InternLink Hub -“A Central Platform for Internships s Placements” Phase: 10
– Final Documentation C Presentation

---

1. Project Overview

InternLink Hub is a Salesforce-based CRM platform designed to streamline internship and placement management for **students, recruiters, and placement officers**. The system provides a centralized hub where opportunities can be posted, applications tracked, and placements managed with transparency.

Through 10 structured phases, the project addressed real-world challenges such as **internship postings, application management, placement tracking, recruiter–student communication, and role-based access control**.

The solution was built on Salesforce with scalable features including **custom objects, flows, Apex triggers, validation rules, reports, and dashboards**.

The project delivered an end-to-end system where:

- **Students** can view internships, apply, and track their application/placement status.
- **Recruiters** can post opportunities, review applicants, and update statuses.
- **Placement Officers** can oversee the entire lifecycle, generate reports, and approve placements.

---

2. Problem Statement

Managing internships and placements in colleges and organizations is often manual, fragmented, and lacks transparency. The challenges include:

- Difficulty in managing student applications across multiple internships.
- Lack of real-time visibility into application and placement status.
- Communication gaps between recruiters, students, and placement officers.
- No centralized dashboards for data-driven decision-making.
- Absence of role-based access, leading to cluttered views for different users. As a

result, the process becomes time-consuming and inefficient.
InternLink Hub addresses this gap by providing a **centralized, automated Salesforce CRM solution** for internship and placement management.

---

3. Solution Overview

The InternLink Hub project was implemented in Salesforce with a focus on **role-based views and automation**. Key features delivered include:

- **Custom Objects** for Students, Companies, Internships, Applications, and Placements.

- **Tabs** configured for easy access based on roles.
- **Flows** for automated placement approval and application submission.
- **Apex Classes/Triggers** for handling backend logic and automation.
- **Validation Rules** for data consistency (e.g., phone numbers with +91 format).
- **Reports s Dashboards** for real-time tracking of applications and placements.
- **Lightning App (InternLink Hub)** combining all functionalities into a single workspace.

---

4. Project Phases s Deliverables

Phase 1: Problem Understanding s Industry Analysis

- Identified issues in manual internship and placement tracking.
- Defined need for automation, transparency, and dashboards.
- Success metrics defined: role-based access, reduced manual work, improved reporting.

Phase 2: Org Setup s Configuration

- Salesforce Developer Org created.
- Defined Roles: Placement Officer, Recruiter, Student.
- Created Profiles with specific object and tab permissions.
- Security Model configured (OWD, FLS, Permission Sets).

Phase 3: Data Modeling s Relationships

- Created custom objects:
  - **Student__c** → Personal details, skills, resume.
  - **Company_c** → Company info and contact details.
  - **Internship__c** → Internship details and requirements.
  - **Application_c** → Links student to internship with status.
  - **Placement__c** → Final placement record with package.
- Relationships defined (Student ↔ Application ↔ Internship ↔ Company).
- Page Layouts, Record Types, and Compact Layouts configured.

Phase 4: Process Automation

- Flows built for:
  - **Placement Officer Flow** → when a student is selected.
  - **Recruiter Flow** → when an application is submitted.

- Validation Rule: Phone number must include **+G1** and 12 digits.

Phase 5: Apex Development

- Apex Classes for custom logic supporting flows.
- Triggers for automated status updates and notifications.
- Test Classes developed (>75% code coverage).

Phase 6: Lightning Web Components (LWCs)

- Not implemented. Used **standard Lightning components** for layouts and dashboards.

Phase 7: Integration

- Not implemented. Future scope to integrate with **Job Portals or HR Systems**. **Phase**

8: Data Management s Deployment

- Created sample student, company, internship, and application records.
- Used Data Loader for bulk import/export.
- Simple deployment in Developer Org.

Phase G: Reporting s Dashboards

- Reports created for:
    - Student details.
    - Internship postings by company.
    - Application statuses.
    - Placement statistics.
- Dashboards created (e.g., **Student Progress Dashboard**) to visualize end-to-end lifecycle.
- Dynamic Dashboards enabled for role-based visibility.

Phase 10: Final Presentation s Wrap-Up

- Consolidated documentation and demo video created.
- Showcased views for System Admin, Placement Officer, Recruiter, and Student.
- Demonstrated reports, dashboards, flows, and automation.

---

5. Security s Compliance

- Data Security: Roles, Profiles, OWD, and FLS enforced.
- GDPR compliance maintained (data usage limited to demo data).
- Audit Trail enabled for activity tracking.

6. Project Outcomes

- Delivered **InternLink Hub Lightning App** with 5 custom objects and role-based tabs.
- Automated key processes with **flows and Apex triggers**.
- Improved **data accuracy** with validation rules.
- Built **reports and dashboards** for transparency in applications and placements.
- Established a complete system for **students, recruiters, and placement officers** to collaborate.

7. Future Enhancements

- **Einstein Analytics** for predictive placement trends.
- **Student Portal LWC** for a more interactive interface.
- **Integration with external job portals** for real-world internship data.
- **Mobile App extension** for students and recruiters.

Test Case 1: Phone Number Validation Rule

Use Case / Scenario:

Testing the validation rule that prevents saving a Student record with an invalid Phone Number format.

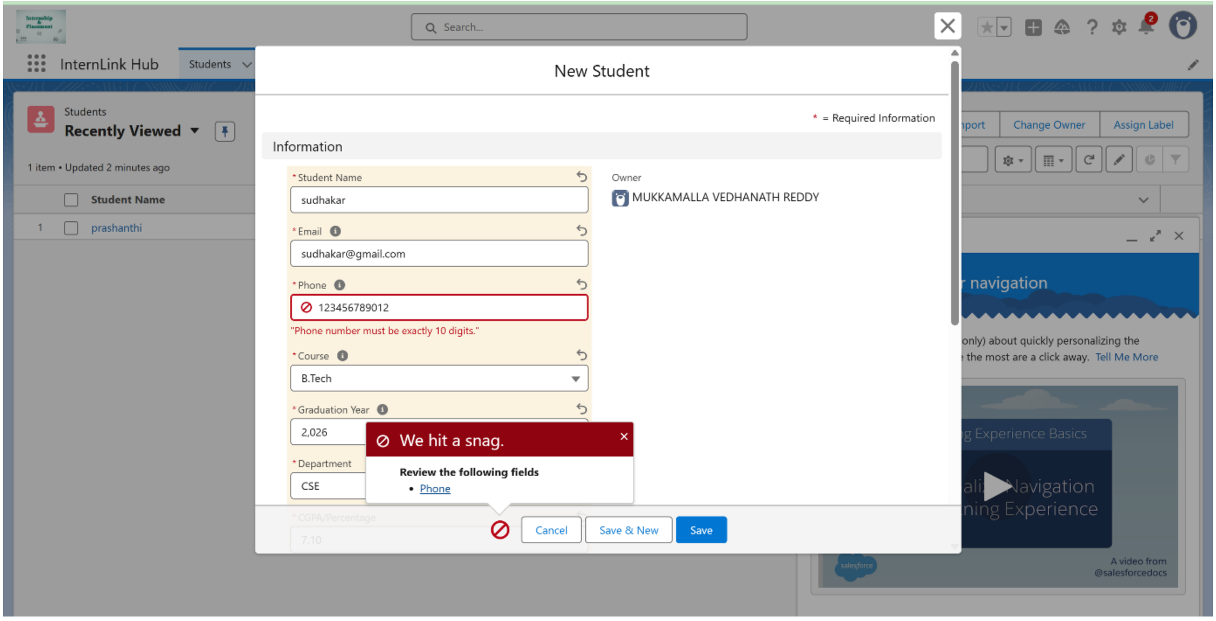
Test Steps (with input):

1. Navigated to the Students tab and clicked New.
2. Entered Student details with a Phone Number as 123 (invalid format).
3. Clicked Save.

Expected Result:

Salesforce should prevent the record from being saved and display the error message:
“Enter a valid phone number with 10 digits.”

Actual Result (with Screenshot):
The actual result matched the expected result. The system correctly blocked the record creation and displayed the specified error message.



New Student

* = Required Information

Information

*Student Name: sudhakar

Owner: MUKKAMALLA VEDHANATH REDDY

*Email: sudhakar@gmail.com

*Phone: 123456789012

"Phone number must be exactly 10 digits."

*Course: B.Tech

*Graduation Year: 2,026

*Department: CSE

We hit a snag.

Review the following fields
- Phone

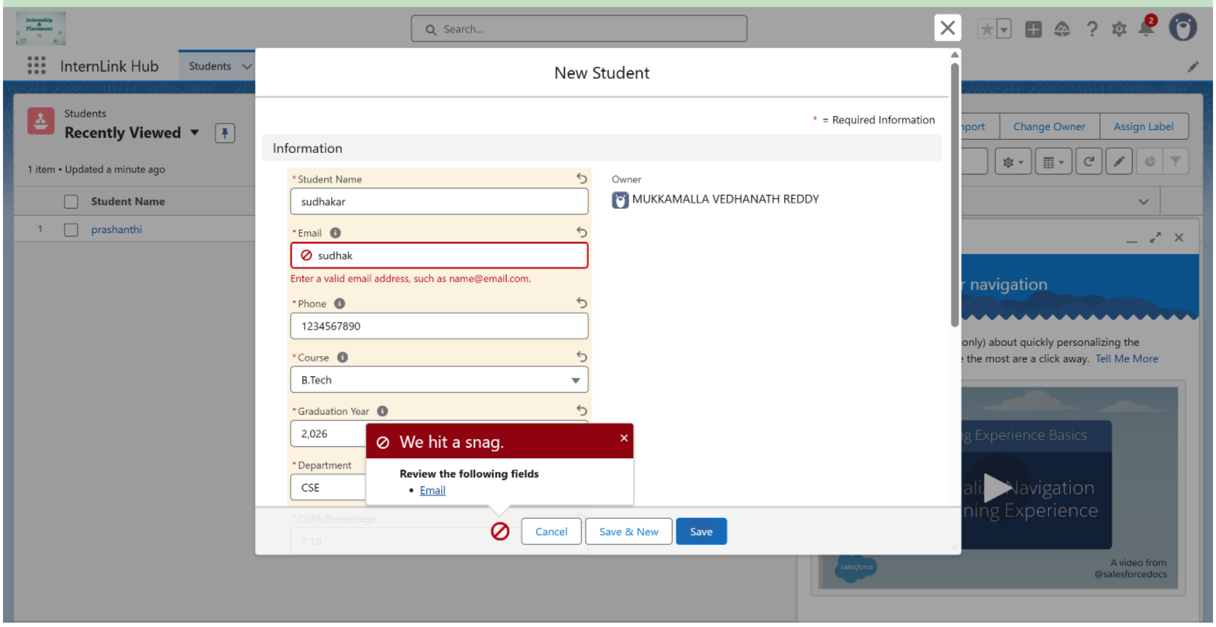
Test Case 2: Gmail Validation Rule
Use Case / Scenario:
Testing the validation rule that prevents saving a Student record unless the Email field contains a valid Gmail address (ending with @gmail.com).

Test Steps (with input):

1. Navigated to the Students tab and clicked New.
2. Entered Student details with Email = john.doe@yahoo.com (invalid, not Gmail).
3. Clicked Save.

Expected Result:
Salesforce should prevent the record from being saved and display the error message:
"Enter a valid Gmail address (must end with @gmail.com)."

Actual Result (with Screenshot):
The actual result matched the expected result. The system correctly blocked the record creation and displayed the specified error message.



New Student

* = Required Information

Information

*Student Name: sudhakar

Owner: MUKKAMALLA VEDHANATH REDDY

*Email: sudhak

Enter a valid email address, such as name@email.com.

*Phone: 1234567890

*Course: B.Tech

*Graduation Year: 2,026

*Department: CSE

We hit a snag.

Review the following fields
- Email

Test Case 3: Opening Name Validation Rule

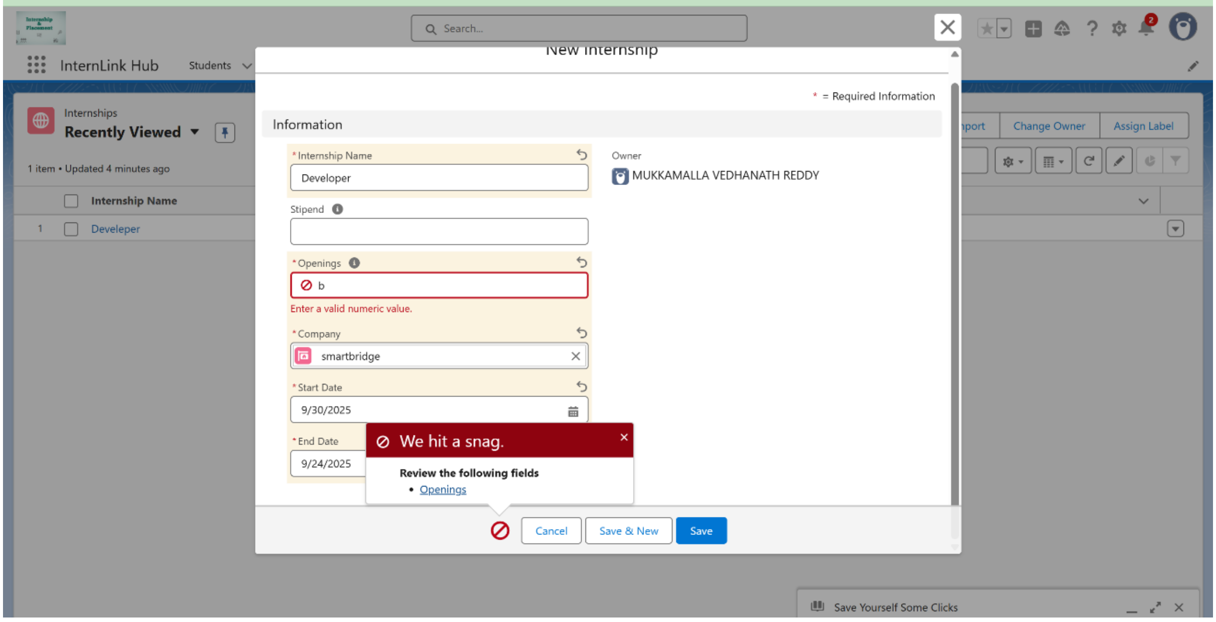
Use Case / Scenario:
Testing the validation rule that prevents saving an Internship record when the Opening Name contains alphabetic characters instead of numeric values.

Test Steps (with input):

Navigate to the **Internships** tab and click **New**.
Enter Internship details with Opening Name as Five (invalid format using numeric format).

Expected Result:
Salesforce should prevent the record from being saved and display the error message:
"Enter a valid number for openings."

Actual Result (with Screenshot):
The system correctly blocked the record creation and displayed the specified error message, ensuring only valid numeric data is accepted for the Opening field.



New Internship

* = Required Information

Information

*Internship Name: Developer

Owner: MUKKAMALLA VEDHANATH REDDY

Stipend:

*Openings: b

Enter a valid numeric value.

*Company: smartbridge

*Start Date: 9/30/2025

*End Date: 9/24/2025

We hit a snag.

Review the following fields

- Openings

Conclusion

The InternLink Hub project successfully streamlined the internship and placement process using Salesforce. By combining **custom objects, automation, Apex, validation, reports, dashboards, and role-based access**, the system improved transparency, reduced manual work, and enhanced decision-making.

Final Status: Project Completed (Phase 1–10)
Deliverables: Documentation, Apex Code, Validation Rules, Reports, Dashboards, Lightning App, and Demo Presentation