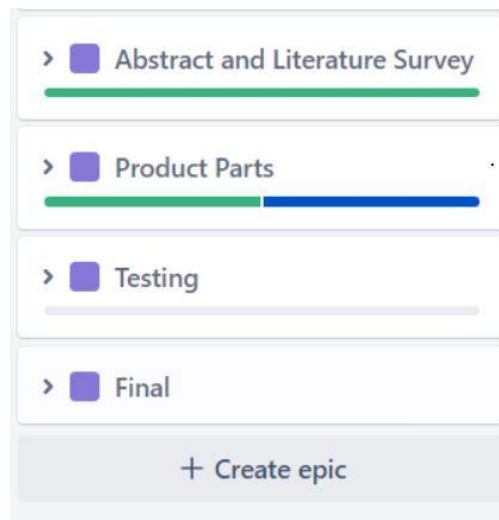


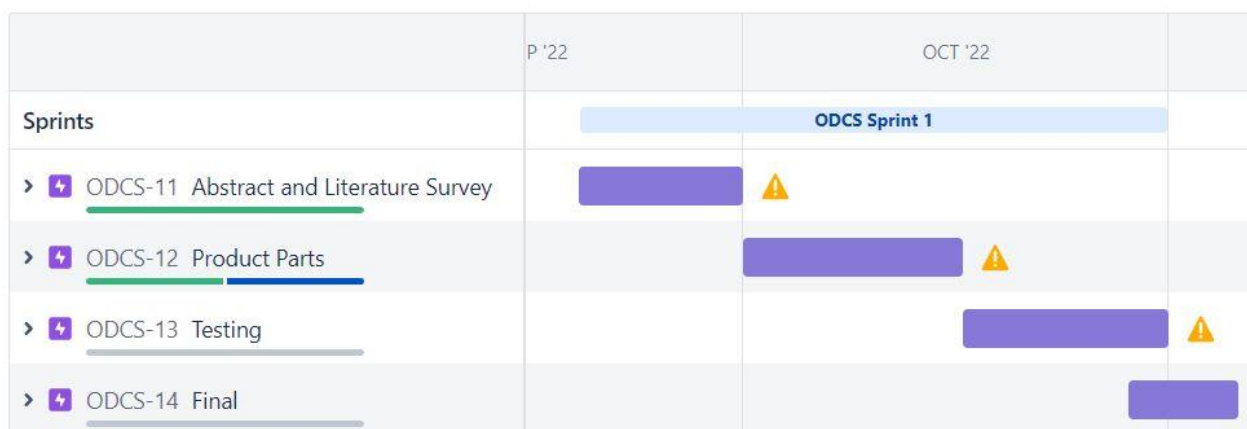
**Name:** Lloyd Louis, **Roll No.** 9522, **TE COMPS B, Batch-A**

## EXPERIMENT NO. 2

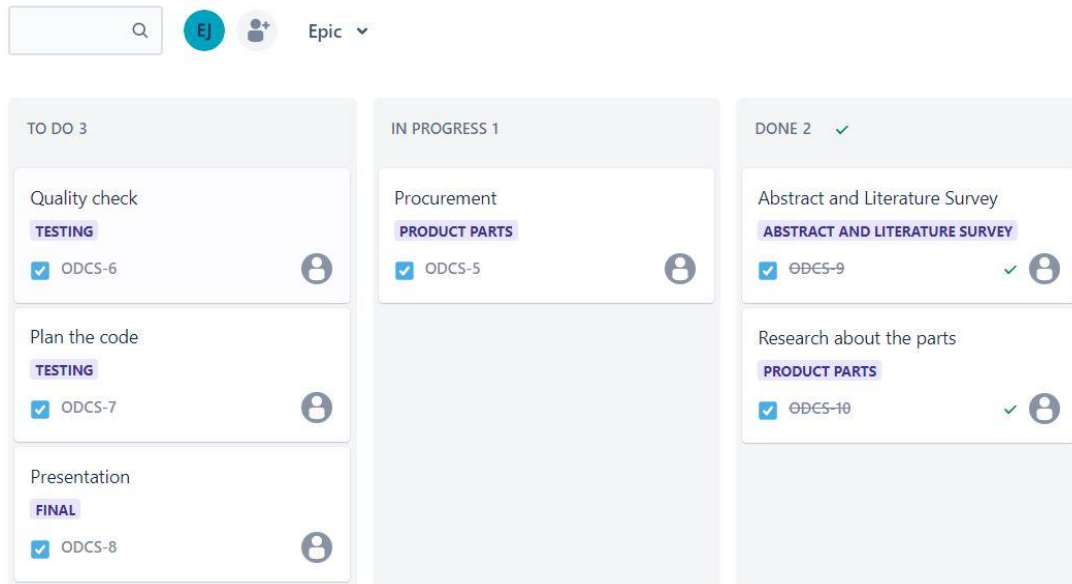
**Objective:** The objective of this lab experiment is to introduce students to the Scrum framework and its implementation using the JIRA tool. Students will gain practical experience in managing a software project using Scrum principles and learn how to utilize JIRA as a project management tool to track and organize tasks, sprints, and team collaboration.



## Timeline



## ODCS Sprint 1



## Backlog



An SRS is basically an organization's understanding of a customer or potential client's system requirements and dependencies.

In SRS following aspects are addressed:

- Functionality.** What is the software supposed to do?
- External interfaces.** How does the software interact with people, the system's hardware, other hardware, and other software?
- Performance.** What is the speed, availability, response time, recovery time of various software functions, etc.?
- Attributes.** What are the portability, correctness, maintainability, security, etc. considerations?

e) **Design constraints** imposed on an implementation. Are there any required standards in effect, implementation language, policies for database integrity, resource limits, operating environment(s) etc.

## **Conclusion:**

- The lab experiment on implementing a project using Scrum on the JIRA tool offers
- students a hands-on experience in agile project management. By utilizing Scrum principles and
- JIRA's capabilities, students learn to collaborate effectively, manage tasks efficiently, and adapt to
- changing requirements. The practical exposure to Scrum and JIRA enhances their understanding of
- agile methodologies, equipping them with valuable skills for real-world software development
- projects. The lab experiment encourages students to embrace the agile mindset, promoting continuous
- improvement and customer-centric software development practices.

## **POSTLAB:**

**Q1. Assess the effectiveness of the Scrum framework for managing software development projects compared to traditional project management methodologies.**

**Ans:** In comparison to traditional project management approaches, the Scrum framework has shown to be particularly successful for managing software development projects. Scrum's iterative and incremental strategy enables frequent product releases, ongoing feedback, and adaptation to changing needs, resulting in improved customer satisfaction and team member participation.

Traditional project management approaches, on the other hand, frequently follow a linear, inflexible procedure, which can limit flexibility and response to changing client demands. The emphasis on self-organizing teams in Scrum allows individuals to take responsibility of work and increases productivity. Furthermore, the use of time-boxed iterations (sprints) in Scrum supports realistic planning and aids in the effective management of project risks.

Scrum's emphasis on frequent meetings, such as Daily Stand-ups and Sprint Reviews, promotes openness and aids in the early discovery of issues.

**Q2. Analyse a Sprint Backlog in JIRA and identify any potential bottlenecks or issues that might hinder the team's progress during the sprint.**

**Ans:** To use JIRA to analyse a Sprint Backlog and discover any bottlenecks or issues impeding the team's progress:

**Examine Work Items:** Examine the items allocated to team members in the backlog. To avoid overloading individuals, ensure that the tasks are well-defined, have clear acceptance criteria, and are fairly allocated among team members.

**Track Progress:** Keep track of the status of tasks as they are completed. Determine which items are taking longer than planned or are routinely blocked. Look at the causes of delays or roadblocks, which might include dependencies, a lack of resources, or external circumstances.

**Examine Workflows:** Look at the workflow statuses of things. Determine any bottlenecks or stages where work tends to accumulate. Unnecessary delays may be avoided by streamlining workflow and optimising operations.

**Assess Communication:** Examine team communication and collaboration. A lack of clear communication or a mismatch of priorities can cause confusion and stymie development. Encourage open dialogue during Daily Stand-ups and Sprint Review sessions.

**Identify any hazards:** Search for any hazards that might jeopardise the Sprint's success. Address any outstanding issues from prior Sprints that may have carried over and hampered progress.

**Monitor Velocity:** Throughout the Sprint, keep track of the team's velocity. If the team's speed is much slower than intended, determine the cause and implement corrective measures.

**Utilise Burndown Charts:** In JIRA, use Burndown charts to visualise progress and compare it to the desired trend. Deviations from planned development might identify areas that need to be addressed.

By conducting this analysis, you can proactively address issues and bottlenecks, ensuring a smoother and more productive Sprint, leading to improved project outcomes.

### **Q3. Evaluate the role of the Scrum Master in handling conflicts within the development team and resolving impediments to maintain a smooth project flow.**

**Ans:** The Scrum Master is critical in settling issues among the development team and ensuring a smooth project flow under the Scrum framework. To begin, the Scrum Master promotes open and honest communication among team members, fostering a collaborative and respectful atmosphere. When disagreements emerge, they serve as a mediator, assisting team members in understanding one another's points of view and reaching agreeable solutions.

Second, the Scrum Master identifies and addresses roadblocks to the team's development. They engage proactively with stakeholders to remove bottlenecks, allowing the team to function more effectively. The Scrum Master ensures that the team's focus stays on providing value, whether it's communicating with product owners, resolving resource restrictions, or managing external dependencies.

The Scrum Master also serves as a coach, supporting the team in the adoption of agile practises and continual improvement. They establish a culture of self-organization and responsibility, allowing the team to efficiently tackle obstacles.

Overall, the Scrum Master's facilitation, dispute resolution, and impediment removal abilities are critical in cultivating a productive and pleasant work environment, allowing the team to consistently create high-quality products.