Recommend Laptop for Tech Newbies

Software Requirement Specification

2020.05.13.

Introduction to Software Engineering 41

TEAM 1 (iDecide)

Team Leader	Kyungyeon Park
Team Member	Allan Tazhenov
Team Member	Hanwoul Lee
Team Member	Hogi Min
Team Member	Sara B.Zaki
Team Member	Seungrok Yoon

CONTENTS

1. Introduction 7

1.1. Purpose 7
1.2. Scope 7
1.3. Definitions, Acronyms, and Abbreviation 7
1.4. References 9
1.5. Overview 9

2. Overall Description 9

2.1. Product Perspective 9
2.1.1. System Interfaces 10
2.1.2. User Interfaces 10
2.1.3. Hardware Interfaces 10
2.1.4. Software Interfaces 10
2.1.5. Communications Interfaces 10
2.1.6. Memory Constraints 11
2.1.7. Operations 11
2.1.7.1. System administrator 11
2.1.7.2. User 11
2.2. Product Functions 12
2.2.1. Register 12
2.2.2. Get User Preferences 12
2.2.3. Search 12
2.2.4. Item Details 13
2.2.5. Add to Cart 13
2.2.6. Purchase Page Link 13
2.2.7. Profile 13
2.2.8. Review Analysis 13
2.3. User Characteristics 13
2.3.1. System Administrator 13
2.3.2. User 14
2.3.3. Vendor 14
2.4. Constraints 14
2.5. Assumptions and Dependencies 15

3. Specific Requirements 15

3.1. External Interface Requirements 15

3.1.1. User Interfaces 15
3.1.2. Hardware Interfaces 23
3.1.3. Software Interfaces 23
3.1.4. Communication Interfaces 24
3.2. Functional Requirements 24
3.2.1. Use Case 24
3.2.2. Use Case Diagram 30
3.2.3. Data Dictionary 30
3.2.4. Data Flow Diagram 36
3.3. Performance Requirements 36
3.3.1. Static numerical requirement 36
3.3.2. Dynamic numerical requirement 36
3.4. Logical Database Requirements 37
3.5. Design Constraints 37
3.6. Standards compliance 37
3.7. Software System Characteristics 37
3.7.1. Product Requirements 38
3.7.1.1. Usability Requirements 38
3.7.1.2. Performance Requirements 38
3.7.1.3. Usability Requirements 38
3.7.1.4. Security Requirements 38
3.7.2. Organizational Requirements 38
3.7.2.1. Environmental Requirements 39
3.7.2.2. Operational Requirement 39
3.7.3. External Requirements 39
3.7.3.1. Safety / Security Requirement 39
3.7.3.2. Regulatory Requirement 39
3.8. Organizing the Specific Requirements 39
3.8.1. Context Model 40
3.8.2. Process Model 40
3.8.3. Interaction Model 41
3.8.4. Behavior Model 41
3.8.4.1. Data Flow Diagram 41
3.8.4.2. Sequence Diagram 41
3.9. System Architecture 41
3.10. System Evolution 42
3.10.1. Limitation and Assumption 42
3.10.2. Evolutions of Hardware and Change of User Requirements 42

4. Supporting Information 43

4.1. Software Requirement Specification 43

4.2. Document History 43

LIST OF FIGURES

[Figure 1] Use case diagram 30
[Figure 2] Entity Relationship Diagram..... 35
[Figure 3] Data flow diagram..... 36
[Figure 4] Context model..... 40
[Figure 5] Overall process model..... 40
[Figure 6] Sequence diagram 41
[Figure 7] System architecture of the system..... 42

LIST OF TABLES

[Table 1] Table of acronyms and abbreviations 8
[Table 2] Table of terms and definitions 8
[Table 3] User interface of input processing using touchscreen 15
[Table 4] User interface of get user preference 16
[Table 5] User interface of search 17
[Table 6] User interface of item details 18
[Table 7] User interface of cart 19
[Table 8] User interface of register 20
[Table 9] User interface of profile 21
[Table 10] Hardware interface of applicable device for the system 23
[Table 11] Software interface of firebase real-time database 23
[Table 12] Communication interface of client and host 24
[Table 13] Use case of register 24
[Table 14] Use case of log-in/out 25
[Table 15] Use case of profile 26
[Table 16] Use case of search 27
[Table 17] Use case of item details 27
[Table 18] Use case of cart 28
[Table 19] Use case of purchase page link 28
[Table 20] Use case of review analysis 29
[Table 21] User 30
[Table 22] CPU 31
[Table 23] VGA 31
[Table 24] Storage 31
[Table 25] RAM 31
[Table 26] Panel 32
[Table 27] Laptop 32
[Table 28] Laptop_Storage 33
[Table 29] Laptop_RAM 33
[Table 30] Laptop_Review 33
[Table 31] Cart 33
[Table 32] Search_History 34
[Table 33] Search_Result 34
[Table 34] Vendor_Link 34
[Table 35] Document History 43

1. Introduction

1.1. Purpose

This document is a Software Requirements Specification (SRS) for providing iDecide (Recommend Laptop for Tech Newbies) services. This service is designed and implemented by Team 1 of the Introduction to Software Engineering at Sungkyunkwan University. The requirements for this are summarized, analyzed, and the system is designed and implemented based on the contents described.

In this document, Team 1 is the main reader, and Team 1 designs and implements the functions of the laptop recommendation service according to this specification. Additionally, professors, TAs, and team members in the Introduction to Software Engineering class can be the main readers.

The purpose of this document is to outline and publish the Requirement Specification for a new mobile application for automated recommendations of technological devices to novices. Unlike many others, iDecide gets personal preference from the user and uses their preferences to find the best suitable laptop based on their needs and budget. This app uses a complex algorithm which tailors to the desires and requirements of the user, almost matching a laptop to their unique personality.

1.2. Scope

The iDecide recommendation system is meant to ease the exhausting hours of searching for the perfect laptop and to create a convenient and easy-to-use application for ordinary users, trying to find and buy an appropriate device in a short amount of time. The system is based on a relational database with AI functions. We will have a database server supporting hundreds of computer manufacturers around the world as well as thousands of devices such as laptops by various companies and from a wide variety of vendors. Above all, we hope to provide a comfortable user experience along with the best possible pricing available to the users.

1.3. Definitions, Acronyms, and Abbreviation

The following table explains the acronyms and abbreviations used in this document.

[Table 1] Table of acronyms and abbreviations

Acronyms& Abbreviations	Explanation
RAM	Random Access Memory
HDD	Hard Disk Drive
CPU	Central Processing Unit
SSD	Solid-State Drive
OS	Operating System
GUI	Graphical User Interface
API	Application Programming Interface
UI	User Interface
HTTP	Hypertext Transfer Protocol

The following table defines certain technical terms used in this document.

[Table 2] Table of terms and definitions

Terms	Definitions
User	Someone who uses a system
System administrator	Someone who quantify the keywords included in the reviews for each laptop and manage the system
Back-End	Application part that is not directly accessed by the user, such as the server and database
Front-End	The user interface, also known as the presentation layer of an application
Algorithm	A set of rules or procedures followed by a computer in problem-solving operations
Client (user device)	A user device/user that connected to server
Server	A computer or computer program which manages access to a centralized resource or service in a network
Software	The programs and other operating information used by a computer
Network	For connect devices together so that they can share information. In this system, it usually means internet

1.4. References

- IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications, In IEEEXplore Digital Library
  http://ieeexplore.ieee.org/Xplore/guesthome.jsp
- Team 5. “Software Requirement Specification”. SKKU, Last Modified: Jun. 7, 2019.
  https://github.com/skkuse/2019spring_41class_team5/blob/master/docs/requirement.pdf
- Multimedia Service Team. “Software Requirement Specification of Multimedia Contents-aware Intelligent Information Service System”. Kangwon National University. (2007)

1.5. Overview

The remainder of this Software Requirements Specifications Document includes three chapters and appendixes. The second chapter provides an overall description of the product perspective, including the several interfaces, the system functionality and system interaction with other systems. This chapter also introduces different types of stakeholders and their interaction with the system. Additionally, the chapter also mentions item details, the system constraints, assumptions, and dependencies of the product. The third chapter provides the requirements specification in detailed terms, including a description of the different system interfaces and the software system characteristics. A range of specification techniques are used to specify the requirements more precisely for a variety of users. It also shows a lot of use cases and a data dictionary. The fourth chapter deals mainly with the prioritization of the requirements. It includes a supporting documentation, a timeline of the documentation of this SRS for the application, iDecide. All members contributed equally to the production of this project. We hope that you, the reader, enjoy viewing this document.

2. Overall Description

2.1. Product Perspective

This product is designed for certain people who does not understand technical words like “RAM”, “CPU”, “HDD”, etc. This application will make it comfortable for those people to

select the product that will satisfy all their requirements, that they will specify in the search menu. This application will use database of tech, customers' requirements and purpose of usage. If person doesn't have any certain purpose system will output the product with the best price/quality product for required price.

2.1.1. System Interfaces

The user's cart information and search histories are stored in the user's local area using SQLite, and the information of the laptop which contains quantitative specification and user review keyword, and profile is stored using firebase.

SQLite which used to save database in local operates quickly because the operation of the database is performed through API calls without using protocol or process communication.

Data stored in firebase is stored in JSON format and synchronized in real time through HTTP communication to all connected clients.

2.1.2. User Interfaces

An interface is provided through the screen of the mobile phone, and information can be input through a simple touch of the mobile phone. Depending on the preferences entered by the user, the user should be able to be recommended a laptop.

The administrator can access Laptop database. The administrator is provided with an interface through the basic GUI of firebase, and supports registration, deletion, maintenance, and management of laptop information.

2.1.3. Hardware Interfaces

The system is intended for Android mobile phone. The mobile phone must have least 1GB RAM and 1.0 GHz single processor.

2.1.4. Software Interfaces

The system is intended for Android OS version at least Android 6.0 (API 23) and targeting Android 10 (API 29).

2.1.5. Communications Interfaces

User device and server communicate with HTTP protocol in JSON format.

2.1.6. Memory Constraints

The system should run on mobile devices with least 1GB RAM for primary memory and the system requires at least 512 MB for installation and execution

2.1.7. Operations

2.1.7.1. System administrator

- Analysis review
  - ✓ Key keywords in the review are predefined by the system administrator
  - ✓ Decide which sites to crawl and crawl other users' reviews
  - ✓ Quantification of a specific keyword is stored in a database

2.1.7.2. User

- Login
  - ✓ User can login using google account, etc.
- Register
  - ✓ User can register into the system using google account, etc.
  - ✓ When user register into the system, user enter some information like username, age, gender, etc.
- Get recommendation
  - ✓ User can get several recommend of laptops after answering several system questions
- Cart
  - ✓ User can add device which is the result of recommendation(search) into the cart
  - ✓ User can see item detail when user touch the item
- Purchase Link
  - ✓ User can buy a specific laptop using purchase link
  - ✓ The link sorted by lowest in real time
- Profile

- ✓ User can modify user information
- ✓ User can see the search history
- ✓ User can see item preference history and recommended laptop when user touch the laptop in search history

- Menu operation
  - ✓ All systems accessed by the user are operated through the touch screen of the user device

2.2. Product Functions

2.2.1. Register

After the user downloaded our application, user must register to the system in order to use it. There will be register button in the login page which redirects user to register page. After filling required fields with information, user can register to the system with using register button. User can register using Google account, etc. and enter some information like age, gender, etc. After the user registered to the system, user can login and use the system.

2.2.2. Get User Preferences

After logging in, user can start searching a product. If user is the state right after register, and there is no preference, get user preference. The preference page do not ask the terminology like CPU and memory which is hard for tech newbie, but ask for main purpose, laptop replacement cycle, budget, etc. Also, user can set priorities of these question. After user fills in preferences, application will show the recommended items after the search button is pressed. During get user preference, the system analysis estimated price from now on. This page can be shown right after register, or if user touch ‘Find Laptop’ button.

2.2.3. Search

After get user preferences, user can start searching a product that user desires to find and that fits to user. Using recommendation algorithm which analyzes product specifications in terms of performance to quantify fitness with user (battery time, CPU, weight, etc.), extracts specific keywords from other users' actual reviews to quantify fitness with users (less noise etc.), system can recommend laptops for user.

2.2.4. Item Details

Display the result of recommendation algorithm in two radial graphs. One shows quantify result related to fitness with user of product specification and the other shows quantify result related to fitness with user of other user's actual review. If user touch laptop in the result of search state or in the cart, this fit visualization will be shown. As well as graphs, comments can be added.

2.2.5. Add to Cart

User will have an option to add this object inside a cart where user can check, which laptops user chose, where user will have an option to delete it from cart or open the link to shop. If user touch the laptop in the cart, the basic information about the laptop will be visualized.

2.2.6. Purchase Page Link

This option will be given in search menu and it will be given inside cart menu. User can open this link if user want to buy this item, and 5~10 link will be shown from lowest to highest.

2.2.7. Profile

In this page user will be able to see basic information user entered when user signed up and user can modify the basic information (nickname, age, etc.). Also, user can see the search results for the preferences of the user which user has set so far (search history).

2.2.8. Review Analysis

Reviews on a device are processed by text-mining methods by the database system. The result would be provided to users as a visualized figure which would help intuitive understanding of the device. Keywords that are important in the review are predefined by the system administrator, the site to be crawled is determined, and the reviews of other users are crawled to quantify the evaluation of specific keywords and stored in the database.

2.3. User Characteristics

2.3.1. System Administrator

System administrator is limited to those who has sufficient knowledge of the system and who has a general understanding of the system. It is assumed that system administrator has sufficient

capabilities to deal with system problems. Also, it is assumed that system administrator has majored in computer science or similar studies, has completed training to become a network administrator or a system administrator, or has equivalent qualifications. In addition, it is assumed that the system administrator has the ability to reflect the define new reference value for the purchase of a laptop into the system. Also, by analyzing the reviews, system administrator should be able to quantify the keywords included in the reviews for each laptop.

2.3.2. User

The user generally referred to in this document is customer. User is assumed that has the ability to read Korean in a smooth way and understand its meaning and wants to buy a laptop or are interested in purchasing it but does not have any professional knowledge of the laptop specifications. It is also assumed that user can read and understand basic English, and that user is equipped with education and literacy to the extent that there is few or no difficulty in installing and using applications on their smartphone. Generally, it is assumed that Korean nationals aged 13 to 65 are users.

2.3.3. Vendor

Vendor is assumed that people who sells various laptops on a personal sales site or open markets and has knowledge to upload and sell things online. It is assumed that vendor is selling a laptop online, and generally assume that the Korean person who selling laptops online is vendor of this system.

2.4. Constraints

The system will be designed and implemented based on the contents mentioned in this document. Other details are designed and implemented by selecting the direction preferred by the developer, but the following items are observed.

- Use the technology that has already been widely proven.
- Laptop search speed should not exceed 4 seconds.
- Avoid using technology or software that requires a separate license or pays for royalty. (Exclude this provision if this is the only technology or software that the system must require.)
- Decide in the direction of seeking improvement of overall system performance.

- Decide in a more user-friendly and convenient direction
- Use open source software whenever possible
- Consider the system cost and maintenance cost
- Consider future scalability and availability of the system
- Optimize the source code to prevent waste of system resources
- Consider future maintenance and add sufficient comments when writing the source code
- Develop with Windows 10 environment and Android Studio whose build tools version is 29.0.3
- Develop with minimum Android version 6.0 (API 23) and target Android version 29
- Emulate the system using Android version 10 (API 29)

2.5. Assumptions and Dependencies

All systems in this document are written on the assumption that they are designed and implemented based on Android devices and open source. Therefore, all contents are written based on the Android operating system with minimum API version 23 and may not be applied to other operating systems or versions.

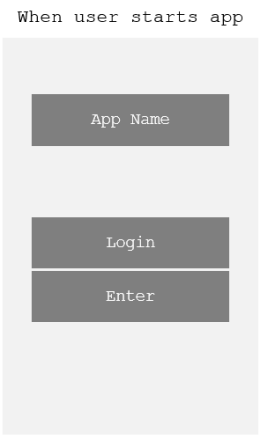
3. Specific Requirements

3.1. External Interface Requirements

3.1.1. User Interfaces

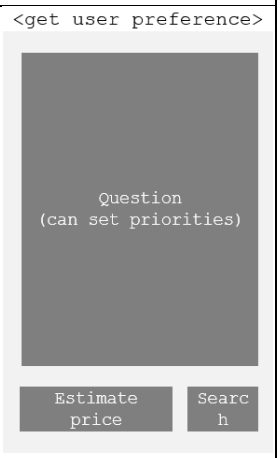
[Table 3] User interface of input processing using touchscreen

Name	Basic User Interaction Using Touchscreen
Purpose/Description	Users transmit their instructions through a touchscreen of the device
Input source/ Output destination	User/ user device equipped with Android OS
Range/ Accuracy/ Margin of error	Range according to the number of buttons on the screen/ Accuracy according to the accuracy of touch from users/ Margin of error of touch sensitiveness
Unit	A click

Name	Basic User Interaction Using Touchscreen
Time/ Velocity	Asynchronous user input/ Instant execution of a user instruction
Relationship with other input/outputs	After receiving all the inputs, the user device transmits the input data to the server for processing the input data and request desired output data
Format and configuration of screen	1. An activity screen connected to an XML file, mainly composed of TextViews and ImageVIews 2.A Textview and an ImageView provide a basis for choosing the following input, and several Buttons are ready to receive the inputs from users 3. Users are to click a desired button to interact with the system 
Format and configuration of window	N/A
Data type	Int type value of a button code
Instruction type	Instruction mapping according to the value of a button code
Exit message	N/A

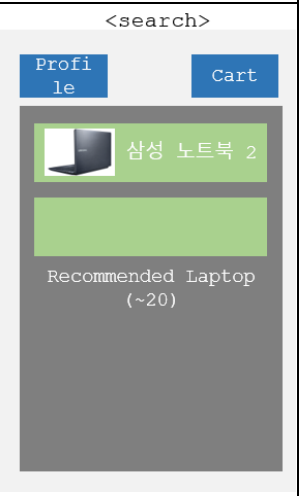
[Table 4] User interface of get user preference

Name	Search Interface1 – Get User Preference
Purpose/Description	Users are asked numerous questions on the usage of a potential device Users are to answer the questions by choosing a button representing the answer regarding their preferences on laptops When users answer all the questions, the data is transmitted to the server from the user device
Input source/ Output destination	Host server / Client Client / Host server
Range/ Accuracy/ Margin of error	N/A
Unit	Screen
Time/ Velocity	After all the input data were received to the user device/ Communication time between the server and the user device

Name	Search Interface1 – Get User Preference
Relationship with other input/outputs	Each preference affects the result of search
Format and configuration of screen	1. An activity screen connected to an XML file, mainly composed of TextViews and ImageViews 2. Several TextViews regarding user preferences are presented on the screen and there might be some ImgaweViews that would help understanding the meaning of preferences 3. When the user clicks the button 'Search', all the preference data is transmitted to the server for generating results 
Format and configuration of window	Activity with XML document
Data type	Image, Text, Widget
Instruction type	N/A
Exit message	N/A

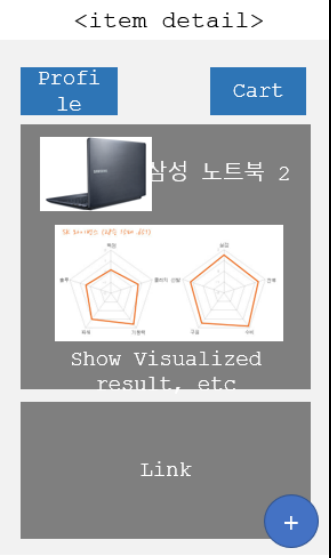
[Table 5] User interface of search

Name	Search Interface 2 – Search
Purpose/Description	Present a list of laptops filtered and reorganized on the screen so that users can see the results of their search Each item shows its estimated price including all the options that users considered necessary Items are basically sorted by lowest price
Input source/ Output destination	Host server / Client
Range/ Accuracy/ Margin of error	Up to 20 laptops that fulfill the conditions with 3 vendors for each model/ Depends on the recommendation algorithm/ Margin of error set by the recommendation algorithm
Unit	Screen
Time/ Velocity	After all the input data were received to the user device/ Communication

Name	Search Interface 2 – Search
	time between the server and the user device
Relationship with other input/outputs	Since the list is affected by the inputs from users, the list should be able to change and show the results dynamically
Format and configuration of screen	1. An activity screen connected to an XML file, mainly composed of TextViews and ImageVIews shows Textview and an ImageView providing searching results 2.The button 'Cart' enables users to get in to the cart page 3.Widget 'More results' that users can see additional results more than 20 if they want is located at the bottom of the screen and the system prepares another 20 additional results in advance for prompt response to the request 
Format and configuration of window	Activity with XML document
Data type	Image, Text, Widget
Instruction type	N/A
Exit message	"Find other laptops!"

[Table 6] User interface of item details

Name	Search Interface 3 – Item details
Purpose/Description	From a presented list of laptops that fulfill the conditions, users can click a specific laptop Image and enter into the item detail page of that device
Input source/ Output destination	Host server / Client
Range/ Accuracy/ Margin of error	N/A
Unit	Screen
Time/ Velocity	Asynchronous user input/ Communication time between the server and the

Name	Search Interface 3 – Item details
	user device
Relationship with other input/outputs	N/A
Format and configuration of screen	1.Linear layout that shows the detailed information of a laptop that fit the conditions with the final price with additional options (such as extra RAM, SSD storage, etc.) with images 2.A figure that can enables users to check the review analysis regarding the item and the vendor (such as credibility of the vendor, duration of delivery, etc.) 3.Linear layout using widgets including Buttons saving a particular device to the cart for comparison among devices chosen by users 4.When a user clicks the button ‘Link’, the user is directed to the vendor’s page for the purchase of the item 
Format and configuration of window	There is a button widget with the symbol ‘+’ for adding the item to the cart for later comparison among devices. After adding an item to cart, the message “Saved in the cart!” pops up.
Data type	Text, Image, Widget
Instruction type	N/A
Exit message	N/A

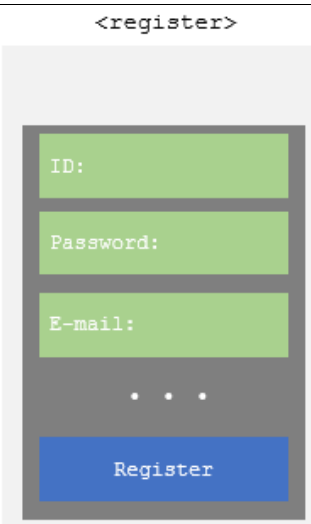
[Table 7] User interface of cart

Name	Customization Interface 1 - Cart
Purpose/Description	After users click a button for saving the item detail to the cart, the user saves that information into its own database using SQLite. When the user clicks the button ‘Cart’, the user is directed to the cart page where s/he can check the items in the cart Users can check the items in the cart and compare items at once in the cart

Name	Customization Interface 1 - Cart
	User device show a visualization of the fitness of devices with user in the cart to users
Input source/ Output destination	User/ User device
Range/ Accuracy/ Margin of error	N/A
Unit	Table
Time/ Velocity	On entering the cart menu/ Depends on the processing time of the user device
Relationship with other input/outputs	N/A
Format and configuration of screen	Linear layout, representing a list of items in the cart Linear layout, representing the comparison list of devices in the cart, reordered according to the input from users
Format and configuration of window	1. An activity screen connected to an XML file in a LinearLayout format, mainly composed of TextViews and ImageVIews shows Textview and an ImageView providing a list of items put into the cart 2. Similar to 'Search' screen, users can click an item and be directed into Item details page 
Data type	Query, Image

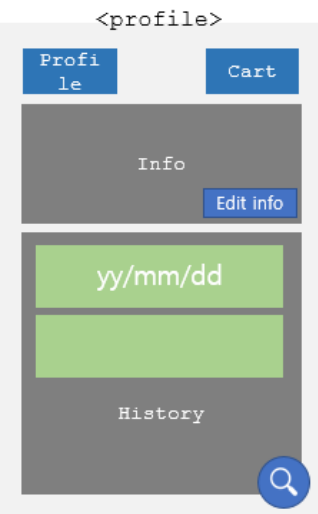
[Table 8] User interface of register

Name	Customization Interface 3 - Register
Purpose/Description	In order to use the service of the system, users have to register to the system , followed by Log-in process. During registration, they are asked to fill out the registration form.
Input source/	Client/ Server

Name	Customization Interface 3 - Register
Output destination	
Range/ Accuracy/ Margin of error	N/A
Unit	Screen
Time/ Velocity	N/A
Relationship with other input/outputs	N/A
Format and configuration of screen	1.Several empty slots for filling out necessary information of users in a vertical direction 2. At the bottom of the screen, there is a button 'Register' which returns all the information given by a user to the server for saving it 3.After successful registration process, the users are directed to the first page for log-in 
Format and configuration of window	N/A
Data type	Query
Instruction type	Instruction mapped to the button
Exit message	"Register succeed!"

[Table 9] User interface of profile

Name	Customization Interface 4 - Profile
Purpose/Description	After log-in, users can enter the 'Profile' page by clicking a button 'profile' in

Name	Customization Interface 4 - Profile
	order to edit their information and find their previous search histories
Input source/ Output destination	User/ Servers
Range/ Accuracy/ Margin of error	N/A
Unit	Page
Time/ Velocity	N/A
Relationship with other input/outputs	N/A
Format and configuration of screen	1. In 'Info' layout, information of a user is presented and the user can edit it by clicking the button 'Edit info'. 2. Below is the history list of previous search. The user can click each history and access to previous items directly. 3. The Widget with the symbol '?' enables the user to search laptops. After clicking it, the user is directed to 'Get user preference' page. 
Format and configuration of window	N/A
Data type	Query
Instruction type	Instruction mapped according to the value of a button code
Exit message	"Profile Was Updated!"/ "Find new laptops!"

3.1.2. Hardware Interfaces

[Table 10] Hardware interface of applicable device for the system

Name	Applicable device for the system
Purpose/Description	Enable users to take advantage of the service provided by the system/Android OS Enabled Smart Phone. (At least Android 6.0)

3.1.3. Software Interfaces

[Table 11] Software interface of firebase real-time database

Name	Firebase Real-time Database
Purpose/Description	Query input/output for managing multimedia/meta data
Input source/ Output destination	Host server/ User, User/Host server
Range/ Accuracy/ Margin of error	Depends on the performance of the Firebase
Unit	Query
Time/ Velocity	Instant reaction
Relationship with other input/outputs	Related to all inputs/outputs from server
Format and configuration of screen	N/A
Format and configuration of window	N/A
Data type	Query
Instruction type	Query statement
Exit message	N/A

3.1.4. Communication Interfaces

[Table 12] Communication interface of client and host

Name	Client and Host
Purpose/Description	Each client requests the connection to the host, requesting list of results of laptop search Host provides a laptop list and laptop information to the client
Input source/ Output destination	Client/Host server
Unit	packet
Time/ Velocity	At least 10Mbps
Relationship with other input/outputs	Related to all inputs/outputs from server
Format and configuration of screen	N/A
Format and configuration of window	N/A
Data type	Query
Instruction type	Query statement
Exit message	N/A

3.2. Functional Requirements

3.2.1. Use Case

[Table 13] Use case of register

Use case name	Register
Actor	Unregistered user
Description	It is a process when an unregistered user tries to register in our system as a member in order to take advantage of all the available functions of the system
Normal Course	1. All the users encounter a log-in page after executing the application 2. After finding out that the service is only provided to the user after registration, an unregistered user clicks the register button in the log-in

Use case name	Register
	page 3. The user is redirected to register page 4. The registration process will be provided using Google API, etc. 5. The user is required to fill in some additional information according to the registration form. The required information includes 1) Email-address (ID) 2) Nickname (doesn't have to be unique) 3) Age 4) Gender 6. The system sends a verification code to the given email-address in order to verify whether the email address is correct and to prepare the situation of finding password 7. After filling out the form, the user is registered and returns to the login page after clicking the register button at the end of the form
Precondition	The user is not registered to the system yet The user enters correct information. The same email address should not be overlapped with that of other users In case of incorrect inputs, the system validates the form of email address and password
Post Condition	The password should be encrypted and saved to the user management database for security
Assumptions	N/A

[Table 14] Use case of log-in/out

Use case name	Log-in/out
Actor	Registered user
Description	Log-in is a process when a registered user of the system tries to get into the system for using the service Log-out is a process when a user who has logged-in tries to get out of the system
Normal Course	<Log-in> 1. A user who has already registered to the system as a member wants to use the service from the system 2. The user enters the email address and password that were the user set for registration

Use case name	Log-in/out
	3. If the information is correct, the system allows the user to get into the system and the user is now able to take advantage of all the service provided by the system <Log-out> 1. If the user wants to get out of the system, the user clicks the button 'Log-out' 2. If the user closed the application without logging-out, the system arbitrarily closes the session for that user
Precondition	<Log-in> The user should already be registered to the system <Log-out> The user should be in a logged-in status
Post Condition	The user should be connected to network
Assumptions	N/A

[Table 15] Use case of profile

Use case name	Profile
Actor	Registered user
Description	It is the page where the user is able to check and modify the user information that the user entered when registration User can also see the search history and do another search
Normal Course	1. A user clicks the button 'Profile' which direct him to the profile page 2. The user can check the information and modify the personal information such as nickname, age, gender, which was required when registration 3. After modifying the information, the user clicks the button 'Finish' to return to the previous page
Precondition	The user should be in a logged-in status and connected to network
Post Condition	After exit from profile page, the information should be delivered to the server and updated If user enter one of the search histories, system must show search results for preferences that the user set last time Although a new laptop is added, search history shows the past search results, not the results that reflect the new laptop

Use case name	Profile
Assumptions	N/A

[Table 16] Use case of search

Use case name	Search
Actor	Registered user
Description	A user can start searching a product that user desires to find and that fits user After searching process, the user would have a list of recommended laptops
Normal Course	1. After logging- in, the user clicks the button ‘Search’ 2. The user is asked to answer several questions related to the criteria on choosing laptops, such as ‘What kind of program do you use the most frequently?’ 3. The user is required to click an answer for the question 4. This process continues until the final question 5. User can set priorities between these question 6. Based on the answers, the system prepares a list of recommended laptops and shows users the list 7. The list shows up to 20 laptops arranged by the lowest cost 8. The user can click the button ‘More results’ for more recommendation
Precondition	The user should be in a logged-in status and connected to network
Post Condition	The search history should be saved to local database (SQLite)
Assumptions	N/A

[Table 17] Use case of item details

Use case name	Item details
Actor	Registered user
Description	This is a page where a user can see detailed information of a particular laptop
Normal Course	1. The user clicks an item image in the list of laptops for more details 2. The user is directed to the item detail page of a particular laptop 3. There the user can check detailed information including the overall rating of the item and reviews from other users on that item 4. Quantitative specification and review analysis visualize in radial graph 5. Quantitative specification and review analysis will be an indicator of

Use case name	Item details
	how appropriate users and laptops are
Precondition	The user should be in a logged-in status, finished searching process There are one or more items that attracted the user
Post Condition	N/A
Assumptions	N/A

[Table 18] Use case of cart

Use case name	Add to cart
Actor	Registered user
Description	This is a process when a user wants to set a particular laptop as a potential item for purchase
Normal Course	There is a button 'Add to cart' at the right side of each item row of the list, generated by searching process The user finds a particular laptop as attractive and clicks the button for saving the information of the item for later purchase
Precondition	The user should be in a logged-in status and connected to network The user finished searching process
Post Condition	The laptop detail should be saved to local database (SQLite)
Assumptions	N/A

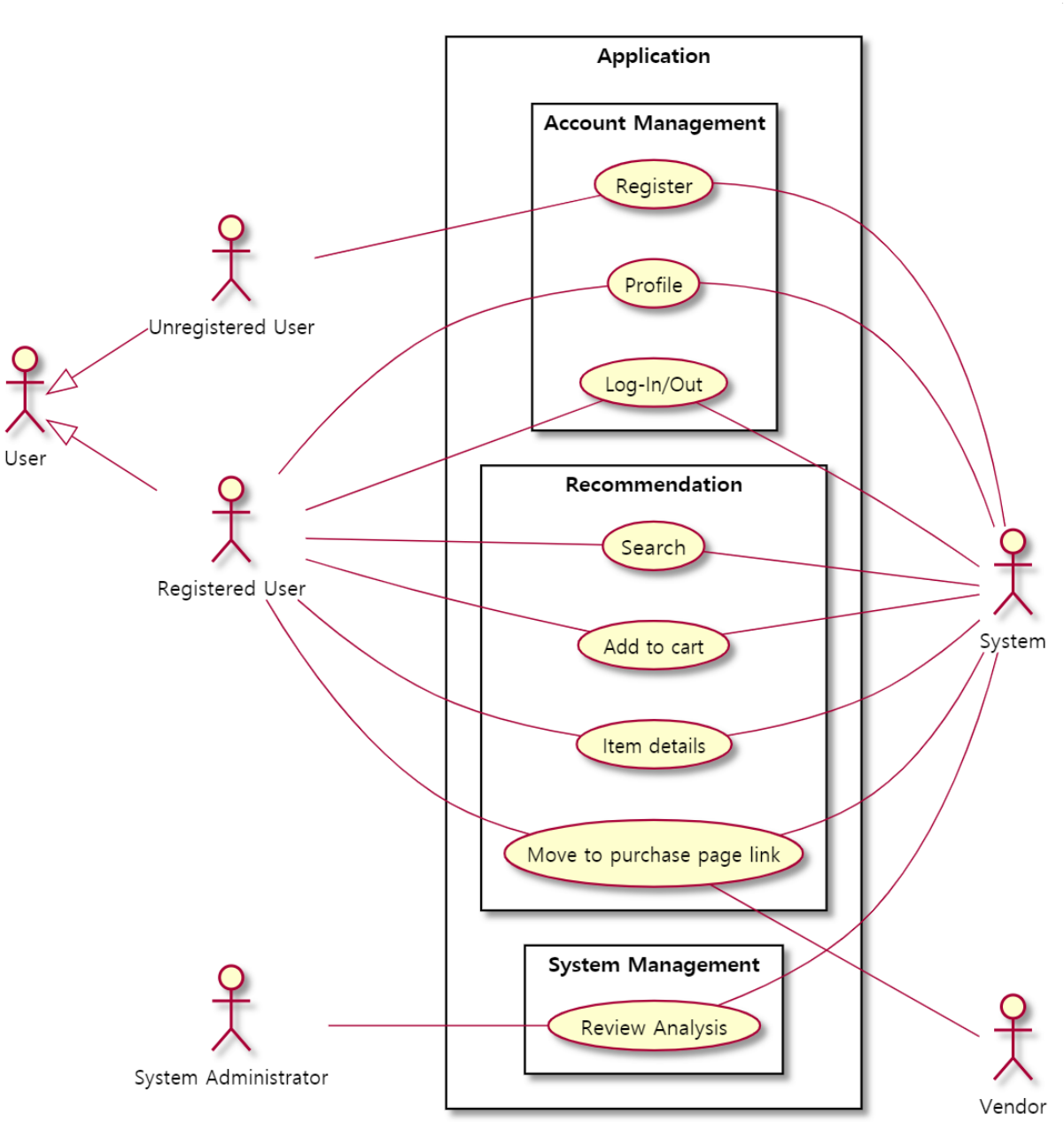
[Table 19] Use case of purchase page link

Use case name	Move to purchase page link
Actor	Registered user
Description	This option will be given in search menu and it will be given inside cart menu. User can open this link if user want to buy this item.
Normal Course	1. The user clicks a link of a particular item 2. The user is directed to the vendor's site for the purchase of the item
Precondition	The user should be in a logged-in status and connected to network The user should finish the searching process
Post Condition	N/A
Assumptions	The user decided to purchase the item

[Table 20] Use case of review analysis

Use case name	Review analysis
Actor	System administrator
Description	This is a visual description of reviews on a device. Reviews on a particular device are processed by text-mining methods by the database system. The result would be provided to users as a visualized figure which would help intuitive understanding of the device
Normal Course	1. The product management system collects massive data(review) on laptops 2. The data is preprocessed and saved in the database 3. While processing the data of laptops, the reviews are also processed and saved in the database 4. When the user clicks a particular item, the server sends the data to the user, judging how suitable it is with the user, and shows the data as instructed in the application
Precondition	The user should be in a logged-in status The user should be in an item detail page
Post Condition	N/A
Assumptions	The number of reviews exceeds the minimum number of reviews required for analysis

3.2.2. Use Case Diagram



[Figure 1] Use case diagram

3.2.3. Data Dictionary

[Table 21] User

Field	Key	Constraint	Description
id	PK	Not Null	User id(email)
nickname		Not Null	User nickname

Field	Key	Constraint	Description
age			User’s age
gender			User’s gender

[Table 22] CPU

Field	Key	Constraint	Description
name	PK	Not Null	CPU name
manufacturer		Not Null	Manufacturer of CPU
chipset		Not Null	CPU chipset
core		Not Null	CPU core number
clock_rate		Not Null	CPU clock rate

[Table 23] VGA

Field	Key	Constraint	Description
name	PK	Not Null	VGA name
manufacturer			Manufacturer of VGA
chipset		Not Null	CPU chipset
vram_capacity			VGA Ram capacity

[Table 24] Storage

Field	Key	Constraint	Description
name	PK	Not Null	Storage name
manufacturer			Manufacturer of Storage
type		Not Null	Storage type (HDD/SDD)
capacity		Not Null	Storage capacity

[Table 25] RAM

Field	Key	Constraint	Description
name	PK	Not Null	Ram name
manufacturer			Manufacturer of RAM
type		Not Null	RAM type (DDR3/DDR4)
capacity		Not Null	Ram capacity
bandwidth			Ram bandwidth
clock_rate			Ram clock rate(MHz)

[Table 26] Panel

Field	Key	Constraint	Description
name	PK	Not Null	Panel name
manufacturer			Manufacturer of panel
type		Not Null	Panel type (IPS etc.)
size		Not Null	Panel size (inch)
resolution		Not Null	Panel resolution (FHD etc.)
aspect_ratio		Not Null	Aspect ratio (16:9 etc.)
screen_brightness			Screen brightness (nit)
color_gamut			Color gamut (%)
frame_rate			Frame rate (Hz)

[Table 27] Laptop

Field	Key	Constraint	Description
product_code	PK	Not Null	Product code
manufacturer		Not Null	Manufacturer of laptop
brand			Laptop brand
device_type			Device type (2in1 etc.)
display_name	FK	Not Null	Ref. Panel
cpu_name	FK	Not Null	Ref. CPU

Field	Key	Constraint	Description
vga_name	FK	Not Null	Ref. VGA
battery_capacity			Battery capacity (Wh)
operating_system			Operating system
weight			Weight of laptop (kg)

[Table 28] Laptop_Storage

Field	Key	Constraint	Description
laptop_product_code	PK/FK	Not Null	Ref. Laptop
storage_name	PK/FK	Not Null	Ref. Storage

[Table 29] Laptop_RAM

Field	Key	Constraint	Description
laptop_product_code	PK/FK	Not Null	Ref. Laptop
ram_name	PK/FK	Not Null	Ref. RAM

[Table 30] Laptop_Review

Field	Key	Constraint	Description
user_id	PK/FK	Not Null	Ref. User
laptop_product_cod	PK/FK	Not Null	Ref. Laptop
review		Not Null	User’s review/comment

[Table 31] Cart

Field	Key	Constraint	Description
user_id	PK/FK	Not Null	Ref. User
laptop_product_cod	PK/FK	Not Null	Ref. Laptop
number		Not Null	Number of items

[Table 32] Search_History

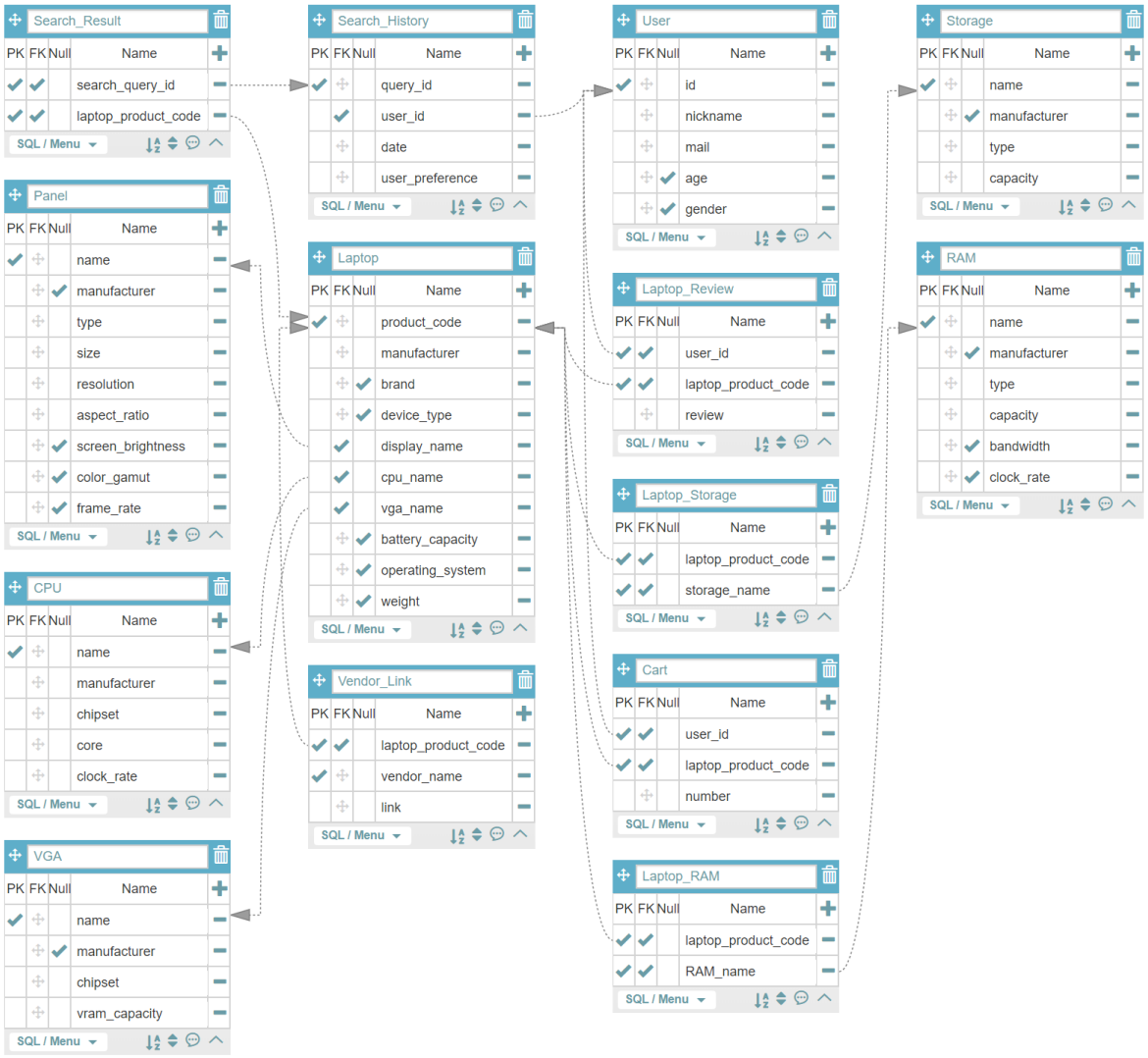
Field	Key	Constraint	Description
query_id	PK	Not Null	Search query id (key)
user_id	FK	Not Null	Ref. User
date		Not Null	Search date (timestamp)
user_preference		Not Null	Search options

[Table 33] Search_Result

Field	Key	Constraint	Description
search_query_id	PK/FK	Not Null	Ref. Seach_History
laptop_product_code	PK/FK	Not Null	Ref. Laptop

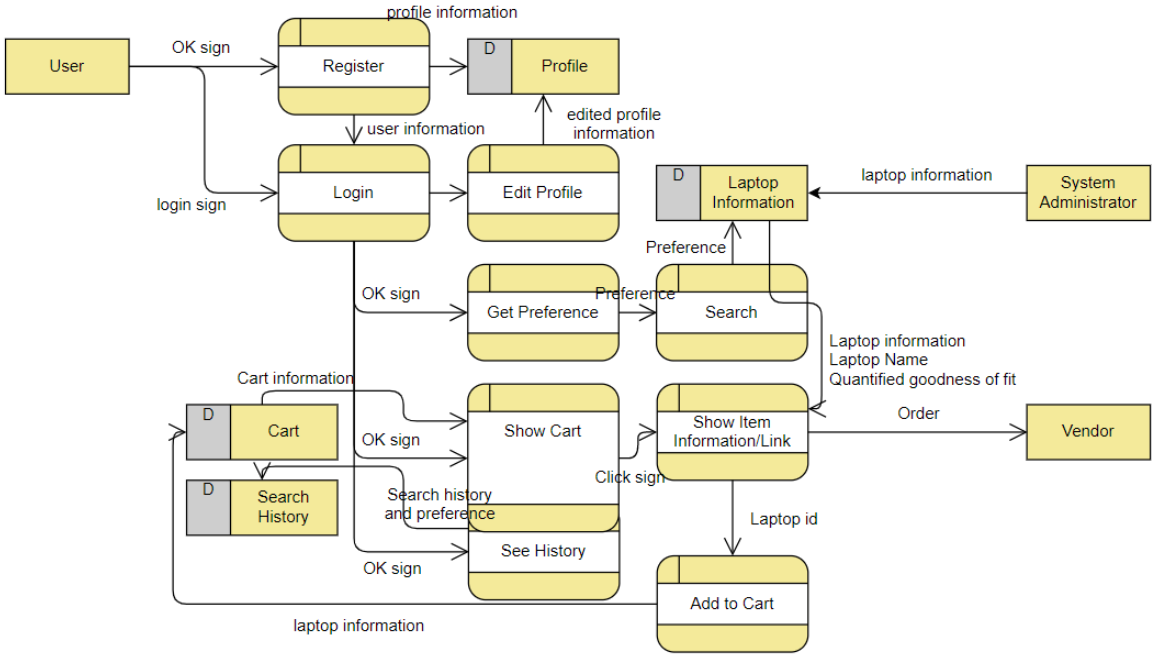
[Table 34] Vendor_Link

Field	Key	Constraint	Description
laptop_product_code	PK/FK	Not Null	Ref. Laptop
vendor_name	PK	Not Null	Vendor
link		Not Null	Link to vendor’s shop



[Figure 2] Entity Relationship Diagram

3.2.4. Data Flow Diagram



[Figure 3] Data flow diagram

3.3. Performance Requirements

The following requirements are based on estimates and may be changed in the final application.

3.3.1. Static numerical requirement

- The system management tool supports only one administrator.
- The system supports only one simultaneous user for each mobile device. The system is not support multiple connection on the same device. However, after disconnecting, user can switch accounts and access.
- The system should run smoothly on mobile devices with least 1GB RAM and 1.0 GHz single processor. And the system support Android 6.0 or more recent version.

3.3.2. Dynamic numerical requirement

- The system run smoothly for least 200 simultaneous users. And the system can be handles least 10,000 active user account and profile.
- The application should run within 10 seconds.

- Each account must be activated within 5 seconds of linking.
- The login process should be completed within 5 seconds.
- Product search results should be displayed within 5 seconds.
- When entering a profile, data must be stored on database within 5 seconds, and database updates must be completed within 5 seconds.
- The system should be able to recommend a laptop within 10 seconds based on the profile and entered options.

3.4. Logical Database Requirements

The system uses Firebase's database library named ‘Firestore’ to manage data. It should be able to store user information and profile data on database. It should be able to store performance and recommendation information for each laptop on database. Duplicate queries must be processed at high speed, and the constraints and performance of general database systems must be satisfied.

3.5. Design Constraints

The system should contain only components that can be distributed under the MIT license. The system should be able to access from various mobile devices with Android operating system, and the administrator must be able to access and manage it through a web browser. The system must be designed to run on Firebase and use the Firestore database.

3.6. Standards compliance

All programs in the system are written in accordance with the JAVA standards, and other matters follow conventional programming techniques. The names of functions and variables in the program use camel notation, and underscore notation is applied to databases. System management tool should be access through web browser in accordance with HTML5 standard.

3.7. Software System Characteristics

Software system characteristics are revealed through non-functional requirements. So, this section describes several non-functional requirements of the system. Non-functional requirements are classified Product Requirements, Organization Requirements, External

Requirements as follows.

3.7.1. Product Requirements

Product requirements specify or constrain the runtime behavior of the software. Our system should satisfy following requirements.

3.7.1.1. Usability Requirements

This is one of most important non-functional requirements of our system. The system should be easy to use by non-technical user and should be organized in such a way that user errors are minimized. The usage of technical terms should be minimized, explain it easily if needed. Each user shall be able to use all the system functions instinctively without learning manual. That means, user interface should be simple, instinctive, and easy to use.

3.7.1.2. Performance Requirements

Recommendation Algorithm is most time-consuming operation in our system. Also, usability degrades as it takes longer time. It must give the result to user within 5 sec.

3.7.1.3. Usability Requirements

The system should provide its purposed service at any given time. The recommendation system should provide the service while error being minimized. Its result should be reliable as expected by user. The average number of errors made by recommendation system shall not exceed 5 percent. Get user input and no result is not allowable.

3.7.1.4. Security Requirements

The users should be properly authenticated before using the system. It should make sure that an unauthorized user cannot gains access as system manager and makes system unavailable. Also, an unauthorized user cannot gains access as system user to confidential information such as user’s personal Information, ID, and password.

3.7.2. Organizational Requirements

These requirements are broad system requirements derived from policies and procedures in the customer’s and developer’s organizations

3.7.2.1. Environmental Requirements

The system's laptop data comes from and rely on 'Naver Shopping'. They are already keeping enormous amount of laptop information and providing rich API to query.

3.7.2.2. Operational Requirement

Users of the system shall identify themselves using their email. The system recommends the most appropriate laptops to non-tech user with fewer inputs. It shows the results with user-friendly graphical forms. It shows the result within 5 seconds. The system gives links of shopping malls to user. The system works as application in smartphone and work at any-time.

3.7.3. External Requirements

It covers all requirements that are derived from factors external to the system and its development process.

3.7.3.1. Safety / Security Requirement

The system should guarantee that any personal information is not accessed by external system. The system should be safe enough to prevent user's data from damaging from external natural disaster.

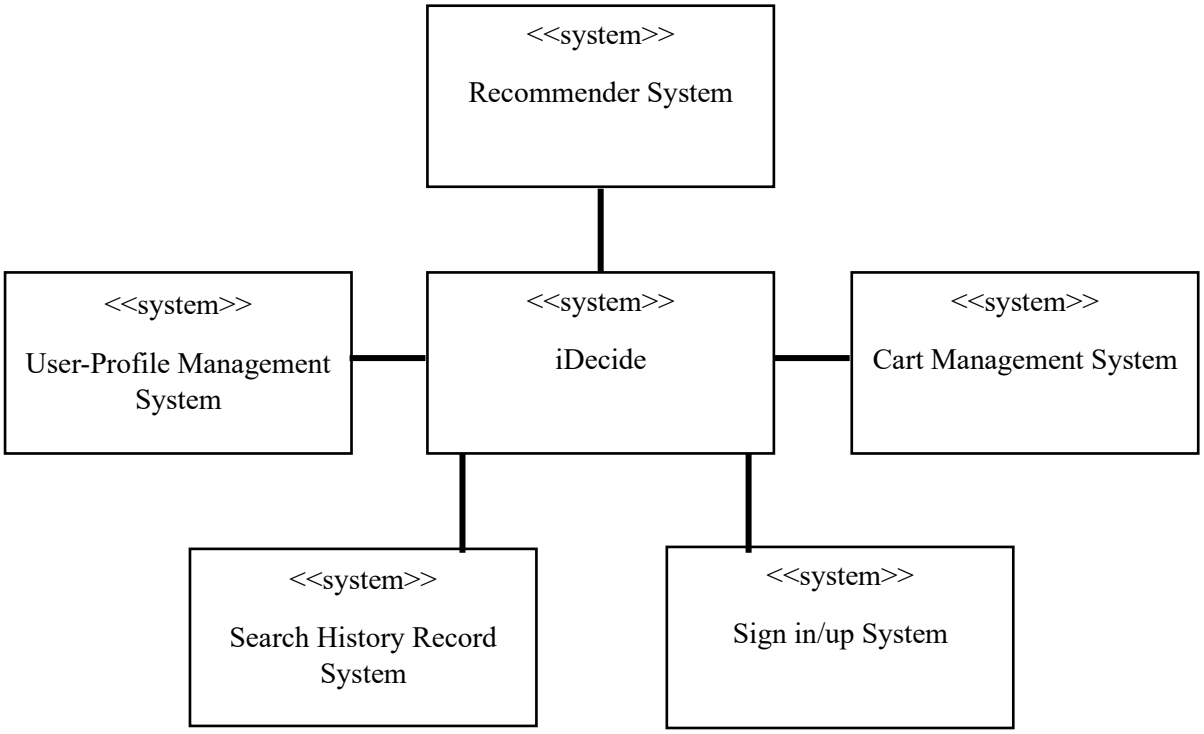
3.7.3.2. Regulatory Requirement

'Naver open API' is not completely free. Permission should be given from Naver Corporation before commercially used. User's privacy should not be violated according to law. The system should be developed in accordance with a national Privacy standard.

3.8. Organizing the Specific Requirements

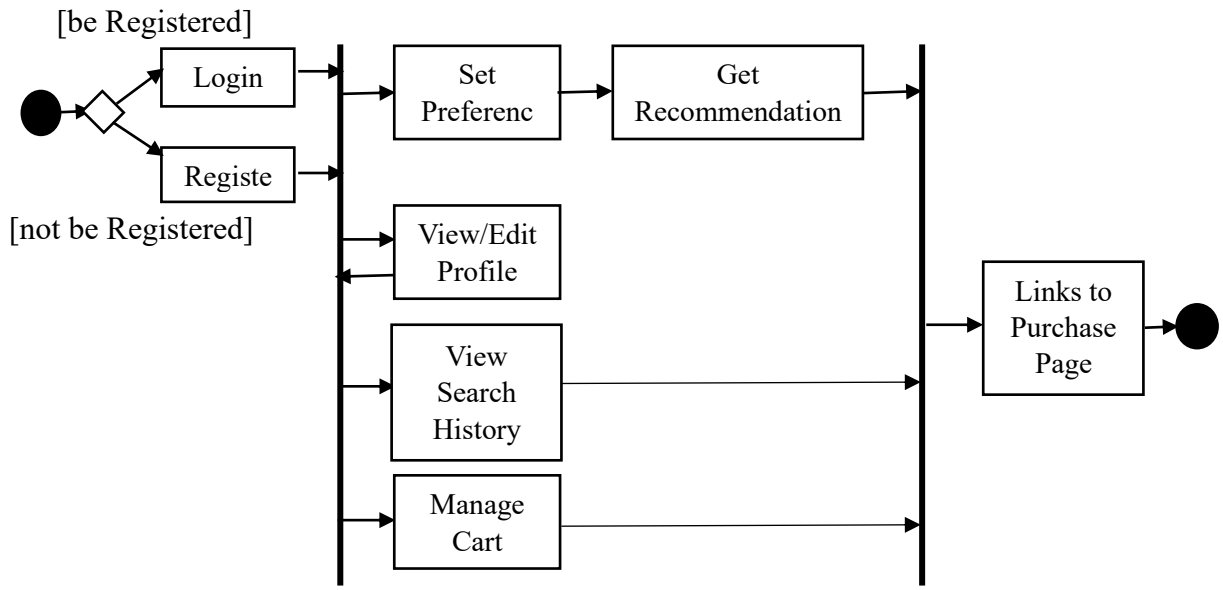
In this section, we describe the system model using graphical notation based on Unified Modeling Language (UML) and tabular form. System model describes the relationship among the system, sub-systems, components, and surrounding environments, showing more specific requirements.

3.8.1. Context Model



[Figure 4] Context model

3.8.2. Process Model



[Figure 5] Overall process model

3.8.3. Interaction Model

See 3.2.2. Use Case Diagram

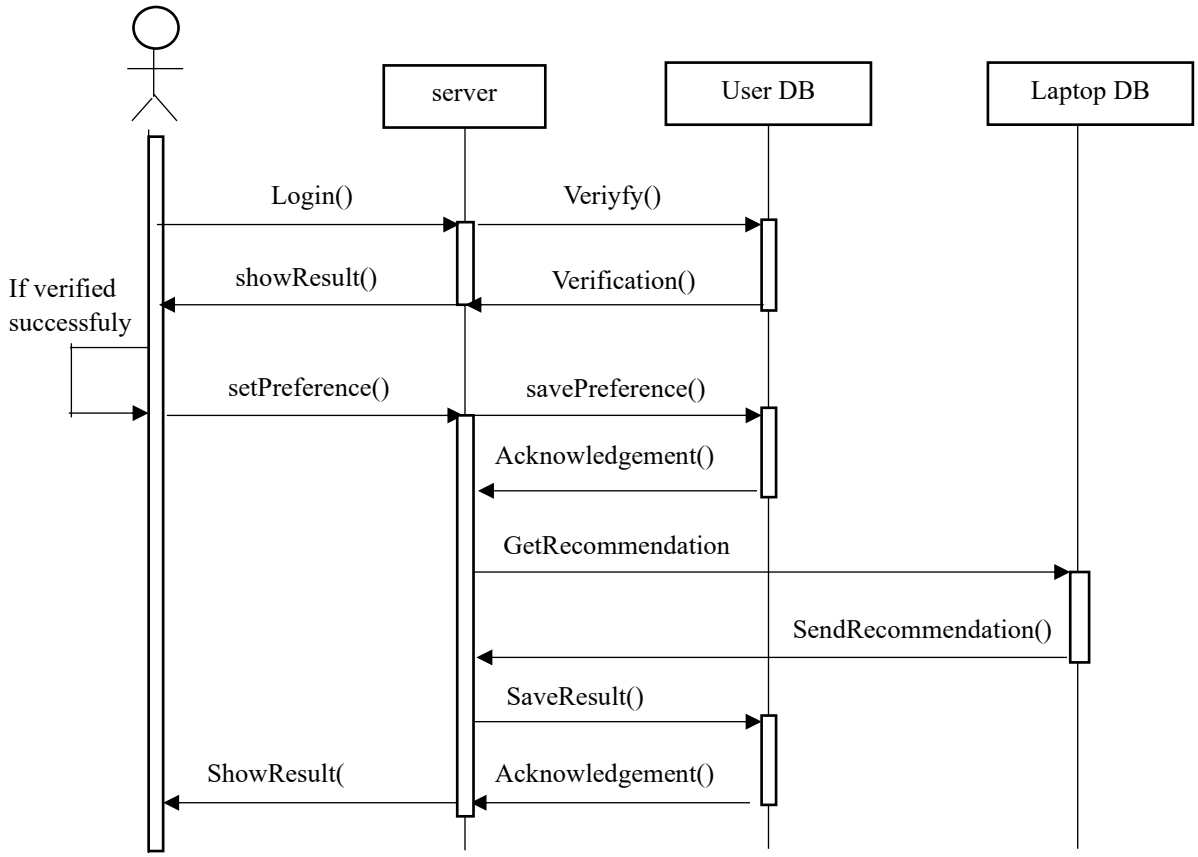
3.8.4. Behavior Model

3.8.4.1. Data Flow Diagram

See 3.2.4. Data Flow Diagram

3.8.4.2. Sequence Diagram

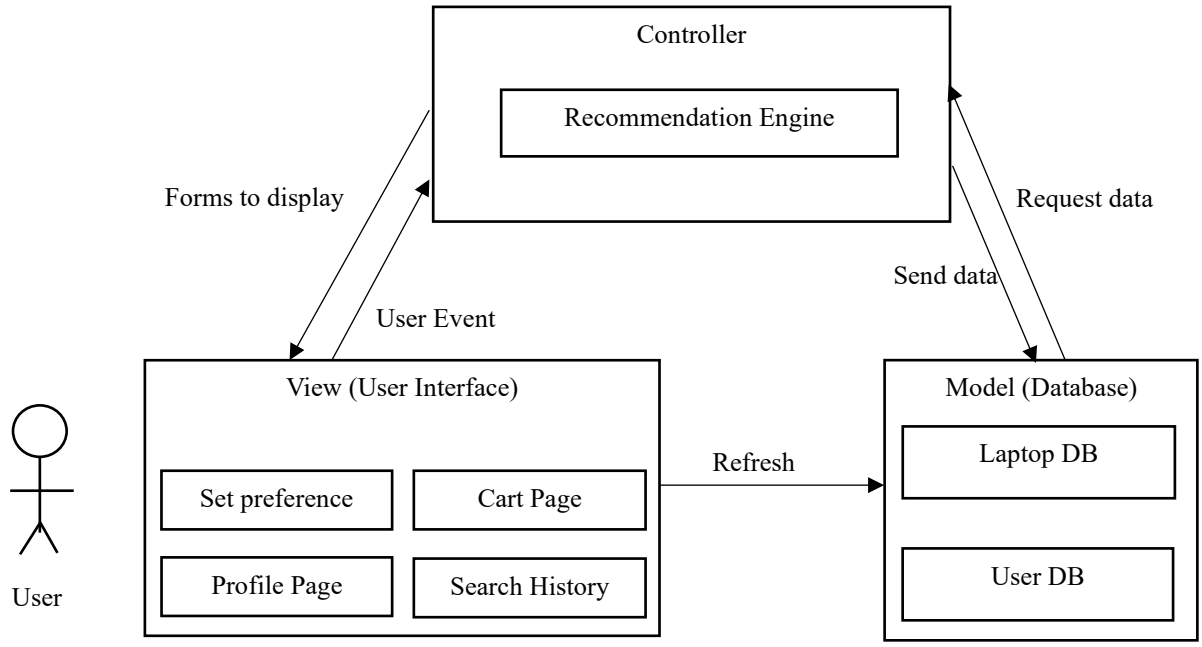
It describes mainly the sequence of processing recommendation which is our main system.



[Figure 6] Sequence diagram

3.9. System Architecture

This Section presents a high-level overview of the anticipated system architecture, showing the distribution of functions across system modules. We use MVC pattern as graphical models of the architecture.



[Figure 7] System architecture of the system

3.10. System Evolution

In this section, we describe the fundamental assumptions on which the system is based, and any anticipated changes due to hardware evolution, changing user needs, and so on. This section is useful for system designers as it may help them avoid design decisions that would constrain likely future changes to the system.

3.10.1. Limitation and Assumption

We are only dealing with laptop at this moment, excluding desktop. The characteristics of laptop and desktop are quite different. Laptop is being released as a complete product and replacing old component with other is not well supported. But, for desktop, many people are inclined to assembles their own desktop, buying individual components like CPU, GPU, RAM, etc. Considering the facts that market of laptop is growing fast compared to desktop, the time schedule of the development, and the simplicity. we are only focusing on laptop. And we assume that each components of laptop is not replaceable at this moment.

3.10.2. Evolutions of Hardware and Change of User Requirements

Each components of laptop would be replaceable like desktop in the future. Our recommendation system can be useless because we are only dealing with complete products, not each individual component. If these paradigm change happens, many people would buy

individual component and make their own laptop for their own purpose like desktop. That means user requirements would be changed. Then, our system would be deprecated. We need to prepare the paradigm change.

So, before it happens, we need to adapt the system to embrace each components of desktop. Each component could be recommended individually according to user's requirements. Or, complete desktop which consists of each recommended component could be recommended.

But there are problems because our main users are expected to be not familiar with hardware. So, we need to show the role of each components using user-friendly images.

4. Supporting Information

4.1. Software Requirement Specification

This software requirements specification was written in accordance with the IEEE Recommendation (IEEE Recommended Practice for Software Requirements Specifications, IEEE-Std-830).

4.2. Document History

[Table 35] Document History

Date	Version	Description	Writer
2020/05/04	0.1	Style and overview	Kyungyeon Park
2020/05/11	1.0	Addition of 2.3, 2.4, 2.5	Kyungyeon Park
2020/05/11	1.1	Addition of 3.1, 3.2	Seungrok Yoon
2020/05/11	1.2	Addition of 3.7	Hogi Min
2020/05/12	1.3	Addition of 1.1, 1.2, 1.3	Sara B.Zaki
2020/05/12	1.4	Addition of 2.1, 2.2, 2.3	Allan Tazhenov
2020/05/12	1.5	Addition of 3.3, 3.4, 3.5, 3.6	Hanwoul Lee
2020/05/12	1.6	Revision of 3.2.1	Seungrok Yoon
2020/05/12	1.7	Revision of 1.1, 2.2 and style	Kyungyeon Park
2020/05/13	1.8	Revision of 2.1, 3.2	Kyungyeon Park

Date	Version	Description	Writer
2020/05/13	1.9	Addition of 3.2.2	Hanwoul Lee
2020/05/13	1.10	Addition of 3.8, 3.9 and revision of 3.7	Hogi Min
2020/05/13	1.11	Revision of 2	Kyungyeon Park
2020/05/13	1.12	Addition of 3.8, 3.9, 3.10	Hogi min
2020/05/13	1.13	Addition of 1.5	Sara B.Zaki
2020/05/13	1.14	Addition of 3.2.3	Hanwoul Lee
2020/05/13	1.15	Revision of 3.1, 3.2	Seungrok Yoon
2020/05/13	1.16	Addition of 1.4 and revision of 3.2.4	Kyungyeon Park