SCD-Lab

Lab#8

Participants:

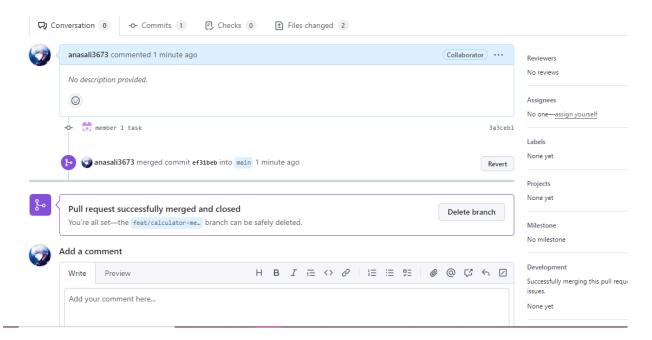
- 1. Anas Ali 22F-3673
- 2. Umair Altaf 22F-3737
- 3. Anas Altaf 22F-3639

Git Tasks:

Task 01:

Member 1:

```
A 2
      return a+b;
     return a*b;
      Main m= new Main();
        int sum=0;
        System.out.println("sum of two nums");
        sum=m.add( a: 3, b: 4);
        System.out.println(sum);
        int mul=0;
        System.out.println("\n multiplication of two nums");
        mul=m.multiplication( a: 4, b: 8);
        System.out.println(mul);
19
```



Member 2:

```
PS C:\Users\Umair\Desktop\calculator> git checkout -b validation
Switched to a new branch 'validation'

PS C:\Users\Umair\Desktop\calculator> git branch
main

* validation
```

```
PS C:\Users\Umair\Desktop\calculator> git branch -d feat/validation
Deleted branch feat/validation (was dd42821).
PS C:\Users\Umair\Desktop\calculator> git checkout feat/validation
error: pathspec 'feat/validation' did not match any file(s) known to git
PS C:\Users\Umair\Desktop\calculator> git checkout -b feat/validation
PS C:\Users\Umair\Desktop\calculator> git add .
PS C:\Users\Umair\Desktop\calculator> git status
Changes to be committed:
Delta compression using up to 4 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (10/10), 1010 bytes | 505.00 KiB/s, done.
Total 10 (delta 6), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (6/6), completed with 4 local objects.
remote: Create a pull request for 'feat/validation' on GitHub by visiting:
remote:
              https://github.com/umairaltaf982/calculator_git/pull/new/feat/validation
To https://github.com/umairaltaf982/calculator_git.git
                      feat/validation -> feat/validation
* [new branch]
PS C:\Users\Umair\Desktop\calculator>
   feat/validation had recent pushes 3 seconds ago
                                                                             Compare & pull request
                                                  Q Go to file
                                                                           Add file 🔻
                                                                                      <> Code →
   ្រំ main 🔻
              ያያ 2 Branches ♥ 0 Tags
   🕝 anasali3673 Merge pull request #1 from umairaltaf982/feat/calculator-methods \: 📟
                                                                     ef31beb · 19 minutes ago 5 Commits
   .idea
                                         member 1 task
                                                                                    21 minutes ago
   src
                                         member 1 task
                                                                                    21 minutes ago
```

Initial commit

Initial commit

Update README.md

42 minutes ago

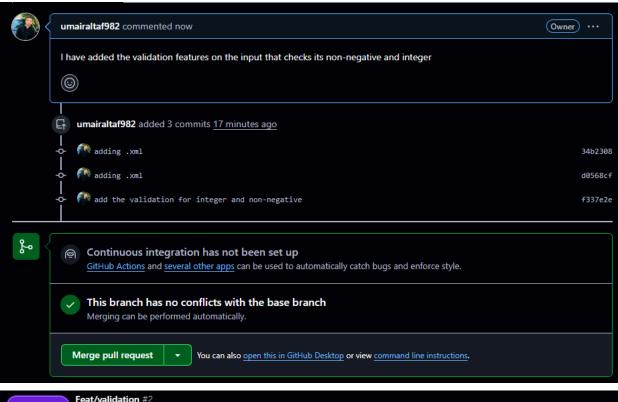
25 minutes ago

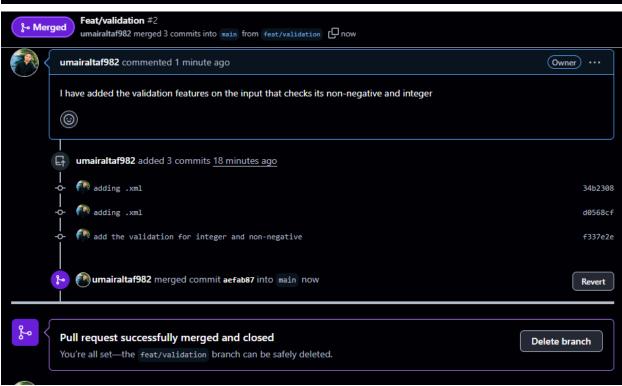
42 minutes ago

.gitignore

README.md

alculator.iml



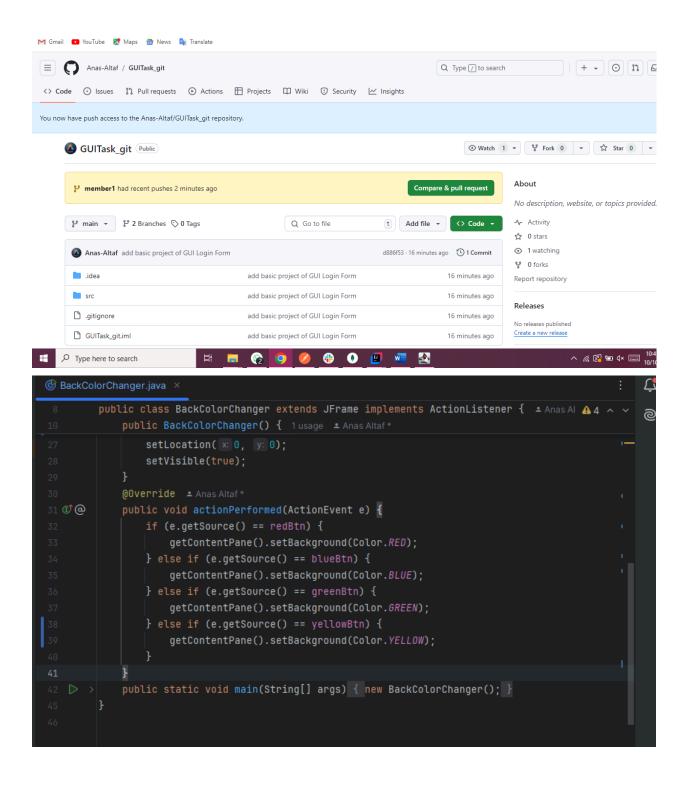


Task 02:

```
F:\University Tasks\FAST-BSE-5B\SCD-Lab\Lab-08\GUITask_qit qit:[main]
git commit -m "add basic project of GUI Login Form"
[main (root-commit) d886f53] add basic project of GUI Login Form
 7 files changed, 108 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 .idea/.gitignore
 create mode 100644 .idea/misc.xml
 create mode 100644 .idea/modules.xml
 create mode 100644 .idea/vcs.xml
 create mode 100644 GUITask_git.iml
 create mode 100644 src/BackColorChanger.java
F:\University Tasks\FAST-BSE-5B\SCD-Lab\Lab-08\GUITask_git git:[main]
git push origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (11/11), 2.04 KiB | 697.00 KiB/s, done.
Total 11 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
                   main -> main
 * [new branch]
F:\University Tasks\FAST-BSE-5B\SCD-Lab\Lab-08\GUITask_git git:[main]
```

Member 1:

```
Local × + v
     Terminal
    Enumerating objects: 7, done.
    Counting objects: 100% (7/7), done.
   Delta compression using up to 4 threads
   Compressing objects: 100% (3/3), done.
   Writing objects: 100% (4/4), 431 bytes | 143.00 KiB/s, done.
   Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
   remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
   remote:
    remote: Create a pull request for 'member1' on GitHub by visiting:
                https://github.com/Anas-Altaf/GUITask_git/pull/new/member1
    remote:
To https://github.com/Anas-Altaf/GUITask_git.git
    * [new branch] member1 -> member1
   PS C:\Users\ALI COMPUTERS\Desktop\GUITask_git>
```



Member 2:

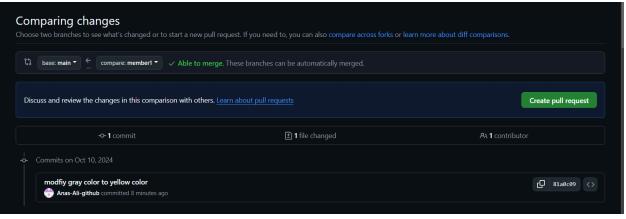
```
> import ...
 private JButton redBtn, blueBtn, orangeBtn, grayBtn; 4 usages
    setTitle("Color Changer");
        setLayout(new FlowLayout());
        redBtn = new JButton( text: "Red");
        blueBtn = new JButton( text: "Blue");
        orangeBtn = new JButton( text: "Orange");
        grayBtn = new JButton( text: "Gray");
        redBtn.addActionListener( |: this);
        blueBtn.addActionListener( : this);
        orangeBtn.addActionListener( |: this);
        grayBtn.addActionListener( |: this);
        add(blueBtn);
        add(orangeBtn);
        add(grayBtn);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize( width: 300, height: 200);
        setLocation(x:0, y:0);
        setVisible(true);
     public void actionPerformed(ActionEvent e) {
         if (e.getSource() == redBtn) {
```

```
public void actionPerformed(ActionEvent e) {
    if (e.getSource() == redBtn) {
        getContentPane().setBackground(Color.RED);
    } else if (e.getSource() == blueBtn) {
        getContentPane().setBackground(Color.BLUE);
    } else if (e.getSource() == orangeBtn) {
            getContentPane().setBackground(Color.ORANGE);
    } else if (e.getSource() == grayBtn) {
            getContentPane().setBackground(Color.GRAY);
    }
}

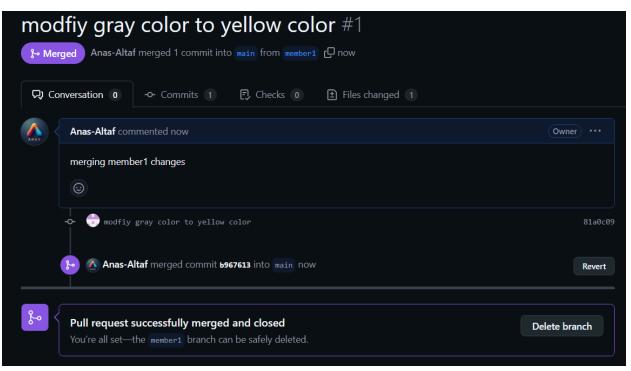
public static void main(String[] args) { new BackColorChanger(); }
}
```

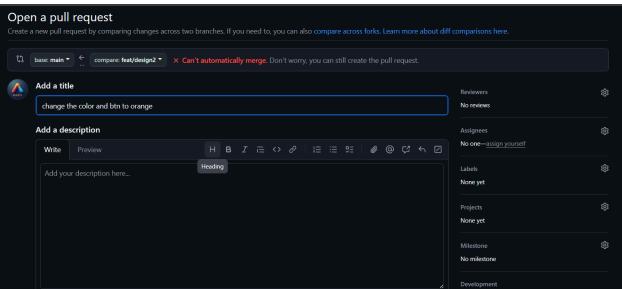
```
On branch feat/design2
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
       modified: src/BackColorChanger.java
no changes added to commit (use "git add" and/or "git commit -a")
PS C:\Users\Umair\Desktop\GUITask_git> git branch
 main
PS C:\Users\Umair\Desktop\GUITask_qit> qit add .
PS C:\Users\Umair\Desktop\GUITask_git> git commit -m "change the color and btn to orange"
[feat/design2 4c339e8] change the color and btn to orange
1 file changed, 6 insertions(+), 6 deletions(-)
PS C:\Users\Umair\Desktop\GUITask_git> git push origin feat/design2
Enumerating objects: 7, done.
Counting objects: 100\% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 428 bytes | 142.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'feat/design2' on GitHub by visiting:
              https://github.com/Anas-Altaf/GUITask_git/pull/new/feat/design2
remote:
remote:
To https://github.com/Anas-Altaf/GUITask_git.git
* [new branch] feat/design2 -> feat/design2
PS C:\Users\Umair\Desktop\GUITask_git>
```

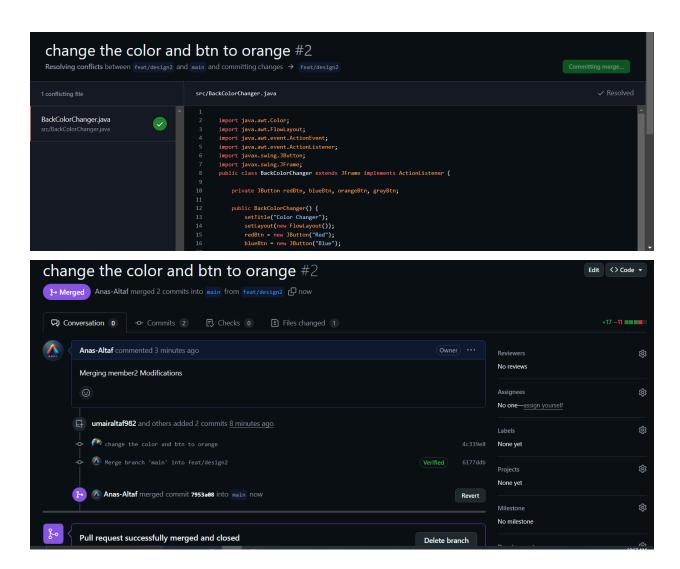
Member 3:











Task 03:

Member 1:

```
Terminal Local × + >

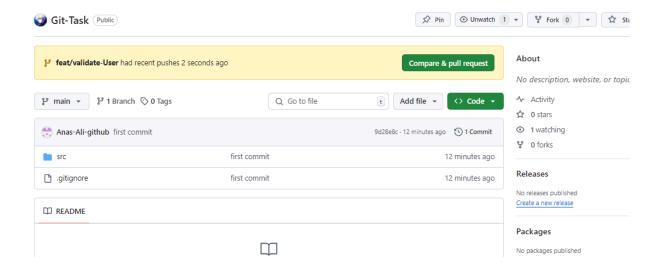
create mode 100644 .idea/vcs.xml
create mode 100644 .idea/vcs.xml
PS C:\Users\ALI COMPUTERS\IdeaProjects\Git-merging-task> git push origin feat/validate-User
Enumerating objects: 10% (13/13), done.
Delta compression using up to 4 threads
Compression objects: 100% (18/18), done.
Writing objects: 100% (18/10), 2.20 KiB | 250.00 KiB/s, done.
Total 10 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:

remote: Create a pull request for 'feat/validate-User' on GitHub by visiting:
remote: https://github.com/anasali3673/Git-Task/pull/new/feat/validate-User

remote:
To https://github.com/anasali3673/Git-Task.git

* [new branch] feat/validate-User -> feat/validate-User
PS C:\Users\ALI COMPUTERS\IdeaProjects\Git-merging-task>

①
```



Member 2:

```
nothing added to commit but untracked files present (use "git add" to track)
warning: in the working copy of '.idea/vcs.xml', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Umair\Desktop\Git-Task> git commit -m "add anonymous file"
[main 5faab8e] add anonymous file
2 files changed, 12 insertions(+)
create mode 100644 .idea/.gitignore
PS C:\Users\Umair\Desktop\Git-Task> git status
Your branch is ahead of 'origin/main' by 1 commit.
 (use "git push" to publish your local commits)
nothing to commit, working tree clean
PS C:\Users\Umair\Desktop\Git-Task> git checkout -b feat/registration
Switched to a new branch 'feat/registration'
PS C:\Users\Umair\Desktop\Git-Task> git branch
PS C:\Users\Umair\Desktop\Git-Task> git pull origin feat/validate-User
remote: Enumerating objects: 13, done.
remote: Counting objects: 100% (13/13), done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 10 (delta 0), reused 10 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (10/10), 2.18 KiB | 25.00 KiB/s, done.
* branch
                     feat/validate-User -> FETCH_HEAD
 * [new branch]
```

```
CONFLICT (add/add): Merge conflict in .idea/.gitignore
Auto-merging .idea/vcs.xml
CONFLICT (add/add): Merge conflict in .idea/vcs.xml
Automatic merge failed; fix conflicts and then commit the result.
PS C:\Users\Umair\Desktop\Git-Task>
```

```
import java.util.HashSet;
import java.util.Set;
import java.util.regex.Pattern;

public class Main {

private Set<String> registeredUsernames = new HashSet<>(); 2 usages

public boolean validateUser(String username, String password, String email) { 1 usage
    if (username == null || username.isEmpty()) {
        log("Registration failed: Username cannot be null or empty.");
        return false;
}

if (password == null || password.isEmpty()) {
        log("Registration failed: Password cannot be null or empty.");
        return false;
}

if (email == null || email.isEmpty()) {
        log("Registration failed: Email cannot be null or empty.");
        return false;
}
```

```
if (registeredUsernames.contains(username)) {
    log("Registration failed: Username '" + username + "' is already taken.");
    return false;
}

if (!isValidUsername(username)) {
    log("Registration failed: Invalid username format.");
    return false;
}

if (!isValidPassword(password)) {
    log("Registration failed: Invalid password format.");
    return false;
}

if (!isValidEmail(email)) {
    log("Registration failed: Invalid email format.");
    return false;
}

registeredUsernames.add(username);
log("User registration successful: " + username);
    return true;
}
```

```
private boolean isValidUsername(String username) { 1 usage
    return username.matches( regex: "^[a-zA-ZO-9]{3,15}$");
}

private boolean isValidPassword(String password) { 1 usage
    return password.matches( regex: "^(?=.*[0-9])(?=.*[!@#$\%^&*])[A-Za-z\\d!@#$\%^&*]{8,}\$");
}

private boolean isValidEmail(String email) { 1 usage
    String emailRegex = "^[A-Za-zO-9+..-]+@[A-Za-zO-9.-]+\$";
    return Pattern.matches(emailRegex, email);
}

private void log(String message) { 8 usages
    System.out.println(message);
}
```

Member 3:

