Report No. 03: Merge Conflicts in Git

Course Code: CSE 404
Course Title: Software Engineering and ISD Laboratory

Submitted by

SHANJIDA ALAM (ID: 353)

Submitted to

Dr. Md. MUSFIQUE ANWAR, Professor

Dr. Md. HUMAYUN KABIR, Professor

Computer Science and Engineering

Jahangirnagar University

Dhaka, Bangladesh

September 12, 2024

Contents

1. INTRODUCTION

If we have worked on a team project in Git, we have probably faced a problem called **"merge conflicts"**. It's like an unexpected problem for us when we think everything is going smoothly. If something changes in our code and some else's changes don't adjust together, then the merge conflict situation should arise.

But don't fear! Merge conflicts are a normal part of working with others on code. With the right approach, we can fix them quickly and get back to what matters—coding.

2. OBJECTIVE

The objectives are:

- Learn what merge conflicts are, and how they can create a problem in our code.
- Figure out why merge conflicts occur in code.
- Learn how to identify the problem and hoe to resolve the merge conflicts.
- Know what not to do.

3. WHAT IS MERGE CONFLICT?

A merge conflict happens when Git can't automatically figure out how to combine changes in the code from different commits. This usually happens when multiple developers are working on the same file, and Git isn't sure which changes to keep. This typically occurs during a merge operation, where changes from different branches are combined.

4. HOW MERGE CONFLICT OCCURS

Root causes of merge conflicts, such as:

- When multiple developers make changes to the same file at the same time.
- When different branches have changes that don't agree with each other.
- Renaming or deleting files.

5. CREATING A MERGE CONFLICT

To show a simple example of how a merge conflict can happen, we can manually trigger a merge conflict from the following set of commands in GIT bash terminal :

- **Step 1:** At first, create a repository in my GitHub account.Here I attach my repository link: https://github.com/shanjida-alam/Merge-conflicts
- **Step 2:** Clone the repository in my local machine.

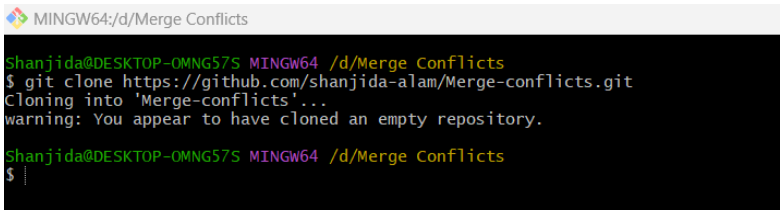

Figure 5.1: Screenshots of git clone

- **Step 3:** Create a text file and some content add in it and push this file in my repository.
- **Step 4:** Now creating a new branch in my repository.

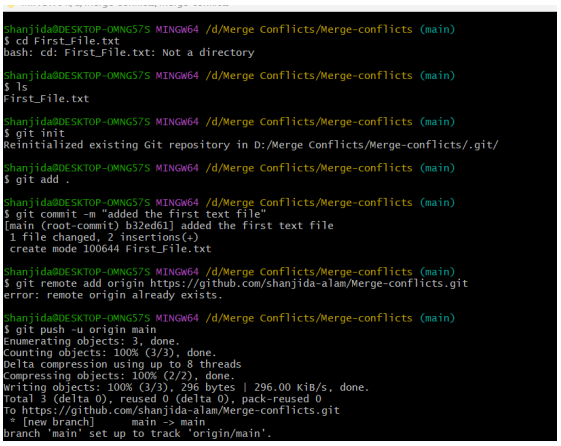

Figure 5.2: Screenshots of push the code in my git hub repository

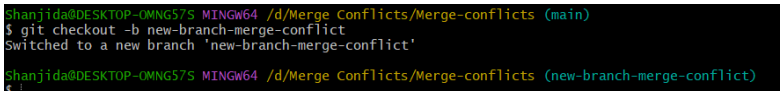

Figure 5.3: Screenshots of creating new branch

- **Step 5:**

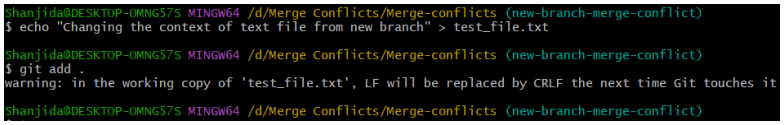

Figure 5.4: Screenshots of add text in the same file in the new creating branch

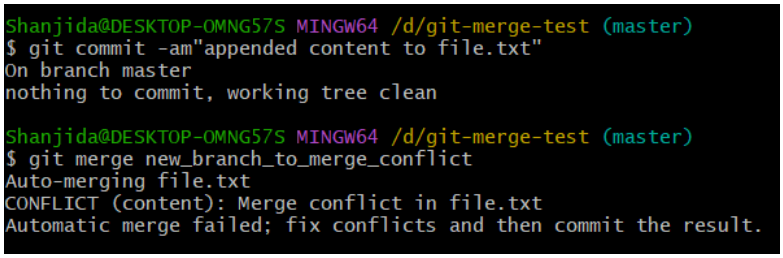

Figure 5.5: Screenshots of trying to merge the branch and creating a conflict