Middle East Technical University Department of Computer Engineering



Software Requirements Specification

Prepared by {Takla Design Group} for the project {Trade Center} Mehmet Emin KÜÇÜK-1819457 Süleyman ARIKAN-1818947 Osman Erdem TÜLÜ-1819614 Efe ERDİL-1819309

CENG 491 Senior Design Project I Fall 2015-2016

Table of Contents

1. Introduction	3
1.1 Problem Definition	3
1.2 System Overview	4
1.3 Definitions, acronyms, and abbreviations	5
1.4 Assumptions and dependencies	6
2. Overall description	6
2.1 Product functions	6
2.1.1 Use-case model survey	7
2.1.2 Actor survey	17
2.2 Interfaces	17
2.2.1 User Interfaces	17
2.2.2 Hardware Interfaces	23
2.2.3 Software Interfaces	23
2.2.4 Communications Interfaces	23
2.3 Constraints	23
3. Specific requirements	23
3.1 Functional Requirements	24
3.1.1 Sign Up	24
3.1.2 Login	26
3.1.3 Log Out	27
3.1.4 Search	27
3.1.5 Advanced Search	
3.1.6 Edit Profile	29
3.1.7 Add Item to Store	
3.1.8 Edit Item	31
3.1.9 Remove Item	
3.1.10 Read Messages	34
3.1.11 Send Message	35
3.1.12 Follow User	35
3.1.13 Add Interest/Need	

3.1.14 DeleteProfile	
3.1.15 Other Functions	
3.2 Nonfunctional Requirements	
3.2.1 Usability	
3.2.2 Reliability	
3.2.3 Performance	
3.2.4 Supportability	
4. Data Model and Description	
5. Figures Table	41
6. References	

1. Introduction

The purpose of this document is to present a detailed description of the Trade Center Project. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

1.1 Problem Definition

Our aim in this project is developing a mobile application for Android platform for selling, exchanging, and sharing items. The goal of this project is solving two main problems.

One of these problems is cash money's position in trading. We want to add exchange and grant options into the concept of trading and annihilate the obligation of using cash for trading. With these options people with mutual wants will be able to exchange or give away their wares. The other problem in trading is namely user awareness in goods for trade. Many items that a person may want are being sold or thrown away behind that person's back. An item that is seen as redundant for a person can be very useful for another person. Our approach to solve this problem is creating an environment where people will always be aware of the goods for trade that they might be interested in.

1.2 System Overview

The basic system architecture diagram is shown in Figure 1.



Figure 1: System Architecture Diagram

In this project we will design a mobile application for Android platform for selling, exchanging, and sharing items. The main differences of this project are exchanging goods without money and sending notification to users according to their selections and positions.

Properties of the application

-Users will be able to login with either their Facebook accounts or e-mails.

-Users will also be able to use the application as guests but with limitations.

-Each user will have their Store, Interests/Needs tabs.

-Users will fill their Store with items they want to sell, exchange or give cost-free. They will be able to choose one or more of these sharing options for each item in their Store. Every item will be under a selected category and have an address.

-In their Interests/Needs tab, users will select categories which they are interested in. If an item has an address which is near to the user (this will be determined by using GPS) who has selected the item's category as an interest, a notification will be send to that user which briefly tells that an item which they can be interested in is available for buying/exchanging/getting for free near him/her.

-Users will be able to communicate by using either the messenger system which the application will have, or their contact information.

TERM	DEFINITON
Database	Collection of all the information monitored by this system.
DBMS	A software package/system to facilitate the creation and maintenance of a computerized database.
User	Any person or company logged on the system.
Android	A mobile device operating system developed by GoogleInc.
SocialMedia	Facebook or twitter.
IEEE	The Institute of Electrical and Electronics Engineers (IEEE) is a professional association headquartered in New YorkCity that is dedicated to advancing technological innovation and excellence.
Software	A document that completely describes all of the functions
Specification(SRS)	must operate.
Class Diagram	A type of static structure diagram in UML that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationship among the classes.

1.3 Definitions, acronyms, and abbreviations

Use Case Diagram	A type of diagram in UML that represents the user's interaction with the system.
Android Studio IDE	Android Studio is the official integrated development environment for developing for the Android platform.

1.4 Assumptions and dependencies

Users of this application are any Android device user that loads this application to their devices. All of the users are in the same class, only one type of user exists. Operating environment is, as just mentioned above, is an Android OS mobile device. An android device that can support basic dependencies of the application is expected for proper user experience. On the other hand, our database server and services can operate on any OS like Windows or Ubuntu that can supply the database server's fundamental dependencies and needs.

2. Overall description

In this part, the system allows users to upload or search any desired items or services. To do this, background information about specific requirements of the system will be provided briefly. In Trade Center the user can keep personal information of items, favorites etc. and then the application will run along and communicate with a database server and web-services running in between each part of the system.

2.1 Product functions

In this section, the main functions that the system will provide will be explained as a higher-level specification. Use case diagrams will be used in order to represent these main functions.

2.1.1 Use-case model survey

In this part we mention brief descriptions of use cases associated with a system, component, or other logical or physical entity.



Figure 2: Registered User Use Case Diagram



Figure 3: Admin Use Case Diagram



Figure 4: Guest Use Case Diagram

2.1.1.1 Use Case: SignUp



Guest

In this case, guest enters his/her e-mail, name, surname, password, address. When the user fills all necessary fields, an activation mail will be sent to user's mail address. User will be able to login and start using the system just after activating his/her account.

2.1.1.2 Use Case:Search



In this case, user is able to search items.



In addition to search guest can search items according to their location, tag and trading options.

2.1.1.4Use Case: Login



Registered User

In this case, user enters his/her email and his/her password and logs in the system.

2.1.1.5Use Case: LogOut



```
Registered User
```

In this case, user touches logout button, and logs out of the system. The user will not use the system until he/she logs in again.

2.1.1.6Use Case:Search



In this case, user is able to search items.

2.1.1.7 Use Case: Advanced Search



Registered User

In addition to search registered user can search items according to their location, tag and trading options.

2.1.1.8 Use Case: ViewProfiles



Registered User

In this case, the user can view his/her own profile, his/her inbox notifications, favorite items, items for sale, favorite sellers.



In this case, user is able to browse the his/her own store.

2.1.1.10 Use Case: ViewInterests/Needs



Registered User

In this case, registered user is able to browse categories according to his/her wishes. User can also see how many tags each category has.

2.1.1.11 Use Case: Add Interests/Needs



Registered User can add new interests/needs by choosing tags.



Registered user can remove interests/needs.

2.1.1.13 Use Case: Enable/Disable Notifications



Registered User

Registered users can enable or disable notifications according to their wish.

2.1.1.14Use Case: EditProfile



In this case, user can change his/her username, email address, profile picture,address and password. If the user changes his/her user name or password, a notification mail will be sent to the user's email address. The changes will apply if the user clicks on the link in the email.



In this case, the user can add as many as items he/she wants. The user can also set the price, set the amount, upload pictures of the item and add details of the item.

2.1.1.16 Use Case: EditItem



Registered User

In this case, the user can edit the items that he/she has already put on the system. The user can add/remove pictures, change the price and details of the item.

2.1.1.17 Use Case: RemoveItem



Registered User

In this case, if the user no longer wishes to sell/exchange the item, he/she can remove the item from the system by touching "Remove the item" button.



In this case, if the user no longer wishes to use the system, he/she can click "Delete Profile" button. After deleting the profile, the user will no longer be able to use the system with the account details that he entered before.

2.1.1.19 Use Case: ReadMessages



The user can read the messages sent from other users and his/her messages sent by him.

2.1.1.20 Use Case: SendMessage



Registered User

The user can send messages to other users for bargaining and for exchange requests



The user can follow other users. He will be able to see the changes about him.

2.1.1.22 Use Case: View Control Panel



Admin can perform different operations by accessing view control panel.

2.1.1.23 Use Case: Login



Admin can login to the system.



Admin can authenticate items that are uploaded by registered users.

2.1.1.25 Use Case: Remove User



Admin can remove the registered users.

2.1.1.26 Use Case: Warn User



Admin can warn the registered users.

2.1.1.27 Use Case: Suspend User



Admin can suspend the registered users' accounts.

2.1.2 Actor survey

Front end users

1. Guest: The data consumers, they only have access to the system frontend.

2. Registered user: Once registered, a visitor is offered extra functionalities to save and load custom occurrence filters.

Back end users

1. Admin: They are responsible for installing, configuring and monitoring the system and managing modules. They are also in charge of managing user accounts and associated roles.

2.2 Interfaces

One system requirement is Internet Access for the application. Also systems should be mobile platforms and have Android 4.0 (or later) operating systems.

2.2.1 User Interfaces

User interfaces are illustrated in following figures.

Trade Center	Q
Email Address	
Password	
Login	
Login with Facebook	
Sign UP!	

Figure 5 : Login Screen

Figure 5 is the initial screen in our application. Users can login with their credentials (they can login with their Facebook account as well) or sign up. If one does not wish to login to the system (guest mode), he/she can still search items by clicking the magnifier located on the top right corner (only simple search is allowed in guest mode).

\equiv	Trade	Center	Q
Oturma Ta Cumhuriya	akımı: #Mob et Cad. 100	pilya Yıl	
	Photo	D	(
Info			`
	Photo	D	
Info			
Upload New Item!			
Stor	e	Interests/Need	s

Figure 6: Store Screen

Figure 6 is only for users. Every user has a store in which they can see their uploaded items. Modification of the item information also takes place in this screen/tab. User can switch to view/edit his/her profile in this tab by clicking the top left icon next to "Trade Center". User can also search items/users by clicking the magnifier, and they can upload new items by "Upload New Item!" button. Switching between two tabs (Store, Interests/Needs) is also possible.

	Trade (Center	Q
	Categor	у	1
Tag	S		
	Categor	у	
Tag	S		
	Choose new Inte	erest/Need	
s	tore	Interests/Net	eds

Figure 7: Interests/Needs Screen

Figure 7 shows the categories/tags user needs/ is interested in. Every category consist tags (sub-categories) within itself. For this example, our user is interested in "Football" and "Basketball" tags in "Sports" category. Categories and tags are decided to be represented with visuals instead of plain text in order to make the interface more simple and elegant. Aside from viewing, users can also modify (add/remove) interests/needs by clicking "Choose new Interest/Need" button. Top left and top right icons are already explained in previous screens.



Figure 8 : Search Screen

Figure 8 does not have a particular tab, since it can be accessed from all other tabs by the user. Here is an example simple search screen, accessed from Interests/Needs tab. When user enters a keyword on Figure 8 and clicks the magnifier icon, results (can be items or people) containing the keyword are returned. If desired, an advanced search for the keyword can be performed by clicking the magnifier next to "Advanced search for 'git'".

	Trade	Center	Q
	User Name/	'Surname	_
Notifications			
P Help			
Contact Us			
🗱 Settings			
Logout			
			-
	Store	Interests/Need	s

Figure 9: Profile Screen

Figure 9 is also accessible from all the other tabs within the application. Users can view/edit their profile by "Settings" button, they can turn on/off the notifications they desire and logout.

2.2.2 Hardware Interfaces

Since our project is based on mobile platforms, a mobile phone with android operating system is a must to use the application Also, an Internet connection, Wi-Fi or 3G, is necessary.

2.2.3 Software Interfaces

The application will work on all smart mobile devices with Android 4.0 (or later) operating systems. There will be a graphical user interface for the user to interact with the application. There will be another interface which will do database interactions at the server side that will be invisible to the user.

2.2.4 Communications Interfaces

The project is a mobile application. For its state that does not require accessing the database, no communications will be required. If database access is required, the application will then need to communicate with a remote server containing the database. For this mentioned remote server, Amazon's server hosting service will be used.

2.3 Constraints

Most important constraint is privacy and security. Users should be accessing only the authenticated data. Other constraints:

-The e-mail should consist of at most 25 characters.

-The e-mail should be unique among the e-mails in the database.

-The password should consist of at most 15 characters.

-The user should fill the required fields during the registration.

3. Specific requirements

In this chapter and its sections, the requirements are explained.

3.1 Functional Requirements

This subsection is a description of each major software function, along with data flow and requirements of the function.

3.1.1 Sign Up

Use Cases	Sign Up
Reference	Section 2.1.1.1
Actor	Guest
Trigger	He/she touches the "Register"button.
Precondition	No precondition.
PostCondition	He/she has signed up.
BasicPath	 He/she presses "Register" button. The system directs him/her to register page to fill in the blanks with personal information. He/she enters first and last name. He/she enters email contact address. He/she enters password.
	 6. He/she confirm password by reentering. 7. He/she touches the submit butten to
	finish registration part.

BasicPath	 8. He/she confirm password by reentering. 9. He/she touches the submit button to finish registration part. 10. The system checks e-mail contact address that someone already has. 11. The system checks unfilled fields. 12. The system checks the e-mail format and validates the address. 13. The system sends him/her a mail which contains a verification code. 14. He/she enters verification code. 15. The system directs him/her to main.
ExceptionPath	 If the system finds blank field(s) in register form then he/she is warned and requested to fill them. If the system does not match password with reentering one then he/she is warned and request to fix them. If the system finds duplicate e-mail then he/she is warned and request to enter different e-mail. If the system finds wrong email format or invalid email then he/she is warned and request to enter

3.1.2 Login

UseCases	LogIn
Reference	Section2.1.1.4
Actor	Registered User, Admin
Trigger	The user touches the "Login" button.
Precondition	The user has already signed up.
PostCondition	The user has logged in.
BasicPath	 The user enter credentials (e-mail and password) The user touches "Login" button. The system checks that e-mail and password are valid. The system directs user to main page.
ExceptionPath	 If the system does not find e-mail in database then user is warned and request to enter correct e-mail. If the system finds user's password wrong then user is warned and request to enter the correct password.

3.1.3 Log Out

UseCases	Log Out
Reference	Section2.1.1.5
Actor	Registered User, Admin
Trigger	The user touches the "Log Out" button.
Precondition	The user has already signed in.
PostCondition	The user has signed out.
BasicPath	 The user presses "Log Out" button. The system directs user to main page.
ExceptionPath	None.

3.1.4 Search

UseCases	Search
Reference	Section2.1.1.2/2.1.16
Actor	User, Guest
Trigger	He/she write name of item, category or tag in search box.
Precondition	No precondition.

PostCondition	He/she directs to search results page.
	 He/she touches search box to write information about item. The system directs him/her to search result page.
BasicPath	
ExceptionPath	None.

3.1.5 Advanced Search

UseCases	Advanced Search
Reference	Section2.1.1.3/2.1.17
Actor	Registered User, Guest
Trigger	He/she touches the advanced search button
Precondition	No precondition.
PostCondition	He/she directs to advanced search results page.

BasicPath	 He/she touches advanced search button. The system directs him/her to advanced search page. He/she enters information in at least one of these boxes. -place to search in -tag -category -item name He/she directs to advanced search results page.
ExceptionPath	None.

3.1.6 Edit Profile

UseCases	EditProfile
Reference	Section2.1.1.14
Actor	Registered User
Trigger	The user touches the "Account settings" button.
Precondition	The user has already signed in.
PostCondition	The user changes personal information.

BasicPath	 The user touches the profile icon button at any page. He/she touches the "Settings" button on profile page. User changes any personal information they want.
ExceptionPath	None.

3.1.7 Add Item to Store

UseCases	Add Item(s)
Reference	Section2.1.1.15
Actor	Registered User
Trigger	The user touches the "Add new Item" button
Precondition	The user has already signed in.
PostCondition	The item is added to his/her store.
BasicPath	 He/she touches the "Add new Item" button in his/her Store tab. The system directs him/her to Upload Item page. The user upload a photo of the item he/she want to upload to his/her Store

BasicPath	 He/she touches the "Add new Item" button in his/her Store tab. The system directs him/her to Upload Item page. The user upload a photo of the item he/she want to upload to his/her Store The user enters information (address, tag, selling option) of the item he/she want to upload to his/her Store The item that the user want to upload to his/her Store goes to an administrator for authentication An administrator verify the item The item is uploaded to the user's Store
ExceptionPath	Administrator rejects the item and notifies the user why the item he/she wanted to upload to his/her Store is rejected.

3.1.8 Edit Item

UseCases	EditItem
Reference	Section2.1.1.16
Actor	Registered User
Trigger	The user touches the "Edit" button which is next to the item on his/her Store.

Precondition	The user has already signed in and added one or more item to his/her Store.
PostCondition	If the system approve alters which he/she did, item page profile will change.
BasicPath	 The user touches the "Edit" which is next to the item on his/her Store page. The system directs him/her to edit item wizard as same add item wizard. He/she adds or delete photo. He/she changes title, category, tag, address, selling option of the item. He/she touches the "Confirm Edit" button. The edited item goes to an administrator for an authentication. Administrator verifies the edited item.
ExceptionPath	If the administrator finds the item incomplete or inappropriate for publishing, the system notifies the user about why it is rejected and the changes are discarded.

3.1.9 Remove Item

UseCases	RemoveItem
Reference	Section2.1.1.17
Actor	Registered User
Trigger	The user touches the "Delete" button next to "Edit" button on his/her Store tab.
Precondition	The user has already signed in and added one or more item to his/her Store.
PostCondition	The item is removed from his/her Store.
BasicPath	 The user touches the "Delete" button. Confirm screen appears. If the user touches "Ok" button, item will be removed from his/her Store. If he/she touches the "Cancel" button, the system directs him/her to Store page.
ExceptionPath	None.

3.1.10 Read Messages

UseCases	ReadMessages
Reference	Section2.1.1.19
Actor	Registered User
Trigger	The user touches the inbox section in messages page.
Precondition	The user has already signed in.
PostCondition	The user reads messages.
BasicPath	 The user touches "Inbox" button Inbox page appears on the screen. The user can see new messages with bold writing. He/she touches the read messages to open and read. He/she touches the unread messages to read. After this process, mail turned greyed and insignificant like read messages. He/she edit all messages by selecting them.
ExceptionPath	None

3.1.11 Send Message

UseCases	SendMessages
Reference	Section2.1.1.20
Actor	User
Trigger	The user touches the message button that appears near another user's name.
Precondition	The user has already signed in.
PostCondition	User sends mail to others.
BasicPath	 The user touches the message button near another user's name .New message page appears on the screen. The user fills message area and the item(s) he/she is interested in. He/she can send this message by touching "Send" button.
ExceptionPath	 If the user doesn't choose an item he/she is interested in, the system requests to choose one.

3.1.12 Follow User

UseCases	Follow User
Reference	Section2.1.1.21
Actor	Registered User

Trigger	The user touches the star on others'
Precondition	The user has already signed in.
PostCondition	The Followers of the followed user is updated with the user who followed him/her
BasicPath	 The user touches the star on others' profile page. The Followers of the followed user is updated with the user who followed him/her The user will be notified with user's acts that are followed by him/her.
ExceptionPath	If the system realizes duplicate user in followed list, the user is warned with warning writing, "The user has already been in your followed list"

3.1.13 Add Interest/Need

UseCases	Add Interest/Need
Reference	Section2.1.1.11
Actor	User
Trigger	The user touches the "Add new Interest /Need" button
Precondition	The user has already signed in.

PostCondition	The chosen category or need is added to his/her Interest/Need.
BasicPath	 He/she touches the "Add new Interest/Need" button in his/her Interests/Needs tab. The system directs him/her to Choose Interest/Need page. The user chooses a category or tag or item to be added to his/her Interests/Needs. New Interest/Need is added to his/her Interests/Needs
ExceptionPath	None.

3.1.14 DeleteProfile

UseCases	Delete Profile
Reference	Section2.1.1.18
Actor	Admin
Trigger	The user touches the "Delete Profile" button on account settings.
Precondition	The user has already signed in.
PostCondition	The user deletes profile from database.

BasicPath	 The user touches the "Delete Profile" button. Confirm screen appears. If the user touches "Ok" button, item profile page will remove. If he/she touches the "Cancel" button the system directs him/hor
	to Item List Page.
ExceptionPath	None.

3.1.15 Other Functions

Brief information about other functions is given in Section 2.1 Use-case model survey.

3.2 Nonfunctional Requirements

For running the system, JDK, JVM or JRE must be installed on the android device. Java will be used as programming language. The Android version of the device on which the application is installed should be 4.0 or higher. A stable internet connection is required at all times.

3.2.1 Usability

The scope of the product is widespread. People from every age should use the application without any effort. An average user is expected to learn the system and all its functionalities completely by 30 minutes. The most time-consuming tasks in the application would include fetching/storing images to database and wide searches. Even these tasks should be performed under 10 seconds.

3.2.2 Reliability

The application will be available unless it is removed from the mobile market. Server will be open constantly, except during maintenance. Maintenance will be done by system admins, the duration will be approximately 5 hours per week. Upon failure, system is allowed to be out of operation for 12 hours.

3.2.3 Performance

10000 users should be able to use the system at the same time. The response time of the system should be 2 seconds at most. Users should not upload images bigger than 5 MB, otherwise, it might cause the system to slow down. Approximately 15 MB of memory is required in the Android device on which the application is be installed. Maximum 150 requests to server side per second is allowed.

3.2.4 Supportability

The system should be maintainable in order to add new features to the newer versions. Also, the application should work on newer versions of the related operation systems (the application itself should be updated then if necessary). Hence, all design aspects should be well-documented and easily understandable.

4. Data Model and Description

UML Class Diagrams will be presented in this section. The purpose is to introduce the objects that the system is required to have a representation for. There are 11 classes namely User, Guest, Registered User, Admin, Control Panel, Store, Interests and Needs, Profile, Item, Tag and Category. Their explanation is given below.

User: User is the super class for all different user types, namely Registered User, Guest and Admin.

Guest: When a user enters the system, he/she does not necessarily have to login to use the system. In this mode, a user can still perform simple and advanced search, and sign up.

Admin: Admin is a user role which is able to perform different operations that are not permitted to regular users. An admin does that by accessing a different UI, Control Panel. Admins can also login to the system.

Registered User: It is the common user type. A registered user has credentials(e-mail, ID, password etc.) and is able to access their profile, store, interests and needs. A registered user

can also login to the system, perform normal and advanced search, follow other users, and logout.

Control Panel: It is the user interface for Admin type users from which Admins can perform different operations, such as removing user, removing item, authenticate item, warn user, and suspend user.

Store: Store is where registered users can view, upload and remove items. They can also update the item's information (photos, address, tag etc.)

Item: Items are objects that a registered user can modify (guitar, laptop, chair etc.) They have attributes such as ID, photo(s), address and a tag.

Tag: It is a sub-category which contains specific items. Every item has a tag (football, jazz music, shirt etc.) Each tag has a name, a symbolic image, and a category.

Category: Every tag (subcategory) belongs to a major category (for example, subcategories such as football, basketball, hockey etc. belong to sports category). Each category has a name and its own symbolic image.

Interests and Needs: Registered users can view/change their interests/needs from here.

Profile: In profile, registered users can change their credentials, enable/disable notifications, delete their profile and send/read messages.



Figure 10: Class Diagram

5. Figures Table

Figure 1: System Architecture Diagram	4
Figure 2: Registered User Use Case Diagram	7
Figure 3: Admin Use Case Diagram	7
Figure 4: Guest Use Case Diagram	8
Figure 5 : Login Screen	18
Figure 6: Store Screen	19
Figure 7: Interests/Needs Screen	20
Figure 8 : Search Screen	21
Figure 9: Profile Screen	22
Figure 10: Class Diagram	41

6. References

[1] IEEE Guide for Software Requirements Specifications," in IEEE Std 830-1984, vol., no., pp.1-26, Feb. 10 1984, doi: 10.1109/IEEESTD.1984.119205, URL: <u>http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=278253&isnumber=6883</u>

[2] Appendix C of Don Widrig, Dean Leffingwell, "Managing Software Requirements: A Unified Approach," Addison-Wesley Professional, Release Date: October 1999, ISBN: 0201615932.