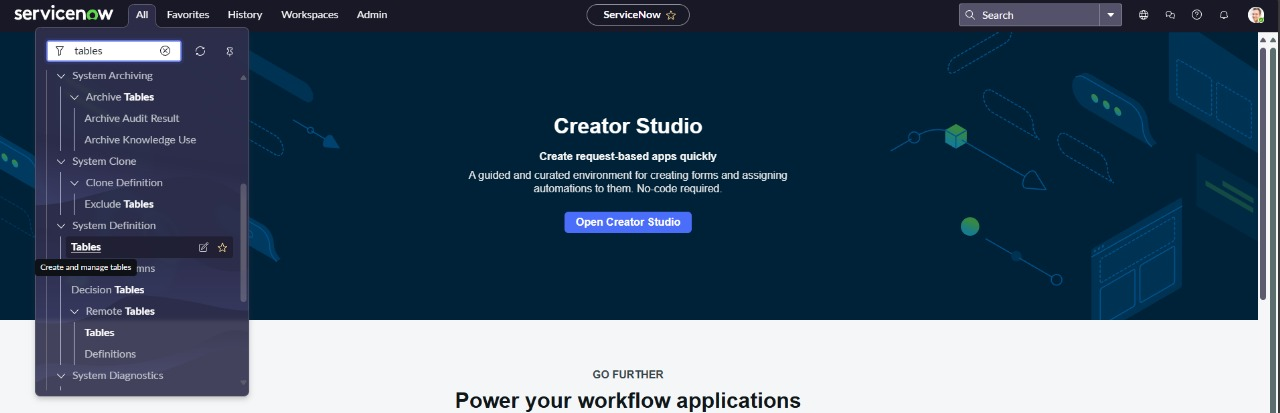
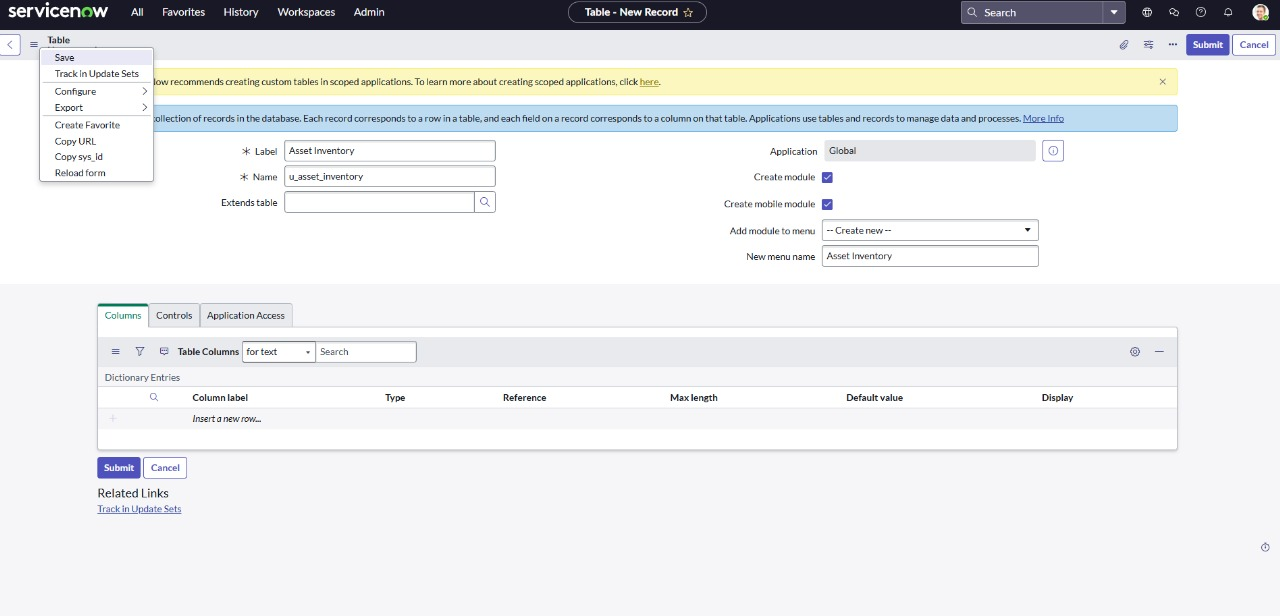
# **Asset Management Portal**

**Milestone 1: Setting Up ServiceNow Instance:**

* Sign up on [developer.servicenow.com](https://developer.servicenow.com) and create a developer account.
* Navigate to the Personal Developer Instance section and request a new instance.
* Fill out the required information and submit the request.
* Instance details (URL, username, password) will be sent via email.
* Log in to the instance using the provided credentials.
* The instance is ready for development and customization.

**Milestone 2: Creation of Table (Asset Inventory):**

* + Navigate to All in the application navigator.
  + In the filter search bar, type Tables and select it from the results.
  + 
  + Click on the New button to create a new table.
  + 
  + In the form, provide the following details:
    - Label: Asset Inventory
    - Name: Auto-Populated(automatically generated based on the label)
    - New menu name: Asset Management Portal
    - 
  + Navigate to the header section of the form, right-click on the header, and select Save from the context menu.

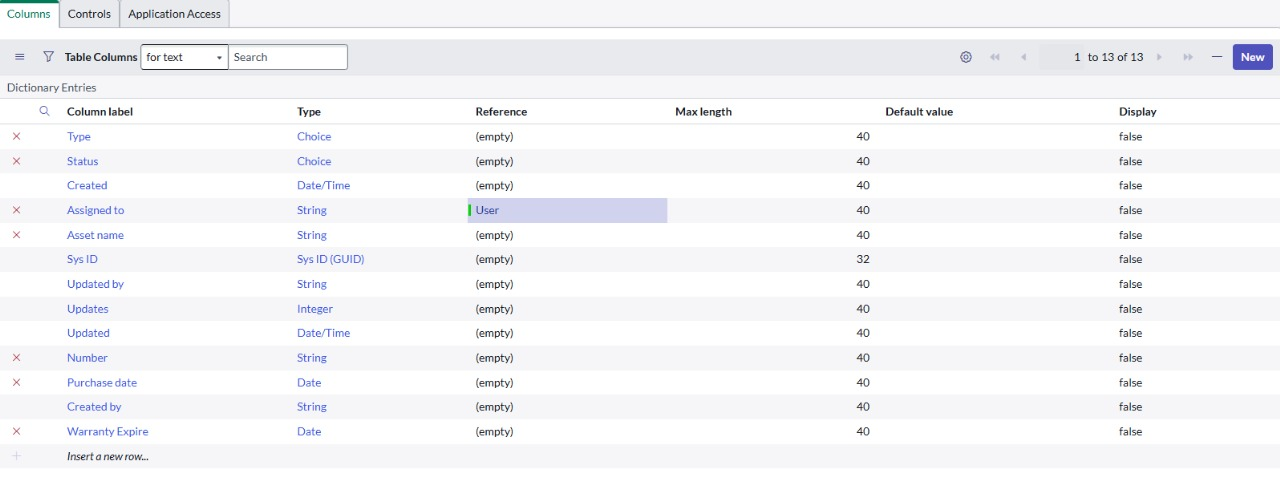
**Milestone 3: Creation of Fields:**

1)After saving the table scroll down

2)Create the following fields

* Assigned to : string
* Status : choice
* Purchase date : date
* Warranty Expire : date
* Asset name : string
* Type : choice
* Number : String

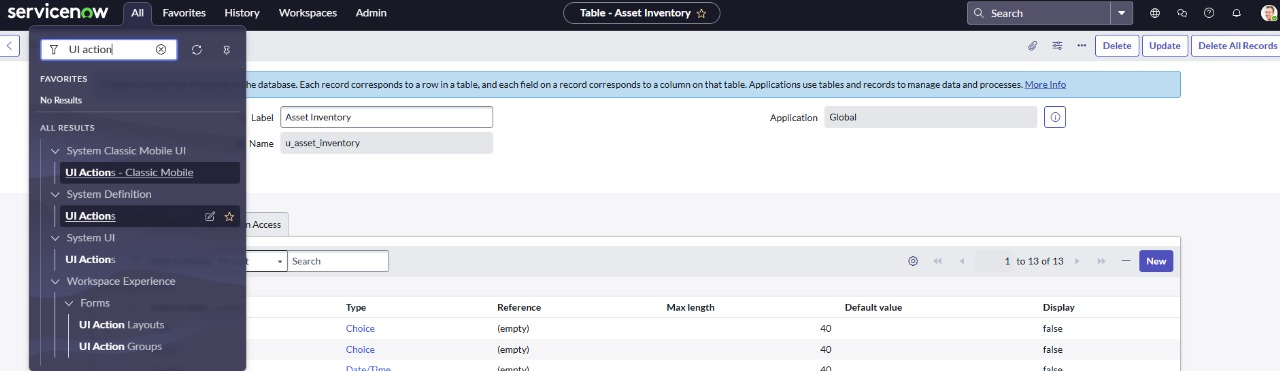
3) Click on save



.

**Milestone 4: Creation of UI Actions:**

* Navigate to UI Actions under System Definition:

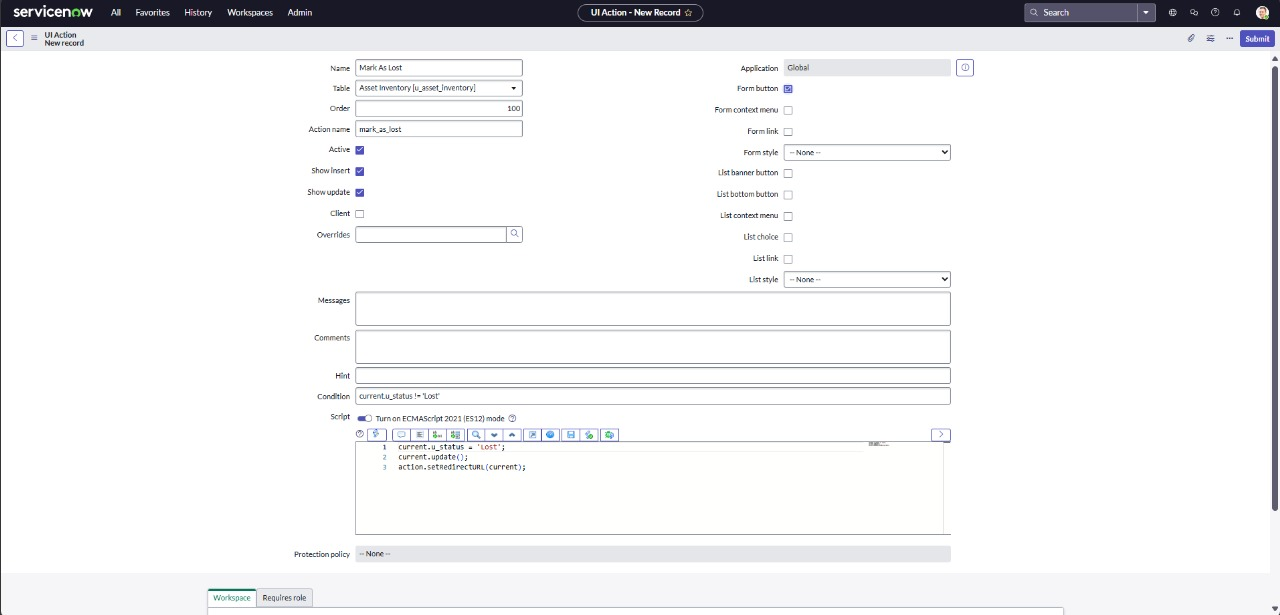


* + Click on new to create a new UI Action
  + Fill in the UI Action Details ;
  + Name: Mark as Lost
  + Table: Asset Inventory
  + Action name: mark\_as\_lost
  + Condition: current.u\_status !=’Lost’
  + Update the script as
    - current.u\_status = ‘Lost’;

current.update();

action.setRedirectURL(current);

* + Check the box Form Button to make it visible on the form layout.
  + Click Save to apply the changes.

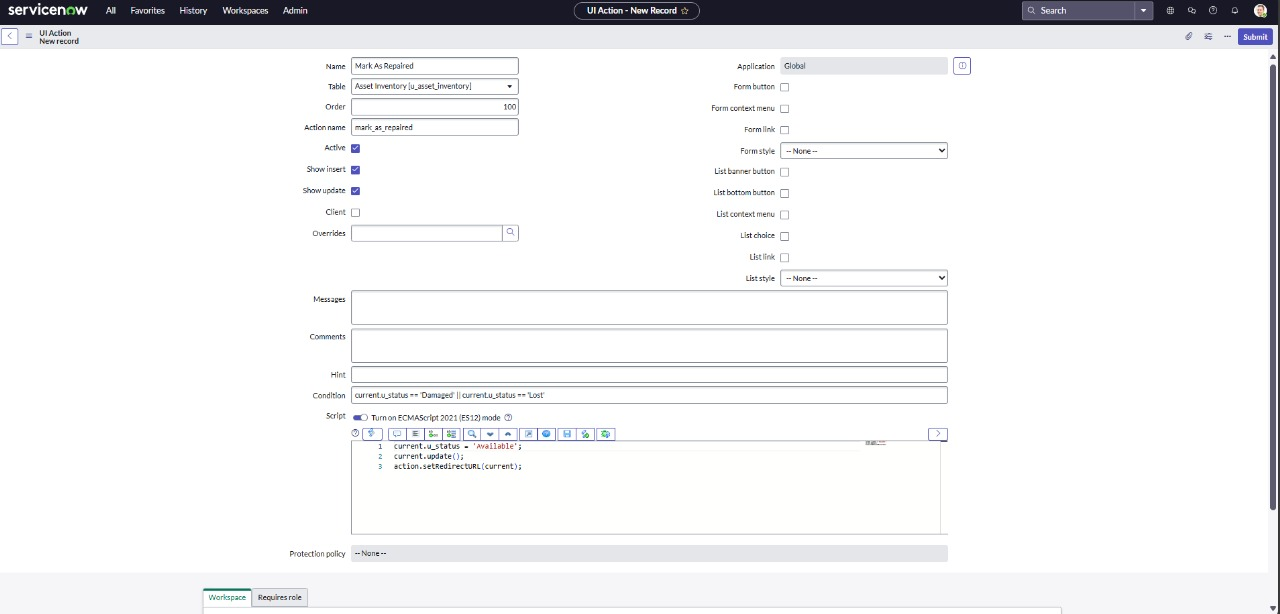


* Navigate to Application Navigator
  + Click on UI Actions under System Definition.
  + Click on New(top-right corner) to create new UI Action
  + Fill in the following details:
    - Name: Mark as Repaired
    - Table: Asset Inventory
    - Action name: mark\_as\_repaired
    - Condition: current.u\_status==’Damaged’
  + Add the Script as:
    - current.u\_status = ‘Available’;

current.update( );

action.setRedirectURL(current);

* + This Script updates the u\_status field to Available saves the record, and then redirects the user back to the current record view.
  + Scroll down to the Form button section.
  + Check the box labeled Form button to ensure this action appears as a button on the form view.
  + Click on Submit to create the UI Action.



**Milestone 5: Creation of Scheduled Job:**

* + Navigate to All in the application navigator.
  + In the filter search bar, type Scheduled Job and select it from the list.
  + Click on the New button to create a new scheduled job.
  + Write the following script:
  + Enter the following details:
    - Name: Warranty Expiry Alert
    - Run : Daily
    - Time : 12:00

Write the following script:

var grAsset = new GlideRecord('u\_asset\_inventory'); // Replace with your table name

var today = new GlideDateTime();

var futureDate = new GlideDateTime();

futureDate.addDays(30); // Get date 30 days from now

grAsset.addQuery('u\_warranty\_expire', '<=', futureDate); // Warranty expiring within the next 30 days

grAsset.addQuery('u\_warranty\_expire', '>=', today); // Warranty expiring after today

grAsset.query();

while (grAsset.next()) {

var email = new GlideEmailOutbound();

email.setSubject("Warranty Expiry Alert: " + grAsset.getValue('u\_assest\_name')); // Use getValue for dynamic field access

email.setBody("The warranty for " + grAsset.getValue('u\_assest\_name') + " (Type: " + grAsset.getValue('u\_asset\_type') +

") is expiring soon on " + grAsset.getValue('u\_warranty\_expiry') + ". Please take action."); // Get values dynamically

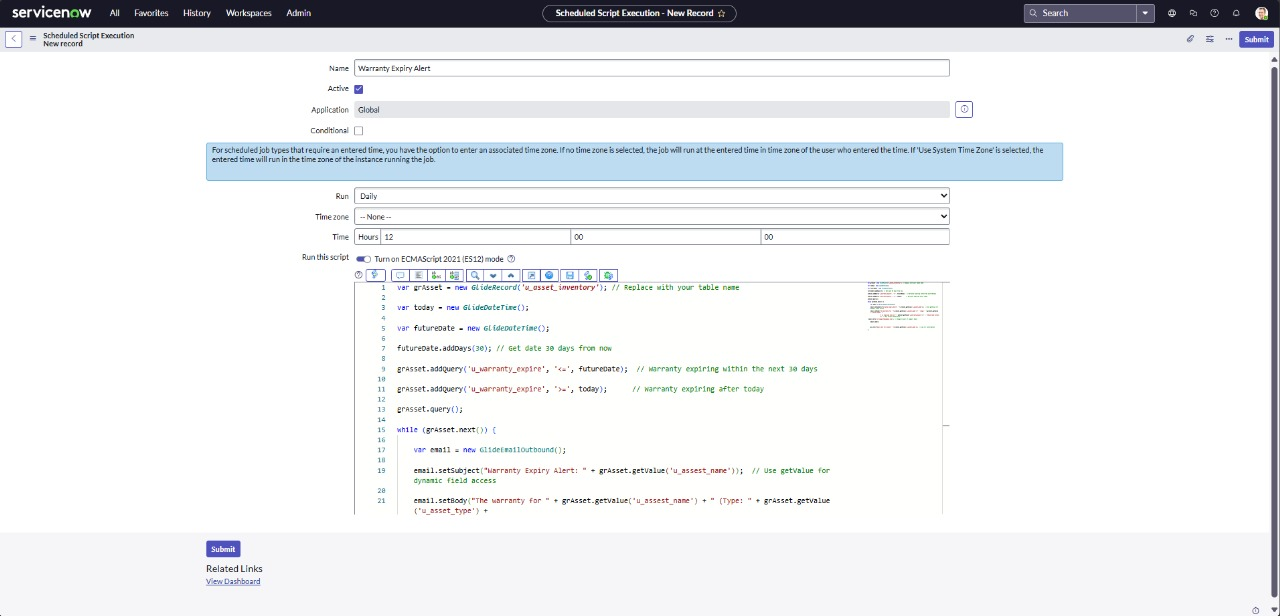
email.setTo('it-support@company.com'); // Change to your IT support email

email.send();

gs.info("Email sent for assest: " + grAsset.getValue('u\_assest\_name')); // Log for confirmation

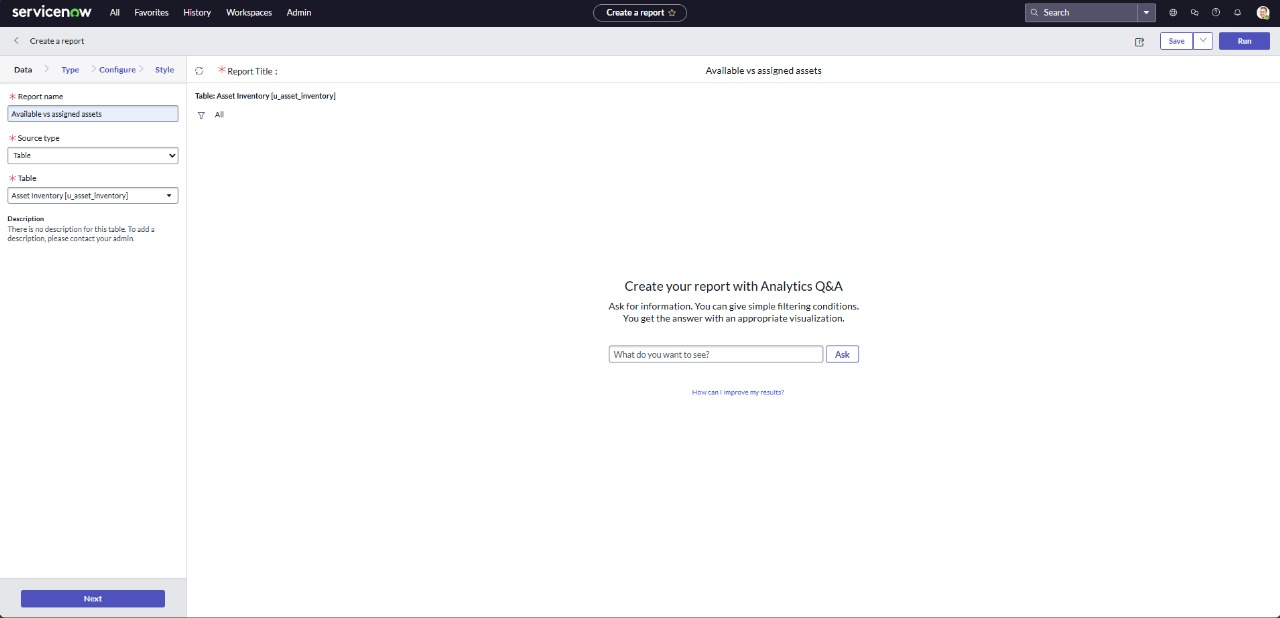
}

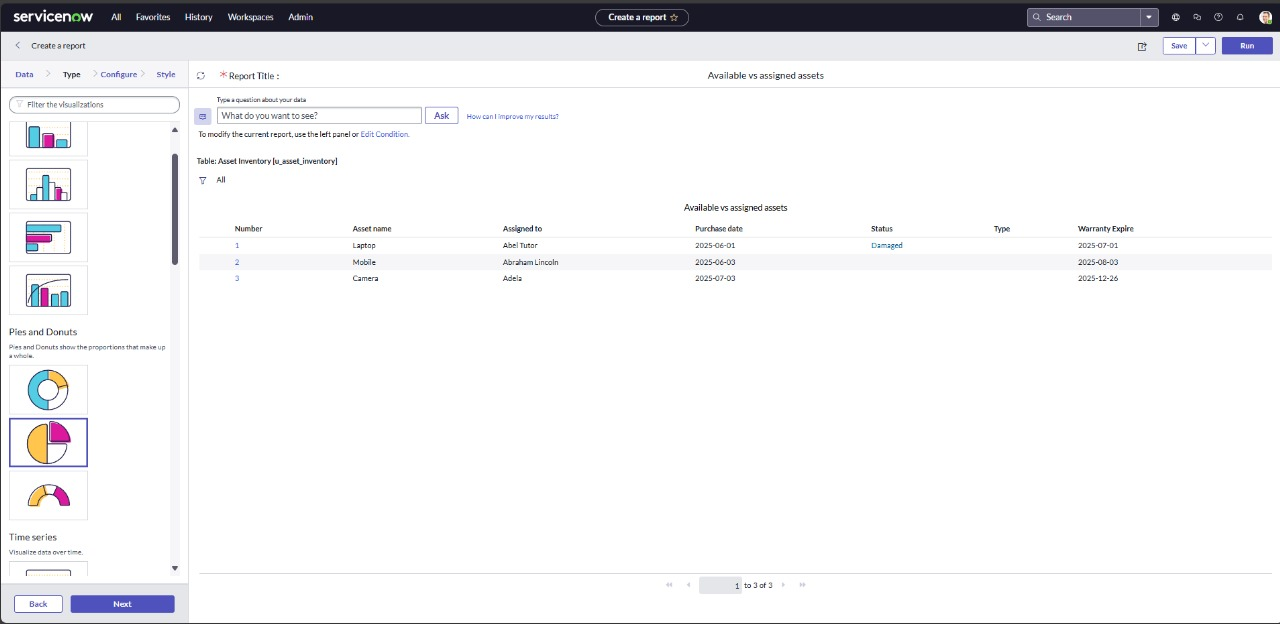
* Click Save to store the scheduled job.

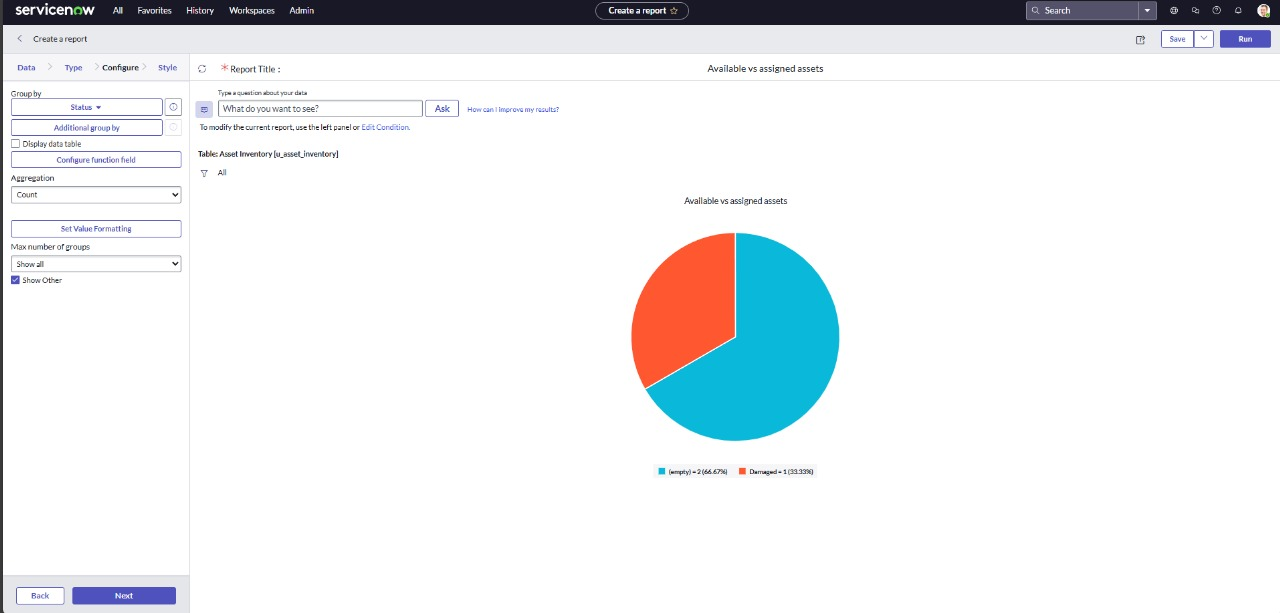


**Milestone 6: Creation of Report :**

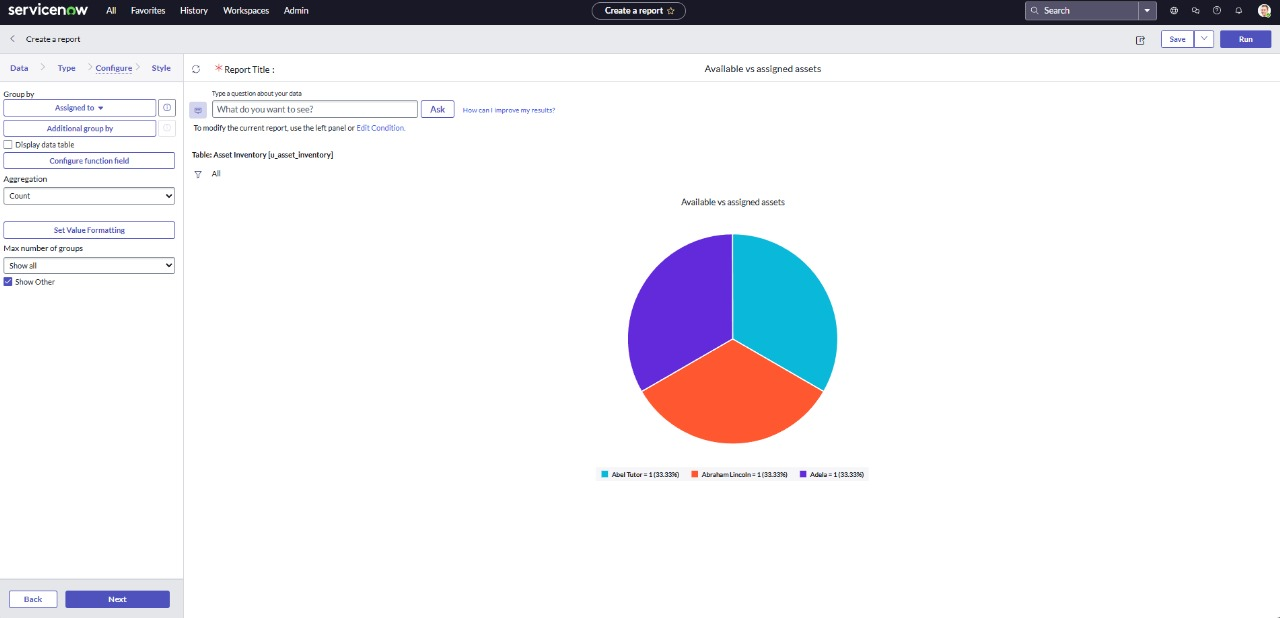
* + Navigate to All in the main menu.
  + In the filter/search bar, type Reports, then select and open the Reports.
  + Click on the New button to create a new Report.
  + Enter the following details:
    - Report Name:Available vs assigned assets
    - Source Type: Table
    - Table: Asset Inventory
    - Type: Pie chart
    - Group By: Status
    - Aggregation: Count.





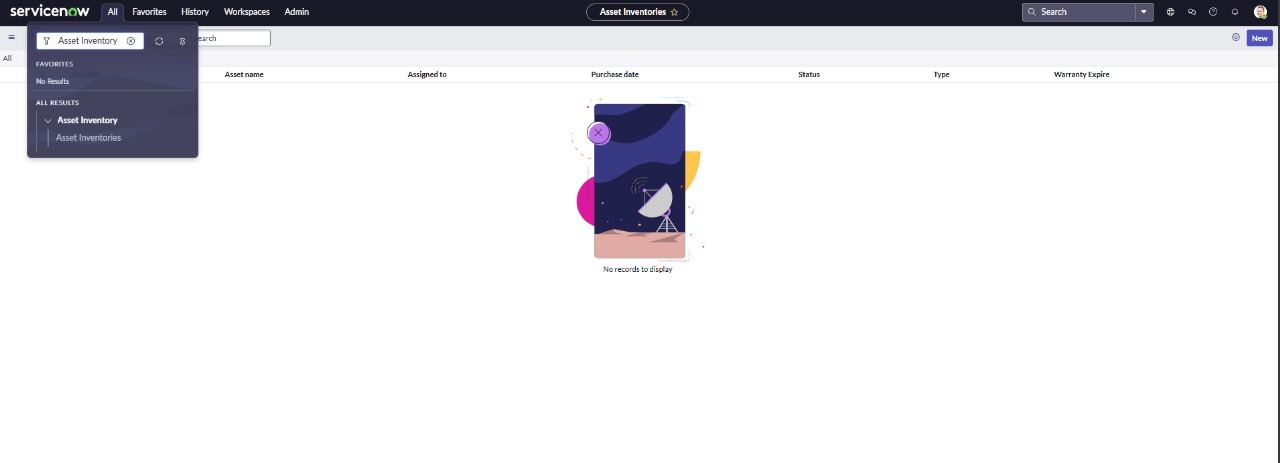


* + Click Save or Done (depending on the interface) to apply the changes.
  + Next click on run.

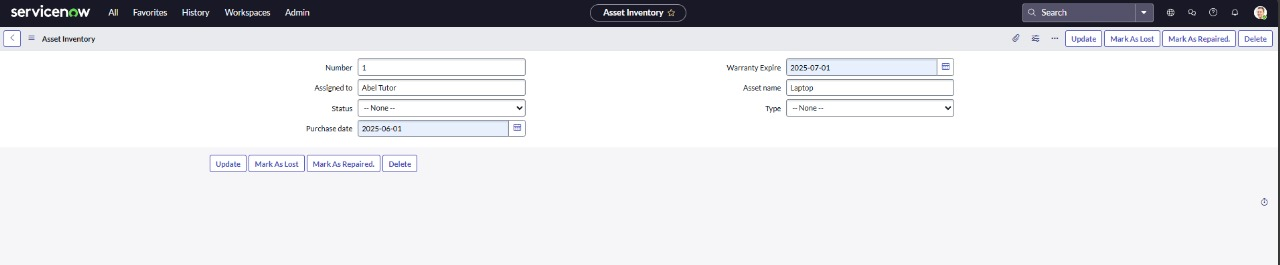


**Milestone 7: Testing of UI Action:**

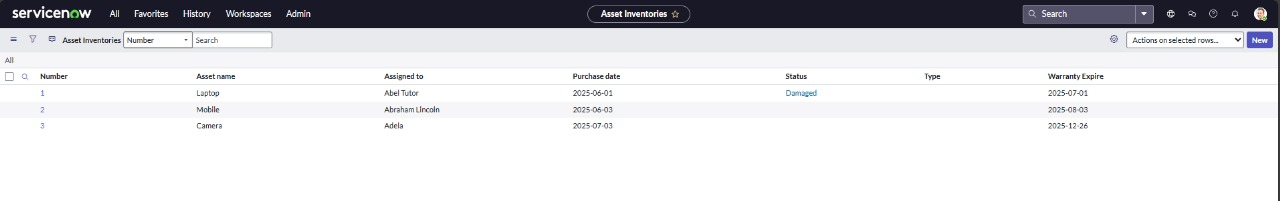
* + From the main navigation, go to All.
  + In the search or filter bar, type Asset Inventory table.



* + Click on the New button to test UI Action.
  + Fill in theAsset inventory table record Details:
    - Asset Name:Laptop
    - Type: laptop
    - Assigned to: Abel Tutor
    - Status: Available
    - Purchase date: 2025-06-01
    - Expiry date: 2025-07-01

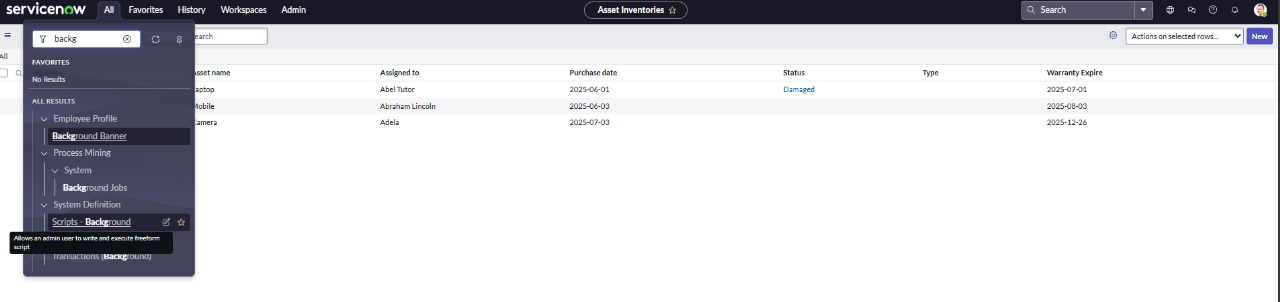


* + Click on Submit.
  + Open the record once again and click on the mark as lost button .
  + Save the record.
  + Check the status is changed to lost.

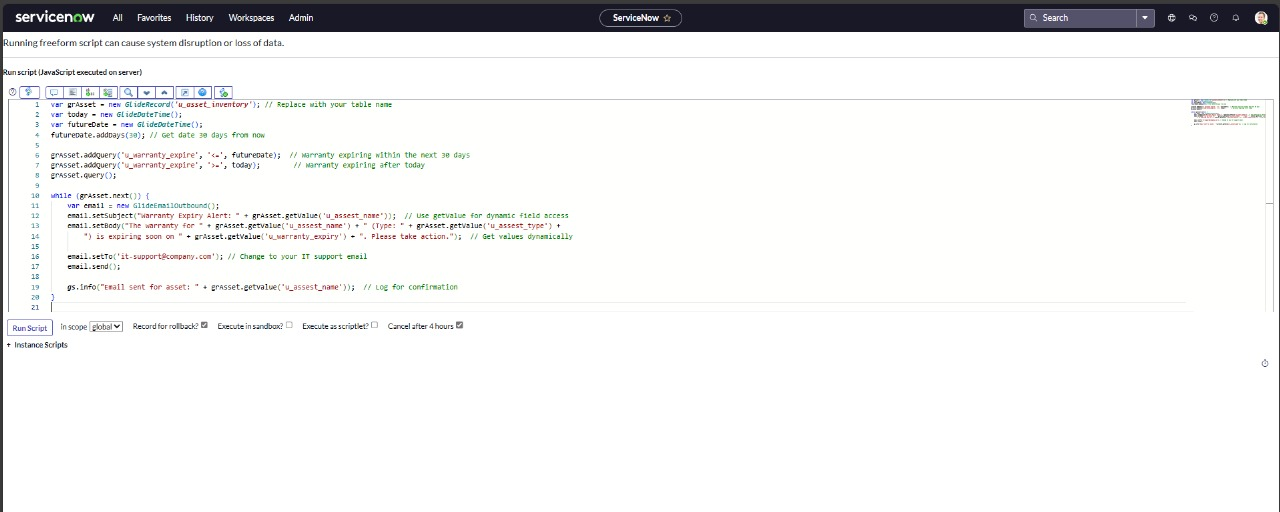


**Milestone 8: Testing of Scheduled Job:**

* From the main navigation, go to All.
* In the search bar, type background scripts, then open the Scheduled Job script in the background scripts.



* Click on run script button to run the code.



* **A**fter running the script, check the result**.**

