**SCHOOL OF COMPUTER SCIENCE & IT DEVI AHILYA VISHWAVIDHALAYA**

INDORE, (DEC-MAY 2022-23)

PROJECT REPORT

**An Inventory Management App in Salesforce**

**Master of Business Administration**

**(Computer Management)**

**IV SEMESTER**

|  |  |
| --- | --- |
| **PROJECT GUIDE -** | **SUBMITED BY -** |
| **Dr. Archana Thakur** | **Sagar Rakesh Sahu - 210416** |
|  | **Siddharth Pandey - 2110418**  **Srajan Sagare - 2110421** |

**SCHOOL OF COMPUTER SCIENCE & IT DEVI AHILYA VISHWAVIDHALAYA**

INDORE, (DEC-MAY 2022-23)

# **CERTIFICATE FROM GUIDE**

It is to certify that dissertation on **“An Inventory Management App in Salesforce**,” submitted by **Mr. Sagar Rakesh Sahu, Mr. Siddharth Pandey and Mr. Srajan Sagare** to the *School of Computer Science & IT, DAVV, Indore* has been completed under my supervision and the work is carried out and presented in a manner required for its acceptance in partial fulfilment for the award of the degree Master of Business Administration (Computer Management).

#### PROJECT GUIDE

**Name: Dr. Archana Thakur**

**Signature:**

#### Date:

**SCHOOL OF COMPUTER SCIENCE & IT DEVI AHILYA VISHWAVIDHALAYA**

INDORE, (DEC-MAY 2022-23)

# **CERTIFICATE**

It is to certify that we have examine the dissertation on **“An Inventory Management App in Salesforce”**, submitted by **Mr. Sagar Rakesh Sahu, Mr. Siddharth Pandey and Mr. Srajan Sagare** to the *School of Computer Science & IT, DAVV, Indore* and hereby accord our approval of it as a study carried out presented in a manner required for its acceptance in partial fulfilment for the award of the degree Master of Business Administration (Computer Management).

***Internal Examiner External Examiner***

*Signature:* Signature:

*Name:* Name:

*Date:* Date:

**SCHOOL OF COMPUTER SCIENCE & IT**

**DEVI AHILYA VISHWAVIDHALAYA**

INDORE, (DEC-MAY 2022-23)

**DECLARATION**

I hereby declare that the project titled **“An Inventory Management App in Salesforce”** submitted by us for the partial fulfilment of the requirement for the award of MBA (Computer management) to *School of Computer Science & IT, DAVV, Indore* comprises my own work and due acknowledgement has been made in text to all other material used.

#### NAME:

Sagar Rakesh Sahu

Siddharth Pandey

Srajan Sagare

**Date:**

**Place: - Indore**

**ACKNOWLEDGMENT**

We acknowledge our sincere thanks to those who have contributed significantly to this project. It is a pleasure to extend deep gratitude to our internal guide **Dr. Archana Thakur (Sr. Lecturer), SCSIT** for her valuable guidance and support and to continuously prompt us for the progress of the project. We thank her for her valuable suggestions towards our project, which helped us in making this project more efficient and user friendly. We thank and acknowledge each and everyone’s efforts that helped us in some or the other way for small and significant things.

# **Contents**

[Abstract](#_TOC_250020)

Chapter 1: Introduction to Cloud Computing

* 1. Service Models of Cloud Computing

### 1.1.1 Software as a Service (SaaS)

### 1.1.2 Platform as a Service (PaaS)

### 1.1.3 Infrastructure as a Service (IaaS)

### Chapter 2: Salesforce

* 1. [Introduction to Salesforce](#_TOC_250019)
  2. [The architecture of Salesforce](#_TOC_250018)
  3. Salesforce Multiple View Controller
  4. Different editions of Salesforce
  5. [System Overview](#_TOC_250017)
  6. [Technologies of Salesforce](#_TOC_250016)
  7. Benefits of Using Salesforce

Chapter 3: Inventory Management Analysis and Business Processes (Former)……..

* 1. The Inventory Management
  2. The Inventory Management Manual Process
  3. Benefits of a Manual Inventory System
  4. Drawbacks of a Manual inventory System
  5. Manual vs. Automated Inventory System

Chapter 4: Inventory Management in Salesforce

* 1. [Development Area](#_TOC_250015)

4.1.1 [Custom Apps](#_TOC_250014)

4.1.2 [Objects](#_TOC_250013)

4.1.3 [Fields](#_TOC_250012)

4.1.4 [Relationships](#_TOC_250011)

4.1.5 [Tabs](#_TOC_250010)

4.1.6 [Securing and Sharing Data](#_TOC_250009)

4.1.7 [Page Layouts](#_TOC_250008)

4.1.8 [Visual Force](#_TOC_250007)

4.1.9 [Apex](#_TOC_250006)

4.1.10 [Chatter](#_TOC_250005)

4.1.11 [Reports and Dashboards](#_TOC_250004)

* 1. Business Process before Salesforce……………………………………...

4.2.1 Traditional Method……………………………………

* 1. Revolution after Salesforce………………………………………………

Chapter 5: Salesforce Selection Methodology

* 1. What is Project Methodology
  2. Use of Selection Methodology
  3. Selecting the most Appropriate Salesforce Solution
  4. [Inventory Administration](#_TOC_250003)

Chapter 6: Inventory Management Implementation

6.1 Inventory Administration

6.2

Chapter 7: Conclusion

# ABSTRACT

Salesforce is a cloud-based customer relationship management (CRM) software that accelerates business relationships and can transform the working lives of the team. Marc Benioff developed it in the late 1990s and now it has been announced as the world's most innovative company for six consecutive years by Forbes Magazine [1]. Unlike traditional CRM software, Salesforce is an internet service. It is available with just a sign-up and logs in through a browser, and it is immediately available. It is based on cloud computing, where the customers, without the need of installing any traditional software, can access the cloud, i.e., through the internet, for their business needs [2].

Inventory Management (IM) is the method of controlling and supervising the storage, utilization, and ordering of components that an organization can track of their items it sells. It is the act of controlling and administering the quantities of products in the sale. For a business, an inventory is the main asset which represents an investment by the owner until the item is sold [3].

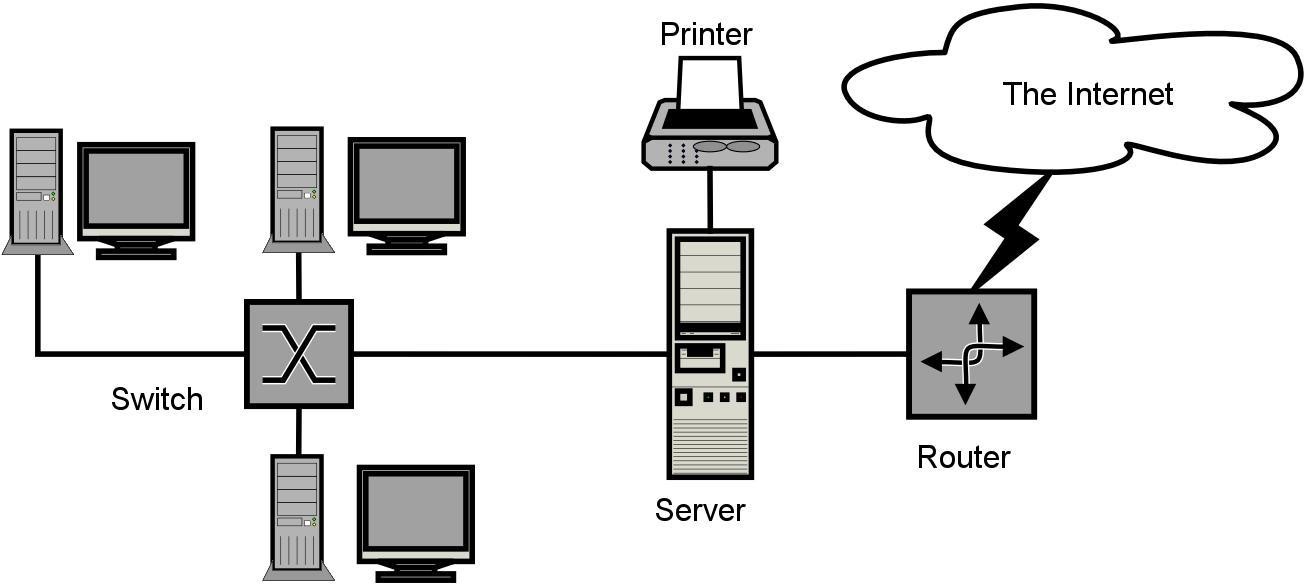
To demonstrate its functionalities of Salesforce, we created an application for inventory management. Here, In the inventory management, there are two parts: one is administration part, and another part is customer portal. The administrator manages the inventory and store operations, and the customer buys the products in the inventory through customer portal.

# **Chapter 1: Introduction to Cloud Computing**

Cloud computing is usually defined as storing and managing the data over the cloud, rather than a local server. Cloud computing is easy to understand. All applications are developed and run in the web browser. Using the internet connection, users and developers will have access to whole applications thus eliminating the complexity and overhead of the maintain environment.

Unlike traditional business applications which are complicated, expensive and need experts to install, run, update and secure, cloud computing can be accessed anywhere with an online connection. In traditional systems, the entire infrastructure must work together. For such type of seamless interaction, and for the smooth run of the system, a constant maintenance is always required. With cloud computing, there is no necessity to invest money in acquiring and supporting hardware and software infrastructure, thus decreasing the potential cost for users and developers.

The main impact of cloud computing is on the responsiveness of IT systems. With the cloud computing environment, we can add users and developers instantly, and the applications can be deployed rapidly into the cloud which reduces the user request response time. As the complexity of the of the internal systems is removed, the organization can speed up the entire IT process.



**Cloud Computing**

**1.1 Service Models of Cloud Computing**

### **1.1.1 Software as a Service (SaaS)**

The end user can access the application which is developed by the provider on a cloud framework. The developed applications are available from different customer devices through interfaces like a web program or a program interface. Cloud infrastructure, servers, networks, storage, and operating systems cannot be managed or controlled by the customer.

Cloud application services represent the increasing cloud market. Software as a service utilizes the internet to deploy the applications overseen by the vendor and whose interface will be able to access on the customer side. Many of the applications developed using SaaS will run in a web browser by using some plugins. There will not be need of any download or establishments.

The major applications which are developed using SaaS are healthcare related applications, client relationship administrations, incorporate email, and collaboration. Some of the costly ventures which are not able to considered as software vendors started using SaaS to get the upper hand and gain income.

### **1.1.2 Platform as a Service (PaaS)**

The Client can deploy onto the cloud infrastructure developed by the customers with the help of libraries, tools, services, and the programming languages which are supported by the client. The underlying infrastructure of cloud and storage, servers, network, or operating systems cannot be managed or controlled by the Customer.

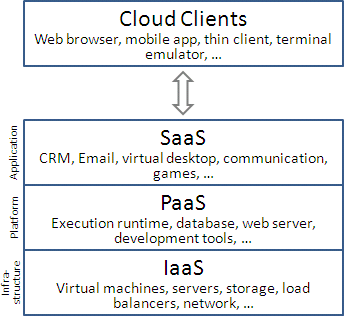
Cloud platform services can be utilized for applications and their advancements when cloud segments are given to programming. Developers can be able to pick the structure using PaaS where the applications can be expanded to create or modify. The testing and deployment of applications become easy and fast if the PaaS is used.

Enterprise PaaS gives a self-service portal to programming engineers for overseeing computing infrastructure from information technology operations. Scalability, Software as a service enablement and multi-occupancy can be acquired by the applications using PaaS. The coding fundamental measure will be decreased for the enterprises using PaaS and the application will be converted to a hybrid model.

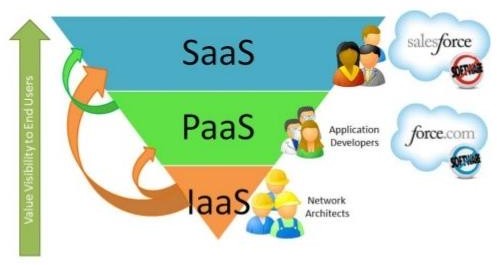
### **1.1.3Infrastructure as a Service (IaaS)**

The customer can arrange systems, storage, processing, and other essential computing resources in which the client can run and send arbitrary software like applications and operating systems. The hidden cloud infrastructure is not controlled or overseen by the customer but rather he can control over operating systems, storage, servers, and network.

Cloud infrastructure services, referred to as IaaS, are models beneficial for overseeing and observing remote data center frameworks such as organizing, processing, stockpiling, and networking services. With the help of IaaS, clients will get utility billing and power benefits.



### Figure 2: Service Models of Cloud Computing



**Service Models**

**Chapter 2: Salesforce**

## **Introduction to Salesforce**

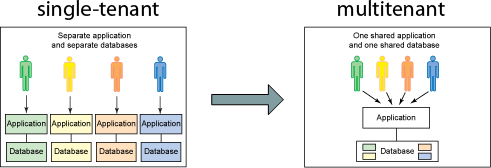
Salesforce is one of the world's prime cloud computing companies and number one on-demand customer relationship management (CRM). Salesforce does not need any software installation or hardware or any infrastructure like servers. All we need to access Salesforce is the internet. This empowers even the most non-techie individuals to be able to use the system and configure it as per their needs.

Established as Salesforce.com(SFDC) and its customer relationship management (CRM) service and then divided into different sectors like sales cloud, service cloud, community cloud, analytics cloud, data cloud, marketing cloud, app cloud, and so on.

Since Salesforce coordinates well with all the platforms and supports all major OS and mobile devices, it is anything but difficult to utilize Salesforce outside of the workplace, thus helps to improve productivity.

## **The Architecture of Salesforce**

Salesforce has a multi-tenant Architecture. Multi-tenancy is the fundamental technology utilized as a part of the cloud to share its resources safely and cost effectively. It is much the same as bank services where various tenants cost-efficiently share a common infrastructure yet safely and with most protection from other tenants. A cloud utilizes multi-tenant infrastructure to share its assets safely among different applications and occupants (organizations, associations, and so on) that use the cloud. Some clouds utilize virtualization-based architecture to confine occupants; others utilize custom software architecture to take care of business. The multi-tenant outline of a cloud service can dramatically affect the application delivery and the profitability of IT organization.



**Multi-Tenancy**

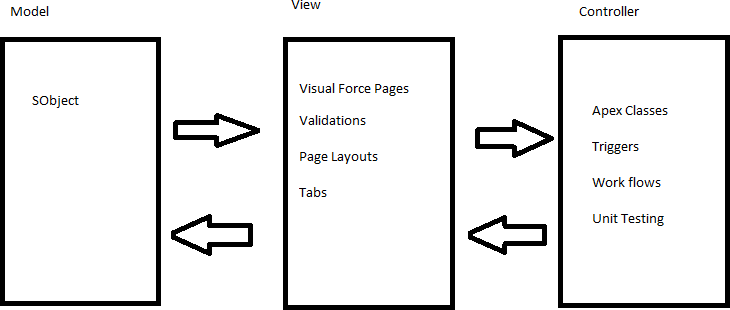
## **Salesforce Multiple View Controller (MVC)**

MVC is a design pattern which separates business logic from interface logic i.e. it separates the graphical interface displayed to the user with the code that manages the user actions [4]**.**

In Salesforce, using SFDC visual force, we can write VIEW pages which are very similar to java servlets page(JSP) pages. Each visualforce (VF) page is corelated with a controller. The controller and model classes can be written using Apex language. In SFDC, controller part comprises of workflows, triggers, Apex classes and model layer comprises of fields, relationships, objects, and View layer comprises of Tabs, page layouts, VF pages.

SFDC MVC mainly consists of three modules namely Model, View, and Controller.

1. Model: Here we represent what schema and data the Salesforce used for the system representation and objects are a model, as every entity is mapped to some subject in Salesforce.
2. View: Here we represent how data and schema and visual force are used to present data to users.
3. Controller: Here we use controllers and interface actions to perform actions when the user interacts with visual force.



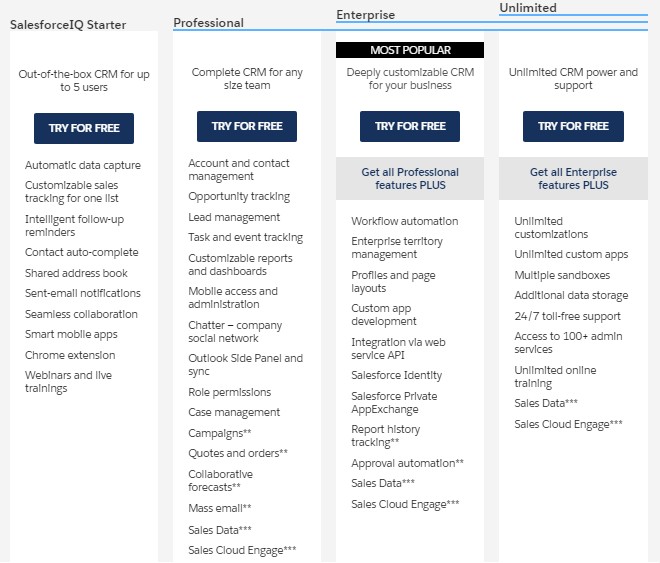
**Salesforce MVC**

## **Different editions of Salesforce**

Salesforce offers different editions of its products and services depending on business needs.

1)Professional Edition: This edition offers full CRM functionality.

1. Enterprise Edition: This edition offers advanced CRM customization and administration tools along with functionalities of Professional edition.
2. Unlimited Edition: Along with functionalities offered by Enterprise Edition, Unlimited Edition even offers full mobile access, premier support, unlimited custom apps and more.
3. Developer Edition: This edition allows developers to extend the Salesforce system, integration with other applications and develop new applications and tools.

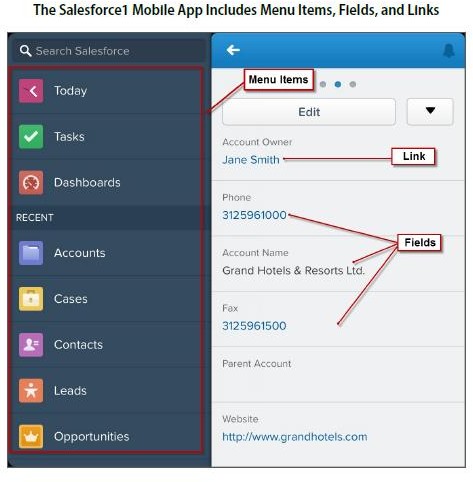


**Different Versions of Salesforce**

## **System Overview**

Salesforce.com allows administrators to configure and design systems for complex implementations. Salesforce combines the power of configuration and custom development in its platform i.e., Force.com Platform. This platform user can make use of custom code, workflows rules, approval processes to implement their business logic and they can integrate the data with other applications, generate reports and do the analytics within no time. The Salesforce CRM model is used in organizations for interactions like emails, meetings, events with customers and for prospects like sales, marketing, and support.

With Force.com, we can run business in the mobile using the Salesforce1 app. We can build and optimize the apps for mobile using HTML 5 and UI framework and it supports all devices with just one code base [5]. Salesforce1 downloadable app can be installed from the App store or Google play on a mobile device.



### Salesforce Mobile App 1

Salesforce even has an app marketplace called AppExchange where we can find pre-built business applications. AppExchange offers thousands of verified and secured apps built by others or we can develop our own app and sell it here. It’s quite similar to the App store and the play store.

Due to its ease of access, ease of use, minimum licensing/proprietary issues, and per user cost, Salesforce becomes a power system from small to large scale industries**.**

## **Technologies of Salesforce**

### Apex

Salesforce has a programming language called Apex. It is a case-insensitive, mostly typed object-oriented programming language with syntax identical Java with curly brackets and dot- notation syntax. Apex is used to run programs and procedure in Force.com such as links, buttons, record insertion and so on with visual force custom controllers.

### Visualforce

Visualforce (VF) is a framework for the Force.com platform with tag-based markup language identical to HTML. With the help of Visualforce, custom pages can be created for mobile apps and desktops with the help of with other front-end technologies like HTML, CSS, jQuery, and JavaScript. With the Visualforce standard and custom controller features, we can build our own business logic in Apex.

1. Lighting

Lightning is a component-based framework for the Salesforce1 mobile app which is built on an open-source Aura framework. With the lighting framework, responsive applications can be built easily. The apps build on the Lighting framework is sold or brought on AppExchange .

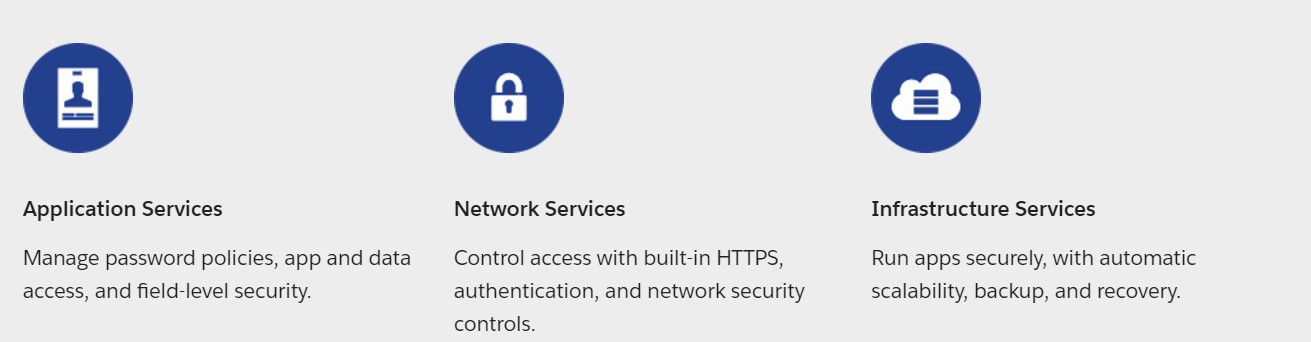
Lightning App builder for Salesforce is a tool for quick application advancement of responsive web interfaces. This interface considers distinctive screens to be assembled given lightning segments. This can be layouts as formats for records or applications.

## **Benefits of Using Salesforce**

### Invest in innovation, not infrastructure

With Salesforce, we can focus on business rather than the back end as Salesforce add new features and automatic upgrades three times a year. These boundary-pushing work made Salesforce one of Forbes’ Most Innovative Companies.

### Do not stress about the data security



1. Make Salesforce work the way we want

Salesforce can be customized to the core to be more agile and productive and the apps can be developed with interfaces with point and click to high-end platforms.

### With AppExchange, find Prebuilt applications in minutes

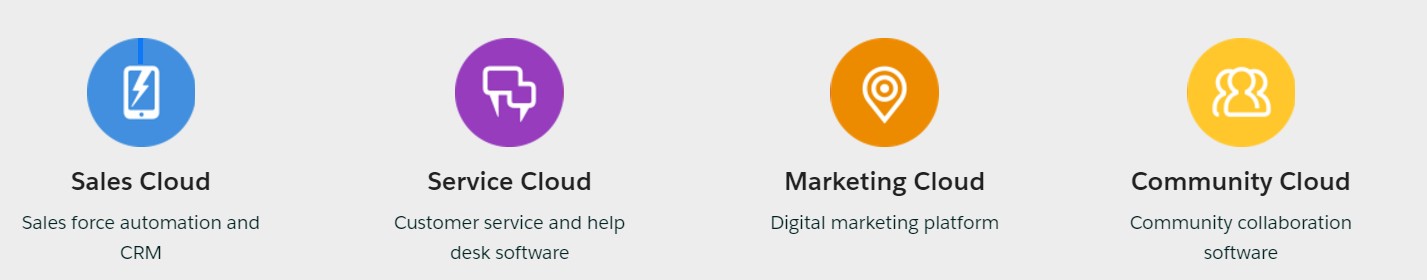
With industry-specific third-party apps, Salesforce can be spread to every division and corporation. These apps are installed by millions of people and are reviewed by thousands. So, with trust on the apps, we can focus on extending the business.

### Work on one platform where everything works together

With Salesforce APIs, core technologies and third-party tools, we can connect and manage data from any system and from anywhere.

Salesforce works in Desktop, Mobile Devices, and iPad too.

Salesforce even have:



**Benefits of Salesforce**

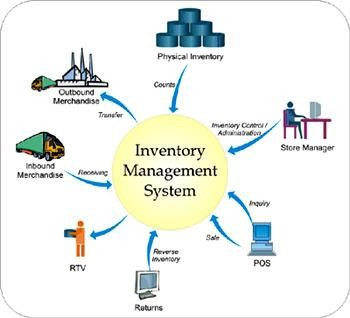
# Chapter 3: The Inventory Management Analysis Business Process (Former)

## **The Inventory Management**

Inventory Management is the process of arranging, storing, and ordering the items in the inventory. It is also the process of controlling and observing the products that are finished and available for sale. For a business, an inventory is the main asset which represents an investment by the owner until the item is sold.

Inventory Management helps us to provide production of an item, its sales, and provides a service level for the customer at low cost. The largest item for some of the companies is their inventories. If there is any problem with the inventory, then it will lead to the failure of the business. So, we must manage the items in the inventory very carefully.

Inventory control helps in reducing the inventory cost by increasing the chance of providing the product promptly to the customer. It is one of the parts of the Inventory Management.



### Inventory Management System

.

P a g e 17 | 50

## **The Inventory Management Manual Process**

With a manual inventory system, your team will manually update stock levels and inventory whereabouts. That means that as inventory comes and goes, your team will update the company’s inventory list by hand.

Alternatively, your company may not even track inventory perpetually. Instead, your business will rely on physical inventory counts to determine what is on hand, what is running low, and what needs to be reordered. If your team does not know what is on hand until they walk into a storage room and physically count it, it is considered physical manual inventory management.

These physical inventory counts could happen as needed or at regularly scheduled times. (No matter what kind of inventory system you use, your business will probably conduct some form of end-of-year inventory count, too.)

It is also possible—but tricky—for a business to conduct perpetual manual inventory. If your team constantly knows what is on hand in real time, but they’re *manually* updating an inventory sheet every time an item comes or goes, then it’s considered a perpetual manual inventory management system.

## **Benefits of a manual inventory system**

Manual inventory systems can be a good choice for very small businesses that do not carry much inventory. For example, if your company stocks just a few dozen products in one location, it can be easy to track inventory by hand as it comes and goes.

And if your team falls behind on keeping inventory, that may not affect your business too much. After all, your inventory is nearby—and conducting a physical inventory count only takes a few minutes.

A manual inventory system is also inexpensive—you can set one up on a sheet of paper or an inventory spreadsheet for free.

## **Drawbacks of a manual inventory system**

Unfortunately, a manual inventory system is quite vulnerable to human error. For example, a few jumbled numbers or data-entry errors can cause costly inventory shortages or stockouts that could affect your business’s bottom line.

Furthermore, a manual inventory system can quickly become unmanageable. That is because as your business grows, you will have more and more inventory to manage. Handling all those items manually can take up way too much of your and your employees’ valuable time—and eventually become so time-consuming that inventory tracking could slip through the cracks.

And even if you and your team maintain a manual inventory system as your business grows, time is a valuable resource. If it is taking hours to track inventory every day, that could be time and money better spent on marketing, hiring, or other business improvements.

## **Manual vs. Automated Inventory System**

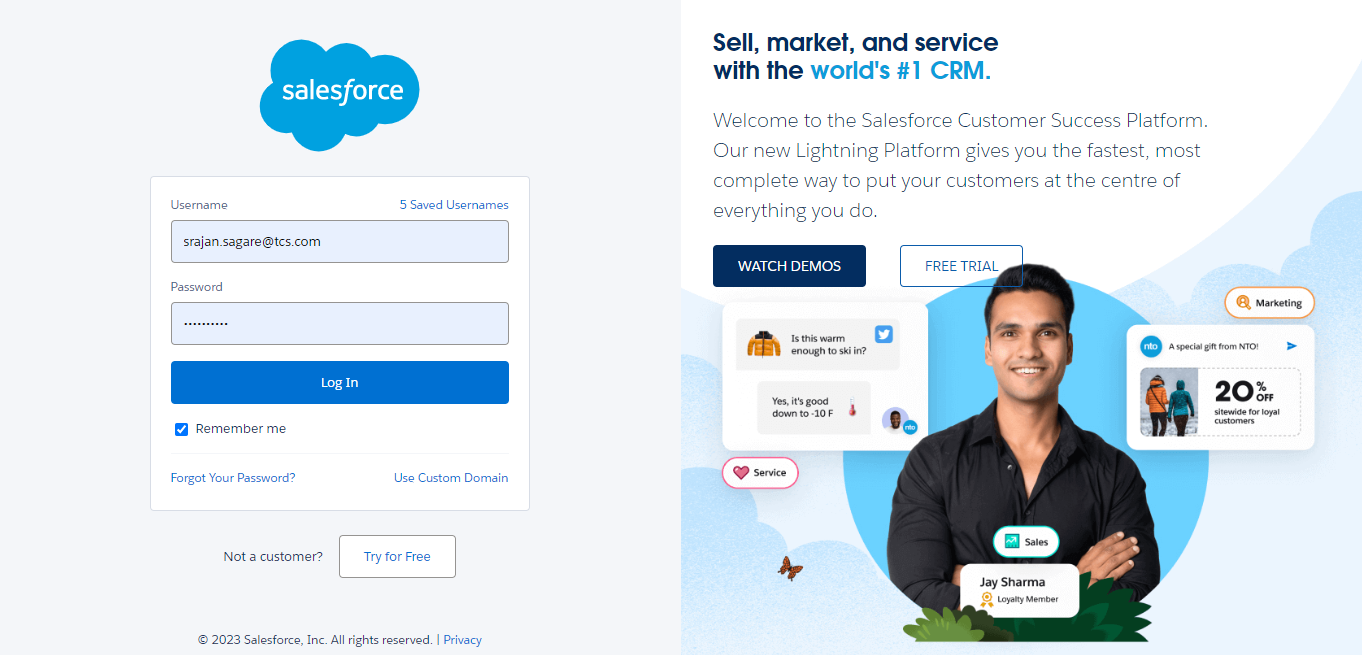
Data on the manual inventory is kept on inventory sheets, and each item is manually noted. Many still employ it today. The manual inventory system involves a person manually maintaining and updating each record, increasing the risk of human error. Furthermore, this system is also susceptible to data loss because inventory sheets can get torn up over time, lost, or replaced. In contrast, an automated inventory system is controlled and managed by a software. It is safe, more efficient, more secure, and almost impossible to lose data. A growing number of people are now using automated inventory systems to manage their inventory and data.

# Chapter 4: Inventory Management in Salesforce

## **Development Area**

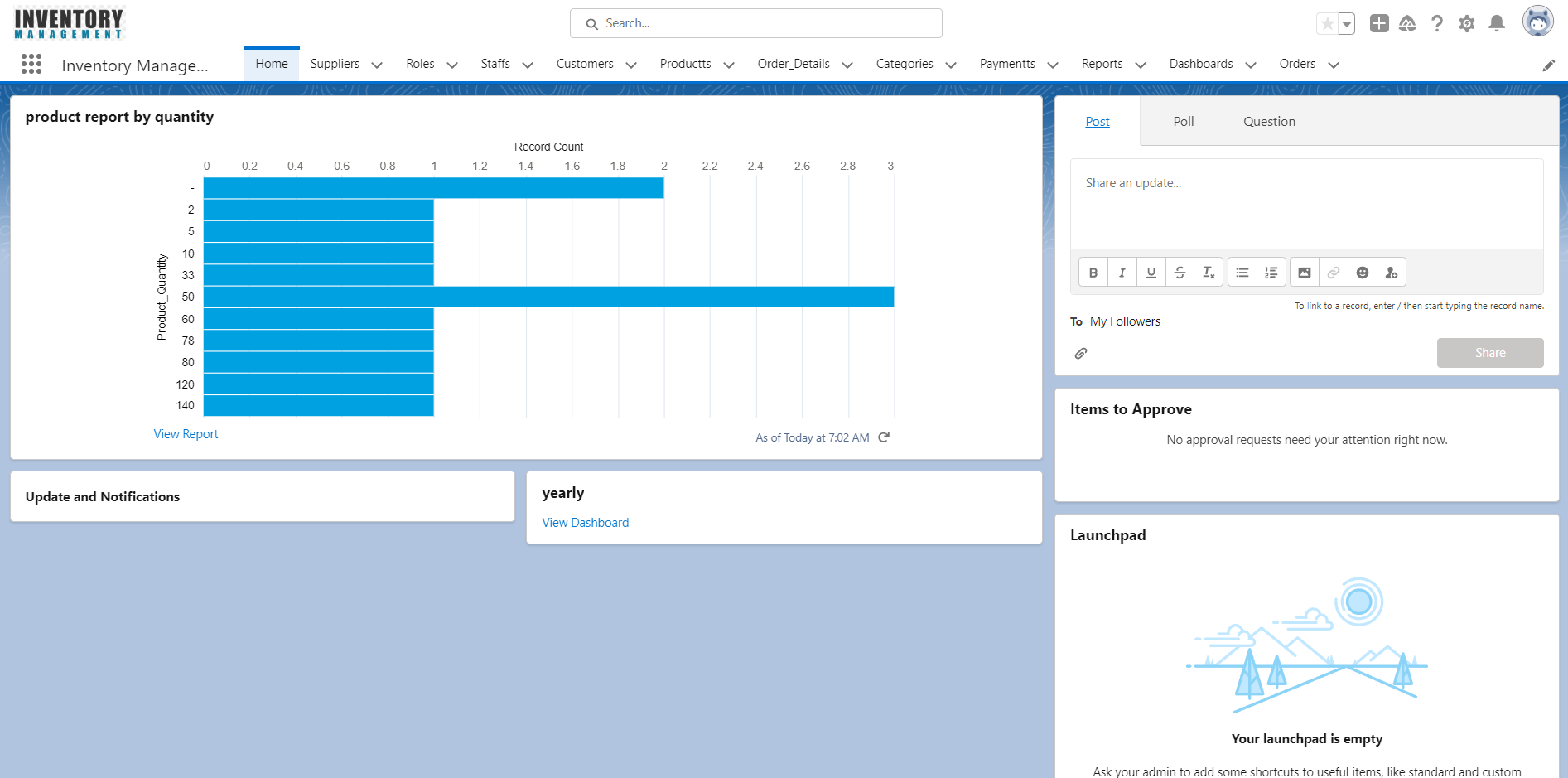
As Salesforce is a cloud-based platform, all it needs is a browser and internet connection. Once a developer or user of force.com platform logs in into salesforce.com, he sees his name on the top right corner of the page, and besides that, there is a setup button which we use mostly while developing an app.

* Open a browser and enter [www.login.salesforce.com](http://www.login.salesforce.com/)



* Enter Username and Password.

It will take us to the Salesforce customized company domain and it will look likes as shown below.



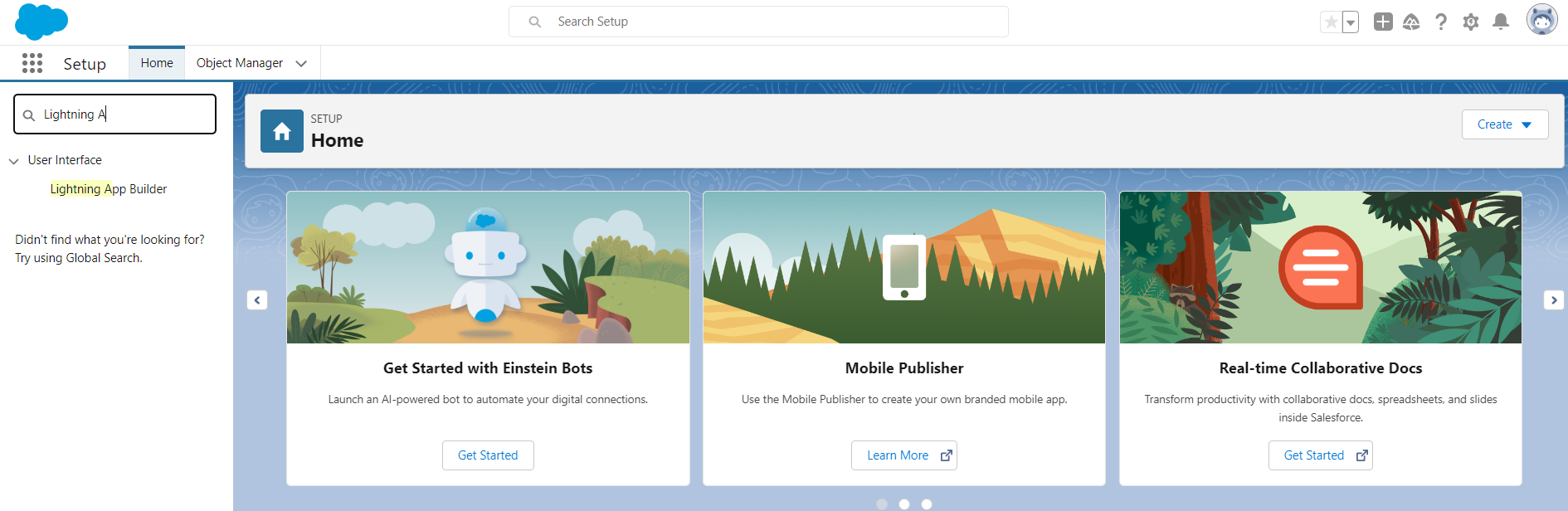
### **Salesforce Home Page**

After the Setup button in the header, we can see the Help button which will give detail information about anything in this platform and lastly, we can see a drop-down list showing Inventory Management, where we can see the list of apps available in this domain. By selecting the Inventory Management app, we can see all the data/ modules related to this app along with some predefined standard data.

### **Custom Apps**

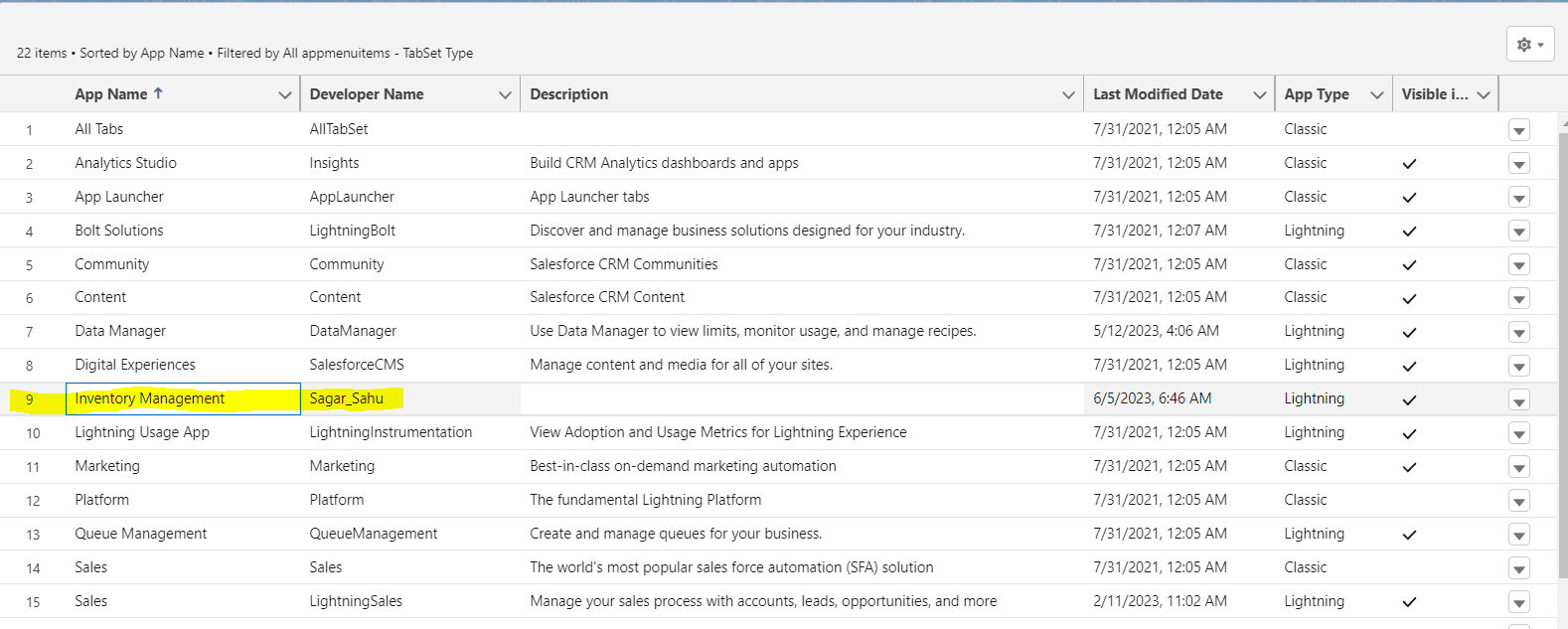
In Salesforce, creating an App is just a point and click away, to create a new app click on Setup and locate Create, then click on Apps and then on New. As simple as shown below, enter the name of the app as Inventory Management after clicking New.

Setup 🡪 Create 🡪 Apps 🡪 New



Salesforce provides some predefined apps like sales which consist of standard objects and standard fields. Now, we are creating our custom app which will consist of custom objects and custom fields to hold the data.





* + 1. **Objects**

Objects in Salesforce are database tables with information. The primary object in the Salesforce data model represents accounts/companies and organizations involved in the business, such as customers, partners, and competitors. A record is similar to a row in a database table.

Objects already created by Salesforce are called standard objects. Objects we create in organization are called custom objects

Created the custom object Brands in the following way,

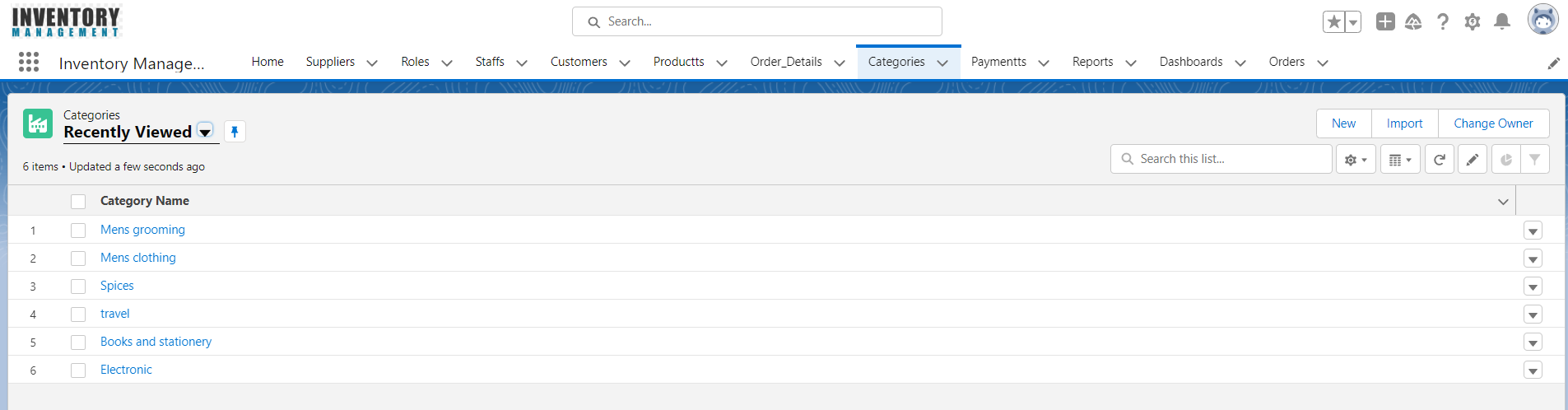
* Click on Create  Objects  Custom Object.
* Enter the Label name as Brand, Plural name as Brands which be used as Tab name and Object name as Brand which will be used as API name. By default, the custom objects are stored with c to access through API.
* Enter the record name Brand Title which is a mandatory field while creating a custom object and select its data type as Text.
* Next, check allow reports, activities and track field history check box fields, which are basically for availing report creation, assigning tasks to the user for a particular Brand record and tracking the history of all field edits and updates.

In similar fashion, For the Inventory Management App, six custom objects were created.

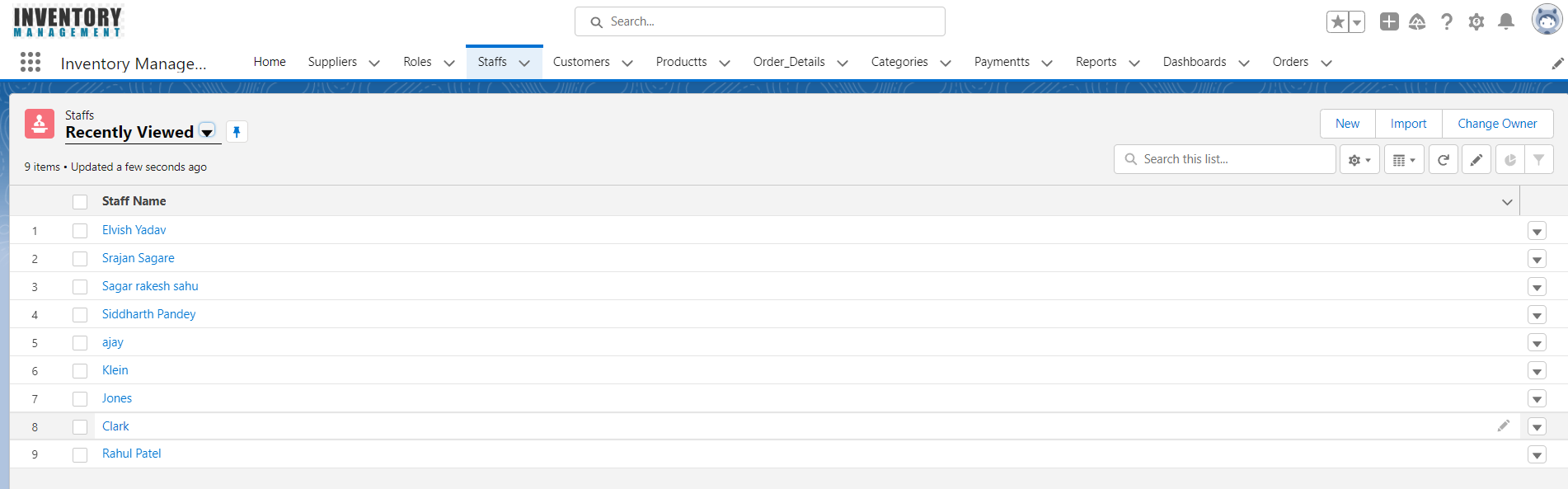
1. Supplier: Tracks all the details of Supplies of products in Inventory



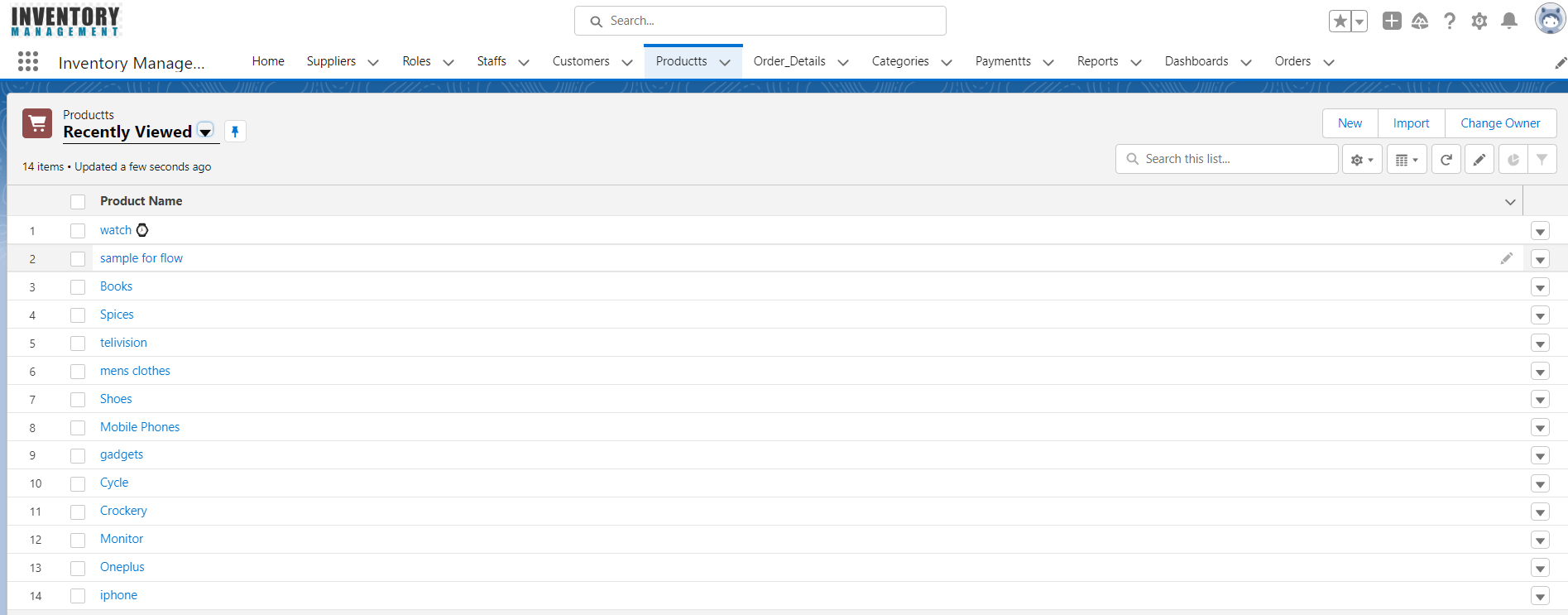
1. Category: Tracks the categories of products in the Inventory



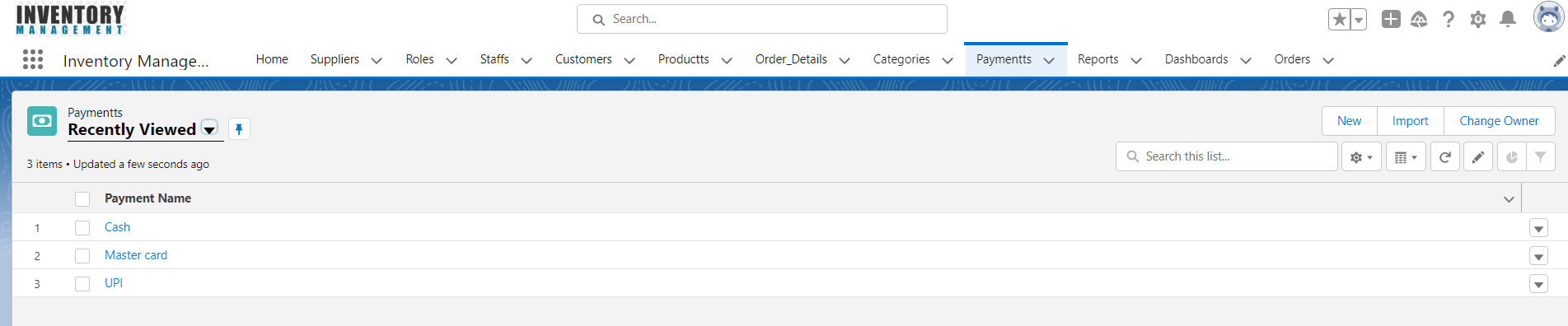
1. Staff: Tracks the Staff of the Organization.



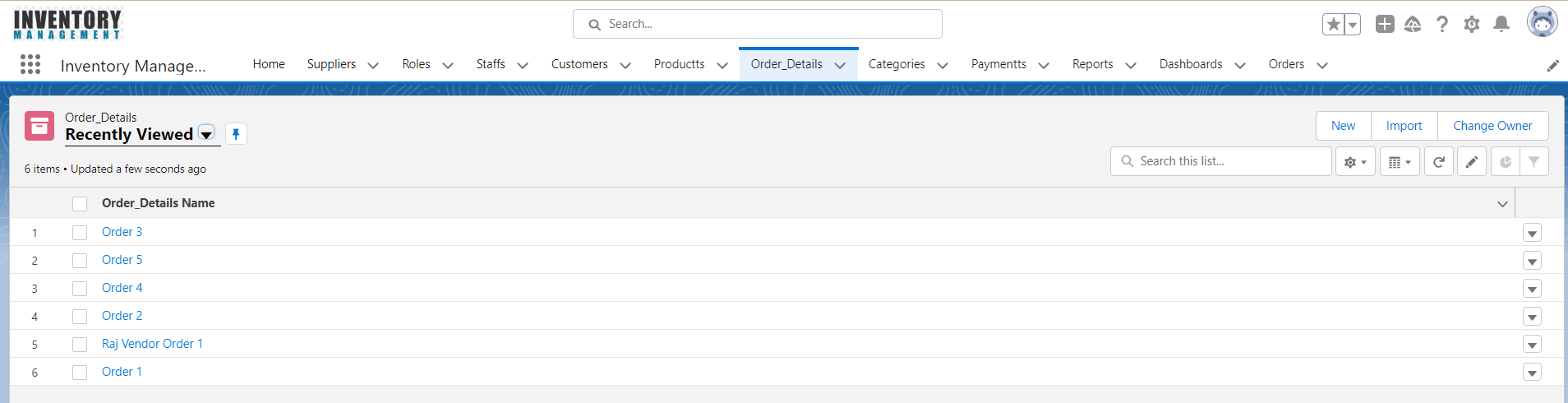
1. Products: Tracks the records of the products in the Inventory



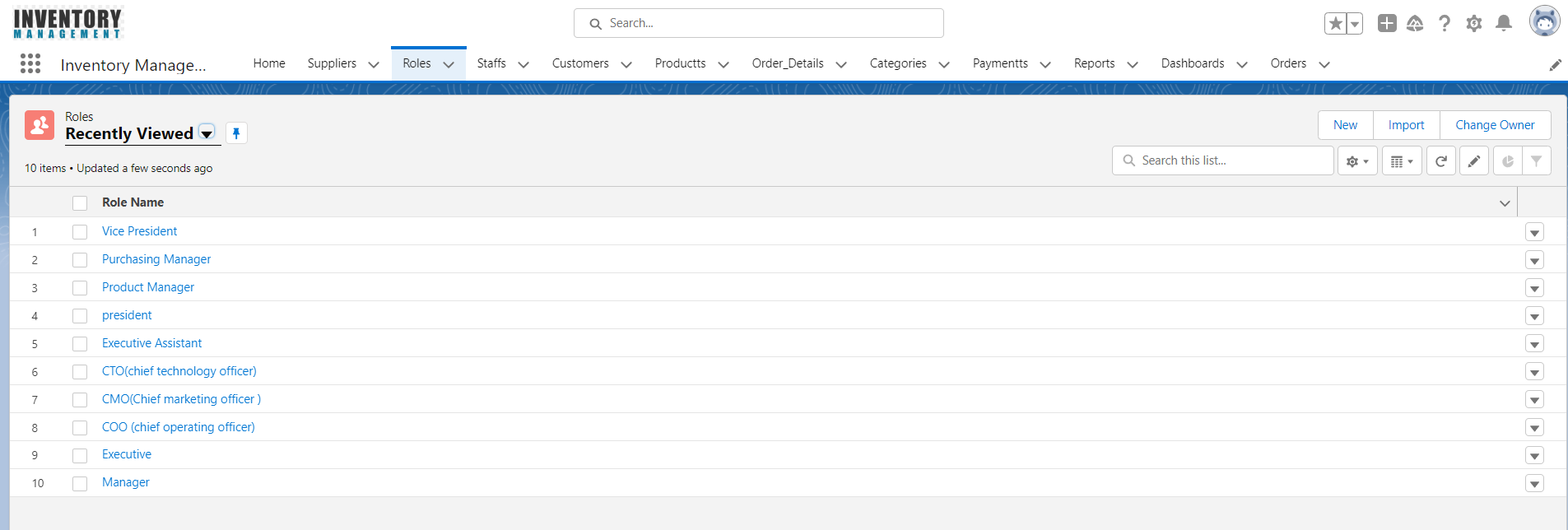
1. Order: Tracks the record of the Purchases at the point of sale.
2. Payments: Allows users to enter a dollar or other currency amount and automatically formats the field as a currency amount.



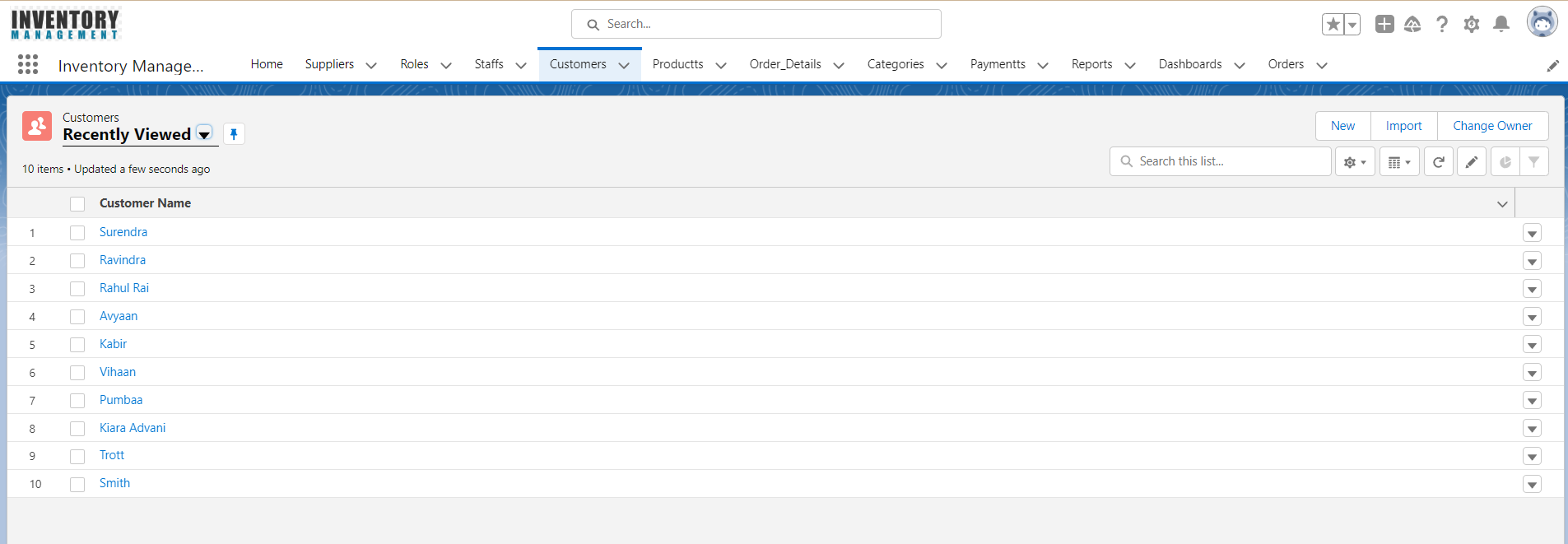
1. Order Details: Tracks the order information of the Customer at the point of purchase.



1. Roles: Tracks the employee working the inventory directly or indirectly.



1. Customer: All the people who places the order using Inventory Management App.



The objects can be accessed by tabs in salesforce.com and menu items as in the salesforce1 platform which is a mobile platform.

* + 1. **Fields**

Salesforce has provided many inbuilt fields called standard fields like “Name,” “Owner,”

Created By,” “Last Modified By” etc. Every object in Salesforce has a set of standard fields that may be applicable for capturing data for that type of an object. Salesforce also allows users to create new fields in the system to capture additional information. These fields are called as custom fields. Salesforce supports a variety of datatypes for these fields like Text, Text Area, Rich Text Area, Number, Currency, Boolean, Email, Phone, etc.

Administrators can then configure how these fields should be set (visible/read- only/edit/mandatory).

Created some more custom fields to the object **Brand** in the following way,

* Create 🡪 Objects 🡪 Brand 🡪 Custom Fields & Relationships.
  + 1. **Relationship**

The Force.com platform supports two types of parent-child relationships between objects. They are lookup relationships and master-detail relationships. These relationships connect objects with other objects. These both work like a foreign-key relationship in a relational database. These relations are the fields in the Salesforce. We can create them as we create custom fields.

The lookup relationship -creates a simple relation between two objects. With this relationship field, we can navigate from records in one object to the related records in another object and can create one-to-one and one-to-many relationships. Lookup relationships are appropriate when a relationship between two objects is required in some cases, but not always. In scenarios like to relate multiple parent records to the child record and to reference commonly shared data, such as reference data, a lookup relationship is used.

The master-detail relationship -is a powerful relationship which is based on a parent-child relationship. The object on which we create a master-detail relationship is the child, and the other one is the parent or master object which will be referenced as a field in the child object. The sharing settings of the child object can be taken from their master object. In a master-detail relationship, if we delete master object records, child objects also automatically deleted. With the master-detail relationship, we can create a roll-up summary field on parent objects.

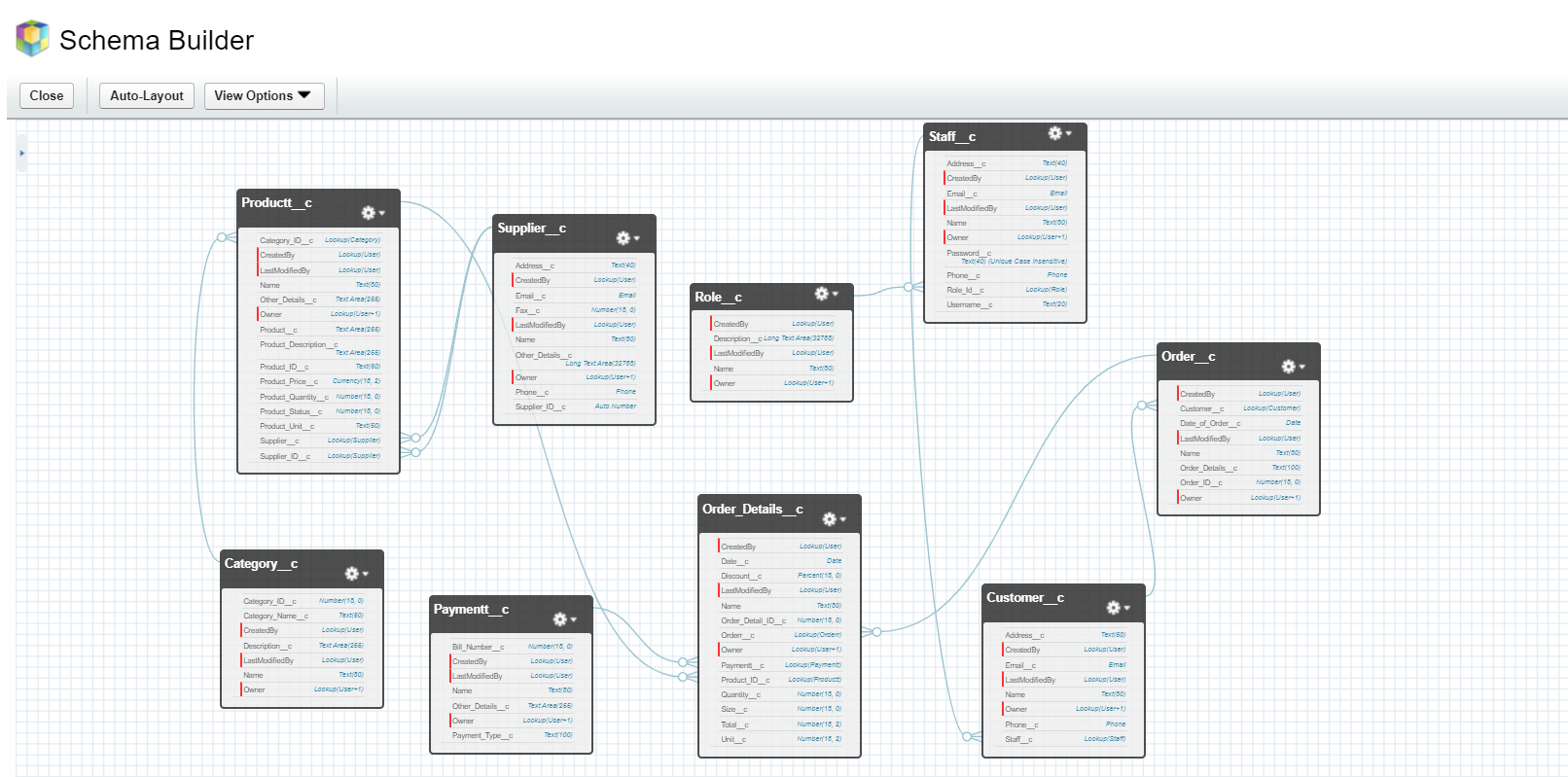
Example:

In Figure 11, charity is the master of events object and donation is the child object of events. So with the Master-detail relationship between them, we can see total money raised, in the charity object which is a roll up a summary of the total donations of the events object.



### An example of a Master-Detail relationship.

In the Inventory Management app, there is a master-detail relationship in between brand and manage product object and in between manage category and manage product objects. The manage product object is the parent of both manage category and brand objects. There is look up the relationship between the delivery order and contact object and then the contact object with a purchase order. The manage the product object have lookup relationship between the purchase order and manage supplier object.



**Relationship between Objects**

* + 1. **Tabs**

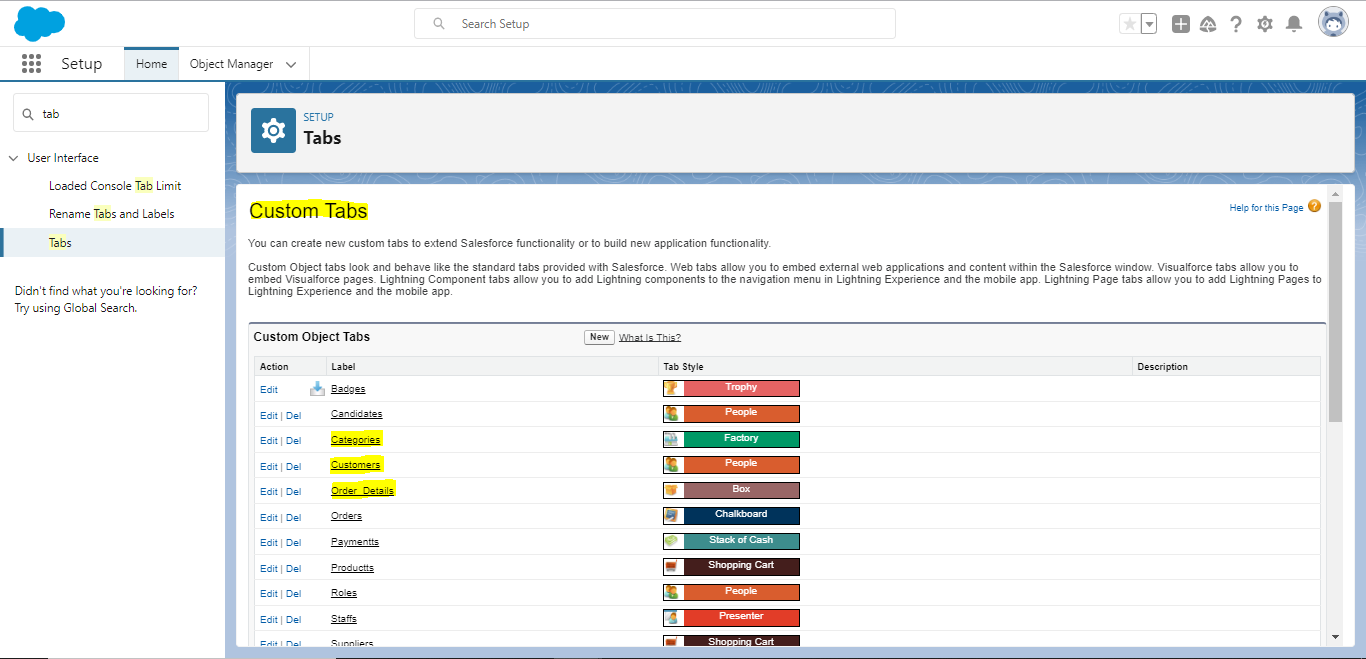
Tabs are the primary way to access the main objects in the application for users in Force.com platform apps. Just like standard objects/fields Salesforce also has the standard/custom tabs. Tabs provide an interface to access records for that particular object. Most standard objects have their tabs exposed. Users can also configure which tabs they would like to see and can rearrange their order. Users can create custom tabs for the custom objects they create. Tabs can also be used to open custom pages and links.

Created a custom Tab to the object **Brand** in the following way,

* Create  Tabs  Brands

The Custom Tabs that were created for the custom objects are Brands Tab, Categories Tab, Manage Stocks Tab, Manage Suppliers Tab, Products Tab, Purchase Orders Tab

The other Tabs that are used for visual force Pages Are Login Tab and Manage Product Tab



* + 1. **Securing and Sharing Data**

Salesforce provides a wide level of security and sharing of data in an organization. Some of them are listed below.

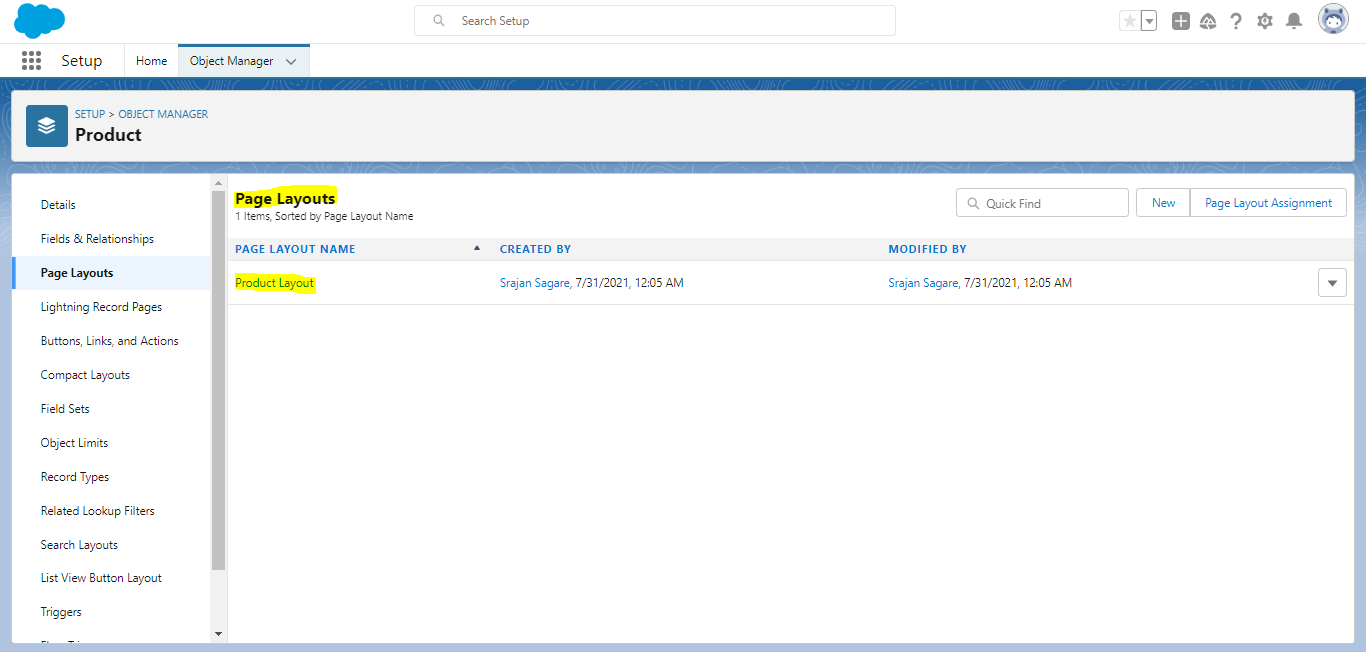
1. Object Level Access: These controls access to objects in the Organization.
2. Field Level Access: These controls access to fields in an object.
3. Record Level Access: These controls access to the records of an Object.
4. Profiles: Every user has one profile and this profile is used to control many things like user permissions, object permissions, field permissions, app settings, tab settings, Visualforce page access, Apex class access, page layouts, login hours, and login IP ranges and so on.
5. Permission Sets: Every user can have one profile but can have multiple permission sets. Permission sets to control access of object permissions, field permissions, user permissions, tab settings, app settings and it also controls Apex class access and Visualforce page access.
6. Field Level Permissions: These controls access of a user to view, edit and delete of fields of an object.
7. Organization-Wide Default Settings : These are default controls set by the organization and these controls access to the records of an object.
8. Role Hierarchies: With this, the user who is above in the role hierarchy will have access to the records of their subordinates.
9. Sharing rules: These controls access to the records of an object but these makes exceptions to the organization-wide default.
10. Manual sharing: It overrides the all the restrictions to access the records. One can share records with specific users.

The profile and the user permissions restrict what the user can see. The modifications made in the desktop Salesforce site are reflected in the app even and mobile users to get their organization’s data without special configuration.

* + 1. **Page Layout**

Page Layouts can be used to control the visibility of the custom fields, button and links on the Object record. We can rearrange the fields as per our convenience and by separate sections also. We can make a field to be visible, required or read-only field to control the access for different users, marks few fields as read only, mark few fields as mandatory, et**c**. Page layouts can be created in the following way,

Create 🡪 Objects 🡪 (Custom Object) 🡪 Page Layouts



* + 1. **Visualforce**

Visualforce is a complete framework for making such UIs, empowering any interface outline and communication to be constructed and conveyed altogether in the cloud. The UIs that work with Visualforce can expand the standard Force.com stage look and feel, or supplant it with an impressive style and set of sophisticated interactions. Since Visualforce markup is rendered into HTML, designers can utilize Visualforce tags along with standard HTML, JavaScript, Flash, or whatever other code that can execute inside an HTML page on Platform. Visual Force pages can enable in the mobile by just checking the “ enable the page for mobile apps” box field after creating it. With the CSS, the visual force pages can be optimized for the mobile devices and match the look of the Salesforce1.

For the Inventory Management app, more than 10 VF pages are created. The log in page, the home page, manage the product page, the insert product page, the invoice page and the payment page, are some of them.

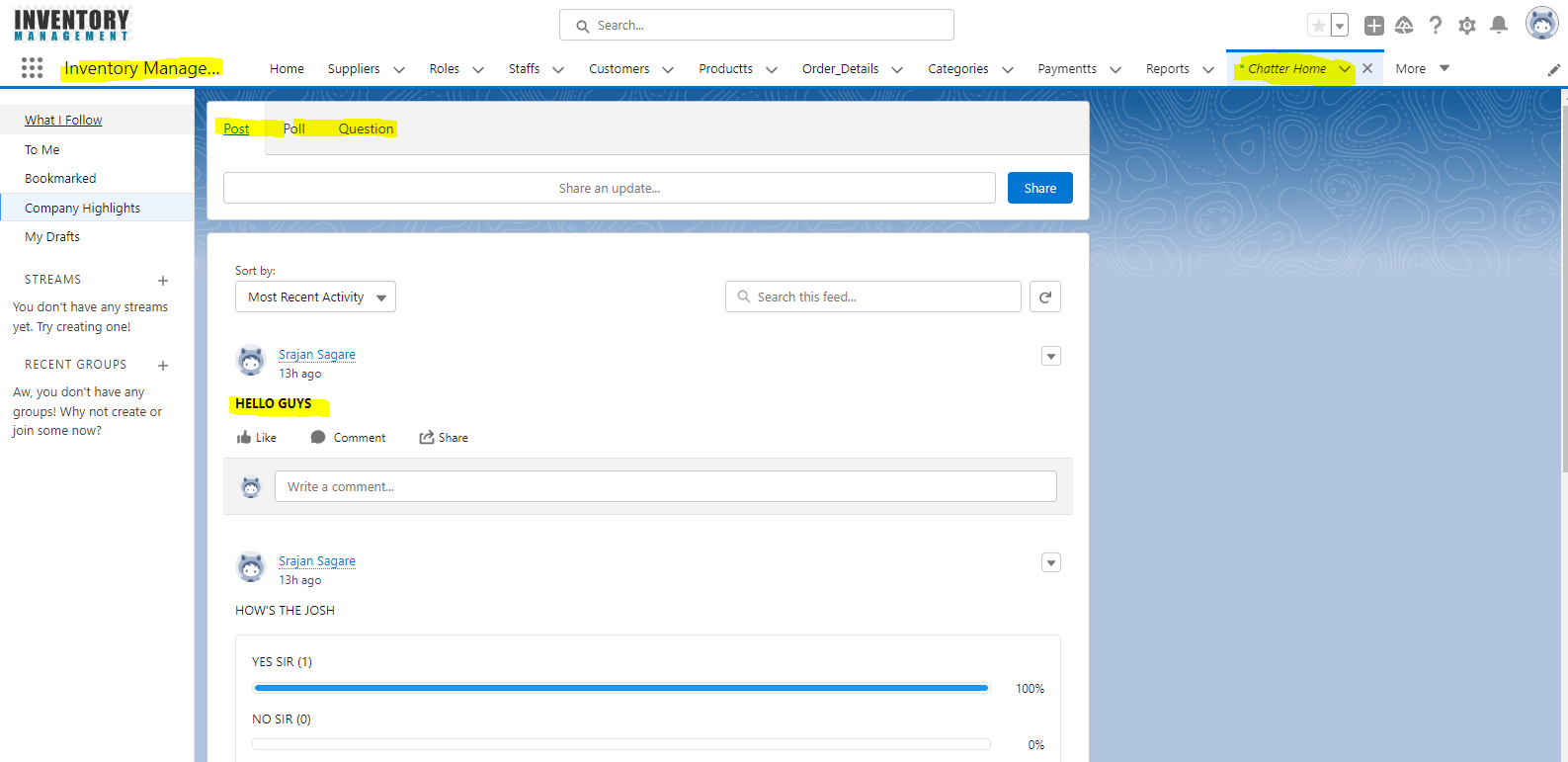
### **Apex**

Salesforce introduced Apex as the first cloud computing programming language. The syntax of Apex is quite like Java. It is particularly intended for building business applications to oversee Data and procedures larger ambiance of the Force.com platform. The Apex lets the developers focus just on elements specific to their application by providing a productive approach to creating functionality and logic, leaving the rest of work for Force.com Platform.

Usually, for every interactive VF page, Apex Class is associated. So more 10 Apex Classes are used in the Application.

### **4.1**.**10 Chatter**

Chatter is a standard Salesforce functionality which is very useful feed tracking system in Salesforce where users can post to feed, comment on feeds, shares the information, attach files and share. Also, the user can follow other users to get updates from that user. The user can follow a particular field of an object and can get updates whenever its value changed or updated. It reduces the mailing efforts for a user to update about things going on the organization. Also, create groups who are working on an assignment and keep an update or follow up with them by enabling chatter. It is like a social network application for the work environment to connect people with updates going in their organization.



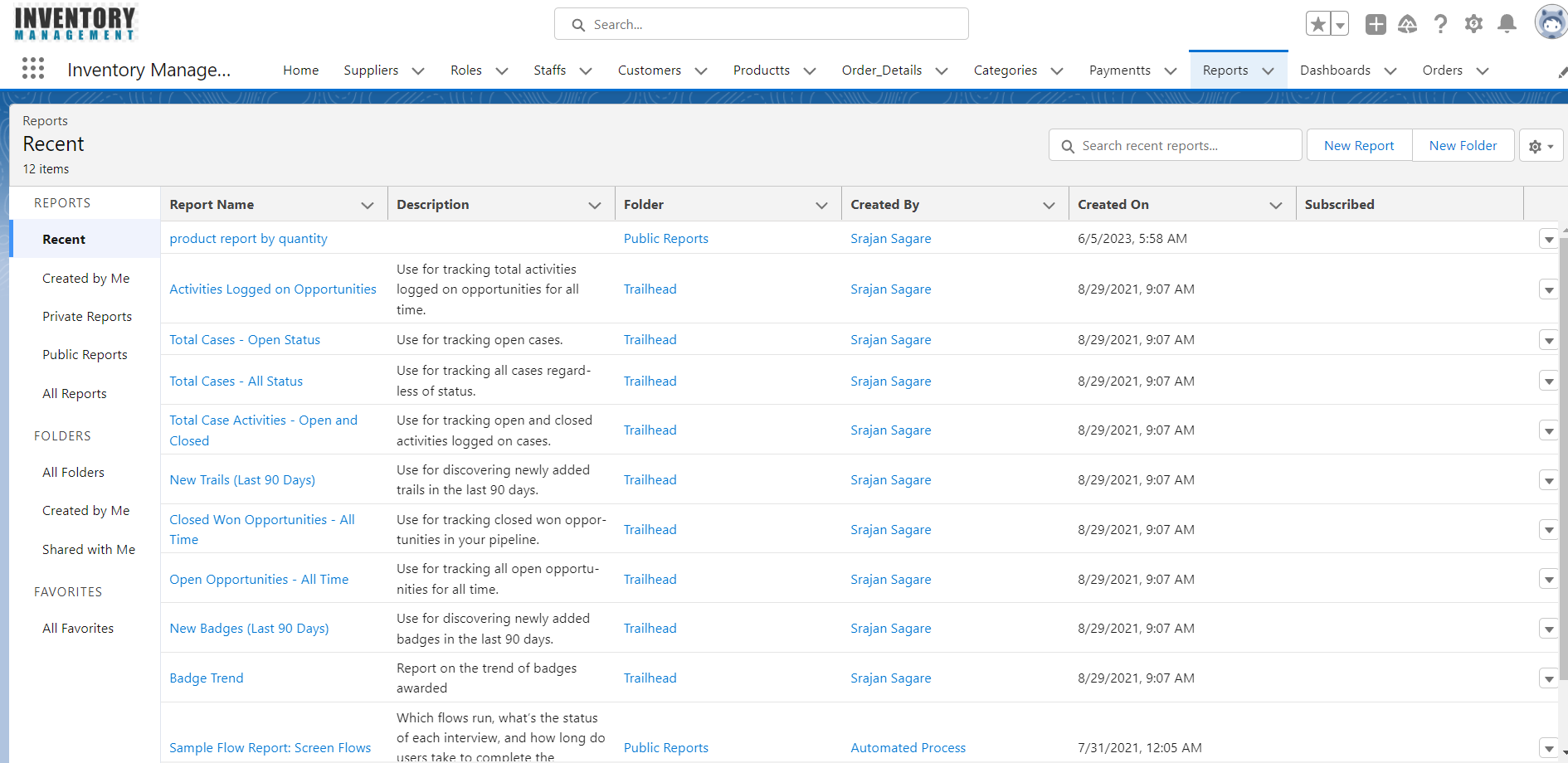
### **4.1.11** **Reports and Dashboards**

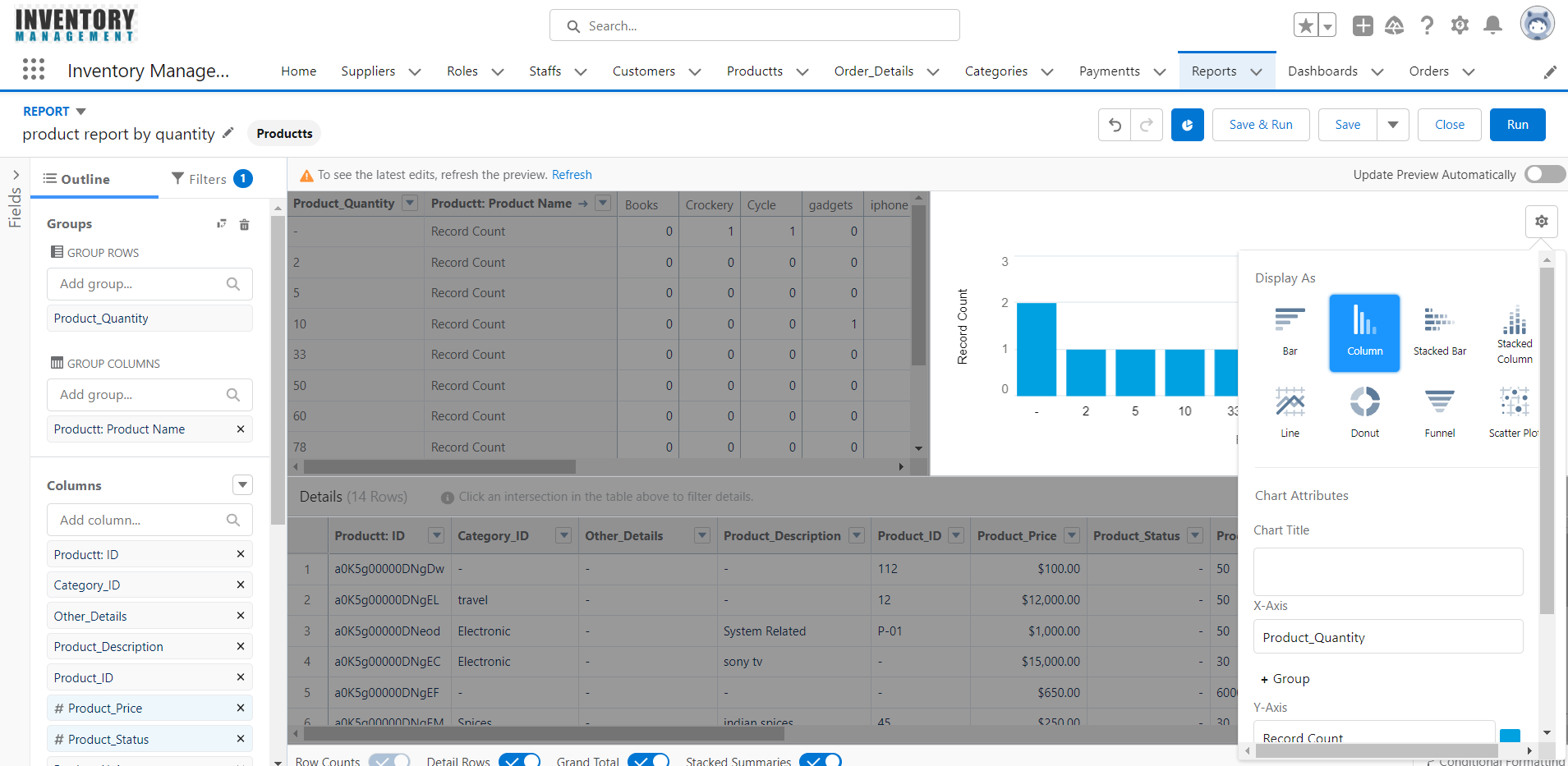
As we know a company deals with lots of data, to check the data record by record is a very lengthy and time taking process. A good application should provide the overview of the data briefly. For this purpose, Salesforce provided a tool called Reports. Using reports, we can generate different formats of data resides in the company at a glance view. A dashboard is a place where we can put all reports together to give the overview of the statistics. For example, if the manager wants to know how many products are sold from the inventory, he can create reports and find these statistics. Salesforce supports four different formats of Reports which are,

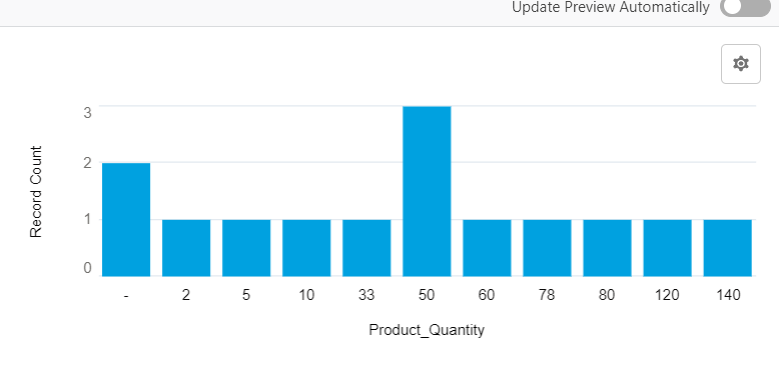
* Tabular Reports: These give a simple list view of data with Grand totals. But these are not used to create groups of data and graphs.
* Summary Reports: These give the summary of the data like subtotals with row-wise also. These are used to create charts, graphs. These reports can be used in the dashboard.
* Matrix Reports: These give the groups of data both by row wise and column wise. These are the most time consuming to generate a report or to set up, but can also be used in the dashboard.
* Joined Reports: These allow to join the multiple reports of different types. These reports can also be used to create charts so that can be used in dashboards.

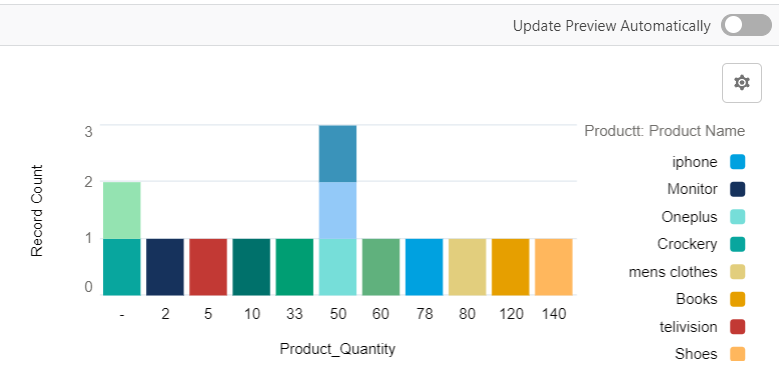
The Reports can be created as follows,

Create 🡪 Apps 🡪 Add Reports Tab 🡪 Reports.



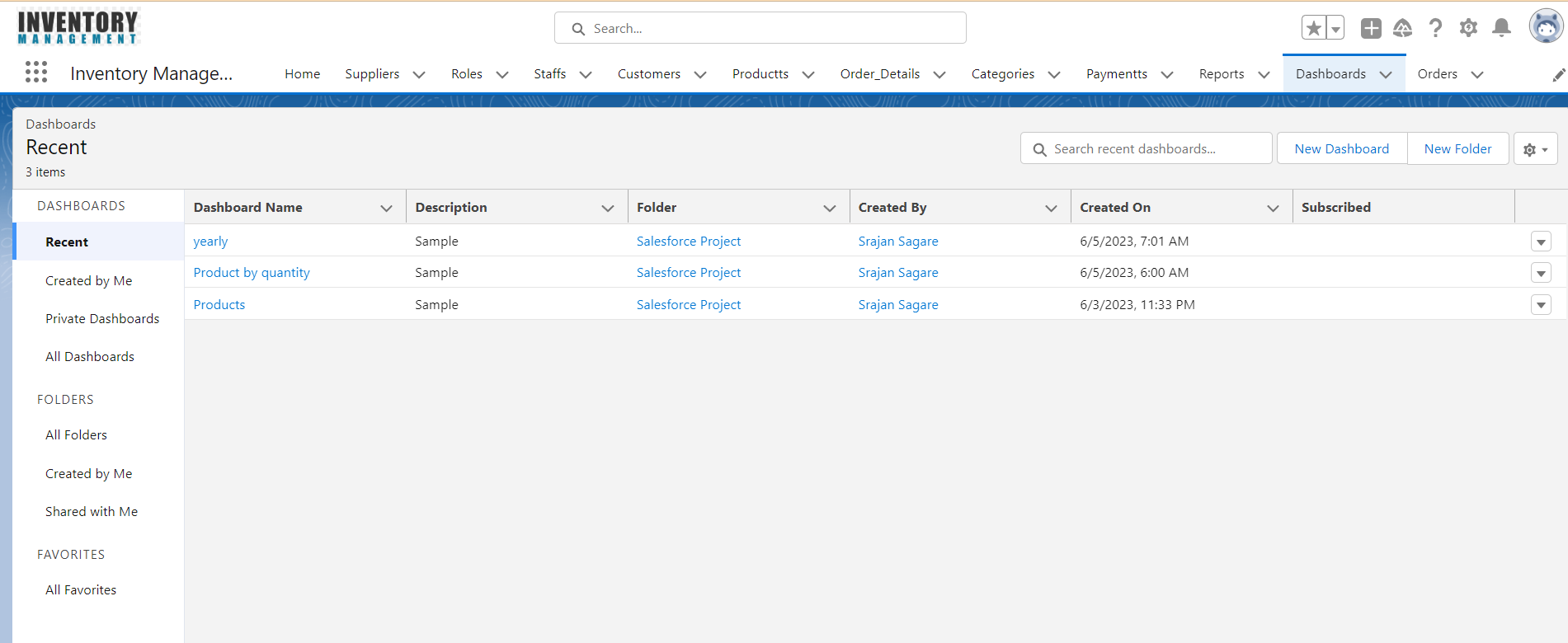


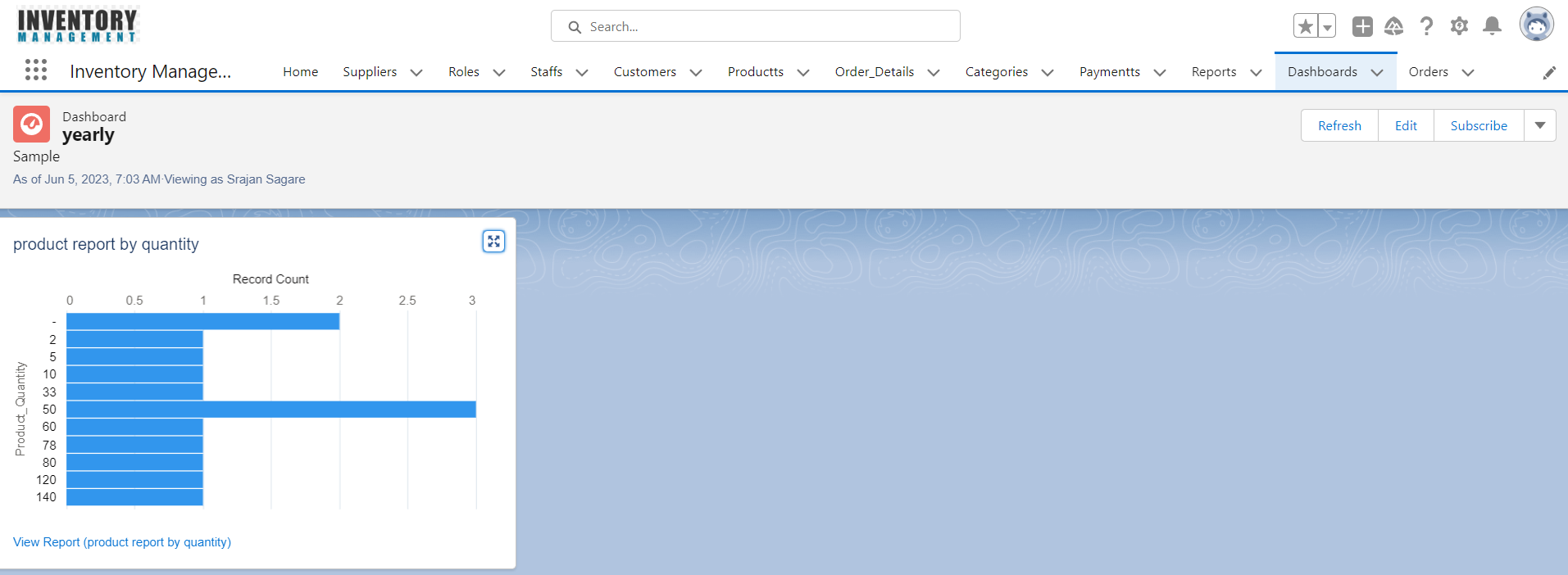




A Dashboard can show the reports in a visual format like Charts, Gauges, Tables, Metrics, or Visualforce pages. Force.com platform supports 20 components to be allowed in a dashboard for an organization. For example, the reports which we created before, like Pie chart and Vertical bar chart we can put up to 20 components in a dashboard. For the dashboard also we created a folder and saved all the dashboards under Inventory Dashboards folder and shared same as Reports folder. Dashboard can be created as,

Reports Tab 🡪Reports & Dashboards 🡪 New Dashboard





# 4.2 Business Process before Salesforce

## **4.2.1 Traditional Method**

Before the implementation of Salesforce, businesses typically relied on manual processes and traditional methods for managing various aspects of their operations. Here are some common business processes that were often handled differently before the advent of Salesforce:

1. Customer Relationship Management (CRM): Managing customer relationships involved manual methods such as maintaining paper-based contact lists, using spreadsheets or legacy systems, and relying on manual data entry and tracking. Customer interactions, sales leads, and account management were often handled through fragmented systems or disparate tools.
2. Sales and Opportunity Management: Tracking sales leads, managing opportunities, and forecasting sales were typically done using spreadsheets, standalone sales tools, or legacy systems. Sales teams would manually update and share information, making collaboration and real-time visibility challenging.
3. Lead Generation and Tracking: Lead generation relied heavily on traditional methods such as cold calling, trade shows, or manual referral systems. Tracking and managing leads involved manually entering information into spreadsheets or databases, making it difficult to effectively nurture and convert leads into sales.
4. Marketing and Campaign Management: Marketing campaigns and promotions were executed through offline channels such as print media, direct mail, or telemarketing. Tracking campaign effectiveness, measuring ROI, and analyzing customer response required manual data collection and analysis.
5. Customer Support and Service: Customer support and service processes were often managed through phone calls, email exchanges, or ticketing systems. Tracking and resolving customer issues, managing service requests, and maintaining a comprehensive view of customer interactions were time-consuming and prone to manual errors.
6. Data Reporting and Analytics: Gathering, consolidating, and analyzing business data involved manual extraction from various sources, followed by spreadsheet-based analysis. Generating reports, forecasting, and gaining actionable insights required substantial time and effort.

# 4.3 Revolution after Salesforce

Salesforce revolutionized these business processes by providing a centralized, cloud-based platform that integrated various functionalities into a single system. It offered features like:

* CRM capabilities for managing customer relationships, sales pipelines, and opportunities.
* Lead management tools for tracking and nurturing leads, improving conversion rates.
* Marketing automation functionalities for executing targeted campaigns and measuring campaign performance.
* Service and support features for managing customer inquiries, case resolution, and self-service options.
* Robust reporting and analytics capabilities for real-time data visibility, insights, and performance tracking.

Salesforce's cloud-based model also brought the advantages of scalability, accessibility, and collaboration, allowing businesses to streamline operations, improve efficiency, enhance customer experiences, and make data-driven decisions.Top of Form

# Sagar

# Chapter 5: Salesforce Selection Methodology

## **5.1 What is Project Methodology**

Methodology in project management refers to a systematic and structured approach that outlines the processes, practices, and guidelines to be followed throughout the project lifecycle. It provides a framework for planning, executing, monitoring, and controlling projects to ensure successful outcomes.

A project methodology typically encompasses the following key aspects:

1. Project Initiation: This phase involves defining the project objectives, identifying stakeholders, conducting feasibility studies, and preparing a business case. It sets the foundation for the project and clarifies its scope, goals, and deliverables.
2. Project Planning: In this phase, the project team develops a comprehensive project plan that includes defining project tasks, estimating resources, establishing timelines, and creating a project schedule. Risk management, communication plans, and stakeholder engagement strategies are also addressed during this stage.
3. Project Execution: The execution phase involves carrying out the project plan, assigning tasks to team members, and implementing project activities. It includes coordinating resources, monitoring progress, and addressing issues and changes that may arise. Regular communication, collaboration, and documentation are essential during this phase.
4. Project Monitoring and Control: This phase focuses on tracking project progress, assessing performance, and ensuring adherence to the project plan. It involves monitoring project metrics, identifying deviations, and taking corrective actions to keep the project on track. Regular status updates, reporting, and milestone reviews are performed to evaluate project health.
5. Project Closure: The closure phase involves wrapping up project activities, conducting final reviews, and transitioning deliverables to the stakeholders or the operational team. Lessons learned and best practices are documented to facilitate future projects. Closure activities also include finalizing financials, conducting project post-mortems, and celebrating project success.

## **5.2 Use of Selection Methodology**

The Salesforce selection methodology is used to ensure that the organization chooses the most appropriate Salesforce solution and implementation partner to meet its CRM (customer relationship management) needs effectively. Here are the reasons why organizations use Salesforce selection methodology:

1. Identify Requirements: The methodology helps organizations define their CRM requirements and objectives clearly. It involves conducting needs assessments, engaging stakeholders, and understanding pain points and challenges within existing CRM systems. This step ensures that the selected Salesforce solution addresses specific business needs and aligns with the organization's goals.
2. Evaluate Options: The methodology provides a structured approach to evaluate different Salesforce offerings and implementation partners. It involves researching Salesforce products, assessing vendor capabilities, conducting demos, and reviewing case studies. This evaluation process helps organizations compare features, functionalities, pricing, and implementation methodologies to identify the best fit for their requirements.
3. Mitigate Risks: By following a structured methodology, organizations can mitigate risks associated with selecting and implementing Salesforce. It ensures that thorough research is conducted, references are checked, and proof of concept or pilot projects are executed. This helps validate the effectiveness and feasibility of Salesforce in addressing the organization's CRM needs before committing to a full-scale implementation.
4. Optimize ROI: The Salesforce selection methodology aims to maximize the return on investment (ROI) for the organization. By thoroughly evaluating costs, considering the total cost of ownership, and assessing the scalability and future growth potential of Salesforce, organizations can choose a solution that aligns with their budget and delivers long-term value.
5. Ensure User Adoption: One critical aspect of the selection methodology is to involve key stakeholders and end-users in the decision-making process. By conducting stakeholder interviews, workshops, and gathering user input, organizations can ensure that the selected Salesforce solution is user-friendly, aligns with existing workflows, and promotes user adoption. This increases the chances of successful implementation and widespread user acceptance.
6. Streamline Implementation: Following a structured selection methodology helps organizations streamline the implementation process. By selecting a reputable implementation partner, negotiating contract terms, and defining project milestones and timelines, organizations can ensure a smooth and well-managed implementation journey. This reduces the risk of delays, budget overruns, and implementation challenges.
7. Achieve Business Objectives: Ultimately, the Salesforce selection methodology is used to help organizations achieve their CRM-related business objectives. By aligning the chosen Salesforce solution with specific requirements, addressing pain points, improving data management, and enhancing customer relationships, organizations can leverage Salesforce to drive growth, improve operational efficiency, and deliver better customer experiences.

By using the Salesforce selection methodology, organizations can make informed decisions, reduce risks, optimize ROI, ensure user adoption, streamline implementation, and achieve their CRM goals effectively. It provides a structured and systematic approach to selecting the right Salesforce solution and implementation partner, setting the foundation for a successful CRM transformation.

Top of Form

## **5.3 Selecting the most Appropriate Salesforce Solution**

The selection methodology for Salesforce involves a systematic approach to evaluating and selecting the most appropriate Salesforce solution for an organization's needs. Here are the key steps typically followed in the Salesforce selection process:

1. Define Requirements: Start by clearly defining the organization's requirements and objectives for implementing Salesforce. This includes understanding the specific business processes, functionalities, and goals that need to be supported by the Salesforce solution. Engage stakeholders from different departments to gather comprehensive input.
2. Conduct Needs Assessment: Conduct a thorough needs assessment to identify pain points, gaps in existing systems, and areas where Salesforce can provide value. Evaluate the current systems, data integration requirements, scalability needs, security considerations, and user adoption factors. This assessment will help align the selection process with the organization's specific needs.
3. Research Salesforce Offerings: Familiarize yourself with the different Salesforce products and editions available, such as Salesforce Sales Cloud, Service Cloud, Marketing Cloud, and others. Understand the features, capabilities, and pricing models associated with each offering. Evaluate how well each product aligns with your organization's requirements and objectives.
4. Vendor Evaluation: Research and evaluate Salesforce implementation partners or consulting firms that can assist with the selection and implementation process. Consider factors such as expertise, experience, industry knowledge, customer reviews, and service offerings. Select a partner that can provide guidance and support throughout the selection and implementation journey.
5. Request for Proposal (RFP): Prepare an RFP or Request for Information (RFI) document that outlines your organization's requirements, goals, and expectations. Share this document with potential Salesforce implementation partners to gather their proposals and recommendations. This step helps in comparing vendor capabilities, pricing, and proposed solutions.
6. Demo and Proof of Concept: Shortlist a few Salesforce implementation partners based on their proposals and conduct demos or proof of concept (POC) sessions. During these sessions, vendors showcase how Salesforce can address your organization's specific requirements. This allows you to assess the user interface, functionality, ease of use, customization options, and integration capabilities of the solution.
7. Reference Checks and Case Studies: Request references from shortlisted vendors and reach out to their existing clients to gather feedback on their experiences. Additionally, review case studies or success stories that highlight similar implementations and the resulting benefits. This step helps validate the credibility and track record of the implementation partners.
8. Evaluate Total Cost of Ownership (TCO): Consider the upfront costs, licensing fees, implementation fees, customization costs, ongoing maintenance, and support costs associated with the Salesforce solution. Assess the TCO over a reasonable timeframe to ensure it aligns with your budget and provides a satisfactory return on investment (ROI).
9. Decision-Making and Contract Negotiation: Evaluate all the information gathered during the selection process and make an informed decision based on the best fit for your organization. Negotiate contract terms, pricing, and implementation timelines with the selected implementation partner. Ensure that the contract includes the agreed-upon scope, deliverables, support, and any other essential terms.
10. Implementation Planning: Once the selection is finalized, collaborate with the implementation partner to create an implementation plan. This plan should include project milestones, timelines, resource allocation, data migration strategies, training requirements, and change management considerations.

By following these steps, organizations can select the most suitable Salesforce solution and implementation partner, ensuring a successful implementation that aligns with their specific requirements and goals.

# Chapter 6: Inventory Management Implementation in Salesforce

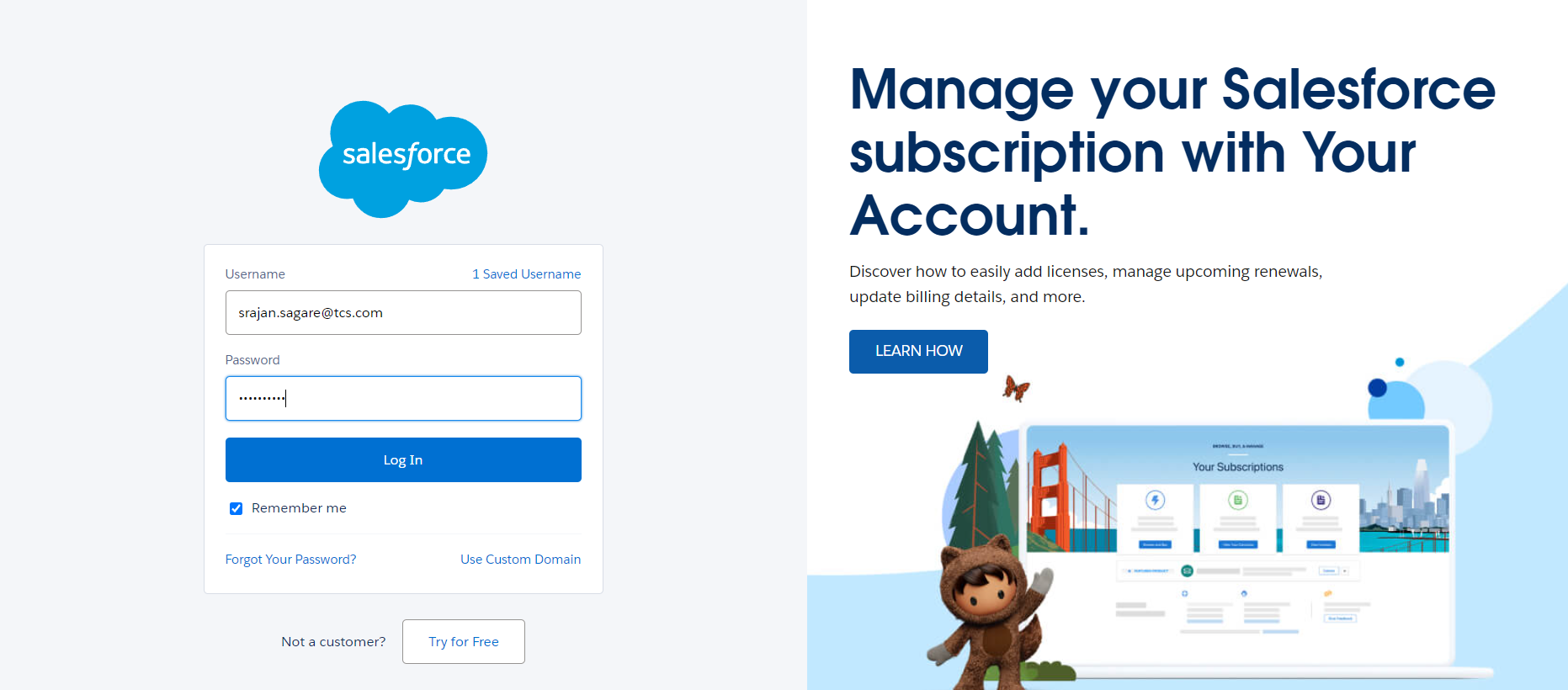
### **6.1 Inventory Administration**

Inventory administration is carried out usually by store manager or store admin.

The Manager can do the following functions.

The Store Manager can add, edit, delete and view the products in the database. The various functions that a manager can do include updating new brands and suppliers, generating reports (based on the category, brand, created date) and the access to see and edit the customer details and all purchase orders details.

To administrate his inventory, Manager needs to log into Salesforce with his credentials.

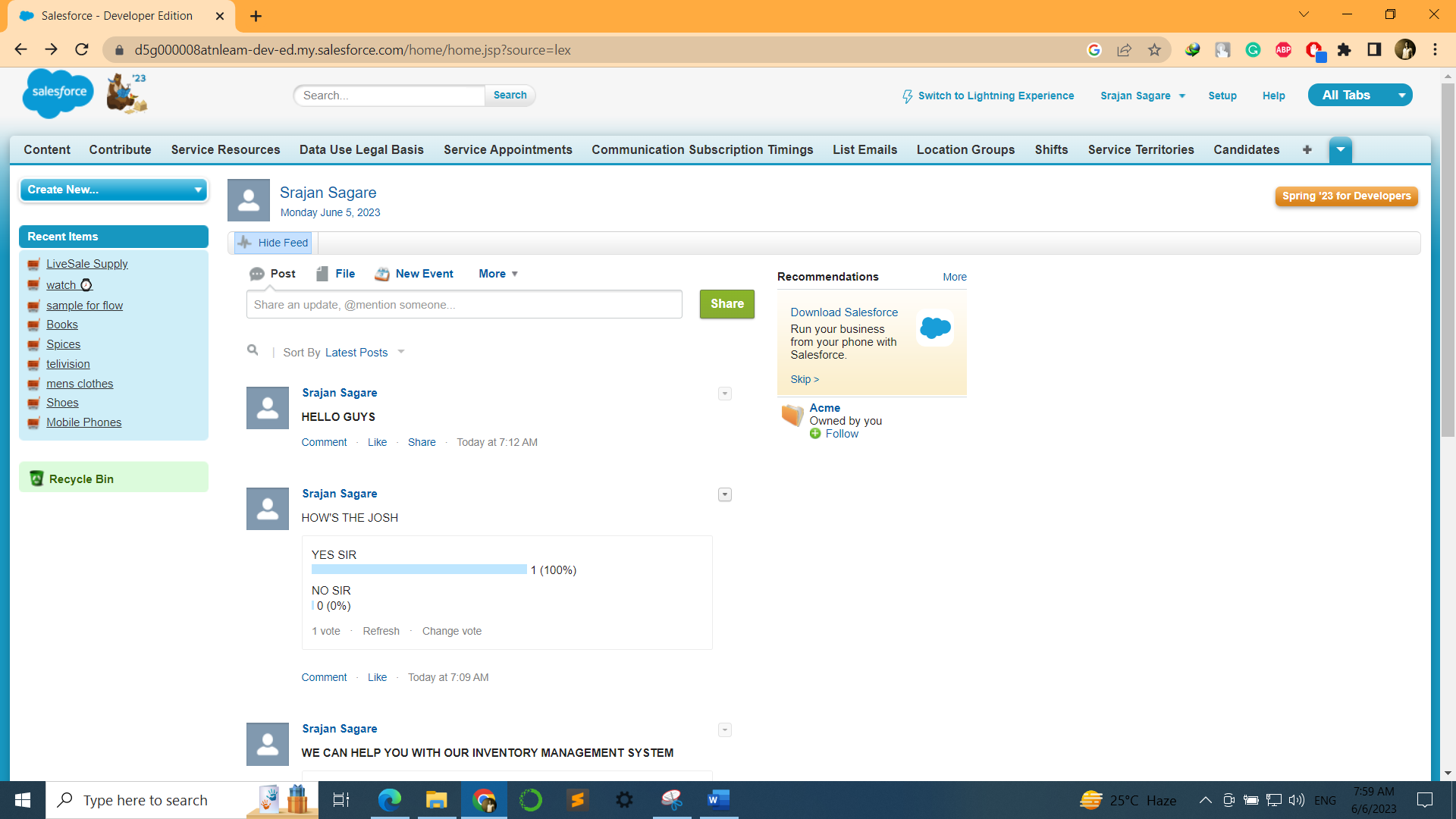


Once the admin logs into the Salesforce, the log in page is redirected to home page. In the home page, the manager finds Standard Page components like apps, tabs, menu and he will also find custom modules like calendar, task to perform, etc., In the home page, the manager can navigate between all the tabs, existing applications, and other components.

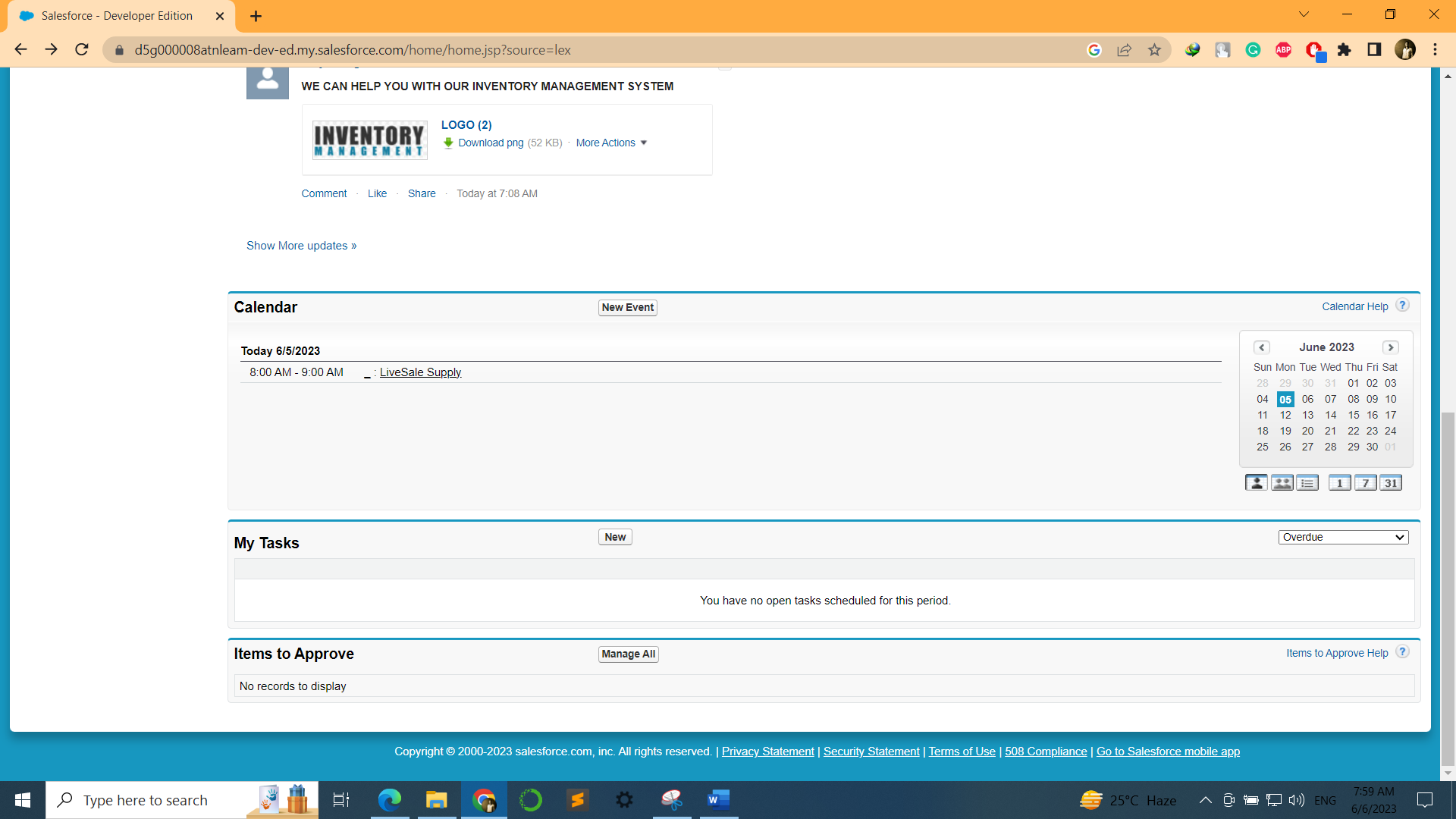
Home Page can be customized depending on the requirements. The layout of the page is fixed while the components like tabs can be customized. In the inventory management app, home page is customized.

In the customized home page, manager can view

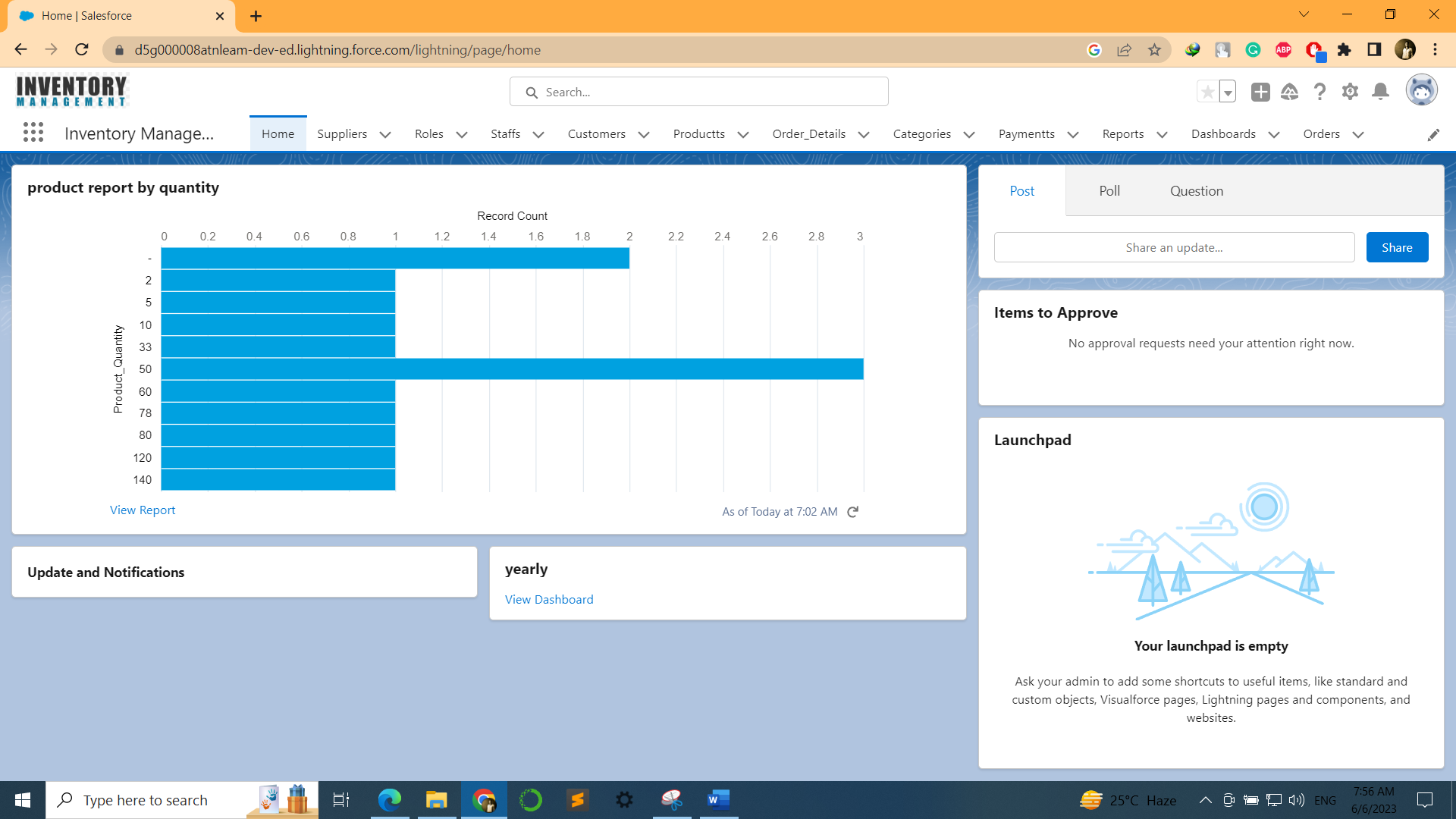
1. Calendar,
2. Tasks to perform,
3. Dashboards
4. Record need his approval
5. Recent Activities



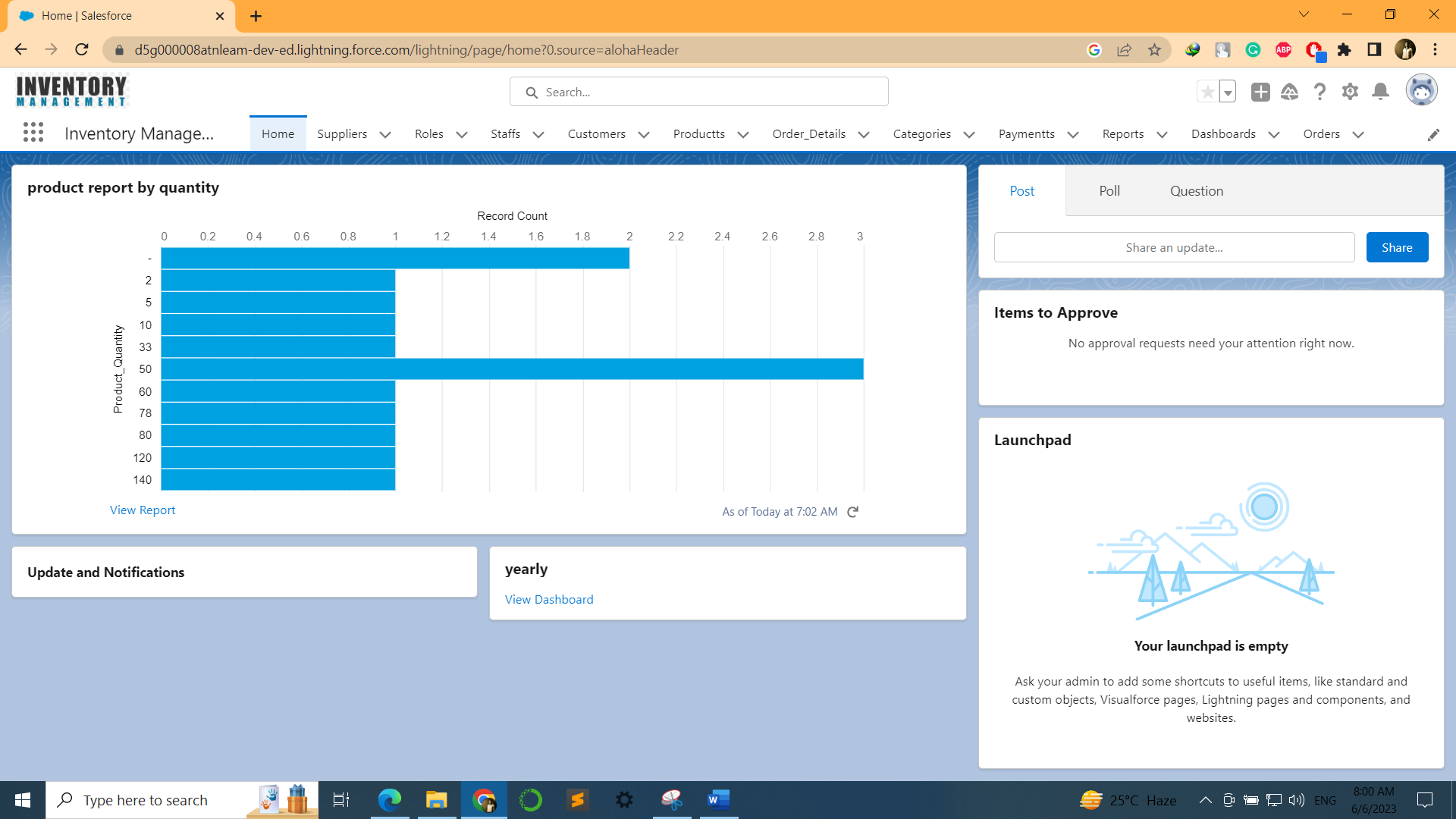
**Home Page of Our Application**



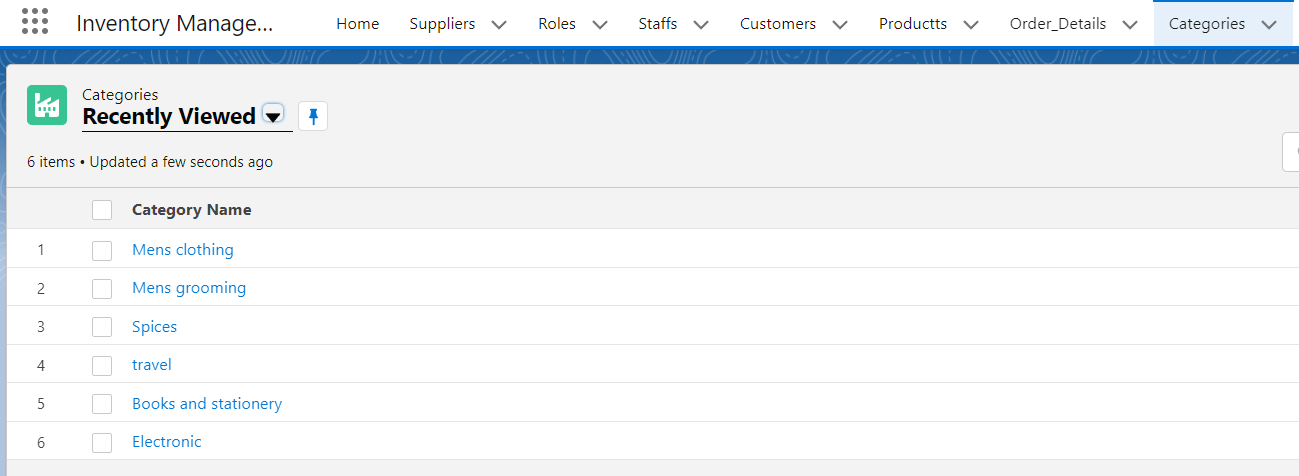
In addition to standard page components like Tabs, Apps, etc.



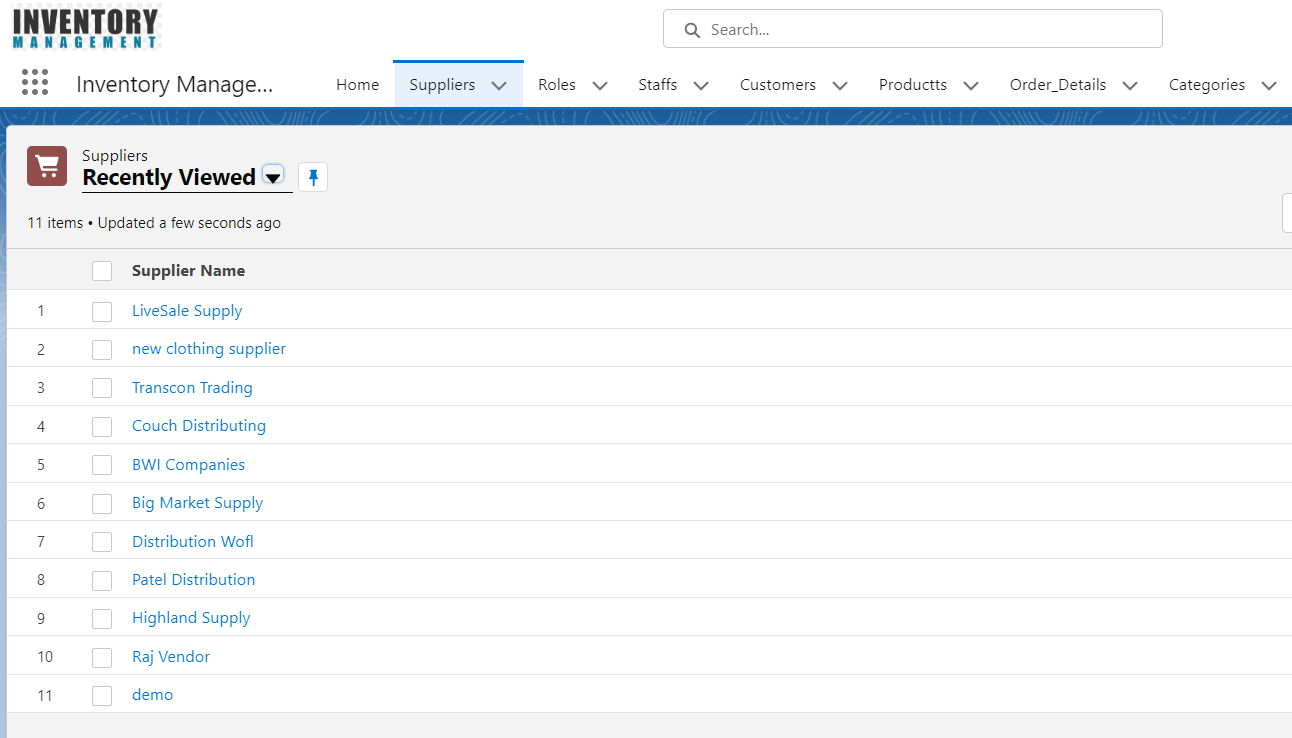
Brands tab has the database of product inventory tagged under various brands. The manager has access to create, delete and view the brands available in inventory. The feature to arrange the brands by filters like recently viewed, recently created and recently modified is also available.



Similarly, the categories tab has provision to create, view and delete the categories of products. The sorting filters like recently viewed, recently created and recently modified can be used for sorting of categories.

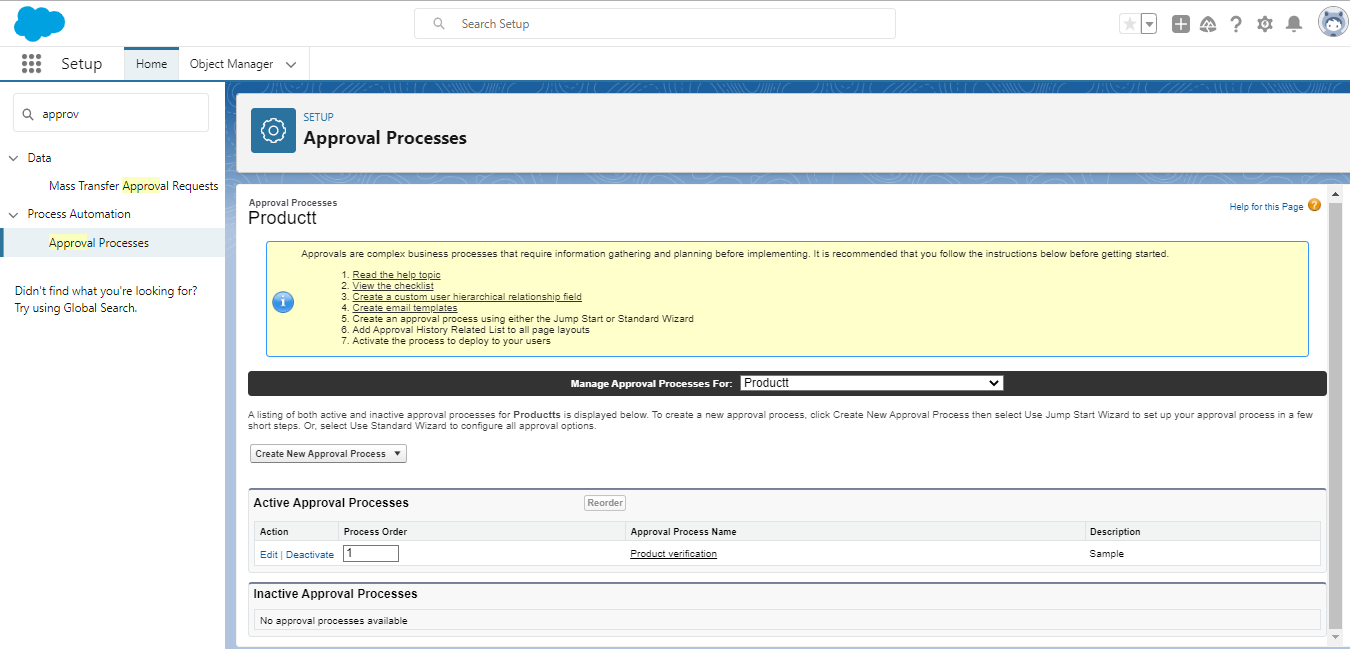


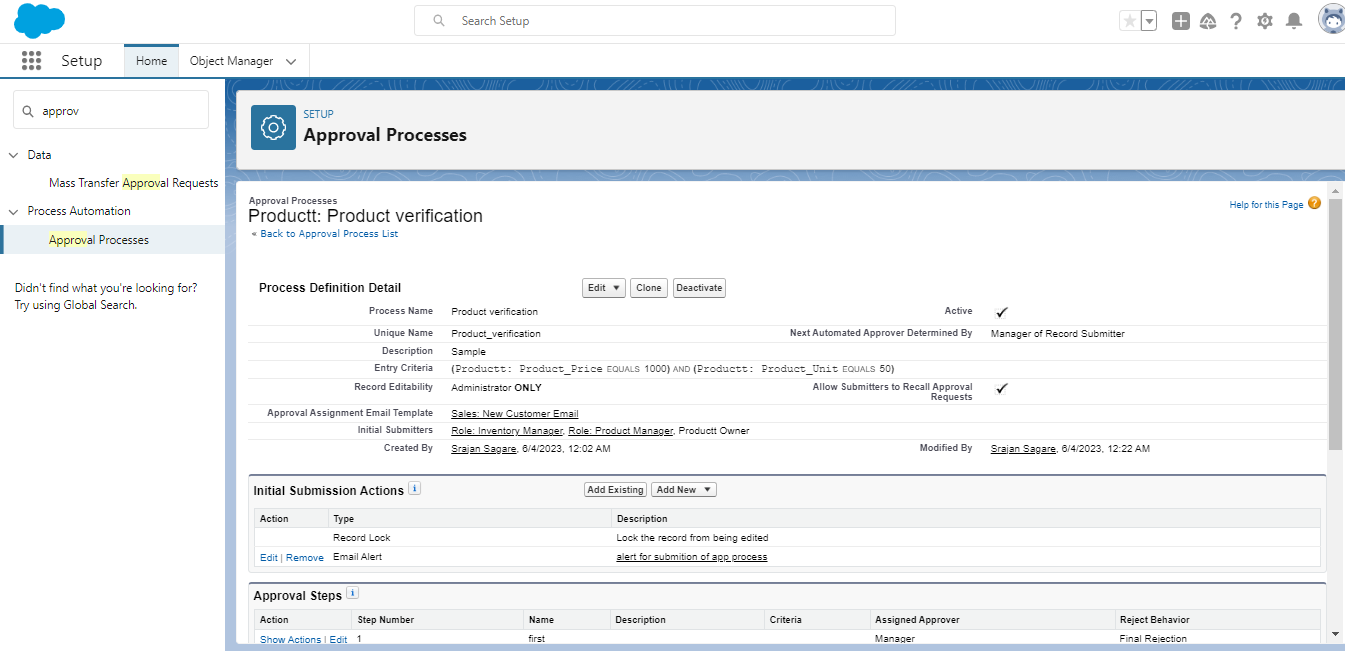
The manage suppliers tab has access to add, view and delete the suppliers for products. Sorting filters like recently viewed recently created and recently modified are provided for this tab also.



Only the store manager has master access to the database. Input fed by employees is only reflected in the inventory database on approval from a manager. The employee has to raise an Approval Process.

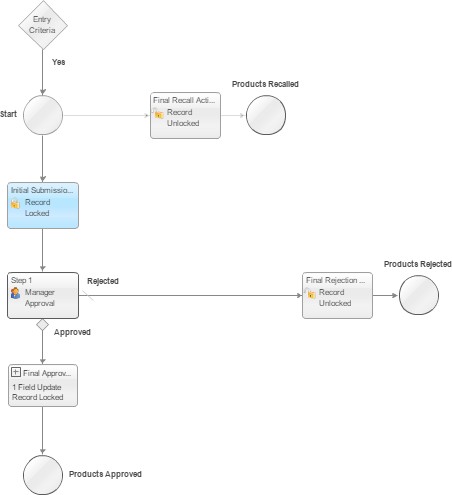
Salesforce App 1 View:





### 6.1.1 Approval Process

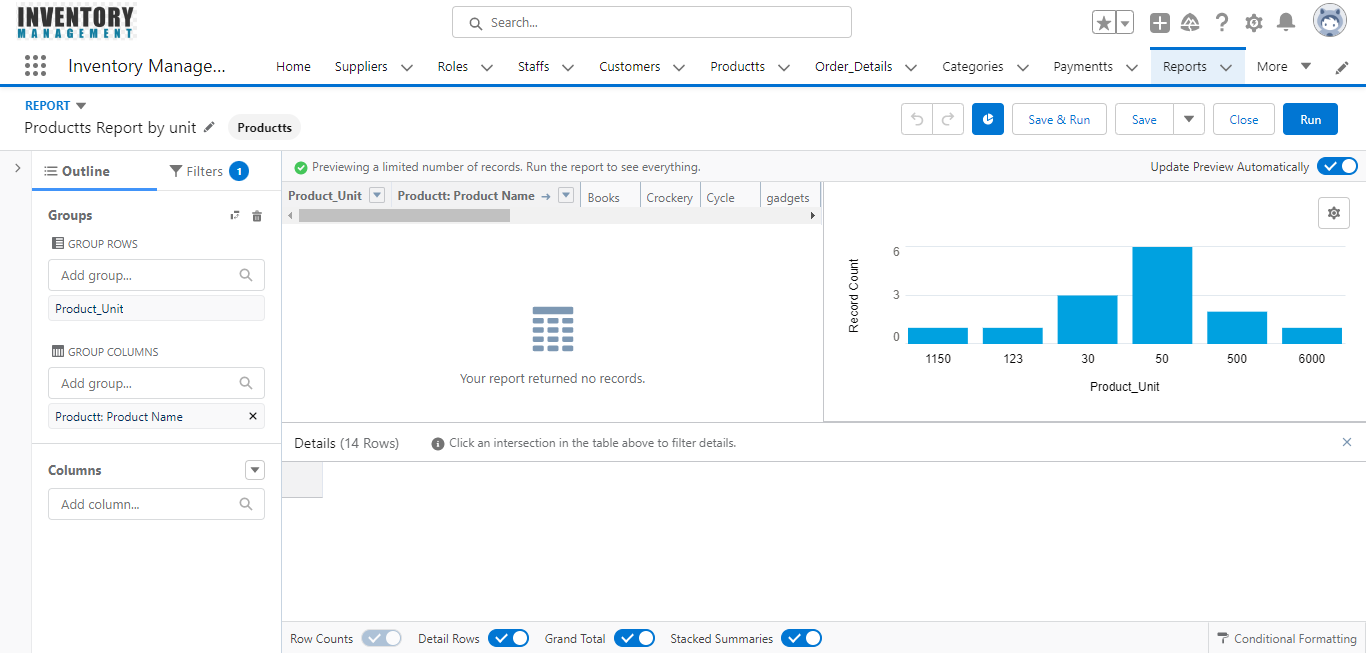
Every product must get the approval of store manager before it goes into the inventory. Below is the approval process flow.



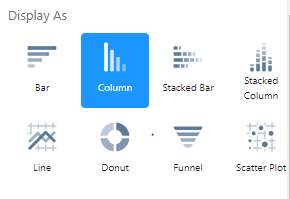
### Product Approval Process

The store manager can generate reports based on Product Report by Unit

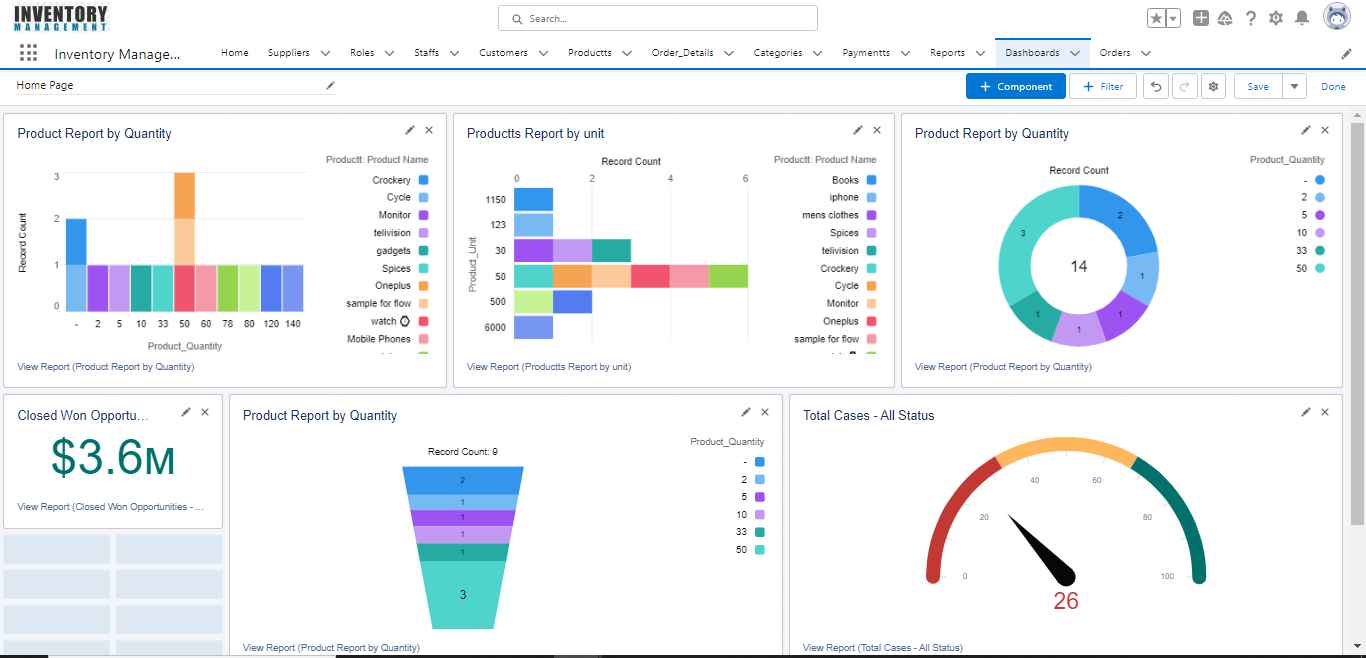
* Categories
* Product
* Time
* Products sold



Data Generated from the reports can be viewed in dashboards in formats like



* Horizontal bar chart
* Vertical bar chart
* Line chart
* Pie chart
* Donut chart and
* Funnel chart



**6.1.2 Workflow Rules:** Workflow Rule is an automated process that enables you to standardize and automate standard internal procedures and processes to save time across your org. It allows you to define a set of criteria and associated actions that are triggered when the criteria are met.

Here are the key components and functionalities of a Workflow Rule:

1. **Criteria**: Workflow Rules are triggered based on defined criteria, typically based on the field values of a record. You can specify criteria using formula expressions or simple comparisons, such as "if the Opportunity Stage is 'Closed Won'."
2. **Triggering Events**: Workflow Rules can be triggered based on different events, such as when a record is created, edited, or when a specific date or time-based trigger is met. For example, you can define a rule to send an email when an Opportunity Stage is changed to 'Closed Lost.'
3. **Actions**: Workflow Rules can perform various actions when the criteria are met. Common actions include updating field values, creating tasks, sending email alerts, creating or updating records, or triggering outbound messages. For example, you can update a custom field or assign a task to a user when certain criteria are met.
4. **Rule Evaluation Criteria**: Workflow Rules can be set to evaluate criteria either when a record is created, when a record is edited and meets the criteria, or every time a record is edited, regardless of whether it meets the criteria.
5. **Rule Order and Dependencies**: You can define the order of execution for multiple Workflow Rules on an object and specify dependencies between them to control the sequence of actions.

Workflow Rules provide a declarative way to automate business processes in Salesforce without writing code. They are primarily used for simple automation needs, such as field updates, email notifications, and basic record creation or updates. For more complex automation requirements, you may consider using Process Builder or Apex triggers.

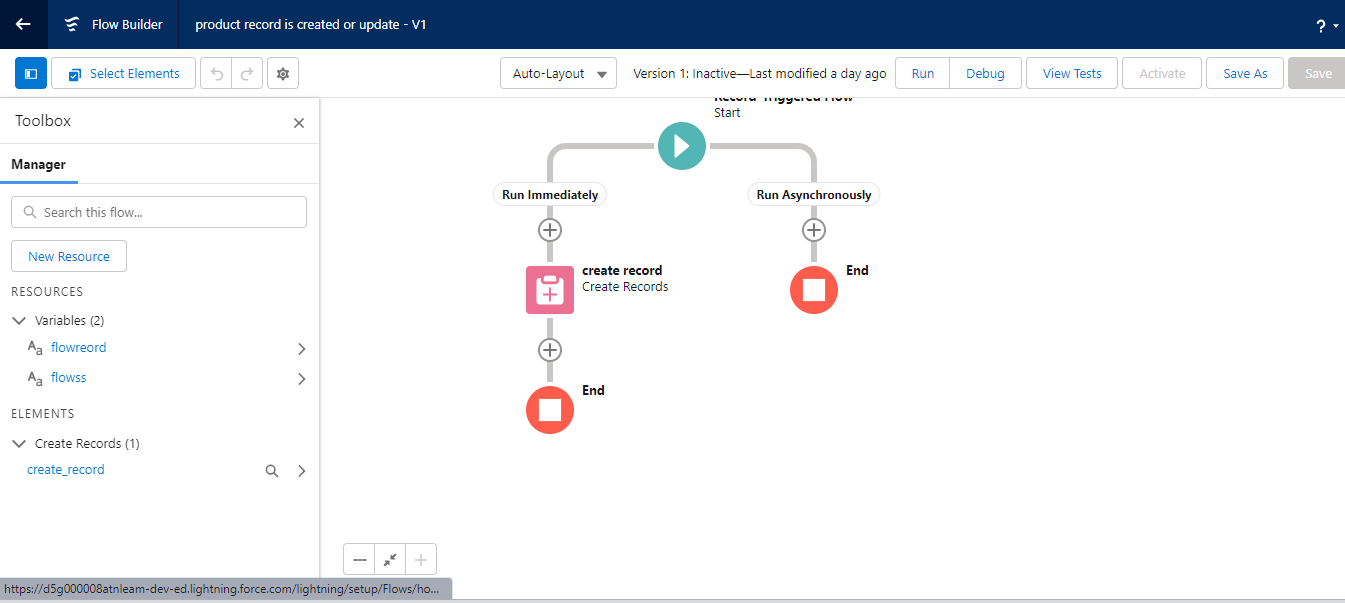
**6.1.3 Flow:** Flow (also known as Lightning Flow or Visual Workflow) is a powerful automation tool that allows you to create interactive and guided business processes. It provides a declarative way to build flows by visually designing a series of screens, decisions, and actions to automate complex business logic.

Here are the key aspects and functionalities of Flow in Salesforce:

1. **Visual Interface**: Flow offers a visual interface where you can design your business processes using a drag-and-drop builder. You can create flows by arranging and connecting elements on a canvas, making it easy to understand and modify the logic.
2. **Data Manipulation**: Flow enables you to collect, manipulate, and display data. You can use variables, formulas, and assignment elements to retrieve and update records, perform calculations, and store data for later use. Flow supports working with Salesforce objects, fields, and records.
3. **Logic and Decision Making**: Flow allows you to incorporate branching and decision elements to evaluate conditions and determine the path the flow should take based on the results. You can define logic to handle different scenarios and create conditional branches based on the data collected or evaluated.
4. **User Interaction**: Flow provides capabilities for user interaction by presenting screens to users. You can design screens to collect input, display information, or guide users through a step-by-step process. Flow supports user input validation, error handling, and dynamically adjusting the flow based on user responses.
5. **Integration and External Actions**: Flow supports integration with external systems, invoking Apex methods, and calling other flows or external services. You can interact with data and services outside of Salesforce to perform complex business logic or retrieve information from external sources.
6. **Error Handling and Exception Flows**: Flow allows you to handle errors and exceptions that may occur during the execution of the flow. You can define error screens, exception flows, and error handling logic to guide users through error resolution or alternate paths.

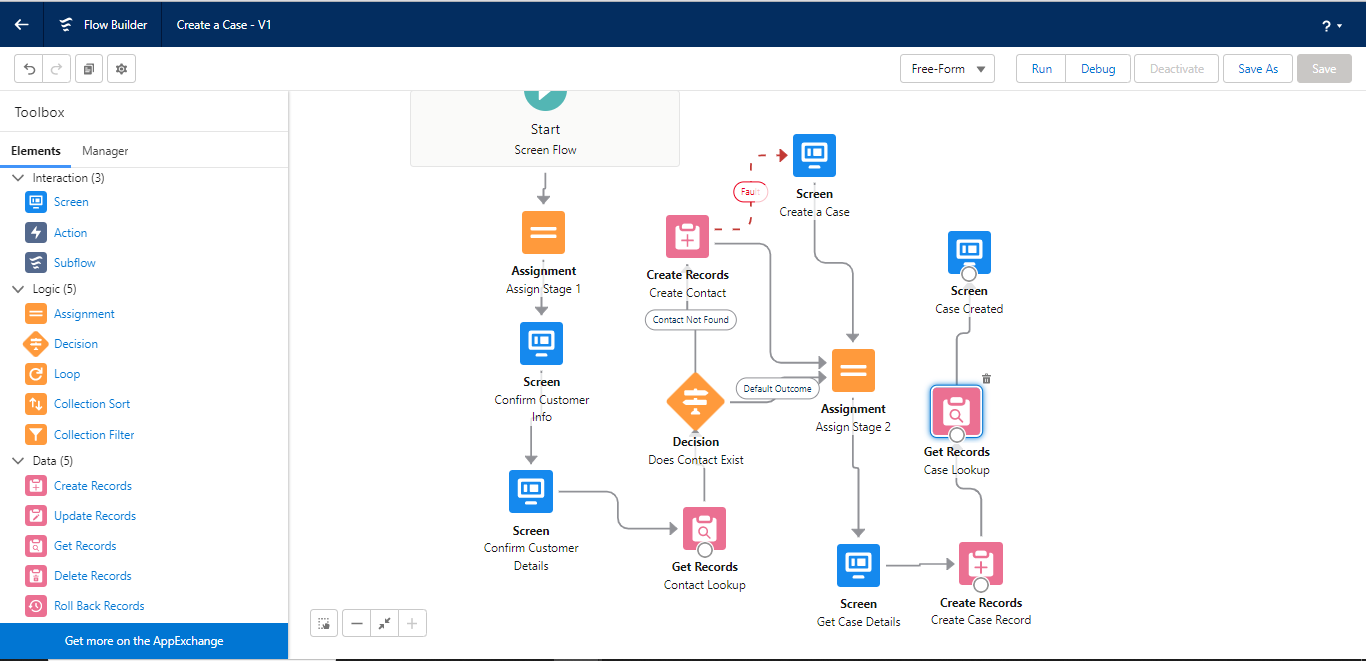
Flow provides a flexible and robust solution for automating complex business processes in Salesforce. It can be used for various use cases, such as guided selling, approval processes, data collection, multi-step wizards, and more. With its declarative nature, Flow empowers administrators and business users to create sophisticated automation without the need for custom code.

Please note that Flow is the recommended automation tool in Salesforce Lightning Experience, offering advanced capabilities and replacing Workflow Rules in many scenarios.



Top of Form

When Record is Created or Updated



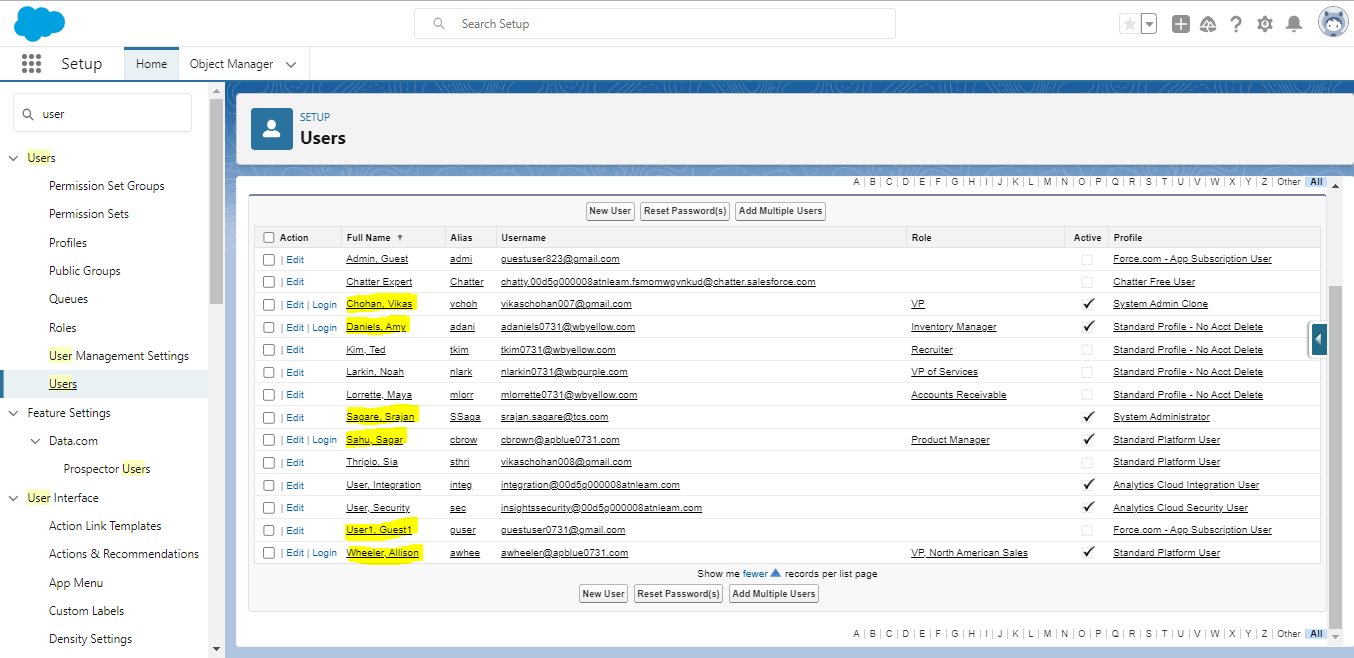
To Create a Case

**6.1.4 Users:** "Users" refer to individuals who have access to your Salesforce org. They are typically the individuals within your organization who interact with and use Salesforce to perform various tasks and operations.

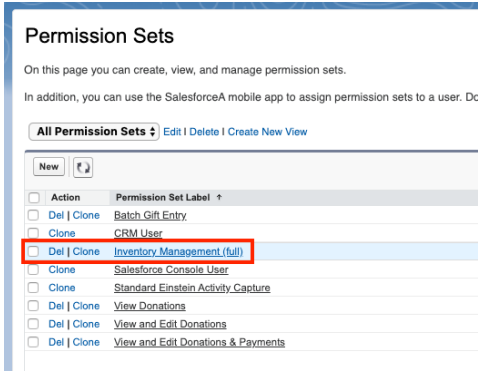
Here are some key points about Users in Salesforce:

1. **Authentication and Login**: Users have unique usernames and passwords or other authentication methods to log in to Salesforce. They require valid user credentials to access the Salesforce org.
2. **Access and Permissions**: Each user in Salesforce is assigned a specific set of permissions and access levels based on their role and responsibilities. These permissions determine what data, objects, and functionalities a user can view, edit, or interact with within the org.
3. **Profiles and Permission Sets**: User profiles and permission sets are used to manage and assign permissions to users. Profiles are assigned to users at the user level and define the initial set of permissions and settings. Permission sets can be assigned to users in addition to their profile, granting them additional permissions.
4. **Roles and Hierarchy**: Users can be assigned to roles within the organization, creating a hierarchy that defines the reporting structure and data visibility. Roles can determine which records a user can access based on their position in the hierarchy.
5. **Collaboration and Sharing**: Users can collaborate with other users by sharing records, files, and information within Salesforce. Sharing settings and rules control the level of access and visibility that users have to specific records and data.
6. **Licenses**: Users in Salesforce are associated with specific licenses that determine the features and functionalities available to them. Different licenses provide different sets of capabilities, such as Salesforce Platform, Sales Cloud, Service Cloud, or Marketing Cloud.
7. **User Profiles and Configuration**: User profiles define the standard permissions and settings for a particular type of user. Administrators can configure user profiles to control access, object permissions, and system settings for specific user roles or functions.
8. **User Administration**: Salesforce administrators are responsible for creating, managing, and maintaining user accounts within the org. They can create new user accounts, modify user details, assign permissions, manage passwords, and perform user-related administrative tasks..

Users play a vital role in Salesforce as they drive the day-to-day operations, data management, collaboration, and business processes within the organization. Managing users effectively ensures secure access, appropriate permissions, and optimal utilization of Salesforce features and functionalities.



**6.1.5 Permission Set Assignment:** In order for the user to access the Inventory Management App, the user needs access to the relevant elements like pages, objects, etc. This is done by assigning the user a permission set. Permission sets are a great way to control user access. To set it up, go to the Setup menu in your org and search for “permission sets”. Setup > Users > Permission Sets > Inventory Management (full). Click on the “Inventory Management (full)” permission set label



**6.1.6 Lightning Web Component:** This component allows users to enter their hourly rate and number of hours worked to calculate their salary.

1.salaryCalculator.html:

<template>

<lightning-card title="Salary Calculator">

<div class="slds-m-around\_medium">

<lightning-input type="number" label="Hourly Rate" value={hourlyRate} onchange={handleHourlyRateChange}></lightning-input>

<lightning-input type="number" label="Number of Hours" value={hoursWorked} onchange={handleHoursWorkedChange}></lightning-input>

<p>Salary: {salary}</p>

</div>

</lightning-card>

</template>

2.salaryCalculator.js:

import { LightningElement, track } from 'lwc';

export default class SalaryCalculator extends LightningElement {

@track hourlyRate;

@track hoursWorked;

@track salary;

handleHourlyRateChange(event) {

this.hourlyRate = event.target.value;

this.calculateSalary();

}

handleHoursWorkedChange(event) {

this.hoursWorked = event.target.value;

this.calculateSalary();

}

calculateSalary() {

const hourlyRateFloat = parseFloat(this.hourlyRate);

const hoursWorkedFloat = parseFloat(this.hoursWorked);

if (!isNaN(hourlyRateFloat) && !isNaN(hoursWorkedFloat)) {

const calculatedSalary = hourlyRateFloat \* hoursWorkedFloat;

this.salary = calculatedSalary.toFixed(2);

} else {

this.salary = 'N/A';

}

}

}

# **Chapter 7: Conclusion**

In conclusion, the Inventory Management project in Salesforce aims to provide a solution for efficiently managing inventory items within an organization. This project utilizes the Salesforce platform, specifically Lightning Web Components (LWC) and Apex, to create a user-friendly interface for inventory management tasks.

The key features of this project include:

1. **Inventory Item Management**: The project allows users to view, create, update, and delete inventory items. It provides a user interface to input item details such as name, quantity, price, and description.
2. **Search and Filtering**: The system enables users to search for specific items based on various criteria such as name or other attributes. It includes filtering capabilities to narrow down the displayed items based on specific conditions.
3. **Integration with Salesforce Database**: The project utilizes Apex controllers and SOQL queries to interact with the Salesforce database. This enables seamless retrieval and manipulation of inventory item data.
4. **User Interface**: The project incorporates a visually appealing and user-friendly interface using Lightning Web Components. It leverages features like Lightning Cards, Lightning Input, and Lightning Datatable to enhance the user experience.
5. **Accessibility and Responsiveness**: The components developed in this project adhere to Salesforce Lightning Design System (SLDS) guidelines, ensuring accessibility and responsiveness across different devices and screen sizes.
6. **Apex Controller**: The project includes an Apex controller class that facilitates server-side operations, such as fetching inventory items from the database, performing CRUD operations, and executing custom business logic.

Overall, the Inventory Management project provides a comprehensive solution for organizations to efficiently manage their inventory items using the power of Salesforce. It empowers users with easy navigation, search, filtering, and CRUD operations, contributing to streamlined inventory management processes and improved operational efficiency.