**Project Title: Inquiry Management and Auto-Response System  
Industry**: Suitable for Education, Retail, Services, Finance  
**Project Type**: Salesforce Lead Management Application **Target Users**: Support Staff, Sales Teams, Customers/Students/Clients

**Problem Statement:**Organizations receive frequent inquiries via forms, emails, or chat. Manual tracking delays response and follow-up, resulting in missed opportunities and low customer satisfaction.

**Solution:**Create a Salesforce app to capture all incoming inquiries, assign them automatically, and send instant confirmation or auto-response to the sender.

**Phase 1: Problem Understanding & Industry Analysis**

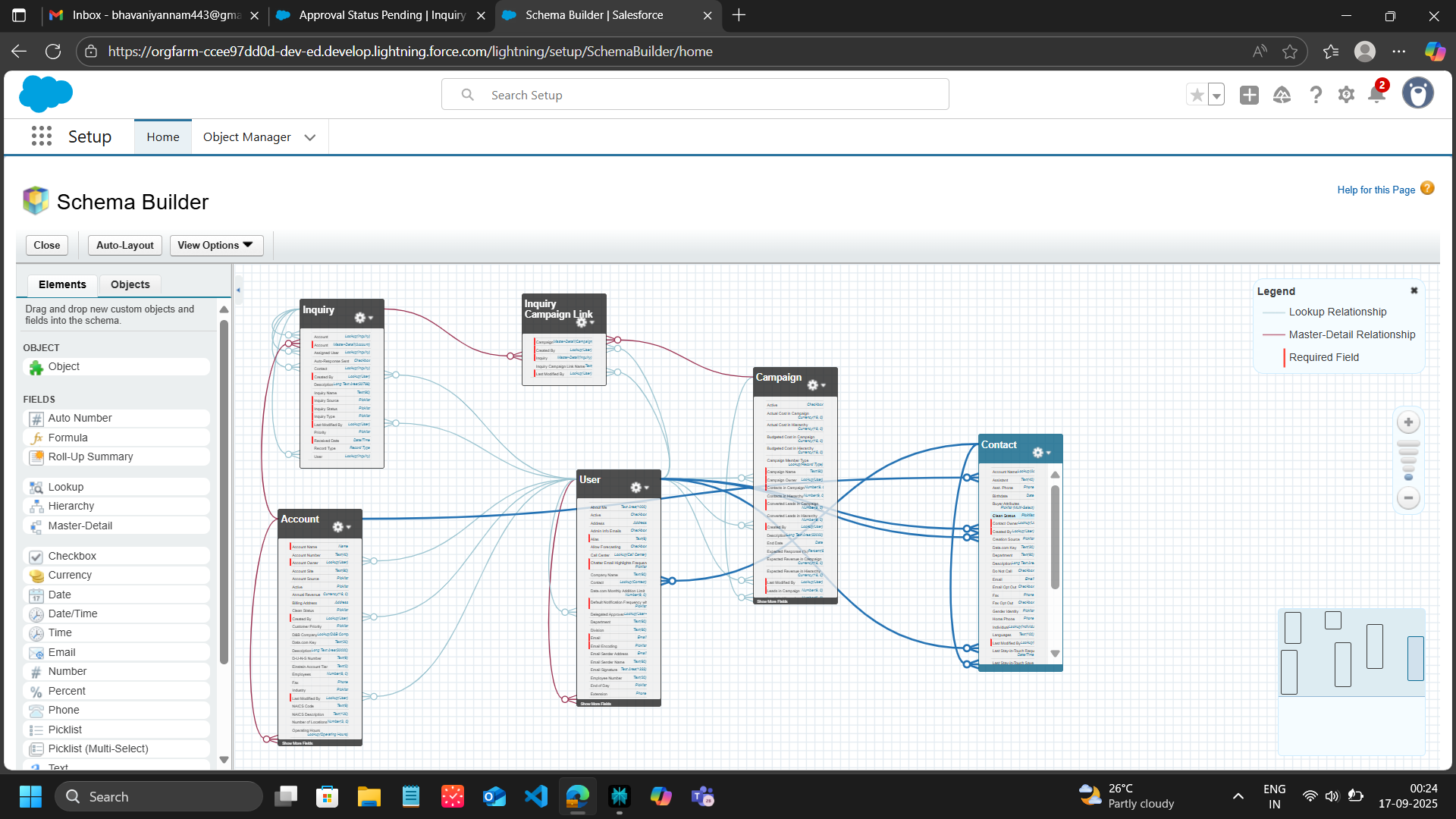
* **Requirement Gathering:** Identify and document the key needs and challenges from all stakeholders.
* **Stakeholder Analysis:**
  + Sales/support team (high interaction with customers)
  + Managers (monitoring metrics and team performance)
  + IT/Admin (system maintenance)
* **Business Process Mapping**: Document the current flow: Customer inquiry → Manual logging → Follow-up → Resolution.
* **Industry-Specific Use Case Analysis:** For example, a college receives course inquiries via website forms; Salesforce logs each inquiry, sends an instant “Thank you, we’ll respond soon,” and routes to admissions staff.
* **AppExchange Exploration:** Explore lead/inquiry management apps for possible extensions or templates.

## **Phase 2: Org Setup & Configuration Steps**

* Selected appropriate Salesforce Edition.
* Set up Company Profile with organization info, fiscal year, time zone, and currency.
* Configured Business Hours and added official Holidays.
* Defined Fiscal Year settings (standard or custom).
* Created and assigned Users, User Licenses, and Profiles.
* Established Role hierarchy and set up Permission Sets.
* Set Organization-Wide Defaults (OWD) for record access.
* Created Sharing Rules for data visibility.
* Set Login Access Policies (login hours, IP ranges, security).
* Set up Developer Edition and Sandboxes for testing and development.

## **Phase 3: Data Modeling and Relationships**

* Designed the core data structure using **Standard Objects** (like Account, Contact, User) and **Custom Objects** as required by business processes.
* Determined and created **Fields** (including data types like text, date, picklist) for each object to capture all relevant information efficiently.
* Used **Record Types** to enable multiple processes or page layouts for the same object depending on business requirements.
* Built **Page Layouts** and **Compact Layouts** to control field visibility and display—ensuring key info is easily accessible for different user roles.
* Utilized **Schema Builder** for visualization and configuration of object relationships for clarity in system design.
* Established critical **Relationships** between objects.
* **Lookup Relationship** for loosely coupled data references.
* **Master-Detail Relationship** for tightly linked objects requiring parent-child data control.
* **Hierarchical Relationship** where applicable, primarily for user role hierarchies.



* Created **Junction Objects** when many-to-many relationships were needed between entities like Campaigns and Inquiries.
* Integrated **External Objects** when connecting with outside data sources for business functionality.
* This phase ensures all data required by business processes is structurally organized, easily accessible, and ready for automation, reporting, and user interaction in later stages.

## **Phase 4: Process Automation (Admin)**

This phase covers automation of key business processes for Inquiry Management using Salesforce declarative tools.

**Validation Rules**  
Ensured mandatory fields are completed and data quality checks. For example, making Inquiry Type and Customer Email required.

**Workflow Rules**  
Created workflows to auto-assign inquiries, send auto-response emails, and create follow-up tasks.

**Process Builder**  
Automated updates and notifications when inquiry status changes, updating related records as needed.

**Approval Process**  
Enabled approval routing for high-value or high-priority inquiries, with status updates and notifications post approval or rejection.

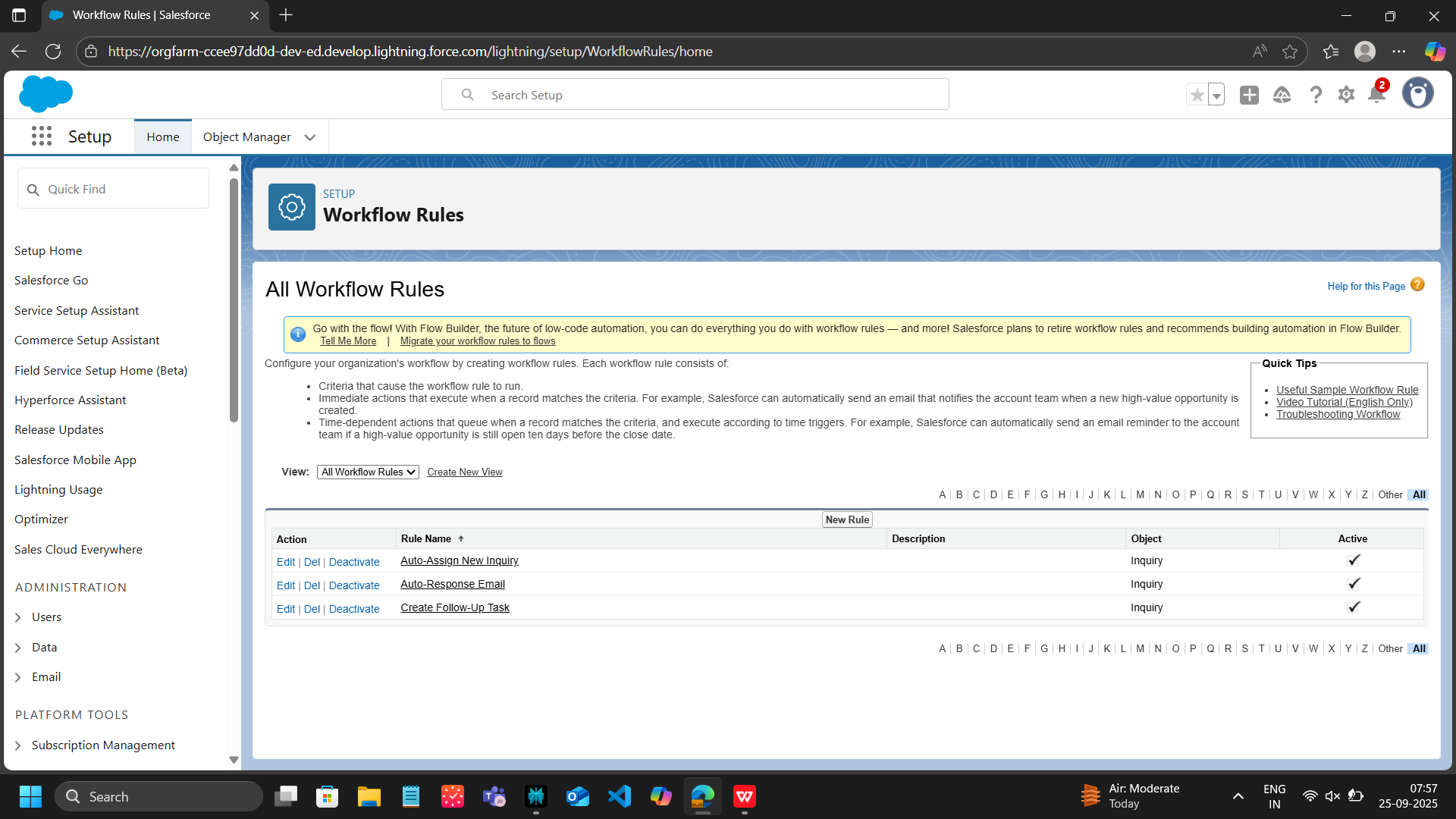
**Flow Builder**  
Developed screen flows for guided input, record-triggered flows for backend automation, and scheduled flows for periodic tasks.

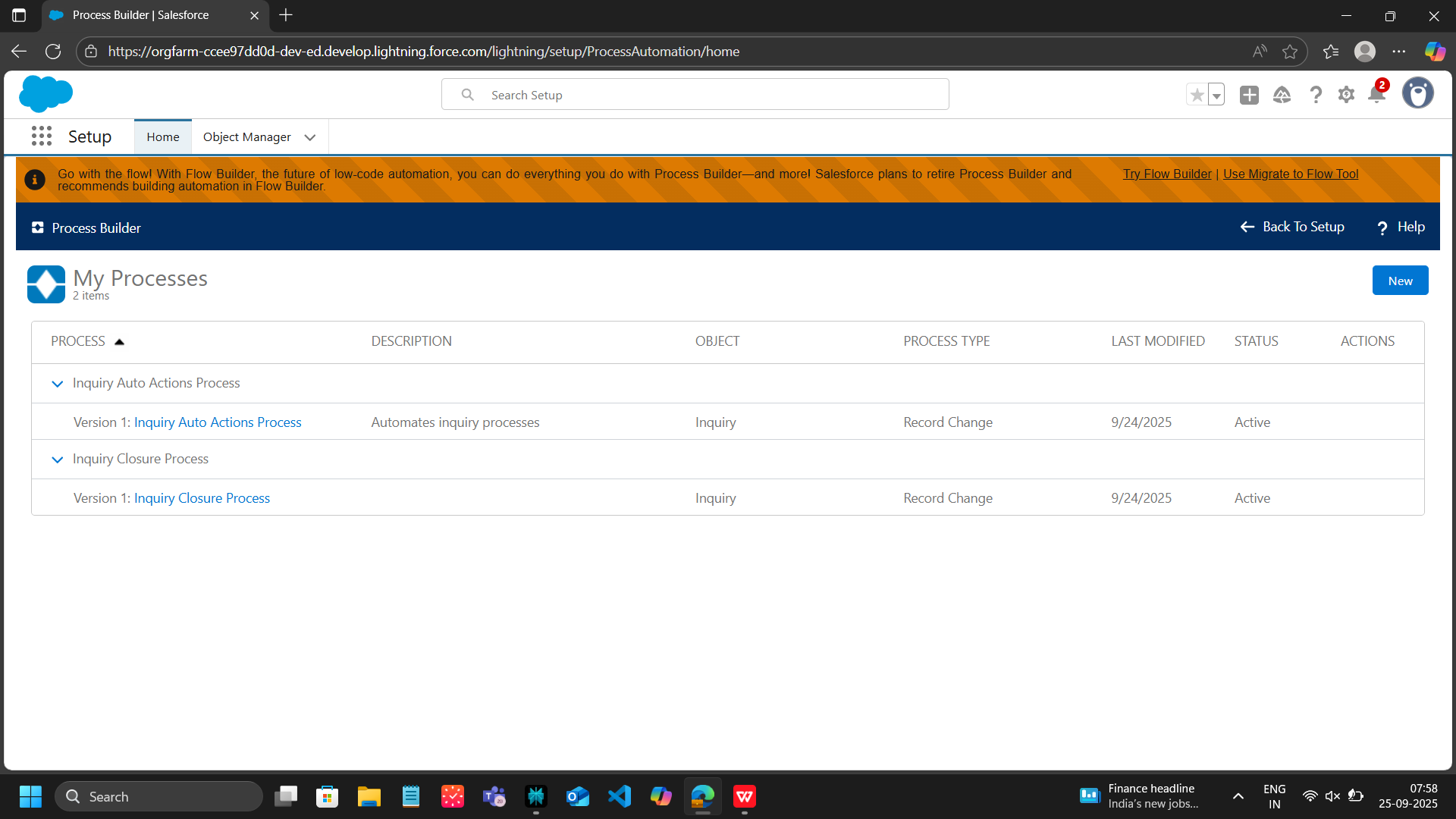
**Email Alerts**  
Configured email templates and alerts linked to workflow and process builder for customer and internal notifications.

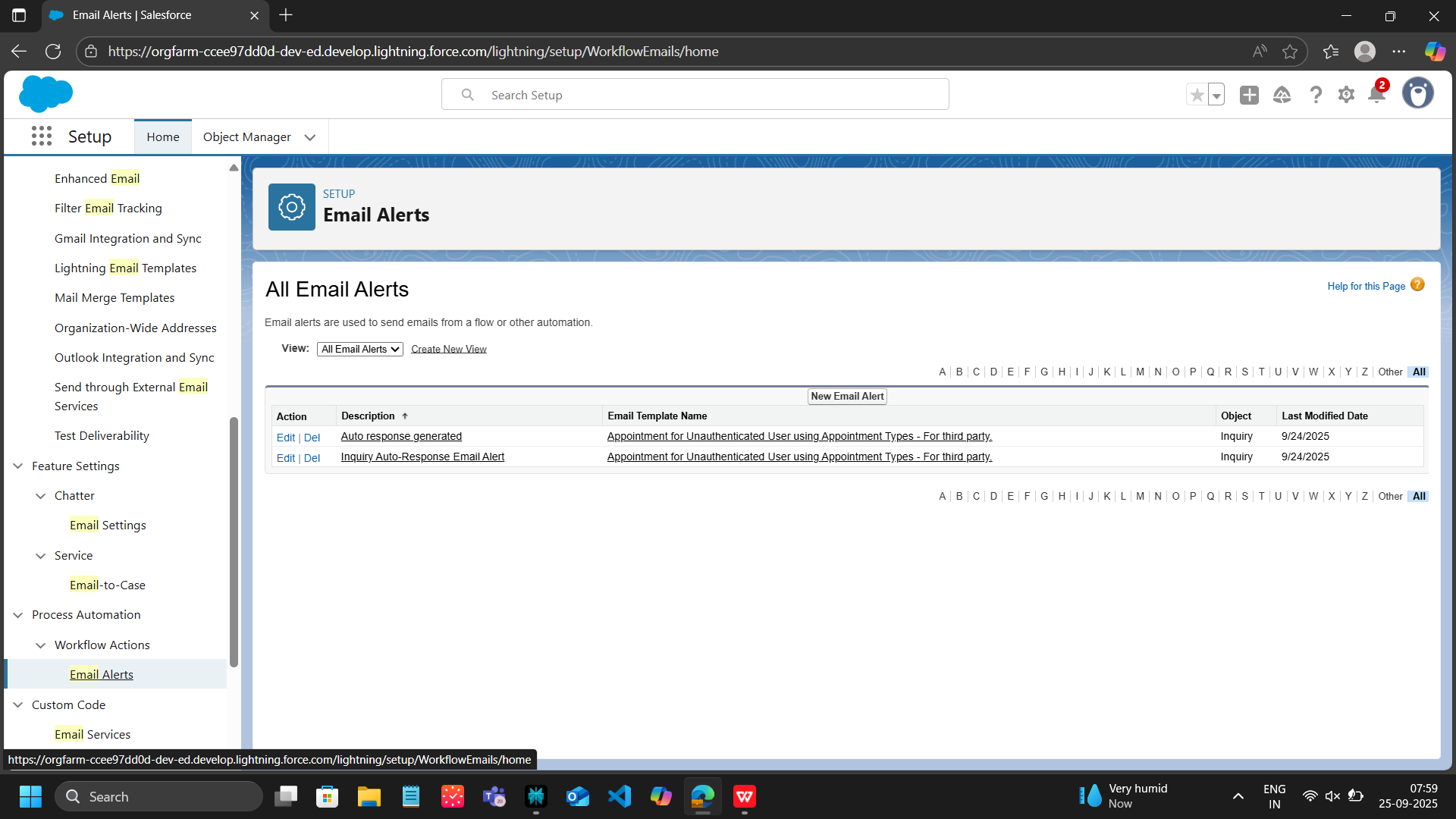
**Field Updates**  
Automatically updated status fields, checkboxes, and other flags to maintain data consistency.

**Tasks**  
Generated follow-up tasks automatically to ensure timely customer engagement.

**Custom Notifications**  
Delivered real-time, in-app notifications to users for important inquiry events.







**Phase 5: Apex Programming (Developer)**

# **Apex Classes and Responsibilities**

* **InquiryService**  
  Contains main business logic for auto-assigning and sending email responses for inquiries.
* **InquiryTriggerHandler**  
  Manages trigger logic for Inquiry records on insert and update events.
* **InquiryTrigger**  
  Salesforce trigger delegating inquiry-related DML events to InquiryTriggerHandler.
* **InquiryBatch**  
  Batch Apex class for bulk asynchronous processing of inquiries.
* **InquiryQueueable**  
  Queueable Apex class for asynchronous inquiry processing tasks.
* **InquiryAsync**  
  Async class handling email confirmations sent asynchronously.
* **InquiryScheduler**  
  Scheduled Apex class that triggers periodic batch or queueable jobs.

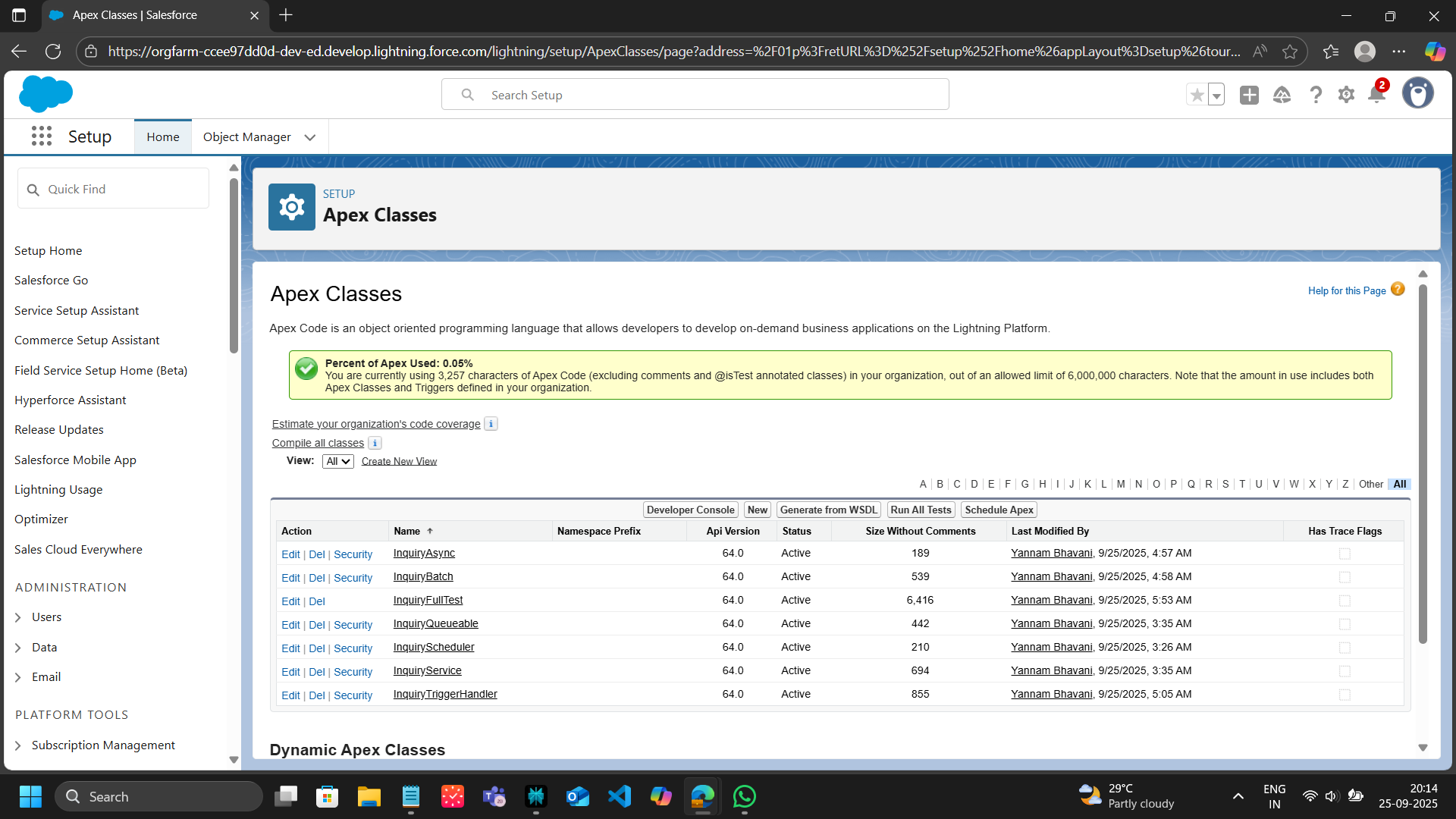
## **Testing Overview**

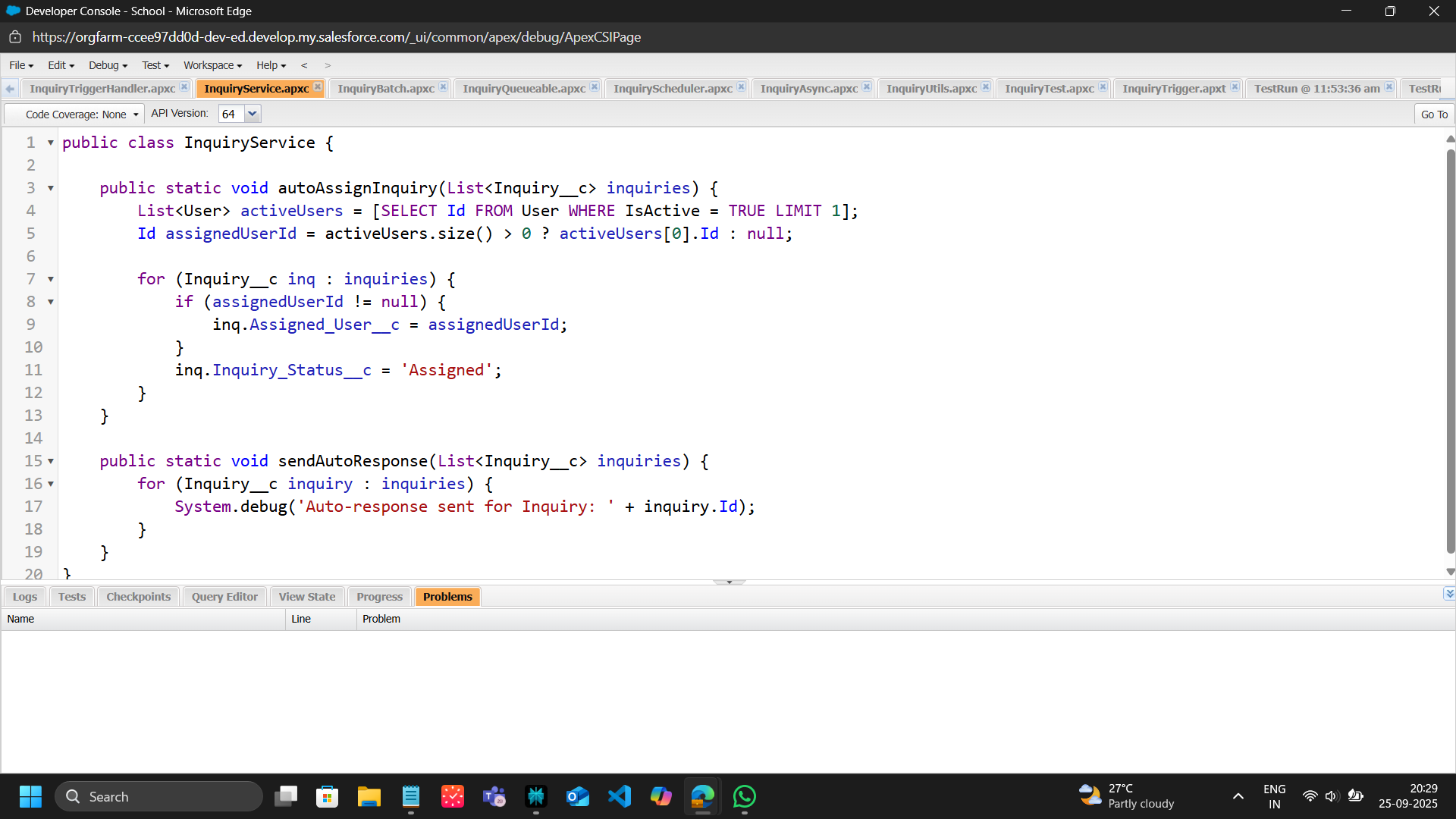
Testing covers all classes with focus on:

* Validations of core functionality, including status changes and automated assignments.
* Bulk processing scenarios for batch and trigger logic.
* Asynchronous operations including queueable, async, and scheduler executions.
* Edge cases, error handling, and null/empty input scenarios.

Code Coverage Status

The overall code coverage achieved is approximately 74%. Critical core components such as service classes, queueables, and schedulers have full coverage. Areas needing more tests include batch processing, async methods, and trigger handler logic to improve coverage towards full readiness.

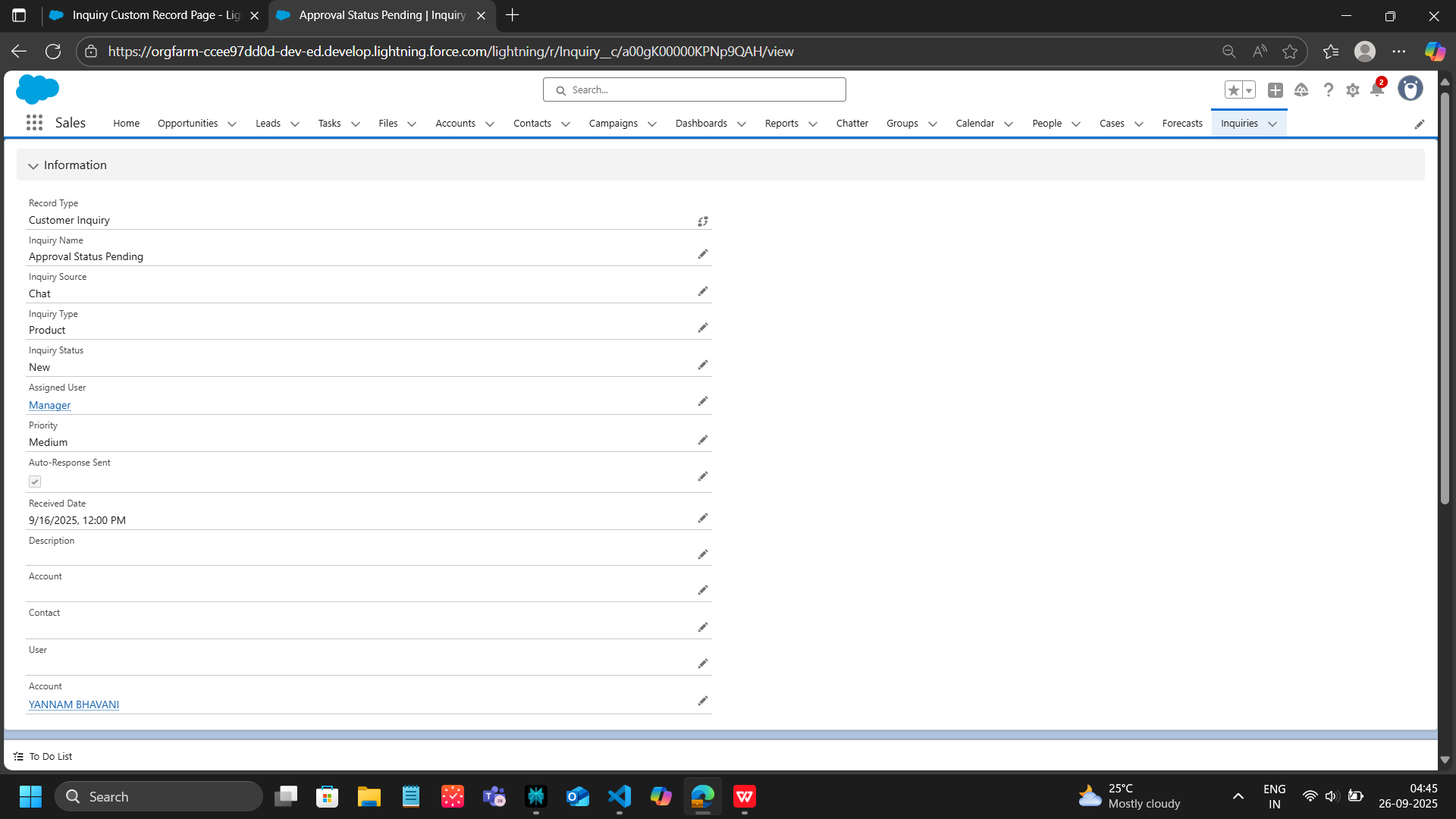


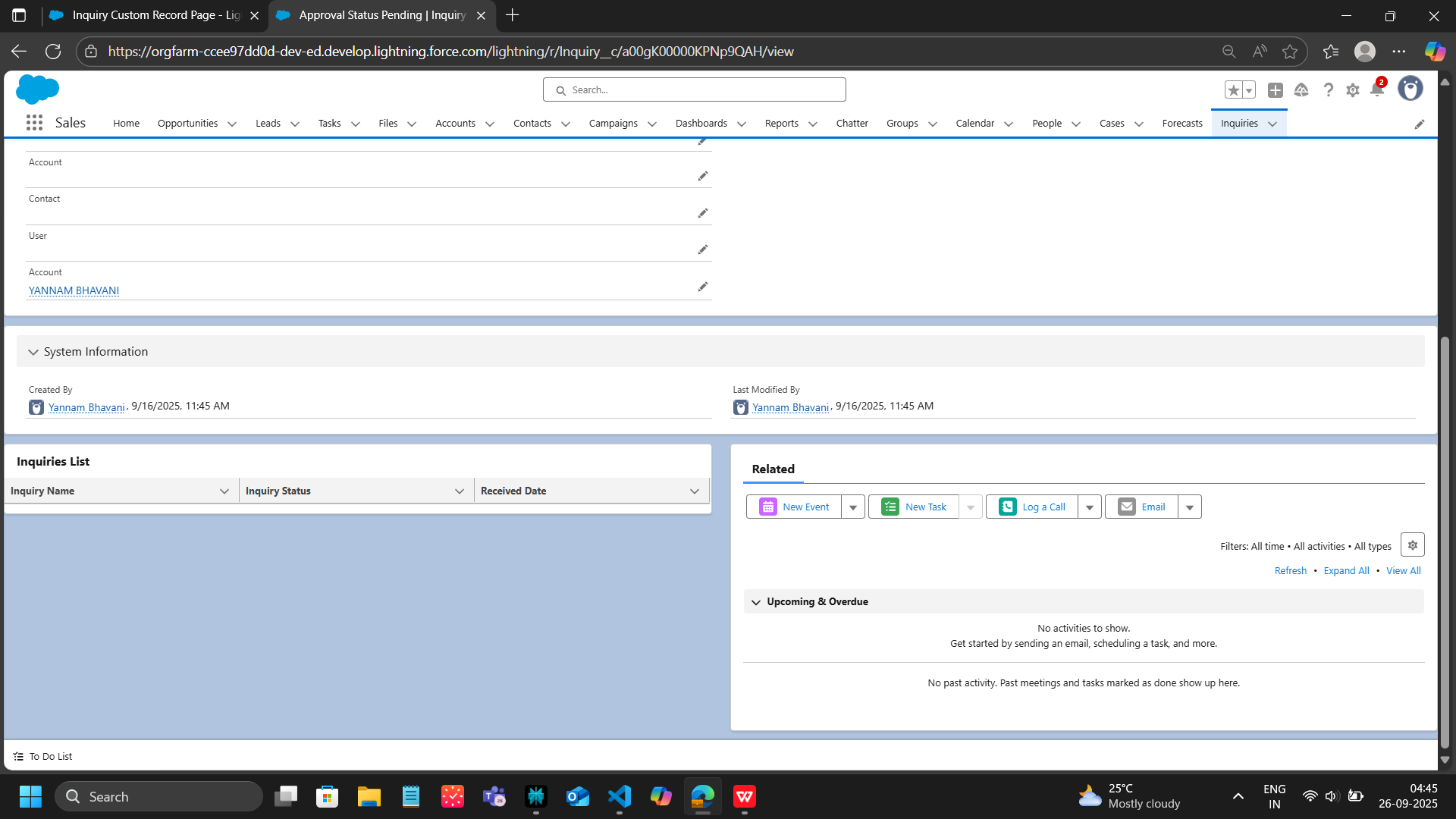


## **Phase 6: User Interface Development**

## **Components Created and Used:**

## Customized the Inquiry record page layout. Created Inquiry Custom Record Page with relevant information and system details. Organized sections into tabs for related inquiries, approval history, and campaign links. Developed a custom LWC named inquiryList to display Inquiry records in a data table, integrated with Apex controller for dynamic data fetch. Apex controller InquiryController created to provide server-side data for LWC. Utilized wire service to fetch inquiry data reactively in LWC.

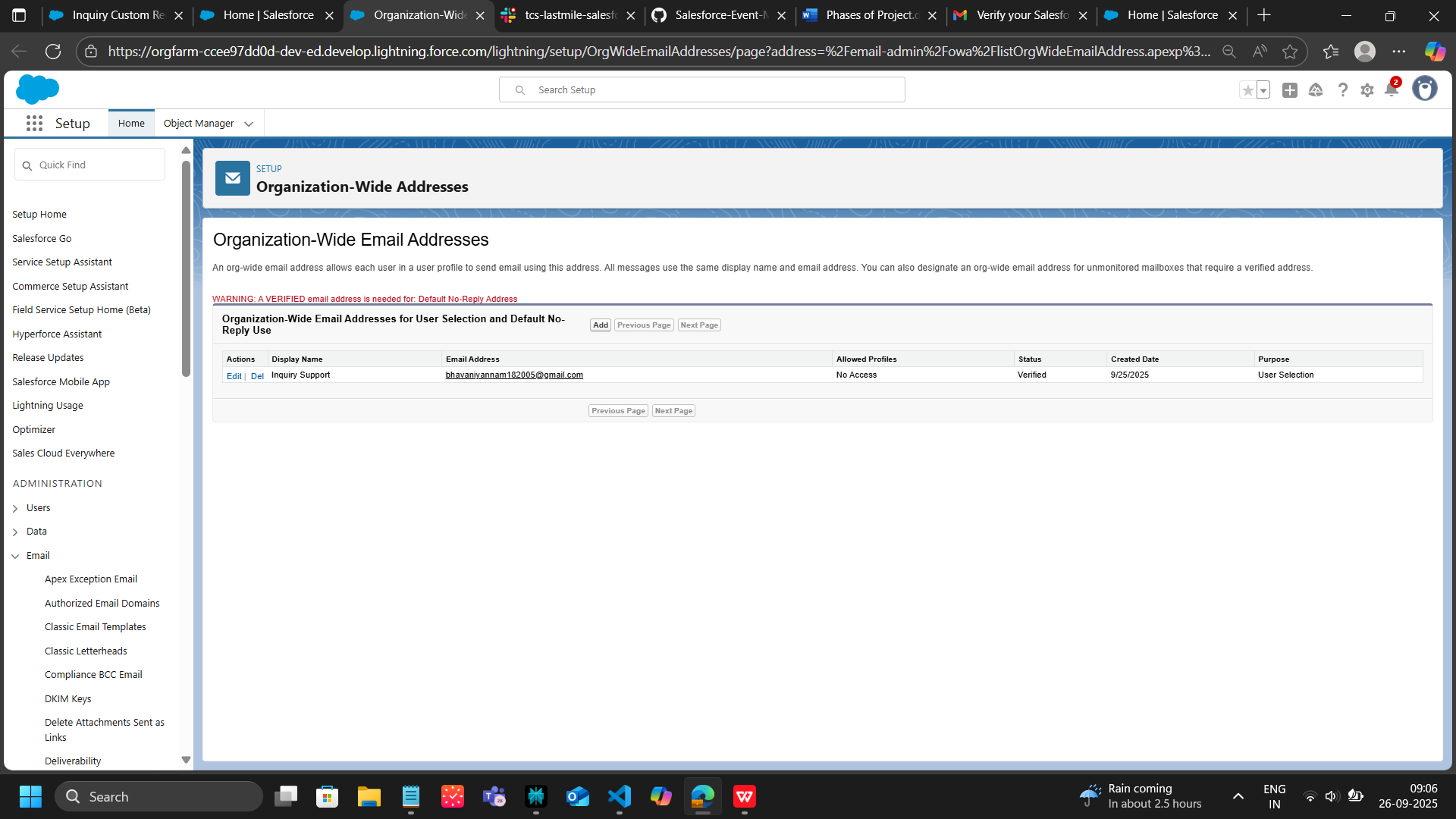


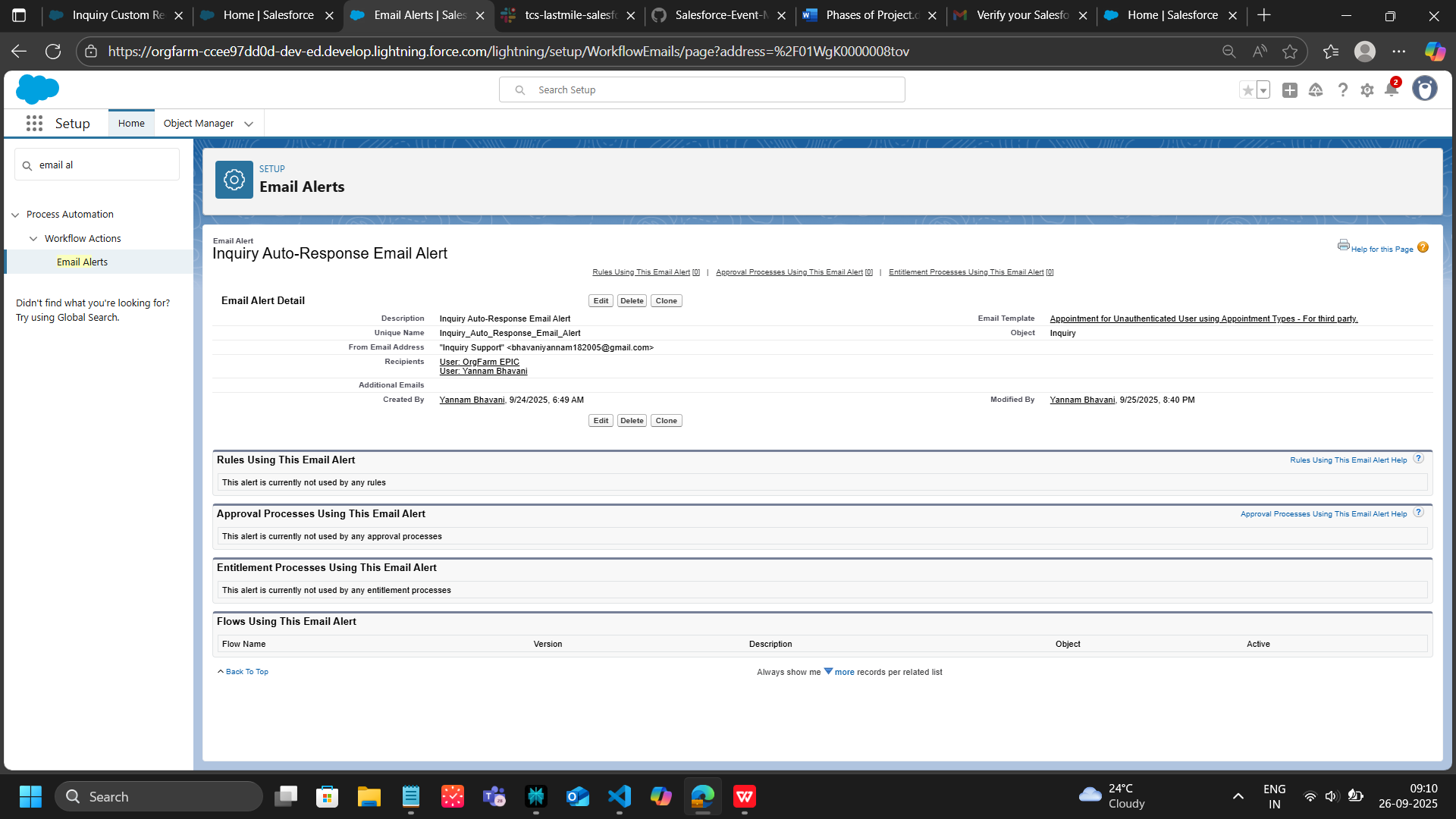


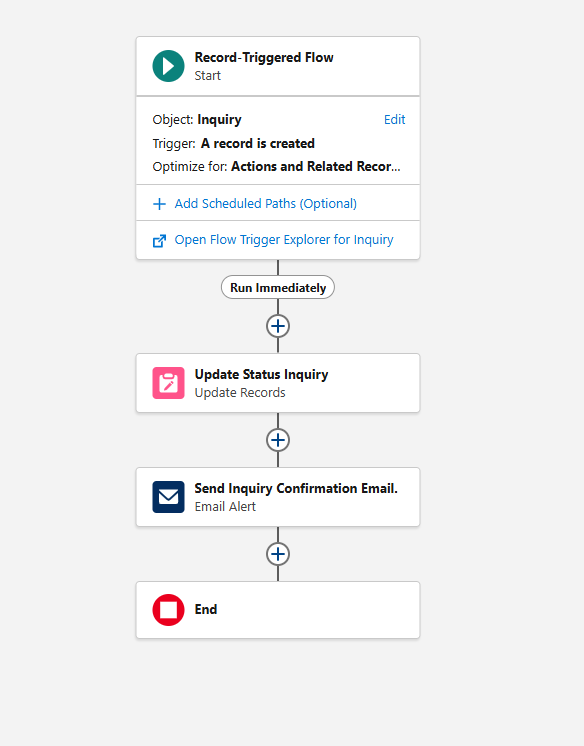
## **Phase 7: Integration & External Access**

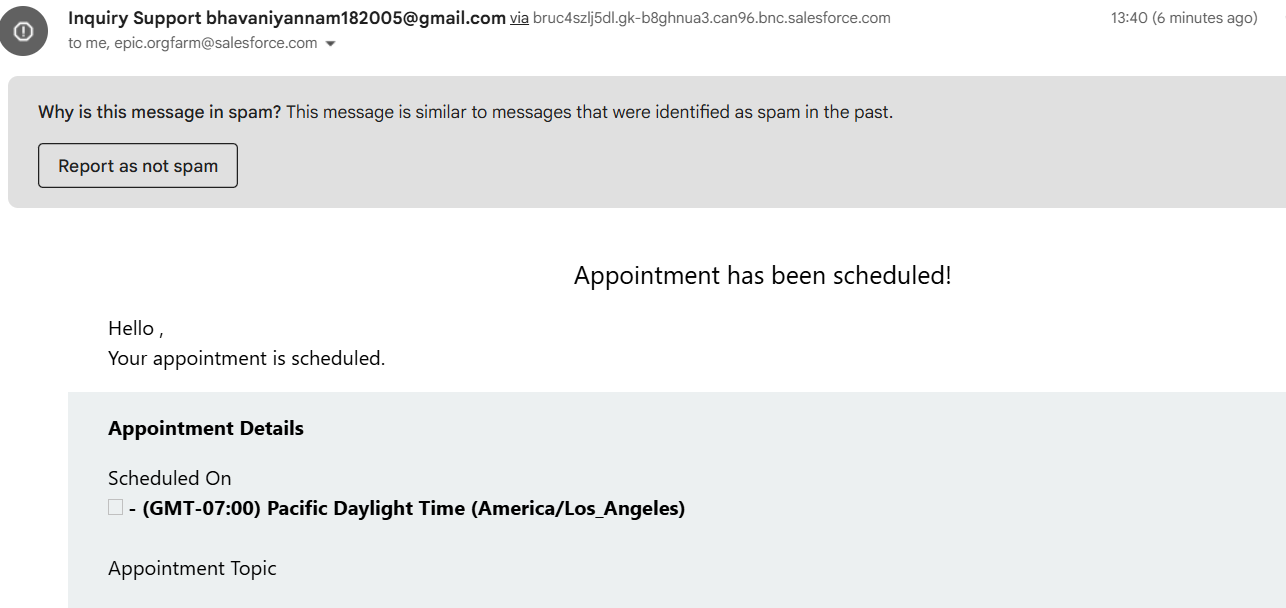
In this phase, the key administrative configurations were established to enable automated email confirmations and ensure secure system communication. The main activities included:

* Configured an Organization-Wide Email Address to be used as the "From" address for system-generated emails.
* Updated existing Email Alerts to use the new Organization-Wide Email Address as the sender.
* Reviewed and verified the configuration of Record-Triggered Flows to ensure the Email Alert action triggers correctly after Attendee or Event record creation.
* Performed end-to-end testing by creating new Event and Attendee records, verifying that confirmation emails are sent successfully.
* Confirmed emails are sent using the configured Organization-Wide Email Address and checked the accuracy and completeness of email content.



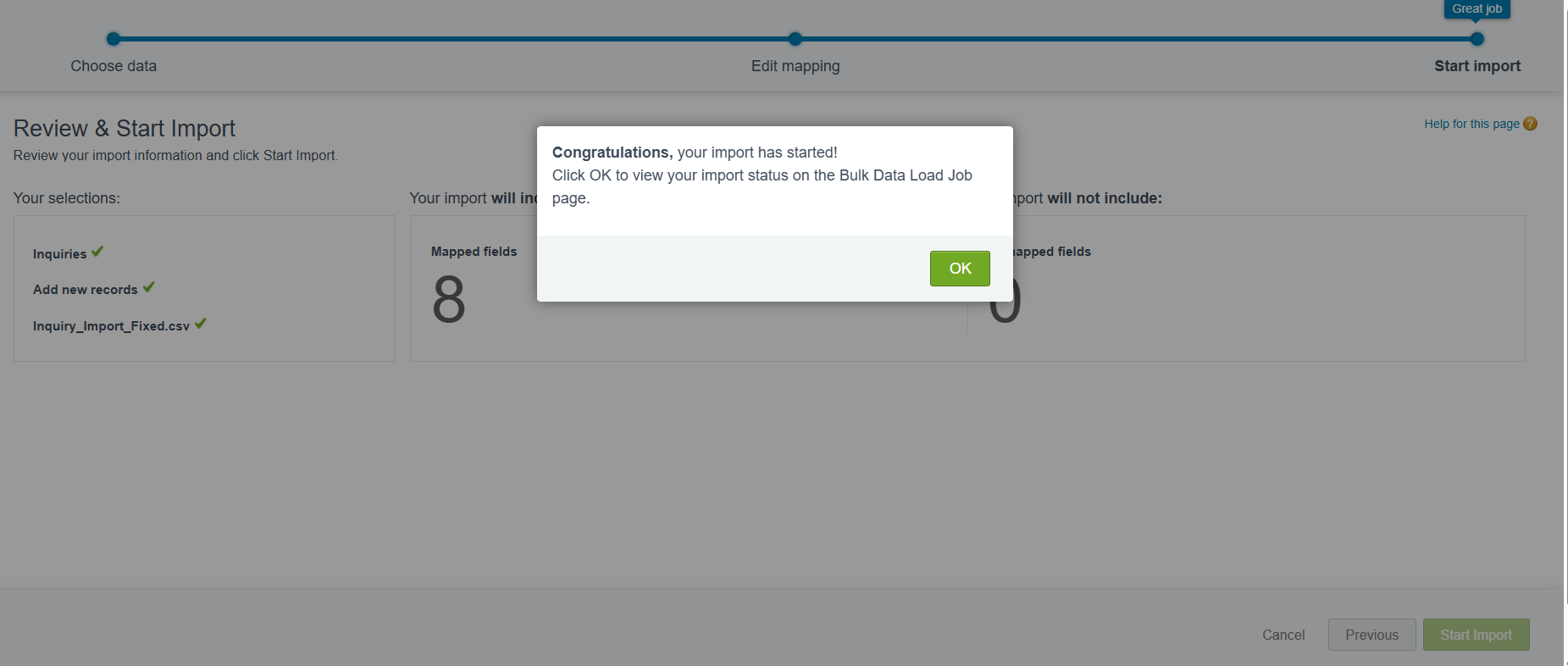


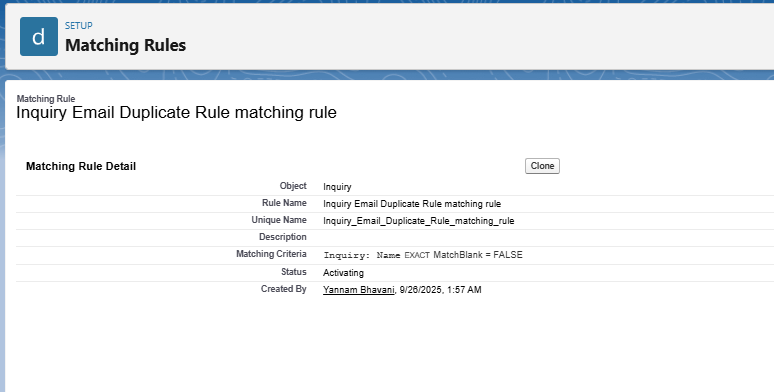




**Phase 8 - Data Management & Deployment**

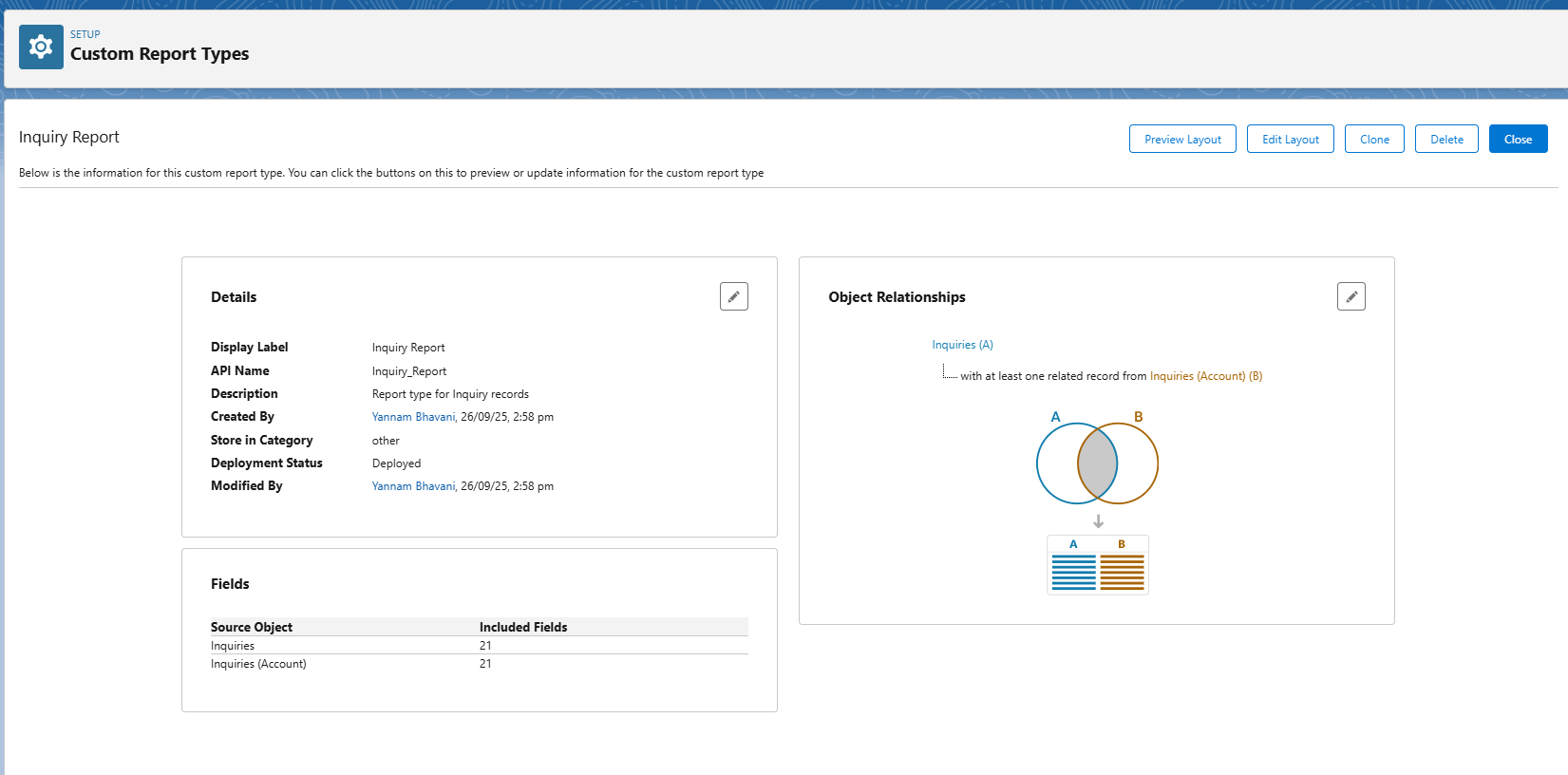
* Data Import Wizard for easy data upload into Salesforce objects.
* Data Loader for large scale or complex data operations.
* Duplicate Rules to prevent or alert on duplicate data records.
* Data Export & Backup for scheduled data preservation.
* Change Sets for deploying metadata changes between orgs.
* Unmanaged and Managed Packages for distributing and upgrading Salesforce apps.
* ANT Migration Tool for scripted, automated deployment using XML metadata.
* VS Code & Salesforce DX (SFDX) for modern source-driven development and version control.





**Phase 9 - Reporting, Dashboards & Security Review**

* Reports: Create Tabular, Summary, Matrix, and Joined reports.
* Report Types: Customizable templates defining reports’ data structure.
* Dashboards: Interactive visuals displaying reports’ data.
* Dynamic Dashboards: Personalized dashboards shown according to user permissions.
* Sharing Settings: Record access controls like organization-wide defaults, sharing rules.
* Field Level Security: Restriction on visibility and editing of individual fields.
* Session Settings: Security controls including session timeouts.
* Login IP Ranges: Restrict login access by IP addresses.



**Phase 10: Final Presentation & Demo Day**Deliver a concise project pitch and live demo. Collect stakeholder feedback, provide thorough handoff documentation, and showcase the project professionally on LinkedIn or portfolio for career visibility.