YSDTP_B4

Team: Salesforce

Topic: Salesforce Training Final Project

Date: 21-01-2025

Project Title: Support360 - A Comprehensive Support Management Solution

---

1. Introduction

1.1 Purpose

This document defines the software requirements for Support360, a Salesforce Service Cloud-based application with customized features, designed to serve as a centralized solution for managing projects and service requests for GlobalTech Solutions. The Salesforce team primarily works on projects, each consisting of several cases that are handled by the team. Additionally, the team also manages service requests from users, ensuring comprehensive support and streamlined operations. The application leverages Salesforce's reporting, automation, and tailored functionalities to unify project tracking, case management, and service request handling.

1.2 Scope

Support360 will provide:

- A centralized platform for managing projects, resources, cases, and tasks.
- Integration within Salesforce to optimize reporting, dashboards, and workflows.
- Enhanced collaboration and automation to streamline business processes.

2. System Overview

Support360 integrates with Salesforce to provide a unified system for managing projects and cases with robust reporting, dashboards, and automation.

3. Functional Requirements

3.1 Case Management Module

1. **Case Creation and Management**:
    - Create Cases under a Project to track support requests, development tasks, bug reports, and change requests.
    - Use Record Types to differentiate Case types.
    - Track Case statuses: To Do, In Progress, Testing, Done.
    - Enable merging of duplicate Cases based on subject and client.
2. **Status Path**:
    - Configure the Case record page to visually represent the "Status" field using a path.
    - Make the path visible to all users.
    - When choosing the Close status from the path, a pop-up will appear and take Case Resolution as input and after clicking submit button success toast will be shown and the resolution field will be updated.
3. **Approval Process**:
    - When case is closed an approval process will start
    - Set approvers dynamically from the Case detail page.
    - Notify Case owners via email and Salesforce notification upon approval completion.
    - Update the Case detail page with the latest approver if the approval is reassigned.
    - Allow approvers to approve requests by replying to the email.

4. **Due Date Notifications and CSV Generation**:
    - Notify Case owners one day before the due date if the Case is not closed.
    - Generate a CSV with Case details and a URL for the Project Manager if the due date is missed.

3.2 Project Management Module

1. **Project Creation and Management**:
    - Create and manage Project records with fields: name, start date, end date, priority, and description.
    - Associate Projects with customers (one customer can have multiple Projects).
    - Each Project contains user stories, and user stories include tasks.
2. **Manager Access**:
    - Only Managers can create, read, and edit Project records.
    - Include a ‘Change Manager’ button in the Project list. When clicked:
        - Display a pop-up to select a new manager (excluding the logged-in user).
        - Notify the new manager via email after the change.
3. **Resource Allocation**:
    - Assign resources (users) to Projects and track availability.
    - Allow a resource to be assigned to multiple Projects simultaneously.

    - Propose workarounds for any Salesforce limitations related to resource management.
4. **Status Tracking**:
    - Track project statuses: Not Started, In Progress, On Hold, Completed.

5. **Project Dashboard**:
    - Include components such as Gantt Chart, In Progress Cases, Completed Cases, and Change Request (CR) List.

3.3 Sub-Task Management

- Create Sub-Tasks under a Case for detailed activity tracking.
- Assign Sub-Tasks to team members with due dates and priority levels.
- Track Sub-Task statuses.

3.4 Reporting and Dashboards

- Develop Project-level dashboards showing:
    - Status and progress.
    - Resource utilization.
- Develop Case-level reports to track:
    - Ticket resolution times.
    - Backlog.
- Create a Personalized Home Page for support team members
    - Case Trends (open, resolved, and escalated cases).
    - Case Backlog by Priority and Status.
    - Team Performance (time to resolution, cases handled per user).

3.5 Collaboration Features

- Enable Chatter for discussions within Projects and Cases.
- Configure notifications and reminders for task updates and deadlines.

3.6 Automation and Validation

- Use workflows and Process Builder to:
    - Notify team members of Case updates.
    - Automate status changes (e.g., mark a Case as Completed when all Sub-Tasks are completed).
- Add validation rules to ensure mandatory fields are completed when creating Projects, Cases, or Tasks.

3.7 Custom Features for Project Management

- Milestone tracking within Projects.
- Budget tracking and comparison with actual expenses.

- Risk management module to document and address project risks.

3.8 Security and Access Control

1. Implement role-based access:
    - Managers have full access.
    - Team members have access only to assigned Cases and Tasks.

3.9 Salesforce Application Setup

- Create a dedicated Salesforce App called **Support360** to encapsulate all related objects, components, and functionalities.
    - Include custom tabs for:
        - Projects
        - Cases
        - Sub-Tasks
        - Reports and Dashboards
    - Use a custom logo and branding for the app.
- Ensure easy navigation and accessibility for all project users.

4. System Design Constraints

- Leverage standard Salesforce tools such as Record Typesr, and Flows.
- Use custom development (Apex, Lightning Components) only when necessary.

5. Assumptions and Dependencies

- **Assumptions:**
    - Users have appropriate Salesforce licenses.
    - GlobalTech Solutions follows Salesforce governance policies.
- **Dependencies:**
    - The system depends on Salesforce’s notification and email services.
    - Third-party integrations may require APIs.

6. Acceptance Criteria

- Projects, Cases, and Sub-Tasks can be created and managed as specified.
- Resource allocation and tracking are functional.

- Approval processes, including reassignment and email approvals, are operational.
- Dashboards and reports meet the stated requirements.
- Security and access controls align with the defined roles.

7. Deliverables

- 1GP Managed Package
- Source code with all components
- Project documentation:
    - User Manual
    - System Design Document
        - Process Document for Apex Classes
        - Process Document for Salesforce Flows
        - Object Design Document
    - Test Cases and Test evidence document
- Integration configurations and documentation.
- Training materials for end-users and administrators.
- Release Note (List of Components, Pre-Deployment Configurations, Post-Deployment Configurations)

8. Project Guidelines

Code Management

- Use a version control system (e.g., Git).
- Commit code with meaningful commit messages.
- Follow branch naming conventions (e.g., `feature/`, `bugfix/`, `hotfix/`).

Task Management

- Maintain clear and updated records in Jira or Redmine for all tasks:
    - Define task objectives, owners, and due dates.
    - Track task progress and ensure timely updates.
    - Regularly review and address overdue tasks.

Development Guideline

- Ensure SOQL queries, DML operations, and other resource usage stay within limits.
- Avoid performing SOQL or DML operations inside loops.
- Design Apex code to handle bulk operations efficiently
- Implement batch processing for operations that might exceed limits.

- Use Salesforce declarative tools like Flows, Process Builder, and Validation Rules wherever possible.
- Opt for custom code (Apex) only when declarative tools cannot meet requirements.
- Provide test class for every Apex class, achieve at least **90% coverage**, with meaningful assertions to validate logic.

Justification for the Project:

- **Comprehensive Coverage**: This project covers a wide range of Salesforce functionalities, including case management, project management, reporting, automation, and security, making it an ideal evaluation tool for beginners.

- **Real-world Application**: The project simulates a realistic business use case, involving case and project management, which is common in Salesforce environments.

- **Scalable Complexity**: It includes both declarative (flows, record types, validation rules) and programmatic (Apex, Lightning Components) solutions, allowing trainees to demonstrate skills in both areas.

- **Collaboration & Automation Focus**: Emphasizes teamwork through case collaboration (Chatter) and process automation, which are key skills in modern Salesforce environments.

Skills Tested:

- **Salesforce Customization**: Ability to create and manage custom objects, fields, and record types, and configure page layouts and lightning components.

- **Automation Skills**: Proficiency in using Salesforce automation tools such as Process Builder, Workflow Rules, Flows, and approval processes.

- **Apex & Lightning Components**: Writing Apex classes and triggers, implementing Lightning Components, and handling SOQL queries, DML operations, and bulk processing.

- **Security & Access Control**: Setting up role-based access, creating profiles, and managing permissions.

- **Reporting & Dashboards**: Ability to design and implement custom reports and dashboards for project and case tracking.

- **Validation Rules**: Ensuring data quality through mandatory field validation and validation rules during record creation and updates.
- **Collaboration Tools**: Using Chatter and notifications to enhance collaboration within projects and cases.
- **Project Management Concepts**: Handling tasks such as resource allocation, project tracking, and milestone management, applying real-world project management principles.