

Experiment 8: Change specification and use any SCM Tool to make different versions

Learning Objective: Students will be able to change the version of the project.

Tools: GitHub

Theory:

Software configuration management: The traditional software configuration management (SCM) process is looked upon by practitioners as the best solution to handling changes in software projects. It identifies the functional and physical attributes of software at various points in time, and performs systematic control of changes to the identified attributes for the purpose of maintaining software integrity and traceability throughout the software development life cycle.

The SCM process further defines the need to trace changes, and the ability to verify that the final delivered software has all of the planned enhancements that are supposed to be included in the release. It identifies four procedures that must be defined for each software project to ensure that a sound SCM process is implemented. They are:

1. Configuration identification
2. Configuration control
3. Configuration status accounting
4. Configuration audits

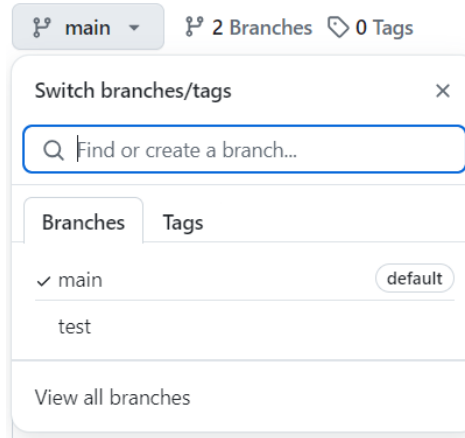
These terms and definitions change from standard to standard, but are essentially the same.

- Configuration identification is the process of identifying the attributes that define every aspect of a configuration item. A configuration item is a product (hardware and/or software) that has an end-user purpose. These attributes are recorded in configuration documentation and baselined. Baselining an attribute forces formal configuration change control processes to be effected in the event that these attributes are changed.
- Configuration change control is a set of processes and approval stages required to change a configuration item's attributes and to re-baseline them.
- Configuration status accounting is the ability to record and report on the configuration baselines associated with each configuration item at any moment of time.
- Configuration audits are broken into functional and physical configuration audits. They occur either at delivery or at the moment of effecting the change. A functional configuration audit ensures that functional and performance attributes of a configuration item are achieved, while a physical configuration audit ensures that a configuration item is installed in accordance with the requirements of its detailed design documentation.


GitHub offers all of the distributed revision control and source code management (SCM) functionality of Git as well as adding its own features. Unlike Git, which is strictly a command-line tool, GitHub provides a Web-based graphical interface and desktop as well as mobile integration. It also provides access control and several collaboration features such as bug tracking, feature requests, task management for every project.

Output :

Version –



Repository –

 shashank26-cmd razorpay err solve		be52111 · 3 months ago	🕒 18 Commits
📁 public	home page		6 months ago
📁 src	razorpay err solve		3 months ago
📄 .eslintrc.cjs	home page		6 months ago
📄 .gitignore	home page		6 months ago
📄 README.md	added more comments		5 months ago
📄 index.html	payement		5 months ago
📄 package-lock.json	home page		6 months ago
📄 package.json	home page		6 months ago
📄 postcss.config.js	home page		6 months ago
📄 tailwind.config.js	home page		6 months ago
📄 vite.config.js	home page		6 months ago

Update from version1 –

main
All users
All time

Commits on Jan 4, 2024

razorpay err solve
 shashank26-cmd committed 3 months ago
 be52111

Commits on Nov 13, 2023

admin dashboard
 shashank26-cmd committed 5 months ago
 d8f76af

moving to admin
 shashank26-cmd committed 5 months ago
 2397ded

Commits on Nov 12, 2023

payment err solve
 shashank26-cmd committed 5 months ago
 b849cdd

success page
 shashank26-cmd committed 5 months ago
 f21269a

Update to version2 –

test
All users
All time

Commits on Apr 17, 2024

reference and contributions
 Krafty-Eaton committed 1 minute ago
 Verified eb8db73

Commits on Jan 27, 2024

Minor Updates
 Krafty-Eaton committed 3 months ago
 ad62621

Learning Outcomes: Students should have the ability to

LO1: Understand software configuration management

LO2: Able to change the version of the project

Outcomes: Upon completion of the course students will be able to configure and manage their project more efficiently and effectively.

Conclusion:

From this experiment we were able to understand what Software management configuration is, how it is done, its importance and the tools used in it. Thus we were able to change the version and manage our project more effectively and efficiently using GitHub and Git.

For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				