



Table of Contents

| CHAPTER | CONTENT |
|----------------|------------------------|
| 1. | Problem Statement |
| 2. | Abstract |
| 3. | Introduction |
| 4. | Objectives |
| 5. | Methodology |
| 6. | Architecture |
| 7. | Workflow Explanation |
| 8. | Benefits of Automation |
| 9. | Results and Outcomes |
| 10. | Future Scope |
| 11. | Appendix |
| 12. | Conclusion |

1. Problem Statement

In many organizations, managing users, groups, and roles across multiple systems is a complex and time-consuming task. Manual access management often leads to inconsistencies, security risks, and administrative overload. When employees join, move, or leave, access rights are rarely updated on time, resulting in unauthorized access or workflow interruptions. Traditional methods lack centralized control, audit trails, and automation, making compliance and governance difficult.

This project aims to solve these challenges by designing an optimized system that automates user provisioning, enforces role-based access control (RBAC), and streamlines approval workflows through a secure and centralized platform..

2. Abstract

This project focuses on the development of a smart access management system that simplifies and automates user, group, and role handling within an organization. The proposed system integrates access control principles and workflow automation to ensure secure, efficient, and compliant management of user privileges.

By implementing Role-Based Access Control (RBAC), the system ensures that every user receives permissions aligned with their role, reducing human errors and potential security breaches. Additionally, workflow automation allows for dynamic approvals, auditing, and real-time monitoring.

The outcome is an intelligent solution that minimizes administrative effort, enhances security, and optimizes operational efficiency in enterprise environments.

3. Introduction

Organizations today rely on digital systems that require precise and secure access control. Managing user permissions manually can cause inefficiency, data breaches, and compliance issues. As companies expand, maintaining accurate records of who can access what becomes increasingly difficult.

User, group, and role management forms the backbone of identity and access governance. Automating these processes through technology ensures that the right individuals have the right access to the right resources at the right time.

This project introduces a centralized, automated approach using RBAC principles and workflow automation to improve security, traceability, and operational productivity in access control management.

4. Objectives

The main objectives of this project are:

1. To design a centralized system for managing users, groups, and roles efficiently.
2. To implement Role-Based Access Control (RBAC) for consistent and secure permission assignment.
3. To integrate automated workflows for approvals and access modifications.
4. To minimize manual intervention and administrative workload.
5. To improve auditability, compliance, and transparency in access management.
6. To enhance overall security by reducing unauthorized or redundant access.

5. Methodology

The implementation of this project is carried out in several structured steps:

Step 1: Creating Users

- Navigate to: All → Users → New
- Create users such as Alice P and Bob P.
- Assign them specific roles and save.

Step 2: Creating Groups

- Navigate to: All → Groups → New
- Create groups:
 - Project Team
- Submit the groups.

Step 3: Creating Roles

- Navigate to: All → Roles → New
- Create roles:
 - Project Member
 - Team Member
- Assign each role to its corresponding group.

Step 4: Creating a Custom Table

- Go to: System Definition → Tables → New

- Label: Project table & Task Table 2
- Check:
 - Create Module
 - Create Mobile Module

Step 5: Assigning Users and Roles

- Assign Alice to the *project manager team* and give her *the following roles*:
 - *Project_Member*
 - *u_project_table*
 - *u_task_table*
- Assign Bob to the *Team Member team* and give him *the following roles*:
 - *Team_Member*
 - *Table_role*

Step 6: Assigning Table Access to the Applications

- While creating a table, an application and module are automatically created.
 - Navigate to: Application Navigator → Search “Project Table” application.
 - Click Edit Module → Assign Project_Member role.
 - Search Task Table 2 → Click Edit Application.
 - Assign Project_Member and Team_Member roles.

Step 7: Setting Access Controls (ACLs)

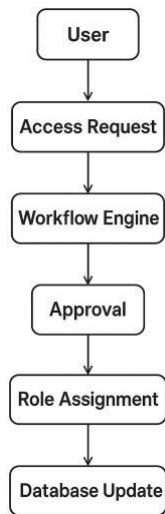
- Navigate to: System Security → Access Control (ACL)
- Create ACLs for the *Task Table*.
- Add the Team_Member role under *Requires Role*.
- Create four ACLs for the required fields.

Step 8: Designing the Flow

- Navigate to: Flow Designer → New Flow
- Create a flow “Task Table”
- Trigger: When a record is created in Operations Related
- Condition: Status is: In progress AND comments is feedback AND assigned to is bob
- Action: Update record → Status completed

6. Architecture Diagram

Below is the workflow architecture of the automation system



Flow Explanation:

1. User submits an access request for a specific role or module.
2. The request record is stored in the Access Management table.
3. The Workflow Engine triggers automatically when a new request is created.
4. The system checks the request type and identifies the required approval path.
5. The manager or admin reviews and approves/rejects the request.
6. Once approved, the system assigns the appropriate role to the user.
7. The database is updated with the new access details.
8. A notification is sent to both the user and approver confirming the update.

7. Workflow Explanation

Manual Workflow (Before Automation):

- User requests access manually through email or form submission
- Admin reviews the request and assigns roles manually.
- Delays occur due to manual approval and communication gaps.
- High chances of errors or unauthorized access due to lack of tracking.

Automated Workflow (After Implementation):

- User submits an access request through the system.
- The Workflow Engine automatically triggers and identifies the request type.
- The approval process is routed to the respective manager or admin.

- Upon approval, the system automatically assigns roles and updates the database.
- Notifications are sent to both user and admin confirming the update.

This automation ensures faster access provisioning, reduces manual effort, and improves security and accuracy in role and access management..

8. Benefits of Automation

1. **Reduced Administrative Overhead:** Automated workflows eliminate repetitive manual tasks.
2. **Enhanced Security:** Permissions are dynamically updated based on roles, reducing unauthorized access.
3. **Improved Compliance:** Automatic logging and reporting support audits and regulatory requirements.
4. **Time Efficiency:** Access provisioning and revocation happen instantly after approval.
5. **Scalability:** The system can handle large organizations with thousands of users easily.
6. **Transparency:** Every access change is visible and traceable.

9. Results and Outcomes

After implementing the system, the organization can expect::

- Up to 60% reduction in access management time.
- Fewer access-related errors and security incidents.
- Improved employee onboarding and offboarding experience.
- Real-time visibility into user access and activity.
- Centralized control and unified reporting for all departments.

Performance testing indicates a significant improvement in efficiency compared to manual methods.

10. Future Scope

- Integrate AI to suggest appropriate access levels based on user behavior.
- Extend support for cloud services like AWS IAM, Azure AD, or Google Workspace.
- Provide mobile-based access request and approval features.
- Using immutable ledgers for enhanced transparency and tamper-proof auditing.
- Introducing dashboards with anomaly detection and usage insights.

11. Appendix

1. Create User

ServiceNow User - alice p

User ID:

First name:

Last name:

Title:

Department:

Email:

Identity type:

Language:

Calendar integration:

Time zone:

Date format:

Business phone:

Mobile phone:

Photo: [Click to add...](#)

Update Set Password Delete

Related Links

- [View linked accounts](#)
- [View Subscriptions](#)
- [Reset a password](#)

Entitled Custom Tables Roles (4) Groups (2) Delegates Subscriptions User Client Certificates

Table Search

User = alice p

2. Create Groups

ServiceNow Group - project team

Name:

Group email:

Manager:

Parent:

Description:

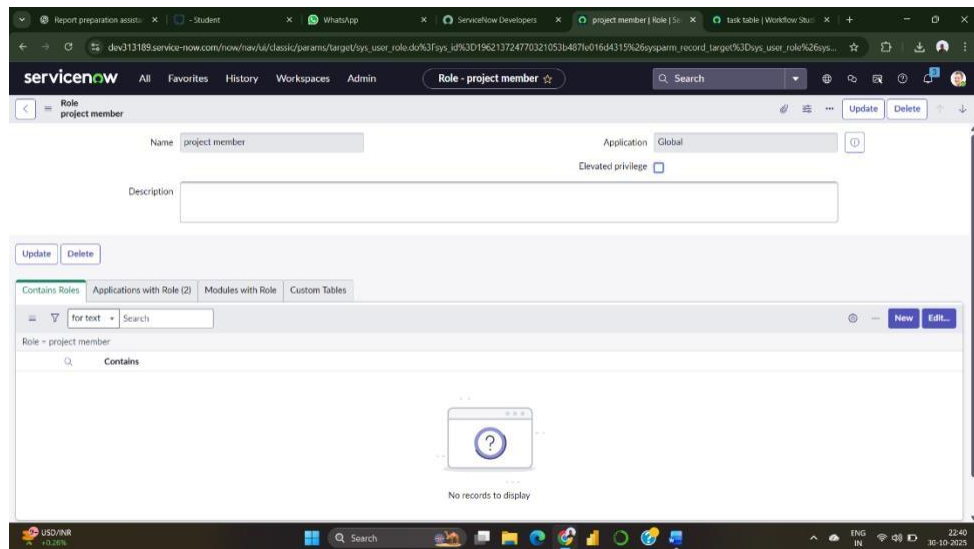
Update Delete

Roles Group Members (2) Groups

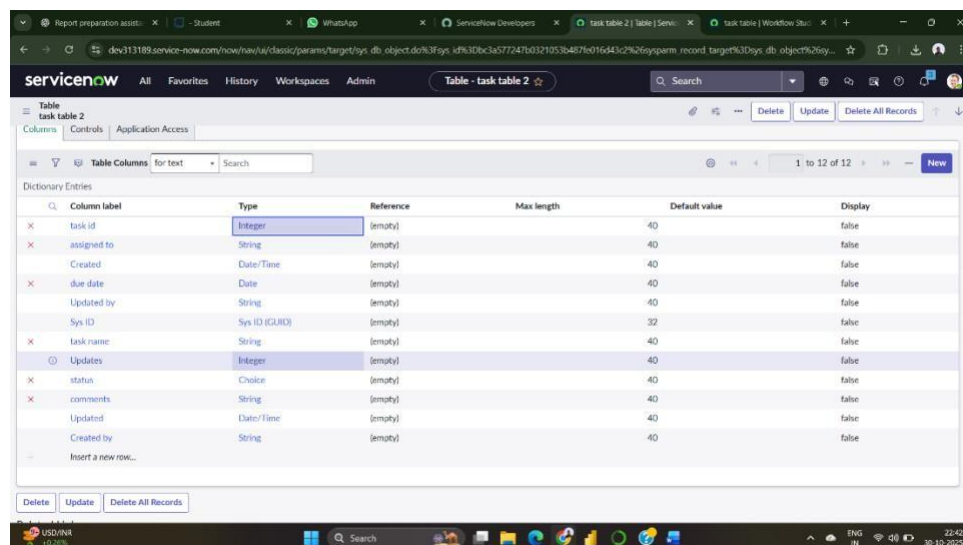
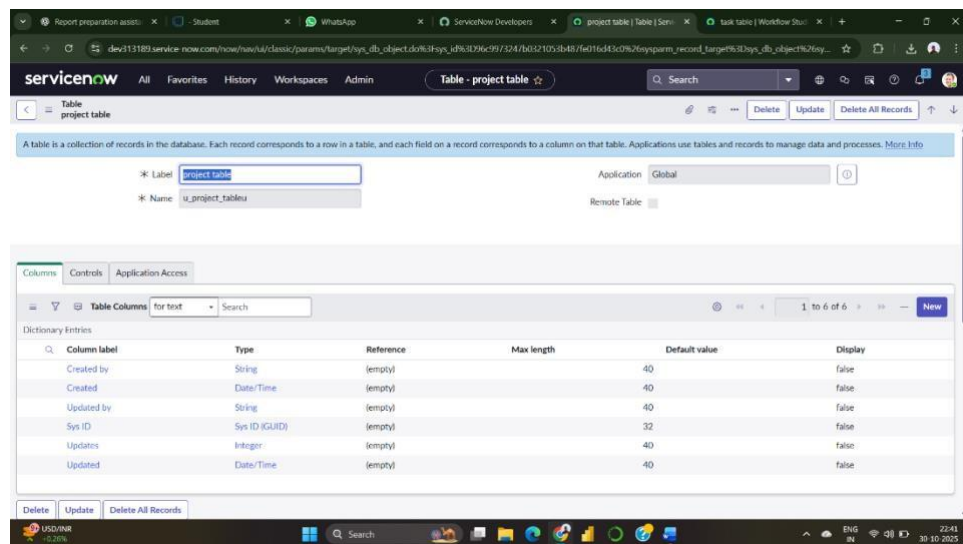
Created Search

| Created | Role | Granted by | Inherits |
|-----------------------|------|------------|----------|
| No records to display | | | |

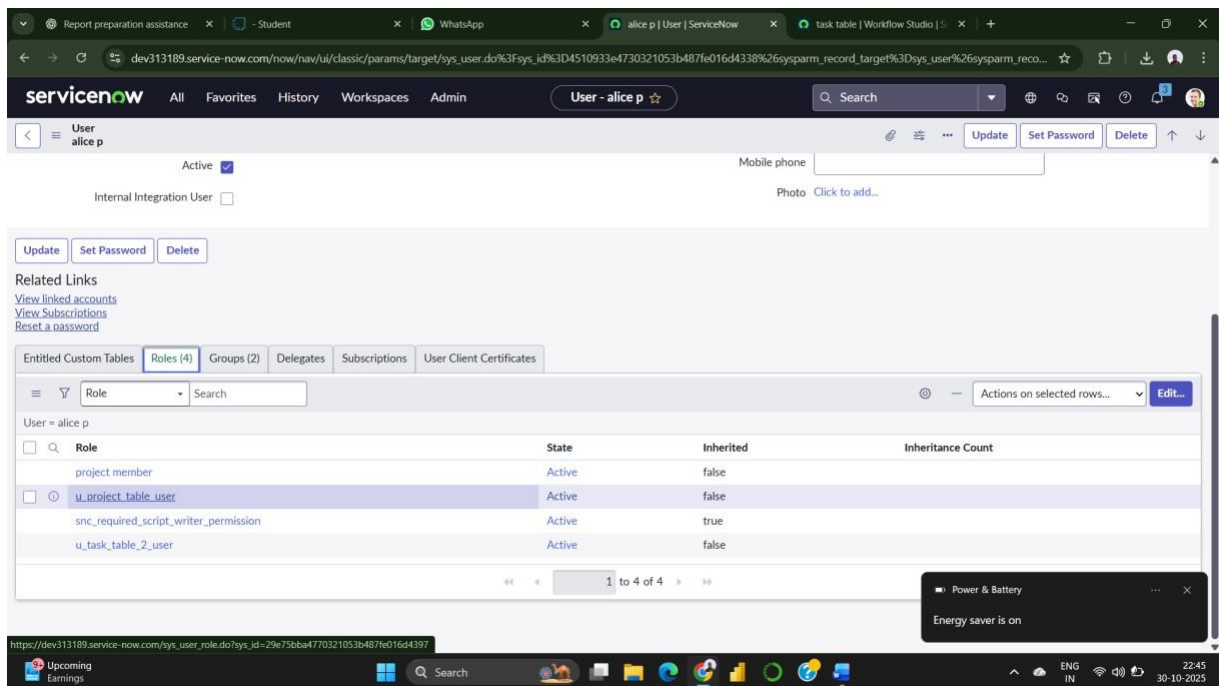
3. Create Role



4. Create a Table and Assign the operations



5. Assign Roles

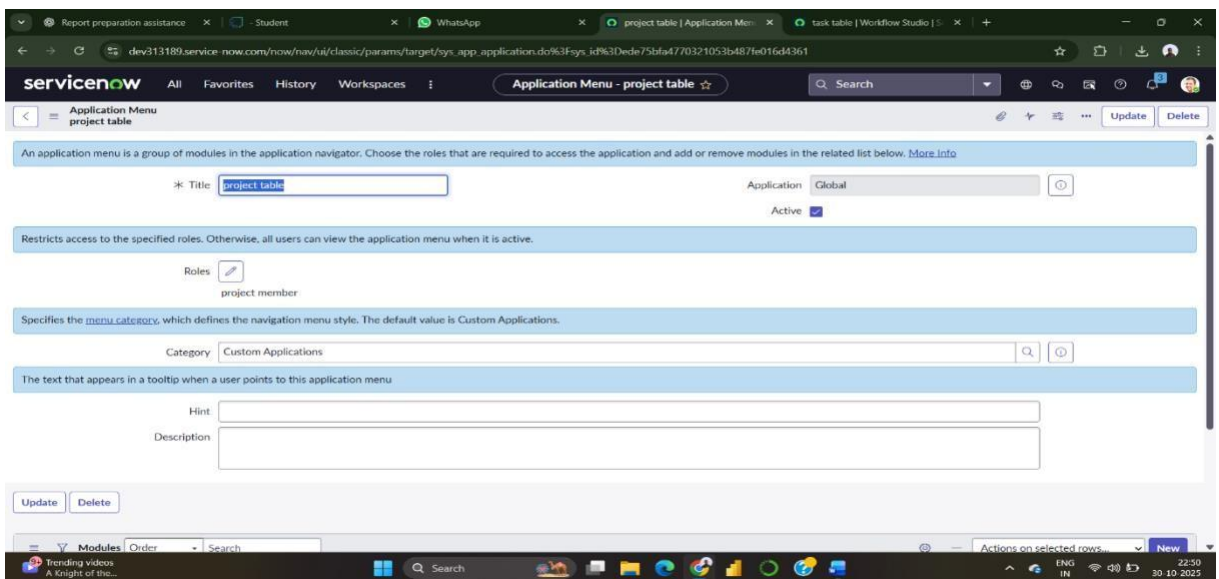


The screenshot shows the ServiceNow user profile for 'User - alice p'. The 'Roles' tab is selected, displaying a table of roles assigned to the user. The table has columns for Role, State, Inherited, and Inheritance Count.

| Role | State | Inherited | Inheritance Count |
|---------------------------------------|--------|-----------|-------------------|
| project member | Active | false | |
| u_project_table_user | Active | false | |
| snc_required_script_writer_permission | Active | true | |
| u_task_table_2_user | Active | false | |

Navigation links: Update, Set Password, Delete. Related Links: View linked accounts, View Subscriptions, Reset a password. Entitled Custom Tables: Roles (4), Groups (2), Delegates, Subscriptions, User Client Certificates.

6. Assign Table Access to Applications



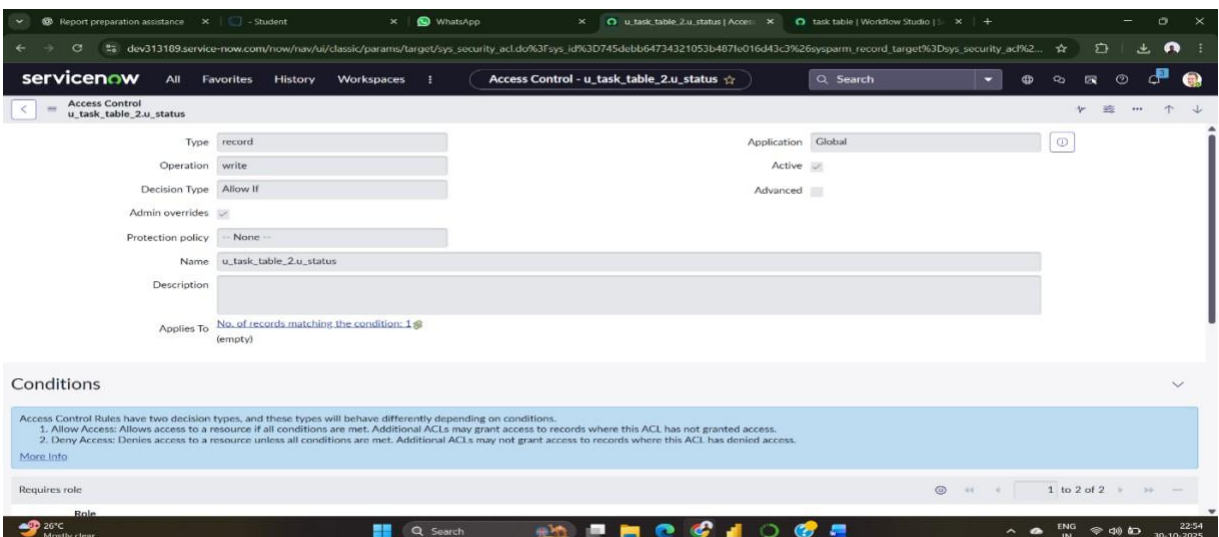
The screenshot shows the ServiceNow Application Menu configuration page for 'project table'. The page includes fields for Title, Application, Active status, Roles, Category, Hint, and Description. The 'Roles' field is set to 'project member'.

Fields and values:

- Title: project table
- Application: Global
- Active: ☒
- Roles: project member
- Category: Custom Applications
- Hint: (empty)
- Description: (empty)

Buttons: Update, Delete. Modules: Order, Search.

7. Assign Access Control



The screenshot shows the ServiceNow Access Control configuration page for 'u_task_table_2.u_status'. The page includes fields for Type, Operation, Decision Type, Admin overrides, Protection policy, Name, Description, and Applies To. The 'Applies To' field is set to 'No. of records matching the condition: 1'.

Fields and values:

- Type: record
- Operation: write
- Decision Type: Allow If
- Admin overrides: ☒
- Protection policy: -- None --
- Name: u_task_table_2.u_status
- Description: (empty)
- Applies To: No. of records matching the condition: 1 (empty)

Buttons: Update, Delete. Modules: Order, Search. Conditions: 1 to 2 of 2.

Report preparation assistance x Student x WhatsApp x Access Controls | ServiceNow x task table | Workflow Studio | x +

dev313189.service-now.com/now/nav/ui/classic/params/target/sys_security_acl_list.do

servicenow All Favorites History Workspaces Admin Access Controls Updated Search Actions on selected rows...

| | Name | Decision Type | Operation | Type | Active | Updated by | Updated |
|-------------------------------------|---------------------------------------|---------------|-------------|--------|--------|------------|---------------------|
| <input type="checkbox"/> | u_task_table_2.u_task_name | Allow If | write | record | true | admin | 2025-10-29 01:11:04 |
| <input type="checkbox"/> | u_task_table_2.u_task_id | Allow If | write | record | true | admin | 2025-10-29 01:08:53 |
| <input type="checkbox"/> | u_task_table_2.u_assigned_to | Allow If | write | record | true | admin | 2025-10-29 01:06:42 |
| <input type="checkbox"/> | u_task_table_2.u_due_date | Allow If | write | record | true | admin | 2025-10-29 01:04:43 |
| <input type="checkbox"/> | u_task_table_2.u_status | Allow If | write | record | true | admin | 2025-10-28 23:13:59 |
| <input checked="" type="checkbox"/> | u_task_table_2 | Allow If | read | record | true | admin | 2025-10-28 21:46:53 |
| <input type="checkbox"/> | u_task_table_2 | Allow If | delete | record | true | admin | 2025-10-28 21:46:53 |
| <input type="checkbox"/> | u_task_table_2 | Allow If | write | record | true | admin | 2025-10-28 21:46:53 |
| <input type="checkbox"/> | u_task_table_2 | Allow If | create | record | true | admin | 2025-10-28 21:46:53 |
| <input type="checkbox"/> | u_project_tableau | Allow If | read | record | true | admin | 2025-10-28 21:43:18 |
| <input type="checkbox"/> | u_project_tableau | Allow If | delete | record | true | admin | 2025-10-28 21:43:18 |
| <input type="checkbox"/> | u_project_tableau | Allow If | create | record | true | admin | 2025-10-28 21:43:18 |
| <input type="checkbox"/> | u_project_tableau | Allow If | write | record | true | admin | 2025-10-28 21:43:18 |
| <input type="checkbox"/> | sn_try_build_agent_knowledge_source | Deny Unless | report_view | record | true | system | 2025-10-23 18:16:19 |
| <input type="checkbox"/> | sn_try_build_agent_freemium_usage_log | Allow If | write | record | true | system | 2025-10-23 18:16:19 |
| <input type="checkbox"/> | sn_try_build_agent_task_telemetry | Deny Unless | report_view | record | true | system | 2025-10-23 18:16:19 |
| <input type="checkbox"/> | sn_try_build_agent_conversation | Allow If | read | record | true | system | 2025-10-23 18:16:19 |

https://dev313189.service-now.com/sys_security_acl.do?sys_id=22aa17b247b0321053b487e0164326&sysparm_record_target=sys_security_acl&sysparm_record_row=68&sysparm_record_rows=44&sysparm_record_list=ORDERBYDESCsys_updated_on

26°C Mostly clear Search 22:56 30-10-2025

8. Create Flow

Workflow Studio task table Active

task table 2 Created where (status is in progress, and comments is feedback, and assigned to is bob)

Trigger: Created

Table: task table 2 [u_task_table_2]

Condition: All of these conditions must be met

status is in progress

AND

comments is feedback

OR

assigned to is bob

New Criteria

Advanced Options

Delete Cancel Done

Status Modified Application: Global

Data Collapse All

Trigger - Record Created

- task table 2 Record
- task table 2 Table
- Run Start Time UTC
- Run Start Date/Time

1 - Update Record

- u_task_table_2 Record
- u_task_table_2 Table
- Action Status

2 - Ask For Approval

- Approval State

Activate Windows GO TO SETTINGS to activate Windows.

The screenshot displays the ServiceNow 'Approvals' page. At the top, there's a navigation bar with 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and an 'Approvals' button. A search bar is also present. Below the navigation bar, there's a filter section with 'Approvals' selected and a 'Created' dropdown. The main area shows a table of approval requests with columns for 'State', 'Approver', 'Comments', 'Approval for', and 'Created'. The table contains 15 rows of data, with the first row being 'Approved' and the others 'Requested'. The 'Approval for' column contains various IDs, some of which are '(empty)'. At the bottom, there's a pagination bar showing '1 to 20 of 664' and a 'Go to Settings to activate Windows' message.

| State | Approver | Comments | Approval for | Created |
|-----------|-------------------|----------|--------------|---------------------|
| Approved | alice p | | (empty) | 2024-10-22 22:26:19 |
| Rejected | Fred Luddy | | (empty) | 2024-09-01 12:19:33 |
| Requested | Fred Luddy | | (empty) | 2024-09-01 12:17:03 |
| Requested | Fred Luddy | | (empty) | 2024-09-01 12:15:44 |
| Requested | Howard Johnson | | CHG0000096 | 2024-09-01 06:15:29 |
| Requested | Ron Kettering | | CHG0000096 | 2024-09-01 06:15:29 |
| Requested | Luke Wilson | | CHG0000096 | 2024-09-01 06:15:29 |
| Requested | Christan Mitchell | | CHG0000096 | 2024-09-01 06:15:29 |
| Requested | Bernard Laboy | | CHG0000096 | 2024-09-01 06:15:29 |
| Requested | Howard Johnson | | CHG0000095 | 2024-09-01 06:15:25 |
| Requested | Ron Kettering | | CHG0000095 | 2024-09-01 06:15:25 |
| Requested | Luke Wilson | | CHG0000095 | 2024-09-01 06:15:25 |
| Requested | Christan Mitchell | | CHG0000095 | 2024-09-01 06:15:25 |
| Requested | Bernard Laboy | | CHG0000095 | 2024-09-01 06:15:25 |

12. Conclusion

The project successfully demonstrates how automation and RBAC principles can optimize user, group, and role management in modern organizations. By replacing manual access control with automated workflows, the system improves security, accuracy, and productivity.

The solution not only simplifies administrative operations but also establishes a strong foundation for scalable and compliant identity management. This innovation can be extended to enterprise and cloud environments, making it a valuable contribution toward smarter, more secure organizational ecosystems.

In recognition of the commitment to achieve
professional excellence



Karoliya Agin Doli R G

Has successfully satisfied the requirements for:

Generative AI in Action



Issued on: Sep 27, 2025
Issued by: IBM SkillsBuild

Verify: <https://www.credly.com/badges/c71ab79e-538e-413e-ba98-0281a616af09>



KAROLIYA AGIN DOLI R.G

servicenow

Successfully completed certification requirements for
Micro-Certification - Welcome to ServiceNow

Issued: October 8, 2025

Jayney Howson

Jayney Howson
Senior Vice President, Global Learning and Development
ServiceNow

THE
WORLD
WORKS
WITH
SERVICENOW™