

Department of Computer Engineering

Academic Term: First Term 2023-24

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

Practical No:	3
Title:	Implementing Project using KANBAN method on JIRA Tool
Date of Performance:	
Roll No:	9567, 9552
Team Members:	Shruti Patil , Mrunal Kotambkar

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

Signature of the Teacher:

Start Sprint

8 issues will be included in this sprint.

Required fields are marked with an asterisk *

Sprint name *

E-waste Management

Duration *

2 weeks

Start date *

10/20/202312:41 AM

End date *

11/3/202312:41 AM

Sprint goal

StartCancel

Sprints

Jira Software

Your work

Projects

Filters

Dashboards

Teams

Apps

Create

Q Search

Ewaste

Software project

You're on the Free plan

UPGRADE

PLANNING

Timeline

Backlog

Board

DEVELOPMENT

Code

Project pages

Add shortcut

You're in a team-managed project

Learn more

Projects / Ewaste

E-waste Management

9 days remaining

Complete sprint

Q

MK

P

Epic

GROUP BY

Epic

Insights

TO DO 10

IN PROGRESS 1

DONE 1

+

WST-1 Login_Page 4 issues

TO DO

Sell

WST-4

P

Payment

WST-20

P

WST-2 Index_page 2 issues

TO DO

TroubleShootingVideo

WST-6

MK

Donate

WST-3

P

Purchase

WST-5

✓

MK

Board

← → ↻ mrukatllassian.net/jira/software/projects/WST/boards/6/backlog?epics=visible

Jira Software Your work Projects Filters Dashboards Teams Apps Create Search

Projects / Ewaste

Backlog

Q MK P Epic Insights

Epic X

Issues without epic

- > Login_Page
- > Index_page
- > Drives
- > Request_pickup

+ Create epic

E-waste Management 20 Oct – 3 Nov (12 issues) 0 0 0 Complete sprint

- WST-3 Donate IN PROGRESS P
- WST-4 Sell TO DO P
- WST-5 Purchase DONE MK
- WST-6 TroubleShootingVideo TO DO MK
- WST-7 Information_about_site TO DO MK
- WST-9 Online/offline meet TO DO P
- WST-10 Score TO DO P
- WST-11 Upload_image TO DO MK
- WST-17 Upload_image_with_location TO DO P
- WST-18 Request_companies TO DO P

Backlog

Epic X

Issues without epic

- > Login_Page
- > Index_page
- > Drives
- > Request_pickup

+ Create epic

Epic

WST-3 Donate	IN PROGRESS	-	P
WST-4 Sell	TO DO	-	P
WST-5 Purchase	DONE	-	MK
WST-6 TroubleShootingVideo	TO DO	-	MK
WST-7 Information_about_site	TO DO	-	MK
WST-9 Online/offline meet	TO DO	-	P
WST-10 Score	TO DO	-	P
WST-11 Upload_image	TO DO	-	MK
WST-17 Upload_image_with_location	TO DO	-	
WST-18 Request_companies	TO DO	-	

Stories

Child issues
Order by

WST-12
Upload_Detail_i...
=
-

TO DO

Subtask

POSTLAB:

Q1 ANS: Kanban and Scrum are both agile methodologies, but they have distinct characteristics when it comes to flexibility, adaptability, and workflow management in various project scenarios.

Flexibility: Kanban: Kanban is highly flexible and focuses on continuous improvement. It allows teams to adapt to changing priorities and requirements easily. New tasks can be added to the backlog and pulled into the workflow as capacity allows. Scrum: Scrum is more rigid in terms of its time-boxed iterations (sprints). Changes to the sprint backlog are discouraged during a sprint to maintain stability, but they can be considered for the next sprint, making it less flexible within a sprint.

Adaptability: Kanban: Kanban is designed for incremental, evolutionary change. It's suitable for projects with constantly changing requirements or where adaptability is a top priority. Scrum: Scrum encourages teams to inspect and adapt at the end of each sprint, allowing for regular adjustments. However, changes are typically planned for the next sprint, which may not be as responsive as Kanban's continuous adaptation.

Workflow Management: Kanban: Kanban visualizes the entire workflow, making it easier to identify bottlenecks and optimize processes. Work items flow through stages based on their priority, and there are no prescribed ceremonies or roles. Scrum: Scrum prescribes specific roles (Product Owner, Scrum Master, and Development Team) and ceremonies (Sprint Planning, Daily Standup, Sprint Review, Sprint Retrospective) to manage the workflow. It provides a more structured approach to work management.

Project Scenarios: Kanban: Kanban is well-suited for support and maintenance projects, where tasks arrive unpredictably, or for teams that need to balance multiple concurrent projects. Scrum: Scrum works best for projects with well-defined goals and a stable backlog. It's effective for delivering features or products in a series of iterations.

In summary, Kanban offers greater flexibility and adaptability, making it suitable for scenarios with changing priorities and ongoing optimization. Scrum provides a more structured approach with fixed sprint durations, making it better for projects with defined goals and less frequent changes. The choice between them depends on the specific needs and nature of the project. Some teams even combine elements of both methodologies to create a hybrid approach that suits their unique circumstances.

Q2 ANS: Analyzing a Kanban board in JIRA and proposing improvements to optimize efficiency and productivity involves a detailed examination of the current workflow. Here are some steps to help you with this analysis and suggestions for improvements:

Review the Current Workflow: Start by examining the existing Kanban board in JIRA. Identify the columns representing different stages of work (e.g. To Do, In Progress, Testing, Done).

WIP (Work in Progress) Limits: Ensure that Work in Progress limits are defined for each column. If not, establish and enforce these limits. WIP limits prevent overloading team members and maintain a smoother flow of work.

Cycle Time Analysis: Analyze the historical data on cycle times (the time it takes for a task to move from one column to another). Identify any bottlenecks or consistently delayed stages.

Classify and Prioritize Work Items: Categorize work items based on priority, complexity, or type. Use swimlanes or labels to differentiate between different types of tasks. This can help in focusing on high-priority items.

Visualize Dependencies: Use links, sub-tasks, or other visual cues to indicate dependencies between tasks. This helps the team see and manage interrelated work items more effectively.

Regular Standup Meetings: If not already in place, introduce daily standup meetings for the team to discuss progress, blockers, and potential improvements. These meetings promote transparency and collaboration.

Continuous Improvement: Encourage the team to regularly

review and reflect on their process. Hold retrospective meetings after a certain number of completed tasks or at the end of each sprint to identify areas for improvement. Automate Repetitive Tasks: Identify any repetitive or manual tasks that can be automated using IRA plugins or integrations