

**Team Members:**

**Safi Ullah (B22F0549SE031)**

**Sharyar Naveed (B22F0782SE014)**

**Muhammad Moin (B22F1629SE148)**

**Course:**

**Software Construction and Development  
(COMP – 370)**

**Program:**

**Software Engineering – 22 – RED**

**Instructor:**

**Dr. Nabeel Ahmed**

**Submitted Date:**

**29 – January – 2024**

**Title:**

**UI-Interface Document**

# User Interface Document

---

## Title:

## GitHub Integration for Agile Team Release Management

---

### Introduction

Agile development teams need an efficient and automated way to track software releases, manage tasks, and monitor CI/CD pipelines. This project is integrated with GitHub APIs to fetch release data, track tasks, and monitor CI/CD pipeline statuses through a centralized dashboard.

---

The interface of the GitHub Integration for Agile Team Release Management system will be modern, user-friendly, and intuitive, designed to provide real-time insights into repository updates, CI/CD pipeline status, and release management.

- **Dashboard Interface**

Upon logging in via GitHub OAuth authentication, users will land on a centralized dashboard that displays an overview of ongoing releases, task statuses, and CI/CD pipeline performance. The interface will be visually rich, utilizing graphs, charts, and color-coded indicators to represent different statuses (e.g., green for successful builds, red for failures, and yellow for ongoing processes).

- **Release Tracking Section**

A dedicated section will list all tracked GitHub repositories, showing key details such as release versions, updated timestamps, commit history, and associated tasks. Users can filter and sort releases based on their status (e.g., pending, deployed, failed).

- **Task Management Panel**

Each release will have an associated task panel, where tasks can be created, updated, or assigned to team members. Users will see progress bars representing the completion status of different tasks, along with options to update status (Pending, In Progress, Completed).

- **CI/CD Status Display**

A real-time CI/CD monitoring section will fetch data from GitHub Actions to show the latest build and deployment statuses. If a build fails, users will receive an alert, and an option to view logs for debugging will be available. Successful builds will display deployment timestamps and affected files.

- **Notification Center**

A notification panel will provide real-time alerts for critical events, including new commits, build failures, and successful deployments. Users can choose to receive notifications via Slack, in-app alerts, or email.

- **User Settings & Permissions**

Admins will have access to a user management section to assign roles and permissions. Users can modify notification preferences, integrate additional repositories, and configure build trigger settings based on their workflow.

- **Overall Look & Feel**

The system will adopt a clean, responsive UI, with a dark and light mode option, making it visually appealing for extended use. It will be optimized for desktop and tablet views, ensuring a seamless experience for Agile teams working in fast-paced environments.