Phase 3: Data Modeling & Relationships

Introduction

A robust and well-structured data model is critical for ensuring that ApexHub can efficiently manage recipes, execution logs, and integrate with Salesforce standard objects. Phase 3 focuses on designing scalable custom objects, defining relationships, enforcing data integrity, and enabling the system to support both automation and reporting.

A clean data model not only improves application performance but also ensures maintainability and reduces future development errors. By standardizing object relationships and validation rules, the team can ensure that data captured by ApexHub is consistent, secure, and meaningful.

Objectives

The objectives of this phase are:

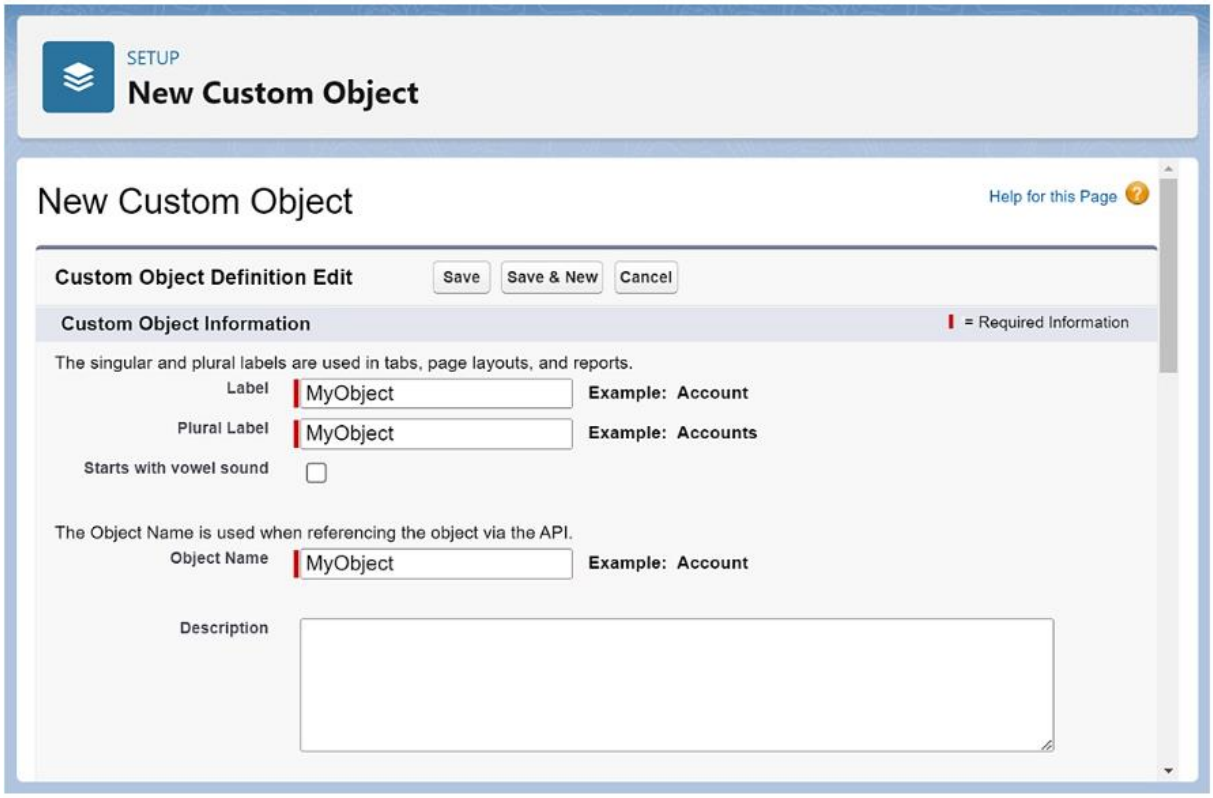
1. Design custom objects to store recipe and execution log information.
2. Establish object relationships to maintain data consistency and enable reporting.
3. Define field-level details, validation rules, and picklists for standardization.
4. Create an ERD (Entity Relationship Diagram) to visualize data structure.
5. Ensure optional integration with standard Salesforce objects such as Account, Contact, and Opportunity.

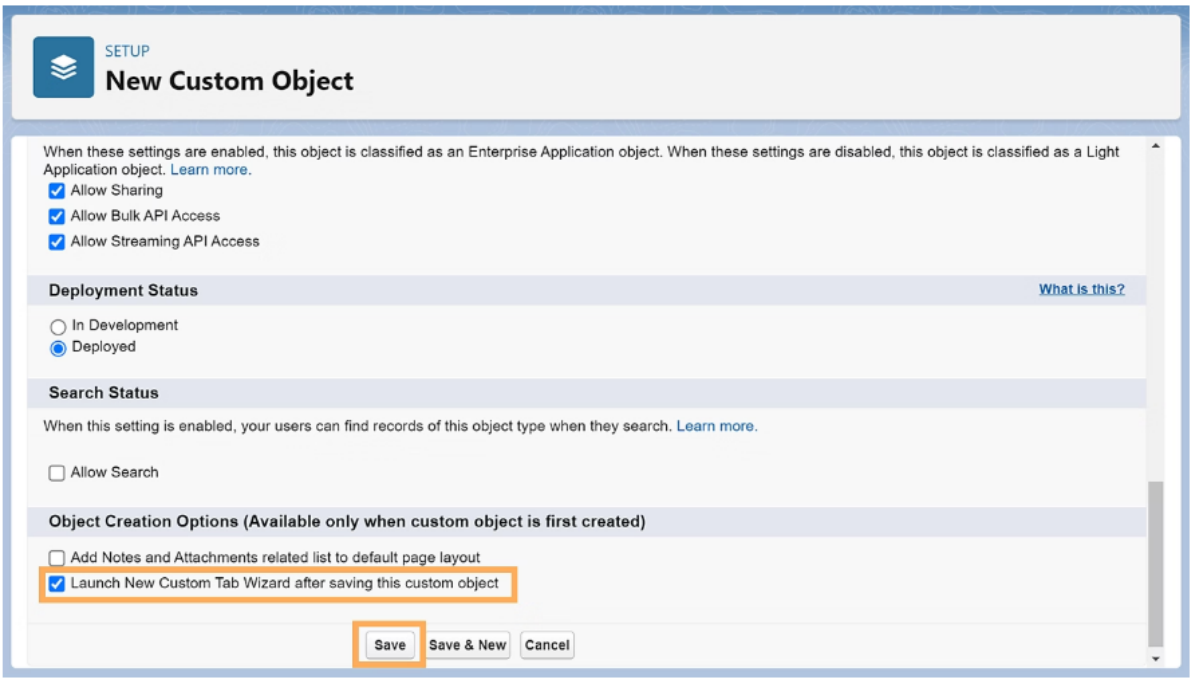
Activities

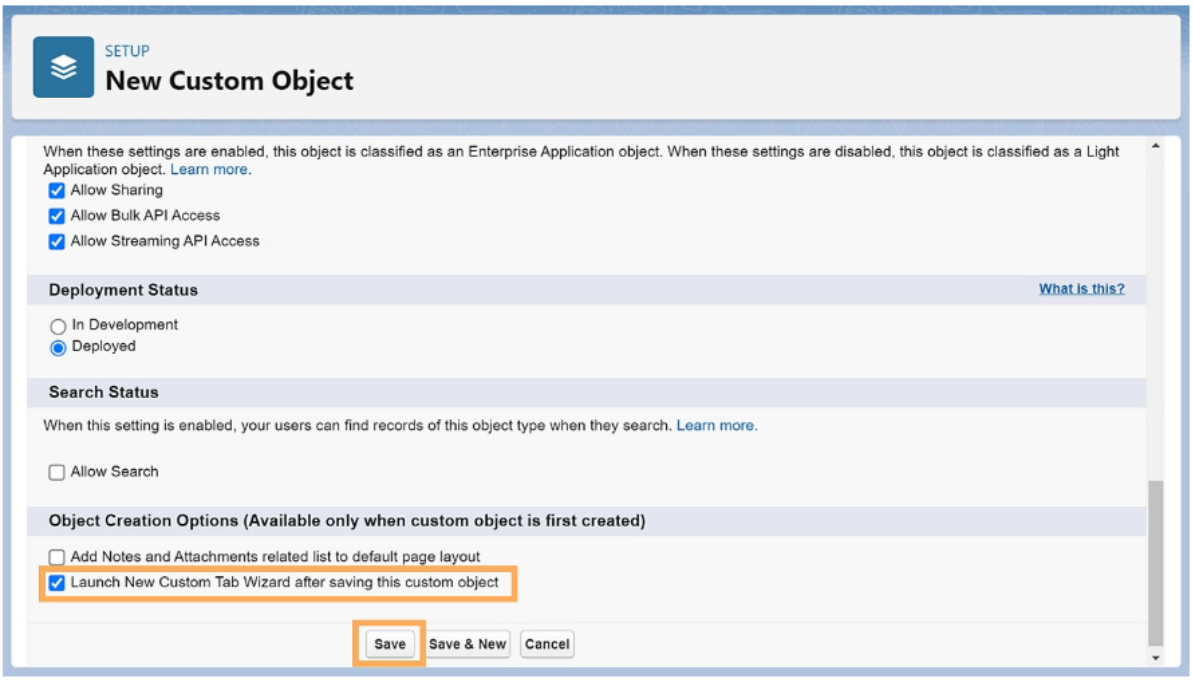
1. Custom Object Design

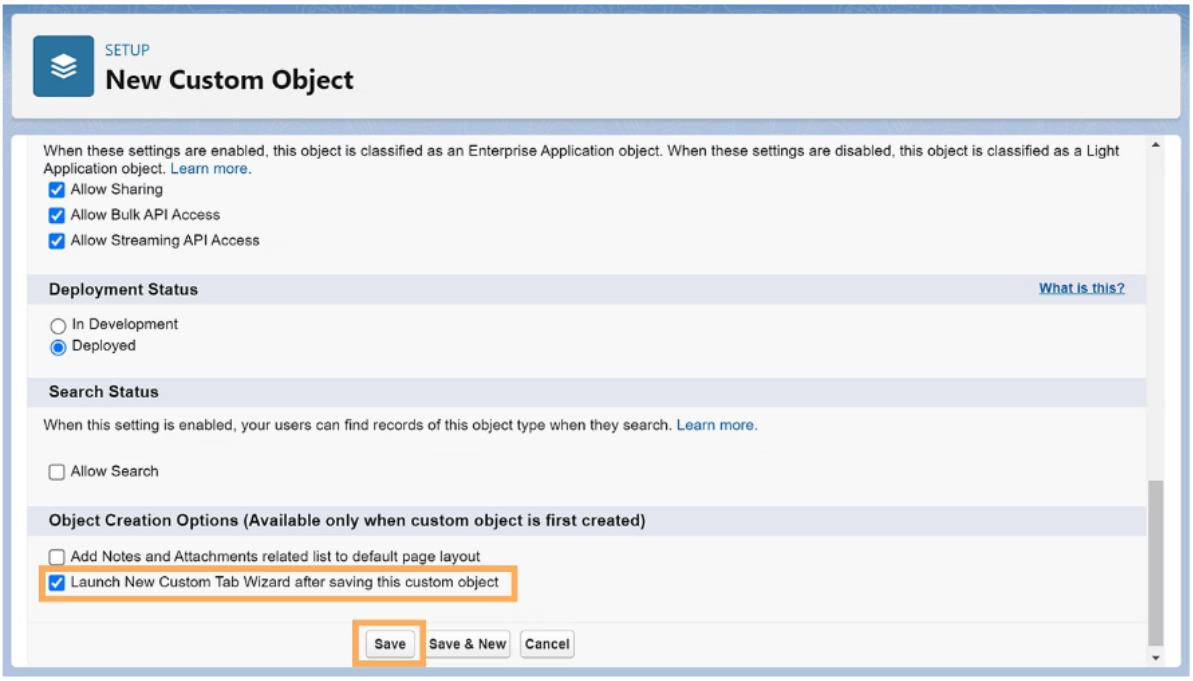
- **Recipe__c:** Stores metadata for each Apex recipe. Key fields include:
    - Name
    - Type (Picklist: Trigger, Batch, LWC, Flow)
    - Category
    - Description
    - LastModifiedBy
- **RecipeLog__c:** Tracks the execution of recipes. Key fields include:
    - ExecutionTime
    - Status (Picklist: Success, Failure, In Progress)
    - Message (Details about execution)
    - Recipe__c (Lookup to Recipe__c)

Purpose: These objects form the core of the ApexHub system, storing both configuration and runtime data.


SETUP
New Custom Object
New Custom Object
Help for this Page
Custom Object Definition Edit
Save
Save & New
Cancel
Custom Object Information
= Required Information
The singular and plural labels are used in tabs, page layouts, and reports.
Label
MyObject
Example: Account
Plural Label
MyObject
Example: Accounts
Starts with vowel sound
The Object Name is used when referencing the object via the API.
Object Name
MyObject
Example: Account
Description


SETUP
New Custom Object
When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. Learn more.
Allow Sharing
Allow Bulk API Access
Allow Streaming API Access
Deployment Status
What is this?
In Development
Deployed
Search Status
When this setting is enabled, your users can find records of this object type when they search. Learn more.
Allow Search
Object Creation Options (Available only when custom object is first created)
Add Notes and Attachments related list to default page layout
Launch New Custom Tab Wizard after saving this custom object
Save
Save & New
Cancel

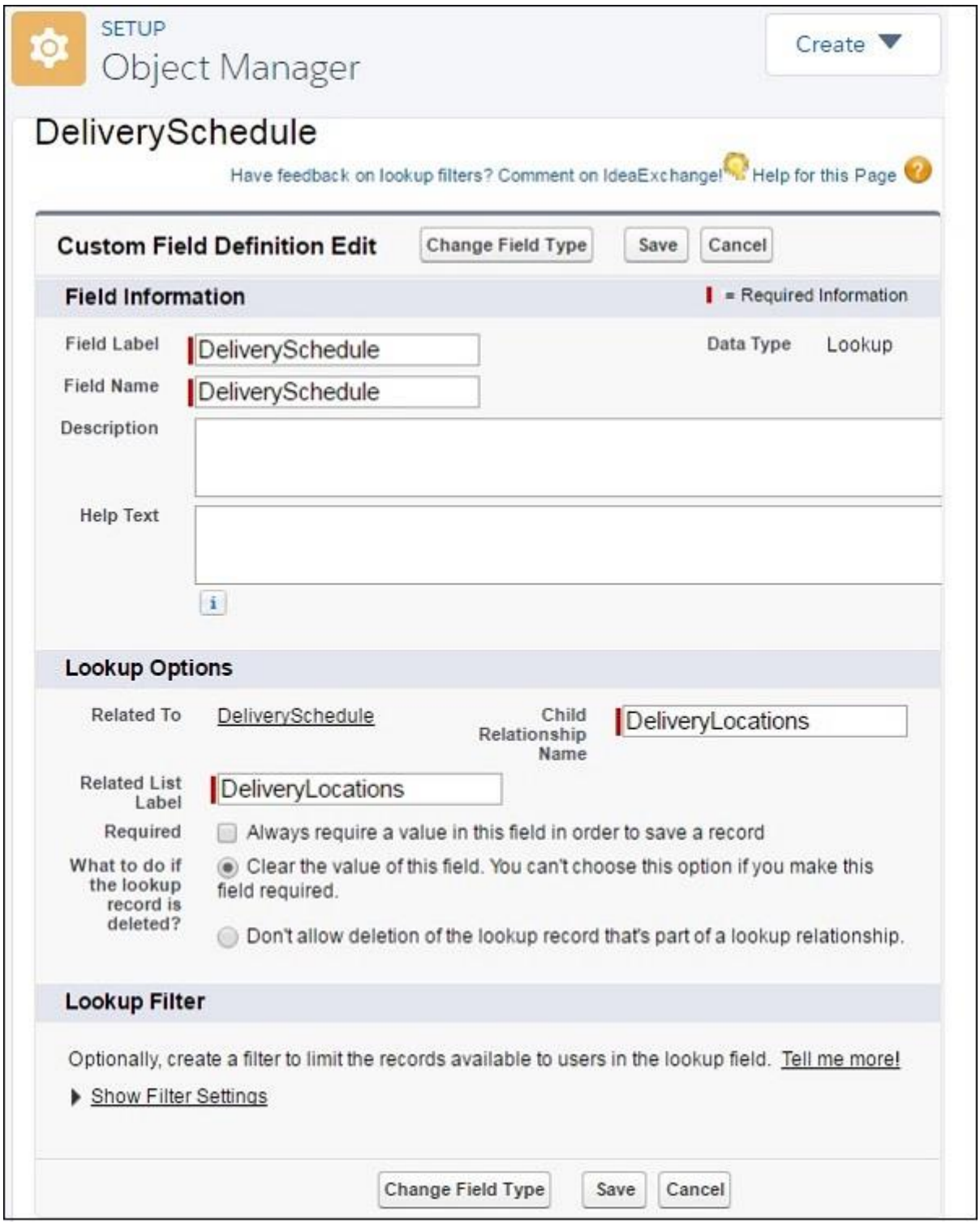




2. Object Relationships

- **RecipeLog__c → Lookup → Recipe__c** (1-to-many)
- Optional integrations with **Account, Contact, Opportunity** for scenarios like linking recipe executions to customers, contacts, or sales opportunities.

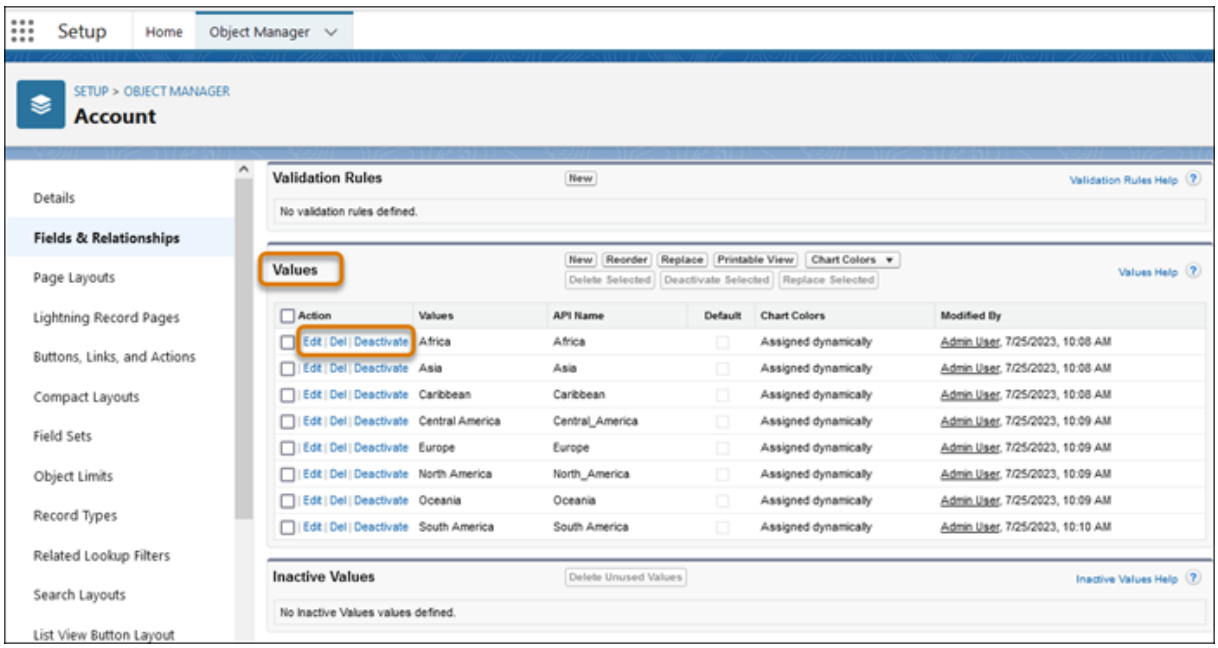
Purpose: This allows tracking multiple executions for each recipe while maintaining clear relational data structure.



3. Field Definitions & Validation

- Picklist fields for **Recipe Type** to standardize entries.
- Validation rules to ensure mandatory fields (e.g., Recipe Name, ExecutionTime, Status) are filled before creating records.
- Field-level security set to ensure sensitive information is visible only to authorized roles.

Purpose: Ensures high data quality and reduces user errors.



4. Entity Relationship Diagram (ERD)

- Visual representation of object relationships:

Recipe__c

|

|---< RecipeLog__c

Account / Contact / Opportunity (optional integration)

Purpose: Provides documentation for developers and stakeholders to understand data flow and dependencies.

5. Initial Validation

- Test creation of Recipe__c and RecipeLog__c records.
- Validate lookups and optional integrations with standard objects.
- Test picklist selections and validation rules for data integrity.

Purpose: Confirms that the data model supports expected operations and enforces business rules.

Deliverables

- Fully defined **custom objects** (Recipe__c, RecipeLog__c).
- Relationships and optional integrations with standard objects.
- **Validation rules** and picklists for standardization.
- **ERD diagram** as documentation.
- Verified field-level security and permissions.

Expected Outcomes

- A scalable and maintainable data model ready for automation and reporting.
- Data integrity enforced via validation rules and picklists.
- Clear visual documentation (ERD) for development and testing.
- Optional integration with Salesforce standard objects for flexibility.