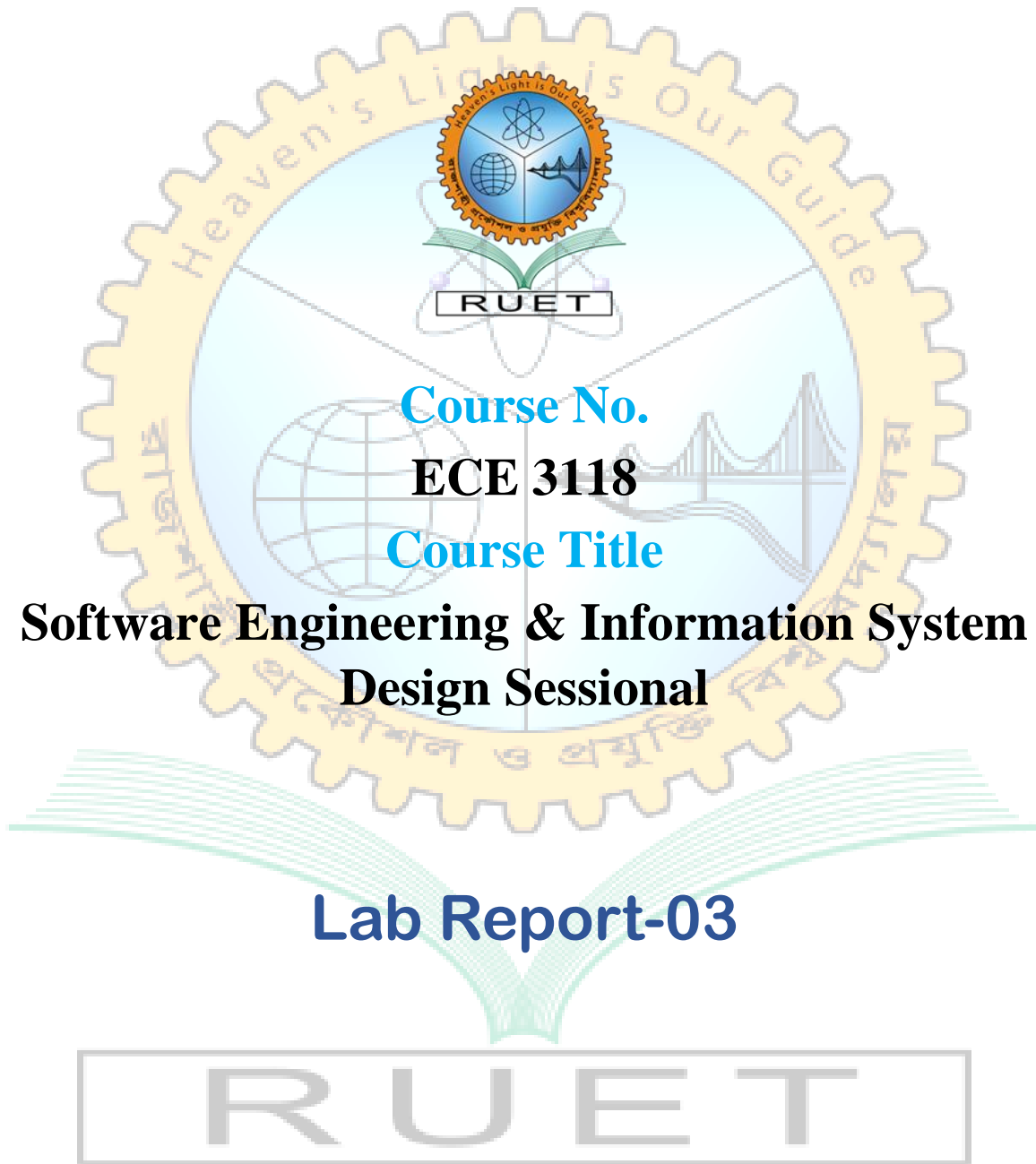


Heaven's Light is Our Guide

Rajshahi University of Engineering & Technology



Course No.

ECE 3118

Course Title

**Software Engineering & Information System
Design Sessional**

Lab Report-03

Submitted To

Oishi Jyoti
Assistant Professor,
Department of ECE, RUET

Submitted By

Tanha Farzana
Roll:2010022
3rd year, Odd Semester

3.1 Experiment No.: 03

3.2 Name of the Experiment: Study of different git commands.

3.3 Objectives:

1. To know about different git commands.
2. To learn importance of git commands.
3. To know different feature of version control.

3.4 Theory:

Git is a distributed version control system, which means that a local clone of the project is a complete version control repository. These fully functional local repositories make it easy to work offline or remotely. Developers commit their work locally, and then sync their copy of the repository with the copy on the server. This paradigm differs from centralized version control where clients must synchronize code with a server before creating new versions of code. Git's flexibility and popularity make it a great choice for any team. Many developers and college graduates already know how to use Git. Git's user community has created resources to train developers and Git's popularity make it easy to get help when needed. Nearly every development environment has Git support and Git command line tools implemented on every major operating system.[1]

Benefits of Git:

- Faster releases
- Built-in integration
- Strong community support
- Git works with any team
- Pull requests
- Branch policies

Some of the basic operations in Git are:

- ❖ Initialize
- ❖ Add
- ❖ Commit
- ❖ Pull
- ❖ Push

3.5 Git Commands:

Git init Command: This command is used to start a new repository.

Git config Command: This command sets the author's name and email address respectively to be used with my commits.

Git status Command: The git status command displays the state of the working directory and the staging area.

Git add Command: The git add command adds a change in the working directory to the staging area. It tells Git that we want to include updates to a particular file in the next commit.

Git commit Command: This command records or snapshots the file permanently in the version history.

Git log Command: Git log helps us see the past commits which helps to see who did what in Git and the repository. It helps us track the changes that happened in the repository.

Git branch Command: Branches allow us to work on different parts of a project without impacting the main branch.

Git show Command: Git-show is a command line utility that is used to view expanded details on Git objects such as blobs, trees, tags, and commits.

3.6 Discussion & Conclusion:

Git revolutionized version control by providing a decentralized, efficient, and reliable system for managing code. Its branching and merging capabilities enable seamless collaboration. Git's popularity and widespread adoption testify to its effectiveness in improving productivity, facilitating team coordination, and ensuring the integrity of software projects.

References:

- [1] "What is Git? - Azure DevOps | Microsoft Learn." <https://learn.microsoft.com/en-us/devops/develop/git/what-is-git#benefits-of-git> (accessed September. 16, 2024).
- [2] "Top 20 Git Commands With Examples - DZone." <https://dzone.com/articles/top-20-git-commands-with-examples> (accessed September. 17, 2024).