Project Title:

Optimizing User, Group, and Role Management with Access Control and Workflows

**Team ID : NM2025TMID08613**

**Team Size : 3**

**Team Leader : Veersath M**

**Team member : SIVA Pranesh R**

**Team member : Mahesh B S**

# Problem Statement:

In a project team consisting of a **Project Manager (Alice)** and a **Team Member (Bob)**, the absence of defined roles, access controls, and automated workflows caused confusion in task ownership and limited visibility into project progress.  
This project implements a structured, role-based access and workflow system within the **ServiceNow platform**, enabling clarity, accountability, and automation in task management.

# Objective:

To design and configure a **ServiceNow-based application** that:

* Defines clear roles and permissions for users and groups.
* Implements **Role-Based Access Control (RBAC)**.
* Automates task workflow transitions using **Flow Designer**.
* Provides real-time visibility using **reports and dashboards**.

# Project Overview:

This application demonstrates how **ServiceNow configuration tools**—Users, Roles, ACLs, Flow Designer, and Analytics—can be integrated to create an efficient and secure project task management system.

**Key Components:**

* Role-based access for Managers and Members.
* Task management table for storing project tasks.
* Automated workflow that updates task status.
* Reports and dashboards for progress tracking.

# Detailed Breakdown:

1. **User, Group, and Role Creation:**

**What it does**: Creates distinct identities and assigns them specific access.

**How it works:**

* + Users like Alice (Project Manager) and Bob (Team Member) were created.
  + Groups were formed based on roles: Project Managers and Team Members.
  + Custom roles such as project\_manager, team\_member, and admin were defined.
  + Roles were assigned to users and linked to their respective groups, setting up the foundation for access control rules.

1. **Custom Scoped Application:** Project Task Tracker

**What it does**: Centralizes task management inside a dedicated app.

## How it works:

* + A scoped application called Project Task Tracker was created.
  + A custom table Project Task was built with fields: Task Name, Description, Status, Assigned To, Due Date, Created By.
  + This table serves as the main database to store and manage task-related information.

# Role-Based Access Control (ACLs)

**What it does**: Restricts actions like creating, updating, reading, and deleting tasks based on user roles.

## How it works:

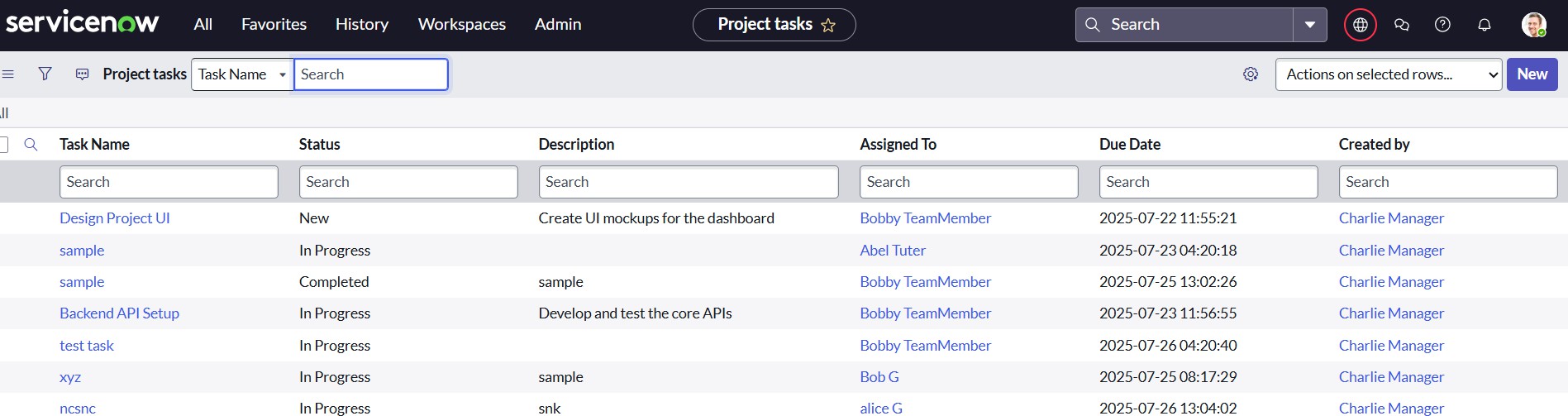
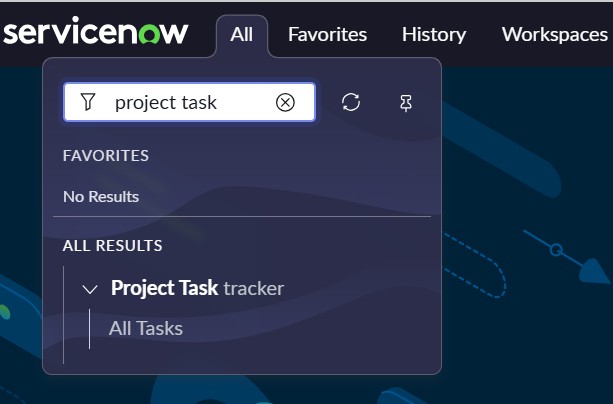
* + ACLs (Access Control Rules) were set up for the Project Task table.
  + Project Managers can create, update, delete any task.
  + Team Members can only read and update tasks that are assigned to them.
  + Roles were applied using “Requires Role” and Condition builder.

# Application Navigation

**What it does:** Helps users quickly access the task list.

## How it works:

* + A custom Application Menu named Project Task Tracker was created.
  + A module All Tasks was added under the menu.
  + This module links to the list view of all project tasks.
  + Role restrictions were used so that only users with correct roles can view this module.

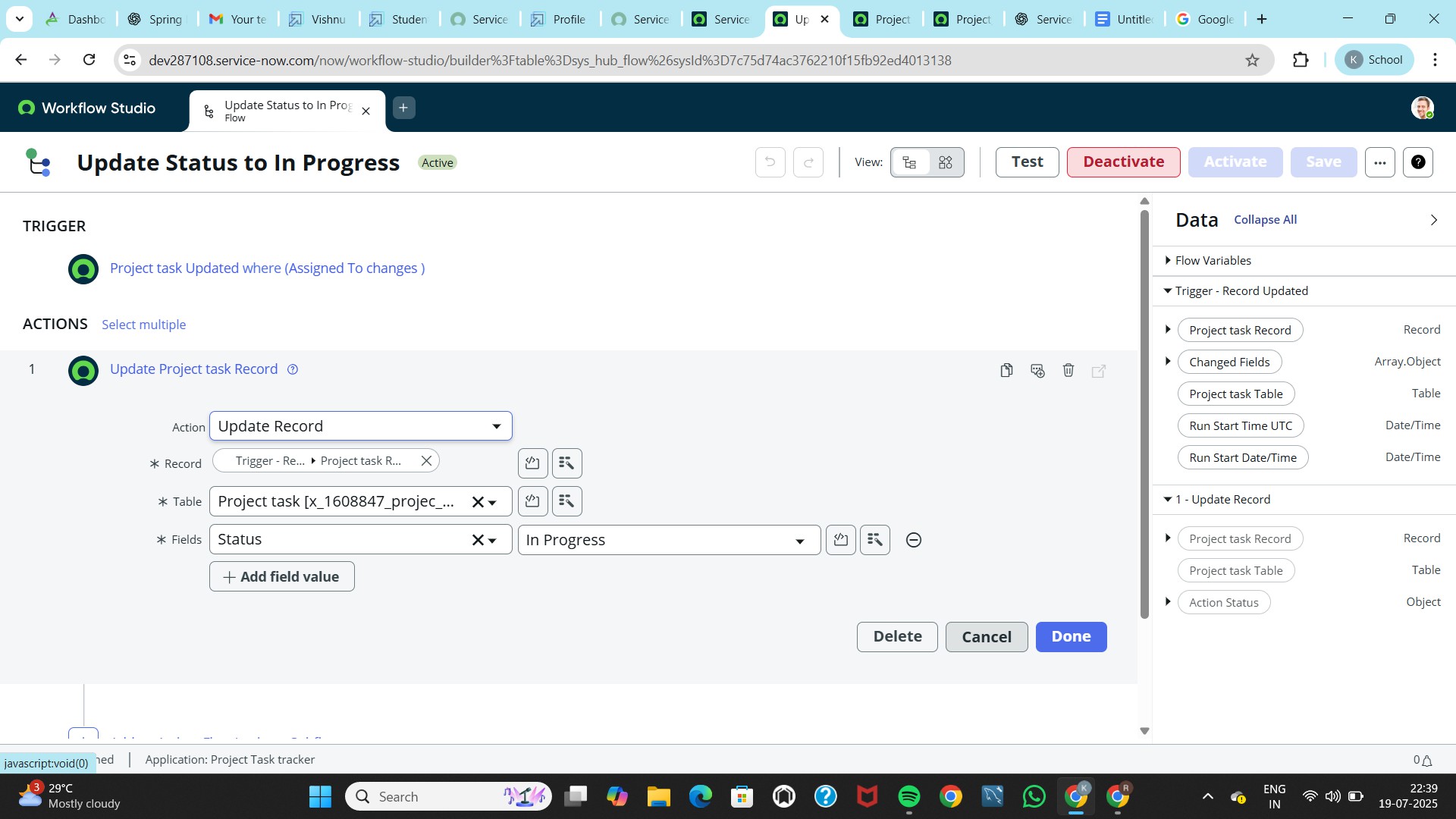
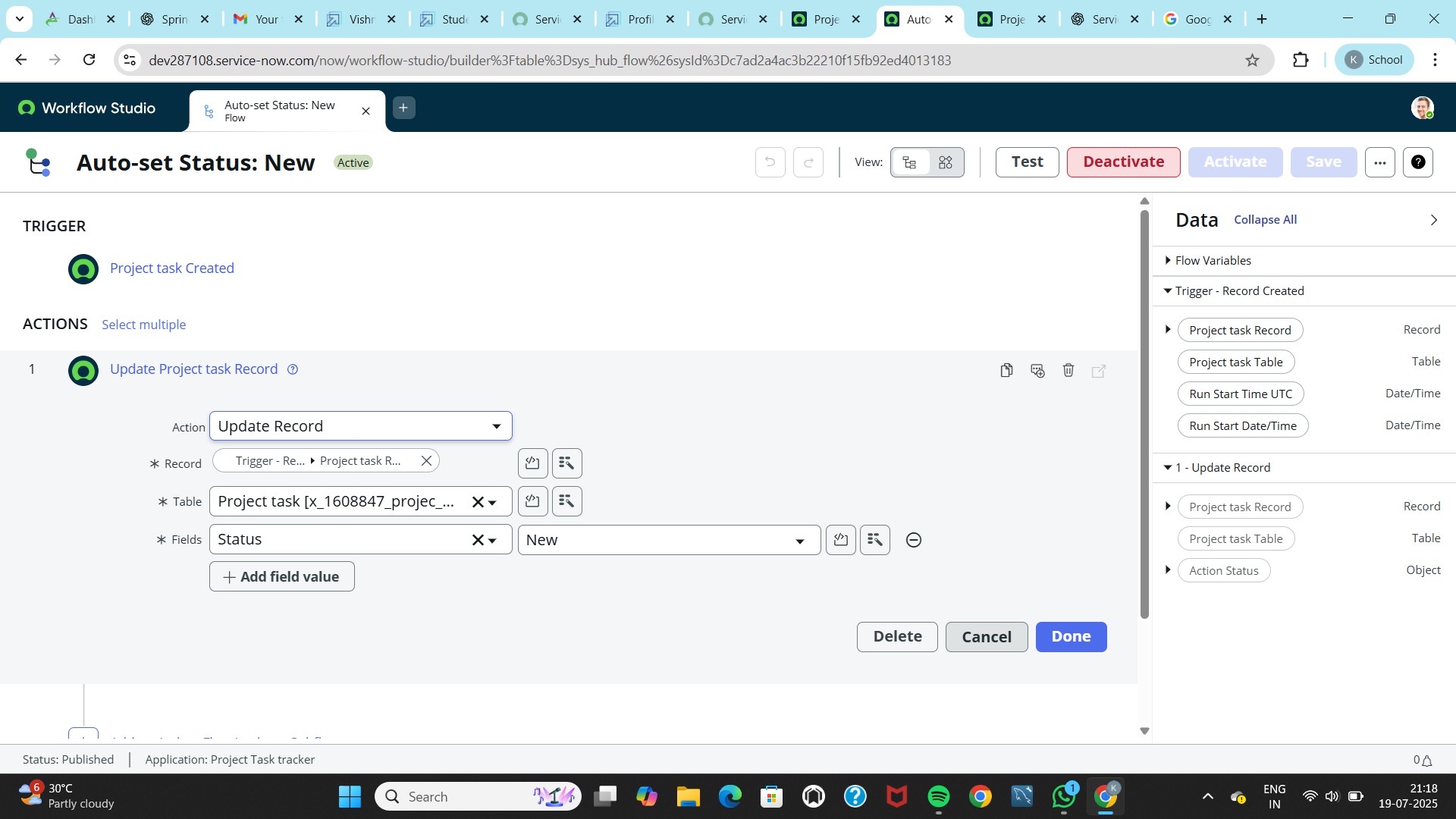


# Workflow Automation Using Flow Designer

**What it does:** Reduces manual updates and keeps task progress accurate.

## How it works:

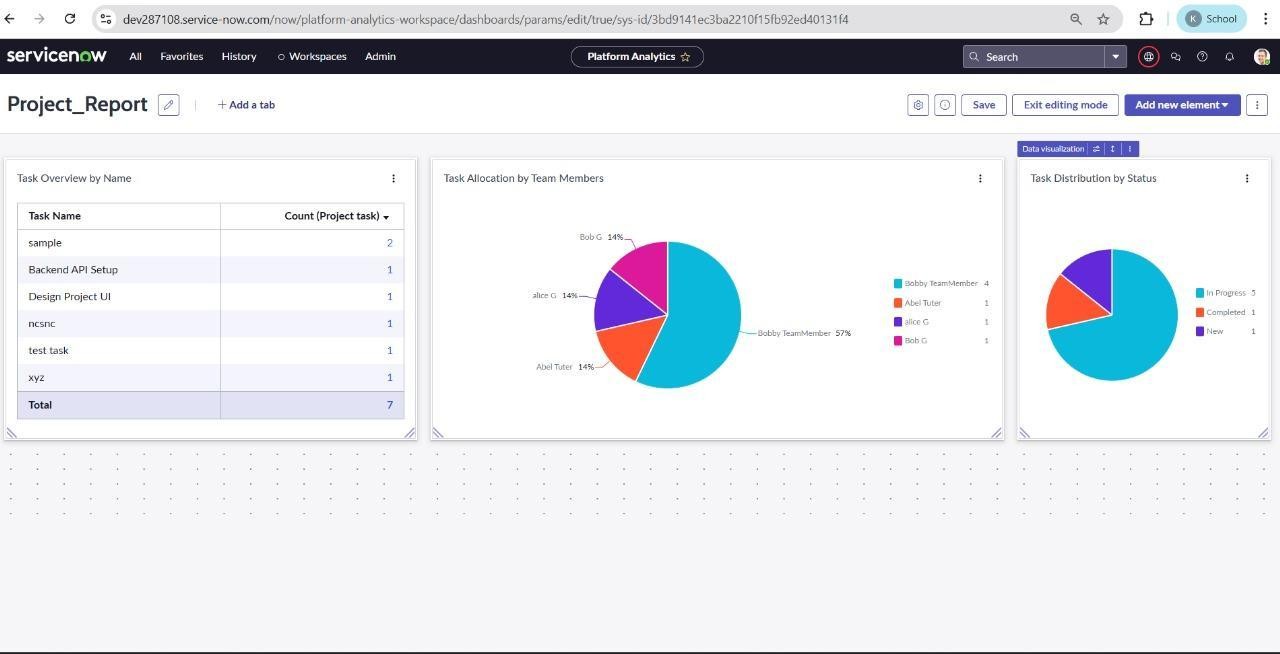
* + A flow was created using ServiceNow Flow Designer.
  + Trigger: When a record in Project Task is updated.
  + Condition: The Assigned To field is not empty
  + Action: Automatically updates the Status field to In Progress.
  + This helps to ensure that any time a task is assigned, its progress is tracked.



# Dashboard & Reporting with Platform Analytics

**What it does:** Visualizes task progress and performance using charts and reports.

## How it works:

* + Multiple reports were created:
  + Pivot Table grouped by Task Name
  + Pie Chart grouped by Status (New, In Progress, Completed)
  + Pie Chart grouped by Assigned Users
  + These reports were added to a dashboard.
  + The dashboard helps managers and team members monitor workload and status easily.

# Outcome:

* + Improved task visibility and accountability
  + Controlled access based on user roles
  + Automated workflow actions to reduce manual efforts
  + Real-time monitoring of task distribution and progress

# Why this is useful:

* + Helps project managers and team members stay organized
  + Prevents unauthorized task modifications
  + Improves collaboration and transparency
  + Supports scalable project tracking with clear metrics

# Conclusion:

This project demonstrates how ServiceNow’s built-in capabilities—**Users, Groups, Roles, ACLs, Flow Designer, and Analytics**—can be configured to create a secure and automated project workflow.  
It serves as a reusable model for teams seeking structured access management and process automations