

[TEMPLATE] - Technical Documentation

- 1
 - 1.1 [Version Control](#)
 - 1.2 [Reference Documents](#)
 - 1.3 [Definitions](#)
- 2 [Introduction](#)
 - 2.1 [Overview](#)
 - 2.2 [Purpose of this document](#)
- 3 [<Process Name> Technical Documentation](#)
 - 3.1 [<Process Name> Technical Solution Overview](#)
 - 3.2 [Apex trigger Logic \(including handlers\)](#)
 - 3.3 [Lightning Components](#)
 - 3.4 [Security \(Apex only\)](#)
 - 3.5 [Email Notifications \(via Apex\) & Email Service](#)
- 4 [Annex](#)

Version Control

Version	Date	Description	Author
V0.0	27/12/2016	Draft	<name>

Reference Documents

Description	Document Link

Definitions

Abbreviation	Definition
CRM	Customer Relationship Management

Introduction

- **Overview**

<include a summary of context / scope that it has been implemented, when existing, create link to functional Documentation>

- **Purpose of this document**

<include a summary of the purpose of this document>

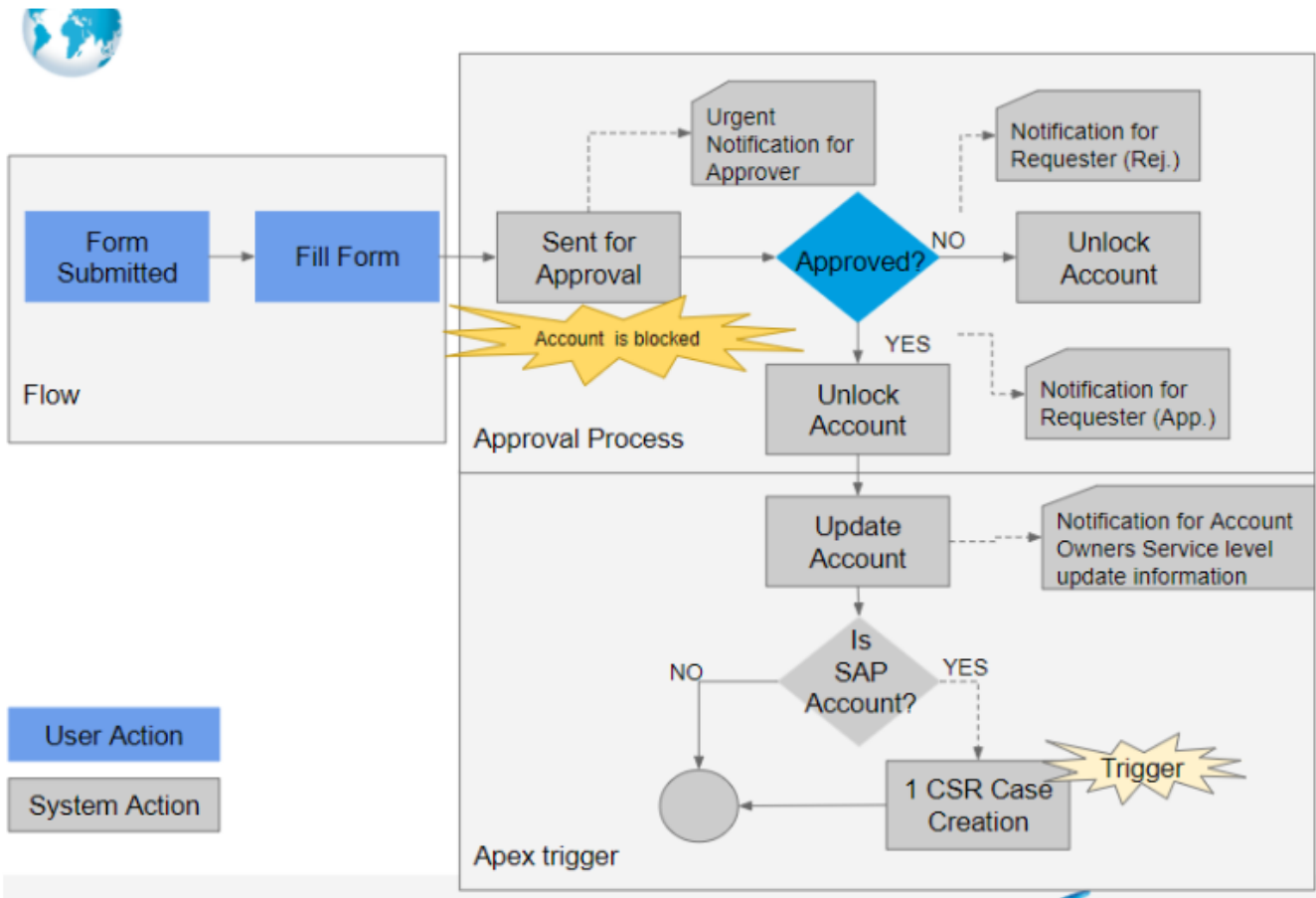
<Process Name> Technical Documentation

- **<Process Name> Technical Solution Overview**

<Include all actions performed by Users Or System that will trigger System Actions to run (Workflows, triggers, flows, assignment rules, approval processes, email notification, Account Locks, components), if the process is too complex to describe in one diagram only, you can split it by Feature, or Phase, ...

Something Similar to the screenshot below (for the Actions refer the type of solution implemented including the names, example: Action: Update Account is being done by system using Apex, For Apex implementation: include what is being done before Insert/Update after Insert/Update

<include rationale of the technical decision, for example, we are using 2 RT, why we need a first one, and a Second One>



• Apex trigger Logic (including handlers)

<Include description of before Insert/Update, After Insert/Update, validations done at code level

Everytime the process trigger calls another trigger, just refer to the documentation of the 2nd trigger

Every time code need to refer to Custom Metadata or Custom Settings, include thsi information as Annex

Context	Method	Description
Before insert	Method a	
	Method b	
After insert	Method c	
	Method d	
Before update	Method e	
	Method f	
After Update	Method g	
	Method h	

Before Delete	Method i	
	Method j	
After Delete	Method k	
	Method l	
Undelete	Method m	
	Method n	

• Lightning Components

<for each of the components describe according to the following points>

- Component Diagram

<when a component is using several components, include the architecture used>

- Component Description

What is the purpose, rationale for customization instead of Standard, explain advanced features implementation,

• Security (Apex only)

- Apex Sharing
- Methods without Sharing

Method Name	Reason why?
-------------	-------------

<List the methods that are running without sharing and describe reason why>

- Bypasses and Custom Permissions

<by development guidelines, all validation rule needs to have the bypass, so we just say generically that all validation have it, and if for some reason we need to avoid bypass, then we need to list the validation rules without bypass, and explain reason why>

• Email Notifications (via Apex) & Email Service

Annex