<Online Auction>

Requirements Specification and Analysis

<1.0>

<11.11.2018>

<TOLGA GÜLDÜTUNA

FEYZULLAH BEERKAY DANIŞ

BERKAY YILMAZ

TEKİN EVRİM>

Prepared for

SE301 Software Engineering



Table of Contents

[1. Introduction 1](#_Toc496873294)

[1.1. Purpose of the System 1](#_Toc496873295)

[1.2. Scope of the System 1](#_Toc496873296)

[1.3. Objectives and Success Criteria of the Project 1](#_Toc496873297)

[1.4. Definitions, Acronyms, and Abbreviations 1](#_Toc496873298)

[1.5. Overview 1](#_Toc496873299)

[2. Current System 1](#_Toc496873300)

[3. Proposed System 1](#_Toc496873301)

[3.1. Overview 1](#_Toc496873302)

[3.2. Functional Requirements 2](#_Toc496873303)

[3.3. Nonfunctional Requirements 2](#_Toc496873304)

[Usability 2](#_Toc496873305)

[Reliability 2](#_Toc496873306)

[Performance 2](#_Toc496873307)

[Supportability 2](#_Toc496873308)

[Implementation 2](#_Toc496873309)

[Interface 2](#_Toc496873310)

[Packaging 2](#_Toc496873311)

[Legal 2](#_Toc496873312)

[3.4. System Models 2](#_Toc496873313)

[Scenarios 2](#_Toc496873314)

[Use case model 2](#_Toc496873315)

[Object model 2](#_Toc496873316)

[Dynamic model 2](#_Toc496873317)

[User interface—navigational paths and screen mock-ups 3](#_Toc496873318)

[3.5. Project Schedule 3](#_Toc496873319)

[4. Glossary 3](#_Toc496873320)

[5. References 3](#_Toc496873321)

REQUIREMENTS ANALYSIS DOCUMENT[1]

\*\*

# Introduction

\*\*

## Purpose of the System

The purpose of the auction app is to provide an app where people can buy or sell products on an online auction by using their phones

## Scope of the System

Users will be able to search the products that they want by entering information about the product and view any product on the auction app, they will be able to see information about the product such as price, category, name and other information depending on the category. Users can sign up to the system and by doing that they will be able to buy a product on the auction app by making an offer for the product on the auction. Registered users will also be able to put a product on an auction and sell it on the auction app but in order to do that they need to get approval from admin. Admin will be able to edit user profiles or delete a user. Registered users will be able to freeze their account and by doing that they will stop any kind of action taking place through their account. If users confronts any problem while using the system they will be able to communicate with admin.

## Objectives and Success Criteria of the Project

The purpose of the Online Auction System is to provide the users a mobile app that allowes people to buy or sell products on online auctions. This app will provide users a service that they will be able to make offer and buy a product on an auction or sell a product on an auction using very little number of steps. Users will be able to do that through their phone anywhere. They will be supported if they confronted a problem while using the system

## Definitions, Acronyms, and Abbreviations

Admin: Admin is an actor in auction app, admin is the actor who manages everything about auction app.

Registered User: An actor of the system, registered user is a user who signed up to the system to be able to buy or sell products.

User: A user who has not signed up to the system and so visitor is not able to buy or sell products.

## Overview

\*Rest of the RAD contains non-functional (includes usability, reliability,

performance, supportability, implementation, interface, operational, packaging,

and legal requirements) and functional requirements (includes high-level

functionality of the system).

\*System models are given. Scenarios are in side of system models section. Scenarios are to tell us the details of functional requirements. Use case models (definitions & diagrams), object model (UML diagram ),dynamic model(functiions of the sequence diagrams) , user interface view (mockup) and Project Schudule (Gantchart) are parts of system models.

section.

\*Finally, RAD consist of two parts. These ara glossary (dictionary) and references (digital sources).

# Current System

If the new system will replace an existing system, this section describes the functionality and the problems of the current system. Otherwise, this section describes how the tasks supported by the new system are accomplished now.

# Proposed System

Documents the requirements elicitation and the analysis model of the new system

## Overview

Presents a functional overview of the system.

## Functional Requirements

Describes the high-level functionality of the system.

## Nonfunctional Requirements

Describes user-level requirements that are not directly related to functionality. This includes usability, reliability, performance, supportability, implementation, interface, operational, packaging, and legal requirements.

### Usability

### Reliability

### Performance

### Supportability

### Implementation

### Interface

### Packaging

### Legal

## System Models

Describes the scenarios, use cases, object model, and dynamic models for the system. This section contains the complete functional specification, including mock-ups illustrating the user interface of the system and navigational paths representing the sequence of screens.

### Scenarios

A scenario is an instance of a use case.

### Use case model

A use case is a generalization of a number of scenarios. Therefore, the number of scenarios must be equal to or greater than the number of use cases.

### Object model

The analysis object model, depicted with UML class diagrams, includes classes, attributes, and operations. The analysis object model is a visual dictionary of the main concepts visible to the user.

### Dynamic model

The dynamic model is depicted with sequence diagrams and with state machines. Sequence diagrams represent the interactions among a set of objects during a single use case. State machines represent the behavior of a single object (or a group of very tightly coupled objects). The dynamic model serves to assign responsibilities to individual classes and, in the process, to identify new classes, associations, and attributes to be added to the analysis object model.

When working with either the analysis object model or the dynamic model, it is essential to remember that these models **represent user-level concepts, not actual software classes or components.**

### User interface—navigational paths and screen mock-ups

## Project Schedule

Prepare Gannt Chart, and add it to this section.

# Glossary

To establish a clear terminology, developers **identify the participating objects** for each use case. Developers should **identify, name, and describe them** unambiguously and collate them into a glossary.

# References

This subsection should:

* Provide a complete list of all documents referenced elsewhere in the RAD, or in a separate, specified document.
* Identify each document by title, report number - if applicable - date, and publishing organization.
* Specify the sources from which the references can be obtained.

The following is an example of listing a book in this section. Check the text to see how it is cross referenced (The whole document is based on [1]).

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.