

REQUIREMENTS DEFINITION AND ACCEPTANCE CRITERIA

SS00XX Rev A

Revision History\

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Revision** | **Release Date** | **Release/Change Description:** | **Author** | **Approval** |
| - | 04/05/2017 | First draft release | C. Ortega | - |
| A | 04/17/2017 | Applied the official template of a procedure. Applied the official format and updated the information in this document. Improved minor grammar and orthography issues. | C. Ortega |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

***Table of Contents***

**1.0 SCOPE 4**

**2.0 PURPOSE 4**

**3.0 DESCRIPTION 4**

**3.1 DESCRIPTION OF A FEATURE 4**

**3.2 FIELDS 4**

**3.3 VIEWS 5**

**3.4 MOCK UPS 5**

**3.5 ACCEPTANCE CRITERIA 5**

**3.6 SIGN OFF 7**

## 

## 

.

1. Scope

This document will apply to all the Sonora Software’s projects that may require the gathering and draft of information during the development of the product.

2.0 Purpose

To define and standardize the way the requirements’ information is described to ensure the uniformity of the project’s information along with the acceptance criteria and improve the quality of the final product.

3.0 Description

**Note: Along with the definition of the elements in this document you will find examples that will help you understand in a better way the objective of each section. This information may not be applicable for all the projects. For further instructions, please review the available information in the project with who it may concern.**

**3.1 Description of a Feature**

A description is a certain detailed account of aspects or characteristics that are intended to give or to express a functionality.

In a project, this may be defined as a brief text about the main objective of a feature or a task.

*For Example:*

*“The user needs to add his personal information to the system.”*

**3.2 Fields**

This section is provided to list all the fields that will be find on a certain section of the application. This may include special validations such as business rules that may affect the fields and the maximum length of characters supported for the field (if apply).

Also, all the required fields will have to be specified as well as any other special condition related to a field (such editable fields, etc.)

*For Example:*

*- Name\* - Limited to 150 characters.*

*- Address\* - Limited to 300 characters.*

*- Zip Code\* - This field will only accept numbers (5).*

*- Email\* - Limited to 150 characters.*

*- Phone (In the format “(520)00-00-00”) – This field will be alphanumeric.*

*The fields with an \* are required.*

*- The user can only add his information one time only and should not be able to edit it later.*

**Note: The Information about “Special Validations” for certain fields (such names, phone number, email addresses, etc.) can already be standardized in the company or it could be defined as part of an agreement in the project documentation and may be accessible to all the members of the team. Please review this information with who it may concern.**

**3.3 Views**

This section will include (if apply) the visual elements (such buttons, colors) that will be needed for the functionality of the task.

*For example:*

*- The “Add” button should appear in the top left of the page and must be LightGreen (#3CB371).*

*- Use Arial (12) for the text, but Calibri (8) for the buttons.*

*- If everything goes right, a modal should display with a message: “Success!”*

*- Include a Save button and a Cancel button.*

*- If there is a problem, then the application should display an Error message: “There has been an error, please verify the information.*

**3.4 Mock Ups**

Add the SVN/SharePoint link (or the exact location of the field) that belongs to the task specifying the name of the mock up or as agreed with the team. This information should be presented as a hyperlink in order to make things easier for the team members.

*For example:*

*New User Mock up: SuperApp\_NewUser\_001.jpg*

**3.5 Acceptance Criteria**

**Note:** **As a precondition, for a good Acceptance Criteria definition, we require having as much information as we can get. Therefore, we encourage the team to gather this information and write it down in the corresponding feature in Axosoft as it was previously described.**

Once you have gather enough information for the Description area and all its components, then we can fill the Acceptance Criteria.

The Acceptance Criteria does not have a defined format, but a good Acceptance Criteria should:

* Help determinate when does the team is fulfilling the expected result of the client.
* Specify when does an event should happen and when should not.
* Specify if there is any major consideration for a certain field in the application.
* Specify any business rule regarding to the current user story or feature.
* Not be technical. Every person/stakeholder who may be part of the team should be able to understand what is expected (Dev’s, testers, PO, PM, sometimes even the client).
* Specify just one rule/consideration at a time.

The Acceptance Criteria can be described with the following format:

*The Given (or a User Role)/ When/ Then*

Where **Given** is some precondition that can also be a specific **User Role**,

**When** is used to represent some following action and (when apply)

**Then** is used to specify an expected result. (when apply)

**Note: The Acceptance Criteria’s format suggested in this document will not always apply to all the projects and, in some cases, they can be easily identified and described, however; the usage of this format will help the team to properly write and understand what is expected to be the objective of the task and information to be reflect in the feature’s summary in Axosoft.**

A template for an Acceptance Criteria can be:

1. As \_\_\_\_\_\_\_\_\_\_\_\_ (a User role: Admin/Project Manager/IT/Customer, a Service, etc.)

**Given**

I (need/require/can/can’t/want)

**When**

(a following action)

*For Example:*

1. *As a Customer*

***Given***

*I can be able to add my information to the application*

***When***

*I enter to the “New user” section.*

1. *As a Customer*

***Given***

*I should not be able to continue with my registration*

***When***

*If one of the required fields has not been filled.*

1. *As a Customer*

***Given***

*I can’t be able to edit my information*

***When***

*once I click on “Accept”*

1. *The application service*

***Given***

*will add the new register to the database*

***When***

*the user click on the “Accept” button*

1. *The API*

***Given***

*Should display the information of the project manager*

***When***

*You use the parameter projectId and userRole*

1. *The API*

***Given***

*Should display if a user is in a project*

***When***

*You use the parameter projectId, userRole and statusUser*

Another correct, but shorter, example of Acceptance Criteria could be:

*For Example:*

*5. The name, address, zip code and email should be required.*

*6. The name field should only admit 150 characters.*

*7. The address field should only admit 300 characters.*

*8. The zip code field should only admit 5 characters.*

*9. The email field should only admit 150 characters.*

*10. The phone field should follow the format “(520)00-00-00”.*

*11. The “Add” button should appear in the top left of the page and must be LightGreen (#3CB371).*

*12. Use the font Arial (12) for the text*

*13. Use the font Calibri (8) for the buttons.*

*14. A modal should display with a message “Success!*

*15. If the register fails then the application should display an error message.*

**3.6 Sign Off**

As a recommendation (this may not apply to all projects), every time that a member of the team change, in anyway, something in the task, a good practice would be to add the following information:

“Edited by: **[Team Member]** on **[Date]** at **[Time]**.”

This will help the team to visualize, in an agile way, whenever the information has change.

**Note: Any change done to a requirement must be communicated with ALL the members in the team and not just assume that they will see and identify the change. This include Business Analysts, Developers, Testers, Project Manager, Product Owner, and who it may concern.**

**PLEASE NOTE THAT THE ABSENCE OF INFORMATION COMPROMISES THE QUALITY OF THE FINAL PRODUCT. THIS MEANS THAT, THE MORE INFORMATION WE HAVE, THE BETTER. REMEMBER THAT THE DELIVERABLE PRODUCT IS RESPONSIBILITY OF THE ENTIRE TEAM AND NOT ONLY OF A CERTAIN AREA.**