

SERVICENOW PROJECT REPORT

Project Title: Asset Management Portal Team

Members:

- Satti Naga Venkata Sri Satyanarayana Reddy
- Kodavati Niharika

College Name: Aditya Engineering College

Submission Date: July 02, 2025

GitHub Repository: <https://github.com/satyanarayanareddysatti/Asset-Management-Portal/tree/main>

1. Project Overview

The 'Asset Management Portal' is a custom-built ServiceNow application designed to track, manage, and automate the lifecycle of both physical and digital assets within an organization. It includes key features such as asset registration, status updates, automated warranty alerts, and administrative controls for damage and loss reporting.

2. Features Implemented

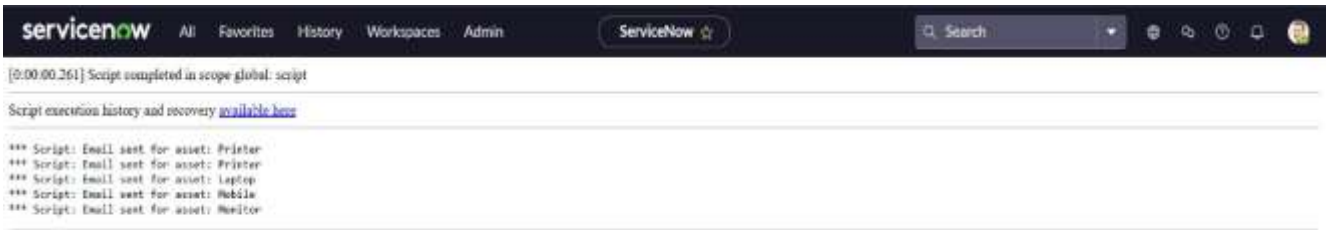
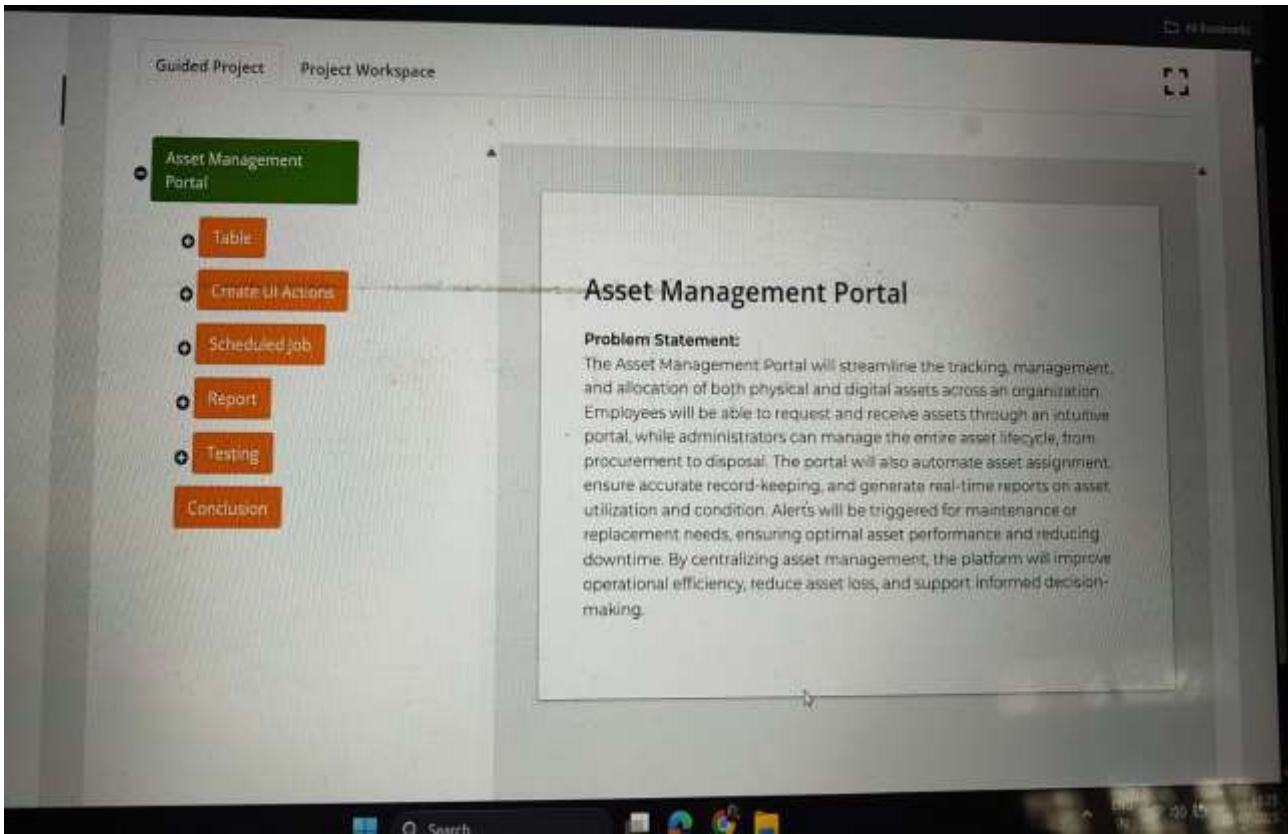
- Asset creation with fields like Asset Name, Type, Purchase Date, Warranty Expiry, etc.
- Choice fields for 'Type' and 'Status' (Available, Damaged, Lost).
- UI Action buttons - 'Mark As Damaged' and 'Mark As Lost'.
- Automated background script to trigger email alerts before warranty expiry.
- Log tracking of script execution to ensure notifications are sent.

3.Script Automation

We wrote a GlideRecord script in the 'Scripts - Background' section of ServiceNow that checks all assets expiring within 30 days and automatically sends email alerts to IT support. Sample log outputs show successful email sends for each asset.

4.Conclusion

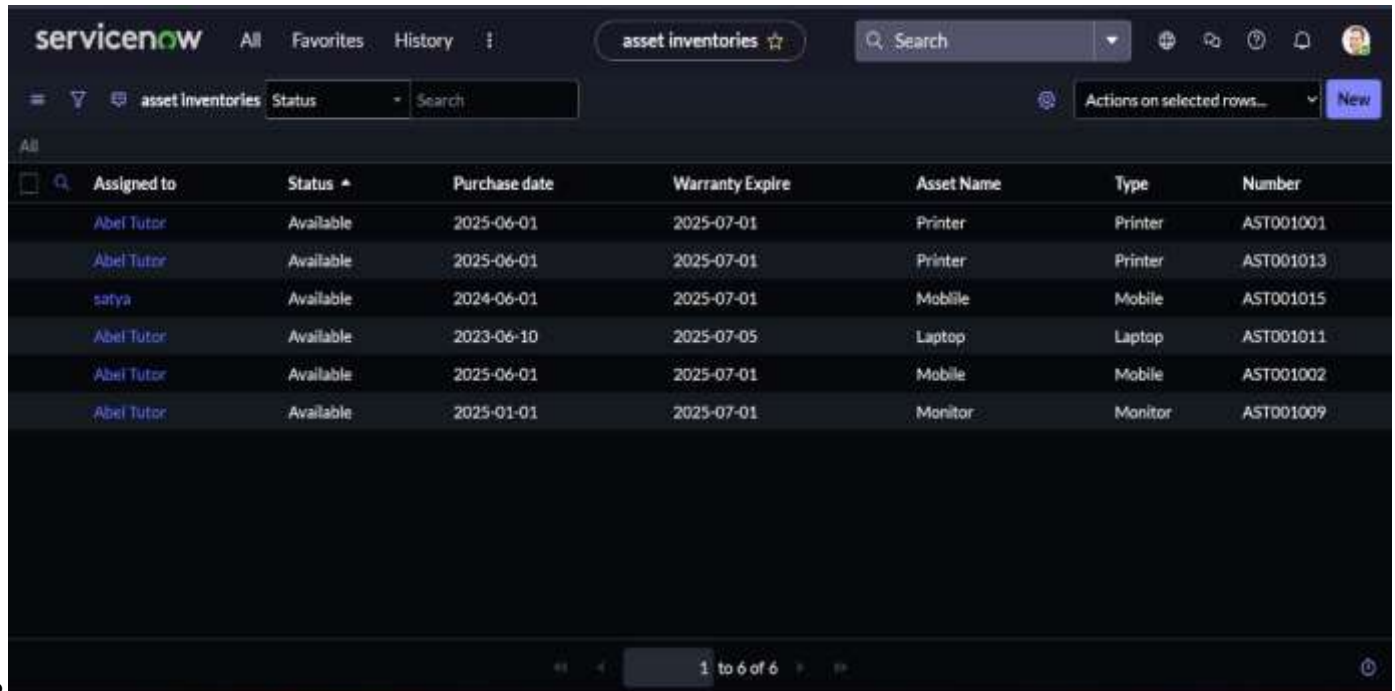
This project showcases practical knowledge of ServiceNow's capabilities, including form creation, Glide scripting, email, notification handling, and GitHub-based project documentation. We successfully designed a mini-ITSM module and demonstrated its use with data and automation.



Modules in Detail

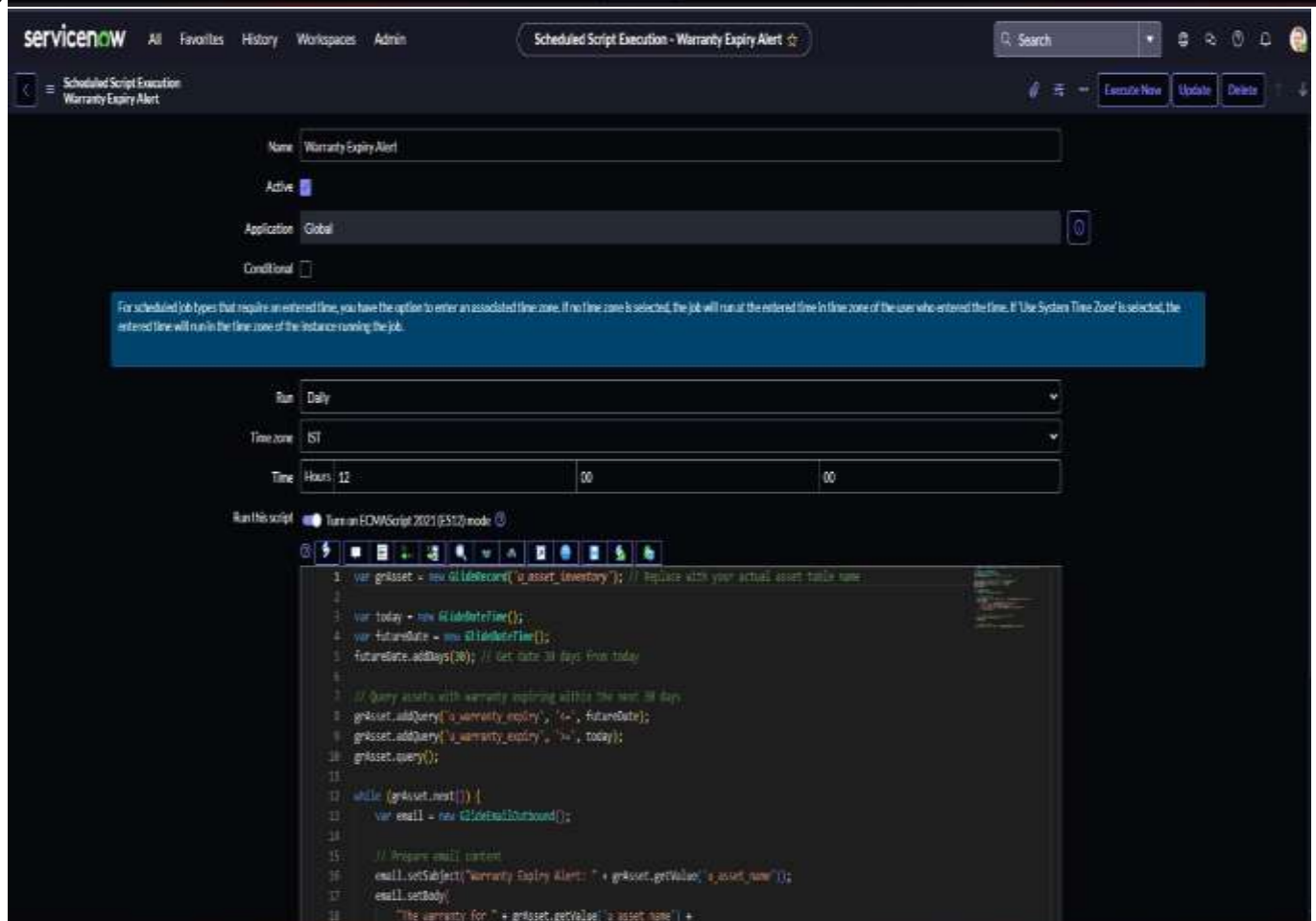
The Asset Management Portal consists of several key modules, each built using the ServiceNow platform

- Asset Registration Module: Captures all required information related to a new asset, such as asset name
- Status Management Module: Allows tracking and updating the condition of each asset (Available, In Use)
- Warranty Tracking Module: Automatically monitors asset warranty status and triggers alerts before expiry
- Incident Reporting Module: Enables users to report asset-related incidents like loss or damage.
- Admin Dashboard: Provides a consolidated view for administrator with asset statistics, incident logs.



The screenshot shows the ServiceNow interface for the 'asset inventories' table. The table lists assets with columns for Assigned to, Status, Purchase date, Warranty Expire, Asset Name, Type, and Number. The data is as follows:

Assigned to	Status	Purchase date	Warranty Expire	Asset Name	Type	Number
Abel Tutor	Available	2025-06-01	2025-07-01	Printer	Printer	AST001001
Abel Tutor	Available	2025-06-01	2025-07-01	Printer	Printer	AST001013
satya	Available	2024-06-01	2025-07-01	Mobile	Mobile	AST001015
Abel Tutor	Available	2023-06-10	2025-07-05	Laptop	Laptop	AST001011
Abel Tutor	Available	2025-06-01	2025-07-01	Mobile	Mobile	AST001002
Abel Tutor	Available	2025-01-01	2025-07-01	Monitor	Monitor	AST001009



The screenshot shows the 'Scheduled Script Execution - Warranty Expiry Alert' form in ServiceNow. The form includes fields for Name, Active status, Application, and Conditional. Below these fields is a blue informational box. The 'Run' dropdown is set to 'Daily', and the 'Time zone' is set to 'IST'. The 'Time' field is set to 'Hours: 12:00:00'. The 'Run this script' section shows a script for querying assets with warranty expiring within the next 30 days and sending an email alert.

For scheduled job types that require an entered time, you have the option to enter an associated time zone. If no time zone is selected, the job will run at the entered time in time zone of the user who entered the time. If 'Use System Time Zone' is selected, the entered time will run in the time zone of the instance running the job.

Run: Daily
Time zone: IST
Time: Hours: 12:00:00

Run this script: ☒ Turn on ECMAScript 2021 (ES12) mode

```
1 var grAsset = new GlideRecord('a_asset_inventory'); // Replace with your actual asset table name
2
3 var today = new GlideDateTime();
4 var futureDate = new GlideDateTime();
5 futureDate.addDays(30); // Set date 30 days from today
6
7 // Query assets with warranty expiring within the next 30 days
8 grAsset.addQuery('a_warranty_expiry', '<=', futureDate);
9 grAsset.addQuery('a_warranty_expiry', '>=', today);
10 grAsset.query();
11
12 while (grAsset.next()) {
13     var email = new GlideEmailOutbound();
14
15     // Prepare email content
16     email.setSubject('Warranty Expiry Alert: ' + grAsset.getValue('a_asset_name'));
17     email.setBody(
18         'The warranty for ' + grAsset.getValue('a_asset_name') +
```

Table

asset inventory View: Tables

🔍

🔍

🔍

Delete

Update

Delete All Records

* Label

asset inventory

* Name

a_asset_inventory

Application

Global

Remote Table

Columns

Controls

Application Access

Table Columns

Type

Search

1 to 13 of 13

New

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
Type	Choice	(empty)	40		false
Status	Choice	(empty)	40		false
Purchase date	Date	(empty)	40		false
Warranty Expiry	Date	(empty)	40		false
Created	DateTime	(empty)	40		false
Updated	DateTime	(empty)	40		false
Updates	Integer	(empty)	40		false
Created by	String	(empty)	40		false
Asset Name	String	(empty)	140		false
Assigned to	String	(empty)	40		false
Updated by	String	(empty)	40		false
Number	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Insert a new row...					

- Security and Access Control: Implemented using roles. Only authorized users (admins) can update it.

The screenshot shows the ServiceNow interface for editing an asset record. The header includes the ServiceNow logo, navigation links (All, Favorites, History), a search bar, and a user profile icon. The breadcrumb trail shows 'asset inventory - AST001011'. The form fields are as follows:

Field	Value
Number	AST001011
Asset Name	Laptop
Purchase date (yyyy-MM-dd)	2023-06-10
Warranty Expire (yyyy-MM-dd)	2025-07-05
Assigned to	Abel Tutor
Status	Available
Type	Laptop

At the top right of the form, there are buttons: 'Update', 'Mark As Damaged', 'Mark As Lost', and 'Delete'. At the bottom left, there are buttons: 'Update', 'Mark As Damaged', 'Mark As Lost', and 'Delete'. A small 'i' icon is visible in the bottom right corner of the form area.

5. Screenshots of Implementation

Below are a few implementation screenshots (you can insert your own here in the actual PDF):

1. Asset Creation Form – Interface showing dropdowns for status and type, and fields for warranty dates.
2. Incident Reporting UI Action – Custom button to report damage/loss.
3. Script Execution Log – Background script logs showing successful email sends.
4. Admin Dashboard – Summary view with search and filter capabilities.

The screenshot shows the 'Scheduled Script Execution - Warranty Expiry Alert' configuration page in ServiceNow. The page includes a search bar, navigation tabs (All, Favorites, History, Workspaces, Admin), and a toolbar with 'Execute Now', 'Update', and 'Delete' buttons. The configuration fields are as follows:

- Name: Warranty Expiry Alert
- Active: ☒
- Application: Global
- Conditional: ☐

A blue informational box states: "For scheduled job types that require an entered time, you have the option to enter an associated time zone. If no time zone is selected, the job will run at the entered time in time zone of the user who entered the time. If 'Use System Time Zone' is selected, the entered time will run in the time zone of the instance running the job."

Run: Daily
Time zone: IST
Time: Hours: 12, Minutes: 00, Seconds: 00

Run this script: ☒ Run an ECMAScript 2021 (ES12) mode

```
1 var grsset = new GlideRecord('u_asset_inventory'); // replace with your actual asset table name
2
3 var today = new GlideDateTime();
4 var futureDate = new GlideDateTime();
5 futureDate.addDays(30); // get date 30 days from today
6
7 // Query assets with warranty expiring within the next 30 days
8 grsset.addQuery('u_warranty_expiry', '<=', futureDate);
9 grsset.addQuery('u_warranty_expiry', '>=', today);
10 grsset.query();
11
12 while (grsset.next()) {
13     var email = new GlideMailOutbound();
14
15     // Prepare email content
16     email.setSubject('Warranty Expiry Alert: ' + grsset.getValue('u_asset_name'));
17     email.setBody(
18         'The warranty for ' + grsset.getValue('u_asset_name') +
```

The screenshot shows the 'Table: asset inventory View Tables' page in ServiceNow. The page includes a search bar, navigation tabs (Columns, Controls, Application Access), and a toolbar with 'Delete', 'Update', and 'Delete All Records' buttons. The configuration fields are as follows:

- * Label: asset inventory
- * Name: u_asset_inventory
- Application: Global
- Remote Table: ☐

Table Columns: Type Search

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
Type	Choice	(empty)	40	false	
Status	Choice	(empty)	40	false	
Purchase date	Date	(empty)	40	false	
Warranty Expiry	Date	(empty)	40	false	
Created	DateTime	(empty)	40	false	
Updated	DateTime	(empty)	40	false	
Updates	Integer	(empty)	40	false	
Created by	String	(empty)	40	false	
Asset Name	String	(empty)	140	false	
Assigned to	String	(empty)	40	false	
Updated by	String	(empty)	40	false	
Number	String	(empty)	40	false	
Seq ID	Seq ID GUID	(empty)	32	false	

Insert a new row...

6. Challenges Faced and Solutions

- Challenge: Automating email notifications using Glide scripts.
- Solution: Carefully structured the GlideRecord loop and used `gs.eventQueue()` for email events. Verified
- Challenge: Managing asset status consistency across modules.
- Solution: Used Business Rules and Client Scripts to auto-update related fields and maintain data integrity
- Challenge: UI Actions were not visible to all roles.
- Solution: Defined conditions under "UI Action

The screenshot shows the ServiceNow interface for an Asset Inventory record. The record number is AST001011. The asset name is Laptop. The purchase date is 2023-06-10, and the warranty expires on 2025-07-05. The status is Available, and the type is Laptop. The asset is assigned to Abel Tutor. At the bottom, there are buttons for Update, Mark As Damaged, Mark As Lost, and Delete.

```
1 // Script to send alert emails for assets with warranty expiring in the next 30 days
2
3 var grinst = new GlideRecord('asset_inventory'); // Replace with your table name if different
4
5 var today = new GlideDate();
6 var futureDate = new GlideDate();
7 futureDate.addDays(30); // 30 days from today
8
9 // Query assets with warranty expiring in the next 30 days
10 grinst.addQuery('warranty_expire', '>=', futureDate);
11 grinst.addQuery('warranty_expire', '<=', today);
12 grinst.query();
13
14 while (grinst.next()) {
15     var email = new GlideEmail();
16
17     // Email subject
18     email.setSubject('Warranty Expiry Alert: ' + grinst.getValue('asset_name'));
19
20     // Email body
21     email.setBody(
22         'The warranty for ' + grinst.getValue('asset_name') +
23         ' (ID: ' + grinst.getValue('asset_id') + ') is ' +
24         'expiring on ' + grinst.getValue('warranty_expire') + '. ' +
25         'Please take necessary action.'
26     );
27
28     // Recipient email (replace with real IT support address)
29     email.setTo('it-support@company.com');
30
31     // Send email
32     email.send();
33
34     // Log confirmation
35     gs.info('Email sent for asset: ' + grinst.getValue('asset_name'));
36 }
37
```


7. Future Enhancements

- Integration with CMDB: Extend the portal to link with Configuration Management Database for deeper I
- Asset History Tracking: Add functionality to track asset movement and changes over time.
- QR Code Integration: Generate QR codes for physical assets for quick scanning and tracking.
- Mobile View: Improve UI responsiveness for mobile devices.
- Reporting & Analytics: Build dashboards to visualize asset usage, damage trends, and budget impact. These enhancements will further increase the value and usability of the Asset Management Portal.

