Department of Computer Engineering

Academic Term: First Term 2023-24

Class: T.E /Computer Sem – V / Software Engineering

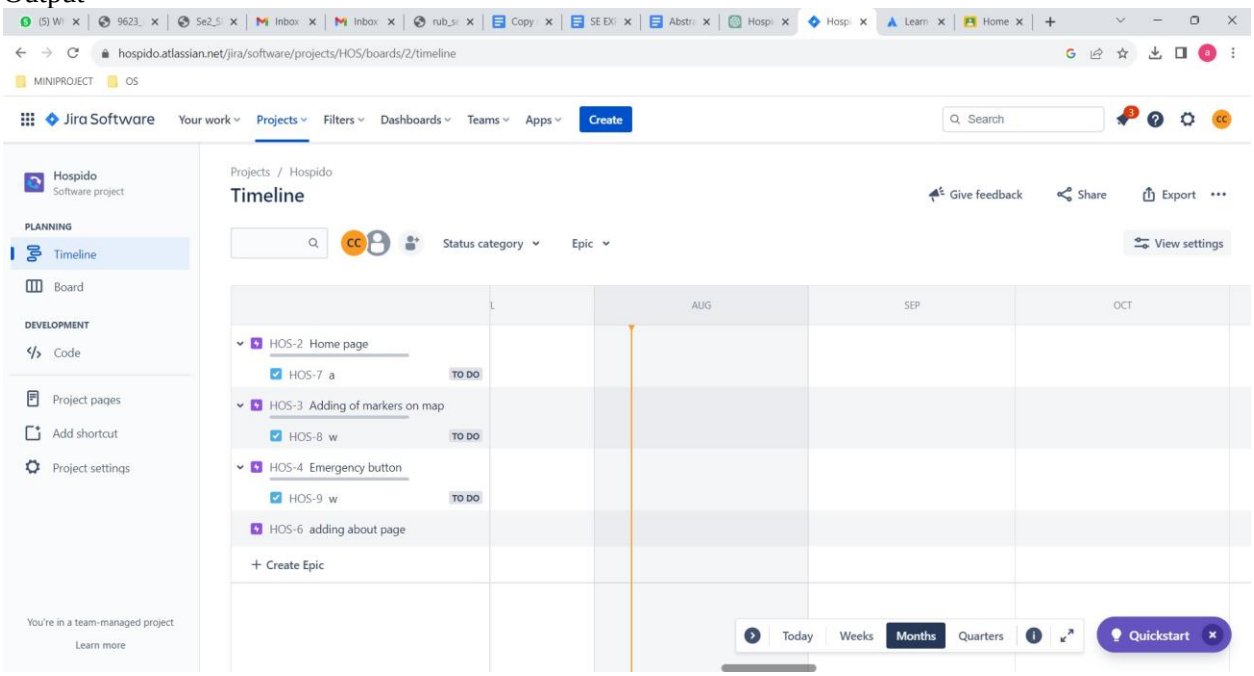
Practical No:	2
Title:	**Software Requirement Specification**
Date of Performance:	2-08-2023
Roll No:	9634
Team Members:	Aditya ,Rianna ,Alex,Chris

Rubrics for Evaluation:

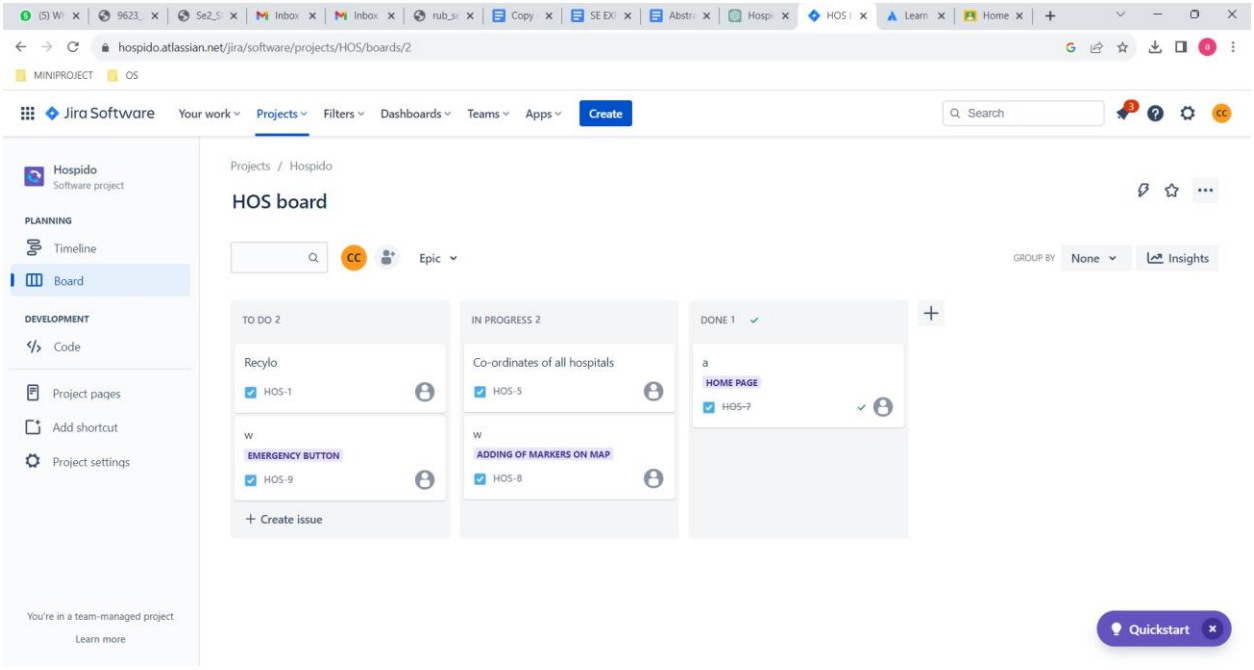
Sr. No	Performance Indicator	Excellent	Good	Below Average	Total Score
1	On time Completion & Submission (01)	01 (On Time)	NA	00 (Not on Time)	
2	Theory Understanding(02)	02(Correct)	NA	01 (Tried)	
3	Content Quality (03)	03(All used)	02 (Partial)	01 (rarely followed)	
4	Post Lab Questions (04)	04(done well)	3 (Partially Correct)	2(submitted)	

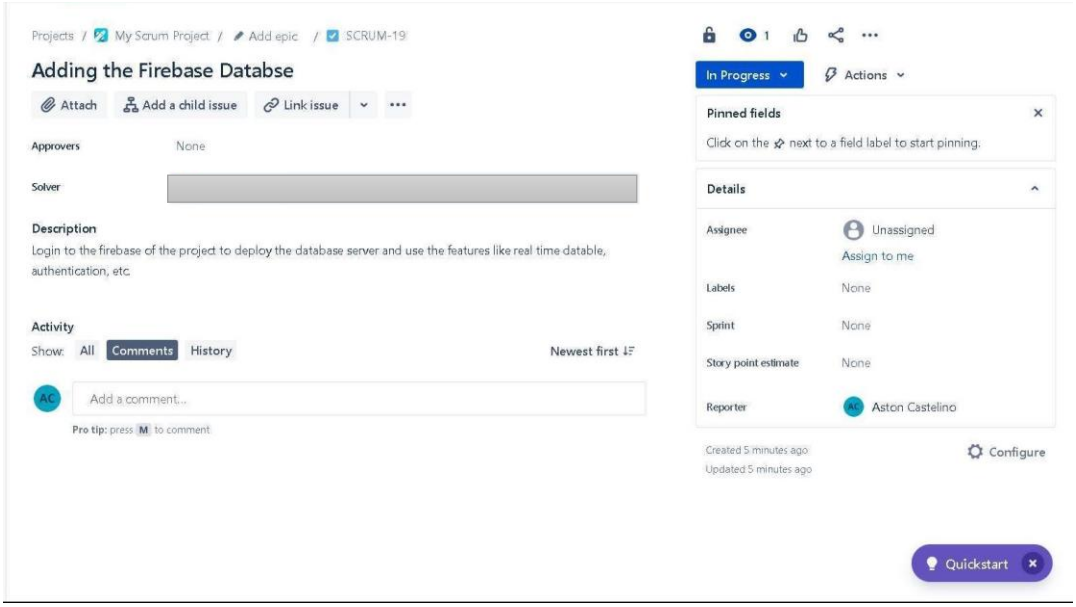
Signature of the Teacher:

Output



Board





Post lab

A)

Scrum has proven to be remarkably effective for managing software development projects when compared to traditional methodologies. Its success can be attributed to several key factors:

1. Flexibility: Scrum embraces changes in requirements, allowing teams to adapt quickly to evolving needs.

2. Incremental Delivery: Short sprints enable the delivery of valuable increments, leading to early feedback and faster time to market.

3. Collaboration: Regular ceremonies enhance communication between team members and stakeholders.

4. Transparency: Scrum ensures a shared understanding of project progress and goals.

5. Continuous Improvement: Retrospectives promote a culture of continuous learning and improvement.

Disadvatages

1. Learning Curve: Scrum requires a shift in mindset and practices, which can be challenging for teams accustomed to traditional project management methodologies.

2. Lack of Detailed Planning: Scrum's focus on adaptability may lead to limited detailed planning, making it challenging to estimate project timelines accurately.

3. Incomplete Requirements: With changing requirements, there is a risk of incomplete or evolving specifications, leading to potential scope creep.

4. Time-Boxed Sprints: Fixed-duration sprints may not be suitable for all projects, and some tasks may not be completed within a single sprint.

5. Resource Allocation: Scrum demands active involvement from all team members, which may strain resources

B) Analyse a Sprint Backlog in JIRA andi dentify any potential bottlenecks or issues that might hinder the team's progress during the sprint.

Some potential bottlenecks or issues that might hinder the team's progress during a sprint while using JIRA or similar tools:

1. Overloaded Sprint Backlog: A backlog with too many tasks can overwhelm the team and affect their ability to complete them within the sprint timeframe.

2. Unclear User Stories or Tasks: Ambiguous user stories can lead to confusion and delays as the team tries to understand and work on them.

3. Dependencies and Blocked Tasks: Unresolved dependencies or blocked tasks can impede progress and delay completion.

4. Inadequate Task Estimations: Incorrect or underestimated task estimations can lead to task overruns and affect sprint progress.

5. Scope Creep: Changes or additions to the sprint backlog during the sprint can disrupt the team's focus and affect planned work.

7. Insufficient Testing and Quality Assurance: Lack of adequate time for testing can lead to incomplete or buggy features.

8. Poor Prioritization: If the sprint backlog is not prioritized effectively, the team may spend time on less critical tasks, affecting overall progress.

C) Evaluate the role of the Scrum Master in handling conflicts within the development team and resolving impediments to maintain a smooth project flow.

The Scrum Master plays a critical role in handling conflicts within the development team and resolving impediments to maintain a smooth project flow in the scrum framework.
The following are a few impediments:-
1. Conflict Resolution: The Scrum Master acts as a mediator, creating a safe environment for open dialogue and constructive resolutions among team members, promoting a harmonious working atmosphere.

2. Obstacle Removal: The Scrum Master identifies and resolves impediments to ensure the team's smooth progress and adherence to the sprint goals, keeping the project on track.

3. Coaching and Empowerment: The Scrum Master empowers the team through coaching and self-organization guidance, fostering a sense of ownership and accountability, leading to increased productivity.

4. Facilitating Scrum Events: The Scrum Master ensures effective planning and execution of Scrum events, maximizing their value and enabling continuous improvement, driving optimal outcomes.

5. Protecting the Team: The Scrum Master shields the team from external disruptions and pressure, allowing them to focus on delivering high-quality increments, boosting motivation and efficiency.

6. Continuous Improvement: The Scrum Master fosters a culture of learning and continuous improvement, encouraging the team to adapt and enhance their processes, fostering innovation.