



# BAHRIA UNIVERSITY, (Karachi Campus)

## Department of Software Engineering

### Assignment 1 - Spring 2024

---

COURSE TITLE:	<b>Intr. to Software Engineering</b>	COURSE CODE:	<b>SEN-210</b>
Class:	<b>BSE-II (A)</b>	Time Allowed:	<b>1 Week.</b>
Course Instructor:	<b>ENGR. RAHEMEEN</b>	Max. Marks:	<b>5 marks</b>
Submission Date:	<b>30-03-2024</b>	Student Name:	<b>Abdullah</b>

---

#### Note:

- Strictly follow the given deadline.
- Assignment must be submitted on LMS.
- Must be in PDF file only.

#### Question No. 1

[CLO2: 5 Marks]

Explain and evaluate different popular tools and technologies used to support the SDLC process (e.g., version control systems, and project management tools). Choose any two of them and discuss their features, benefits, and limitations.

#### Solution:

Two popular tools used to support the Software Development Life Cycle (SDLC) process are version control systems (VCS) and project management tools. Let's discuss Git as a version control system and Jira as a project management tool, highlighting their features, benefits, and limitations.

### 1) Git (Version Control System):

#### Features:

- Distributed Version Control: Git allows developers to have their own local repository, enabling them to work offline and commit changes locally before pushing them to the central repository.
- Branching and Merging: Git facilitates easy branching for parallel development efforts, and merging features allow developers to integrate changes from different branches efficiently.
- Lightweight and Fast: Git is known for its speed and efficiency in handling large projects with a vast number of files and commits.
- Open Source and Extensible: Git is open source, widely adopted, and supports integration with various third-party tools and services.
- Support for Collaboration: Git enables multiple developers to work on the same project simultaneously by managing conflicts and facilitating code reviews.

#### Benefits:

- Version Control: Git tracks changes to source code, allowing developers to revert to previous versions if needed and providing a history of changes for better understanding and collaboration.
- Collaboration: Git's distributed nature and branching features promote collaboration among developers and teams working on different aspects of a project.

- **Flexibility:** Git can be used for projects of any size and complexity, from small personal projects to large enterprise-level applications.
- **Community Support:** Being open-source, Git benefits from a vast community of users and contributors who continuously improve the tool and provide support.

### **Limitations:**

- **Learning Curve:** Git has a steep learning curve for beginners due to its command-line interface and various commands and options.
- **Complex Merging:** While Git provides powerful merging capabilities, complex branching structures and extensive changes can sometimes lead to challenging merger conflicts that require careful resolution.
- **Storage Requirements:** Maintaining a complete history of changes in Git repositories can lead to significant storage requirements, especially for large projects with frequent commits.

## **2) Jira (Project Management Tool):**

### **Features:**

- **Issue Tracking:** Jira allows teams to create and track issues, tasks, bugs, and user stories throughout the development process.
- **Agile Methodologies Support:** Jira supports agile methodologies like Scrum and Kanban, providing boards, sprints, backlogs, and customizable workflows.
- **Customization:** Jira offers extensive customization options for workflows, issue types, fields, and dashboards to tailor the tool to specific project requirements.
- **Integration:** Jira integrates with a wide range of development tools, version control systems, and communication platforms to streamline collaboration and project management.
- **Reporting and Analytics:** Jira provides reporting and analytics features to track project progress, identify bottlenecks, and make data-driven decisions.

### **Benefits:**

- **Centralized Project Management:** Jira serves as a central hub for project management, allowing teams to plan, track, and collaborate on tasks and activities from a unified platform.
- **Agile Support:** Jira's support for agile methodologies enables teams to adopt iterative development practices, prioritize tasks, and deliver value to stakeholders incrementally.
- **Customization:** Jira's flexibility and customization options enable teams to adapt the tool to their specific workflows, processes, and project management practices.
- **Integration:** Jira's integration capabilities with other tools and systems enhance productivity and collaboration by reducing manual effort and streamlining data exchange.

### **Limitations:**

- **Complexity:** Jira can be complex to set up and configure, especially for teams new to agile methodologies or project management tools.
- **Cost:** While Jira offers a range of features and functionalities, the pricing structure can be a limitation for smaller teams or organizations with budget constraints.
- **Learning Curve:** Like any comprehensive project management tool, Jira has a learning curve for users, particularly those who are not familiar with agile concepts or software development workflows.