User Requirements Document

Group 1

April 24, 2013

Abstract

This is the User Requirements Document for the Software Engineering Project. This document is based on the ESA standard for software development and the work of many previous SEP groups.

Contents

Document Status Sheet

Document status overview

General

Document title:	User Requirements Document
Identification:	Choose some id, e.g.: SQAP0.0.pdf
Author:	
Document status:	Draft

Document history

Version	*Date*	*Author*	*Reason of change*
version	date	author	reason

Document Change Records since previous issue

General

Datum:	2012-07-09
Document title:	Software Quality Assurance Plan
Identification:	SQAP0.0.pdf

Changes

Page	*Paragraph*	*Reason to change*
`pageref`	`ref`	reason

Chapter 1

Introduction

1.1 Purpose

This document describes the procedures and control methods to obtain the desired quality level of the end products and the process by which these end products are created. This document serves as a guide for the managers and developers of the Project Name project. All team members must read this document and apply the procedures stated in it. The document applies to all phases of software development as defined in the Project Management Plan [**?**]. Detailed information about the software quality assurance activities for these phases will be added in appendices during the project.

1.2 Scope

A list of software products to be developed and their intended use.

1.3 List of definitions

2IP35	The Software Engineering Course
AD	Architectural Design
ADD	Architectural Design Document
AT	Acceptance Test
ATP	Acceptance Test Plan
Client	The client
CM	Configuration Manager
DD	Detailed Design
DDD	Detailed Design Document
ESA	European Space Agency
TU/e	Eindhoven University of Technology
OM	Operations and Maintenance Plan
PM	Project Manager
QM	Quality Manager
SCMP	Software Configuration Management Plan
SEP	Software Engineering Project
SL	Software Librarian
SPMP	Software Project Management Plan
SQAP	Software Quality Assurance Plan
SR	Software Requirements
SRD	Software Requirements Document
STD	Software Transfer Document
SUM	Software User Manual
SVVP	Software Verification and Validation Plan
SVVR	Software Verification and Validation Report
TR	Transfer phase
UR	User Requirements
URD	User Requirements Document
VPM	Vice Project Manager

1.4 List of references

TODO: only all applicable documents!

1.5 Overview

Short description of the rest of the SRD and how it is organized.

Chapter 2

General description

2.1 Relation to current projects

The context of this project in relation to other current projects.

2.2 Relation to predecessor and successor projects

The context of this project in relation to past and future projects.

2.3 Function and purpose

A general overview of the function and purpose of the product.

2.4 Environment

Hardware and operating system of target system and development system. Who will use the system (user roles in URD).

2.5 Relation to other systems

Is the project an independent system, part of a larger system, replacing another system? The essential characteristics of these other systems.

2.6 General constraints

Reasons why constraints exist: background information and justification (analogues to URD).

2.7 Model description

A description of the logical model.

Chapter 3

Specific requirements

3.1 Functional requirements

A list of all functional requirements (what should the system do).

01	*could have*
Users can set a geometry for the canvas	
02	*must have*
Users can define a initial concentration distribution with black and white	
03	*could have*
Users can choose which two colors are used for the initial concentration distribution	
04	*should have?*
Users can define a initial concentration distribution with more than two different colors	
05	*must have*
Users can define a mixing protocol for a rectangular geometry as a sequence of movements of the upper and lower walls	
06	*could have*
Users can define a mixing protocol for a non-rectangular geometry as a sequence of movements that are applicable to the geometry	
07	*must have*
Users can define a step to indicate the timeperiod that each movement from the mixing protocol is applied	
08	*could have*
Users can define a different step for each separate movement in the mixing protocol	
09	*must have*
Users can view an image of the endresult of applying the mixing protocol on the initial concentration distribution	
10	*should have*
Users can save the image from 06 locally to their device, without losing transparency (i.e. PNG or GIF format)	
11	*should have*
Users can remove previously stored images from their device	
12	*should have*
Users can view an animation of applying the mixing protocol on the initial concentration distribution	
13	*should have*
Users can save the animation from 09 locally to their device, without losing transparency (i.e. APNG or AGIF format	
14	*should have*
Users can remove previously stored animations from their device	
15	*should have*
Users can view the mixing performance of the mixing protocol in a graph	
16	*should have*
Users can save the performance results locally on their device	
17	*should have*
Users can retrieve the performance results that are stored locally on their device	
18	*should have*
Users can retrieve performance results from multiple mixing protocols simultaneously, after which they are depicted in one graph	
19	*should have*
Users can remove performance results that are stored on their device	

3.2 Non-functional requirements

A list of all non-functional requirements (performance, interface, operational, resource, verification/testing, portability, maintainability, reliability, security, safety, documentation, other, ...), linked to functional requirements. Each category of non-functional requirements has its own subsection.

01	must have
The interface contains a canvas which represents the mixing area	
02	should have
The user can define the initial concentration distribution by painting on the canvas with his/her finger	
03	must have
The interface contains an easy to use input element to define the sequence of movements of the mixing protocol (i.e. a button or by swiping over the screen)	
04	must have
The interface contains a numberfield to set the step parameter of the mixing protocol	
05	must have
Waiting time between submitting input and receiving output should not be more than 5 seconds	
06	should have
Waiting time between submitting input and receiving output should not be more than 3 seconds	
07	could have
Waiting time between submitting input and receiving output should not be more than 1 seconds	

Chapter 4

Requirements traceability matrix

A table showing how each user requirement of the URD is linked to software requirements in the SRD.