Department of Computer Engineering

Academic Term: First Term 2023-24

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

Practical No:	**3**
Title:	**Kanban Method: Agile Development**
Date of Performance:	10/08/2023
Roll No:	9605
Team Members:	

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:

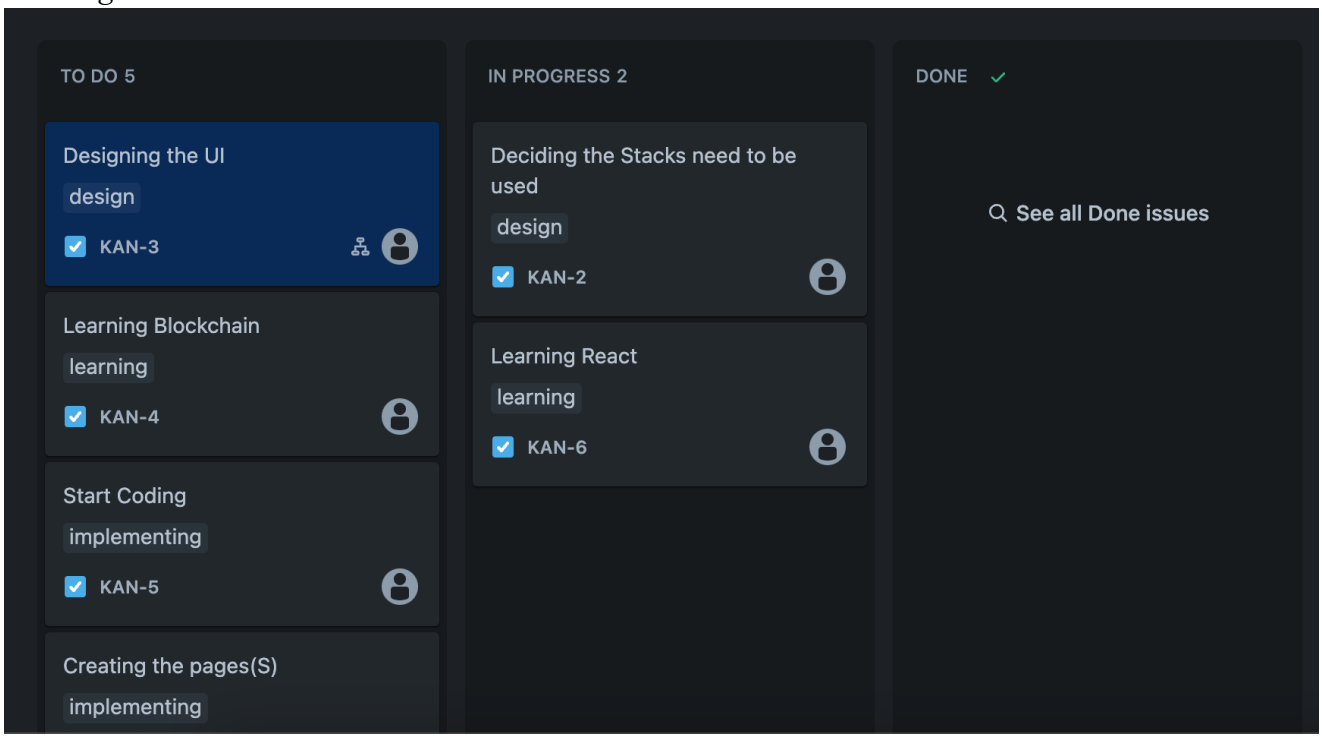
Department of Computer Engineering

Academic Term: First Term 2022-23

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

Working Screenshots:

Creating Issues



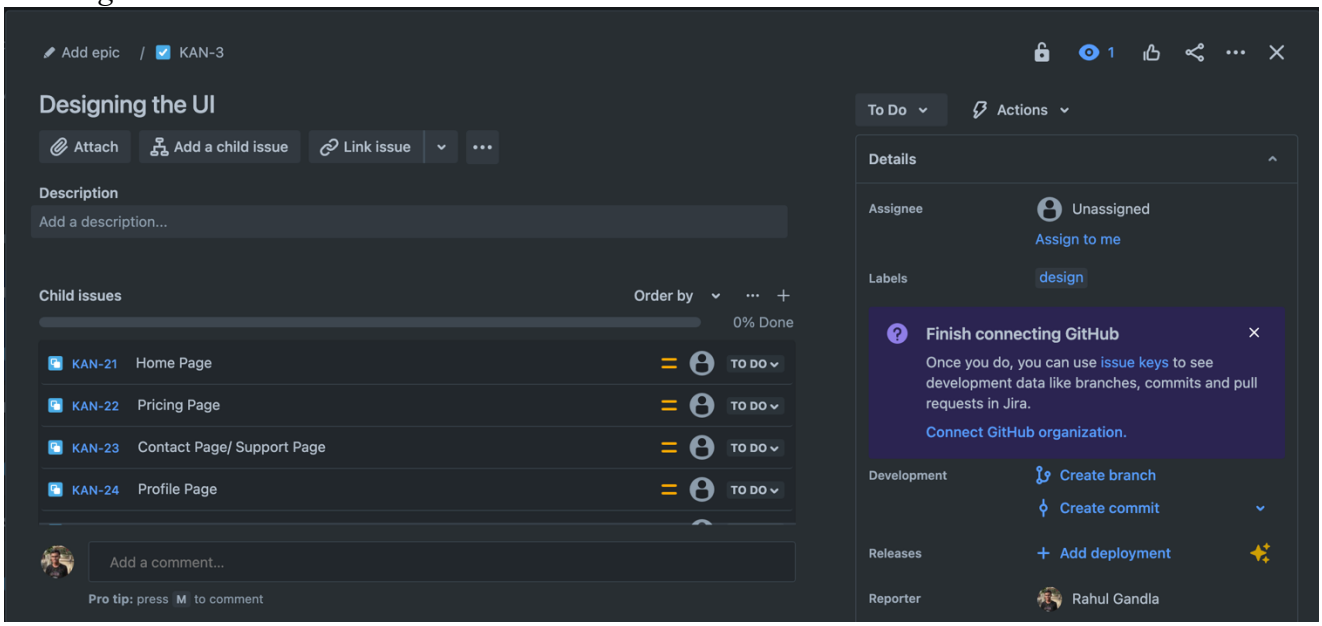
Signature of the Teacher:

Department of Computer Engineering

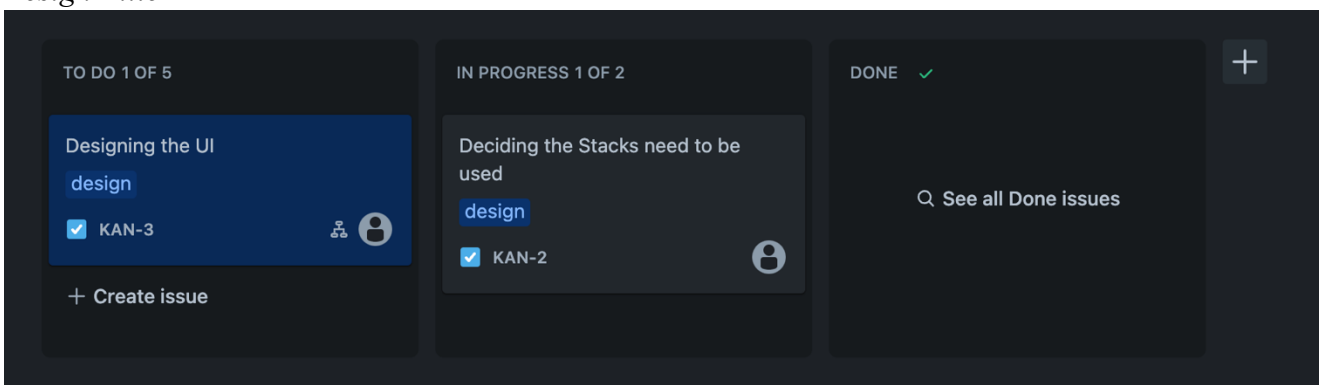
Academic Term: First Term 2022-23

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

Creating Child Issues:



Creating Filters and Filtering:
Design Filter



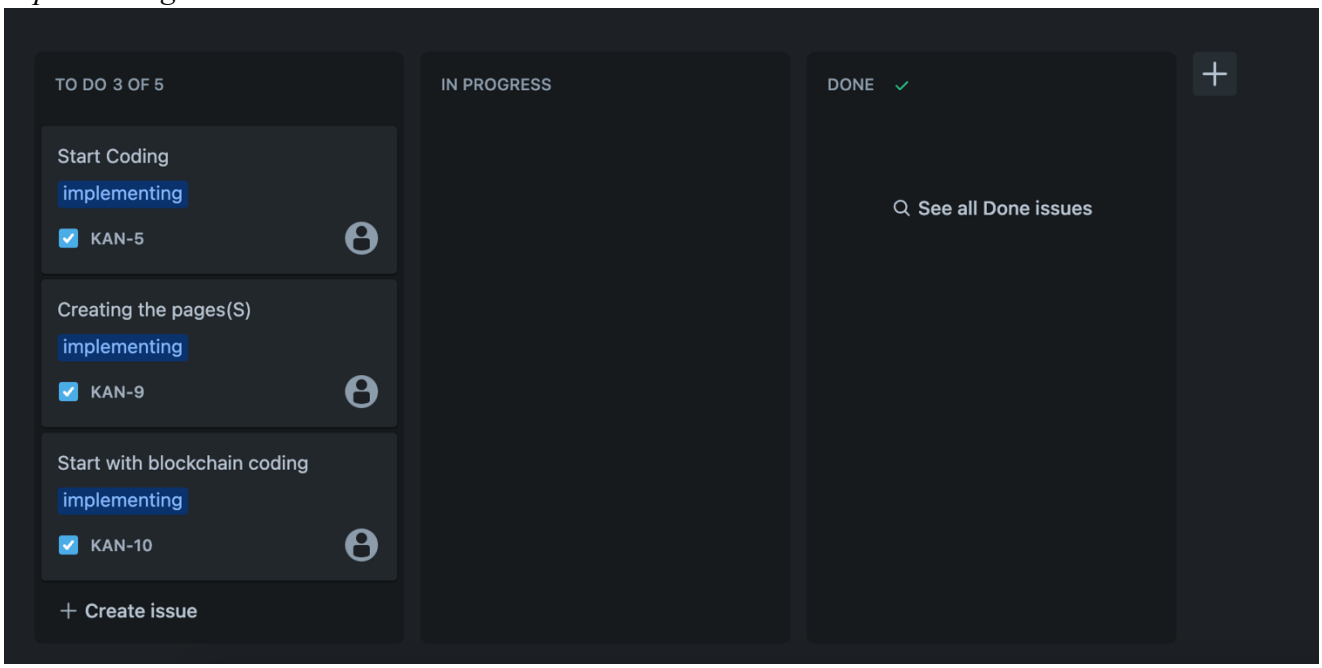
Signature of the Teacher:

Department of Computer Engineering

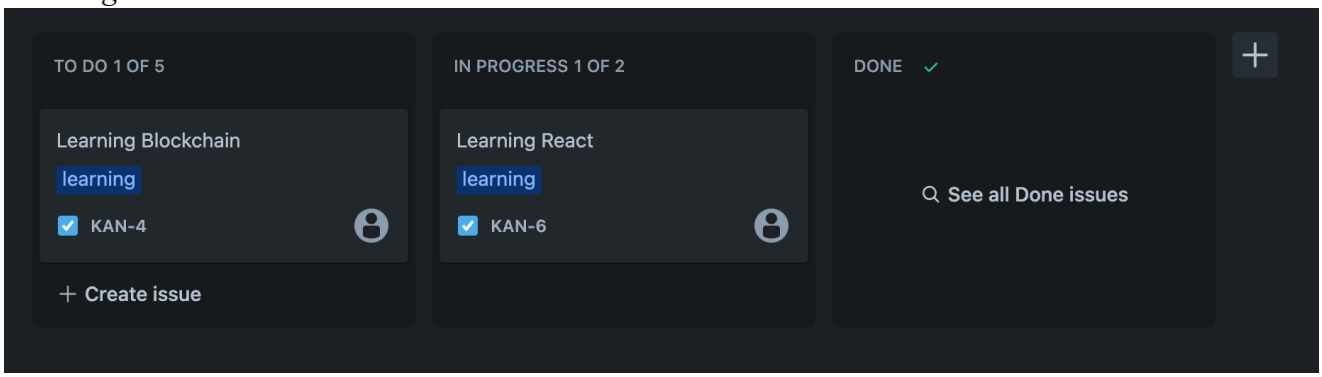
Academic Term: First Term 2022-23

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

Implementing Filter:



Learning Filter:



Signature of the Teacher:

Department of Computer Engineering

Academic Term: First Term 2022-23

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

1. **Compare and contrast the Kanban and Scrum methodologies in terms of flexibility, adaptability, and workflow management in different project scenarios.**

Kanban and Scrum are both popular Agile methodologies used in project management, particularly in software development. They have some similarities but also significant differences in terms of flexibility, adaptability, and workflow management. Let's compare and contrast them in various project scenarios:

1. Flexibility and Adaptability:

- Kanban:
  - Flexibility: Kanban is highly flexible and adaptive. It doesn't prescribe fixed timeframes or iterations, allowing teams to work on tasks as they come in and finish them at their own pace.
  - Adaptability: Kanban can easily accommodate changes in priorities. Teams can adjust their work based on shifting customer needs or emerging requirements, making it suitable for projects with evolving demands.

- Scrum:
  - Flexibility: Scrum provides flexibility within its fixed timeframes (sprints). While changes are possible within a sprint, they are discouraged to maintain sprint stability.
  - Adaptability: Scrum is more adaptive to changes during sprint planning or when a new sprint begins. It encourages change after each sprint but discourages changes during a sprint to preserve focus.

Signature of the Teacher:

Department of Computer Engineering

Academic Term: First Term 2022-23

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

2. Workflow Management

-Kanban
- Workflow Visualization: Kanban relies on a visual board that represents the entire workflow, making it easy to see the status of tasks and identify bottlenecks.
- Work in Progress (WIP) Limits: Kanban uses WIP limits to control the number of tasks in progress, promoting a steady and efficient workflow. Teams pull in new work only when there is capacity.

- Scrum:
- Fixed Timeframes (Sprints): Scrum divides work into fixed-length timeboxes (sprints), typically 2-4 weeks. Teams plan and commit to a set of user stories to complete within each sprint.
- Timeboxing: Scrum uses timeboxing to enforce discipline and create a sense of urgency. All work is expected to be completed within the sprint's duration.

3. Project Scenarios:

- Kanban:
- Maintenance Projects: Kanban is well-suited for ongoing maintenance and support tasks where work arrives unpredictably and must be addressed as it arises.
- Continuous Improvement: Kanban is effective for teams focused on incremental improvements and efficiency gains.

- Scrum:
- New Product Development: Scrum is often preferred for new product development where there's a need for fixed-length development cycles to deliver increments of the product.
- Highly Complex Projects: Scrum may be better for complex projects with well-defined goals, as it provides a structured approach to achieving those goals.

4. Team Structure:

- Kanban:
- Kanban doesn't prescribe specific roles. It can work with various team structures and roles, making it adaptable to different organizational setups.

Signature of the Teacher:

- Scrum:
  - Scrum defines specific roles, including Scrum Master, Product Owner, and Development Team. These roles have defined responsibilities within the framework.

In summary, the choice between Kanban and Scrum depends on the project's nature, the level of flexibility required, and the team's preferences. Kanban is more flexible and adaptive, making it suitable for projects with changing priorities and maintenance work. Scrum, on the other hand, provides a structured approach with fixed timeframes, making it better for new product development or projects with well-defined goals. Many teams also choose to combine elements of both methodologies to create a hybrid approach that suits their specific needs.

2. **Analyze a Kanban board in JIRA and propose improvements to optimize the team's efficiency and productivity.**

1. Review the Current Workflow:
   - Understand the current workflow represented on the Kanban board. Identify columns that represent different stages of work, such as "To Do," "In Progress," "Review," and "Done."

2. Identify Bottlenecks:
   - Look for columns with a high number of cards or tasks that seem to get stuck. These could be bottlenecks in your workflow.

3. Implement WIP Limits:
   - If not already in place, introduce Work in Progress (WIP) limits for each column. This helps prevent overloading team members and promotes a smoother flow of work. Start by setting conservative limits and adjust them as needed.

4. Prioritize Backlog Items:
   - Ensure that items in the "To Do" column are well-prioritized based on their value and dependencies. High-priority items should be at the top.

5. Define and Communicate Policies:
   - Clearly define the policies for moving cards from one column to another. For instance, what

Signature of the Teacher:

criteria must be met for a task to move from "In Progress" to "Review"?

6. Monitor Lead Time and Cycle Time:
   - Use JIRA's reporting features to track and analyze lead time (the time it takes for an item to go from "To Do" to "Done") and cycle time (the time spent actively working on an item). Analyzing these metrics can help you identify areas for improvement.

7. Automate Repetitive Tasks:
   - Consider automating repetitive and manual tasks using JIRA's automation rules. This can save time and reduce the risk of errors.

8. Visualize Blocked Items:
   - Create a column or flag for blocked items. When a task is blocked, make it highly visible on the Kanban board. This will help the team address blockers promptly.

9. Promote Daily Standup Meetings:
   - Encourage daily standup meetings to discuss progress, blockers, and priorities. This fosters better communication and alignment among team members.

10. Continuously Review and Improve:
    - Regularly review the Kanban board and the team's performance metrics. Use retrospective meetings to identify areas for improvement and implement changes accordingly.

11. Integrate with Other Tools:
    - If your team uses other tools or systems for specific tasks (e.g., code repositories, testing tools), consider integrating them with JIRA to streamline workflows.

12. Training and Onboarding:
    - Ensure that all team members are familiar with the Kanban board and JIRA's features. Provide training and onboarding for new team members as needed.

13. Experiment with New Practices:
    - Be open to experimentation. Try out different practices and techniques to see what works best for your team. Continuous improvement is a core principle of Kanban.

Signature of the Teacher:

14. Align with Business Goals:
    - Ensure that the Kanban board and workflow align with the broader business goals and priorities. This ensures that the team is working on the most valuable tasks.

Remember that the effectiveness of these improvements will depend on your team's unique context and challenges. Regularly assess the impact of changes and iterate on your Kanban board and processes to continually optimize efficiency and productivity.

3. Evaluate the impact of Work In Progress (WIP) limits on a Kanban board and how it affects the team's throughput and cycle time.

Certainly, here's an evaluation of the impact of Work In Progress (WIP) limits on a Kanban board and how they affect the team's throughput and cycle time:

Work In Progress (WIP) limits play a crucial role in Kanban, influencing a team's efficiency, throughput, and cycle time. Here's an assessment of their impact:

1. Smoothing Workflow: WIP limits prevent excessive work from piling up in a particular stage of the workflow. By limiting the number of items allowed in each column, WIP limits encourage a more balanced distribution of work across the board. This smoothing effect ensures that the team can maintain a steady flow of work.

2. Focus and Reduced Multitasking: Implementing WIP limits encourages team members to focus on completing their current tasks before starting new ones. This reduction in multitasking improves individual and team productivity. Team members are less likely to be overwhelmed by an excessive number of concurrent tasks.

3. Identifying Bottlenecks: WIP limits make bottlenecks and blockages more visible. When a column reaches its limit, it's a clear indication that the team needs to address the issue causing the bottleneck. This leads to faster problem resolution and smoother workflow.

4. Improved Quality: With fewer items in progress, team members can dedicate more time and attention to each task, leading to higher-quality work. Rushing through tasks to clear the board is

less likely when WIP limits are in place.

5. Reduced Cycle Time: By limiting the number of items in progress, WIP limits help reduce cycle time—the time it takes for an item to move from the "To Do" column to the "Done" column. Smoother workflow, reduced multitasking, and quicker issue resolution all contribute to shorter cycle times.

6. Predictability: WIP limits contribute to predictability in the team's performance. When WIP limits are consistently enforced, the team can better estimate the time it takes to complete tasks, making it easier to manage stakeholder expectations and plan future work.

7. Continuous Improvement: Over time, teams can adjust WIP limits based on their experience and performance data. This allows for a process of continuous improvement, fine-tuning the limits to match the team's capacity and optimize workflow.

8. Team Collaboration: WIP limits encourage team collaboration and communication. When a team member finishes a task and notices that another team member has capacity, they can easily hand off work, promoting a sense of shared responsibility and collaboration.

In summary, Work In Progress (WIP) limits in a Kanban board have a positive impact on a team's efficiency, throughput, and cycle time. They help maintain a steady workflow, reduce multitasking, improve work quality, and make bottlenecks more visible. By implementing and continuously adjusting appropriate WIP limits, teams can optimize their processes and deliver work more predictably and efficiently.

Signature of the Teacher: