Requirement Analysis and Specification Document RASD

KONG XIANGYI   ZHANG YUEDONG

October 30, 2020

ii

Contents

Chapter 1

Introduction

1.1 Purpose

This document is Requirement Analysis and Specification Document(RASD). The main purpose of this document is the following points

- Communicates an understanding of the requirements to the audience and explains both the application domain and the system to be developed.

- Contractual: Make this project formal and written so that it has legal effect.

- As the baseline for project planning and estimation. i.e. size, cost, schedule.

- As the baseline for software evaluation

    It can support system testing, verification and validation activities

    It should contain enough information to verify whether the delivered system meets requirements

- As the baseline for change control, such as requirements change, software evolves.

And this RASD has the following intended audiences

- Costumers & Users : Some user may interest in validating system goals and high-level description of functionalities.

- Systems and Requirements Analysts: The RASD may help them to write various specifications of other systems that inter-relate.

- Developers, Programmers: The RASD may help the to implement the requirements

- Testers: The RASD may help the to determine that the requirements have been met

- Project Managers: The RASD may help them to measure and control the analysis and development processes

1.2   Scope

1.2.1   Description of the given problem

At the end of 2019, a global epidemic broke out, swept almost all countries in the world in a few months, starting in 2020, people's life rhythm has been completely disrupted by this epidemic, a lot of cities are blocked, people are allowed to exit their homes only for essential needs, all had to wear masks and respect the social-distancing at least 1.5 m. In the public area, the human community has to take measures to avoid the crazy spread of the virus, restaurant began to use dividers separate the tables, supermarkets and museums began to restrict flow, the school also adopted two classes mode online and onsite.

In this context, a new problem arises, how to delay the spread of the virus through technical means?

Since grocery shopping is the most needed activity under the lock-down, so let's narrow the problem to grocery shopping.

In the supermarket, In order to meet these strict rules, many challenges have arisen, so, we can turn to technology, in particular to software applications, to help navigate the challenges created by the imposed restrictions.

So, this project appeared - Customers Line-up(CLup).

Customers Line-up(CLup) is an easy-to-use application, it has two main goals.

- First of all, the CLup have to allow store managers to regulate the influx of people in the building.

- And then, it will saves people from having to line up and stand outside of stores for hours on end.

1.2.2 World Phenomena

WP_1	cell
WP_2	cell
WP_3	cell

1.2.3 Shared Phenomena

SP_1	cell
SP_2	cell
SP_3	cell

1.2.4 Goals

G_1	cell
G_2	cell
G_3	cell

1.3 Definitions, acronyms, abbreviations

1.3.1 Definitions

- Customer:

- QR code:

- Store Manager:

1.3.2 Acronyms

- RASD – Requirement Analysis and Specification Document

- CLup - Customers Line-up

- UI - User Interface

1.3.3 Abbreviations

- WP_n : n-th world phenomena

- SP_n : n-th shared phenomena

- G_n : n-th goal

- D_n : n-th domain assumption

- R_n : n-th functional requirement

1.4   Reference documents

- Specification Document: "R&DD Assignment A.Y. 2020-2021"

- Slides of the "Software Engineering 2" course A.Y. 2020-2021

- IEEE Recommended Practice for Software Requirements Specifications - IEEE Std 830-1998

1.5   Overview

The RASD document consists of five chapters.

Chapter 1 is the introduction chapter, it's an overview of the RASD and project, it describes the purpose of the CLup.

Chapter 2
Chapter 3

Chapter 2

Overall Description

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.4 Constraints

2.5 Assumptions and Dependencies

Chapter 3

Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communication Interfaces

3.2   Functional Requirements

3.2.1   User Class 1

Functional Requirement 1.1

3.2.2   User Class 2

Functional Requirement 2.1

3.3 Performance Requirements

3.4   Design Constraints

3.4.1   Standards compliance

3.4.2   Hardware limitations

3.5 Software System Attributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Maintainability

3.5.5 Portability

3.6   Other Requirements