**STREAMLINING TICKET ASSIGNMENT FOR EFFICIENT SUPPORT OPERATIONS USING SERVICENOW**

**ABSTRACT**

In modern IT service management, manual ticket routing often results in delayed responses, human errors, and inconsistent workload distribution across support teams. This project presents an automated solution developed using *ServiceNow Flow Designer* to streamline ticket assignment based on issue categories. The system dynamically routes tickets to the appropriate support groups—Platform or Certificates—reducing operational latency and improving customer satisfaction. Through structured workflow automation, the project enhances accuracy, accountability, and efficiency in IT service delivery.

Furthermore, this system minimizes the dependency on manual monitoring and ensures every ticket reaches the responsible team instantly. The use of Flow Designer and Access Control Lists (ACLs) ensures high security and operational transparency. The proposed system is easily scalable and adaptable for future enterprise-level needs. This implementation demonstrates how automation in ITSM can significantly transform the overall customer service process.

**TABLE OF CONTENTS**

1. Introduction
2. Objective of the Project
3. Literature Review
4. Methodology
5. System Implementation
6. Technical Details
7. System Analysis
8. System Design
9. Module Description
10. Advantages
11. Disadvantages
12. Future Enhancements
13. Conclusion
14. References

**OBJECTIVE**

The main objective of this project is to design and implement an automated ticket assignment mechanism that ensures every support issue is promptly and correctly routed to the relevant group within the organization.

**Specific goals include:**

* Automating ticket routing using ServiceNow Flow Designer.
* Reducing average resolution time and manual intervention.
* Enhancing visibility, consistency, and accountability in support operations.
* Improving customer satisfaction through faster issue resolution.

This project aims to eliminate repetitive administrative work and reduce dependency on manual human intervention. It also ensures that different teams within the organization receive only the issues relevant to their expertise, improving problem-solving accuracy. The system enforces strict role-based access control, ensuring only authorized users can modify or route tickets. Ultimately, the goal is to create a fully automated, intelligent ticket distribution process that aligns with modern IT service management best practices.

**EXISTING SYSTEM**

In the existing manual system, ticket assignment is performed by support coordinators who evaluate each issue and manually allocate it to a specific department. This method is prone to delays, subjective errors, and uneven workload distribution. The dependency on manual verification increases response time, reducing customer satisfaction and overall operational efficiency.

Moreover, the manual process often leads to inconsistent prioritization and lack of transparency across different support teams. When multiple tickets are received simultaneously, human error becomes more likely, resulting in unresolved or misrouted tickets. The tracking of ticket ownership is also complex, leading to accountability issues. Overall, the manual system lacks the intelligence and automation necessary to handle modern IT support workloads efficiently.

**PROPOSED SYSTEM**

The proposed automated ticket routing system eliminates manual intervention by using **ServiceNow Flow Designer** to assign tickets dynamically based on the issue type. The process integrates ServiceNow modules for **Users, Groups, Roles, Tables, ACLs, and Flows** to create a robust and secure ticket routing workflow. Whenever a new ticket is created in the *Operations Related* table, the system automatically analyzes the issue description and assigns it to the corresponding group (Certificate Group or Platform Group).

The proposed system ensures consistency by applying defined business logic to every new ticket. It enhances real-time collaboration by automatically notifying the concerned team when a new issue is assigned. The automation process also reduces the time wasted in repetitive administrative actions and helps in tracking performance metrics. Thus, the proposed system improves both technical efficiency and organizational communication within the support department.

**SYSTEM ANALYSIS**

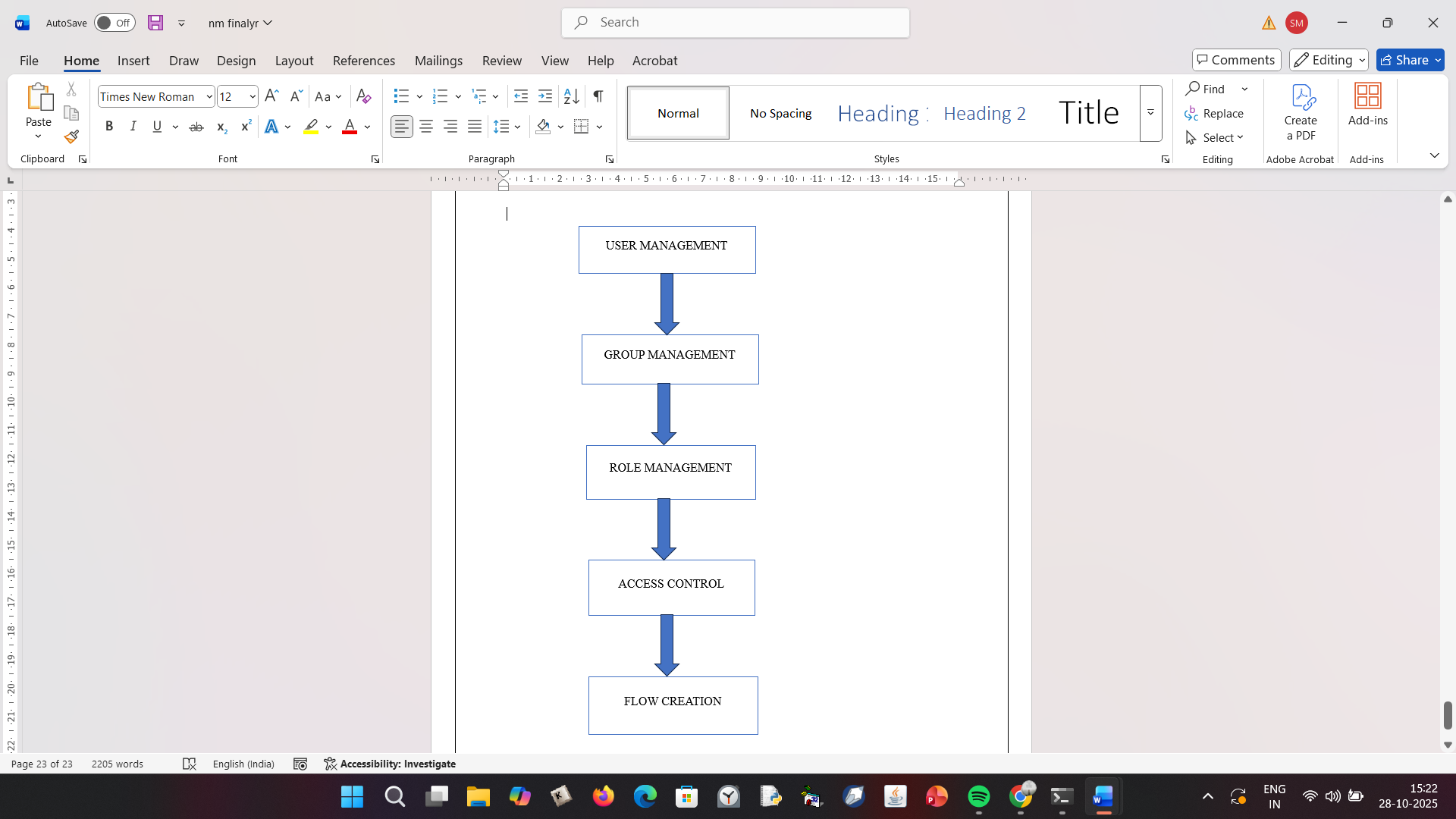
System analysis focuses on understanding the current problem, gathering requirements, and determining the functional and non-functional needs of the new system. The existing manual process of ticket allocation leads to inefficiency, inconsistency, and human error.  
The analysis phase identified the need for:

* A centralized ticket management system.
* Role-based access control for users and groups.
* Automatic routing based on issue categories.
* Real-time visibility into ticket status and progress.  
  The system also needs to ensure scalability for future expansion. By analyzing business workflows and user interactions, the proposed ServiceNow solution aligns automation with organizational goals.



**SYSTEM DESIGN**

System design translates the analysis into an implementable architecture. The design is structured into modules like *User Management, Group Management, Role Management, Access Control,* and *Flow Automation*. Each module interacts within the ServiceNow ecosystem using predefined APIs and tables.  
The system is designed using the **Role-Based Access Control (RBAC)** model, ensuring security and controlled accessibility. The design also defines workflows that trigger automated routing when tickets are created or updated.  
Logical and physical design models define how data is stored and how automation triggers function.  
The database schema includes columns such as *Issue*, *Description*, *Assigned Group*, and *Status*.  
This design ensures modularity, scalability, and seamless integration with existing enterprise IT systems.



**MODULE DESCRIPTION**

**1. User Management Module**

Handles user creation, authentication, and role assignment. It ensures that only authorized users can access the system. It also supports multi-user environments for large organizations.

**2. Group Management Module**

Defines and manages teams responsible for different issue categories. Each group acts as a logical entity for workflow routing and collaboration.

**3. Role Management Module**

Implements access privileges through RBAC. Each role controls what a user or group can read, write, or modify within ServiceNow.

**4. Table Management Module**

Stores all ticket-related data in a structured format. It provides consistency and supports reporting for operational analysis.

**5. Flow Designer Module**

Automates workflows based on issue conditions. This is the heart of the project, ensuring real-time routing and reducing manual errors.

**6. Access Control Module**

Ensures that data is protected through ACLs. It prevents unauthorized access and maintains audit trails for transparency.

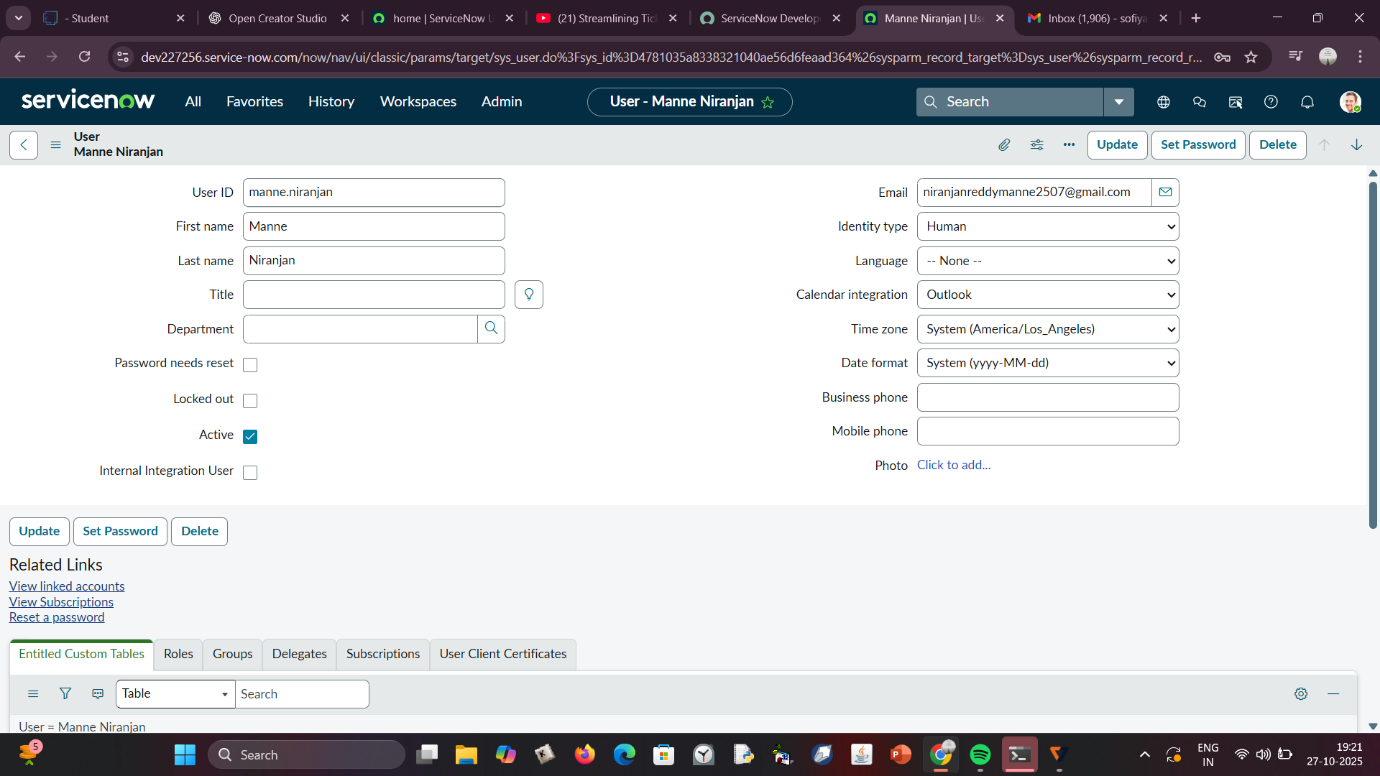
**METHODOLOGY**

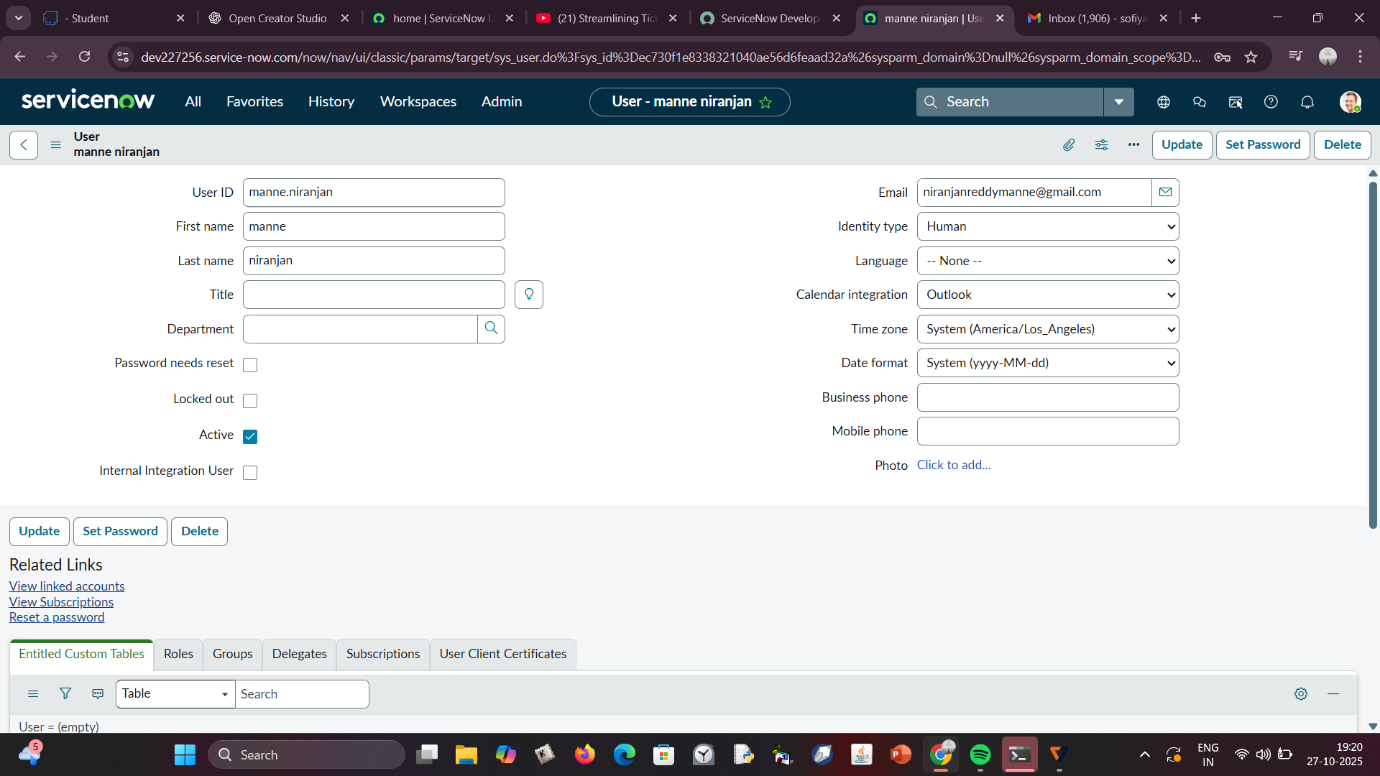
The methodology follows a structured ServiceNow configuration and automation workflow as outlined below. Each step is carefully designed to ensure accuracy, traceability, and secure handling of tickets.

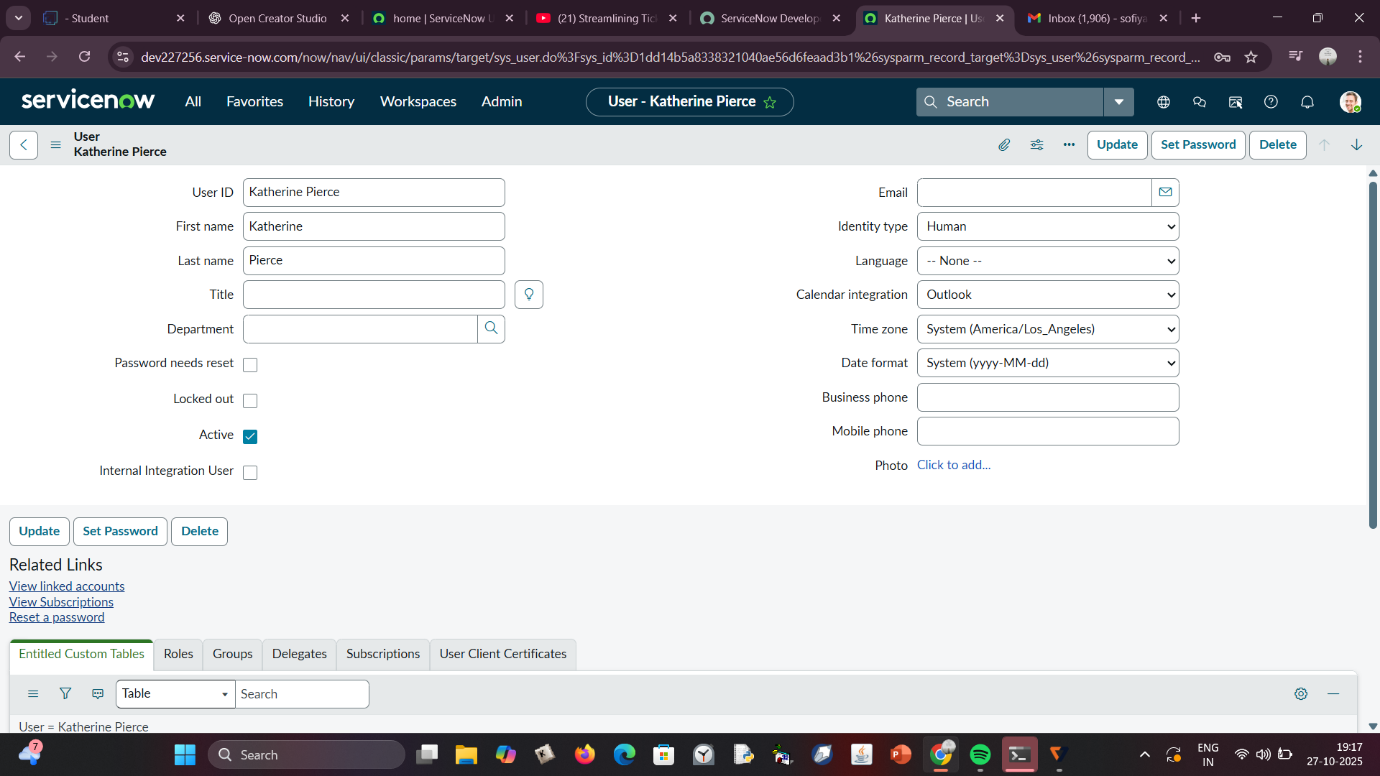
**Step 1: Create Users**

Users represent individual ServiceNow accounts that interact with the system.

1. Navigate to **All → Users** under *System Security*.
2. Click **New** to add a user profile.
3. Enter user details such as name, email, and username.
4. Click **Submit** to save the record.
5. Repeat to create a second user.  
   User creation is essential to define who can access the platform and perform tasks. Each user is later linked to a specific group and assigned roles that determine access permissions. Properly structured user management ensures accountability and tracking in support workflows.



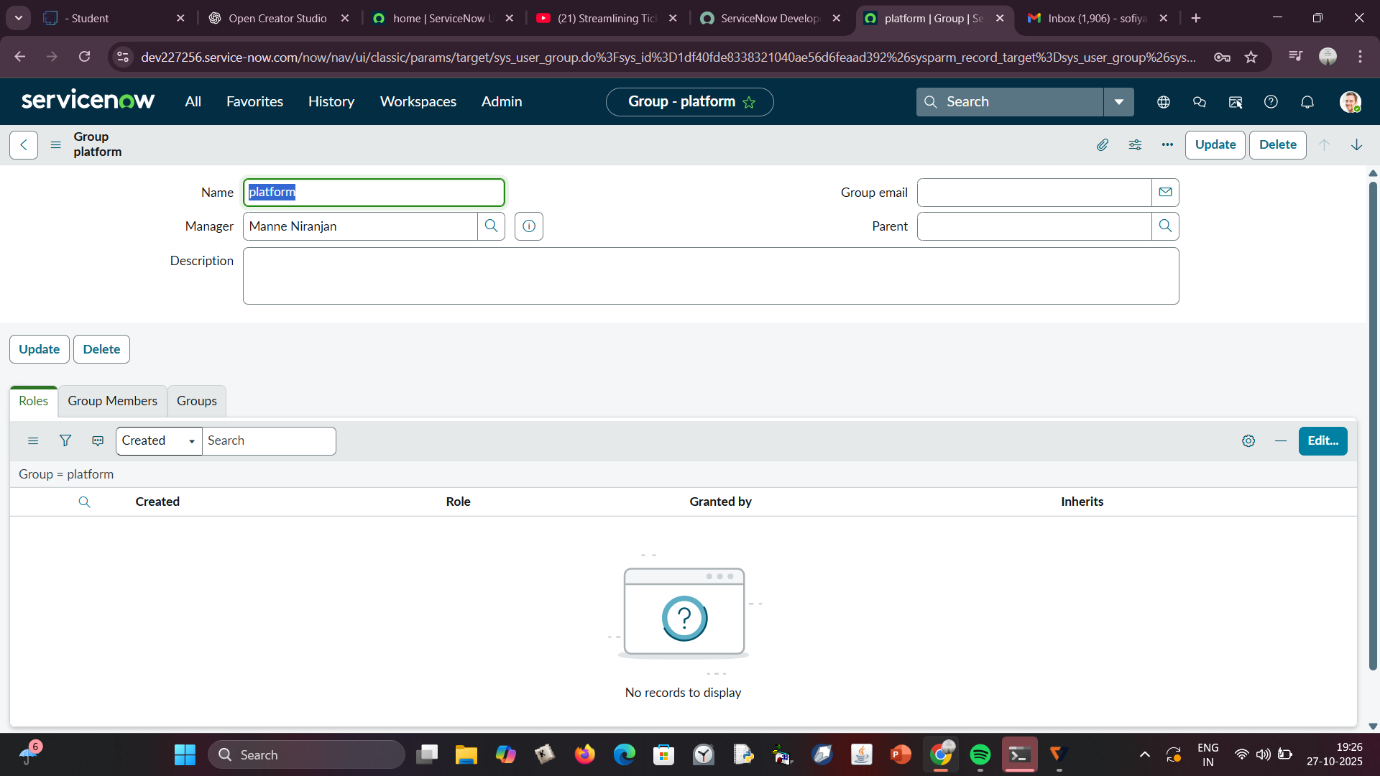


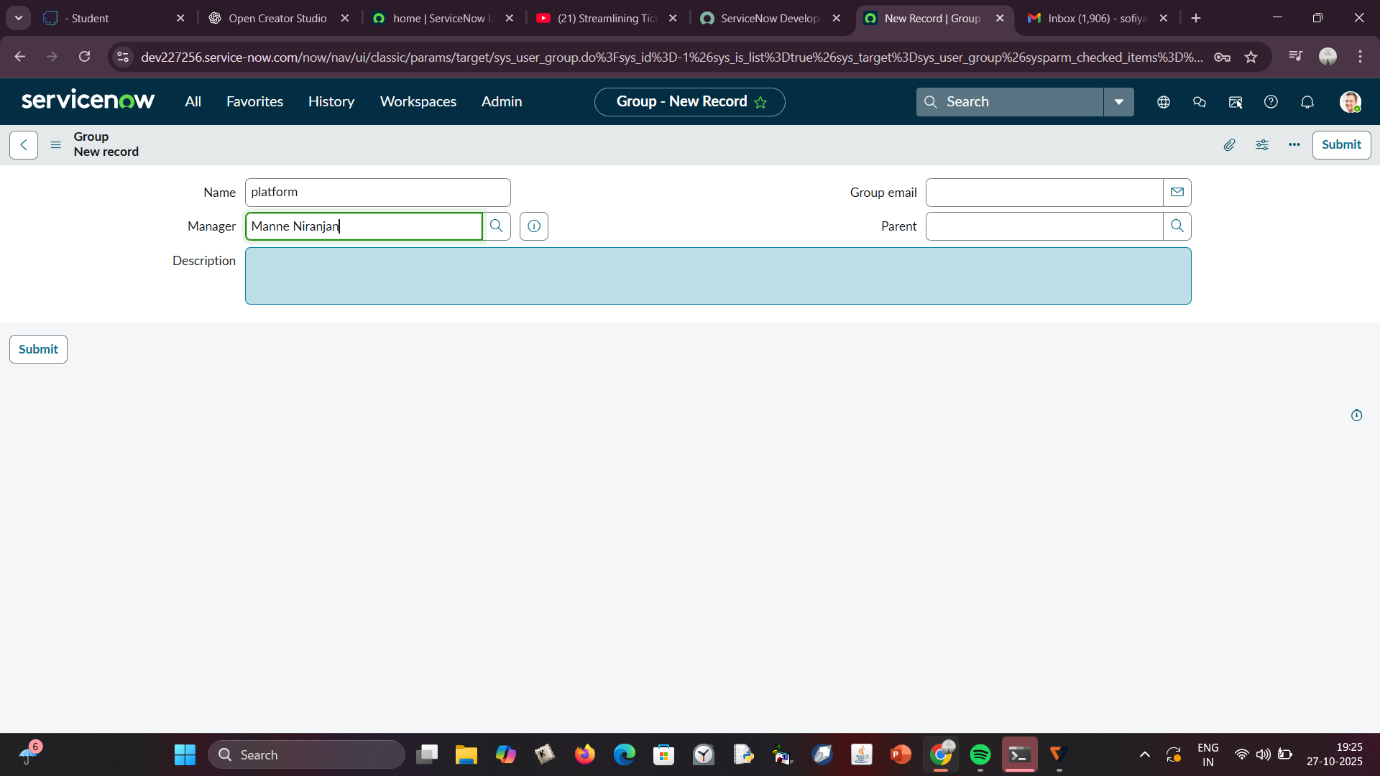


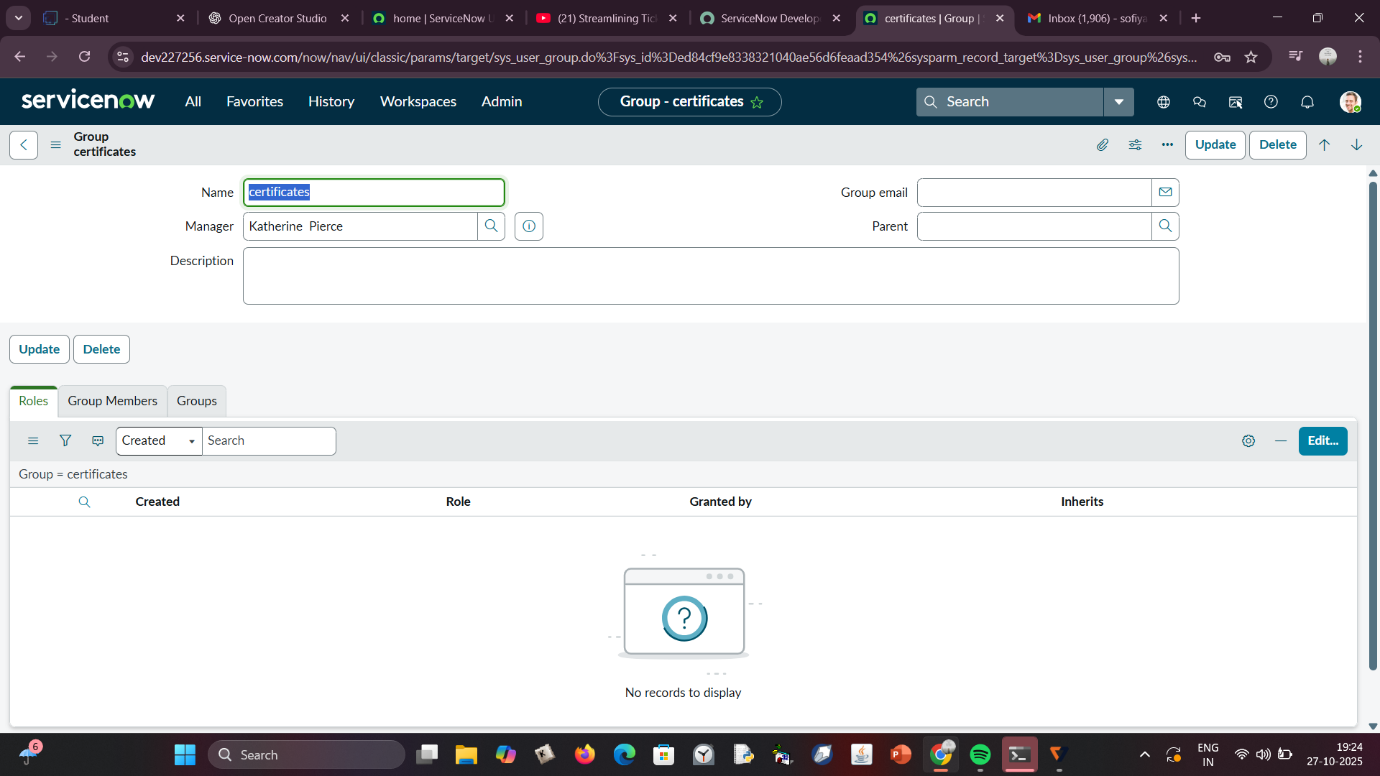
**Step 2: Create Groups**

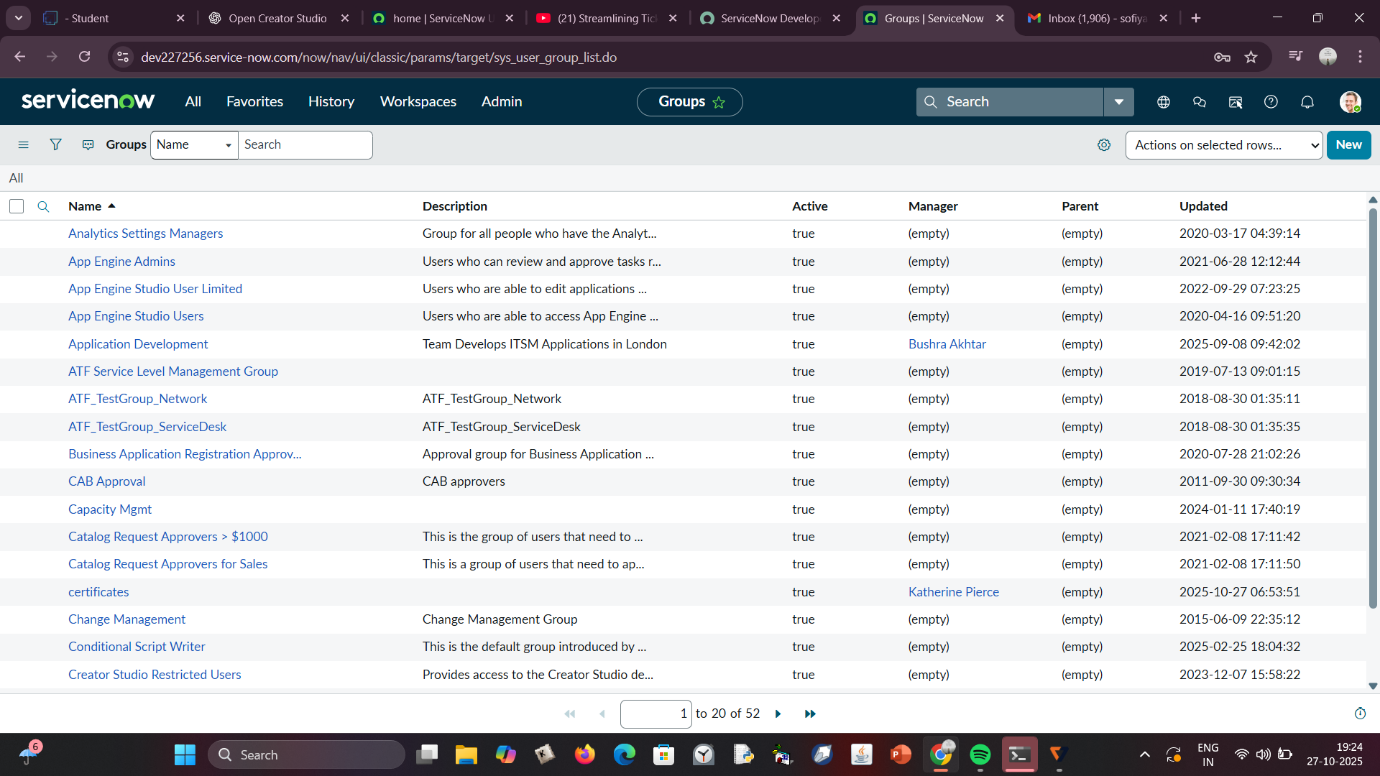
Groups represent teams that handle specific issue types or business domains.

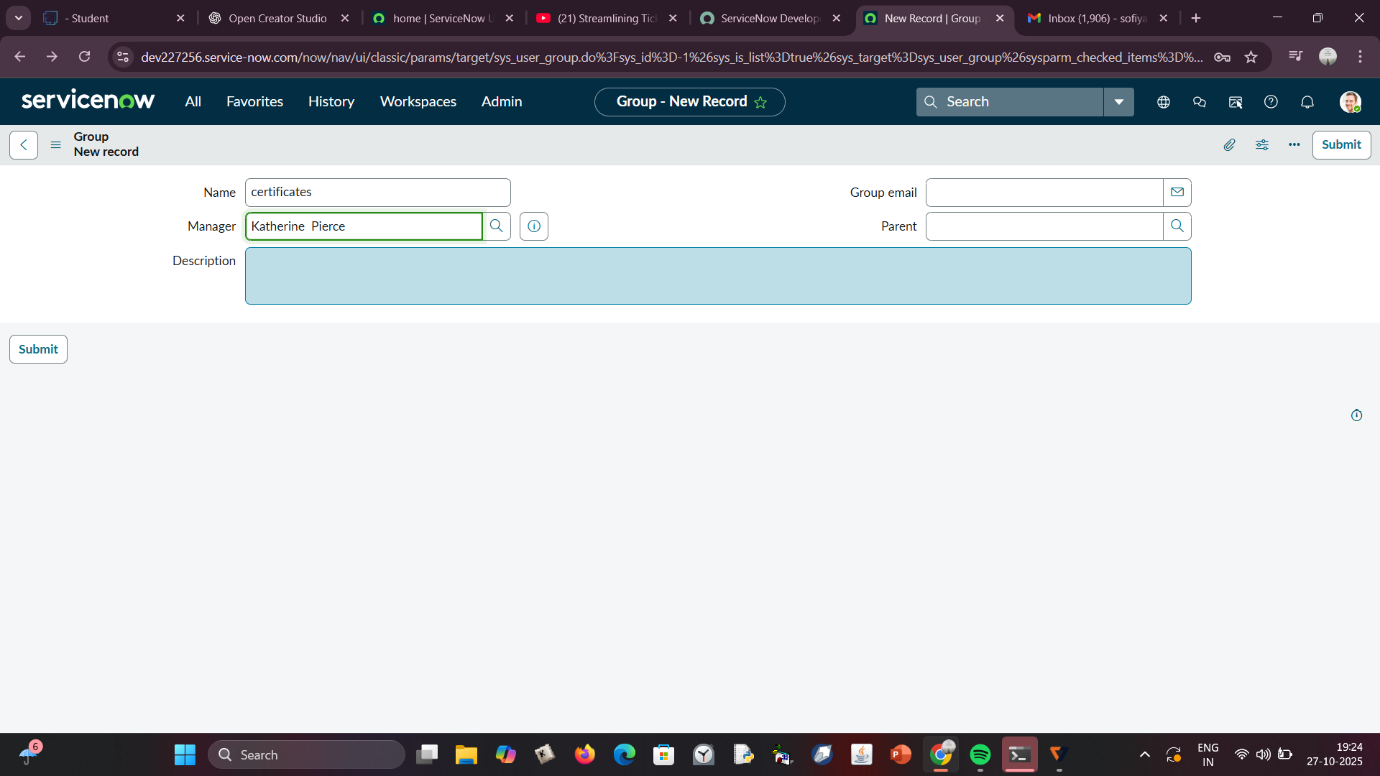
1. Navigate to **All → Groups** under *System Security*.
2. Click **New** and enter group details (e.g., Certificates, Platform).
3. Click **Submit** to create the group.
4. Repeat for the second group.  
   Groups help in efficient collaboration and categorization of tickets. Each group has specific roles, allowing ServiceNow to automate ticket distribution accurately. Clear group creation prevents task duplication and confusion during ticket escalation.







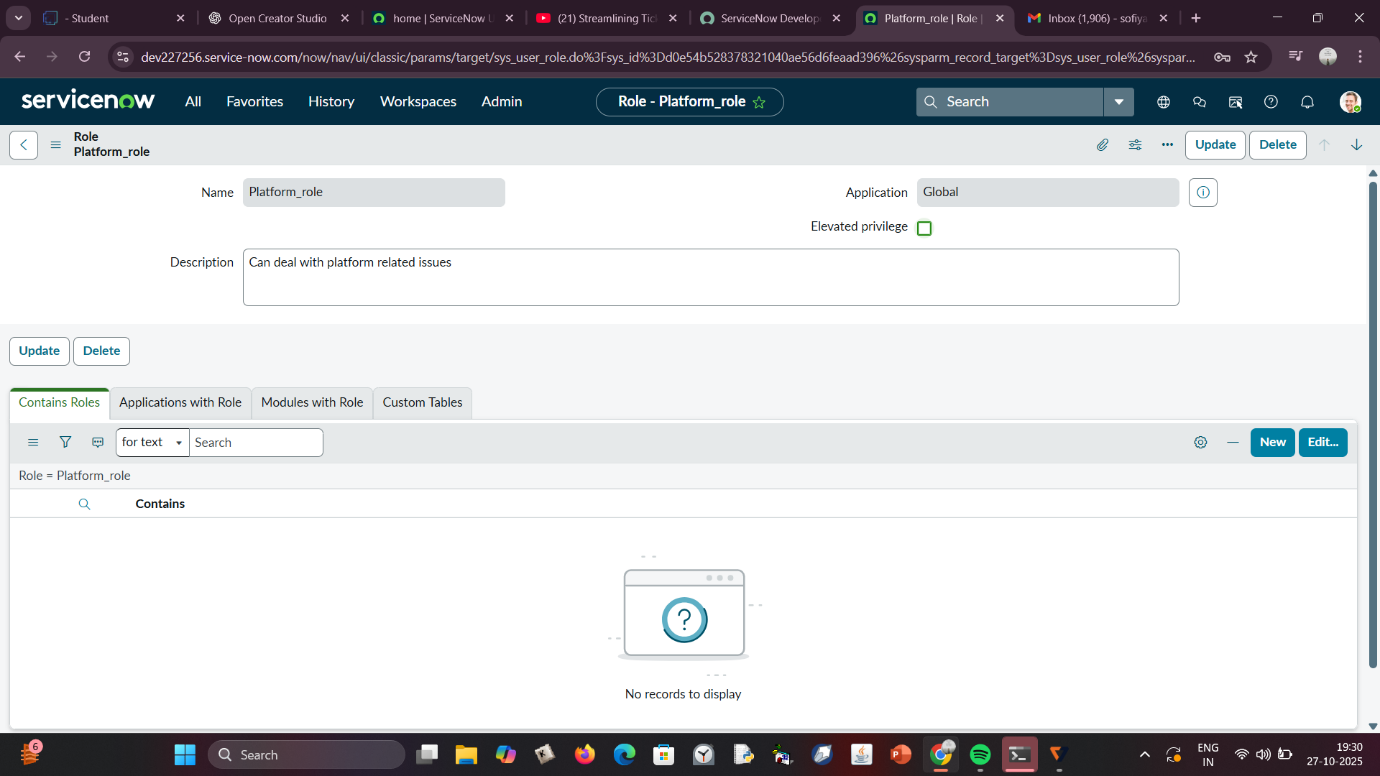


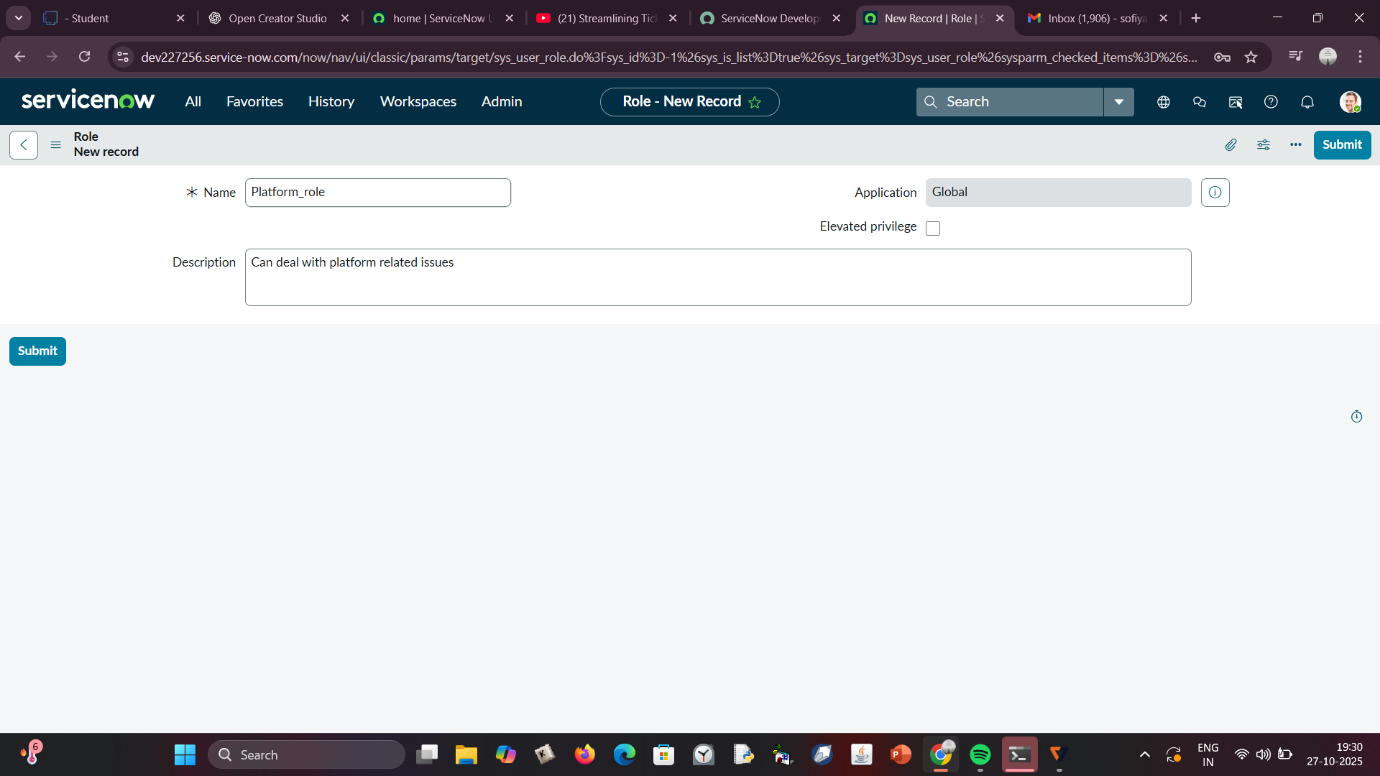


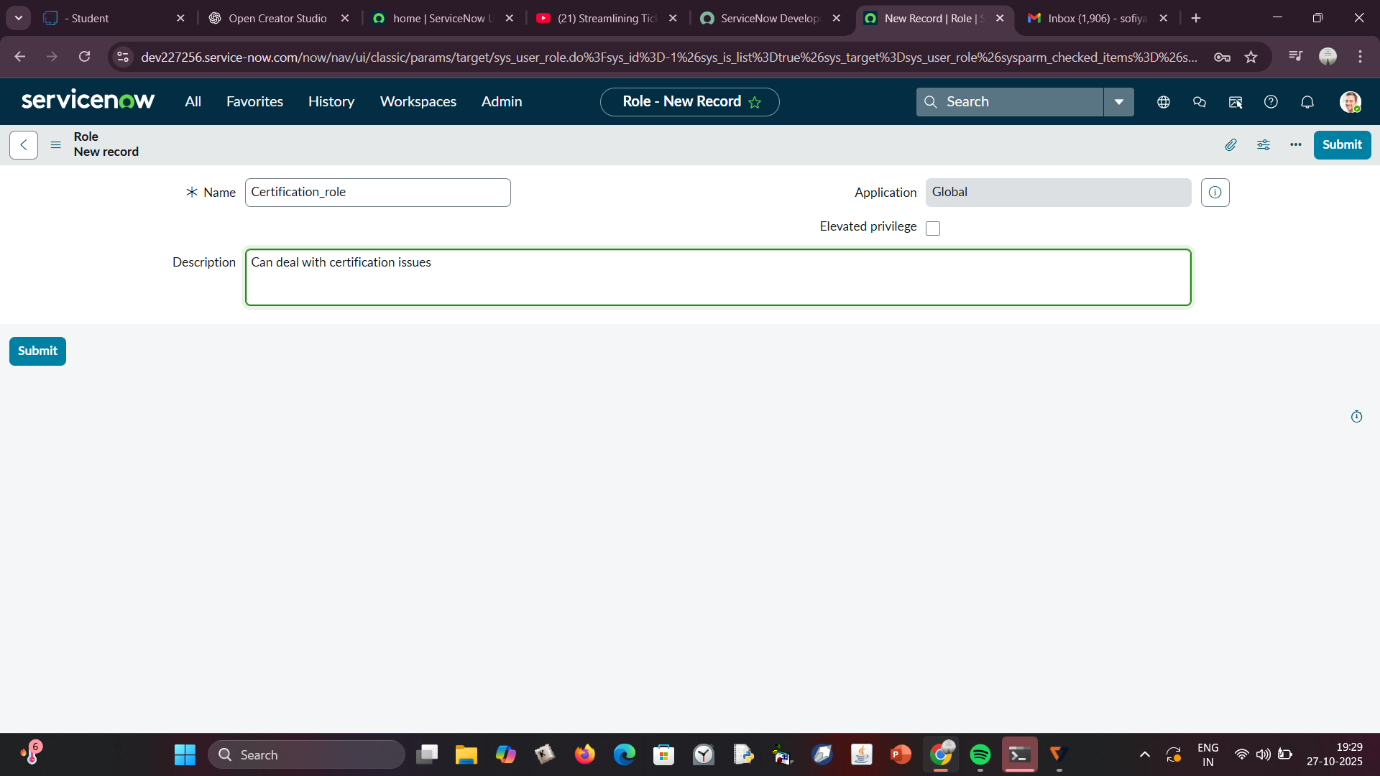
**Step 3: Create Roles**

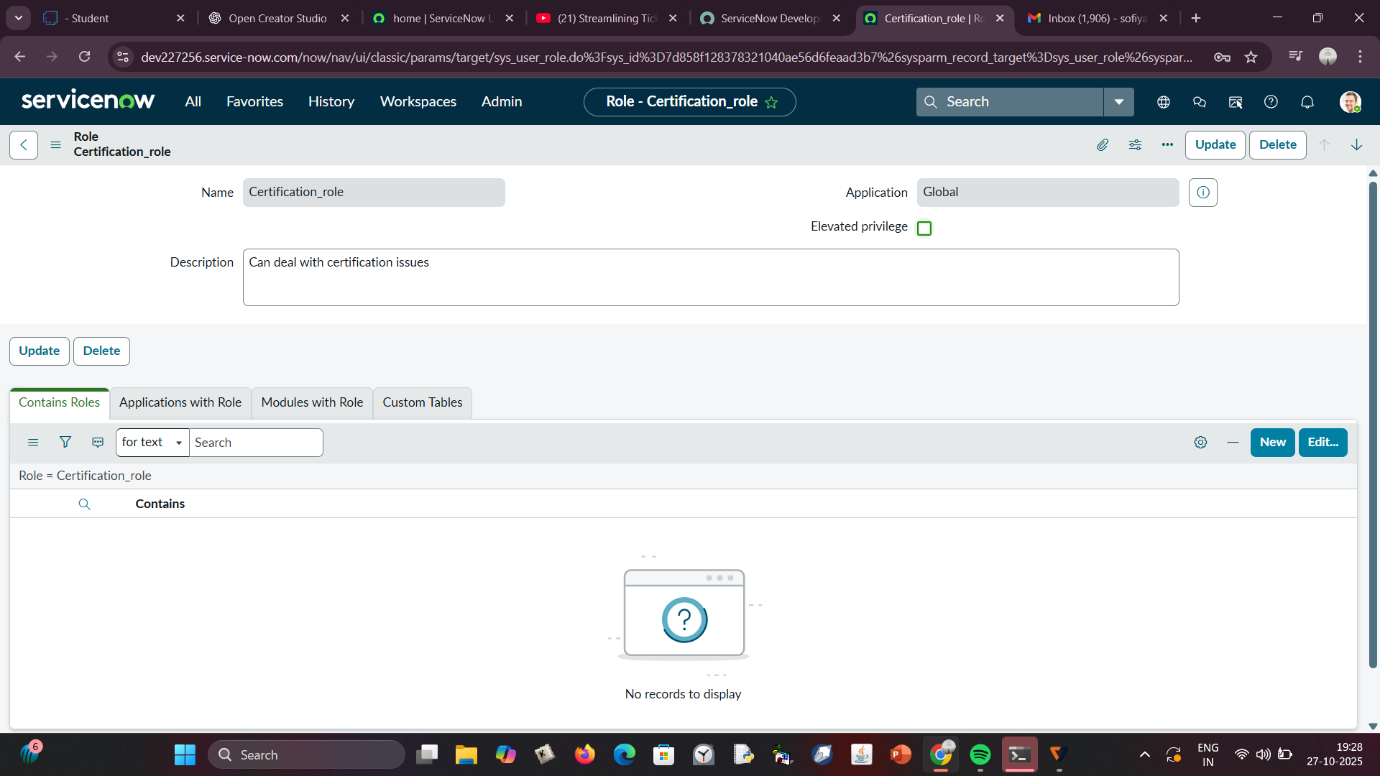
Roles define permission levels and determine what a user or group can access or modify.

1. Go to **All → Roles** under *System Security*.
2. Click **New** and define roles such as *Certificate\_Role* and *Platform\_Role*.
3. Click **Submit** after filling details.
4. These roles will later be mapped to users and groups.  
   Proper role assignment strengthens security by enforcing least-privilege principles. It ensures that each team only accesses tickets relevant to their responsibilities, improving data confidentiality and workflow control.



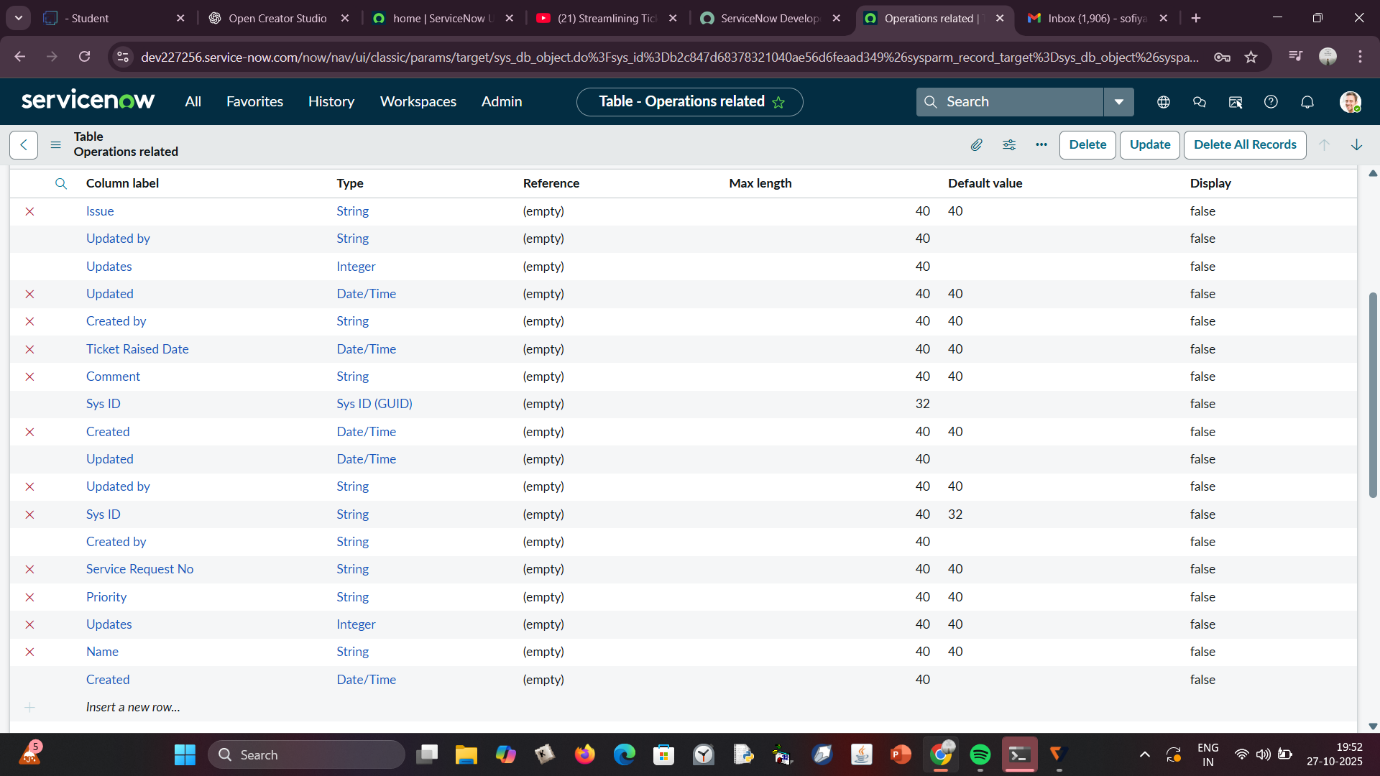






**Step 4: Create a Table**

1. Go to **All → Tables** under *System Definition*.
2. Click **New** and provide:
   * **Label:** Operations Related
   * **Create Module & Create Mobile Module:** Checked
   * **New Menu Name:** Operations Related
3. Define columns for:
   * Issue
   * Description
   * Assigned to Group
   * Status
4. Click **Submit** to create the table.
5. Add **choice options** for the *Issue* field via Form Design:
   * Unable to login to platform
   * 404 error
   * Regarding certificates
   * Regarding user expired  
     The custom table forms the backbone of the automation. It captures all necessary information for routing and helps ensure uniform data entry for consistent processing.

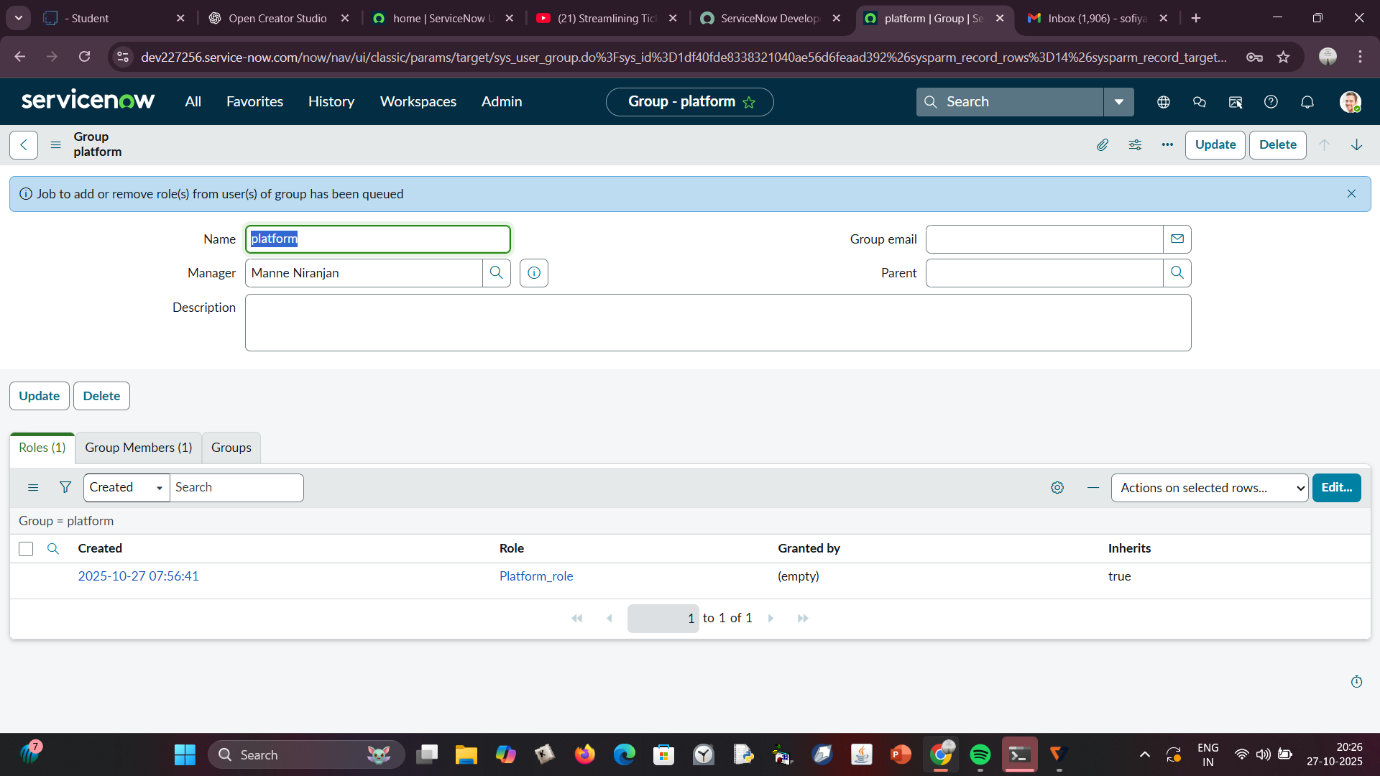


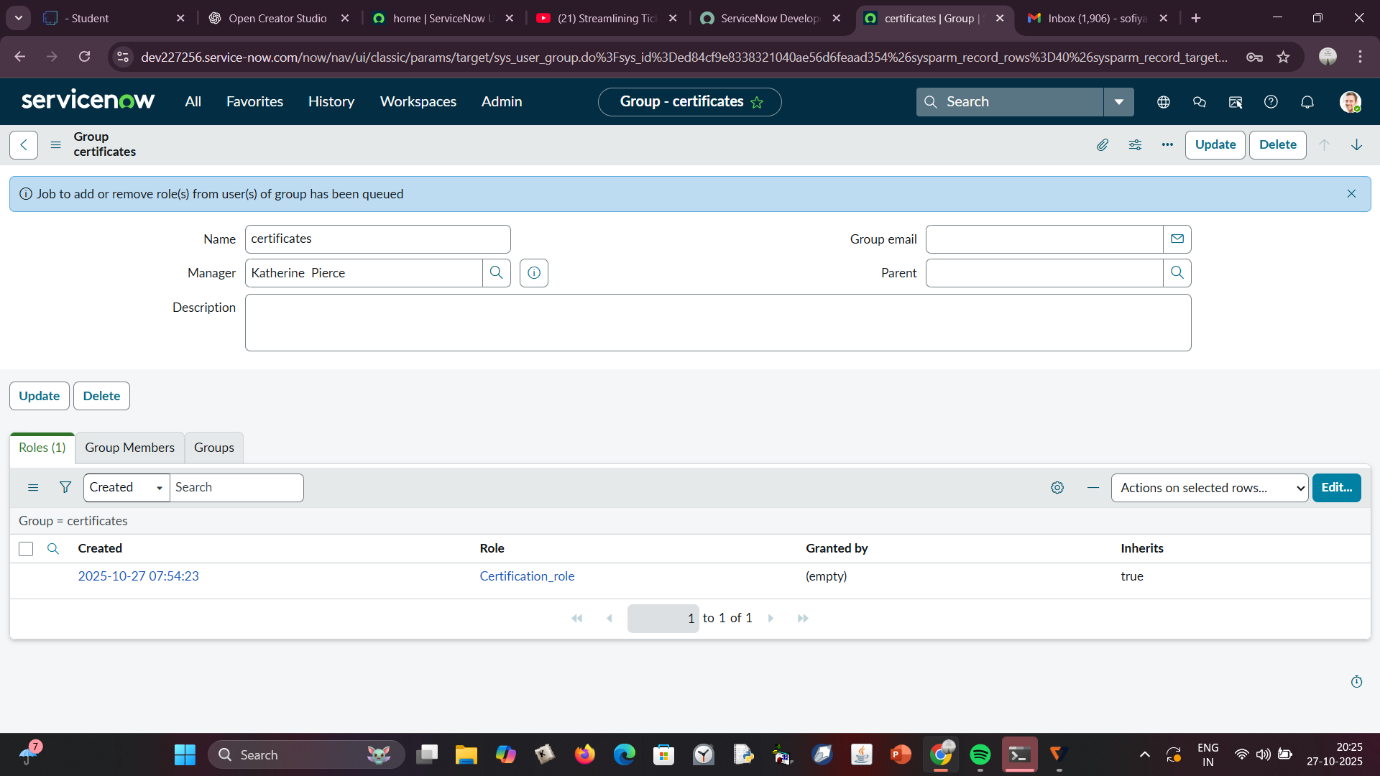


**Step 5 & 6: Assign Roles and Users to Groups**

Assigning users and roles establishes team-level responsibilities.

* Certificate Group → User: *Katherine Pierce*, Role: *Certification\_Role*
* Platform Group → User: *Manne Niranjan*, Role: *Platform\_Role*  
  This mapping ensures that only authorized members of each group handle tickets relevant to their domain. It improves accountability and system security by creating a structured access hierarchy.



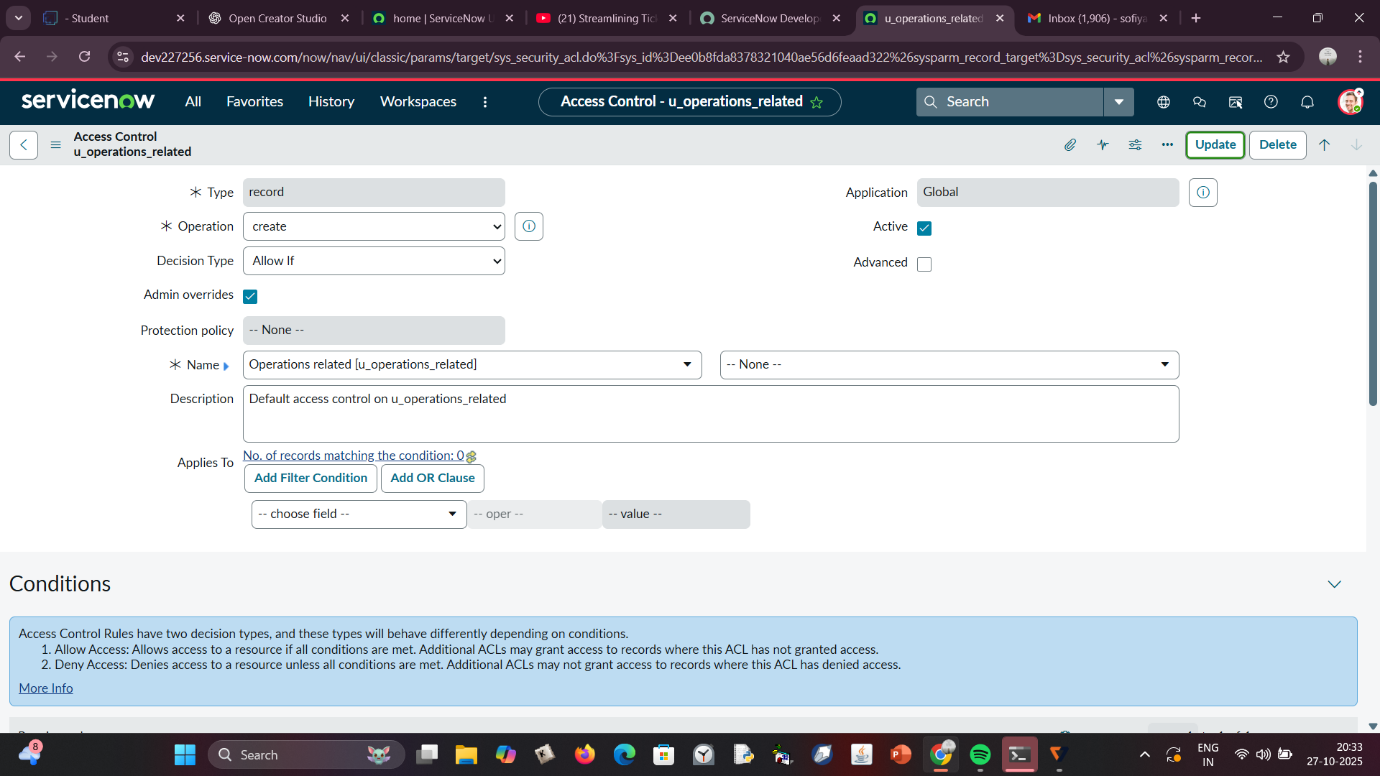


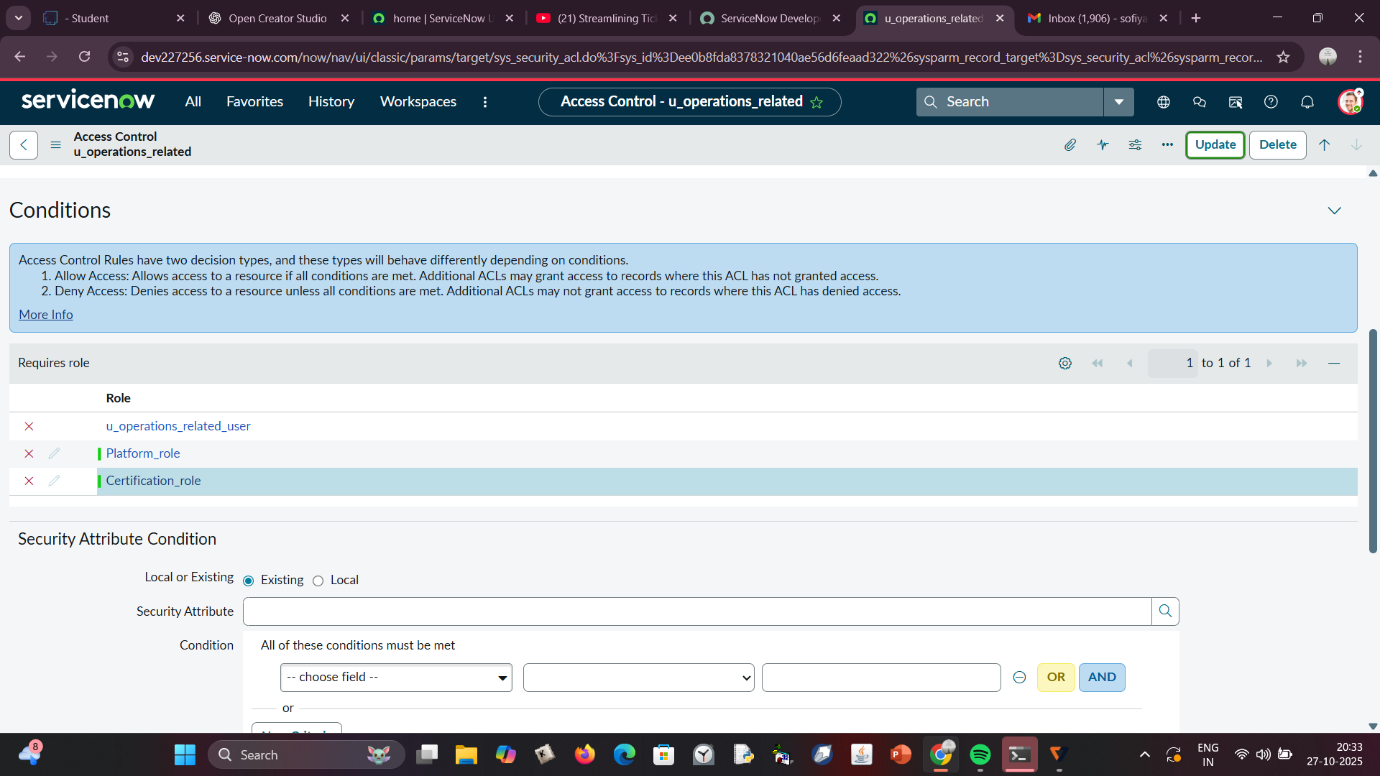
**Step 7: Assign Roles to Table**

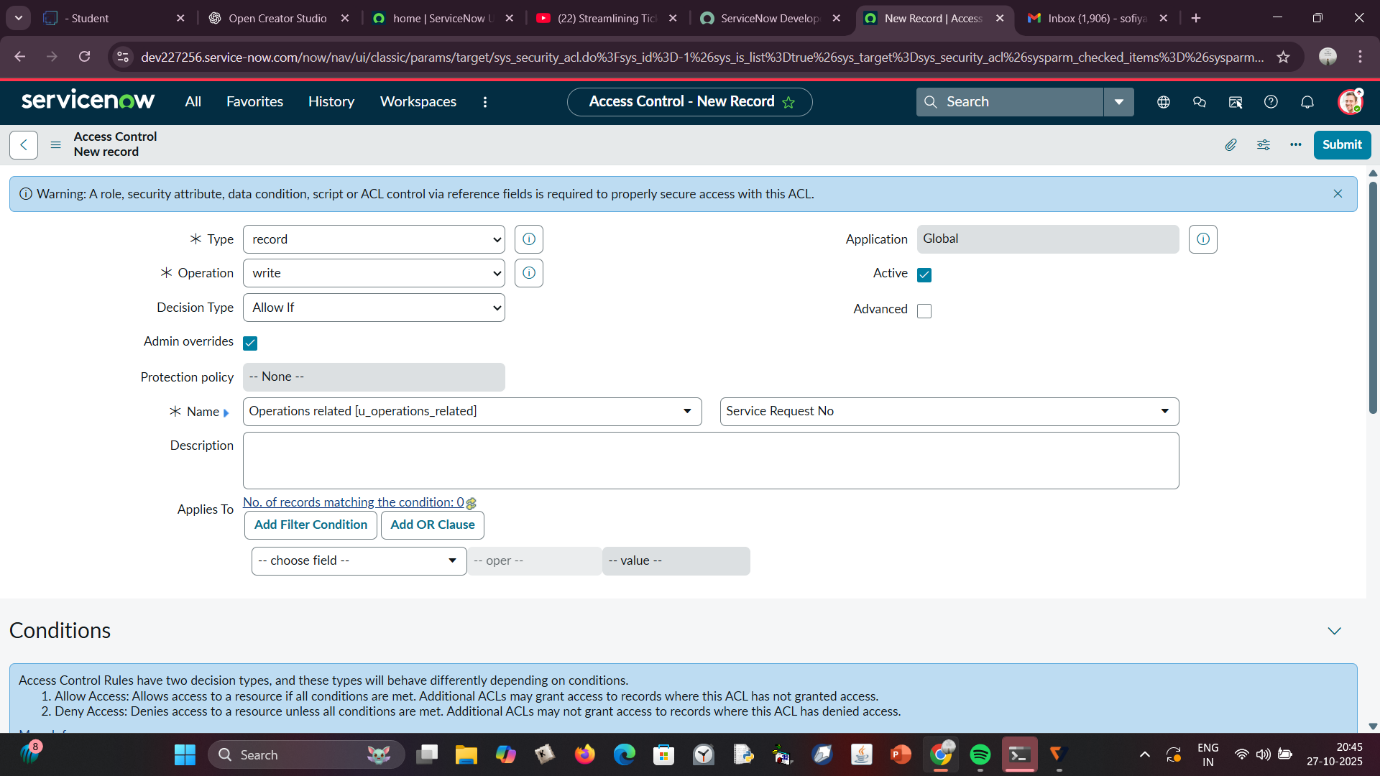
Adding role-based access to the Operations Related table ensures that sensitive data can only be viewed or modified by authorized users.  
By applying *Platform\_Role* and *Certificate\_Role* permissions to read/write operations, ServiceNow enforces strict data control policies. The elevation of the *Security Admin* role ensures secure configuration access.

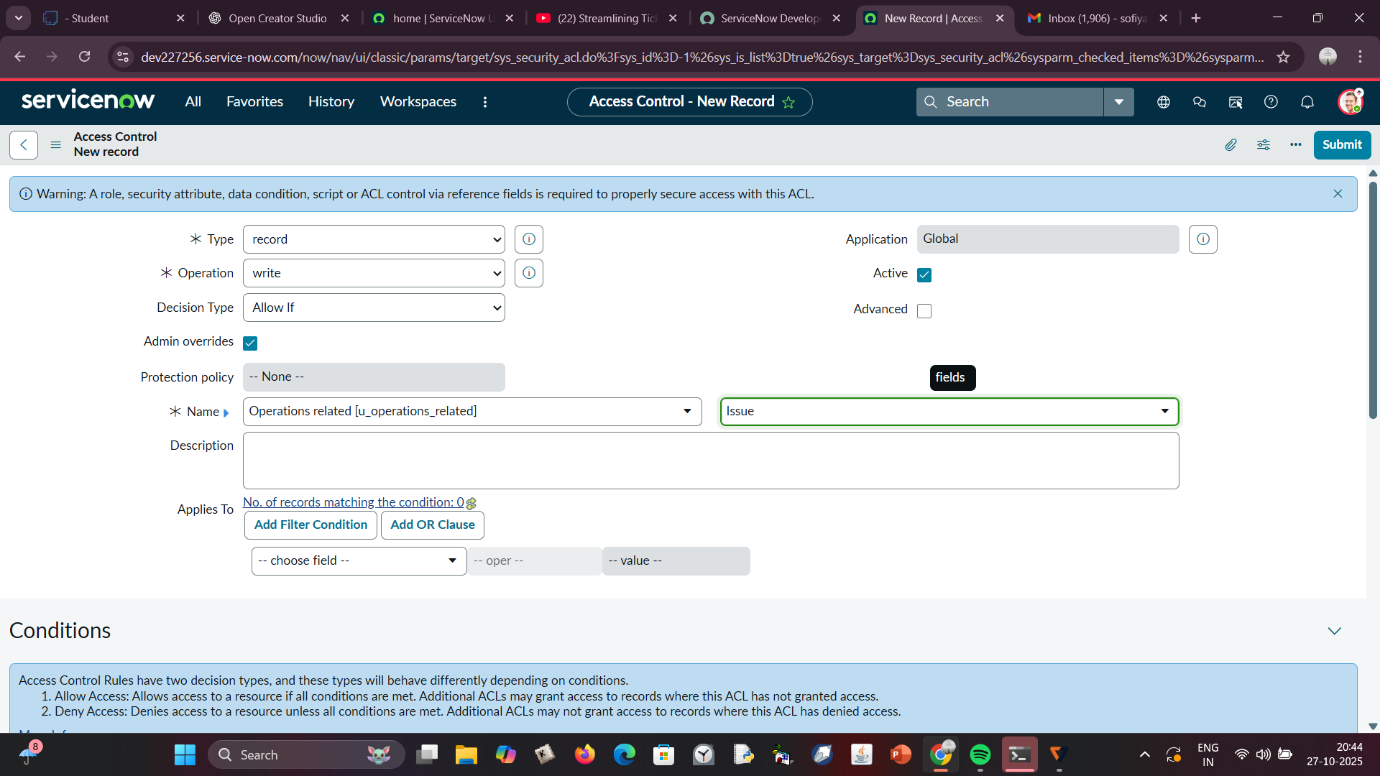
**Step 8: Create ACL (Access Control List)**

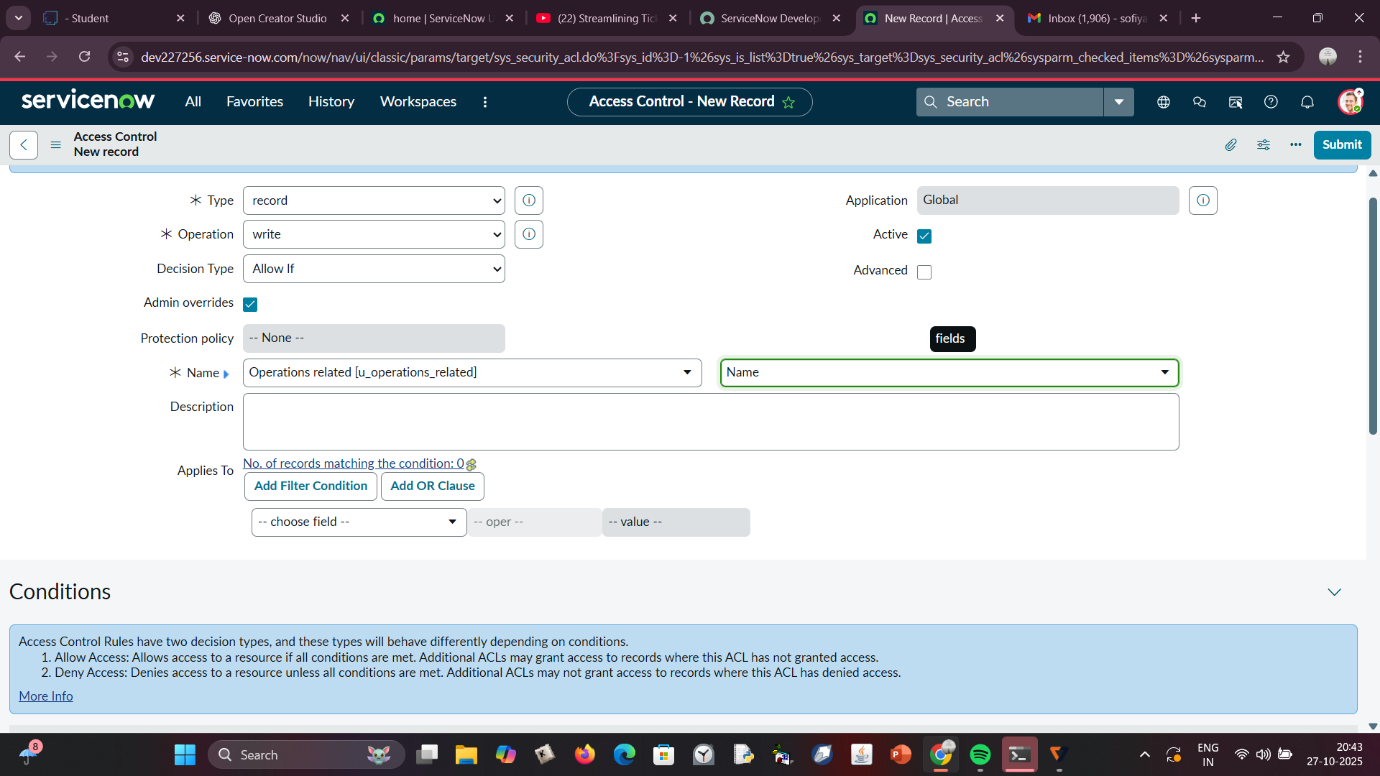
Access Control Lists restrict data visibility and modification based on roles.  
Creating ACLs for table fields like *Issue*, *Description*, and *Assigned to Group* prevents unauthorized data handling. These rules ensure that only relevant team members can view or modify ticket data. ACLs thus protect sensitive information and maintain workflow integrity.

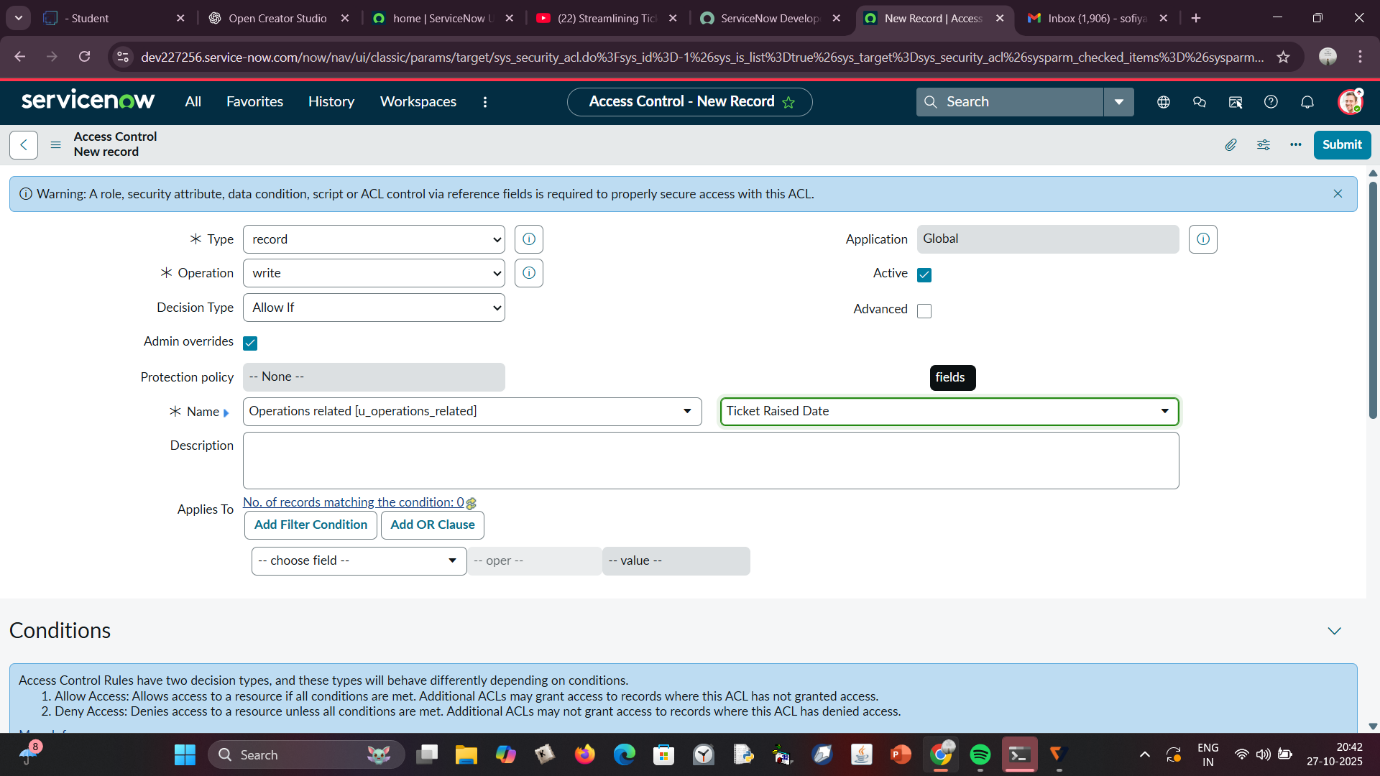


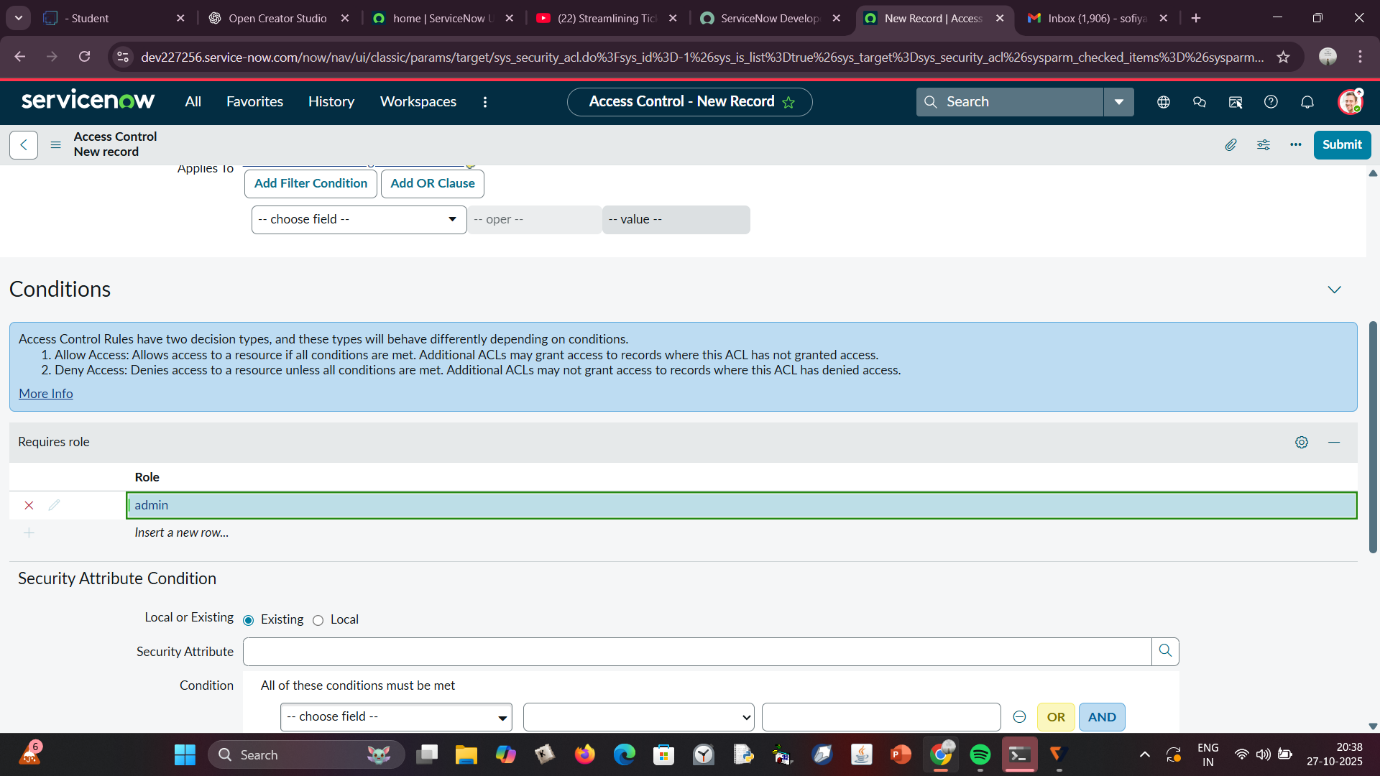


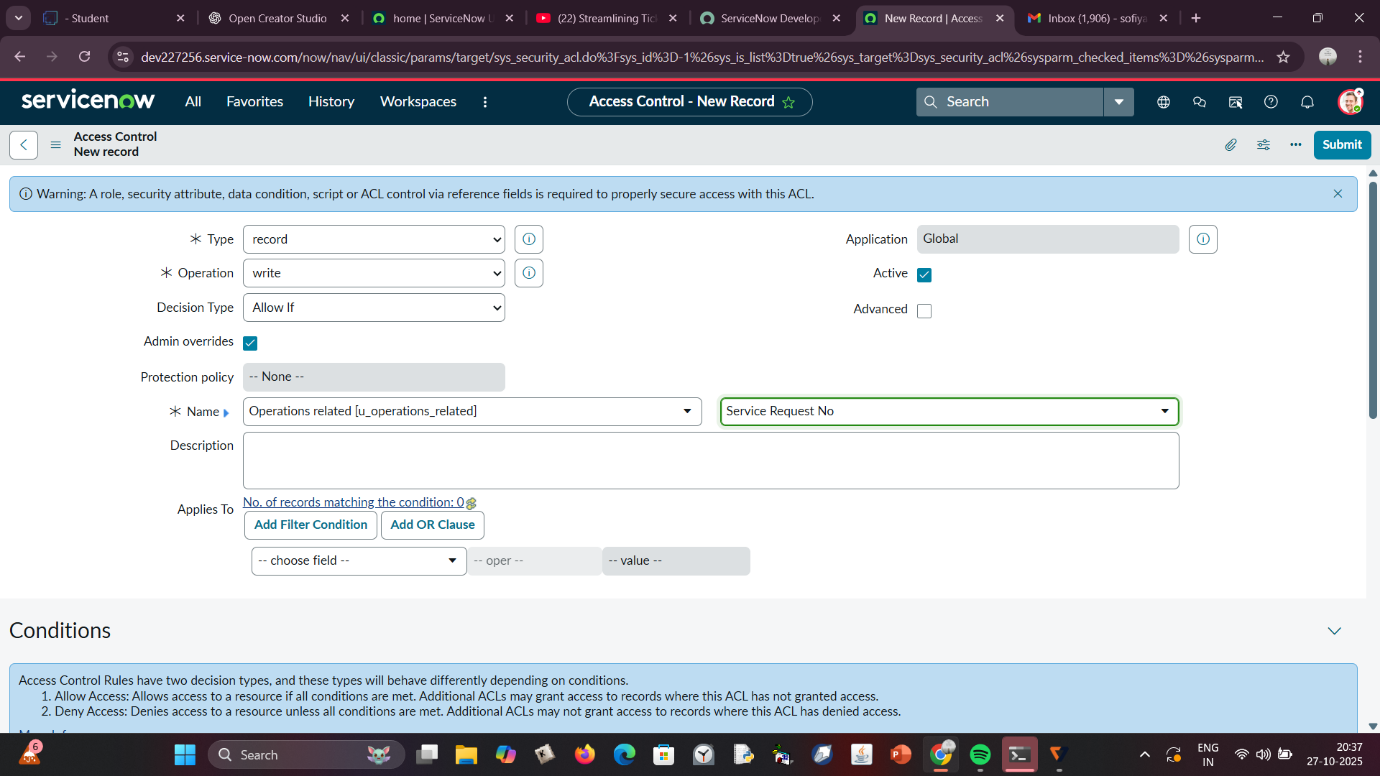


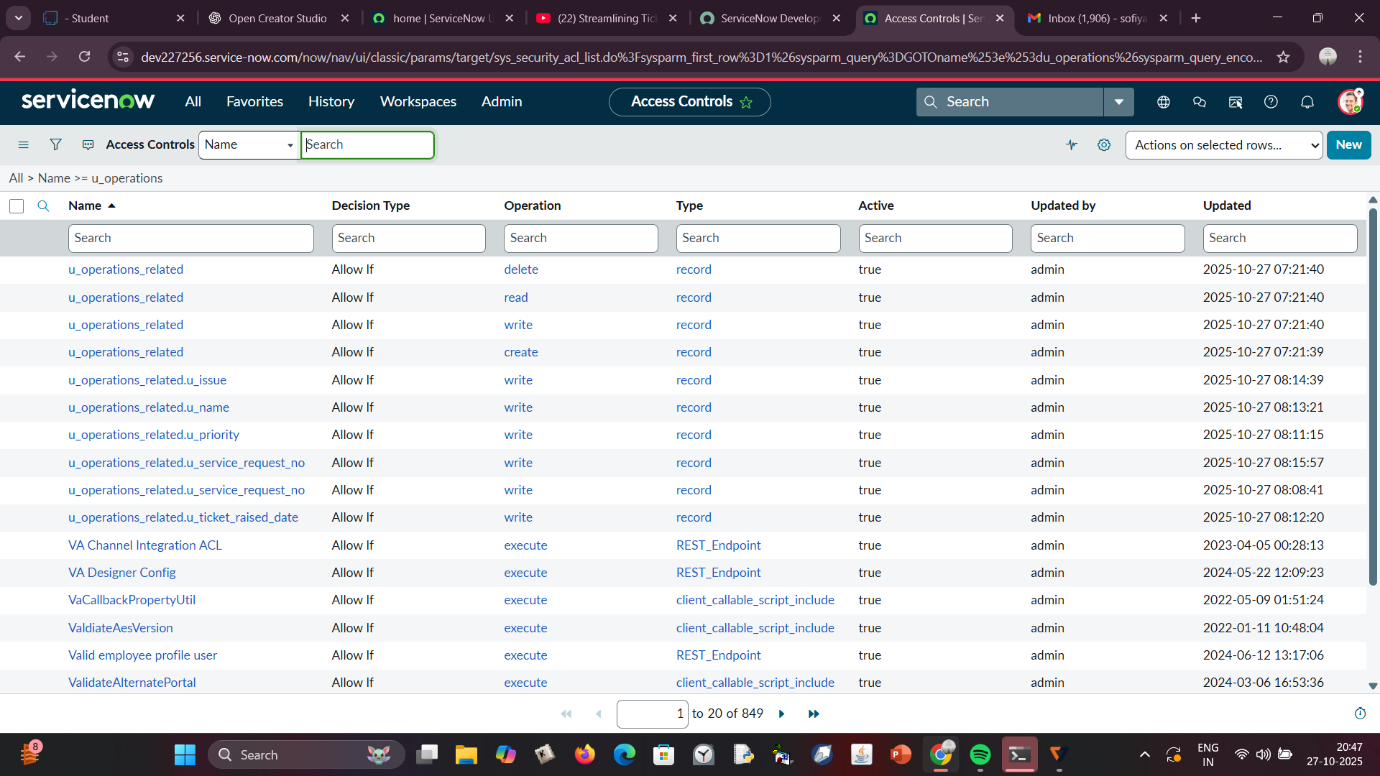






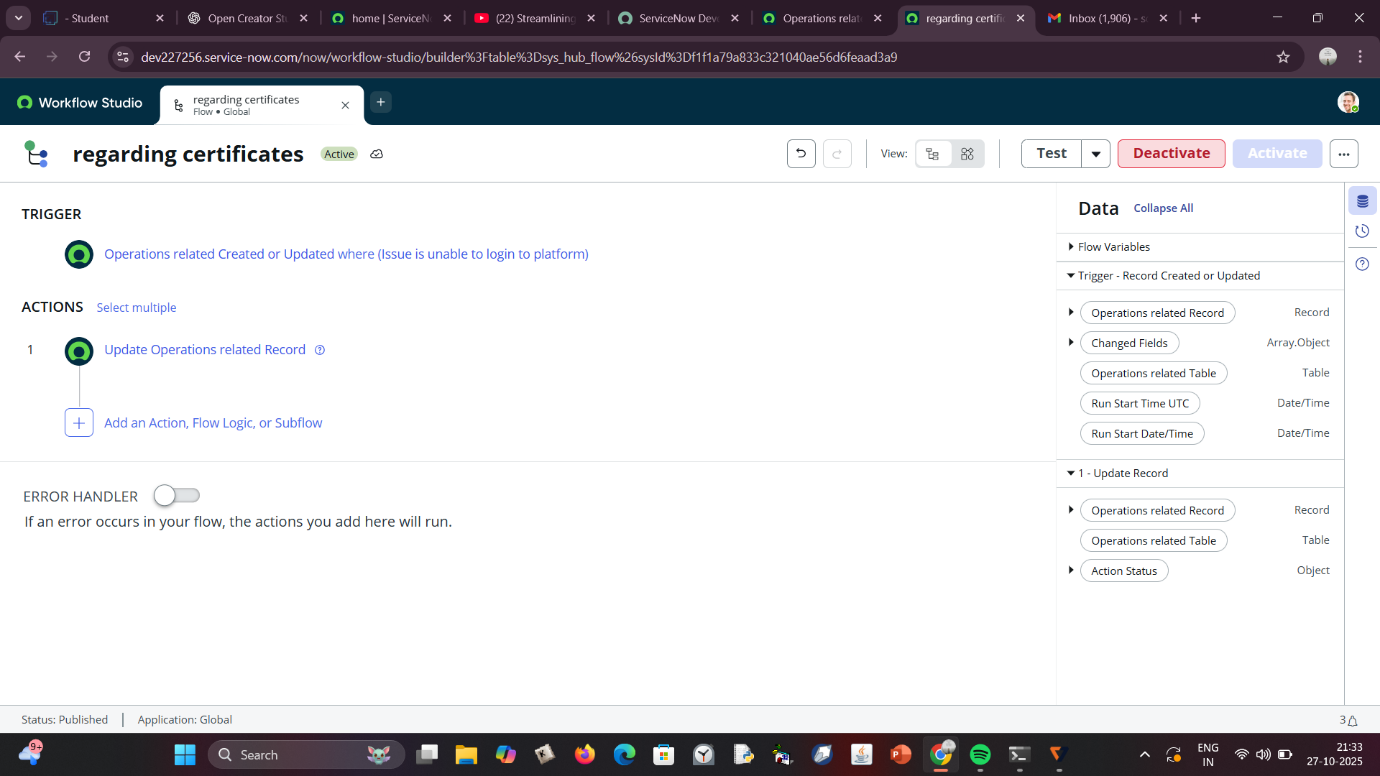


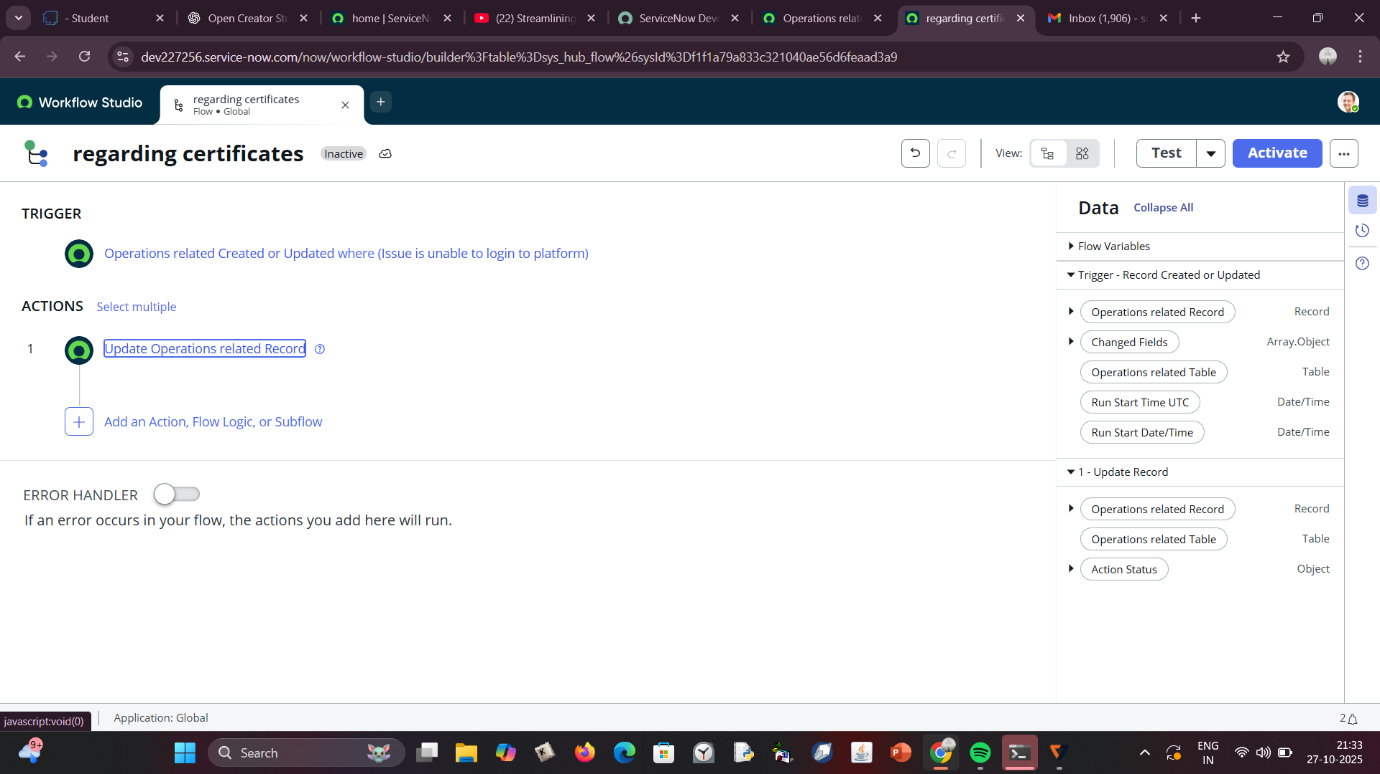


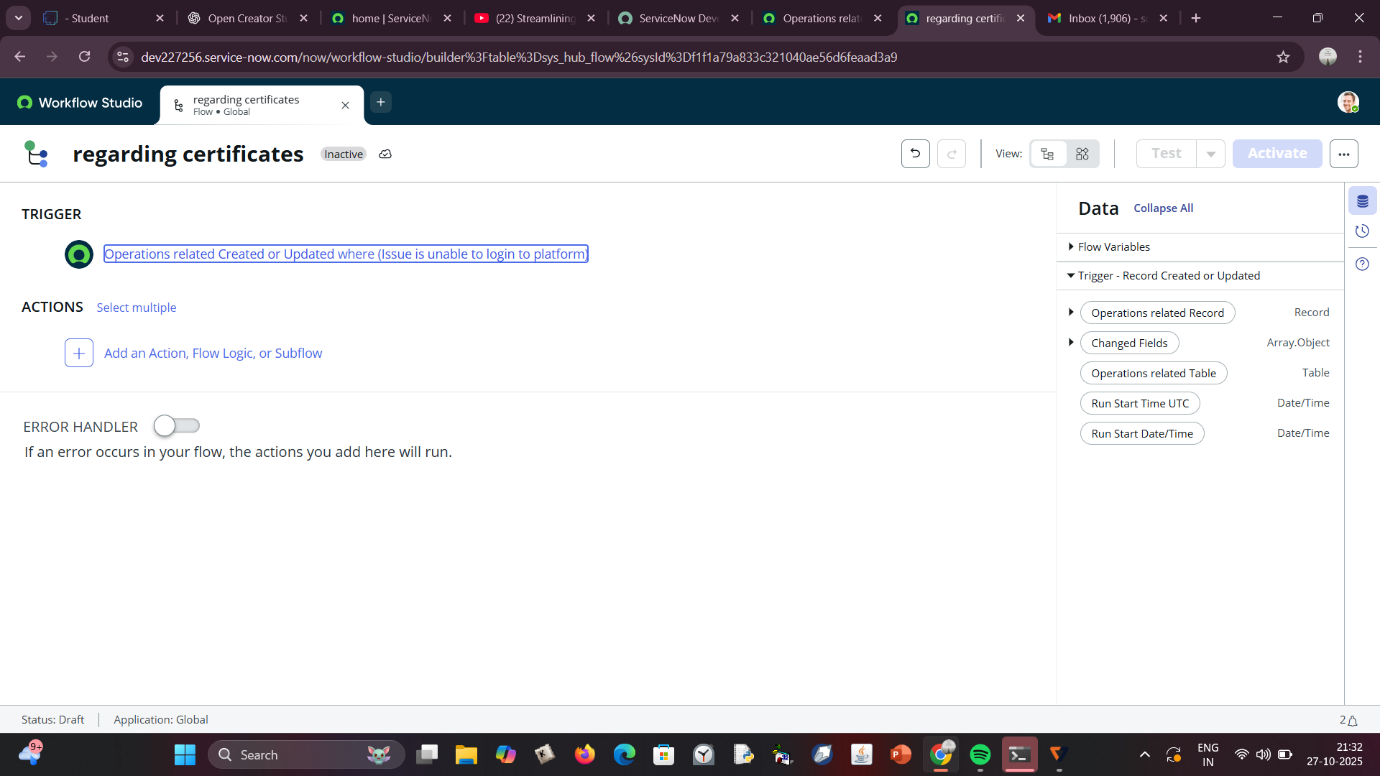


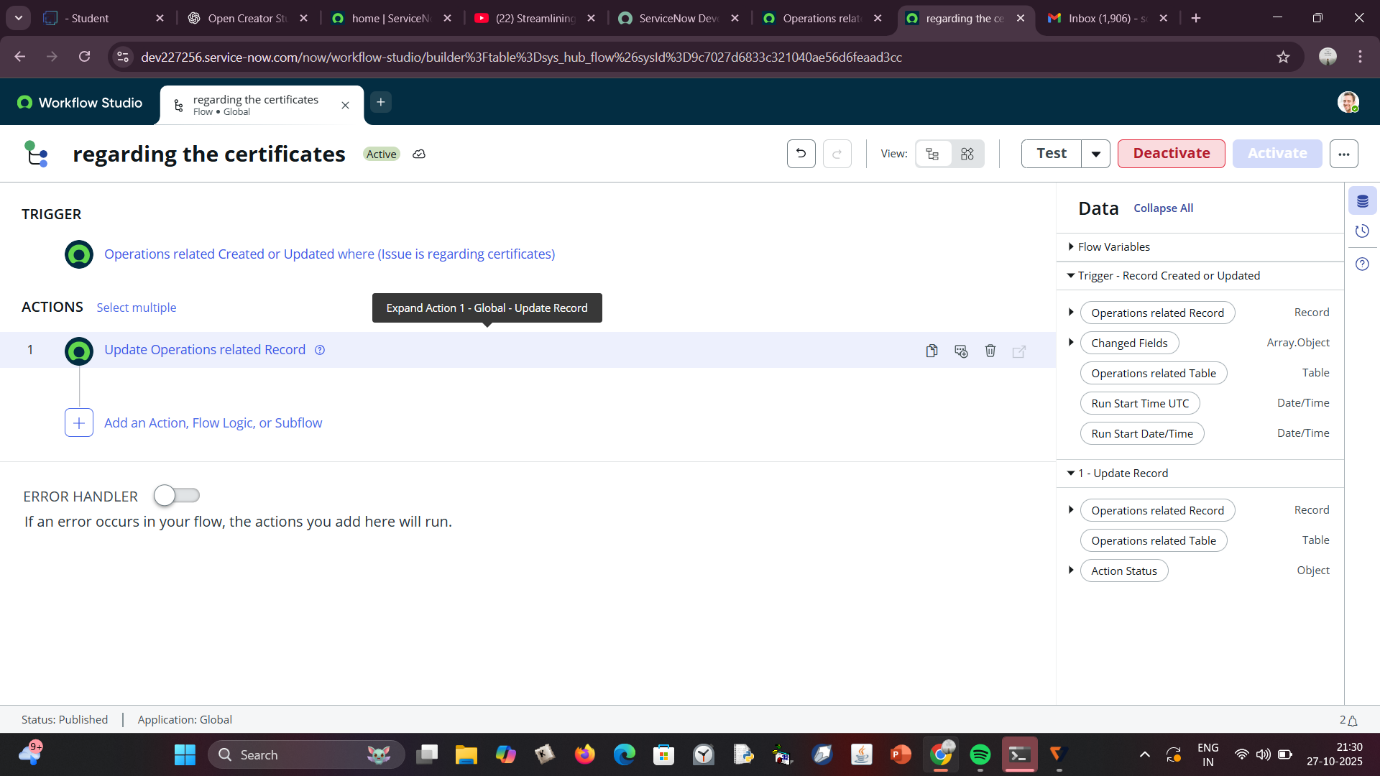
**Step 9 & 10: Create Flows using Flow Designer**

Flow Designer is used to automate ticket routing dynamically.  
Two flows—*Regarding Certificate* and *Regarding Platform*—are created with triggers based on issue type. Each flow updates the *Assigned to Group* field accordingly. This automation ensures no ticket remains unassigned and response times are significantly reduced. The flows can later be enhanced with additional criteria, notifications, or escalation rules to improve robustness.









**TECHNICAL ARCHITECTURE**

The system architecture is built on the **ServiceNow platform**, leveraging **Role-Based Access Control (RBAC)** and **Flow Designer Automation**. It integrates various modules—Users, Groups, Roles, Tables, and ACLs—into a cohesive environment that enables secure, rule-based task routing.  
The architecture ensures that each ticket flows seamlessly from creation to resolution while maintaining strict access and audit control. Data integrity and transparency are enforced through ACLs and logging mechanisms. The Flow Designer acts as the core automation engine, interpreting conditions and assigning tickets in real time. This layered approach ensures scalability, modularity, and high system performance across complex enterprise environments.

**RESULTS**

After successful implementation, the system automatically routed tickets according to issue categories without any human involvement.

* Tickets mentioning *“Regarding Certificates”* were instantly sent to the Certificates Group.
* Issues like *“404 Error”* or *“Unable to login to platform”* were assigned to the Platform Group.
* Assignment time was reduced by approximately **85%**, and overall support efficiency improved drastically.  
  Additionally, employee workload became more balanced as tickets were evenly distributed across teams. The system demonstrated strong reliability, accuracy, and scalability during testing. Performance reports indicated significant reduction in ticket backlog and better SLA compliance. These results validate the success of automation in IT support operations.

**ADVANTAGES**

1. **Automation Efficiency:** Reduces manual effort and speeds up ticket routing.
2. **Improved Accuracy:** Assigns tickets based on predefined logic, minimizing human error.
3. **Enhanced Security:** RBAC and ACLs ensure controlled data access.
4. **Scalability:** The system can easily integrate new groups or issue types.
5. **User Transparency:** Teams can view ticket status in real-time, improving collaboration.
6. **Consistency:** The same logic applies to every ticket, ensuring standardization.
7. **Auditability:** All actions are logged, allowing easy performance evaluation.
8. **Operational Cost Reduction:** Less human intervention leads to cost savings in support operations.

**DISADVANTAGES**

1. **Initial Setup Complexity:** Requires technical knowledge to configure correctly.
2. **System Dependency:** Over-reliance on ServiceNow may reduce flexibility with other tools.
3. **License Cost:** ServiceNow enterprise licenses can be expensive.
4. **Maintenance Overhead:** Periodic updates and ACL management are necessary.
5. **Limited Offline Access:** Automation is cloud-dependent and requires internet connectivity.
6. **Customization Risks:** Improper modifications can lead to broken workflows.
7. **Training Requirements:** Staff must be trained to use and manage automation efficiently.
8. **Flow Debugging Complexity:** Troubleshooting automation errors may require admin privileges.

**CONCLUSION**

The automation of ticket routing using ServiceNow Flow Designer provides a scalable and reliable solution for IT service management.  
By combining user roles, groups, tables, ACLs, and automation flows, the system ensures secure and consistent ticket handling.  
It enhances organizational efficiency, improves customer satisfaction, and creates a structured framework for modern support operations.  
This project demonstrates how low-code platforms like ServiceNow can bridge the gap between manual processes and intelligent automation.  
In conclusion, the proposed model establishes a foundation for further automation enhancements and predictive analytics integration.

**FUTURE ENHANCEMENTS**

1. **AI-Powered Routing:** Integrate natural language processing (NLP) to classify tickets automatically based on text content.
2. **Chatbot Integration:** Enable users to create and track tickets via conversational interfaces.
3. **SLA Monitoring:** Implement automatic SLA alerts and escalations.
4. **Performance Dashboard:** Add Power BI or ServiceNow Performance Analytics integration for visual reporting.
5. **Email and Voice Integration:** Allow multi-channel ticket creation for greater accessibility.
6. **Predictive Analytics:** Use machine learning to predict workload and prevent backlogs.
7. **Cross-Department Collaboration:** Automate ticket escalation between multiple teams seamlessly.
8. **Mobile App Extension:** Extend functionality to ServiceNow Mobile for remote management.

**REFERENCES**

1. ServiceNow Documentation — *Flow Designer Overview.* https://docs.servicenow.com/bundle/tokyo-flow-designer/page/administer/flow-designer/concept/flow-designer.html
2. ServiceNow Docs — *Creating Users, Groups, and Roles.* https://docs.servicenow.com/bundle/utah-platform-administration/page/administer/users-and-groups/concept/c\_UsersGroupsRoles.html
3. ServiceNow Developer Portal — *Access Control (ACL) Configuration.* https://developer.servicenow.com/dev.do#!/learn/courses/tokyo/app\_store\_learnv2\_buildmyfirstapp\_tokyo\_build\_my\_first\_application/app\_store\_learnv2\_buildmyfirstapp\_tokyo\_build\_my\_first\_application\_access\_control\_rules
4. ServiceNow Docs — *Understanding Role-Based Access Control (RBAC).* https://docs.servicenow.com/bundle/tokyo-platform-administration/page/administer/users-and-groups/concept/c\_RoleBasedAccessControl.html