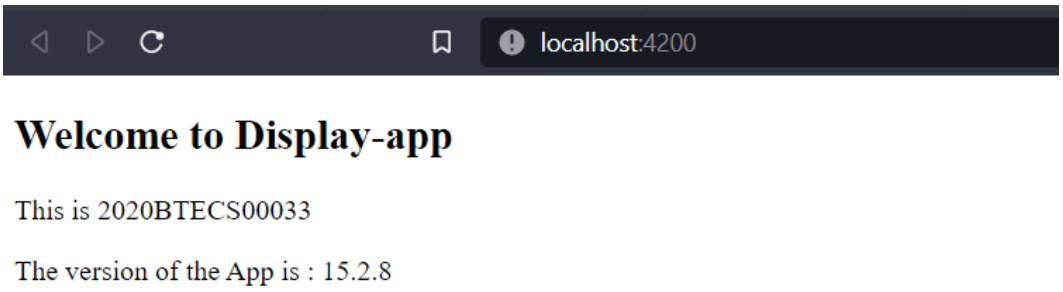
Q 5. What is “branch”, “tag” And “trunk” In SVN?

In SVN, a "branch" is a copy of the codebase that can be modified independently of the main codebase, often used for new features or experimental changes. A "tag" is a snapshot of a specific version of the codebase that is marked with a name, often used to mark a release version. The "trunk" is the main development line of the codebase in SVN.

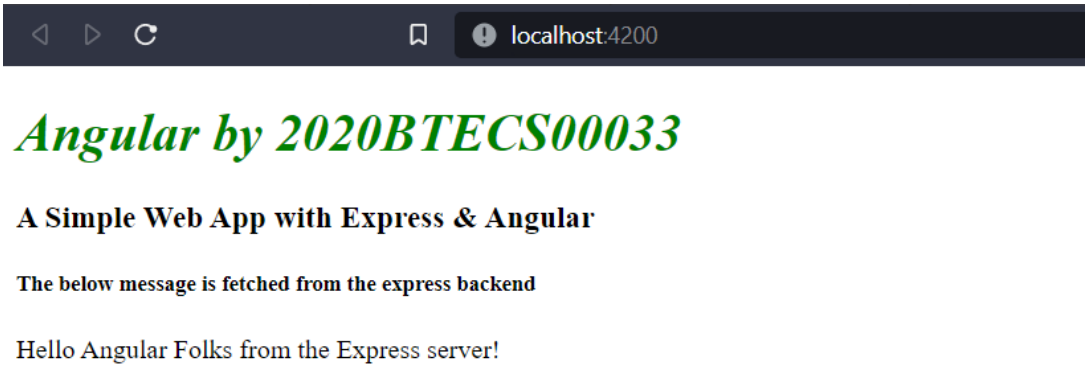
Q 6. How CVS is different from SVN?

CVS is a centralized VCS, while SVN is an improved version of CVS that also has support for branching and merging. SVN has better handling of binary files, and allows atomic commits, while CVS does not. SVN also has better support for renaming and moving files than CVS.

Q 7. Demonstrate a display the app version in angular.



Q 8. Build a simple web app with Express and Angular.



Q 9. What is git version control?

Git Version Control System

- A version control system is a software that tracks changes to a file or set of files over time so that you can recall specific versions later. It also allows you to work together with other programmers.
- The version control system is a collection of software tools that help a team to manage changes in a source code. It uses a special kind of database to keep track of every modification to the code.
- Developers can compare earlier versions of the code with an older version to fix the mistakes.
- Git is a distributed version control system that enables software development teams to have multiple local copies of the project's codebase independent of each other.
- These copies, or branches, can be created, merged, and deleted quickly, empowering teams to experiment, with little compute cost, before merging into the main branch (sometimes referred to as the master branch).
- Git is known for its speed, workflow compatibility, and open source foundation.
- Most Git actions only add data to the database, and Git makes it easy to undo changes during the three main states.

Git has three file states: modified, staged, and committed.

1. A modified file has been changed but isn't committed to the database yet.
2. A staged file is set to go into the next commit.
3. When a file is committed, the data has been stored in the database.

With Git, software teams can experiment without fearing that they'll create lasting damage to the source code, because teams can always revert to a previous version if there are any problems.

Q10. Demonstrate creation of repositories in git.