**JIRA Bug Tracking/Reporting and supports Project management tool----used by testers & developers**

**JIRA also supports agile methodologies**

**JIRA is compltly a REST API based tool,Jira provides URL for each of the functionality**

Jira by testing world

https://www.youtube.com/watch?v=Te8hjicjkcE&list=PLIMhDiITmNrJs\_7MTiJdqYyh8nQ3N3bd0

1. Sprint or iteration 2 weeks to 1 month

Eg: Modules of a project

Work order

work permit

Sja

Risk assisment

**Work permits(requirements of work permit)**

--Create permit

--- Print

---Business RUles

--- Plot location

--- Search

--- Lists tab

--- Attachments

--- Work flow

sprint 1: 2 weeks:

--Create permit –**Tester**: Madhu **Developer:** Praveen

--- Print

---Business RUles

sprint 2: Duration- weeks:

sprint 3: Duration- weeks:

sprint 4: Duration- weeks:

Project-----🡪modules-----------🡪requirements-----🡪sprints

**PROJECTS**

Scrum software development

Use this project to manage your Agile development work. Create a backlog, organise work into sprints, check progress using reports, and more. This project includes a Scrum board, a basic Agile workflow and issue type configuration, which you can change later on.

**ISSUE TYPES**

* Bug
* Task
* Sub-task
* Story
* Epic

WorkFlow To Do**-----In progress-------Done/Completed**

Basic software development

Use this project to work on new features for your product and also track any bugs. This project provides you with a basic workflow and issue type configuration, which you can change later on.

**ISSUE TYPES**

* Bug
* Task
* Sub-task
* Improvement
* New Feature
* Epic

WorkFlow To Do**-----In progress----In Review---Done/Completed**

Kanban software development

Use this project to optimise the flow of work in your development project. Add constraints to work-in-progress, analyze how long it takes to complete issues, find bottlenecks in your process, and more. This project includes a Kanban board, a basic Agile workflow and issue type configuration, which you can change later on.

**ISSUE TYPES**

* Bug
* Task
* Sub-task
* Story
* Epic

**Backlog------Selected for development-----In progress-----Done/Completed**

**Business**

Project Management

Create your tasks, organize and track their progress, and deliver your work on time. Estimations and time tracking allow you to report on where your project is at any stage.

**ISSUE TYPES**

* Task
* Sub-task

WorkFlow To Do**-----In progress-------Done/Completed**

Process Management

Create your tasks and track them at every step, from start to finish. You can use this project to review documentation, approve expenses, or other processes.

**ISSUE TYPES**

* Task
* Sub-task

Task Management

Create simple tasks, organize them and get them done. You can use this project to manage your tasks or assign them to someone else.

**ISSUE TYPES**

* Task
* Sub-task

WorkFlow To Do**--------Done/Completed**

**JIRA Terms:**

EPIC(module)

Features/Stories(Requirements of modules)

Task(different operations of a Epic/module)

Sub Task

EFFORT(to track the take time)--- we can specify the time duration of the task thn it will show us how much time left to complete it

Bug(not a current feature or previous one error is called BUG)

Error COrrection subtask( when vr testing any feature and BUG is occurred over thr we will create Error Correction subtask )-------error under current features of a sprint

It will acqure all the project data/properties bcz it will b assigned under sme feature

for bug v should only sepecfy/enter the details

**WORK FLOW OS PROJECT IN JIRA**

**Backlog(**Features/Stories/Requirements of modules/Epics in a project to be done)

Requirements will be 1st in backlog, tester and developer will analyse thmnd if every thing is ok it will be moved further (**Selected for Development )**

**Selected for Development:** if every thing is clear with the requirement it will be moved to here

**In progress**

**Done**

**Release**

**Backlog------Selected for development-----In progress----Review----In Testing----------In Approval-----Done/Completed**

**sub task----Backlog--------In progress------Review(upto review only thy fix it over thr thy will not procced it forward)**

Project Management using JIRA

**PROJECT CREATION IN JIRA**

Create a new project-----**kanban software development**---------NAME,KEY

**EPIC-----create---issue type** Epic-------summary---reporter-----description

**Stories-----create---issue type** Story--------summary---reporter-----description-----**Epic Link**(u can link dis story to its epic)

**Task-----create---issue type** Task -------summary---reporter-----description-----**Epic Link**(u can link dis story to its epic)

**Sub Task----specific Task----more----create subtask**

**BUG-----create---issue type** Bug -------summary---reporter-----description

**Error Correction subtask-------specific feature/story----more----create task---Error Correction subtask**

**1 Reporting a BUG**

**BUG-----create---issue type** BUG-------summary---reporter-----description

**2 Formating a BUG**

We can format the bug description in JIRA using sme commands

\*Strong\*------ bold

{colo:red}TEXT{color} ----text in b/w will change to the specified color

{quote)

TEXT -------- this text will b given quotes

{quote)

h1.TEXT--------it will change to Heading format

-TEXT - -------------- that text will b deleted

\* TEXT------------- \* with space will get text with bullets/listing format

# TEXT------------- # with space will get text with numbers format

#\*/\*# TEXT ------------ sublist

---- 4 hyphens ul get horizontal line

FOR TABEL

||TEXT || TEXT|| -------Headers of table

|TEXT | TEXT|-------------data of the Headers

Tester will Reporting&formatting a BUG then it should be fixed by a developer soo the DEV lead will search the bug by its BUG ID and he will assign it to a developer.

Then developer can c the issue/task assigned to him and he will fix it(startReview-todo-done) thn developer will assign to the tester.

Then tester will **Retest** that fixed bug and report it’s a Done

**3 Search Issue/tickets in JIRA**

**Issues---search for issues---**it will list all the issues which r assigned to u in all the projects

**Filter** option in the right side---

My open issue

Reported by me

Recently viewed

All issues

**Other Filter**

it will list **al the issues** which r assigned to u in **all the projects**

so, to get only issues from a particular project u have option called **Project-----** here u can select wt project u want

so, to get only a particular issue from the project u have option called **Type-----** here u can select wt type of issue u wantlike(Epic,Story,bug,Task and Sub task issue types)

particular project--- particular issue ---now Filter by **Issue Status**(open,closed,inprogress,backlog,selected for devlepoment, to do, done)

u can even filter by **ASSIGNEE---**issues assigned to sme one

**CONTAINS TEXT-**we search a issue by it entering its text here

**4Defect Life Cycle**

Bug life cycle in case of valid Defect

Developer will not accept dis Defect

**Tester** will Reporting & formatting a BUG then it should be fixed by a developer so the DEV lead will search the bug by its BUG ID and he will assign it to a developer.

Then **developer** can c the issue/task assigned to him and he will Verify the bug is valid or not and fix it(startReview-todo-done) thn developer will assign to the tester. Then tester will **Retest** that fixed bug and report it’s a Done

BUT if the **Develope**r will not accept dis Defect then he will put a comment and reject the Defect/Bug and assign the defect back to the Tester.

Then **tester** will c the comments mentioned by the developer

**YES** If tester satisfied with the developer comments------then **CLOSE** Defect/Bug

**NO** If tester not satisfied with the developer comments------thn **REOPEN** Defect/Bug—provide ur explanation/comments and attach required documents/snapshorts regarding that-----assign the defect back to the developer

Then **Developer**will c the comments and documents/snapshorts mentioned by the tester if he think it as a defect----IN DEVELOPMENT ------FIXED

Then DEV LEAD ------**New Biuld**------Defect for Testing-----READY For The TESTING-----assigned to tester

Then Tester-----Review New Built-----ITS TESTING(Test/Verify the fixed defect**)**

**REOPEN---- if bug is not fixed (comments)**

**CLOSE----if bug is fixed (comments)**

Bug life cycle in case of valid Defect

Developer will accept dis Defect(verify and fix defect)

Tester(report Defect/Bug)------DEV LEAD(assign Developer)—Developer(IN TESTEST-FIXED)----DEV LEAD(New Build—set this Defect/bug ready for testing)---Tester(Verify/retest the fixed Defect/bug--COMMENTS)--CLOSED

5 QA Task Management using JIRA

Task Management project

Test Lead----create---**Task---Project Name---issue type** Task-------summary---

priority----reporter-----description-----

Test Lead----Task----Assign to a Team Member

Tester---Backlog(Review)---In Progress----RoadBlocked(if v r not getting anything nd got blocked over thr)-----Comments(mention urprblmnd report it)-----Assign to concern person(thy will c it and clear our doubts )

Tester---whn v get conformation from thm again---In Progress----Done

Sub task

(module----feature-----task---subtask)

(Epic----Story-----task---subtask)

Task will be divided in to subtasks and will be assigned to different ppl

One task performed by the multiple users/team members

Test Lead----create---**Task ---more----create sub task**

Once all the subtask are done/closed thn only v can close the parent Task