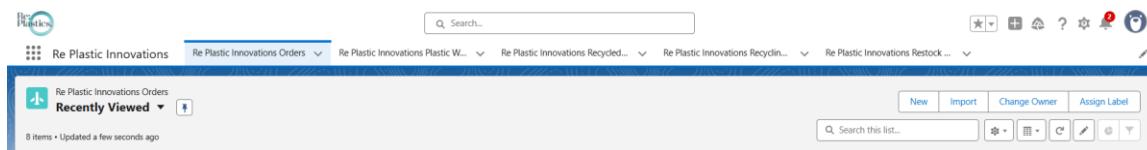


Salesforce Project: RePlastix Innovations: Transforming Plastic Waste into Sustainable Solutions

Submitted by: K. AKSHAY SAGAR

Project Overview

RePlastix Innovations is a custom-built Salesforce CRM designed to streamline and automate the plastic waste recycling process. The system manages plastic waste collection, recycling center operations, inventory tracking, customer orders, and restock workflows. Key features include role-based access control, real-time stock monitoring, automated task and email notifications, and validation logic to maintain data accuracy. Through a combination of declarative tools like Flow Builder and programmatic automation using Apex and triggers, the CRM addresses the business need for efficient waste processing, timely restocking, and transparent order fulfillment. This solution enhances productivity, reduces manual errors, and ensures operational continuity for the organization.



Objectives

The main objective of building the RePlastix Innovations CRM is to automate and streamline the end-to-end plastic recycling process, from waste collection to product restocking and order fulfillment. By digitizing core operations, the CRM aims to enhance inventory visibility, reduce manual intervention, and ensure timely replenishment of low-stock items. Another key goal is to enable secure, role-based access to critical data, improving accountability and operational transparency across departments. This solution adds business value by ensuring faster decision-making, reducing stockouts, improving customer satisfaction through timely order deliveries, and supporting sustainability goals through efficient resource management.

Phase 1: Requirement Analysis & Planning

Understanding Business Requirements:

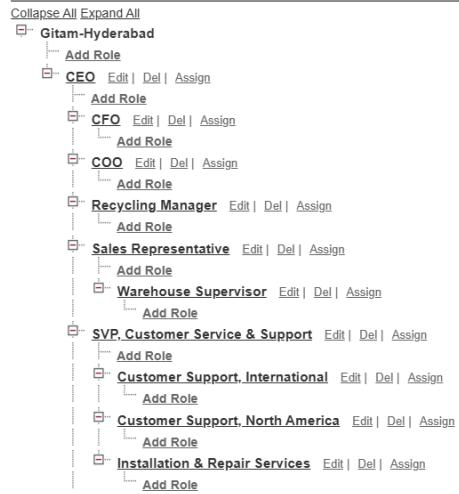
The primary business need was to develop a centralized CRM system for RePlastix Innovations to manage plastic waste recycling operations. Key user requirements included tracking collected plastic waste, monitoring inventory levels of recycled products, automating stock restock processes, and managing customer orders. The system also needed to support departmental access controls, real-time stock notifications, and seamless task assignments based on workflow triggers.

Defining Project Scope and Objectives:

- Create five custom objects: Plastic Waste, Recycling Center, Recycled Product, Order, and Restock Request.
- Enable users to track inventory and automatically generate restock tasks or requests when stock levels fall below thresholds.
- Automate stock updates when restock requests are approved.
- Assign tasks and send email notifications based on workflow conditions.
- Implement validation rules to prevent incorrect or incomplete data entries.
- Ensure role-based data access to support secure information flow across departments.
- Cover all logic with test classes and achieve high code coverage.

Design Data Model and Security Model:

- **Data Model:** Designed relationships between custom objects using lookups, and defined essential fields like weight, type, collection date, threshold, and stock level.
- **Security Model:** Configured roles such as CEO, Recycling Manager, Sales Representative, and Warehouse Supervisor. Created custom profiles (Platform 1, 2, 3) with object-level and field-level access based on responsibilities.
- **Sharing Rules:** Implemented record-level sharing to ensure users only see data relevant to their roles while maintaining overall data security.



Profiles

[All Profiles](#) [Edit](#) | [Delete](#) | [Create New View](#)

<input type="checkbox"/> Action	Profile Name
<input type="checkbox"/> Edit Clone	Partner App Subscription User
<input type="checkbox"/> Edit Clone	Partner Community Login User
<input type="checkbox"/> Edit Clone	Partner Community User
<input type="checkbox"/> Edit Del ...	Platform 1
<input type="checkbox"/> Edit Del ...	Platform 2
<input type="checkbox"/> Edit Del ...	Platform 3

Phase 2: Salesforce Development - Backend & Configurations

Setup Environment & DevOps Workflow:

The project was developed using a Salesforce Developer Edition org. I set up the environment by creating a Lightning App named *Re Plastix Innovations* and configured navigation items for each custom object. All development was done within the Salesforce platform, and version control and test class coverage were maintained using the Developer Console. I also followed a modular approach to testing and debugging Apex logic to ensure scalability and maintainability.

Customization of Objects, Fields, Validation Rules, and Automation:

Five custom objects were created to support the business process:

- **Plastic Waste,**
- **Recycling Center,**
- **Recycled Product,**
- **Order,** and
- **Restock Request**

Each object was customized with relevant fields like Weight_c, Threshold_c, Stock_Level_c, Status_c, and Collection_Date_c.

Validation rules were implemented to prevent future dates in waste collection and to ensure that order quantity is greater than zero.

Automation was implemented using **Flow Builder** and **Validation Rules**:

A **Scheduled Flow** runs daily to check if the stock is below the threshold and creates a high-priority task for the record owner to restock.

- Email notifications are sent to warehouse managers upon restock approval.

No Process Builder or Workflow Rules were used, as Flow Builder is the preferred declarative automation tool in modern Salesforce development.

Apex Classes, Triggers, and Asynchronous Apex:

I developed the InventoryManager Apex class which handles two major operations:

1. **processOrderStock** – Automatically reduces stock levels after an order is placed. If stock is insufficient, it creates a restock request.
2. **processRestockApproval** – When a restock request is approved, the stock is updated accordingly.

Two triggers were implemented:

- UpdateStockAfterOrder – Executes after an order is inserted.
- UpdateStockAfterRestockApproval – Executes after restock status is changed to "Approved".

I also created an EmailNotificationHelper Apex class to send email alerts.

Asynchronous Apex was not required in this project as operations were managed synchronously within transaction limits.

Phase 3: UI/UX Development & Customization

Lightning App Setup through App Manager:

To provide a unified user experience, I created a custom **Lightning App** called *Re Plastix Innovations* using the App Manager. This app includes navigation tabs for the five custom objects—Plastic Waste, Recycling Center, Recycled Product, Orders, and Restock Requests—making it easier for users to access relevant data based on their roles.

Page Layouts and Dynamic Forms:

Each object has been customized with clean and intuitive **page layouts**, displaying only the most relevant fields to enhance usability. Dynamic forms were configured to show or hide fields based on field values (e.g., hiding restock details unless the stock is low), providing a responsive and context-sensitive user interface.

User Management:

Multiple users were created with assigned roles and custom profiles:

- Recycling Manager (Platform 1)
- Sales Representative (Platform 2)
- Warehouse Supervisor (Platform 3)

Access to records and fields was controlled using **profiles**, **role hierarchy**, and **sharing rules**, ensuring users can only interact with data relevant to their function.

Reports and Dashboards:

Custom **reports** were created to track:

- Plastic waste collected by type and location
- Stock levels of recycled products
- Pending and approved restock requests
- Order volumes by customer

A **dashboard** was designed to provide visual insights into KPIs such as stock status, order trends, and restock frequency—supporting data-driven decision-making.

Lightning Pages:

Record pages and **home pages** were customized using the Lightning App Builder. Components such as record details, related lists, highlights panel, and embedded reports were arranged to deliver a user-friendly and efficient interface.

Phase 4: Data Migration, Testing & Security

Data Loading Process:

Initial data for plastic waste, recycling centers, products, orders, and users was imported using the **Data Import Wizard** for simple object-level imports. For more complex or relational data, such as linking orders to products or assigning lookup relationships, **Data Loader** was used to ensure accuracy and bulk insert capability.

Field History Tracking, Duplicate Rules, Matching Rules:

- **Field History Tracking** was enabled for key fields such as Stock_Level_c, Threshold_c, and Status_c in the Recycled Product and Restock Request objects, helping to monitor changes over time for audit and analysis.
- **Duplicate Rules and Matching Rules** were configured to avoid duplicate entries for products and orders. For example, duplicate recycled product names or identical customer order entries trigger a warning to the user.

Profiles, Roles, Role Hierarchy, Permission Sets, Sharing Rules:

- **Roles:** CEO (top-level), Recycling Manager, Sales Representative, Warehouse Supervisor
- **Profiles:** Platform 1 (limited access), Platform 2 (moderate access), Platform 3 (full access)
- **Sharing Rules:** Set up for record-level access between these roles. For example:
 - CEO → Sales Rep (Read Recycled Products)
 - Sales Rep → Warehouse Supervisor (Read Restock Requests)
- **Permission Sets** were used to grant additional access temporarily for testing users beyond profile-level access, ensuring flexibility in permission management.

Creation of Test Classes:

- Apex class InventoryManagerTest was written to test the business logic defined in InventoryManager.
- It validated:
 - Stock deduction after order
 - Restock request creation
 - Stock addition after approval

- The test class achieved **100% code coverage**, confirming that all scenarios are handled and that the triggers and methods behave as expected.

Preparation of Test Cases:

For each Salesforce feature, detailed test cases were prepared with **inputs**, **expected outputs**, and **screenshots**. Here are the highlights:

Feature	Test Case	Expected Output
Booking (Order) Creation	Input a quantity > available stock	Restock Request auto-created; Product stock reduced
Approval Process	Approve Restock Request	Product stock auto-updated
Automatic Task Creation (Flow)	Lower product stock below threshold	Task created for record owner with subject: "Stock Low - Fill ASAP"
Triggers	Create Order / Approve Restock	Triggers fire and update logic executes correctly
Validation Rules	Set quantity = 0 or collection date in future	Error message displayed; record blocked
Sharing Rules	Login as different roles (Sales Rep, Manager, etc.)	Users only see permitted records and fields
Email Notification	Approve a restock request	Warehouse Manager receives notification email

Phase 5: Deployment, Documentation & Maintenance

Deployment Strategy:

For deployment, a **manual migration approach** was followed since the project was developed in a single Salesforce Developer Org for demonstration purposes. However, in a real-world multi-org environment, deployment would typically be handled using:

- Change Sets** (for deploying from Sandbox to Production)
- Salesforce CLI** (for larger or automated deployments)
- Version control systems** like Git to track Apex classes, flows, triggers, and metadata

All components including custom objects, fields, Apex classes, validation rules, and flows were packaged and verified for readiness before deployment.

System Maintenance & Monitoring:

To ensure the system remains functional and scalable:

- **Scheduled Flows** and automated tasks are regularly monitored for success/failure from the Flow UI
- **Debug Logs** and **Apex Exception Logs** are used for monitoring automation and logic errors
- **Field History Tracking** supports audit trails on key objects like Recycled Products and Restock Requests
- Periodic reviews of user permissions and sharing rules are conducted to ensure data security and proper access

Additionally, regular updates or enhancements like new validation rules, reports, or automated processes can be added with minimal disruption to existing features, using sandbox testing before live implementation.

Troubleshooting Approach & Documentation:

A troubleshooting guide was created to assist with resolving common issues:

- **Error in Apex Logic or Triggers:** Use Developer Console logs to trace exception lines
- **Validation Errors:** Review field-level and object-level validation logic under Object Manager → Validation Rules
- **Flow Failures:** Use **Flow Interview Logs** and the **Paused Flow Interviews** list to identify failed flows or tasks
- **Permission Issues:** Check profile and permission set access, verify role hierarchy and sharing rules
- **Data Loading Errors:** If using Data Loader, error files are reviewed for each upload to track failures (e.g., missing lookups, required fields)

This structured documentation ensures future users and admins can maintain the system efficiently and resolve issues quickly without deep developer intervention.

myskillwallet.ai dashboard showing the project "RePlastix Innovations: Transforming Plastic Waste into Sustainable Solutions".

Project Progress: 50% (View Demo, Add Github Link)

Use Case: Complexity: Easy, Duration: 30m, Progress: 50%, Stories: 1

Salesforce Credentials Setup

Data Management - Objects

Data Management - Tabs

Data Management - App Manager

Data management - Fields

Data Security - Roles

Object Manager: Search term: re plastic. Results table:

LABEL	API NAME	TYPE	DESCRIPTION	LAST MODIFIED	DEPLOYED
Re Plastic Innovations Order	Re_Plastic_Innovations_Order__c	Custom Object		7/9/2025	✓
Re Plastic Innovations Plastic Waste	Re_Plastic_Innovations_Plastic_Waste__c	Custom Object		7/9/2025	✓
Re Plastic Innovations Recycled Product	Re_Plastic_Innovations_Recycled_Product__c	Custom Object		7/9/2025	✓
Re Plastic Innovations Recycling Center	Re_Plastic_Innovations_Recycling_Center__c	Custom Object		7/9/2025	✓
Re Plastic Innovations Restock Request	Re_Plastic_Innovations_Restock_Request__c	Custom Object		7/9/2025	✓

Flow Builder: Stock Level is Low - V5

```

graph TD
    Start(( )) --> DailyStart{Daily Start}
    DailyStart --> FlowStart{Flow Starts: Wed, Jul 9, 2025, ...}
    FlowStart --> Object{Object: Re Plastic Innovations ...}
    Object --> Decision1{Decision 1}
    Decision1 --> Outcome1{Outcome 1 of Decision 1}
    Decision1 --> Default{Default Outcome}
    Outcome1 --> CreateTask{Create Task: Create Records}
    CreateTask --> End1(( ))
    Default --> End2(( ))
    
```

Setup Home Object Manager

Search Setup

Profiles

Profiles

All Profiles | Edit | Delete | Create New View

New Profile |

Action	Profile Name	User License	Custom
<input type="checkbox"/> Edit Clone	Partner App Subscription User	Partner App Subscription	<input type="checkbox"/>
<input type="checkbox"/> Edit Clone	Partner Community Login User	Partner Community Login	<input type="checkbox"/>
<input type="checkbox"/> Edit Clone	Partner Community User	Partner Community	<input type="checkbox"/>
<input type="checkbox"/> Edit Del ...	Platform_1	Salesforce Platform	<input checked="" type="checkbox"/>
<input type="checkbox"/> Edit Del ...	Platform_2	Salesforce Platform	<input checked="" type="checkbox"/>
<input type="checkbox"/> Edit Del ...	Platform_3	Salesforce Platform	<input checked="" type="checkbox"/>

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All

File | Edit | Debug | Test | Workspace | Help | < >

UpdateStockAfterOrder.apxt | UpdateStockAfterRestockApproval.apxt | InventoryManager.apxc | EmailNotificationHelper.apxc | InventoryManagerTest.apxc | Run Test | Go

Code Coverage: None | API Version: 64 |

```

1 @isTest
2 public class InventoryManagerTest {
3
4     @testSetup
5     static void setupTestData() {
6         // Create product
7         Re_Plastic_Innovations_Recycled_Product__c product = new Re_Plastic_Innovations_Recycled_Product__c(
8             Stock_Level__c = 50,
9             Threshold__c = 10
10 );
11     insert product;
12
13     // Create orders
14     List<Re_Plastic_Innovations_Order__c> orders = new List<Re_Plastic_Innovations_Order__c>{
15         new Re_Plastic_Innovations_Order__c(Recycled_Product__c = product.Id, Quantity__c = 20),
16         new Re_Plastic_Innovations_Order__c(Recycled_Product__c = product.Id, Quantity__c = 40)
17     };
18     insert orders;
19 }
20
21 @isTest
22 static void testProcessOrderStock() {
    List<Re_Plastic_Innovations_Order__c> orders = [SELECT Id, Recycled_Product__c, Quantity__c FROM Re_Plastic_Innovations_Order__c];
}

```

Logs | **Tests** | Checkpoints | Query Editor | View State | Progress | Problems

Status	Test Run	Enqueued Time	Duration	Failures	Total
✓	707gK000008zmo7	Thu Jul 17 2025 10:42:30 GMT...		0	2

Overall Code Coverage

Class	Percent	Lines
Overall	100%	
EmailNotificationHelper	100%	10/10
InventoryManager	100%	37/37
UpdateStockAfterOrder	100%	2/2
UpdateStockAfterRestockApproval	100%	7/7

Plastic Innovation

Re Plastic Innovatio... | Re Plastic Innovations Orders | Re Plastic Innovations Plastic ... | Re Plastic Innovations Recycl... | More |

Re Plastic Innovations Recycled Product

Plastic Order Record

Related Details

Re Plastic Innovations Recycled Product

Plastic Order Record

Owner: K. Akshay Sagar

Stock Level: 212

Threshold: 100

Price: \$900,000

Stock Low On Product: Sufficient Stock

Created By: K. Akshay Sagar, 7/9/2025, 5:08 AM

Last Modified By: K. Akshay Sagar, 7/15/2025, 8:51 PM

Activity

Filters: All time • All activities • All types | Refresh | Expand All | View All

Upcoming & Overdue

- > Please Look in this Stock ... | No due date | You have an upcoming task
- > Please Look in this Stock ... | No due date | You have an upcoming task

View More

July - 2025 This Month

- > Please Look in this Stock Is Lo... | Jul 9 | You had a task

No more past activities to load.

Conclusion

The RePlastix Innovations Salesforce CRM project successfully demonstrates how technology can be leveraged to automate and streamline complex business processes in the plastic recycling industry. By integrating custom objects, validation rules, flows, Apex logic, and a secure access model, the system addresses critical operational challenges such as inventory shortages, delayed restocking, and inefficient order management. The solution not only enhances productivity and data accuracy but also supports real-time decision-making through automated alerts and reporting tools. Overall, the project meets its objectives by providing a scalable, user-friendly, and secure CRM tailored to the unique needs of RePlastix Innovations, laying a strong foundation for future enhancements and sustainability-focused growth.

Project Resources:

 **Trailhead Profile:**

<https://www.salesforce.com/trailblazer/mzfgcggmwhdr2a4fk>

 **Demo Video:**

<https://drive.google.com/file/d/1DmOVnfb1F-en34u3Lw52uNLAZaxiTNIIm/view?usp=sharing>

 **GitHub Repository (if any):**

<https://github.com/akanapur2425/RePlastix-Innovations-Transforming-Plastic-Waste-into-Sustainable-Solutions>

Submitted by: K. AKSHAY SAGAR