

**PROJECT TITLE**

**WORKFORCE ADMINISTRATION SOLUTION (DEV)**

College Name: **MAILAM ENGINEERING COLLEGE**

Code: **4216**

**Department of Computer Science and Engineering**

**Team Leader:**

Name: Meera C Reg No: 421622104082

**Team Members:**

Name: Monika S Reg No: 421622104088

Name: Nisha N Reg No: 421622104098

Name: Jayasri L Reg No: 421622104058

Registered Email Id: meerachandrasekaran6@gmail.com

TrailheadUrl : <https://trailhead.salesforce.com/users/smartbridgesupport/trailmixes/trailmix-by-smartinternz-developer>



# Project Overview

The **Workforce Administration Solution** is a robust software platform designed to streamline workforce and asset management processes within an organization. Built on the Salesforce platform, it serves as a centralized system for managing employee data, monitoring project assignments, tracking performance, and maintaining detailed asset records.

The solution addresses the challenges of manual workforce management by automating key processes. It enables organizations to track the number of projects an employee is working on, evaluate their performance, and ensure accountability for assigned assets. This automation not only reduces manual effort but also enhances data accuracy and operational efficiency.

Key features include real-time data monitoring, user interface customization, and robust data security, ensuring seamless operations. The platform supports bulk data imports, role-based access controls, and collaboration through group dashboards, making it highly adaptable for diverse organizational needs.

With customizable reports and dashboards, decision-makers gain valuable insights into workforce performance and resource utilization. This facilitates informed decision-making, improved resource allocation, and enhanced productivity.

In summary, the Workforce Administration Solution is an innovative tool that transforms traditional workforce management processes into a modern, efficient, and scalable system, supporting the growth and strategic goals of organizations.

Contents

[1. Project Overview 2](#_Toc184138401)

[2. Objectives 4](#_Toc184138402)

[3. Salesforce Key Features and Concepts Utilized 4](#_Toc184138403)

[ Custom Objects and Fields: 5](#_Toc184138404)

[ Process Automation: 5](#_Toc184138405)

[ Validation Rules: 5](#_Toc184138406)

[ Reports and Dashboards: 5](#_Toc184138407)

[ Lightning Components: 6](#_Toc184138408)

[ Data Security and Sharing: 6](#_Toc184138409)

[ Approval Processes: 6](#_Toc184138410)

[ Chatter Collaboration: 6](#_Toc184138411)

[ Mobile Accessibility: 6](#_Toc184138412)

[ Integration with Third-Party Tools: 6](#_Toc184138413)

[4. Detailed Steps to Solution Design 7](#_Toc184138414)

[Step 1: Creating Developer Account and Activation 7](#_Toc184138415)

[Step 2: Create Custom Objects 8](#_Toc184138416)

[Step 3: Create Custom Tabs 9](#_Toc184138417)

[Step 4: Create a Lightning App 10](#_Toc184138418)

[Step 5: Create Fields 12](#_Toc184138419)

[Step 6: Create OWD Setting 13](#_Toc184138420)

[Step 8: Creating a Page Layout 14](#_Toc184138421)

[Step 9: Creating a Chatter Group for Your Organization 15](#_Toc184138422)

[Step 10: Creating Record Type 17](#_Toc184138423)

[Step 11: Creating a Permission Set 18](#_Toc184138424)

[Step 13: Create Report 19](#_Toc184138425)

[5. Testing and Validation: Create Dashboard, Leave Object, and Apex Trigger 20](#_Toc184138426)

[1. Testing and Validation for Dashboard Creation 20](#_Toc184138427)

[2. Testing and Validation for Leave Object Creation 21](#_Toc184138428)

[3. Testing and Validation for Apex Trigger 21](#_Toc184138429)

[4. Overall Validation Process 22](#_Toc184138430)







# Objectives

The primary objective of the **Workforce Administration Solution (Dev)** project is to develop an efficient system that enhances workforce and asset management processes within an organization. The key goals include:

* **Streamlining Workforce Management:**  
   Centralizes the storage and management of employee data, project assignments, and asset records to ensure seamless information access.
* **Automating Processes:**  
   Reduces manual tasks by automating repetitive processes, such as updating project statuses, tracking employee performance, and managing asset assignments.
* **Improving Resource Allocation:**  
   Ensures optimal utilization of employees and assets by providing real-time insights into project workloads and resource availability.
* **Tracking Performance Metrics:**  
   Monitors individual and team performance with detailed dashboards, enabling timely interventions and improvements.
* **Enhancing Data Integrity and Security:**  
   Implements role-based access and validation rules to ensure accurate and secure data management.
* **Delivering Actionable Insights:**  
   Provides robust reporting and analytics tools to support data-driven decision-making and strategic planning.
* **Increasing Efficiency and Satisfaction:**  
   Ensures efficient project execution and transparency, leading to higher productivity and employee satisfaction.

These objectives contribute to the overarching goal of creating a scalable, secure, and intuitive solution that enhances productivity, optimizes resource management, and supports organizational growth.







# Salesforce Key Features and Concepts Utilized

## Custom Objects and Fields:

* + Created custom objects to manage employee data, project assignments, and asset records.
  + Defined fields to store relevant details such as project metrics, asset specifications, and performance indicators.

## Process Automation:

* + Used Flow Builder to automate workflows like project updates and task tracking.
  + Developed Apex Triggers to handle complex processes such as performance evaluations and automated notifications.\

## Validation Rules:

* + Applied validation rules to enforce data integrity and accuracy for workforce details, project records, and asset assignments

## Reports and Dashboards:

* + Designed custom dashboards to visualize employee performance, project progress, and asset utilization.
  + Created detailed reports for analyzing resource allocation and organizational efficiency.

A blue text on a black background

Description automatically generated



A black and blue text

Description automatically generated



## Lightning Components:

* + Utilized Salesforce Lightning for a dynamic user interface.
  + Integrated drag-and-drop functionality and visual tools for process tracking, enhancing user experience.

## Data Security and Sharing:

* + Configured role-based access controls to safeguard sensitive information, ensuring secure data sharing across teams.

## Approval Processes:

* + Established workflows for approvals, such as resource allocation and project milestone confirmations, to improve efficiency.

## Chatter Collaboration:

* + Enabled real-time collaboration among team members using Salesforce Chatter for project updates and discussions.

## Mobile Accessibility:

* + C onfigured the Salesforce Mobile App to provide users with on-the-go access to workforce and project data.

## Integration with Third-Party Tools:

* + Integrated with external systems to manage inventory, accounting, and reporting seamlessly, creating a cohesive operational ecosystem

A blue text on a black background

Description automatically generated

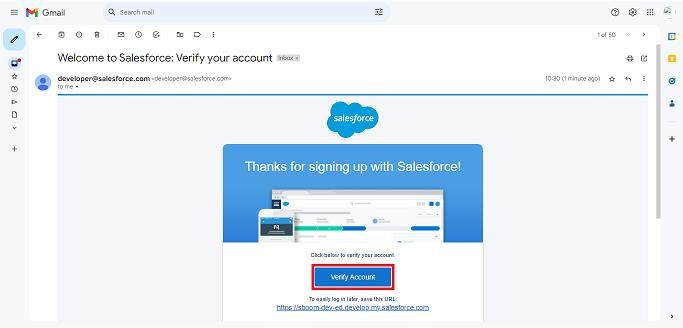
A black and blue text

Description automatically generated

# Detailed Steps to Solution Design

## Step 1: Creating Developer Account and Activation

* Creating a developer org in salesforce.
  + To Activate the account, click on the verify account. Give a password and answer a security question and click on change password.
  + Give a password and answer a security question and click on change password. Then you will redirect to your salesforce setup page.









## Step 2: Create Custom Objects

* + Navigate to **Setup > Object Manager > Create > Custom Object**.
  + Create the following objects:

1. **Employee Object**
2. **Project Object**
3. **ProjectTask**
4. **Asset**
5. **Asset Service**
   * Define labels, plural labels, record name format, and data type for each object.









## Step 3: Create Custom Tabs

* + Tabs allow users to view and create records for your objects.
  + Select the object from the list (e.g., Employee Object, Project Object, ProjectTask, Asset, or Asset Service).
  + Select an object and choose a tab style.
  + Ensure "Append tab to users' existing personal customizations" is checked.
  + Save your changes.

A screenshot of a computer

Description automatically generated







## Step 4: Create a Lightning App

* + Go to Setup > search “App Manager” in Quick Find > select App Manager > click on New Lightning App.

A screenshot of a computer

Description automatically generated

* + Fill in the app details:
* **App Name**: Workforce Administrator Solution
* **Developer Name**: Auto-populated
* **Description**: Provide a meaningful description
* **Image**: Optional
* **Primary Color**: Keep default
  + Click Next, leave **App Options** and **Utility Items** as default, then click **Next**.
  + Add Navigation Items:
  + Search and add **Employees**, **Projects**, **ProjectTask**, **Assets**, **Asset Services**, **Reports**, and **Dashboards**.
  + Select **Asset** (custom object) created earlier.



A blue text on a black background

Description automatically generated

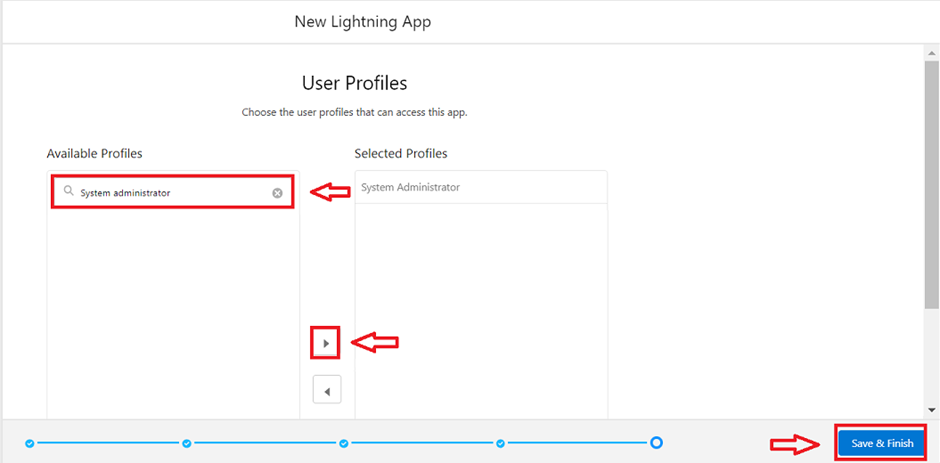
A black and blue text

Description automatically generated

* **Add User Profiles**:

* Search for System Administrator profile > click the arrow >

Save & Finish.







A black and blue text

Description automatically generated

## Step 5: Create Fields

* + Add custom fields for each object.
  + Example fields for the **Employee Object:**

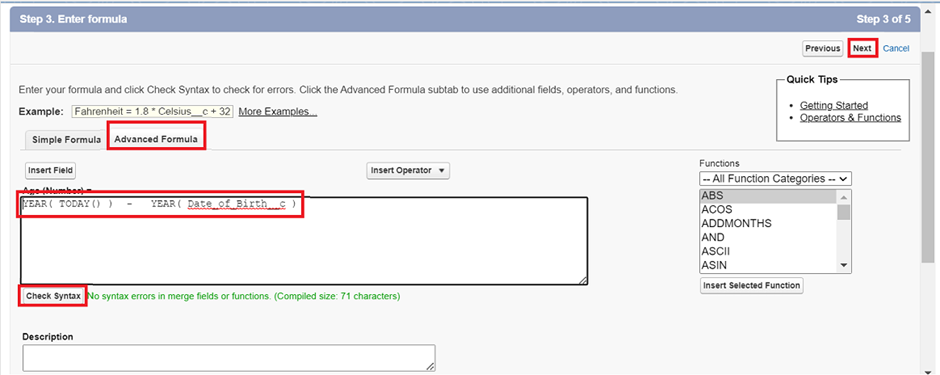
1. **Name of the Owner**
   * Field Name: Employee Name
   * Data Type: Text (Length: 18)

A screenshot of a computer

Description automatically generated

1. **Name of the Company**

* Field Name: **Age**
* Repeat for other objects.



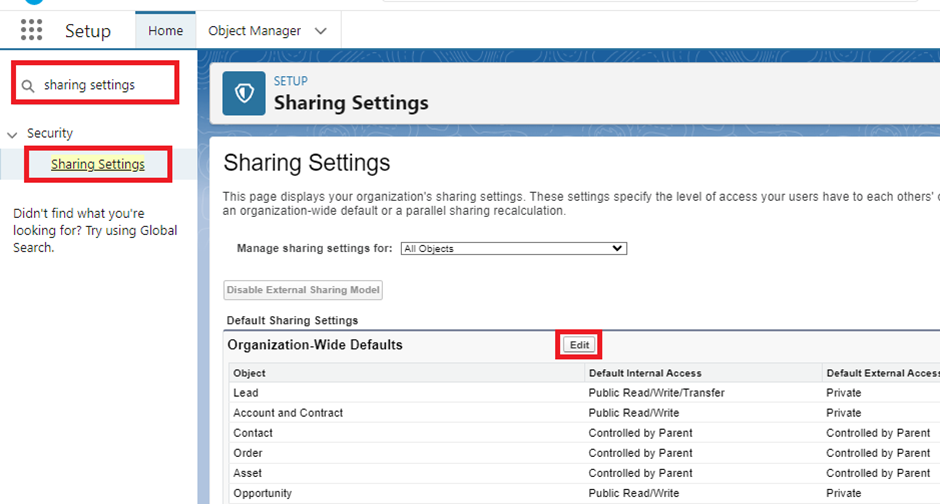






## Step 6: Create OWD Setting

* + Go to **Setup** > type "**Sharing Settings**" in **Quick Find** > click on **Sharing Settings**.
  + Click **Edit** in the Organization-Wide Defaults section.



* + Search for **Employee object**, set **Default Internal Access** and **Default External Access** to **Private**.
  + Check the **Grant Access** Using Hierarchies box.
  + Click **Save**.

A blue text on a black background

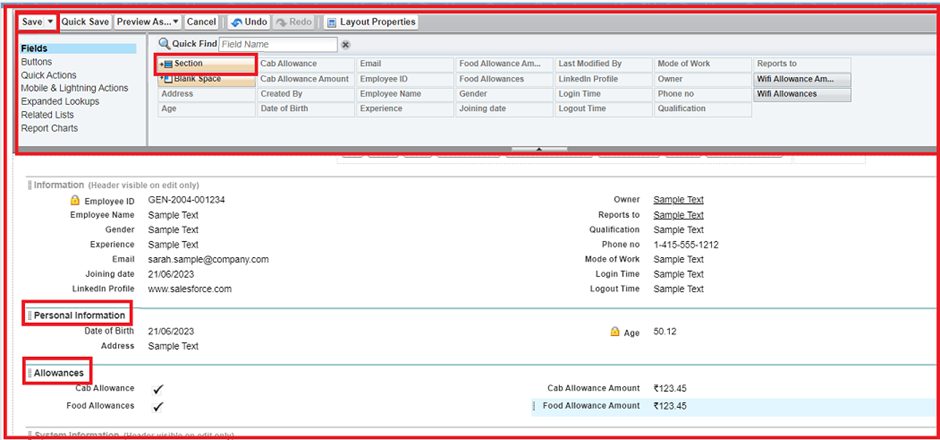
Description automatically generated

A black and blue text

Description automatically generated

## Step 8: Creating a Page Layout

* Go to **Setup** in Salesforce and click on **Object Manager**.
* Search for the **Employee object** and select **Edit** from the dropdown.
* Click on **Page Layout** and then click **New** to **create a new page layout**.
* **Name** the page layout "**On Site Employee Layout**" and click **Save**.
* Customize the layout by dragging and dropping sections like **Personal Information** and **Allowances**, adding relevant fields such as Date of Birth, **Address**, and **Age**. **Save** the layout.



* To create another page layout, repeat the process with different section names and fields, and click Save when done**.**

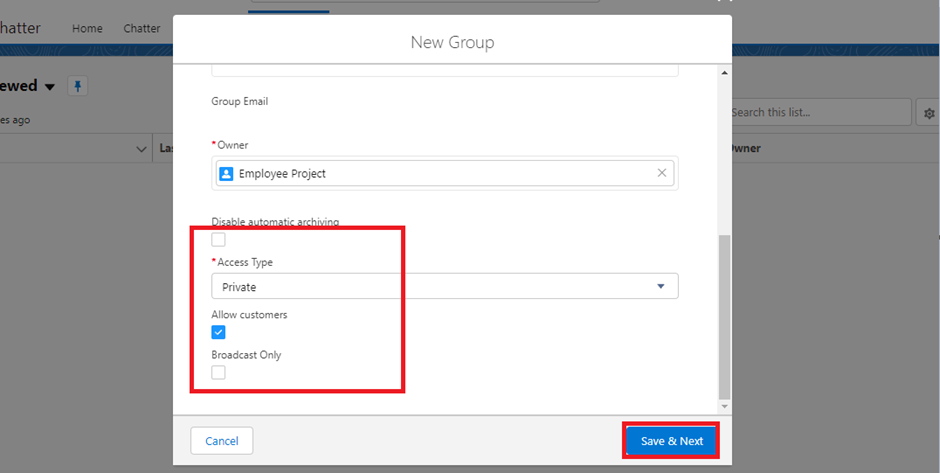






## Step 9: Creating a Chatter Group for Your Organization

* + Click the **App Launcher** in Salesforce.
  + Enter **Groups** in the **Search apps and items** box and select **Groups**.
  + Click **New** to create a new Chatter group.
  + Fill in the group details such as **Group Name**, **Description**, **Access Type** (Private), and select **Allow Customers** as unchecked.



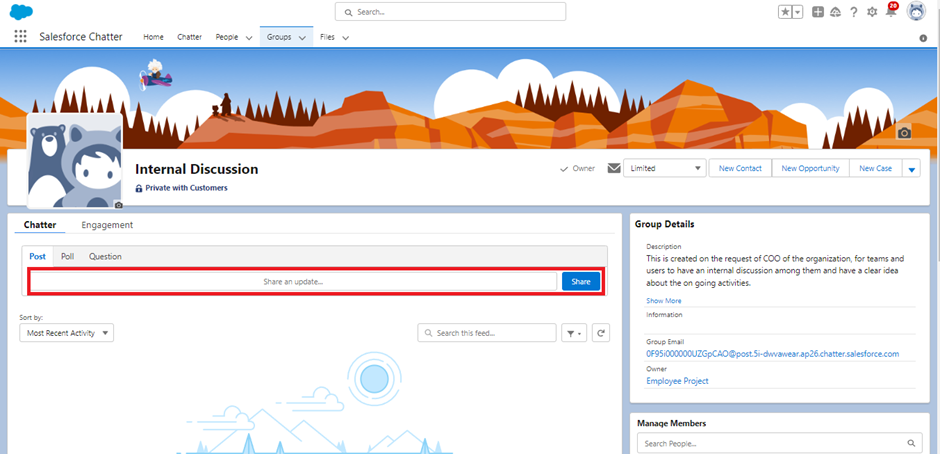
A blue text on a black background

Description automatically generated

A black and blue text

Description automatically generated

* + Click **Save & Next**, skip the **Upload Picture** section, and click **Next**. On the **Manage Members** screen, click **Add** next to the users you created earlier. Click **Done**.



* To post an update, in the group interface, type "Welcome to the Internal Discussion Group, here you can post anything related to ongoing projects" and click Share. You can also like or comment on posts.

A blue text on a black background

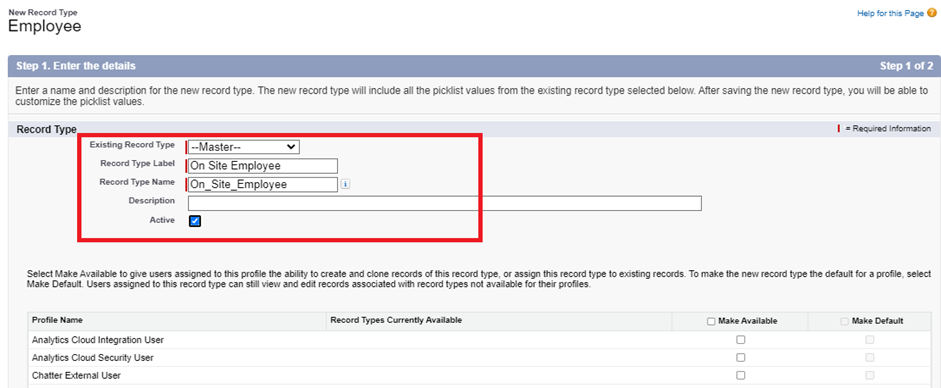
Description automatically generated

A black and blue text

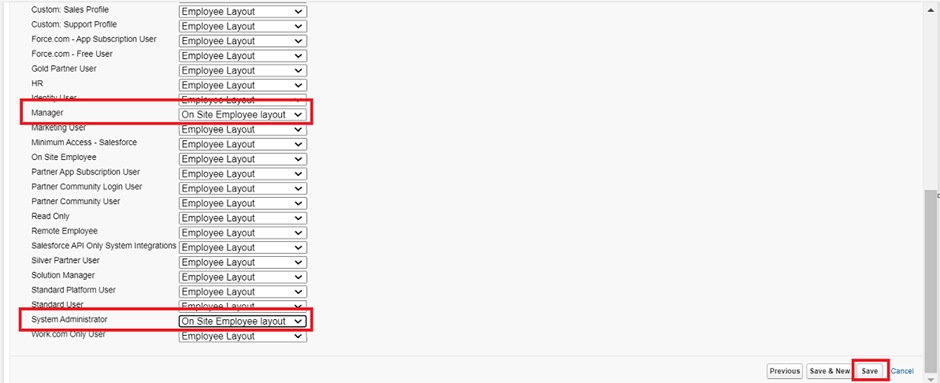
Description automatically generated

## Step 10: Creating Record Type

* + Go to **Setup** > click on **Object Manager** > search for the **Employee** object > select **Edit** from the dropdown.
  + From the left panel, click **Record Types** > click **New**.
  + Enter the **Record Type Label** as "**On Site Employee**" and make it active.



* + Uncheck "**Make Available**" and select the **Manager & System Administrator** profiles. Click **Next**.
  + Select "**Apply a different layout for each profile**", then change the page layout to **On Site Employee Layout** for the Manager and System Administrator profiles.



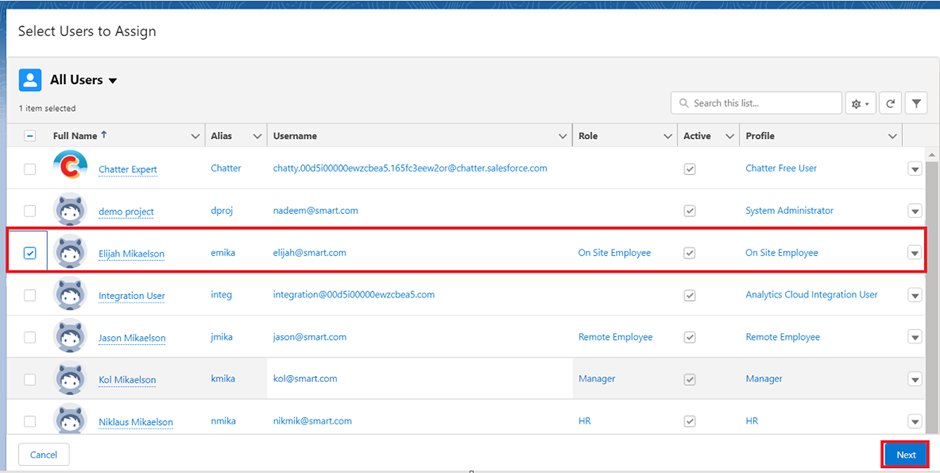






## Step 11: Creating a Permission Set

* + Go to **Setup** > type "**Permission Sets**" in Quick Find > select **Permission Sets** > click **New**.
  + Enter the label name as "**Per to Emp**" and click **Save**.
  + Under **Apps**, select **Object Settings**, then click on the **Employee** object and set **Read** and Create permissions.
  + Click **Save**, then click on **Manage Assignments** and select **Add Assignment**.



* + Choose a user with the "**On Site Employee**" profile, click **Next**, then click **Assign** and finally **Done**.

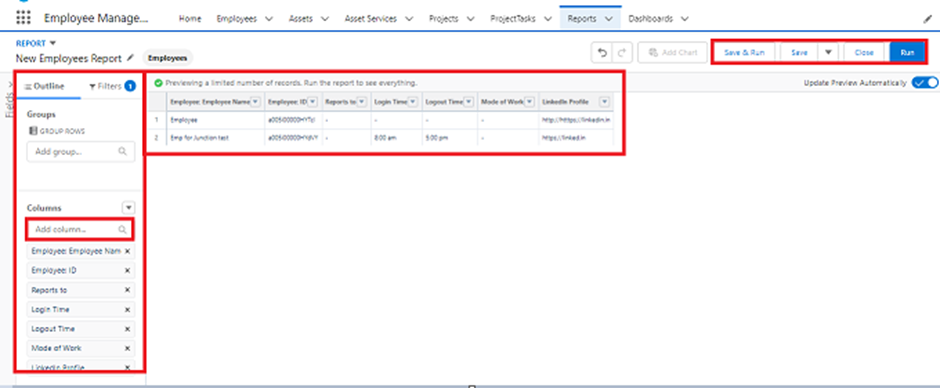






## Step 13: Create Report

* + Go to the app > click on the **Reports** tab.
  + Click **New Report**.
  + Select the report type from the category, report type panel, or search panel > click **Start Report**.
  + Customize your report by adding fields from the left pane as needed.



* + Click **Save** or **Run** to generate the report.







# Testing and Validation: Create Dashboard, Leave Object, and Apex Trigger

Effective testing ensures the functionality, performance, and reliability of the Salesforce solution. Here's a concise approach:

## Testing and Validation for Dashboard Creation

Focuses on validating backend components like Apex classes and triggers.

**Key Steps:**

* + Go to the **App** and navigate to the **Dashboards** tab.
  + **Test Report Selection**: When creating a dashboard, ensure that the report selected for the component is correct. Verify that the report data loads accurately and the correct filters are applied.
  + **UI Testing**: After adding components to the dashboard, check that all components (charts, tables, etc.) display properly. Ensure that the layout is responsive and user-friendly.
  + **Functionality Testing**: Ensure that once the dashboard is saved, it refreshes automatically with updated data when changes are made to the underlying report.
  + Performance **Validation**: Verify that the dashboard loads quickly and does not experience lag, especially with large data sets.

A blue text on a black background

Description automatically generated



A black and blue text

Description automatically generated

## Testing and Validation for Leave Object Creation

* **Field Validation**: Ensure that all fields in the **Leave** object are created correctly:
* **Employee Name**: Check if this is a Lookup field correctly linking to the Employee object.
* **No. of Days**: Ensure this field is set as **Number** and accepts valid input.
* **Status**: Verify that the **Status** field is a **Picklist** with values "Submitted," "Approved," and "Rejected," and ensure it is marked as **Read-Only** for users.\
* **Tab Validation**: Ensure the Leave object tab is accessible and displays the necessary fields. Verify that users can easily navigate to the Leave object and view all records.

## Testing and Validation for Apex Trigger

* **Apex Trigger Creation**: After creating the EmpInsert Apex trigger, test the following:
* **Trigger Logic**: Test if the trigger fires correctly when a new **Employee\_\_c** record is inserted. Verify that it checks for duplicate **Employee\_Name\_\_c** values and raises an error if an existing employee with the same name is found.
* **Error Handling**: Ensure that the error message ('Employee with same name is existing') appears when attempting to insert a duplicate **Employee\_\_c** record.
* **Unit Testing for Apex Trigger**: Write test classes to validate the trigger behavior. Ensure that the trigger is tested with both valid and invalid **Employee\_\_c** records to confirm it works as expected.
* **Performance Validation**: Test the performance of the trigger with a large set of **Employee\_\_c** records to ensure there are no performance issues.







## Overall Validation Process

* + **Integration Testing**: Ensure that the **Dashboard**, **Leave Object**, and **Apex** **Trigger** work seamlessly together. For example, ensure that leave records related to employees show correctly in the dashboard reports.
  + **User Acceptance Testing**: Have end-users test the functionality to ensure the system meets their expectations and requirements.
  + **Security Testing**: Verify that the **Read-Only** permission for the **Status** field in the **Leave** object is enforced for all users, and ensure no unauthorized changes can be made.