A picture containing text

Description automatically generated



Student : Mustafa Dilawar Khan

Enrollment : 01-131232-074

Computer Programming Lab (Spring-2024)

Teacher: Engr. M Waleed

Department of Software Engineering

Bahria University, Islamabad

Lab Journal: 06

# Lab No: 06 Mastering Version Control with Git, GitHub, and Collaboration

## Introduction

This lab introduces a crucial aspect of modern software development: collaborating on Object-Oriented Programming (OOP) projects using version control systems, with a focus on Git and GitHub. In the realm of software engineering, the ability to work effectively in teams, manage code changes, and contribute to shared repositories is indispensable. This lab is designed to bridge the gap between theoretical OOP concepts and practical application in a team environment, leveraging industry-standard tools.

## Tools Used

Git and Github

# Lab Tasks:

### Task 01:

**Collaborative Class Project**

**• Objective:**

Work in teams to develop a part of an OOP project, using Git and GitHub for version control and collaboration.

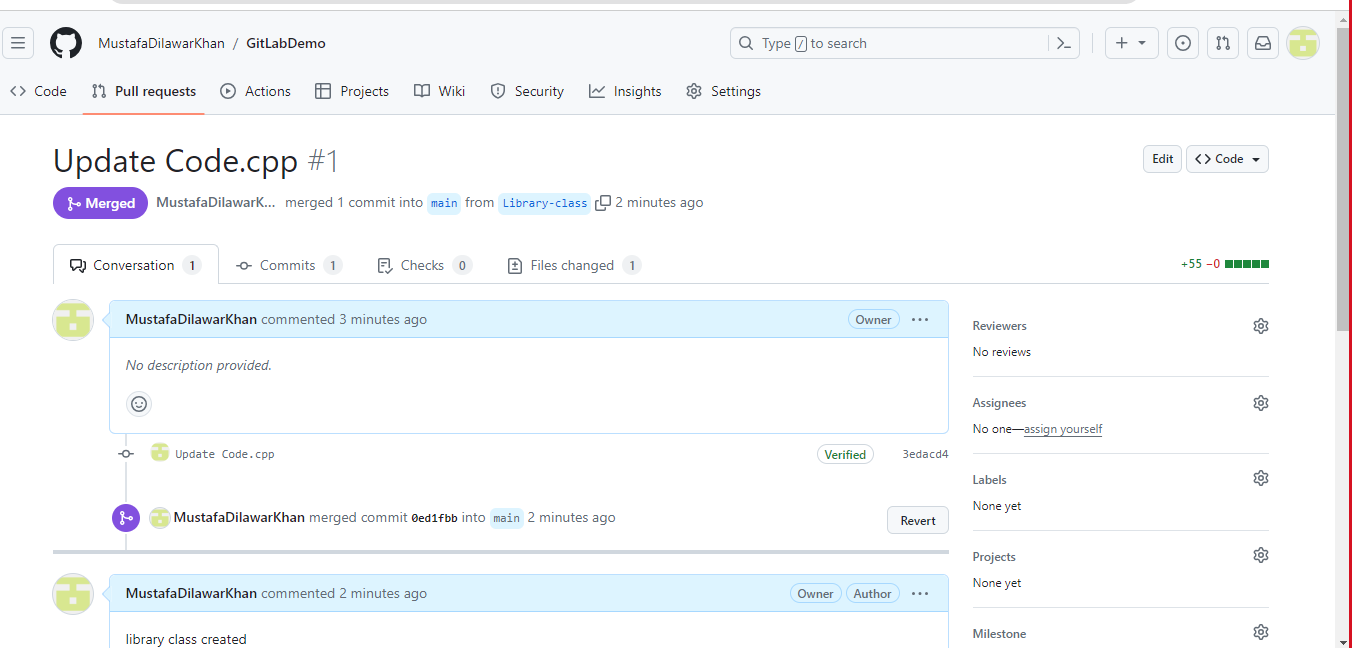
**• Instructions:**

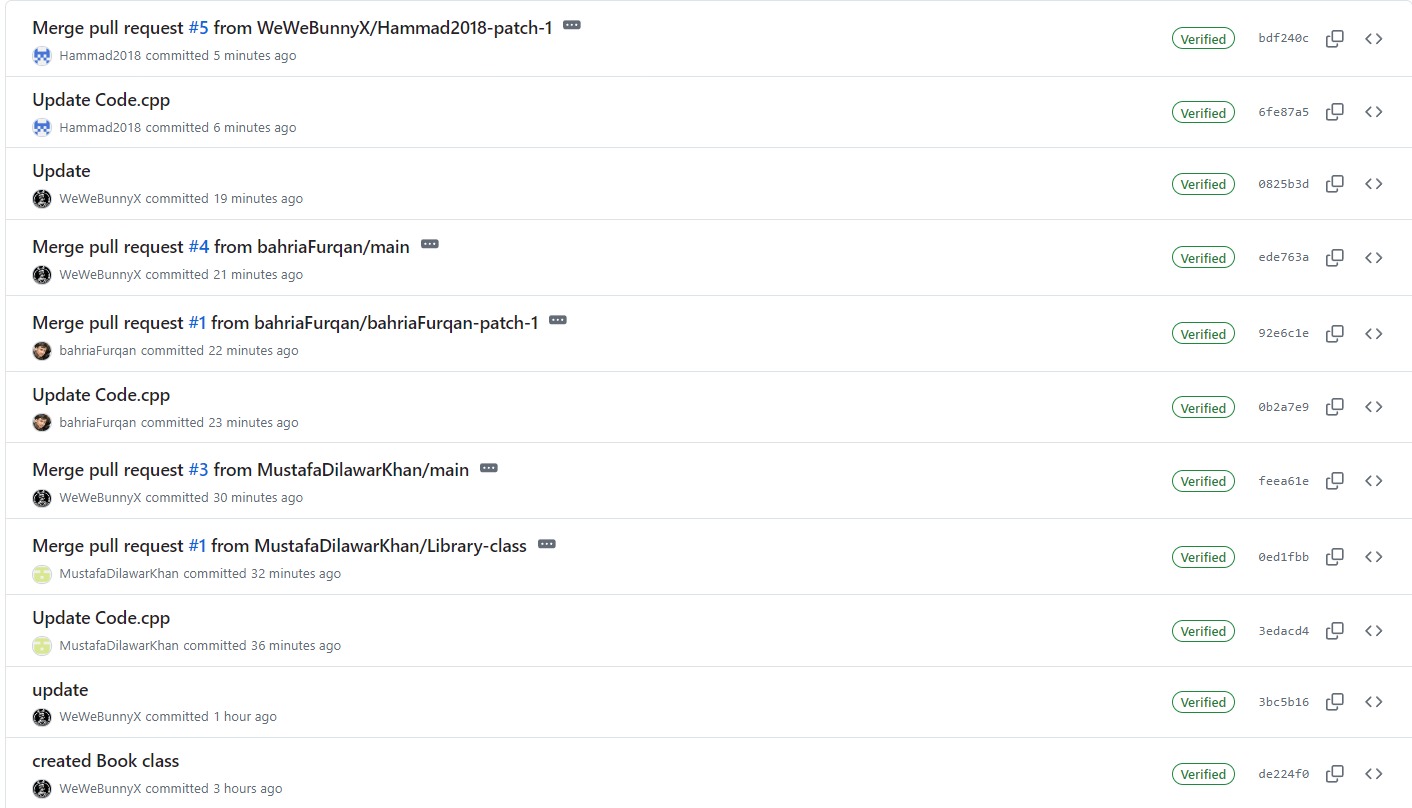
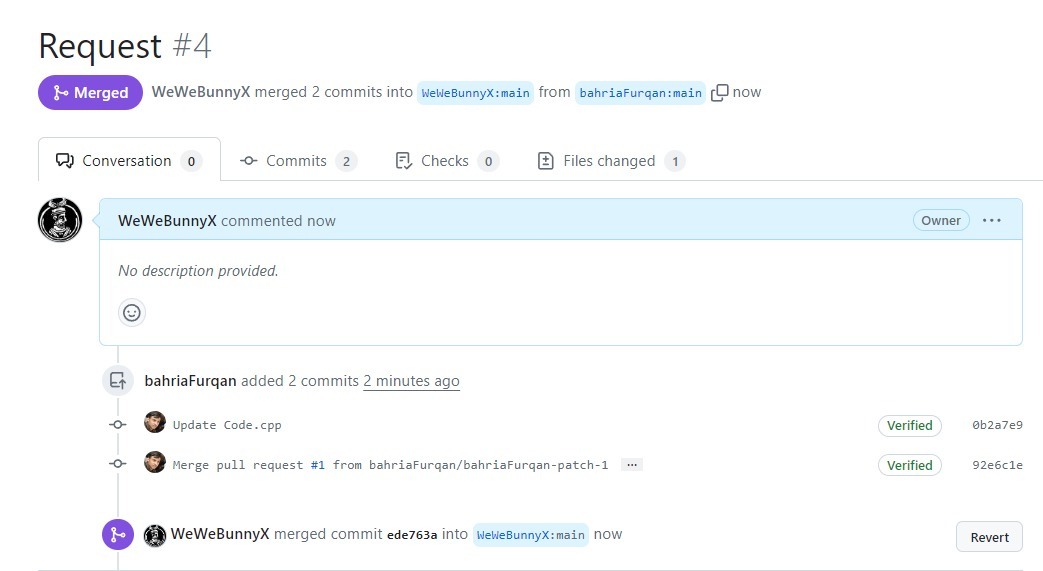
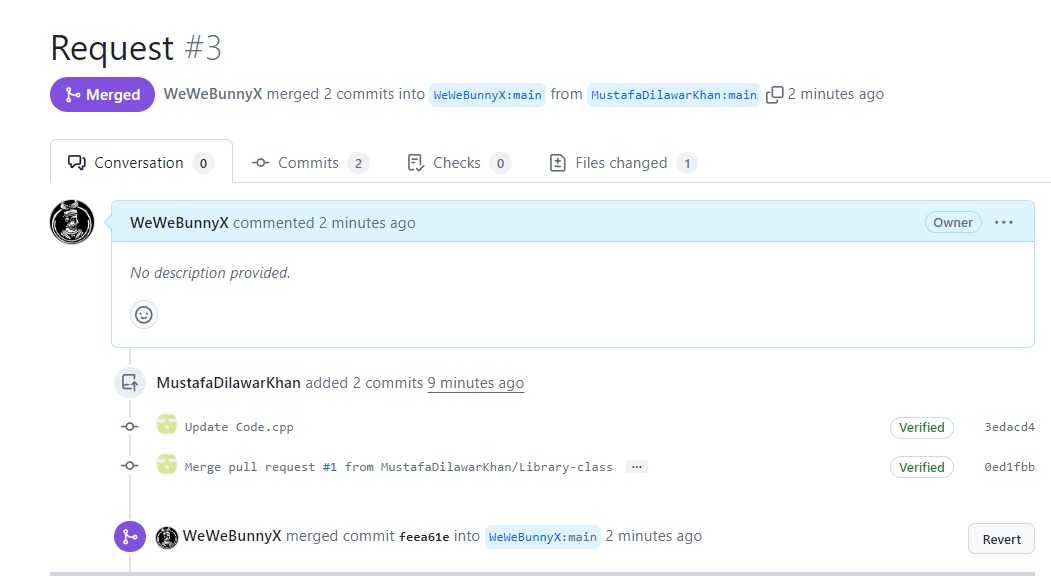
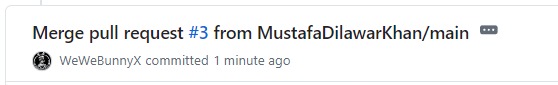
Each team member forks the project repository, clones their fork, creates a feature branch for their task, and opens a pull request to integrate their work

### Link:

<https://github.com/WeWeBunnyX/GitLabDemo/commits/main/>

## Screenshot:





# Home Task:

### Task 02:

### Code:

#include <iostream>

#include <vector>

#include <string>

#include <ctime>

Using namespace std;

Class Reservation {

Private:

String name;

String timeSlot;

Public:

Reservation(string n, string ts) : name(n), timeSlot(ts) {}

String getName() const {

Return name;

}

String getTimeSlot() const {

Return timeSlot;

}

};

Class Restaurant {

Private:

Vector<string> availableTimeSlots;

Vector<Reservation> reservations;

Public:

Void initializeTimeSlots() {

// Initialize available time slots

// Example: 12:00 PM, 12:30 PM, 1:00 PM, …

availableTimeSlots.push\_back(“12:00 PM”);

availableTimeSlots.push\_back(“12:30 PM”);

availableTimeSlots.push\_back(“1:00 PM”);

availableTimeSlots.push\_back(“1:30 PM”);

availableTimeSlots.push\_back(“2:00 PM”);

availableTimeSlots.push\_back(“2:30 PM”);

availableTimeSlots.push\_back(“3:00 PM”);

availableTimeSlots.push\_back(“3:30 PM”);

availableTimeSlots.push\_back(“4:00 PM”);

availableTimeSlots.push\_back(“4:30 PM”);

availableTimeSlots.push\_back(“5:00 PM”);

}

Void displayAvailableTimeSlots() {

Cout << “Available Time Slots:” << endl;

For (const auto& slot : availableTimeSlots) {

Cout << slot << endl;

}

}

Bool makeReservation(string name, string timeSlot) {

// Check if the time slot is available

For (const auto& reservation : reservations) {

If (reservation.getTimeSlot() == timeSlot) {

Return false; // Time slot already reserved

}

}

// Make the reservation

Reservations.push\_back(Reservation(name, timeSlot));

// Remove the reserved time slot from available time slots

For (auto it = availableTimeSlots.begin(); it != availableTimeSlots.end(); ++it) {

If (\*it == timeSlot) {

availableTimeSlots.erase(it);

break;

}

}

Return true;

}

Void displayReservations() {

Cout << “Reservations:” << endl;

For (const auto& reservation : reservations) {

Cout << “Name: “ << reservation.getName() << “, Time Slot: “ << reservation.getTimeSlot() << endl;

}

}

};

Int main() {

// Initialize Restaurant object

Restaurant restaurant;

// Initialize available time slots

Restaurant.initializeTimeSlots();

// Display available time slots

Restaurant.displayAvailableTimeSlots();

// Get user input for reservation

String name, timeSlot;

Cout << “Enter your name: “;

Getline(cin, name);

Cout << “Enter preferred time slot: “;

Getline(cin, timeSlot);

// Customer makes a reservation

Bool success = restaurant.makeReservation(name, timeSlot);

If (success) {

Cout << “Reservation successful!” << endl;

} else {

Cout << “Sorry, the selected time slot is not available.” << endl;

}

// Display all reservations

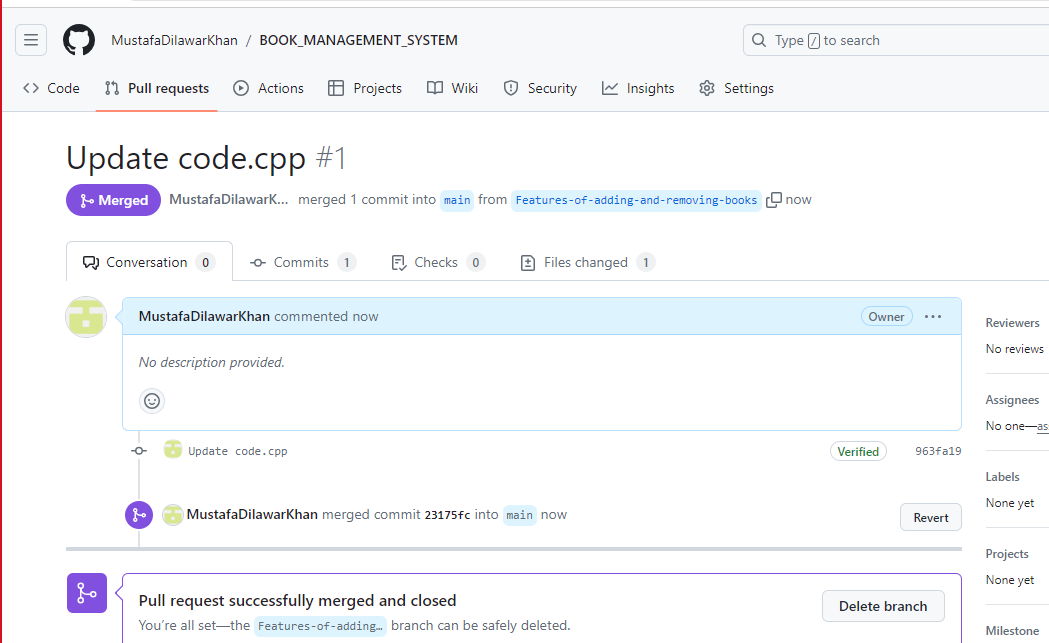
Restaurant.displayReservations();

Return 0;

}

<https://github.com/MustafaDilawarKhan/BOOK_MANAGEMENT_SYSTEM>

### Screenshot:



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

