Student Management System Using Salesforce CRM

Phase 1: Problem Understanding & Industry Analysis

1. Requirement Gathering

Educational institutions require a system to manage student details, course enrollments, attendance, grades, and fees in a centralized manner. The requirements were gathered from administrators, teachers, and students to identify challenges in manual management.

2. Stakeholder Analysis

- **Students** – Need easy enrollment, grade tracking, and fee updates.
- **Teachers** – Need to record attendance, manage grades, and monitor performance.
- **Administrators** – Require centralized control over student data, reporting, and fee collection.
- **Parents/Guardians (Future)** – Require visibility into student performance and attendance.

3. Business Process Mapping

- **Student Admission** → Creation of student record.
- **Course Enrollment** → Mapping students to multiple courses.
- **Attendance Tracking** → Daily attendance marking per course.
- **Grade Management** → Entering marks and calculating GPA.
- **Fee Management** → Recording payments, balances, and due alerts.
- **Reporting & Analytics** → Generating performance, fee, and attendance reports.

4. Industry-specific Use Case Analysis

- Current systems in schools and colleges rely on **spreadsheets/manual records**, causing redundancy and errors.
- Industry trend is shifting to **cloud-based CRMs** like Salesforce for data accuracy and accessibility.
- Similar use cases:
  - University student portals.

- Training centers managing course enrollments.
- Coaching institutes tracking performance.

5. AppExchange Exploration

- Explored Salesforce **AppExchange** solutions for education management.
- Found apps like:
    - **Salesforce Education Cloud** – Comprehensive education CRM.
    - **Student Success Hub** – Focused on K-12 student outcomes.
    - **Attendance & Fee Management Apps** – Useful but often paid solutions.
- Decided to **build a custom Student Management System** tailored for specific institutional needs instead of relying fully on prebuilt paid apps.

Phase 2: Org Setup & Configuration – Student Management System

1. Salesforce Edition

- **Developer Edition** (Dev Org) used for project setup.
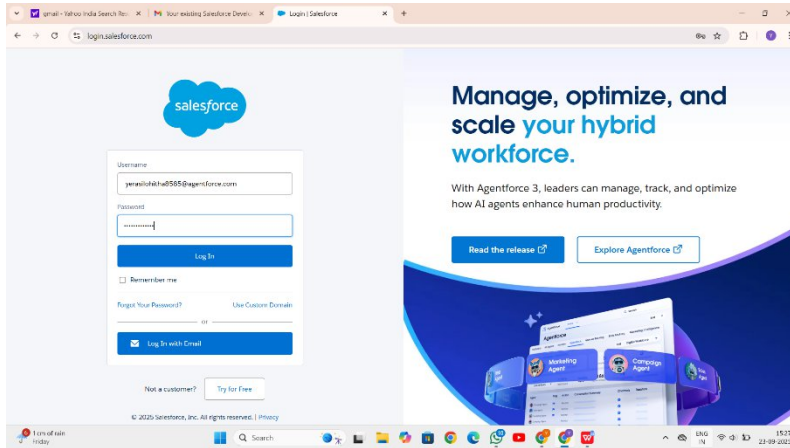
2. Company Profile Setup

- Configured **Institution Name, Address, Default Time Zone, Currency, and Locale**.
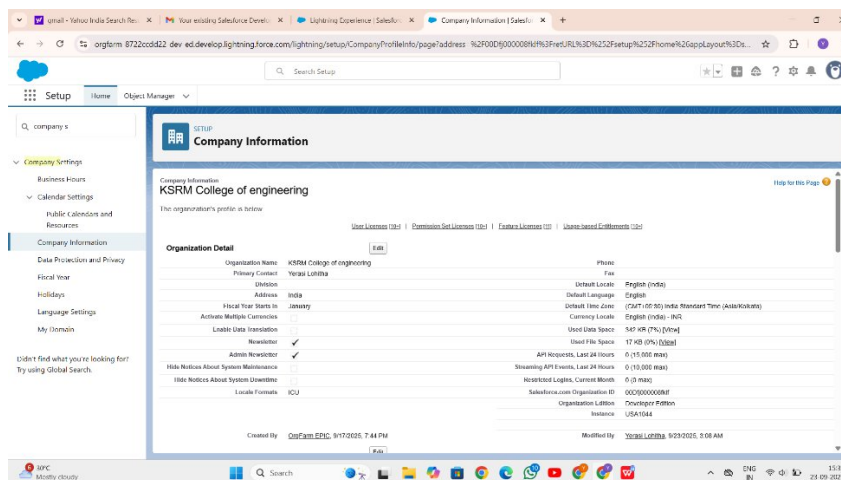
3. Business Hours & Holidays

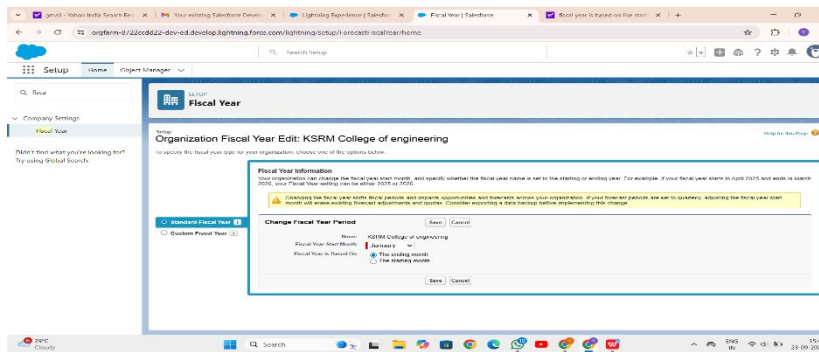- Defined **Standard Business Hours** (9 AM – 6 PM, Mon–Sat).
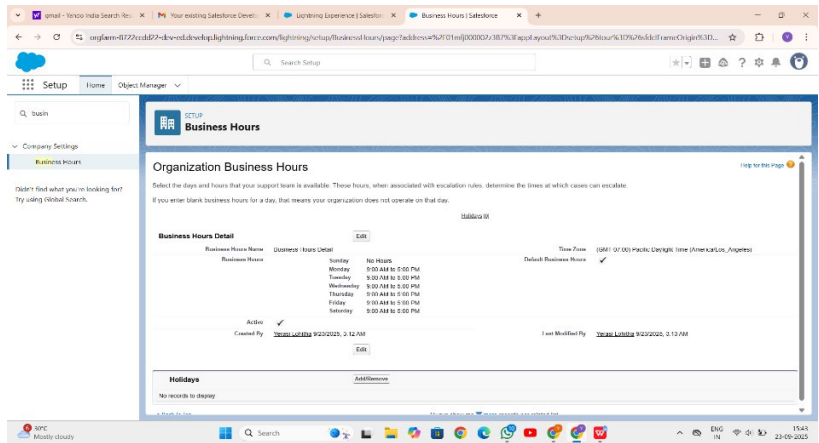
- Added **holidays** (e.g., Independence Day, New Year).
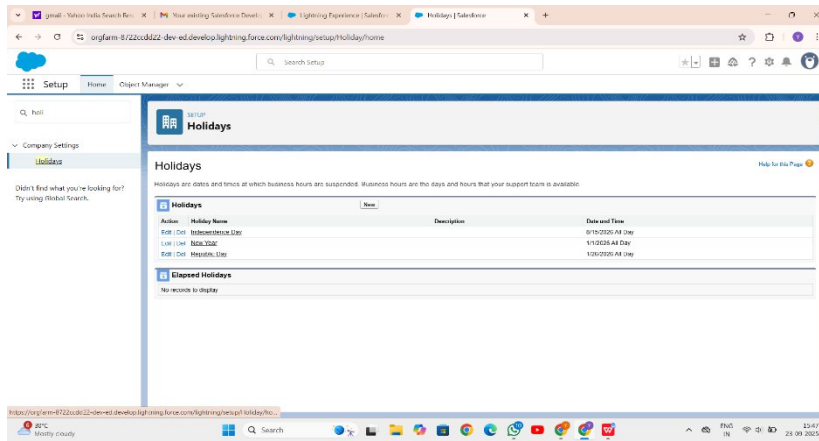
4. Fiscal Year Settings

- Configured **Standard Fiscal Year** (Jan–Dec).

5. User Setup & Licenses

- Created sample users with **Salesforce licenses**:

    o **System Administrator** – Full access.

    o **Faculty User** – Limited teaching access.

    o **Student User** – Read-only/student data access.

6. Profiles

- Created/Modified Profiles:

    o **System Administrator** (default full access).
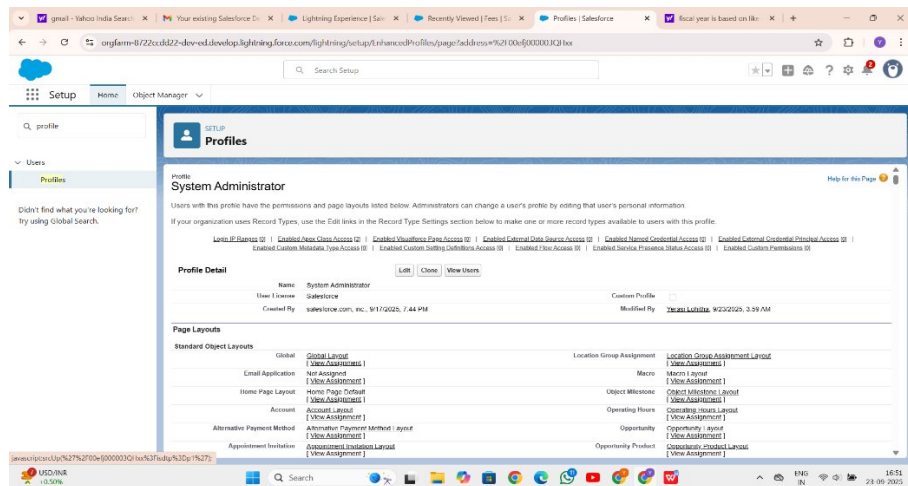
    o **Teacher Profile** – Access to Students, Courses, Attendance, Grades.
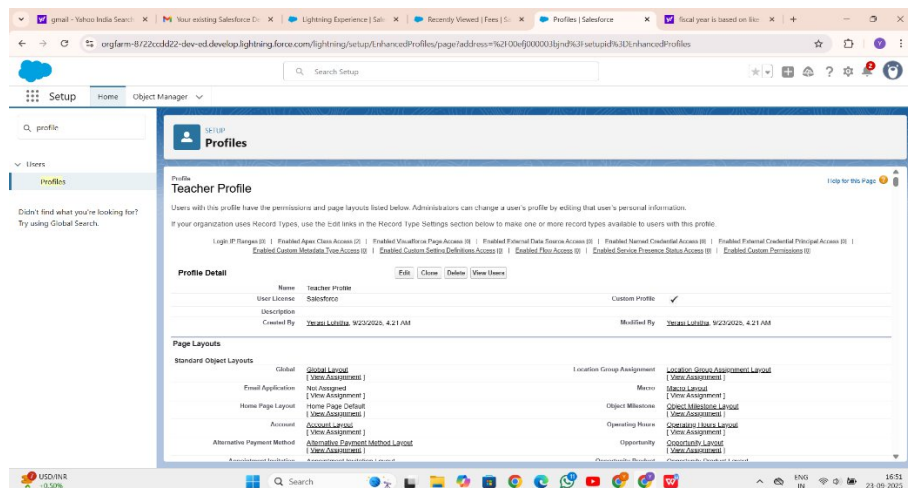
- **Student Profile :** To check the student details in the read only mode.
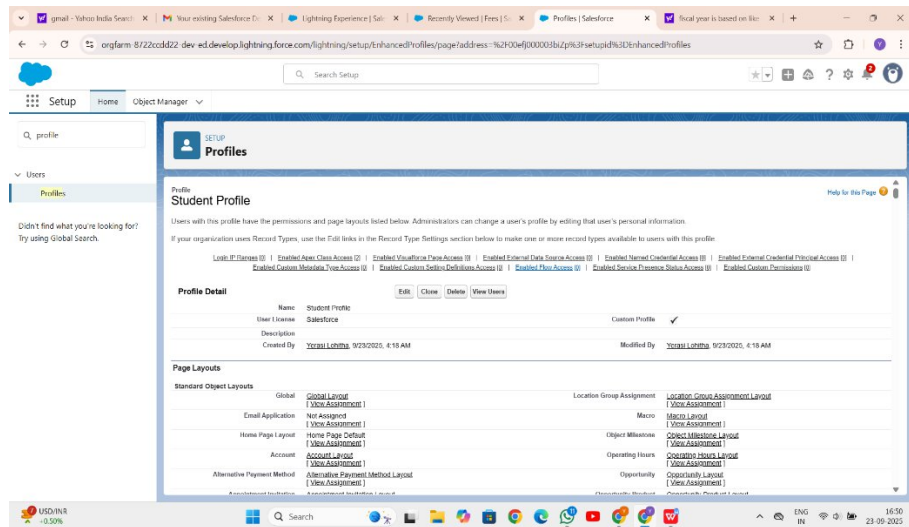
- **Finance Profile** – Access to Fees object.
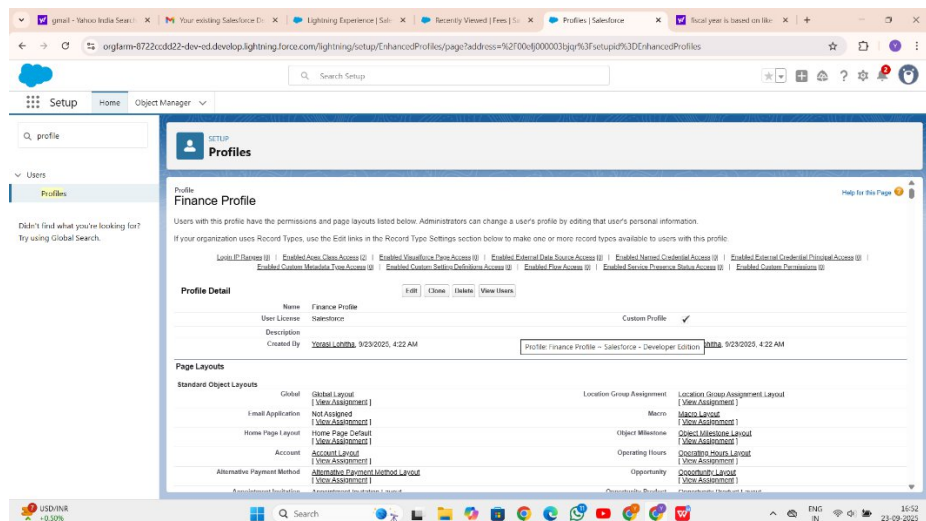
7. Roles

- Defined Role Hierarchy:

  - **Principal (Top Role)**
    - HOD
    - Finance Officer
    - Teacher
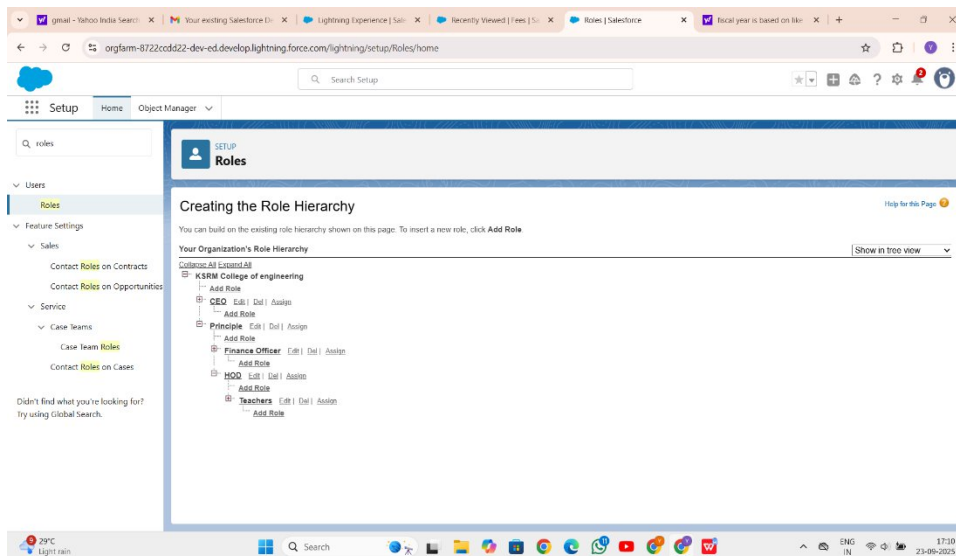
8.Permission Sets

- Created Permission Sets for additional access:
    - **Report Access** – Extra reporting permissions.
    - **Fee Management Access** – Extended access to Fees object.
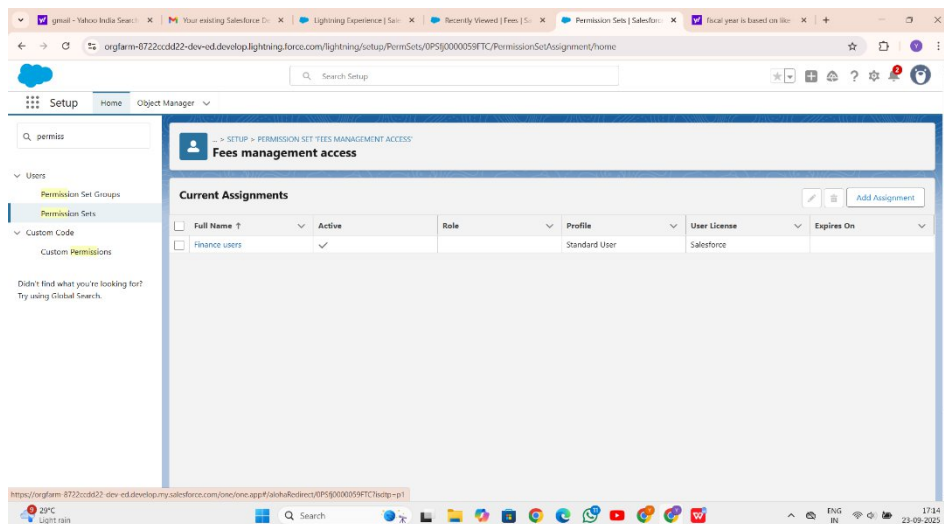
9. Objects Created

•**Student (Custom Object)** – Manage student details.

•**Course (Custom Object)** – Store course information.

•**Enrollment (Junction Object)** – Relationship between Student & Course.

•**Attendance (Custom Object)** – Track daily student attendance.

•**Grades (Custom Object)** – Store marks, grades, GPA.

•**Fees (Custom Object)** – Manage payments and pending balances.

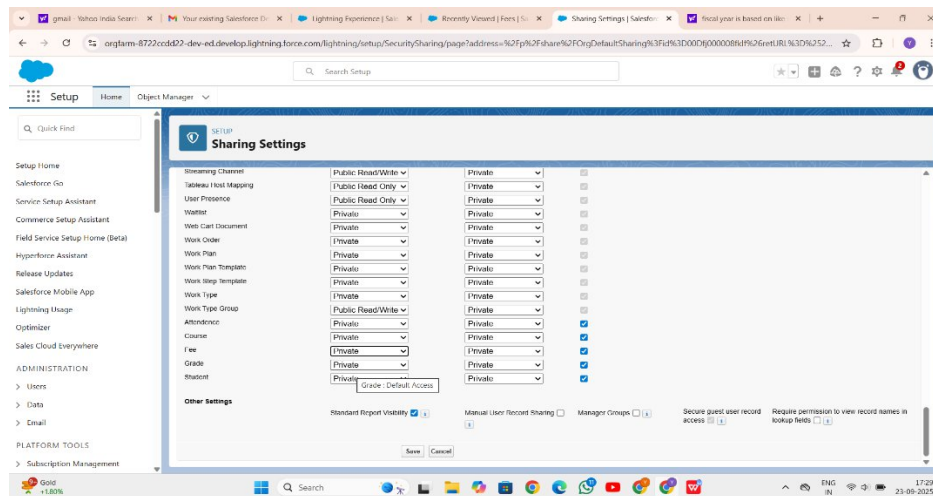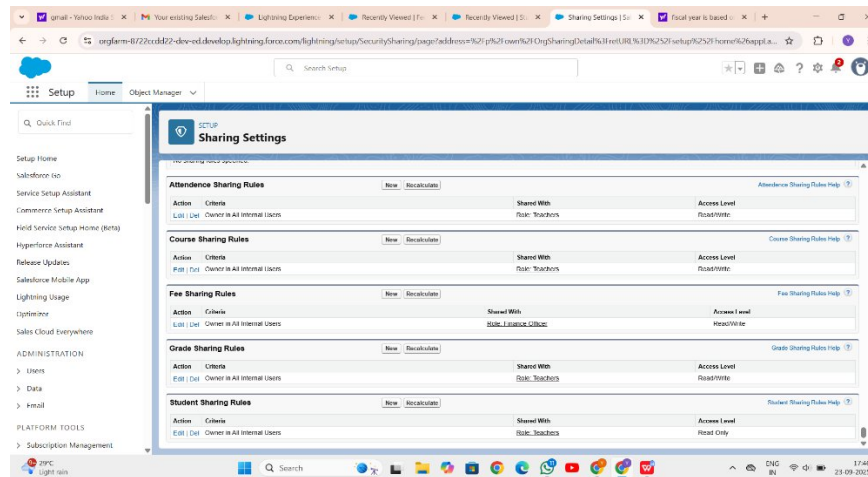## 10. OWD (Org-Wide Defaults)

- Student → Private

- Course → Public Read Only

- Enrollment → Private

- Attendance → Private

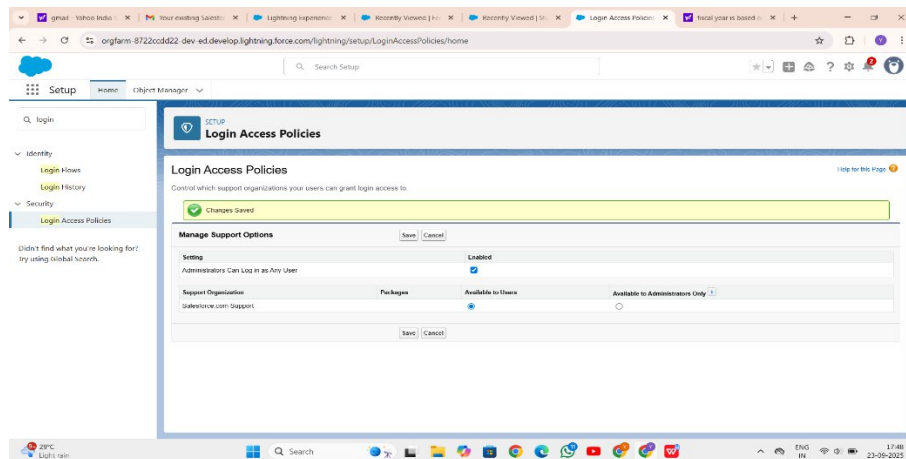- Grades → Private

- Fees → Private

11. Sharing Rules

- Shared **Courses** with all Faculty.

- Shared **Student Records** read-only with Faculty (except Finance objects).

- Finance users granted access to **Fees object**.

12. Login Access Policies

- Enabled **Administrators Can Log in as Any User**.

- Configured **Trusted IP Ranges** for secure login.

13. Dev Org Setup

- Enabled **Dev Hub** in Developer Org.

- Connected CLI:

- sf org login web --set-default-dev-hub --alias DevHub

14. Sandbox Usage

- **Not available** in Developer Edition.
- Documented sandbox usage for real-world projects (Dev, Test, UAT).

15. Deployment Basics

- Practiced deployment commands:

- sf project retrieve start --manifest manifest/package.xml -o DevHub

- sf project deploy start --target-org ScratchOrg

- Source code and configuration stored in **GitHub repo** for version control.