Requirements Document

Report

Author: Xingrong Zong

Supervisor: [Supervisor]

Semester: VT21

Course code: 2DV608

1. Table of contents

[1 Introduction 3](#_Toc64061690)

[1.1 Purpose of the requirements document 6](#_Toc64061691)

[1.2 Scope of the product 6](#_Toc64061692)

[1.3 Definitions, acronyms, and abbreviations 7](#_Toc64061693)

[1.4 References 7](#_Toc64061694)

[1.5 Overview of the remainder of the document 7](#_Toc64061695)

[2 General description 7](#_Toc64061696)

[2.1 Product perspective 7](#_Toc64061697)

[2.2 Product functions 7](#_Toc64061698)

[2.3 User characteristics 7](#_Toc64061699)

[2.4 General constraints 7](#_Toc64061700)

[2.5 Assumptions and dependencies 8](#_Toc64061701)

[3 Specific functional, non-functional, and interface requirements 8](#_Toc64061702)

[4 Appendix 8](#_Toc64061703)

[5 Index 8](#_Toc64061704)

# Introduction

Netflix: Imagine designing and developing an online video streaming service.

Amazon: Imagine designing and developing and online shopping portal.

**Stakeholders**: identify the complete list of stakeholders

End-users of the system – users who use Netflix to watch videos and who use Netflix to stream videos.

Managers and others involved in the organizational processes -

Engineers responsible for the system development and maintenance – software engineers.

Customers of the organization who will use the system to provide some services.

External bodies such as regulators or certification authorities, etc – safety certification authorities.

Requirements: elicit and label three **functional** and three **non-functional** requirements (properly label requirements)

Stakeholders requirements:

* End-users and customers need a high-level statement of the requirements.
* System developers need a more detailed system specification.

System requirements:

Requirements **analysis**: perform a systematic ‘checklist-based’ requirements analysis.

* Premature design
* Combined requirements
* Unnecessary requirements - remove
* Use of non-standard hardware
* Conformance with business goals
* Requirements ambiguity - modify
* Requirements realism
* Requirements testability

Requirements **classification**: classify the identified requirements using the ‘faceted approach’.

Identify the number of dimensions or facets and identify keywords which describe each of these.

* System – requirements that affect the entire system such as performance or reliability requirements.
* User interface – requirements that are concerned with user interaction.
* Database – requirements that are concerned with the data managed by the system.
* Communication – requirements that are concerned with the external communication facilities in the system.

**Risk** **assessment** of requirements: systematically assess the risks of your requirements.

* Performance risks – may affect the overall performance of the system.
* Safety and security risks – may cause problems in meeting overall system requirements for safety and security.
* Process risks – may require changes to the normal development process/
* Implementation technology risks – may require the use of unfamiliar implementation technology, such as AI techniques, the use of N-version programming for fault tolerance, etc.
* Database risks – may involve non – standard data which is not available in an existing system database.
* Schedule risks – may be technically difficulty and may threaten the planned development schedule for the system.
* External risks – involves external contractors.
* Stability risks – requirement may be volatile and subject to evolution during the development process.

**Systematic** **validation** of requirements: perform a ‘systematic validation’ of the requirements.

Concerned with checking a final draft of a requirements document which includes all system requirements and where known incompleteness and inconsistency has been removed. The document and the requirements should follow defined quality standards.

Convert the specification described by the graphical model or formal notation to a natural language representation.

Use forms or tables where different components in the model are described in different fields or columns.

Data flow diagram

* Are the requirements complete – does the checker know of any missing requirements or is there any information missing from individual requirement descriptions?
* Are the requirements consistent – do descriptions of different requirements include contradictions?
* Are the requirements comprehensible – can readers of the document understand what the requirements mean?
* Are the requirements ambiguous – are there different possible interpretations of the requirements?
* Is the requirements document structures – are the descriptions of requirements organized so that related requirements are grouped?
* Are the requirements traceable – do the requirements include links to related requirements and to the reasons why these requirements have been included?
* Does the requirements document as a whole, or to do the individual requirements conform to defined standards?

**Test** **cases** for requirements: propose one test case for each requirement.

* What **usage scenario** (the context the test be applied) might be used to check the requirement?
* Does the requirement include enough **information** to allow a test to be defined?
* Is it possible to check the requirement using a single test or are **multiple** test cases required?
* Design a **Test Recording Form** for each requirement
  + The requirement **identifier**
  + Related requirements
  + A brief description of the test which could be applied and why this is an **objective** requirements test
  + A description of requirements **problems** which made test definition difficult or impossible
  + **Recommendation** for addressing requirements problems which have been discovered

Requirements document: create a final ‘requirements document’ with the above 7 topics.

## Purpose of the requirements document

The requirements document is an official statement of the system requirements for customers, end-users, ad software developers.

How to use the document:

Target audience: Users of this document

* System Customers: Specify the requirements and read them to check that they meet their needs. Customers specify changes to the requirements.
* Managers: Use the requirements document to plan a bid for the system and to plan the system development process.
* System Engineers: Use the requirements to understand what system is to be developed,
* System Maintenance Engineers: Use the requirements to understand the system and the relationships between its parts.
* System Test Engineers: Use the requirements to develop validation tests for the system.

Technical background:

Pointers to overview sections:

## Scope of the product

Requirements: elicit and label three **functional** and three **non-functional** requirements (properly label requirements)

Functional requirements:

* The system should enable the user to watch a video.
* The system should enable the user to choose the quality of a video.
* The system should enable the user to choose the playback speed of a video.
* The system must provide some facility for authenticating the identity of s system user.

Non-functional requirements:

* The authentication process should be completed in 5 seconds or less.

## Definitions, acronyms, and abbreviations

* Functional requirements = FRs
* Non-functional requirements = NFRs

## References

## Overview of the remainder of the document

Summarize the purpose of the system and principal system requirements.

Organize the summary on a per chapter basis.

# General description



## Product perspective

## Product functions

## User characteristics

## General constraints

## Assumptions and dependencies

# Specific functional, non-functional, and interface requirements

Eg: external interfaces, functionality, performance requirements, logical database requirements, design constraints, system attributes, and quality characteristics

# Appendix

# Index