**PROJECT TITLE -** AssembleIT – The PC Builder Application

1. **PROJECT OVERVIEW**

The PC Builder Application in Salesforce enables users to customize personal computers by selecting various components and their versions. Key features include real-time compatibility checks, dynamic pricing updates, and a user-friendly interface that visually represents the build. Built using Lightning Web Components and Apex, the application integrates with Salesforce to store user profiles for personalized experiences. The project was completed in 1 week and aims to streamline the PC customization process, making it an essential tool for tech enthusiasts.

1. **OBJECTIVES**
2. Simplify Customization:  
   **Business Goal**: Enhance user engagement and satisfaction.  
   **Outcome**: Users can easily select and customize PC components, leading to increased time spent in the application.
3. Ensure Compatibility:  
   **Business Goal:** Reduce returns and customer dissatisfaction.  
   **Outcome:** Implement automatic compatibility checks to minimize errors in component selection.
4. Real-Time Pricing:  
   **Business Goal:** Improve purchasing confidence and conversion rates.  
   **Outcome:** Provide instant pricing updates, enabling users to make informed budget decisions.
5. Visualize Builds:  
   **Business Goal:** Increase user interaction and excitement.  
   **Outcome:** Offer a visual representation of the completed PC, enhancing the overall building experience.
6. User Engagement:  
   **Business Goal:** Foster community and word-of-mouth marketing.  
   **Outcome:** Allow users to save and share their builds, promoting social interaction and feedback.
7. Salesforce Integration:  
   **Business Goal:** Streamline user management and data utilization.  
   **Outcome:** Leverage Salesforce capabilities for personalized user profiles, improving customer retention and targeting.
8. **SALESFORCE KEY FEATURES AND CONCEPTS UTILIZED**

1. Custom Objects

* Description: Custom objects allow you to store data specific to your application.
* Utilization: Creating objects like PC\_Build, Component, and PC\_Build\_Component enables you to organize and manage information about PC configurations and components.

2. Custom Fields

* Description: Fields in custom objects store specific data attributes.
* Utilization: Define fields such as Price, Type, and Version in the Component object to capture detailed information about each part.

3. Record Types

* Description: Record types enable you to create different business processes for different users.
* Utilization: If you have various types of components (e.g., Gaming vs. Workstation), you can create record types for the Component object to manage these categories distinctly.

4. Picklist Fields

* Description: Picklists provide a predefined list of values for a field.
* Utilization: Implement picklist fields for Type and Status to ensure consistent data entry and user selection.

5. Validation Rules

* Description: Validation rules help enforce data quality by preventing the saving of invalid data.
* Utilization: Create validation rules to ensure selected components are compatible, or that the build has all necessary components.

6. Salesforce Lightning Components

* Description: Lightning Components are used to build dynamic web applications.
* Utilization: Develop a user-friendly interface for selecting components and displaying the build summary using Lightning Components.

7. Salesforce Flow

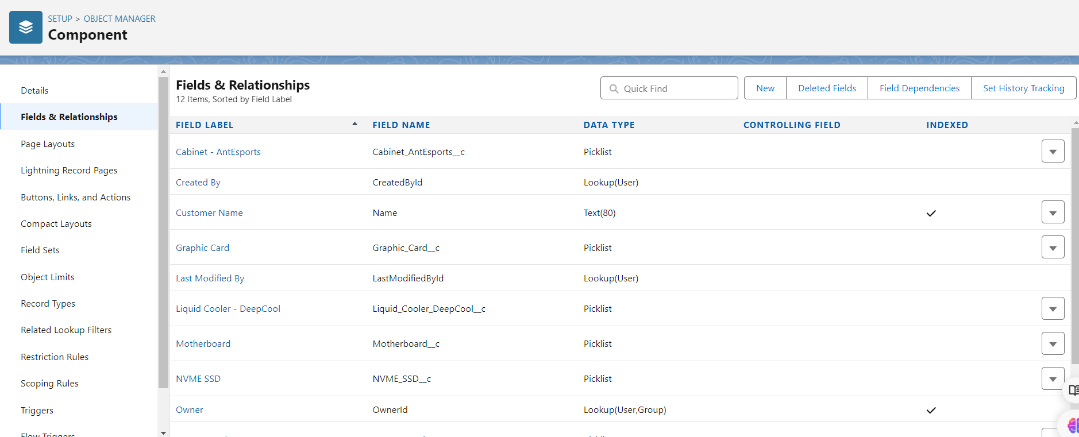
* Description: Flow allows for automation through visual workflows.
* Utilization: Use Flow to guide users through the process of creating a PC build step-by-step, collecting their selections and providing feedback.

8. Reports and Dashboards

* Description: Reports and dashboards provide visual insights into your data.
* Utilization: Create reports to analyze popular components and user builds, helping you understand trends and improve the application.

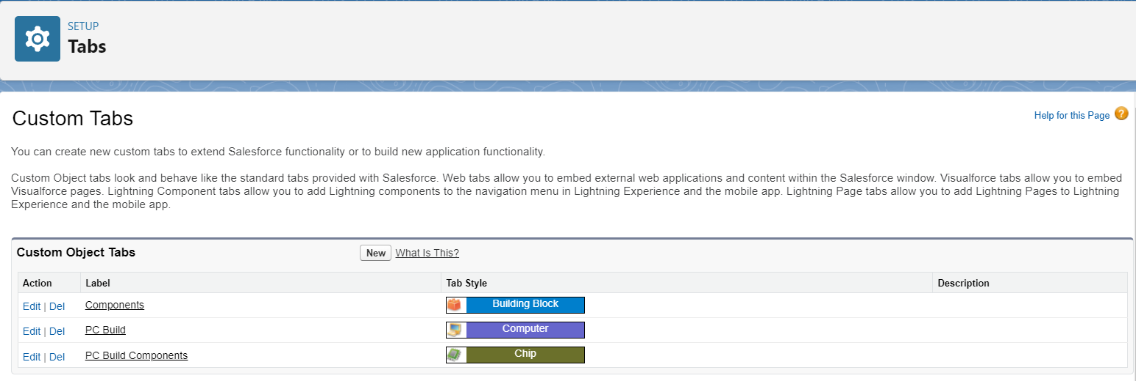
1. **DETAILED STEPS TO SOLUTION DESIGN**

* Firstly, I have created three objects and the fields in them.
* A screenshot of a computer

  Description automatically generated****The data types used in the fields are lookup, text,number,etc.

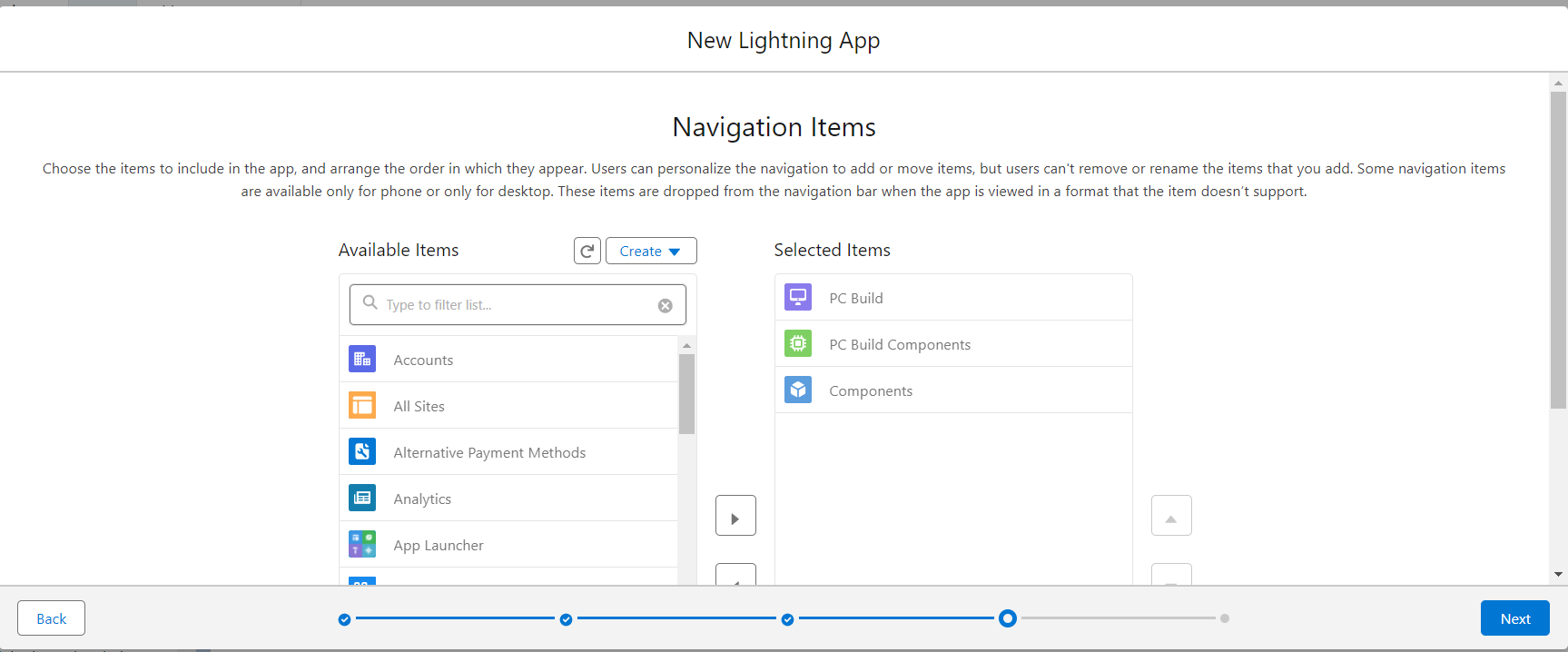
A screenshot of a computer

Description automatically generated

Then, tabs are created in the custom tabs. The custom objects are used to create the tabs using the relative tab style.

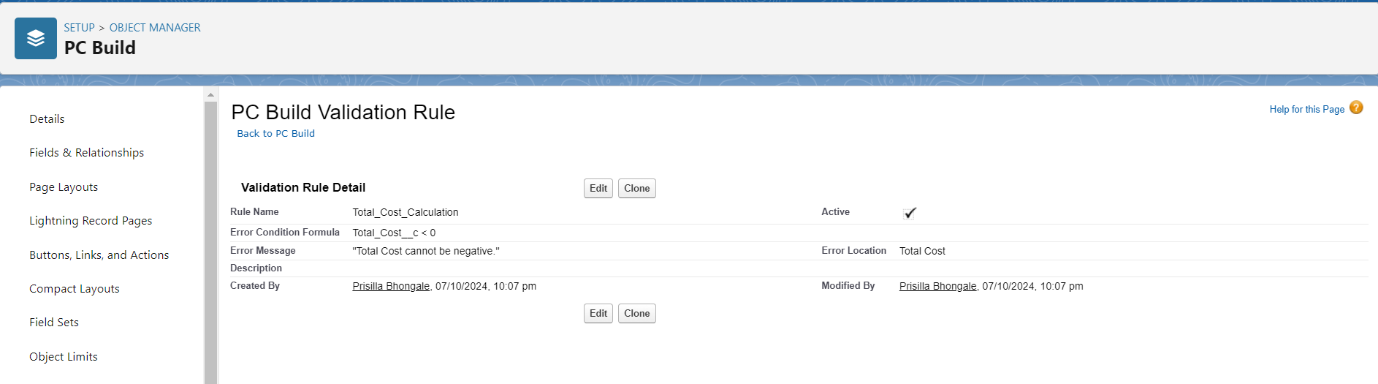
* After creating the tabs, I have created the lightning app in app manager.
* The tabs are added that have been created using custom objects and also tbe user profiles are added.
* The user profiles added are Standard Platform user, Standard user and System Administrator.

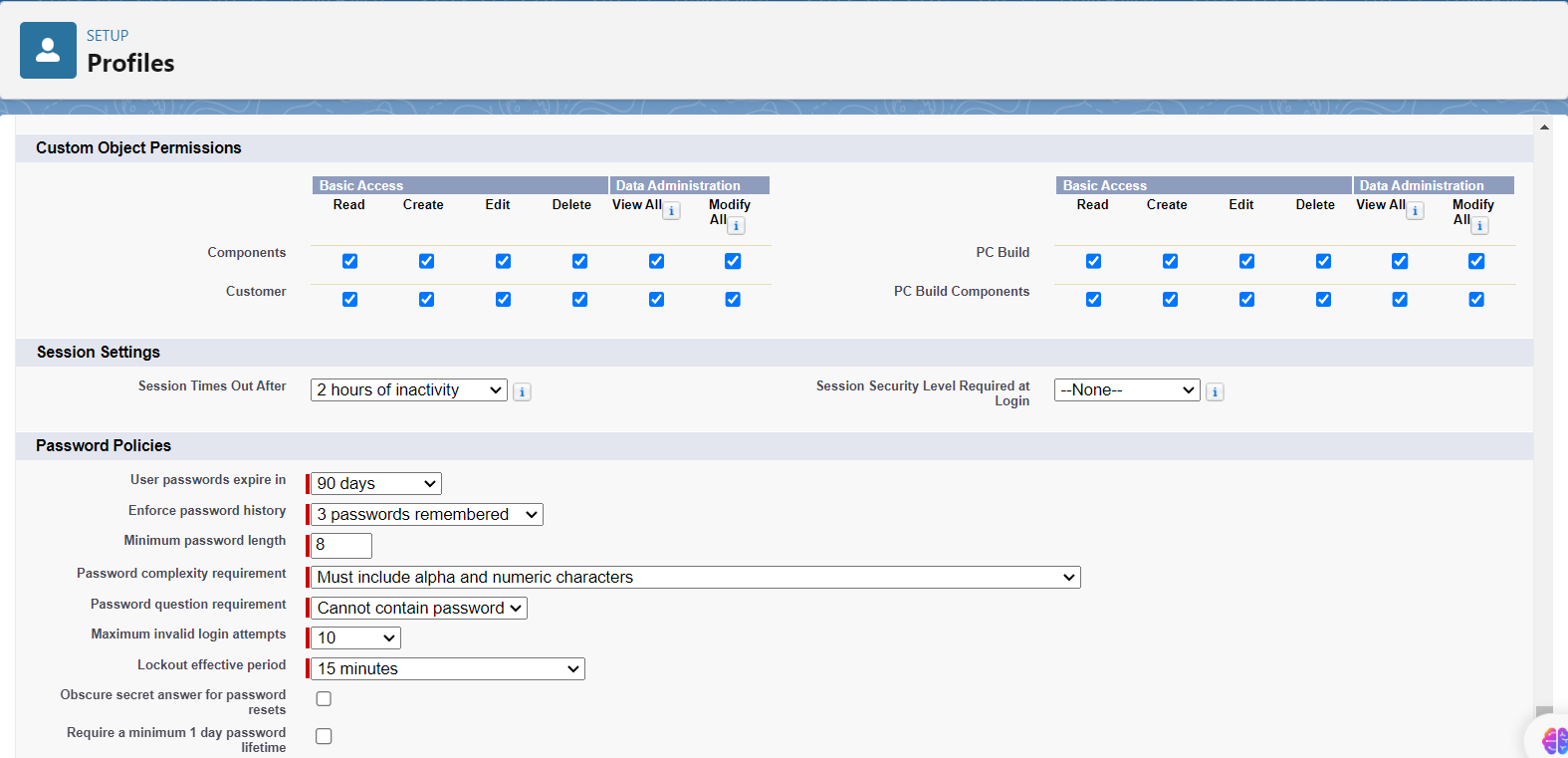
A screenshot of a computer

Description automatically generated

* After lightning app is created, Validation rules are made.
* Validation Rules are the rules that are made to be followed compulsorily or else it prints the error message and the process is not carried forward.

A screenshot of a computer

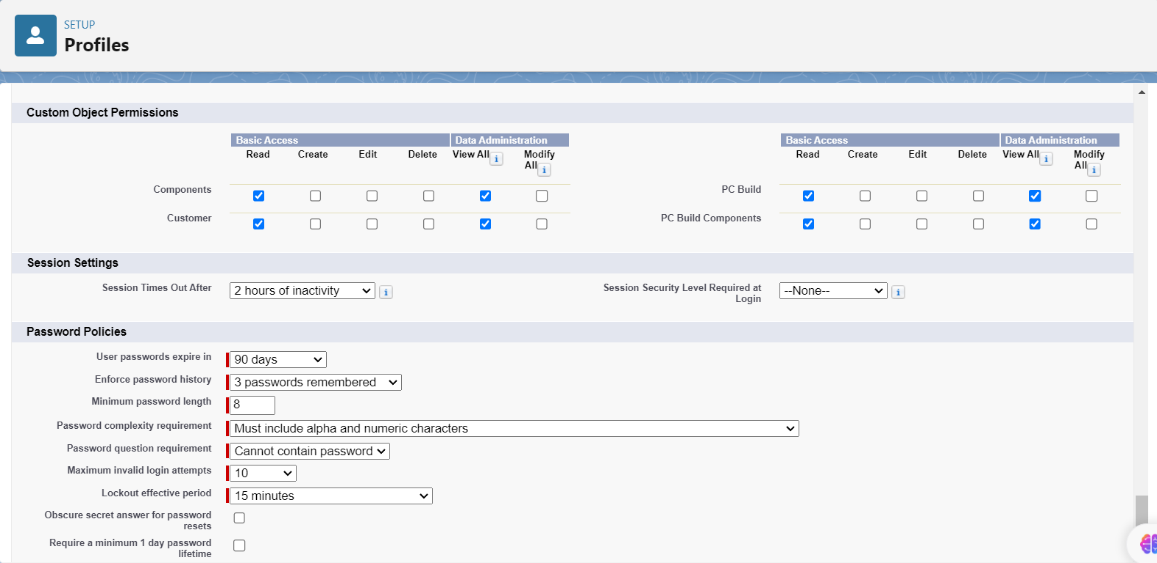
Description automatically generated

* For profiles, I have cloned the profile of Standard User and made the Manager Profile.
* Then the custom app setting is set to AssembleIt i.e.,The application that is being developed.
* A screenshot of a computer

  Description automatically generatedThe manager is given the access to all the custom object permissions.

The Standard Platform user is cloned and made as Technician profile.

* The technician is given only to read and view all permission.

A screenshot of a computer

Description automatically generated

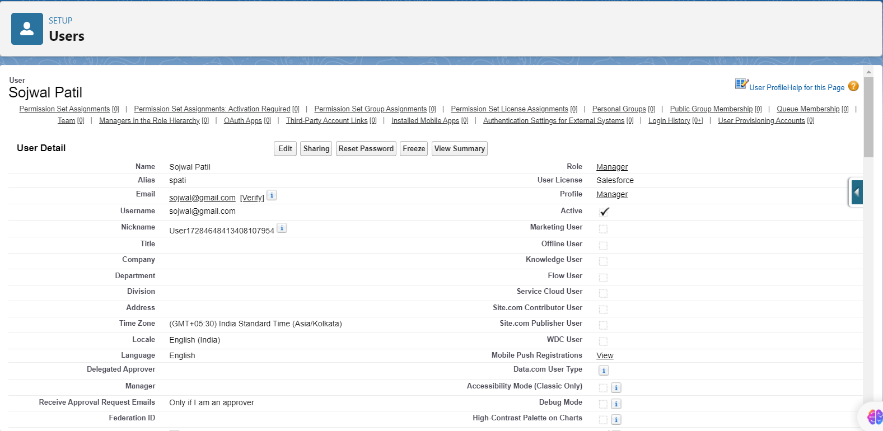
* In Role & Role Hierarchy, two roles are added the Manager role and the technician role.
* The Manager is added under CEO and the Technician is added under the Manager.

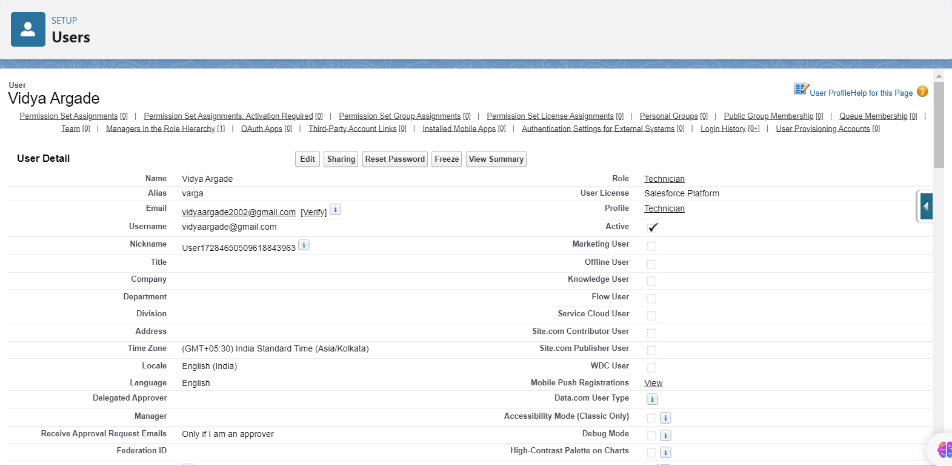
A screenshot of a computer

Description automatically generatedA screenshot of a computer

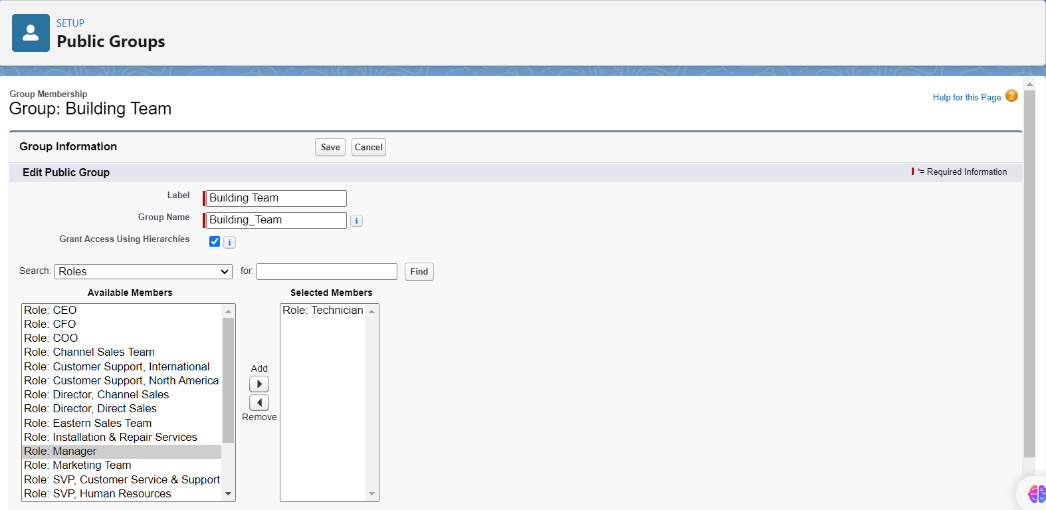
Description automatically generated

* In Users, two users are added as Manger and Technician.
* Sojwal Patil is given the Manger role and profile and Vidya Argade is given the Technician role and profile

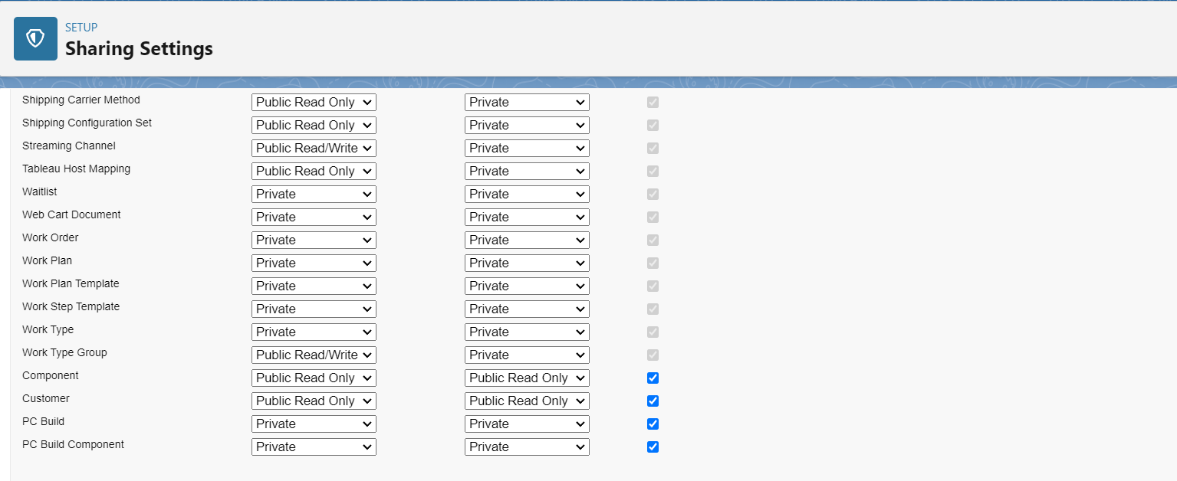




* I have created a public group in which technician is added.
* Public group is made to have appropriate access to technicians to the resources they need.



* Sharing settings is used change the access to the objects.



* I have created a flow named PC Build Selector.
* A diagram of a computer

  Description automatically generatedIn it the process in which the application will work is stored.
* Firstly, the customer details will be filled.
* I have created the flow in this manner that the values filled will get directly stored to object and the field.

A screen shot of a computer

Description automatically generated

* After the customer details are filled The build name and the components needed are selected.
* A grid of lines with a white background

  Description automatically generated with medium confidenceThe versions of each component is selected.
* Report is also created.
* New report is created according to the PC Build name, Components and Total cost.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* This is the Dashboard of my application.

A screenshot of a computer

Description automatically generated

1. **KEY SCENARIOS ADDRESSED BY SALESFORCE IN THE IMPLEMENTATION PROJECT**
2. **Seamless Customer Information Collection**:
   * **Scenario**: Capture customer details (name, email, phone number) before they begin selecting components for their custom PC build.
   * **Solution**: Salesforce Flow was utilized to create a step-by-step process for entering customer details and linking them to the Customer object. This ensures all relevant customer data is captured and stored efficiently.
3. **Efficient Component Selection**:
   * **Scenario**: Allow customers to select various PC components (such as motherboard, CPU, etc.) with different versions available for each component.
   * **Solution**: Custom picklist fields in the Component object were used for each type of component, with options for versions. The selected components were then stored in the PC\_Component object, enabling easy tracking and association with customer builds.
4. **Real-Time Cost Calculation**:
   * **Scenario**: Customers need to see the total cost of the components they select in real-time.
   * **Solution**: A formula field in the PC\_Build object was implemented to calculate and display the total cost of all selected components, giving customers an immediate view of the cost of their custom PC.
5. **Validation of Component Selections**:
   * **Scenario**: Ensure that only valid and complete PC builds are processed, preventing incomplete submissions.
   * **Solution**: Validation rules were applied to ensure customers select all required components and fields. This enhanced data accuracy and user experience by prompting users to complete any missing information before finalizing the build.
6. **Custom Reporting for Build Analysis**:
   * **Scenario**: Track and analyze customer builds, including which components are selected most frequently and how much customers are spending on average.
   * **Solution**: Custom Salesforce reports were created using the data from the PC\_Build, PC\_Component, and Customer objects. These reports provide insights into customer behavior and product trends, which can inform business decisions and marketing strategies.
7. **Dashboard for Real-Time Visualization**:
   * **Scenario**: Provide a dashboard that visualizes key performance indicators (KPIs) such as total sales, popular components, and customer data.
   * **Solution**: A dynamic Salesforce dashboard was created to visualize key metrics, such as total build costs, customer counts, and top-selected components. This helps business owners and administrators monitor the success of the application in real-time.
8. **Automated Data Management and Linking**:
   * **Scenario**: Ensure all customer data, build data, and component data are automatically linked and stored without manual intervention.
   * **Solution**: Salesforce automation features, such as lookups and master-detail relationships, were implemented to link customer data with their respective builds and components, minimizing manual data handling and ensuring data consistency.
9. **Role-Based Data Access**:
   * **Scenario**: Control access to sensitive customer and financial data based on user roles.
   * **Solution**: Role hierarchy and sharing settings were configured to control who can view or edit certain data, ensuring that only authorized users have access to sensitive information.
10. **CONCLUSION**

The PC Building Application built in Salesforce allows users to seamlessly select computer components and calculate the total cost of their custom PC builds. By utilizing Salesforce's robust features, the application efficiently captures customer details, enables component selection through custom objects, and calculates the total cost in real-time.

With the integration of:

* Custom Objects: For storing customer information, build details, and component selections.
* Flows: To guide users through the process of providing their information, selecting components, and viewing the total cost.
* Validation Rules: Ensuring that only valid component selections are made, maintaining data accuracy.
* Reports: Allowing you to track builds, components, and costs with detailed analysis.
* Dashboards: Providing real-time visual insights into the most popular components, total build costs, and customer trends.

This application not only streamlines the PC building process but also offers valuable insights to improve decision-making and customer service. By leveraging Salesforce's automation and reporting capabilities, the application is a powerful tool for both customers and administrators, ensuring a smooth and customized PC-building experience.