TalentBridge Project Report

Comprehensive Documentation Across 10 Phases

Prepared by: Palleboyina Deekshitha

Date: 26th September 2025

Introduction

TalentBridge is a recruitment management solution built on Salesforce. The system streamlines job postings, candidate applications, interviews, offers, and onboarding tasks. This document details the implementation process across 10 key phases, from industry analysis to final demo.

Phase 1: Problem Understanding & Industry Analysis

I**ndustry Problem**: Traditional recruitment systems lack transparency and coordination between recruiters, HR, and candidates. Processes like job posting, candidate screening, interview scheduling, and onboarding are often siloed.

Solution: TalentBridge addresses this by providing a **unified Salesforce-based HR Portal** with the following goals:

- Streamline job and candidate management.
- Automate application and interview processes.
- Provide dashboards for performance tracking.
- Enhance collaboration between HR and recruiters.

Industry Relevance: Recruitment technology (HRTech) is trending toward automation and AI. Our project positions Salesforce as a customizable recruitment solution.

Phase 2: Org Setup & Configuration

Org Setup: Salesforce Developer Org created and configured for HR Portal.

User Roles & Profiles:

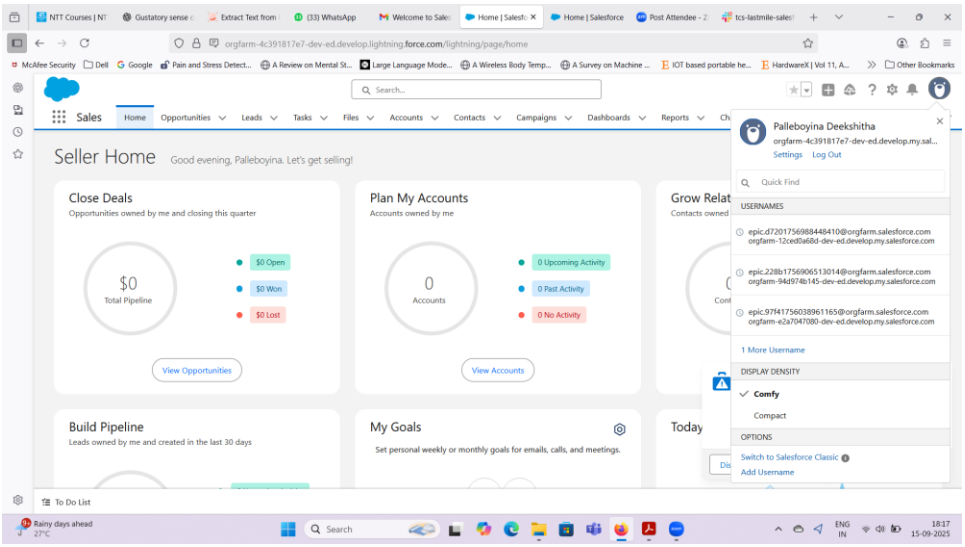
- **Recruiter** – job posting, candidate sourcing.
- **HR** – candidate selection, onboarding, final decisions.

- **System Administrator** – full access, configuration.

Permission Sets: Defined for recruiters and HR users to access custom objects like Jobs_c__c, Candidate__c, Application__c, OnboardingTask__c.

App & Tabs: Created a custom app **TalentBridge** with tabs for Jobs, Candidates, Applications, and Dashboard.

Screenshot: Salesforce Org Home



Phase 3: Data Modeling & Relationships

Custom objects were created to represent Jobs, Candidates, Applications, Interviews, Offers, and Onboarding tasks. These objects were linked via lookup and master-detail relationships to support a unified hiring workflow.

Custom objects created to represent recruitment lifecycle:

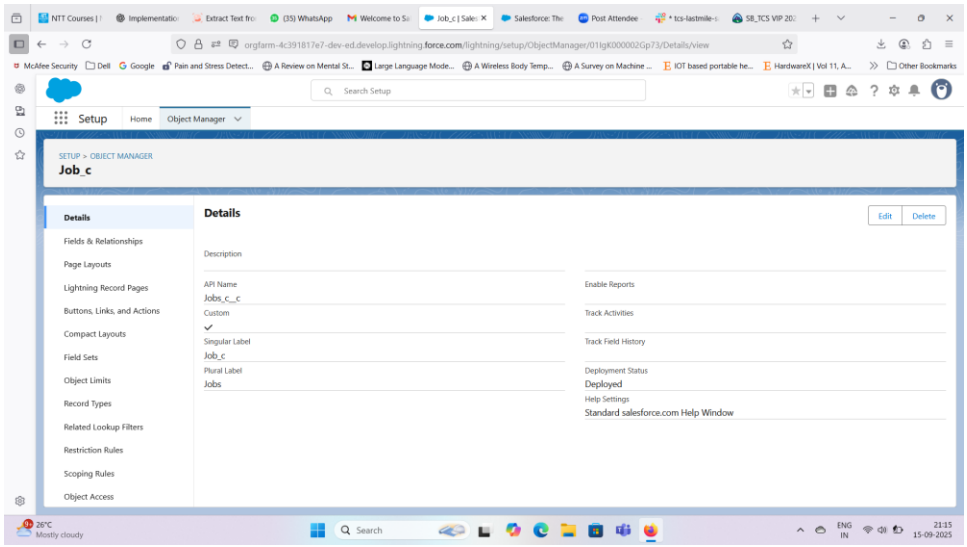
1. **Jobs_c__c** – Stores job openings.
    - Fields: Title__c, Location__c, Status__c, CreatedDate.
    - Status values: Open, Closed.
2. **Candidate__c** – Represents applicants.
    - Fields: Name, Email__c, Experience__c.
3. **Application__c** – Links candidates to jobs.
    - Fields: Candidate__c (Lookup), Job__c (Lookup), Stage__c.
    - Stage values: Applied, In Review, Selected, Rejected.

4. **OnboardingTask__c** – Tracks onboarding tasks.
    - Fields: TaskName__c, AssignedTo__c, TaskType__c (Picklist), Status__c.

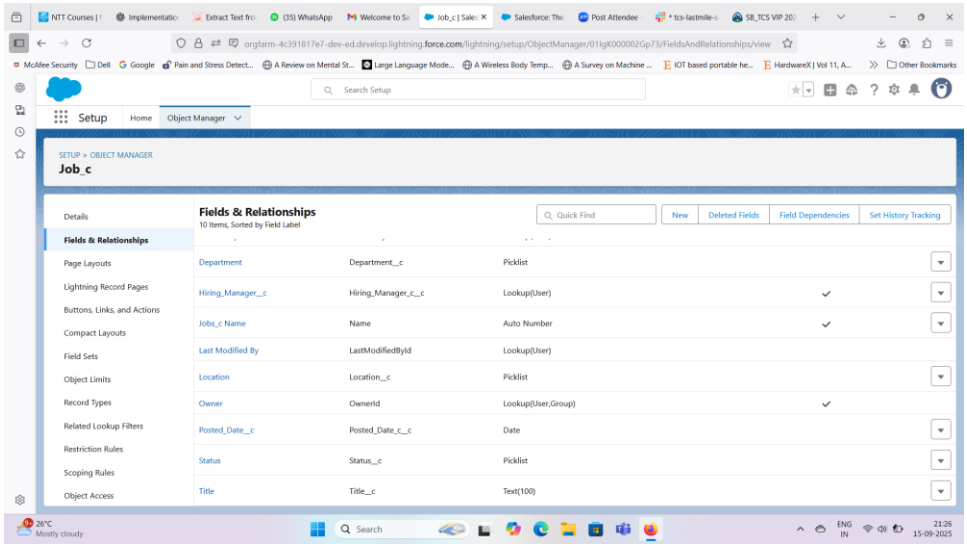
Relationships:

- **Job → Application → Candidate** (Master-Detail/Lookup).
- One job can have many applications, one candidate can apply for many jobs.

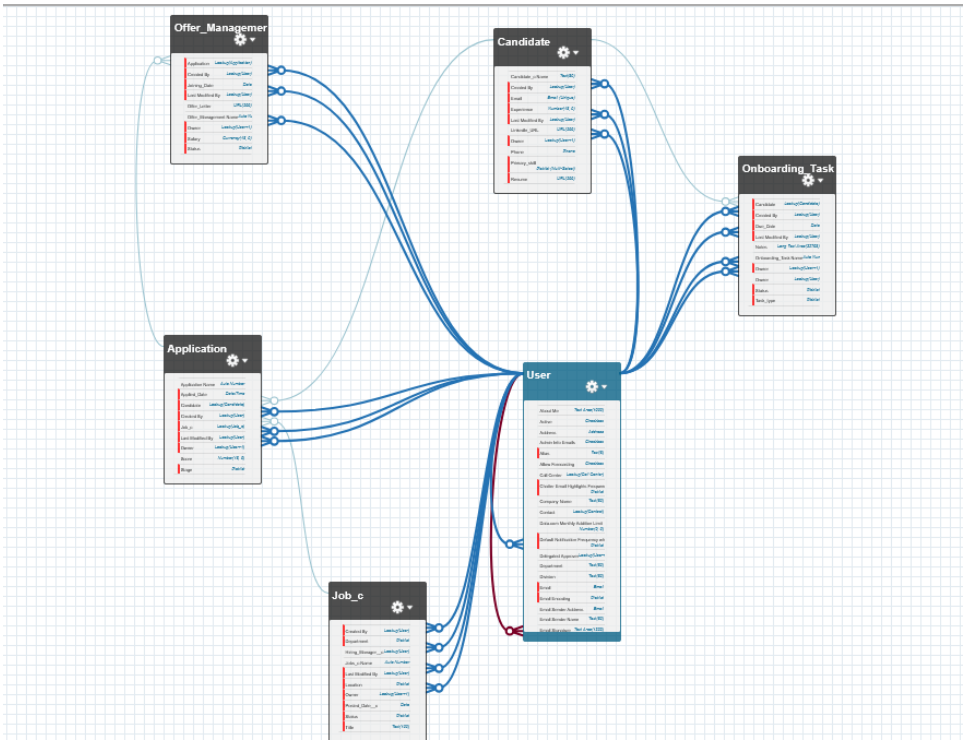
Screenshot: Job_c Object Setup



Screenshot: Jobs Object Fields



Screenshot: Data Model Schema



Phase 4: Process Automation (Admin)

Record-Triggered Flows:

- When Application stage = *Selected*, send an email to candidate.
- When Job status = *Closed*, prevent new applications.

Approval Process:

- Optional HR approval for final job closure.

Validation Rules:

- Candidate email format check.
- Job cannot be closed unless all applications are processed.

Phase 5: Apex Programming (Developer)

Custom Apex controllers were developed to fetch aggregated data for dashboards. This included counts of Jobs, Applications by stage, and Candidates.

- **Apex Classes Developed**:

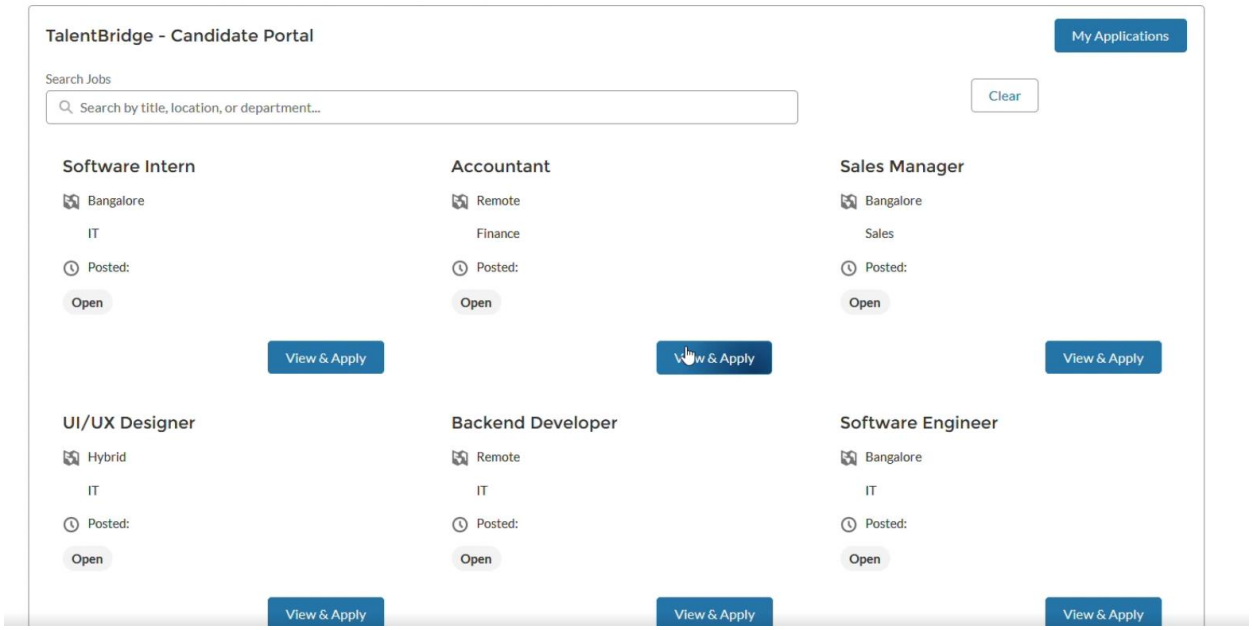
1. **HRPortalController.cls** – Core CRUD and logic for jobs, applications, candidates, interviews, and onboarding tasks.
    - Methods: createJob, updateApplicationStatus, scheduleInterview, sendOfferLetter, getAllCandidates, getHRUsers.
2. **TalentBridgeDashboardController.cls** – Provides summarized data for dashboards.
    - Returns: Job stats, Application stats, Candidate stats, Jobs by Day, Applications by Day.

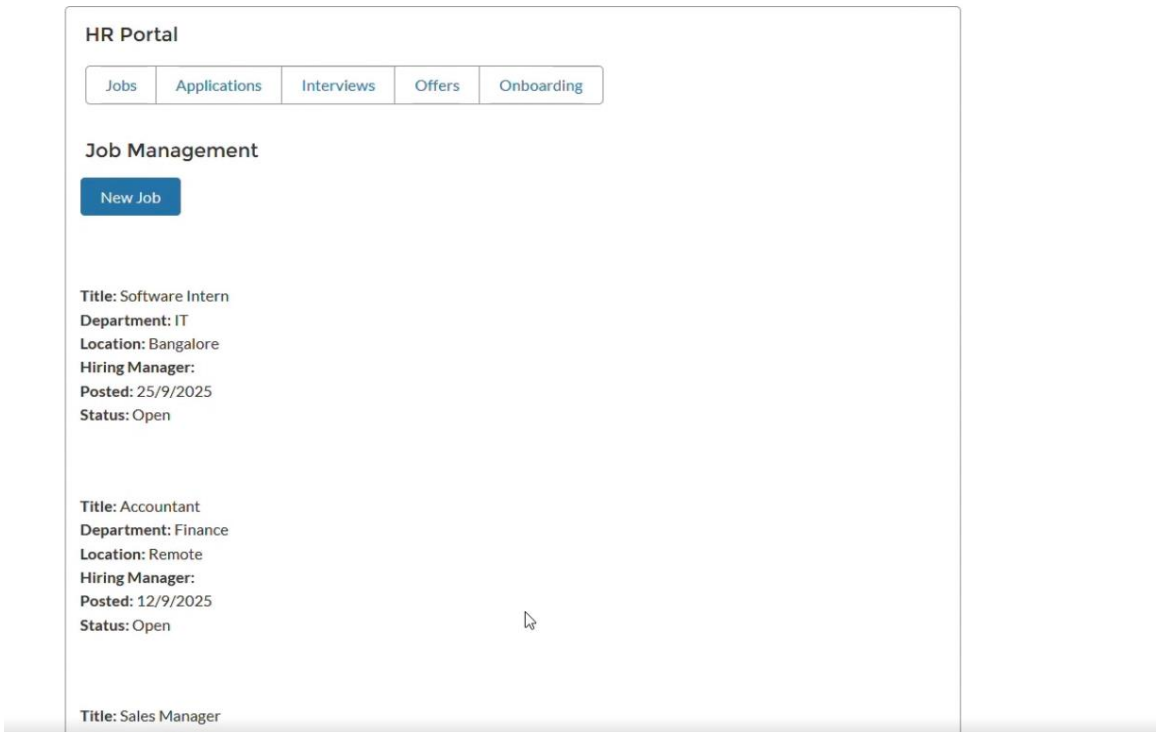
- **Best Practices Applied**:
    - @AuraEnabled methods for LWC integration.
    - SOQL queries optimized with selective fields.
    - Exception handling with proper error messages.

Phase 6: User Interface Development

- **Lightning Web Components (LWCs)**:
    1. **talentBridgeLWC** – Central dashboard displaying KPIs (Jobs, Applications, Candidates), trends, and statistics.
    2. **applicationManagement** – Manage candidate applications.
    3. **onboardingManagement** – Handle onboarding tasks with picklists.
    4. **interviewScheduling** – Schedule and manage interviews.
    5. **offerManagement** – Generate and send offer letters.
- **UI Enhancements**:
    - Styled KPI cards with color highlights.
    - Responsive grid layout using SLDS.
    - DataTables for tabular job/application trends.

Screenshot: Candidate Portal and HR portal


TalentBridge - Candidate Portal
My Applications
Search Jobs
Search by title, location, or department...
Clear
Software Intern
Bangalore
IT
Posted:
Open
View & Apply
Accountant
Remote
Finance
Posted:
Open
View & Apply
Sales Manager
Bangalore
Sales
Posted:
Open
View & Apply
UI/UX Designer
Hybrid
IT
Posted:
Open
View & Apply
Backend Developer
Remote
IT
Posted:
Open
View & Apply
Software Engineer
Bangalore
IT
Posted:
Open
View & Apply


HR Portal
Jobs
Applications
Interviews
Offers
Onboarding
Job Management
New Job
Title: Software Intern
Department: IT
Location: Bangalore
Hiring Manager:
Posted: 25/9/2025
Status: Open
Title: Accountant
Department: Finance
Location: Remote
Hiring Manager:
Posted: 12/9/2025
Status: Open
Title: Sales Manager

Phase 7: Integration & External Access

Email Integration: Offer letters and application status notifications sent via Salesforce email templates.

Future Scope:

- Candidate portal via Salesforce Experience Cloud.
- Integration with LinkedIn or external job boards using REST APIs.

Phase 8: Data Management & Deployment

Objects and metadata were deployed using Salesforce DX. Data was validated in the sandbox environment before deployment.

- **Data Migration**: Sample job, candidate, and application records loaded using Data Loader.
- **Change Sets / SFDX**:
    - Deployed Apex classes and LWCs from sandbox/dev org to production-ready org.
    - Handled field-level security assignments.
- **Error Fixes**: Adjusted object API names (Jobs_c__c) to ensure alignment.

Phase 9: Reporting, Dashboards & Security Review

Reports and dashboards were created to visualize hiring progress. Apex sharing and profile-based security were configured.

Reports:

- Applications by Stage.
- Jobs by Status.
- Candidates per Job.

Dashboards:

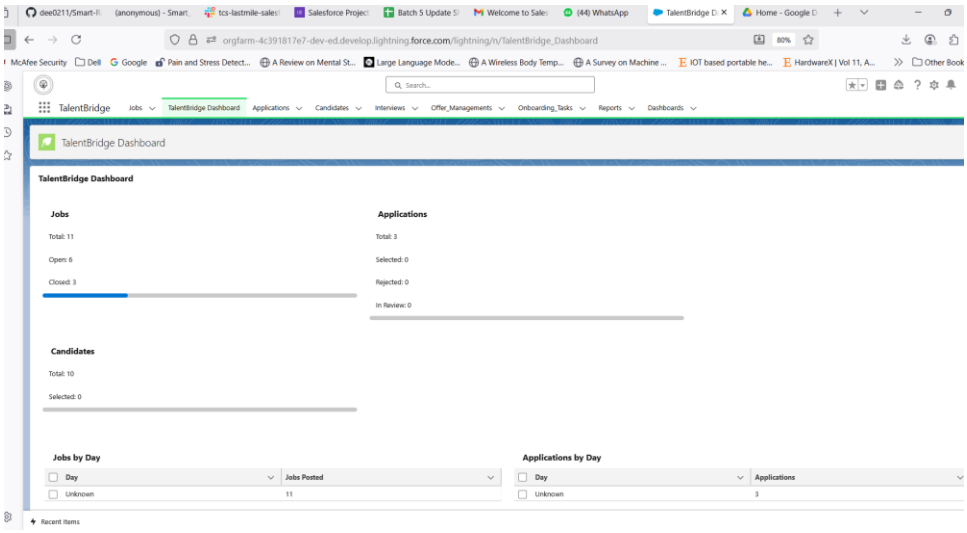
- TalentBridge Dashboard (custom LWC using Apex data).
- Visualization of job openings, application trends, and candidate counts.

Security Review:

- Applied with sharing in Apex controllers.

- Checked CRUD/FLS compliance before DML/queries.
- Profiles and permission sets aligned with user roles.

Screenshot: TalentBridge Dashboard



Phase 10: Final Presentation & Demo Day

The final demo showcased the TalentBridge dashboard, end-to-end recruitment process, and automation capabilities. The project highlighted Salesforce's strength in rapid app development.

Business Impact:

- Reduced manual coordination between HR and recruiters.
- Real-time tracking of hiring funnel.
- Improved candidate experience with timely communication.

Conclusion

The TalentBridge project successfully delivered a recruitment management platform that addresses key challenges in hiring workflows. With scalability, automation, and reporting, the solution provides a strong foundation for real-world HR operations.