# CMSC 345

### Software Design and Development

# Fall 2013

# System Requirements Specification Template

Writing Instructions

Do not have anything the Software Quality Assurance Organization could call a defect.

Be sure that your document is

* Complete - No information is missing.
* Clear - Every sentence's meaning must be clear to all parties.
* Consistent – The writing style and notation is consistent throughout the document and the document does not contradict itself.

Remember that you are required to do a team review of this document.

When you think you are done with the SRS, ask yourself, "Could someone who was not part of the development of this SRS design the system?”

[Put team company name and product name here]

System Requirements Specification

#### Table of Contents

Page

## 1. Introduction

* 1. Purpose of This Document
  2. References
  3. Purpose of the Product
  4. Product Scope

2. Functional Requirements

3. Non-Functional Requirements

3.1 Customer Constraints

3.2 External Interfaces

3.3 Other

4. Deliverables

5. Open Issues

## Appendix A – Team percent contribution, Team sign off, Customer acceptance

## Document Versioning Control

|  |  |  |
| --- | --- | --- |
| **Version Number** | **Date** | **Changes from Previous Version** |
| 1.0 |  | N/A |

## 1. Introduction

* 1. Purpose of This Document

***State the purpose of the document and indicate the intended readership. Briefly summarize the content. [One substantial paragraph]***

Herein are detailed the requirements associated with the geo-plotting software requested by the customer. All functional requirements, defined as those that refer to a technical performance of the software, are detailed as use-cases in *section 2*. Non-functional requirements, being the remaining non-technical requirements, are detailed under a priority list in *section 3*. Intended for general readership by those involved with the development with this project, this document will also list all intended deliverables to the customer in addition to an ongoing issues list.

1.2 References

***Provide a list of all applicable and referenced documents and other media (e.g., textbooks, UML references, documents provided by the customer, documents provided by your instructor, web sites) that were used in the creation of this document.***

Use Case Template: *Basic Use Case Template***,** byAlistair Cockburn, http://members.aol.com/acockburn/papers/uctempla.htm, accessed 1/17/08.

1.3 Purpose of the Product

***This section provides a short description of the product end user’s work in its current state and the situation that triggered the need for your customer to request the product. It describes the task(s) that the user wants to accomplish with the product and how the product will meet his/her needs. It is the product justification. [One to two substantial paragraphs]***

The software referred to by this document is being developed for the sole purpose of completing a class project at the University of Maryland, Baltimore County. The software shall be used to plot the shortest possible land route from the user’s location to a destination selected from a stored list. Further, it was requested that the software takes the form of an application.

* 1. Product Scope

**This section identifies the boundary between the system to be developed and the outside world. That is, it identifies what will be included in the system and what will not. You will use a top-level use case diagram for this purpose. In addition to referring the reader to the diagram, give a brief description of how the diagram illustrates the system’s scope and its relationships to any external systems. Briefly describe the classes of users (the primary actors) and their relationships to the various system functionalities. Make sure to number the use cases in the diagram. Note: You do not need use cases for logging in or out of the system. [One to two substantial paragraphs]**

We need more concrete information on our system to complete this section.

2. **Functional Requirements**

Each functional requirement should be represented using a use case.

Refer the reader once again to the top-level use case diagram from Section 1.4. In addition, include separate sub-use case diagrams, where appropriate, for each of the top-level use cases.

In addition to the diagrams, every use case should be documented using the use case specification format below. A suggested format for this section is to begin with a brief introduction of what the section contains, and then alternate sub-use case diagrams with their corresponding use case specifications. Make sure that all use case numbers and names correspond exactly with those in the top-level diagram of Section 1.4. Give all sub-use case diagrams figure numbers and labels (e.g., “Figure 2.1. Administer Exam”).

|  |  |  |
| --- | --- | --- |
| **Number** | *Unique use case number* | |
| Name | *Brief noun-verb phrase* | |
| **Summary** | *Brief summary of use case major actions* | |
| **Priority** | *1-5 (1 = lowest priority, 5 = highest priority)* | |
| **Preconditions** | *What needs to be true before use case “executes”* | |
| **Postconditions** | *What will be true after the use case successfully “executes”* | |
| **Primary Actor(s)** | *Primary actor name(s)* | |
| **Secondary Actor(s)** | *Secondary actor name(s)* | |
| **Trigger** | *The action that causes this use case to begin* | |
| **Main Scenario** | **Step** | **Action** |
|  | *Step #* | *This is the “main success scenario” or “happy path.”* |
|  | *Step #* | *Description of steps in successful use case* “execution” |
|  | *Step #* | *This should be in a “system-user-system, etc.” format.* |
| **Extensions** | **Step** | **Branching Action** |
|  | *Step #* | *Alternative paths that the use case may take* |
| **Open Issues** | *Issue #* | *Issues regarding the use case that need resolution* |

3. **Non-Functional Requirements**

Decide on a standard format for the non-functional requirements (NFRs). Include a unique number for each NFR, a priority (1 = lowest, 5 = highest), and a clear, concise description. It is possible that your system has no NFRs. If this is the case, briefly state so.

3.1 Customer Constraints

These are the non-functional requirements that your customer has specified up front as being non-negotiable. For example, your customer may want the product developed as a desktop application to run on his/her office computer. In this case, the following non-functional requirements would be appropriate:

The system shall be a desktop application.

The system shall run on the customer’s office computer, which is a Dell Latitude D610 running under Microsoft Windows XP Professional, Version 2002, Service Pack 2. Or, it must be a web based application and must operate under the

Firefox browser. Or, it must be a smart phone app for brand XXX.

It is possible that your customer has specified no constraints. If this is the case, briefly state so.

3.2 External Interfaces

If your customer requires your product to read from data files that are external to the system (i.e., you do not have control over), the exact formats of these files (field descriptions, data types, range of possible values, and possible formats) must be specified. The same must be specified for any data files that must be written to that are external to the system. This situation usually arises when the customer keeps his/her own spreadsheet or database and would like your system to be able to share files with the spreadsheet or database. If there are no interfaces to external files or systems, briefly state so. For example, using MySQL, or Cloud access.

3.3 Other

Place the remaining non-functional requirements here. Do not include any NFRs that are related to the user interface. These will be included in the User Interface Design Document. Also, do not include NFRs that relate to hardware or software. Hardware and software specifications will be part of the System Design Document. As stated earlier, if your customer has already restricted you to particular hardware or software, put these NFRs in the Customer Constraints section above.

4. **Deliverables**

Provide a list of all deliverable items (that is, all artifacts that you will deliver to the customer). This list will include items such as the product itself (What format? Source code? Executable code? Object code?), documentation, and training resources (if any). Specify when (date) and in what format (e.g., hard copy, CD) each will be delivered. You may assume that the deliverable items are as follows, although you may have more (e.g., training resources):

Hard copies of each of the following:

* Systems Requirement Specification
* System Design Document
* User Interface Design Document
* Administrator Manual

A CD and 3-ring binder (delivered at the product demo) containing the following:

* Documentation:
  + "Read Me First" document
  + System Requirements Specification
  + System Design Document
  + UI Design Document
  + Code Inspection Report
  + Test Report
  + Administrator Manual
* All source code
* The executable program (if applicable)
* Any other software required for program installation, etc.

Do not simply cut and paste this section into your document. Please come up with a more appropriate format. A tabular format works well.

5. **Open Issues**

List and briefly discuss issues that do not yet have a conclusion. Give specific target resolution dates. Be honest.

## Appendix A - Team percent contribution, Team sign off, Customer acceptance

**Sign off Agreement Between Customer and Contractor**

Describe what the customer and your team are agreeing to when all sign off on this document. [One paragraph] Include a statement that explains the procedure to be used in case there are future changes to the document. [One paragraph]

**Team Review Sign-off**

Provide a brief paragraph stating that all members of the team have reviewed the document and agree on its content and format. Provide lines for typed names, signatures, dates, and comments for each team member. The comment areas are to be used to state any minor points regarding the document that members may not agree with. Note that there cannot be any major points of contention.

**Document Contributions**

Remember that each team member must contribute to the writing (includes diagrams) for each document produced.

[file://localhost/Users/Stephen/Downloads/srsC.docx](Air:Users:Stephen:Downloads:srsC.docx#_blank)