

# Ticketing Systems

# Objectives

---

- Overview of ticketing systems
- Why they matter in tech roles
- Change Requests
- SLAs
- Integration with source code and documentation

# What is a Ticketing System?

---

- A centralized way to track work
- Includes issues, bugs, tasks, change requests



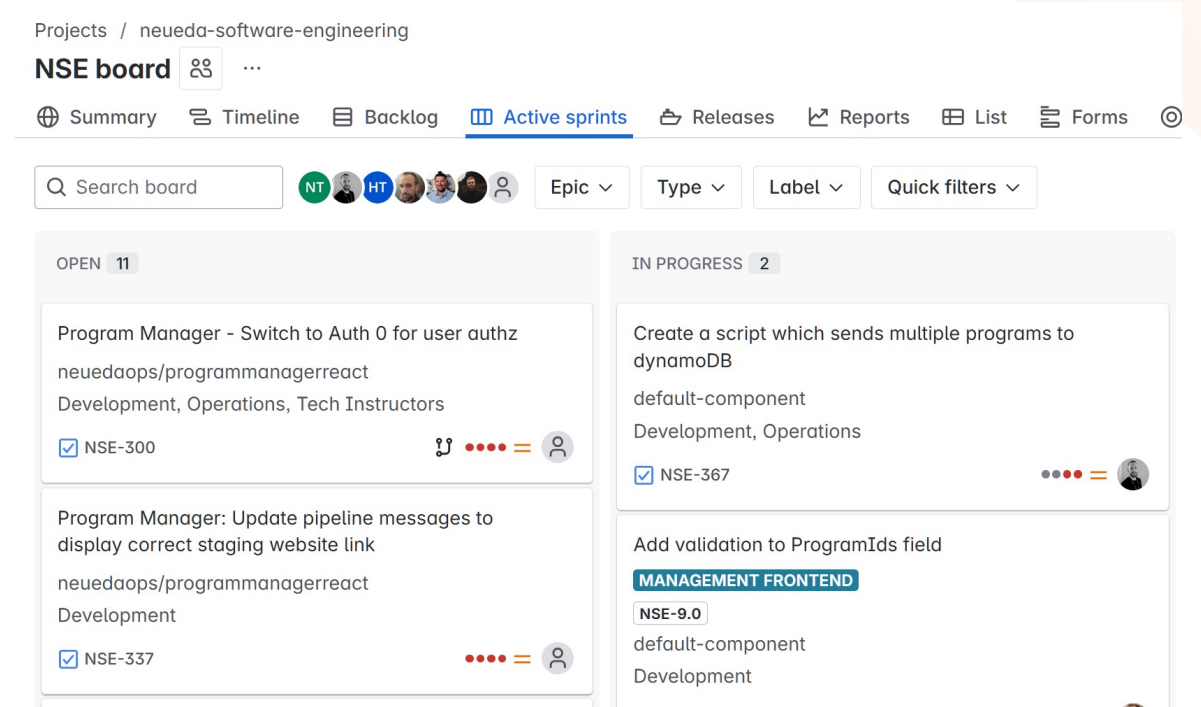
# Why Ticketing Systems Matter in Banking Tech

- Compliance and audit trail
- Structured workflows
- Cross-team collaboration



# Common Systems Used in the Industry

- Jira
- ServiceNow
- Azure DevOps
- Other banking-specific systems



# Core Concepts of Ticketing Systems

---

- Tickets (or issues)
- Projects
- Workflows
- Status transitions

# Understanding Projects

---



- Projects are a collection of tasks or issues that need to be completed to achieve a goal.
- A project can be anything from building a house to creating a software application.
- Projects have a defined start and end date, and are typically designed to achieve a specific outcome or objective.

# Projects in Jira

- Jira allows you to create and manage projects

The screenshot displays the Jira Software interface. At the top, the navigation bar includes the Jira Software logo, a 'Your work' dropdown menu, and links for Projects, Filters, Dashboards, Teams, Plans, and Apps, followed by a 'Create' button. Below this, the 'Your work' section is active, showing a 'Recent projects' list with four project cards: 'training-faculty' (7 open issues, 4 boards), 'neueda-dev' (12 open issues, 1 board), 'IT Support' (0 open issues, 1 board), and 'Digital Delivery' (1 open issue, 1 board). Each card provides quick links to 'My open issues' and 'Done issues'. To the right, a 'Projects' sidebar features a search bar and a list of projects: Digital Delivery, IT Support, Kanban, neueda-dev, and training-faculty, each with a star icon for favoriting. At the bottom of the main content area, filters for 'Worked on', 'Viewed', 'Assigned to me' (20), and 'Starred' are visible.



# Projects and Issues

---

- Issues are individual tasks that need to be completed as part of a project.
- Issues can be anything from fixing a bug to adding a new feature.
- Issues are organized into a project, which allows you to track progress and prioritize work.








# Issues Example

Projects / neueda-dev

## Issues

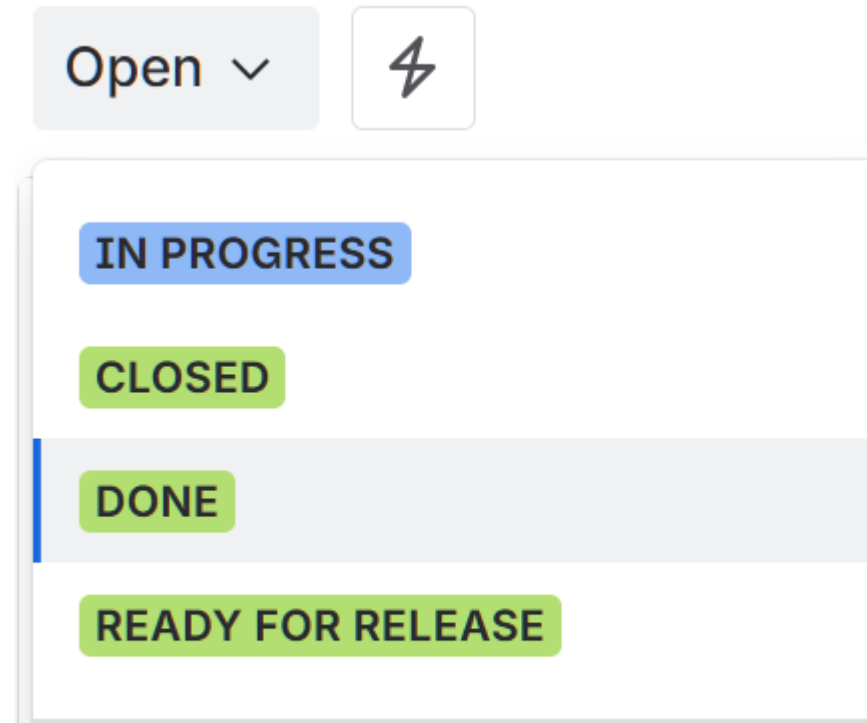
 Search issues  **Project:** neueda-dev  **Type**  **Status**  **Assignee**  **More** 

Type	Key	Summary
	ND-76	As a student I want a better VM experience that minimises latency problems so that I can work more effectively on courses.
	ND-75	As an instructor, I want a consistent location where I can go for documentation about onboarding, VM instructions, and general instructor help so that I can be more effective in the classroom.
	ND-74	As a student I want to be able to use Office 365 in my VM during training
	ND-73	As a student, I can speak to a chatbot within my VM
	ND-72	As a course user, the URL I receive will be short

# The Life of a Ticket

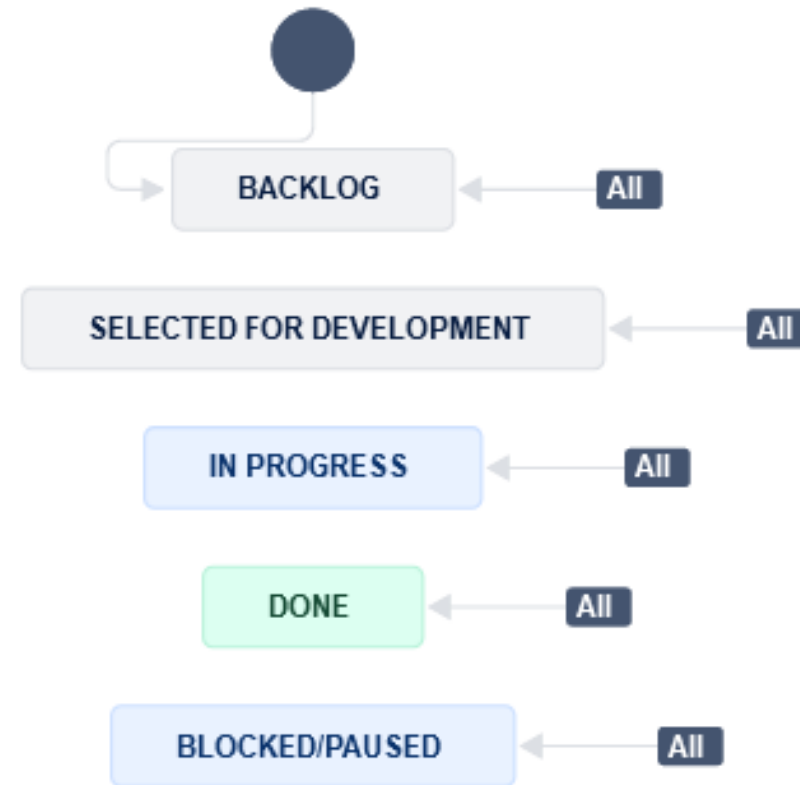
---

- You define your own statuses, they can be things like
  - Created
  - Assigned
  - Worked on
  - Resolved
  - Closed



# Custom Workflows

- Create and customize workflows
- Automate processes
- Streamline team collaboration





# Types of Tickets

- Bugs
- Tasks
- Stories
- Epics
- Change Requests

### Create Bug

Required fields are marked with an asterisk \*

Project \*  
 neueda-software-engineering (NSE) ▼

Work type \*  
 Bug ▼

☒ Task  
☐ Story  
☒ Bug  
☐ Epic

Field Tab HubSpot

# Basic Ticket Anatomy

- Summary
- Description
- Priority
- Assignee
- Reporter
- Labels or Tags
- Attachments

**Create Task**

Work type \*  
☒ Task  
[Learn about work types](#)

Status  
Open  
This is the initial status upon creation  
[Field Tab](#) HubSpot

Summary \*

Parent  
Select parent  
Your work type hierarchy determines the work items you can select here.

Components \*  
Select Component

Description  
Type @ to mention a teammate and notify them about this work item.

**Create Task**

Priority  
Medium  
[Learn about priority levels](#)

Team  
Choose a team  
Associates a team to an issue. You can use this field to search and filter issues by team.

Labels  
Select label

Attachment  
Drop files to attach or [Browse](#)

Linked work items  
blocks  
Type, search or paste URL

Assignee  
Automatic [Assign to me](#)


Sprint  
Select sprint  
Jira Software sprint field

# Roles in a Ticketing System

- Reporter
- Assignee
- Watcher
- Project Manager
- Approver

Details

Assignee

 Unassigned

[Assign to me](#)

Reporter



1 branch

Development

 [Create commit](#)

 [Create pull request](#)

Labels

None

Stakeholder

Development

Operations

Tech Instructors

# Creating a Ticket: Best Practices

---

- Be clear and concise
- Use appropriate fields
- Link to other tickets or documentation



# Examples of Good vs. Bad Tickets

---

- Vague: 'System not working'
- Clear: 'Login page returns 500 error when username includes special characters'

# Searching and Filtering Tickets

- Jira Query Language (JQL) basics
- Using filters, saved searches

My open issues ☆

Basic

JQL

`assignee = currentUser() AND resolution = Unresolved order by updated DESC`

--	--	--	--	--

≡ Filters

🔍 Search work items

▼ Default filters

≡ My open issues

≡ Reported by me

≡ All issues

≡ Open issues

≡ Done issues

≡ Viewed recently

≡ Created recently

≡ Resolved recently

≡ Updated recently

# Agile Boards

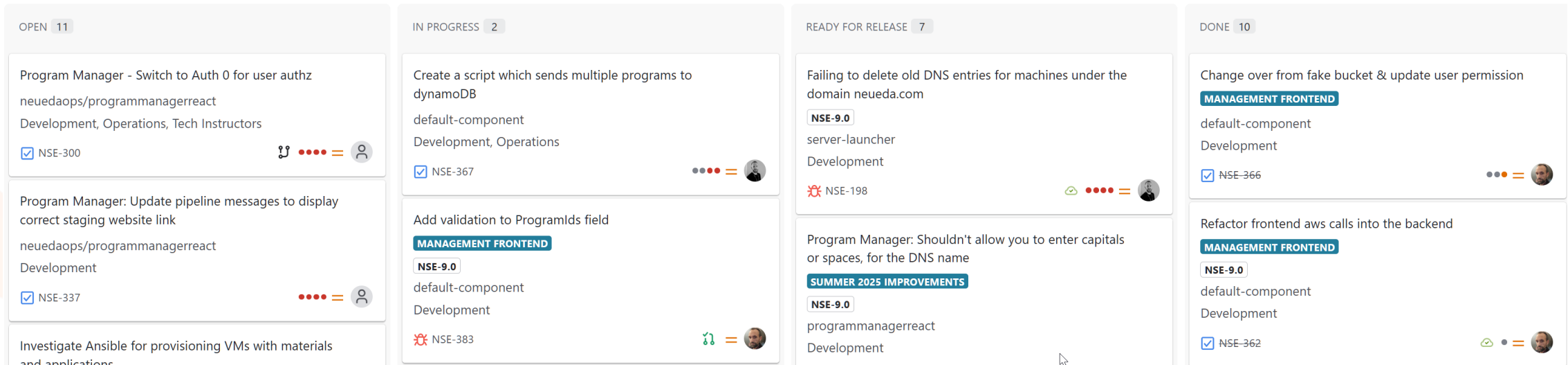
---



- An agile board is a visual representation of a team's work process
- It helps to monitor project progress and work status
- It is a tool that can be customized to fit different team's needs

# Using Agile Boards

- Scrum and Kanban boards
- Viewing progress visually
- Understanding swimlanes and columns



# How Agile Boards Work

---

- Agile boards have columns representing different stages of the project
- Tasks move across the board as they progress
- Team members collaborate by updating tasks on the board

# Purpose of Agile Boards

---



- Agile boards help teams visualize the work process
- Agile boards help teams identify bottlenecks and areas for improvement
- Agile boards promote team collaboration and communication

# Understanding Epics and Stories

---

- Epics = big units of work
- Stories = smaller, actionable pieces
- Linking stories to epics

# Subtasks and Task Breakdown

---

- How complex tasks are broken down
- Why subtasks improve clarity and tracking



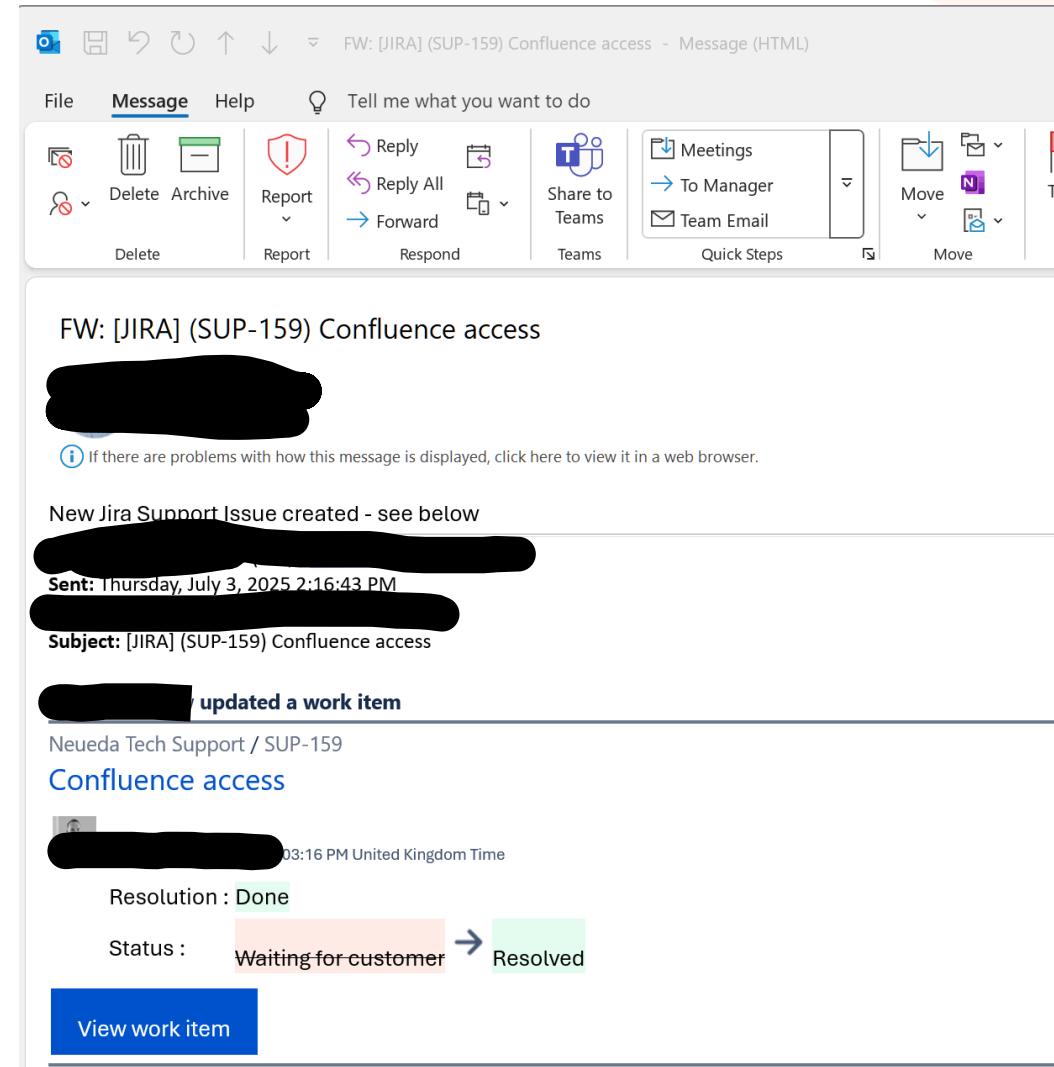
# Assigning and Reassigning Work

---

- How assignment works
- When to reassign
- Etiquette of handing over tickets

# Handling Comments and Communication

- Collaborating in the ticket
- When to update vs. comment
- Email notifications



# SLA and Priority

---

- What is SLA (Service Level Agreement)
  - Gmail 99.9% available
- How priority levels guide response time
- Business vs technical priority

Gmail SLA. Google shall use all reasonable commercial efforts to ensure that the Gmail web interface is operating and available to Customers **99.9%** of the time in any calendar month. In the event Customer experiences any of the service performance issues defined below due to Google's failure to provide Services, Customer will be eligible to receive the Service Credits described below (the "Gmail SLA").

Period	Allowed Downtime
Per Year	8 hours 45 minutes
Per Month	43 minutes 50 seconds
Per Week	10 minutes 5 seconds
Per Day	1 minute 26 seconds

# Introduction to Change Requests

---

- What is a Change Request (CR)
- Why change needs formal management
- Examples in banking tech



# Initiating a Change Request

---

- How CRs are raised
- Typical fields: justification, impact, risk
- Change Advisory Board (CAB) process

# Storing and Tracking Change Requests

---

- CRs stored as tickets
- May be linked to a parent project or epic
- Workflow includes approvals and signoff

# Change Workflow Example

---

- Submitted → Under Review → Approved → Scheduled → Implemented → Closed

# Audit and Compliance Considerations

---

- Why change tracking matters in banking
- Access controls
- Immutable logs and audit trails



# Summary

---

- Overview of ticketing systems
- Why they matter in tech roles
- Change Requests
- SLAs