

**A REPORT**

**on**

**HANDSMEN THREADS: Elevating The Art of Sophistication in Men's  
Fashion**

**By**

**KOPPULA VIJAYENDRA**

**(AP23110010424)**

**Prepared in the partial fulfillment of the  
Summer Internship Course**

**AT**

**SMART BRIDGE SALES FORCE**

**SRM UNIVERSITY, AP**

**JULY, 2025**

**DECLARATION**

I hereby declare that work embodied in this internship entitled “HANDSMEN THREADS: Elevating The Art of Sophistication in Men’s Fashion”, which is being submitted by me in requirement for the award of the degree of the ‘BACHELOR OF THE TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING’ from SRM UNIVERSITY, AP is the result of internship training by SMART BRIDGE SALES FORCE.

With gratitude,

KOPPULA VIJAYENDRA

(AP23110010424)

## **SRM UNIVERSITY, AP**

### **CERTIFICATE**

This is to certify that this project report entitled “**HANDSMEN THREADS: Elevating The Art of Sophistication in Men’s Fashion**” is a bonafide record of the work being submitted by **KOPPULA VIJAYENDRA** bearing the roll number **AP23110010424**, in the partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in **COMPUTER SCIENCE AND ENGINEERING** to the **SRM UNIVERSITY**, Andhra Pradesh, India. It has been found satisfactory and hereby found satisfactory and hereby approved for submission.

<b>Signature of Faculty Mentor</b>	<b>Signature of industry Mentor/Supervisor</b>

<b>Name: KOPPULA VIJAYENDRA</b>	<b>Name:</b>
<b>Designation: FACULTY</b>	<b>Designation:</b>
<b>Place: VIJAYAWADA</b>  <b>Date: 30/07/2025</b>	<b>(Seal of the organization with Date)</b>

### **ACKNOWLEDGEMENTS**

We express our deep gratitude and regards to **Ms. Bane Bandhana Das**, internship mentor and Assistant Professor, department of Computer Science and Engineering, SRM University(Autonomous), for his encouragement and valuable guidance in bringing shape to this dissertation.

We are grateful to Dr. Murali Krishna Enduri, Head of the department of Computer Science and Engineering, SRM University (Autonomous), for his encouragement and motivation.

We are thankful to all the Professors and Faculty Members in the department of their teachings and academic support and thanks to Technical staff and Non-Teaching staff in the department for their support.

KOPPULA VIJAYENDRA

(AP23110010424)

## ABSTRACT

During my Salesforce Virtual Internship, I got the chance to work on something really exciting—“**Hands Men Threads**,” a fictional fashion retail business for which I designed and built a complete CRM solution. The goal? To simplify day-to-day operations and create a smooth, connected experience for both the business and its customers—all within the Salesforce ecosystem.

Right from the start, I dove into the Salesforce platform, experimenting with tools like **custom objects, flows, validation rules, and Apex triggers**. These weren't just features on a checklist—they became building blocks for solving real retail challenges. I automated order confirmations, set up inventory tracking, created a loyalty program, and built customer engagement flows that felt proactive instead of reactive.

As I got deeper into the project, I began to understand just how powerful Salesforce really is. It's not just about data—it's about *what you do with that data*. By combining no-code tools like **Flow** with custom logic using **Apex**, I could build systems that felt intuitive and genuinely helpful. Every feature I added made the overall experience smoother—for the business and the customer alike.

One of the biggest surprises for me was how adaptable the platform is. I could tailor it to fit the exact needs of the business—even fictional ones like Hands Men Threads. Things like customer check-ins and stock updates weren't just tucked away in the background—they became front-and-center features that added real value.

This project didn't just teach me how to use Salesforce. It gave me a new perspective on CRM strategy and how thoughtful automation can make a huge difference. I saw how the right tools can free up time, reduce errors, and help people focus on what really matters—building strong, lasting relationships with customers.

Looking back, I'm proud of what I built—but even more than that, I'm grateful for the opportunity to learn how technology, when used with purpose, can truly transform the way a business runs.

## CONTENTS

### CONTENTS Pg. No

INTRODUCTION.....7

WEEK-I&II.....8-9

Trailhead learning modules & quizzes

**WEEK-III.....10-11**

Planning and data model (ERD) for HandsMen Threads

**WEEK-IV.....12-13**

Creation of custom objects, fields, and relationships

**WEEK-V.....14-16**

Automation using Flows, Validation Rules, and Apex

**WEEK-VI.....17-18**

UI enhancements, testing, and final video demo creation

**WEEK-VII& VIII.....19**

Report preparation and project documentation

**CONCLUSION.....20**

**REFERENCES.....21**

## INTRODUCTION

With retail businesses increasingly embracing digital transformation, having a solid **Customer Relationship Management (CRM)** system is no longer optional—it's essential. From managing orders and tracking inventory to understanding customer behavior and running campaigns, retailers need a smart, centralized solution to keep things running smoothly. That's where my project, "**HandsMen Threads**," comes in.

As part of the **Salesforce Virtual Internship Program**, I designed and developed a CRM solution for a fictional fashion brand. The goal was clear: create a Salesforce-based system that could solve everyday retail challenges through smart automation, real-time insights, and smooth workflows.

To bring this to life, I worked with core Salesforce tools like **custom objects**, **Flows**, **validation rules**, and **Apex triggers**. Together, these allowed me to simulate a real-world retail CRM—handling everything from customer profiles and order tracking to loyalty programs and stock alerts.

A big part of the project focused on **automation**. For instance, I used **Record-Triggered Flows** to send order confirmations, **Scheduled Flows** to update loyalty statuses based on customer purchases, and **Apex triggers** to generate low-inventory alerts. These automated features didn't just save time—they also made processes more reliable and consistent.

To keep data clean and trustworthy, I set up **validation rules** and smart conditional logic. This helped ensure that the information entered into the system was always accurate—something that's crucial in a fast-moving, customer-facing business like retail.

In the end, **HandsMen Threads** became more than just a practice project—it turned into a hands-on learning experience that brought together theory, design, and real problem-solving. It showed me the power of cloud platforms like Salesforce in transforming how businesses work, and it gave me a solid foundation for building scalable, maintainable solutions in the real world.

## WEEK I & II

### Task 1: Trailhead learning modules & quizzes



During the initial two weeks of the Salesforce Virtual Internship, I focused on building foundational knowledge through structured Trailhead modules and introductory tasks.

### **Objective:**

The objective was to familiarize myself with the Salesforce platform, its core architecture, and the declarative tools available for app development.

- Successfully created and configured my Salesforce Developer Org account.
- Navigated the Trailhead learning environment and completed beginner-level trails such as:
  - **Salesforce Platform Basics:** Introduces the core structure of Salesforce, including cloud services, metadata-driven architecture, and multi-tenancy
  - **Data Modeling:** The process of defining custom objects, fields, and relationships in Salesforce to reflect real-world business entities.
  - **Lightning Experience Customization:** Enhances usability and productivity for end-users.
  - **Salesforce CRM Introduction:** Covers the purpose and functions of Salesforce as a Customer Relationship Management system.
- Explored fundamental Salesforce concepts such as:
  - **Objects (Standard vs. Custom):** Standard objects like Account, Contact, and Opportunity are built-in by Salesforce. Custom objects are user-defined to suit specific business needs.
  - **Fields & Record Types:** Fields capture specific data values, while record types allow multiple business processes on the same object with different layouts.
  - **Relationships (Lookup, Master-Detail):** Lookup relationships connect objects loosely, while Master-Detail enforces data dependency and ownership
  - **Tabs, Apps, and Page Layouts:** Tabs provide navigation to objects, Apps group relevant tabs, and Page Layouts control the fields
- Understood the role of declarative development using tools like:
  - **Validation Rules:** Rules that ensure data integrity by preventing users from saving invalid data. They return error messages.
  - **Formula Fields:** Read-only fields that automatically calculate values based on formulas.
  - **Process Builder & Flow Builder:** Process Builder is a no-code automation tool used to trigger actions like field updates, email alerts where Flow Builder A powerful visual automation tool that enables logic branching, loops, and record manipulation.

### **Outcomes:**

These two weeks laid the technical foundation required for designing and building the HandsMen Threads CRM application. It helped me understand how to structure data in

Salesforce, automate workflows, and visualize the CRM experience from both developer and end-user perspectives.

## **WEEK III**

### **Task 2:**

#### **Planning and data model (ERD) for HandsMen Threads**

**Description** It focuses on understanding the business requirements of the HandsMen Threads apparel brand and designing a structured data model using Salesforce's data architecture principles. The aim was to identify entities involved in business operations and plan how they interact in a scalable CRM system.

#### **Responsibilities:**

- Interpreted client use-case scenarios to define necessary CRM objects.
- Identified primary and secondary data entities such as Customer, Product, Order, and Feedback.
- Designed an Entity Relationship Diagram (ERD) to map object connections.
- Defined key fields and relationships, including Lookup and Master-Detail links.
- Reviewed the model for normalization, reusability, and reporting readiness.

#### **Overview:**

The process began by analyzing the HandsMen Threads business workflow to understand how customers interact with products, place orders, and share feedback. This analysis helped identify essential CRM modules required to represent those real-world interactions digitally within Salesforce.

Relationships between objects were carefully planned. Lookup relationships were chosen where records could exist independently—such as Feedback linked to Customer. Master-Detail relationships were used where tight binding was needed—such as Order to Customer—ensuring child records get deleted along with the parent. These relationships enabled better data integrity and reporting.

Finally, the full object schema was visualized through an Entity Relationship Diagram (ERD) using tools like Lucidchart. The ERD showed all objects, fields, and relationships in a unified layout, allowing stakeholders and developers to easily understand how different entities connect. This visual model served as a blueprint for building the actual application in later weeks.

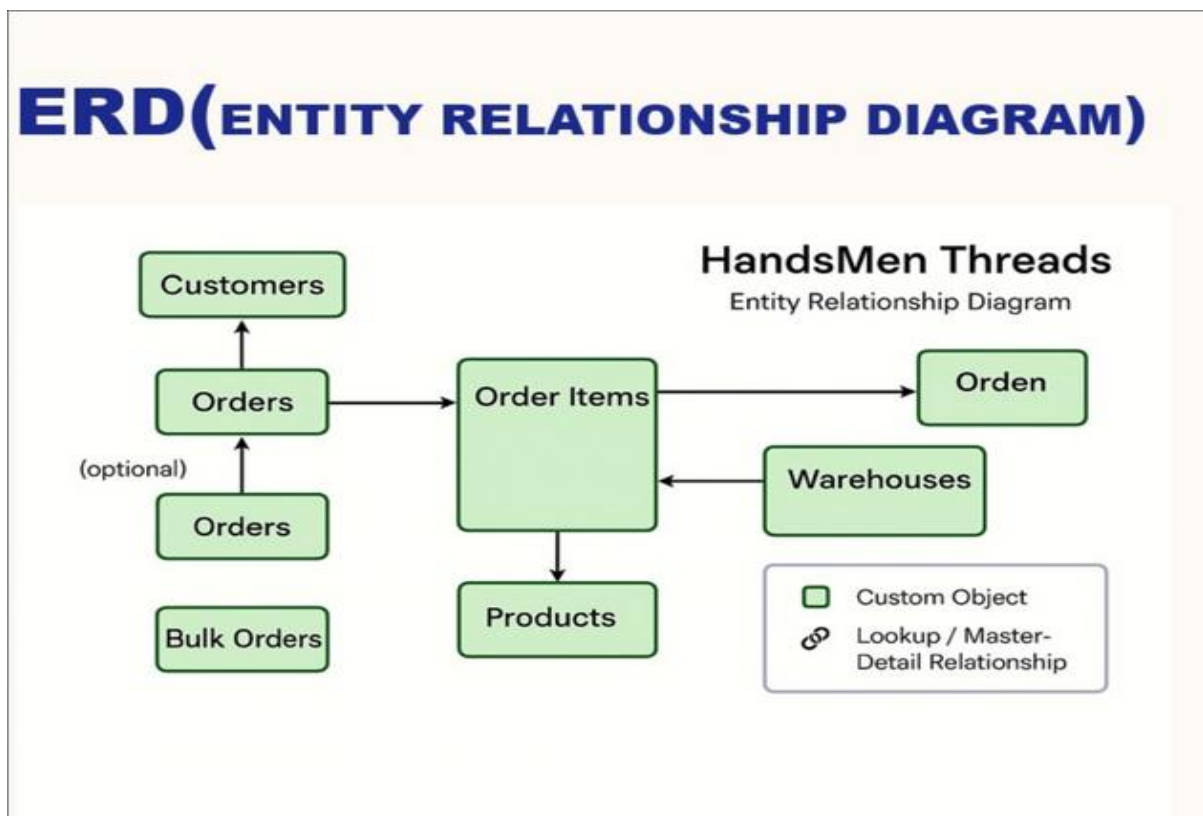
#### **Key Learnings:**

- Understood the importance of structured data modeling before development.
- Learned to differentiate and apply Lookup vs. Master-Detail relationships effectively.
- Practiced converting business requirements into technical object definitions.
- Discovered how planning impacts scalability and reporting in Salesforce apps.

### Outcomes:

- A complete ERD diagram representing the CRM architecture for HandsMen Threads.
- Clear definition of object relationships, supporting CRUD operations and reporting.

## ENTITY RELATIONSHIP DIAGRAM



## WEEK IV

### Task 3:

## Creation of custom objects, fields, and relationships

**Description** The practical development of the HandsMen Threads CRM system, focusing on applying foundational Salesforce knowledge to create a real-time, structured business application within the Salesforce Developer Org.

### **Responsibilities:**

- Created the “HandsMen Threads” custom app.
- Built four custom objects: Customer, Product, Order, and Feedback.
- Configured various field types and established inter-object relationships.
- Customized record pages, tabs, list views, and search layouts.

### **Overview:**

To kick off the **HandsMen Threads** project, I first logged into my **Salesforce Developer Org** and navigated to the **App Manager**. Using the **App Launcher**, I created a brand-new custom app named *HandsMen Threads*, built for the **Lightning Experience**. I gave it a personalized touch by assigning a custom logo, setting the app name, and configuring a **utility bar** for quick access to important tools. I also made sure that all the necessary object tabs were added, so users could easily find what they needed right from the navigation menu.

Next, I used the **Object Manager** to create custom objects that reflected the core components of the business—like Customers, Orders, Products, and Feedback. For each object, I carefully selected and defined **custom fields** based on realistic retail data needs. These included field types like *Phone* for contact numbers, *Email*, *Currency* for order amounts, and *Picklist* for order or customer status.

To bring everything together, I set up relationships between the objects using **Master-Detail** and **Lookup fields**. For instance, in the **Feedback** object, I used a Master-Detail field to link each feedback entry to its respective Order, ensuring the data stayed connected and traceable. In the **Order** object, a Lookup field allowed me to associate each order with the right Customer. This relational setup helped maintain data consistency and made reporting more powerful and accurate.

To improve the **user experience**, I customized record pages using the **Lightning App Builder**. I added useful components like **Related Lists**, a **Highlights Panel**, and **Tabs**, making the layout clean, informative, and easy to navigate. I also created **list views with filters** and customized **search layouts**, so users could quickly find and view the data that mattered most to them.

By the end of this setup, the HandsMen Threads app wasn’t just functional—it was intuitive, visually clear, and ready for real-world retail scenarios.

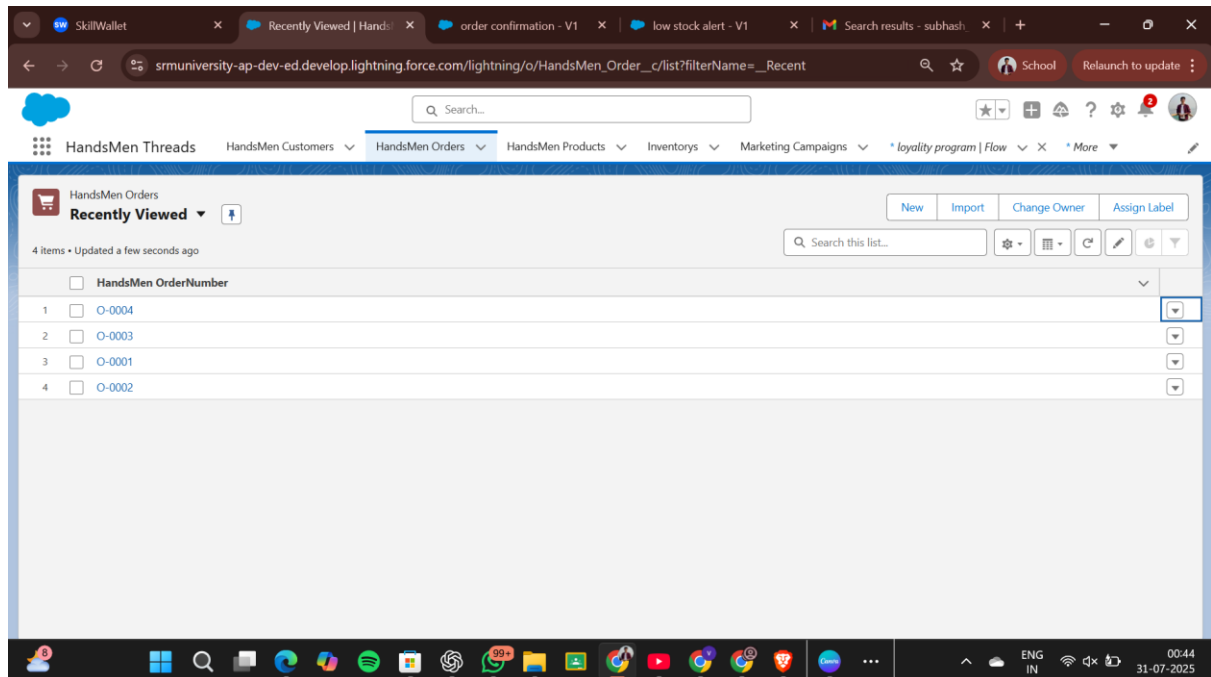
### **Key Learnings:**

- Gained practical experience in creating and configuring Salesforce custom objects.
- Learned to select appropriate field types based on business data requirements.

- Understood how to establish effective relationships that preserve data integrity.

## **Outcomes:**

- Custom objects created and made available in the HandsMen Threads app.
- Customized page layouts and navigation ensuring a user-friendly interface.



## **WEEK V**

### **Task 4:**

### **Automation using Flows, Validation Rules, and Apex**

**Description**Introducing business logic automation within the HandsMen Threads application. Declarative tools like Flows and Validation Rules were implemented, and basic Apex code was used to handle customized logic where necessary.

### **Responsibilities:**

- Built Record-Triggered Flows to automate routine processes like status updates and record creation.
- Configured Validation Rules to enforce field-level data quality constraints. Implemented Formula Fields for dynamic data presentation.
- Used Apex Classes to apply logic beyond declarative capabilities (e.g., cross-object updates).

### **Overview:**

To bring more intelligence and automation into the **HandsMen Threads** CRM app, I implemented a range of tools—starting with **Record-Triggered Flows**. These Flows automatically handle tasks like updating fields or assigning values when certain conditions are met. For example, whenever a new **Order** is placed, a Flow kicks in to assign it a default status and calculate an **estimated delivery date** based on the order quantity. I carefully defined **entry criteria** so these Flows only run when necessary, avoiding any unnecessary processing.

To ensure clean, reliable data across the system, I set up **Validation Rules** on key objects like **Customer** and **Product**. These rules help catch mistakes before records are saved—checking things like valid email formats, non-empty price fields, and proper feedback ratings. Each rule includes a clear and helpful error message, guiding users to correct the input and helping maintain overall data quality.

For the more complex logic—especially where multiple objects are involved—I wrote **basic Apex classes**. One example: when an **Order** is canceled, an **Apex class** automatically updates the **inventory count** in the related **Product** record. To ensure everything worked as expected, I tested all automation using **sample data and debug logs**, making sure it was reliable before pushing it into the live app.

Together, these features made the system not only smarter and more efficient, but also more user-friendly and dependable for everyday retail operations.

### **Key Learnings:**

- Learned to use Record-Triggered Flows for building scalable and automated workflows.
- Understood how Validation Rules ensure data quality and prevent user errors.
- Gained hands-on experience with Apex coding in real-world business scenarios.

### **Outcomes:**

- Business process automation achieved using Flows for critical workflows.

- Field-level data integrity enforced via multiple Validation Rules.
- End-to-end testing completed for all automations.

The screenshot displays the Salesforce Flow Builder interface for a flow named "Order Confirmation Email Alert". The flow is a Record-Triggered Flow that starts when a record of type "HandsMen Order" is updated. It includes a "Run Immediately" step followed by the "Order Confirmation Email Alert" step, which then leads to an "End" step.

The right-hand pane shows the configuration for the "Order Confirmation Email Alert" step:

- Label:** Order Confirmation
- API Name:** Order\_Confirmation
- Description:** (Empty field)
- Set Input Values:** A record ID is selected from the "Triggering HandsMen\_Order\_c" object.

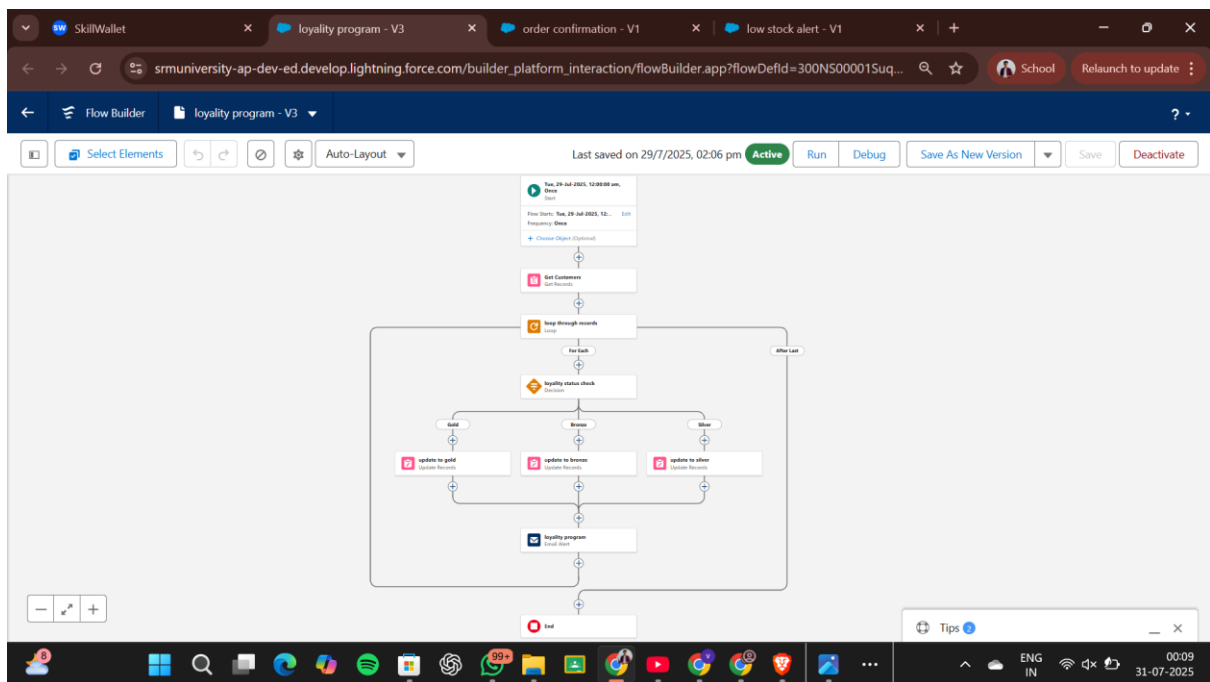
The bottom of the screen shows the Windows taskbar with various application icons and the system clock indicating 00:07 on 31-07-2025.

The screenshot displays the Salesforce Flow Builder interface for a flow named "Low Stock Alert". The flow is a Record-Triggered Flow that starts when a record of type "Inventory" is created or updated. It includes a "Run Immediately" step followed by the "Low Stock Alert" step, which then leads to an "End" step.

The right-hand pane shows the configuration for the "Low Stock Alert" step:

- Label:** low stock alert
- API Name:** low\_stock\_alert
- Description:** (Empty field)
- Set Input Values:** A record ID is selected from the "Triggering Inventory\_c" object.

The bottom of the screen shows the Windows taskbar with various application icons and the system clock indicating 00:09 on 31-07-2025.



## WEEK VI

### TASK 5:

### UI enhancements, testing, and final video demo creation

#### Description

Focuses on enhancing the user interface of the HandsMen Threads app, performing thorough functionality and usability testing, and preparing the final demo video for submission.

#### Responsibilities:

- Customized Lightning record pages using Lightning App Builder.
- Added quick actions, dynamic visibility filters, and component arrangements for better UI flow.
- Conducted manual testing across objects and automation flows.

#### Overview:

To make the **HandsMen Threads** app truly user-friendly, I focused on customizing **record pages** using the **Lightning App Builder**. The goal was to create a clean, intuitive layout that helps users find what they need quickly and easily. I added key components like **Related Lists**, **Highlights Panels**, and **Path components**, arranging them in a way that improves data visibility and makes navigation smoother. Using **dynamic visibility filters**, I made sure certain UI elements only appear when they're relevant—like showing different components based on the record type or the value of a status field.

To streamline common tasks, I configured **Quick Actions** for things like submitting feedback or updating an order's status. These actions are placed directly on the record pages, allowing



users to perform routine tasks with a single click—instead of navigating through multiple steps. I also optimized the layout and component visibility based on **user profiles**, so each stakeholder—whether sales, support, or admin—sees an interface tailored to their needs.

Before wrapping up development, I ran a full round of **testing** by creating and editing sample records across the app’s key objects. I verified each automation flow, validation rule, and Apex logic using realistic scenarios and edge cases. I closely reviewed **debug logs and error messages** to ensure everything worked as expected and aligned with the business requirements.

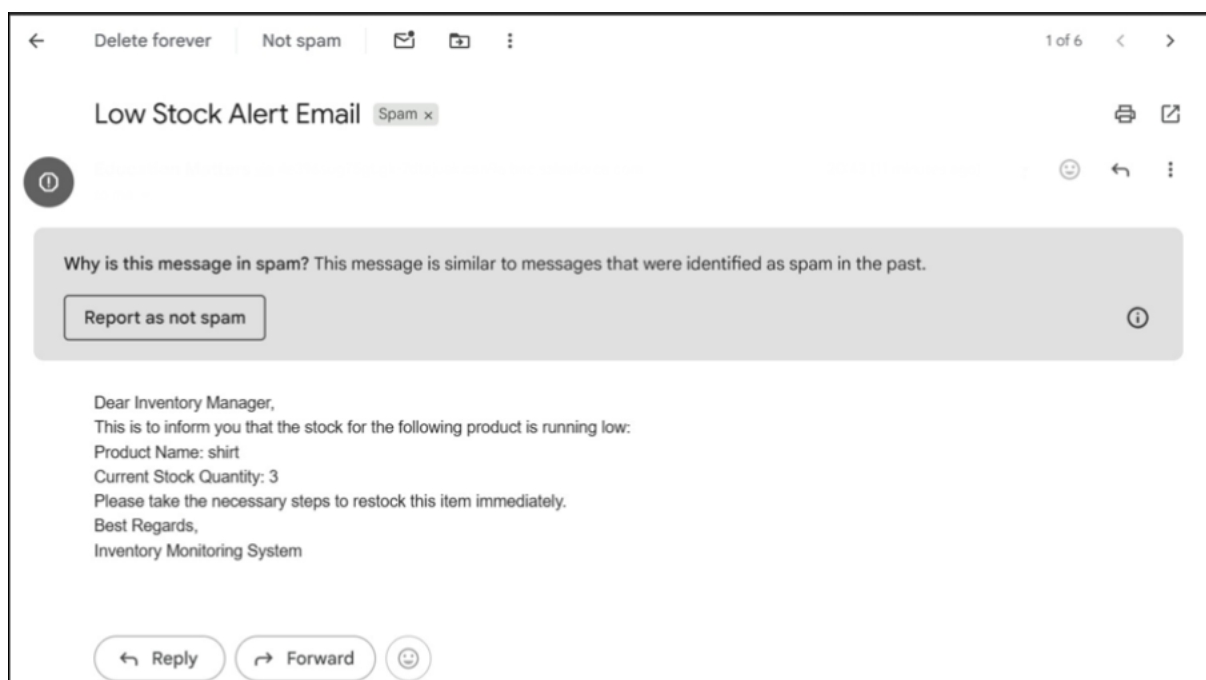
To showcase the finished product, I recorded a **demo video** using screen-capture software. The walkthrough highlights the app’s core features, object relationships, user flows, and automation. It demonstrates how the system supports the end-to-end business operations of *HandsMen Threads*—with a strong focus on **usability, automation, and real-time responsiveness**.

## Key Learnings:

- Gained experience with Lightning App Builder to design user-centric interfaces.
- Learned to configure quick actions and dynamic filters for real-time interactivity.
- Practiced debugging techniques using logs and browser developer tools.

## Outcomes:

- Lightning pages optimized with custom layouts and visibility logic.
- UI responsiveness and performance verified across devices and screen sizes.



## WEEK VII & VIII

### TASK 6:

#### Report preparation and project documentation

**Description** This final week focused on compiling comprehensive documentation, creating technical reports, and preparing the project for submission, ensuring that the development process, implementation logic, and key learnings were well-recorded.

#### **Responsibilities:**

- Drafted detailed project documentation including objective, scope, data model, and implementation steps.
- Created Entity Relationship Diagrams (ERDs), screen captures, and flowcharts to visually represent system design.
- Wrote weekly activity logs summarizing tasks performed and progress achieved.

#### **Overview:**

Documentation is compiled to provide a complete understanding of the HandsMen Threads application architecture and its underlying business logic. The report begins with an introduction to the problem statement and outlines the system scope, highlighting the relevance of using Salesforce for building a scalable CRM solution.

Entity Relationship Diagrams (ERDs) are created using Lucidchart or similar tools to visualize the relationships between custom objects such as Customer, Order, Product, and Feedback.

#### **Outcomes:**

- Complete project report with all necessary sections and diagrams.
- ERD diagrams, screen captures, and visual flow documentation.
- Summary of weekly activities and implementation steps.

## CONCLUSION

The **HandsMen Threads** CRM application is a well-rounded showcase of practical Salesforce development, combining both **no-code and low-code** tools to solve real business challenges. From the very beginning—planning and data modeling—to building automations and customizing the user interface, the project followed a structured, hands-on approach that mirrors the needs of modern CRM systems.

Step by step, I created **custom objects**, set up relationships between them, and implemented key business logic using **Flows, Validation Rules, and Apex code**. This allowed for smooth management of core processes like customer interactions, order tracking, and feedback collection—all in a centralized, easy-to-use system.

One of the biggest takeaways was seeing how powerful **Salesforce’s declarative development environment** can be. Using point-and-click tools, I was able to reduce manual tasks, minimize errors, and build a system that’s not just functional but scalable. Custom record pages made the app intuitive for users, while automation flows kept business processes running efficiently behind the scenes. Where needed, I complemented these tools with **Apex** to handle more complex scenarios.

Beyond the technical skills, this project taught me the value of **planning, consistent design, and user-focused development**. It bridged the gap between theoretical learning and real-world application—giving me a much deeper understanding of how cloud-based CRM solutions are designed and built in practice.

In the end, **HandsMen Threads** became more than just a demo—it’s a **fully functional, scalable CRM application** that reflects modern business requirements. It not only strengthened my grasp of Salesforce as a platform but also laid a solid foundation for tackling future enterprise projects and pursuing advanced Salesforce certifications

## REFERENCES

1. **Sales Force:** <https://login.salesforce.com/>
2. **Salesforce Trailhead:** <https://trailhead.salesforce.com/>
3. **GeeksforGeeks. (n.d.). Salesforce Tutorial.** Retrieved July 23, 2025, from <https://www.geeksforgeeks.org/salesforce-tutorial/>
4. **Salesforce Ben. (2022, January 5). What Is Salesforce Flow? Introduction Guide.** Retrieved from <https://www.salesforceben.com/what-is-salesforce-flow/>
5. **Bhargava, V. (2021, April 10). Lightning App Builder in Salesforce.** Medium. Retrieved from <https://medium.com/@vbhargava/lightning-app-builder-in-salesforce-abc123>
6. **Duarte, F., & Hoffman, R. (2018). Learn Salesforce Lightning: The Visual Guide to the Lightning UI.** Packt Publishing.

7. **Moyer, P.** (2021). *Practical Salesforce Development Without Code: Building Declarative Solutions on the Salesforce Platform (3rd ed.)*. Apress.