Software Requirements Specification

Template

Version X.x ● November 14, 2016

Enter the company mission statement, text that describes the nature of this document, its role in the Software Development Lifecycle, or other text that clarifies its purpose.

Company Name

Street Address 1

Street Address 2

City

Zip/Postal Code

Country

Tel: +xxx-x-xxx-xxxx Fax: + xxx-x-xxx-xxxx

Last edited: 14 November 2016

Copyright © 2008 [Company]. All rights reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission from [Company].

All copyright, confidential information, patents, design rights and all other intellectual property rights of whatsoever nature contained herein are and shall remain the sole and exclusive property of [Company]. The information furnished herein is believed to be accurate and reliable.

However, no responsibility is assumed by [Company] for its use, or for any infringements of patents or other rights of third parties resulting from its use.

The [Company] name and [Company] logo are trademarks or registered trademarks of [Company].

All other trademarks are the property of their respective owners

Software Requirements Specification

[Organization Name]

[Project Name]

1. Version: [Version Number] Revision Date [Date]

|  |  |  |
| --- | --- | --- |
| **Executive Sponsor** | | |
| [Name] | [Email] | [Telephone] |
| Signature | | Date |

|  |  |  |
| --- | --- | --- |
| **Technology Sponsor** | | |
| [Name] | [Email] | [Telephone] |
| Signature | | Date |

|  |  |  |
| --- | --- | --- |
| **Project Manager** | | |
| [Name] | [Email] | [Telephone] |
| Signature | | Date |

|  |  |  |
| --- | --- | --- |
| **Security Officer** | | |
| [Name] | [Email] | [Telephone] |
| Signature | | Date |

Table of Contents

[1 Introduction 8](#_Toc210195675)

[1.1 Purpose of this document 8](#_Toc210195676)

[1.2 Scope 8](#_Toc210195677)

[1.3 Organization Profile 8](#_Toc210195678)

[1.4 Assumptions 9](#_Toc210195679)

[1.5 General Constraints 9](#_Toc210195680)

[1.6 Dependencies 9](#_Toc210195681)

[2 Product Functions 10](#_Toc210195682)

[2.1 Product Perspective 10](#_Toc210195683)

[2.2 Product Functions 10](#_Toc210195684)

[2.3 Operating Environment 10](#_Toc210195685)

[2.4 User Characteristics 11](#_Toc210195686)

[3 Detailed Functional Requirements 12](#_Toc210195687)

[3.1 Functional Requirement n 13](#_Toc210195688)

[3.1.1 Use Case X.Y 13](#_Toc210195689)

[3.2 Functional Requirement n 13](#_Toc210195690)

[3.2.1 Use Case X.Y 13](#_Toc210195691)

[3.3 Functional Requirement n 14](#_Toc210195692)

[3.3.1 Use Case X.Y 14](#_Toc210195693)

[4 System Requirements 15](#_Toc210195694)

[4.1 Software Requirements 15](#_Toc210195695)

[4.1.1 Software Functionality 15](#_Toc210195696)

[4.1.2 Software Characteristics 15](#_Toc210195697)

[4.2 Hardware Requirements 15](#_Toc210195698)

[4.2.1 Hardware Functionality 15](#_Toc210195699)

[4.2.2 Hardware Characteristics 15](#_Toc210195700)

[4.3 User Requirements 16](#_Toc210195701)

[4.4 Input and Output Requirements 16](#_Toc210195702)

[4.5 Communications Requirements 16](#_Toc210195703)

[4.5.1 Communications Hardware 17](#_Toc210195704)

[4.5.2 Communications Software 17](#_Toc210195705)

[4.6 Usability Requirements 17](#_Toc210195706)

[5 Non-Functional Requirements 18](#_Toc210195707)

[5.1.1 Accuracy 18](#_Toc210195708)

[5.2 Audit Trail 18](#_Toc210195709)

[5.3 Availability 18](#_Toc210195710)

[5.4 Capacity Limits 19](#_Toc210195711)

[5.5 Data Retention 19](#_Toc210195712)

[5.6 Operations 19](#_Toc210195713)

[5.7 Performance 19](#_Toc210195714)

[5.8 Portability 20](#_Toc210195715)

[5.9 Recoverability 21](#_Toc210195716)

[5.10 Reliability 21](#_Toc210195717)

[5.11 Security 21](#_Toc210195718)

[5.12 Timing 22](#_Toc210195719)

[6 Data Requirements 23](#_Toc210195720)

[6.1 Data Structures and Relationships 23](#_Toc210195721)

[6.2 Data Framework and Relationships 23](#_Toc210195722)

[6.3 Data Inputs 23](#_Toc210195723)

[6.4 Data Outputs 23](#_Toc210195724)

[6.5 Inter-functional Data Definitions 23](#_Toc210195725)

[6.6 Component Cross Reference 23](#_Toc210195726)

[7 External Interface Requirements 24](#_Toc210195727)

[7.1 Software Interfaces 24](#_Toc210195728)

[7.2 Hardware Interfaces 24](#_Toc210195729)

[7.3 Communications Interfaces 24](#_Toc210195730)

[7.4 User Interfaces 24](#_Toc210195731)

[8 Design Constraints 25](#_Toc210195732)

[8.1 Software Design Constraints 25](#_Toc210195733)

[8.1.1 Software Interfaces 25](#_Toc210195734)

[8.1.2 Software Packages 25](#_Toc210195735)

[8.1.3 Database 25](#_Toc210195736)

[8.1.4 Operating System 25](#_Toc210195737)

[8.1.5 Tolerance, Margins and Contingency 26](#_Toc210195738)

[8.2 Hardware Design Constraints 26](#_Toc210195739)

[8.2.1 Hardware Requirements and Environment 26](#_Toc210195740)

[8.2.2 Hardware Standards 26](#_Toc210195741)

[8.2.3 Hardware Interfaces 26](#_Toc210195742)

[8.3 User Interface Constraints 26](#_Toc210195743)

[8.3.1 User Characteristics 26](#_Toc210195744)

[8.3.2 Environment/Operational Constraints 26](#_Toc210195745)

[8.4 Memory Constraints 26](#_Toc210195746)

[9 Improvements and Impacts 27](#_Toc210195747)

[9.1 Improvements to Existing Capabilities 27](#_Toc210195748)

[9.1.1 Upgrades and Enhancements to Existing Capabilities 27](#_Toc210195749)

[9.2 Impacts 27](#_Toc210195750)

[9.2.1 User Impacts 27](#_Toc210195751)

[9.2.2 Operational Impacts 28](#_Toc210195752)

[10 Requirements Traceability Matrix 29](#_Toc210195753)

Document History

Paper copies are valid only on the day they are printed. Contact the author if you are in any doubt about the accuracy of this document.

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| 1. Revision Number | 1. Revision Date | 1. Summary of Changes | 1. Author |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Reference Documents

Please see the following documents for more information:

|  |  |  |
| --- | --- | --- |
| 1. Document Name | 1. Version | 1. Author |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Distribution List

This document has been distributed to:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. Name | | | | 1. Position | 1. Company | 1. Action | | |
|  | | | |  |  |  | | |
|  | | | |  |  |  | | |
|  | | | |  |  |  | | |
|  | | | |  |  |  | | |
|  | | | |  |  |  | | |
|  |  |  | | | |  |

# Introduction

This phase of the Software Development Lifecycle is required to understand and document the users' needs for the system. The Functional Requirement document captures, in significantly more detail than the Project Statement, the scope, business objectives, and requirements of the current/proposed system.

The emphasis throughout this document is on what the system will do. During analysis and specification, the technical aspects and constraints should be considered, but should not be influenced by how it will be implemented. The technical aspects of the system will be addressed in the Design Phase.

## Purpose of this document

Introduce the purpose of the particular software product and specify the intended audience, including the revision or release number.

For example: This Software Requirements Specification defines the functional and non-functional requirements for [system].

## Scope

Describe the scope of the product that is covered by these Software Requirements Specifications, particularly if this document describes only part of the system or a single subsystem. The scope establishes the boundaries of the requirements and should identify clarify features/requirements outside of scope, for example, if certain requirements were not included due to budgetary or time constraints.

|  |
| --- |
| 1. Scope includes |
|  |
|  |
|  |

|  |
| --- |
| 1. Scope excludes |
|  |
|  |
|  |

## Organization Profile

Provide information about the organization sponsoring this document, e.g. locations, numbers, personnel, and relationships or interfaces with other organizations and entities. You may also want to discuss the user groups and their levels of proficiency with the system, especially if issued were identified during the requirements gathering process.

## Assumptions

List and describe any assumed factors (as opposed to known facts) that could affect the requirements. These could include third-party or commercial components that you plan to use, issues around the development, operating environment, company policy regarding hardware procurement or preferences towards specific programming languages. The project could be negatively affected if these assumptions are incorrect or change during the project lifecycle.

|  |  |
| --- | --- |
| 1. Assumption | 1. Impact |
|  |  |
|  |  |
|  |  |
|  |  |

## General Constraints

Constraints are conditions on how the system must be designed and constructed, such as legal requirements, technical standards, or strategic decisions. Constraints exist because of real business conditions, for example, a delivery date is a constraint only if there are real business consequences that will happen as a result of not meeting the date.

Specific design constraints are discussed later in the chapter, *Design Constraints*.

|  |  |
| --- | --- |
| 1. Constraint | 1. Impact |
|  |  |
|  |  |
|  |  |
|  |  |

## Dependencies

Identify any dependencies that the project has on external factors, such as software components that you intend to reuse from another project or technical resources that must be in place for the system to operate.

|  |  |
| --- | --- |
| 1. Dependency | 1. Description |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Product Functions

This chapter provides a general description of the product(s) characteristics. It does not state specific requirements; these sections provide information that makes the requirements, defined in detail in the following chapters, easier to understand.

## Product Perspective

Describe the origin of the product being specified in this document. For example, state whether it is a follow-on member of a product family, a replacement for an existing system, or a new product. If the functional requirements define a component(s) of a larger system, relate the requirements of the larger system to the functionality of this product and identify the interfaces between them. Provide a diagram that illustrates the major components of the system, subsystem interconnections, and external interfaces.

## Product Functions

Summarize how the functions are organized in such a way that that they can be easily understood to the reader.

For example: The product produced from these requirements shall:

* Add customers to the Contacts database
* Add, change or delete invoices
* Prepare billing and invoicing functionality
* Provide details of customer credit ratings

|  |  |
| --- | --- |
| 1. Function | 1. Description |
| 1. #1 |  |
| 1. #2 |  |
| 1. #3 |  |
| 1. #4 |  |
| 1. #5 |  |

## Operating Environment

Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must coexist. This helps place the system in context from a technical perspective.

## User Characteristics

Describe the characteristics of user groups who will interact with the system and any characteristics that might affect the system design, such as educational level, experience, technical expertise and geographical location.

|  |  |  |
| --- | --- | --- |
| 1. Role Name | 1. No. of Users | 1. Responsibility / Activity |
| 1. User |  |  |
| 1. Super User |  |  |
| 1. Administrator |  |  |
| 1. Etc |  |  |
|  |  |  |

Table 1 — User Roles

# Detailed Functional Requirements

In this chapter, itemize the functional requirements associated with a feature. These are the software capabilities that must be present for the user to perform the services provided by the feature.

The following sub- sections contain all the software requirements to a level of detail sufficient to enable designers to design and testers to test the system to satisfy those requirements.

This section specifies-at a minimum-the transformation of inputs into outputs and all functions performed by the system in response to an input or in support of an output. This description may consist of a model of the requirements, such as data flow diagrams, entity relationship diagrams, state transition diagrams, and a data dictionary.

Functional requirements specify functions that a system or component must be able to perform, for example: "Display the heart rate, blood pressure and temperature of a patient."

Typical functional requirements include

* Business Rules
* Transaction corrections, adjustments, cancellations
* Administrative functions
* Authentication
* Reporting Requirements
* Legal Requirements

Each requirement should be uniquely identified with a sequence number. This means that you must number each requirement with a numbering scheme that allows you to insert additional requirements later, for example, FR-01, FR-02, FR-03 etc. Requirements should be concise, complete, unambiguous, verifiable, and necessary.

Define one requirement per numbered item only.

Functional Requirements define the actions that must take place in the software when accepting and processing the inputs and in processing and generating the outputs. These are generally listed as “shall” statements, starting with “The software shall…”

This chapter consists of *n* subsections depending on the number of requirements that must be captured to define the system. The following sub-section provides a sample format that allows you to capture your Functional Requirements; you can modify this to suit your needs.

## Functional Requirement n

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Priority | 1. 1=High, 2= Med, 3 = Low |
| 1. Purpose | 1. The software shall… |
| 1. Input | 1. Describe the inputs to the function, including sources, valid ranges of values, timing considerations, operator requirements, and special interfaces. |
| 1. Operations | 1. Describe the operations to be performed within the function, including validity checks, responses to abnormal conditions, and types of processing required. |
| 1. Output | 1. Describe the outputs from the function, including output destinations, valid ranges of values, timing considerations, and considerations for handling of illegal values, error messages, and interfaces required. |

Provide a X.x.nf subsection for each function when functional decomposition is used to specify the requirements. Label and title each subsection appropriately for a specific function, where nf is the sequential subsection number and X is the name of the specific function.

### Use Case X.Y

When use cases are used to specify requirements, provide a sub-function for each use case. Label and title each sub-function appropriately for a specific use case, where X is the sequential sub-function number and Y is the name of the specific use case.

Within each use case sub-function, specify the use case information, including the actor, pre-conditions, post-conditions, scenarios, and alternate scenarios.

## Functional Requirement n

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Priority | 1. 1=High, 2= Med, 3 = Low |
| 1. Purpose |  |
| 1. Input |  |
| 1. Operations |  |
| 1. Output |  |

### Use Case X.Y

When use cases are used to specify requirements, provide a sub-function for each use case. Label and title each sub-function appropriately for a specific use case, where X is the sequential sub-function number and Y is the name of the specific use case.

## Functional Requirement n

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Priority | 1. 1=High, 2= Med, 3 = Low |
| 1. Purpose |  |
| 1. Input |  |
| 1. Operations |  |
| 1. Output |  |

### Use Case X.Y

When use cases are used to specify requirements, provide a sub-function for each use case. Label and title each sub-function appropriately for a specific use case, where X is the sequential sub-function number and Y is the name of the specific use case.

# System Requirements

## Software Requirements

### Software Functionality

Describe the software’s required capabilities, e.g. databases, operating systems, and diagnostics.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Software Functionality |
| 1. [Req #] | 1. The software shall… |
| 1. [Req #] | 1. The software shall… |
| 1. [Req #] | 1. The software shall… |

### Software Characteristics

Describe the required characteristics of the software, e.g. reusability of code.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Software Characteristics |
| 1. [Req #] | 1. The software shall… |
| 1. [Req #] | 1. The software shall… |
| 1. [Req #] | 1. The software shall… |

## Hardware Requirements

### Hardware Functionality

Describe the required capabilities of the hardware, e.g., support multiple operating systems.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Hardware Functionality |
| 1. [Req #] | 1. The hardware shall… |
| 1. [Req #] | 1. The hardware shall… |
| 1. [Req #] | 1. The hardware shall… |

### Hardware Characteristics

Describe the characteristics of the hardware.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Hardware Characteristics |
| 1. [Req #] | 1. The hardware shall… |
| 1. [Req #] | 1. The hardware shall… |
| 1. [Req #] | 1. The hardware shall… |

## User Requirements

Describe the requirements of the system, user or business, with consideration to all major categories of users. Provide the type of security or other distinguishing characteristics of each set of users.

User requirements often use a numbering system that is separate from the functional requirements. For example, you can label the requirements with a leading “U” or other label indicating user requirements.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall allow the user to… |
| 1. Req # | 1. The software shall allow the user to … |
| 1. Req # | 1. The software shall allow the user to … |

## Input and Output Requirements

Describe manual and automated input requirements such as data entry and data extracts from other applications.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Input Requirements |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

Describe the output requirements for the software product, such as printouts, reports, files and other outputs that the system will process and produce.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Output Requirements |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Communications Requirements

Describe the system’s communication requirements. Specify the desired response times where appropriate. Provide a diagram of the system’s communication requirements, including type and peak data volumes.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

### Communications Hardware

Describe communication hardware requirements, such as storage devices, input devices, and printers.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The communication hardware shall… |
| 1. Req # | 1. The communication hardware shall… |
| 1. Req # | 1. The communication hardware shall… |

### Communications Software

Describe communication software requirements for the proposed system, such as compilers, operating system, and database management systems.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The communication software shall… |
| 1. Req # | 1. The communication software shall… |
| 1. Req # | 1. The communication software shall… |

## Usability Requirements

Define the usability requirements, such as menu structures, screen colors, navigation, and online help.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

# Non-Functional Requirements

Non-functional requirements specify the requirements not covered by the functional requirements. They specify criteria that judge the operation of a system, rather than specific behaviors. Typical non-functional requirements include Availability, Performance, Response Time, and Throughput.

### Accuracy

Describe the accuracy requirements to be imposed on the system.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Audit Trail

List the activities recorded in the application’s audit trail. For each activity, list the data recorded.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Availability

State the time periods during which the system must be available to users, for example, “The system must be available to users Monday through Friday between 06:00 and 18:00 GMT.

If the application will be available in several time zones, state their earliest start and latest stop times. Consider daylight savings time. Identify peak times, i.e. when system unavailability is least acceptable.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Capacity Limits

Specify the system’s capacity requirements in relation to the maximum numbers of transactions, concurrent users, and other quantifiable information. List the required capacities and expected volumes of data in business terms.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Data Retention

Identify the length of time data must be retained and requirements for its archival and destruction. For example, “The system shall retain information for 10 years”. Identify different forms of data: system documentation, audit records, and database records.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Operations

Describe operational requirements and contingencies for areas such as failure modes, start-up and close-down, maintenance periods, error and recovery handling.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Performance

Describe specific performance requirements for the system and subsystems. Provide details of requirements such as, the number of events that must be processed, response times, maximum data volumes to be stored, number of inputs and outputs connected, and the number of transactions to be processed in a specified time.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Portability

Specify attributes that relate to the ease of porting the software to other host machines and/or operating systems. This may include:

* Percentage of components with host-dependent code
* Percentage of code that is host dependent
* Use of a proven portable language
* Use of a particular compiler
* Use of a particular operating system

Once you’ve selected the relevant characteristics, explaining the rationale for including this characteristic and how it will be tested and measured. The following table may help identify the key characteristics (rating High or Medium), then identifying which are preferred when trading off design or implementation decisions.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Characteristic | H/M/L | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 1 | Correctness |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Efficiency |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Flexibility |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Integrity/Security |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Interoperability |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Maintainability |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Portability |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 | Reliability |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 9 | Reusability |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 10 | Testability |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 11 | Usability |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 12 | Availability |  |  |  |  |  |  |  |  |  |  |  |  |  |

## Recoverability

Recoverability is the ability to restore function and data in the event of a failure. Identify recoverability requirements, such as: if the application is unavailable to users because of a system failure, how soon after failure is detected must functionality be restored?; if a database is corrupted, to what level of currency must it be restored?; if the processing site (hardware, data, and onsite backup) is destroyed, how soon must the application be able to be restored?

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Reliability

Reliability is the probability that the system processes work correctly and completely without being aborted.

Identify damage that can result from system failure, such as loss of life, complete or partial loss of the ability to perform a mission-critical functions; loss of revenue; loss of productivity, and minimum acceptable level of reliability.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Security

Provide a list of the security requirements with consideration to the following:

* Identify the type(s) of security required, such as physical security ad access by user role or types.
* Identify security classification, protection types, and controls for user access.
* Identify security requirements for PC, server, network, dial-up access etc.
* Identify the consequences of the following breaches of security: loss of data; disclosure of sensitive information or privacy information; corruption of software, introduction of viruses.
* Identify access control requirements by data attribute. For example, user group A has permission to view an attribute but not update it while user group B has permissions to update or view it.
* State if there is a need for certification and accreditation of the security measures.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

## Timing

Describe the timing requirements to be imposed on the system, such as, response time between input data and system availability, response time to queries, and deviations from specified response times.

|  |  |
| --- | --- |
| 1. [Req #] | 1. Title |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |
| 1. Req # | 1. The software shall… |

# Data Requirements

Describe how the system will process and store the data elements and logical data groupings. Identify data archiving requirements.

## Data Structures and Relationships

Identify the main inputs and outputs from the target system. Identify where the data is stored and inter-functional flows within the system.

## Data Framework and Relationships

Define the relationships between data items. Use entity-relationship diagrams (ERDs) to describe the data structures.

## Data Inputs

Identify all inputs to the target system. Specify their structure, e.g. subparts and inter-relationships.

|  |  |
| --- | --- |
| 1. [#] | 1. Data Input |
| 1. # |  |
| 1. # |  |
| 1. # |  |

## Data Outputs

Identify all outputs to the target system. Specify their structure, e.g. subparts and inter-relationships.

|  |  |
| --- | --- |
| 1. [#] | 1. Data Output |
|  |  |
|  |  |
|  |  |

## Inter-functional Data Definitions

Identify any data items within the target system that carry data between the functional components.

## Component Cross Reference

Describe how particular functional components use data items. Provide a matrix to illustrate the cross reference.

# External Interface Requirements

## Software Interfaces

Identify the applications with which the subject application must interface. State the following for each such application: name of application, application owner, interface details (if determined by the other application).

Include the use of other required software products (e.g. a RDBMS) and interfaces with other applications systems (e.g. the linkage between a Finance system and HR system).

|  |  |  |  |
| --- | --- | --- | --- |
| 1. Software Interfaces | | | |
| 1. Name | 1. Specification # | 1. Version # | 1. Source |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Hardware Interfaces

Provide a detailed description of hardware interfaces. Description includes complete technical specification of and the perceived limitations to each defined hardware interface.

## Communications Interfaces

Describe the communications interfaces to other systems or devices, such as local area networks. Include a detailed description of software interface to other communication packages/interfaces, including a technical specification of each defined communication package/interface.

## User Interfaces

Describe how the user interfaces will be designed for this particular function.

# Design Constraints

This section specifies the constraints imposed on the system by compliance to software standards and hardware limitations. Constraints are grouped into three areas covering software, hardware and user interfacing.

Describe any items that will limit the options available to the developers, such as regulatory policies, hardware limitations (memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards.

## Software Design Constraints

The following requirements may constrain the design of [system]:

* The system shall deliver content in multiple languages over web, mobile and other channels
* The system shall enable authorized users to work off-line and upload data entered and transactions performed later.
* The system shall track the user who performs a transaction
* The system shall time-stamp all transactions.

|  |  |
| --- | --- |
| 1. # | 1. Software Constraint |
|  |  |
|  |  |
|  |  |

### Software Interfaces

Describe requirements for internal interfaces to software modules necessary to ensure coherent operation within the overall system. Include constraints to ensure inter-operability with other systems.

### Software Packages

Discuss any specific packages, which the users or developers might require.

### Database

State if the system must be implemented using a specific database and specific databases which the user might require.

### Operating System

State if the system must be implemented on a specific operating system.

### Tolerance, Margins and Contingency

Identify any constraints on program size and performance, indicating margins and contingency for the expansion of data handling capabilities.

## Hardware Design Constraints

### Hardware Requirements and Environment

Specify the hardware requirements to the functional component specification. Describe the working environment, mandatory level of hardware reliability, and mechanical and physical constraints.

|  |  |
| --- | --- |
| 1. # | 1. Hardware Constraints |
|  |  |
|  |  |
|  |  |

### Hardware Standards

Identify the engineering standards relating to safety, power supplies, electrical and other interference.

### Hardware Interfaces

Discuss the requirements for electrical and mechanical interconnections.

## User Interface Constraints

### User Characteristics

Identify which types of users will interact with specific functional components.

### Environment/Operational Constraints

Identify operational requirements for styles of interaction and details of the operational environment, if they affect the user interfaces.

## Memory Constraints

Identify limits on primary and secondary memory, for example:

|  |  |
| --- | --- |
| 1. # | 1. Memory Constraint |
|  | PCs shall run on a system with at least 32 meg. RAM. |
|  | 1. Host PC shall install and run within 120 megabytes of free hard drive space. |
|  |  |

# Improvements and Impacts

Describe the proposed methods and procedures to the new system, including existing system components which will be incorporated to the new design. Where appropriate, outline how the new system will eliminate or degrade any capabilities in an existing system.

## Improvements to Existing Capabilities

Discuss proposed functional improvements over the current system.

|  |  |
| --- | --- |
| 1. Current System | 1. Proposed System |
|  |  |
|  |  |
|  |  |

### Upgrades and Enhancements to Existing Capabilities

Discuss upgrades and enhancements to existing capabilities.

|  |  |
| --- | --- |
| 1. # | 1. Upgrade and Enhancements |
| 1. 1 |  |
| 1. 2 |  |
| 1. 3 |  |

## Impacts

Summarize the associated costs of the new system on the existing organization and the anticipated impacts on the operational environments.

### User Impacts

Describe the addition, change, or elimination of user responsibilities in order to use the new system.

Identify the following areas: Roles/functions that may be eliminated; Number/skills of additional personnel; Changes in staffing levels, location, and position; Number/skills required for contingency operations.

|  |  |
| --- | --- |
| 1. # | 1. User Impacts |
| 1. 1 |  |
| 1. 2 |  |
| 1. 3 |  |
| 1. 4 |  |

### Operational Impacts

Identify the operational impacts that the new system will have on the existing operations. Outline the proposed interface between the user and other operating centers. Identify the following:

* Data input methods
* Data quantity, type, and timeliness
* Data retention requirements
* Impact on users
* New data sources

|  |  |
| --- | --- |
| 1. # | 1. Operational Impact |
| 1. 1 |  |
| 1. 2 |  |
| 1. 3 |  |

# Requirements Traceability Matrix

Identify the location of the Requirements Traceability Matrix. This provides traceability from the functional requirements documented in this document to the design elements documented in the System Design Document.