

Community Case Comments

Project Initiation Document

Table of Contents

Project Initiation Document	1
1. Problem or Opportunity Statement.....	3
1.1 Primary Outcome.....	3
2. Scope.....	4
2.1 In Scope	4
2.2 Out of Scope	4
2.3 Deliverables	5
3. Timeline.....	6
3.1 Overall Implementation Timeline	6
4. Solution Overview	7
5. Risks	10
6. Assumptions.....	11

1. Problem or Opportunity Statement



1.1 Primary Outcome

Currently when creating a community, you can leverage the drag-and-drop empowered ‘builder’ to quickly form the look, feel and features. Although in certain cases the out of the box (OOTB) components provide common limitations. Case Comments via community is one of those limited, yet frequent problem areas – Using OOTB features, you can provide logged in users the option to create a new comment against a case, but not without an undesirable quirk.

The quirk with the OOTB component, is that it displays a checkbox labelled “public” on both the related list and pop-up modal, which while has context and meaning for internal users, is confusing/concerning for external users – especially when the case itself is dealing with a sensitive topic.

The internal use-case for this checkbox is to indicate if the comment should be visible to external users or not. So, if the checkbox is not ticked the external user would not be able to read the comment (even if they made it themselves) and thus, to ensure that external users can read comments they create, Salesforce will automatically tick the box after submission. As you can imagine, this automatic set to ‘public’ will doubly confuse the external user when they did not activity tick the public checkbox on comment creation.

OOTB Related List:

Case Comments (2)				New
User	Public	Created Date	Comment	
 Jon Grant	<input checked="" type="checkbox"/>	6/21/2021, 2:37 PM	Test2	▼
 Jon Grant	<input checked="" type="checkbox"/>	6/21/2021, 8:54 PM	Test	▼
				View All

OOTB Pop-up modal:

Quirk in brief: Salesforce provide no configurable method to remove/hide this 'public' checkbox.

The idea of this project is to create a Lightning Web Component (LWC) paired with an Apex Class that can be swiftly deployed into any Salesforce instance and dropped into a community/digital experience – replacing the standard case comments component with a customized and non-quirky component.

This component should allow for logged in community users to view and create a case comment and be mobile optimized. In effect the component should be similar in nature to the OOTB, just less confusing – creation doesn't necessarily need to be a pop-up modal.

2.Scope

2.1 In Scope

1. Front-End (Intended for Community User)
 - a. Input text box
2. Component Configuration (Intended for a Salesforce Admin)
 - a. Can customize the number of comments displayed per page.
 - b. Can customize the related case record id and defaulted to '{!recordId}'.
3. Documentation (Intended for a Salesforce Developer)
 - a. Paired documentation /read-me file on how to update the component.

2.2 Out of Scope

1. Updating/deleting comments.

2.3 Deliverables

1. Repo containing the boilerplate metadata components, will likely contain at least the following:
 - a. Lightning Web Component
 - b. Apex Class
 - c. Test Apex Class
 - d. Read me (instructions on how to implement common scenarios)

Item	Feature/Category	Estimate (days)	Notes
1	Case Comment - Related List, with pagination	3-6	
2	Case Comment – Create New	2-4	
3	Documentation	1	
4			
5			
6			
7			
	TOTAL		

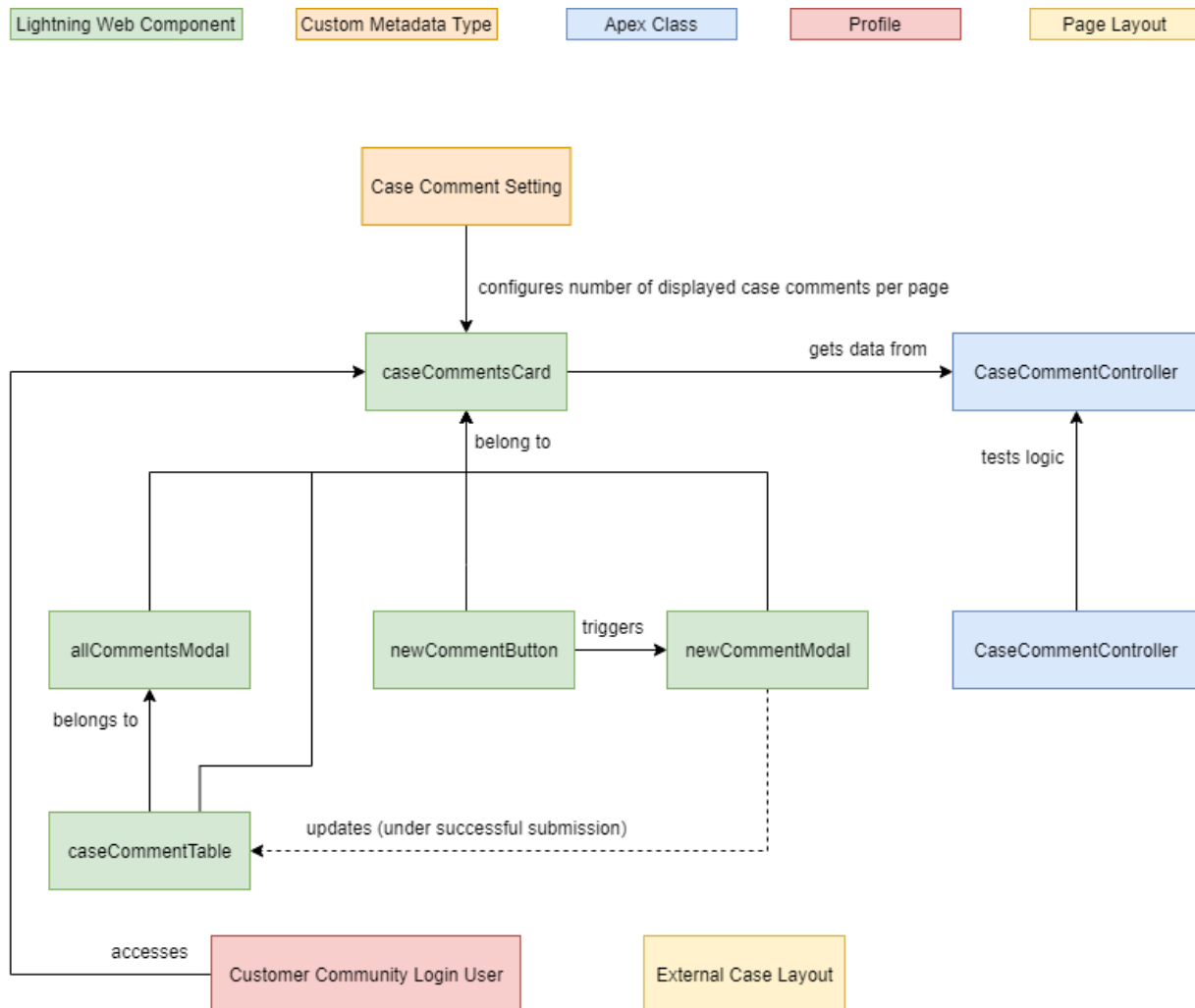
3. Timeline

3.1 Overall Implementation Timeline

#	Feature/Category	Time Spent (days)
1	Understand the Requirement and design solutions	1
2	Write initial LWC components and Apex Classes	1
3	Fix main bugs for the components and Apex Classes	2
4	Implement configuration functionality for admins	1
5	Refactor code and test	1
6	Documentation	1

4.Solution Overview

High level overviews of xxx architectures are below, along with key architectural decisions made or to be made for each.



a. Lightning Web Components

- `caseCommentsCard`

The component displays a specific number of case comments per page.

It is added to the case detail page of the Experience Cloud Site.

- `allCommentsModal`

A child component of `caseCommentsCard` component.

It displays all the case comments on the case.

It is displayed when View All is clicked.

- `caseCommentTable`

A child component of `caseCommentsCard` and `allCommentsModal` components.

It displays the case comments in the table view.

- `newCommentModal`

A child component of `caseCommentsCard` component.

It allows users to submit a new case comment on the case.

- `newCommentButton`

A child component of `caseCommentsCard` component.

It invokes the popup of `newCommentModal` component.

b. Apex Classes

- `CaseCommentController`

The Apex class that provides related data for `caseCommentsCard` component.

- `CaseCommentControllerTest`

The Apex test class that tests the logic in `CaseCommentController`.

c. Custom Metadata Types

- Case Comment Setting

The custom metadata type that configures how many case comments to be displayed per page in caseCommentsCard.

d. Profiles

- Customer Community Login User

The profile that is assigned to the external users in the Experience Cloud Site.

It allows the users to access caseCommentsCard.

It allows the users to read and create case comments.

e. Page Layouts

- External Case Layout for Case

The page layout that is assigned to the external users in the Experience Cloud Site.

It excludes the confusing out of the box component under related lists on the Case Detail page.

5.Risks

#	Risk outline	Mitigation
R01	Scope creep Additional items requested in scope increase project time/cost without improving delivery of the business case.	Upfront contingency to cover late-identified items critical to delivery of the business case. Any scope increases require sign-off through Portfolio Management.
R02	The Solution looks different from the OOTB due to the limitation of Salesforce	Reach the delivery manager and the client to discuss the limitation and the solution
R03	Spend time on functionality that the OOTB component has but is not clearly required in the project	Reach the delivery manager to discuss whether to implement it if it is going to take much time
R04		
R05		
R06		

6.Assumptions

1. Do not implement the functionality that the OOTB component has but is not clearly required in the project if it is going to take much time.
2. The customer accepts the component not being under related list due to the limitation of Salesforce.