

|  |
| --- |
| Fitts:  Requirement Specification |
|  |
| Team 2   |  |  | | --- | --- | | Nguyen Khanh Binh | 2019711305 | | Lee Gumsik |  | | Haziq |  | | Jang Hanbin |  | |



Table of Contents

[1. User Requirements Definition 3](#_Toc24130997)

[1.1 Functional Requirements 3](#_Toc24130998)

[1.2 Non-functional Requirements 4](#_Toc24130999)

[2. System Requirements Specification 5](#_Toc24131000)

[2.1 Functional Requirements – Frontend 5](#_Toc24131001)

[2.2 Functional Requirements – Backend 9](#_Toc24131002)

[2.3 Non-Functional Requirements 11](#_Toc24131003)

[3. System Architecture 13](#_Toc24131004)

[3.1 Frontend Diagram 13](#_Toc24131005)

[3.2 Backend Diagram 13](#_Toc24131006)

[3.3 Use-case Diagram 13](#_Toc24131007)

# 1. User Requirements Definition

This section covers the definitions for the user requirements of the proposed system. The requirements can be categorized into functional and non-functional requirements.

## Functional Requirements

The functional requirements include, but are not limited to, the following items:

1. Sign up/Sign In

* This is basic requirement for users to use the Fitts application. At first, Users create an account of Fitts system with ID, password and e-mail or Users can create account via Facebook or Kakao account. After that, users can sign in the system with ID, Password. When uses sign in, system should authenticate the user by matching given information with account information in the DB.

1. User personalization

* After login, Users must input the information about body like height, weight, body shape type, shoes size, and style of clothing.

1. Feed

* Show up the photos of other users which write and post the photo to review about the clothes that they bought with a link to that product.

1. Shop

* Show up many products with the price available in Fitts system and a shop that sell that product. Users can bookmark the product and the shop. It works like normal shopping website.

1. Disclosure of Product Information

* Aside from what is shown in the photo, it should include that person body information.

1. My page

* My page provides users with information related to ordering and activity. Users can check their orders and the shipping information for the order. It also provides bookmark products, love products, coupons, rewards, …

1. Fitts point

* On My page show up a Fitts points. This is the point earn from writing the review for product, user can stack up this point to buy coupon or product in Fitts shop. Earn point by posting reviews, shopping, inviting peoples, and complete some events

1. Search

* Users can search for other users, shop or product in this area.

1. Cart

* All the users added product to buy will show up here.

1. Review

* User can upload reviews about outfits, shoes, bags using their own creativity

## 1.2 Non-functional Requirements

The non-functional requirements can be further classified into Product Requirements, Organizational Requirements, and External Requirements as follows:

1. Product Requirements
2. Performance

* It needs the same performance with others ecommerce application to buy the products.
* It also needs the good system to upload and download image for products. It needs to balance the network to make it fast enough when users need update feed.
* The Fitts point must be stable, reliable and still have the interesting for users.

1. Security

* Review must be reliable and must guarantee the anonymity of review writer. And each personal information must be protected from outer access and there for we need security. And, Id and passwords should be protected by saving it securely.

1. Usability

* The UI of the application should have a clear screen UI to provide important information and as least button as possible deal to the density of product will show up in the feed screen. The UX should be clear and easy to use for users.

# 2. System Requirements Specification

This chapter specifies the functional and non-functional requirements. We focus on the following detail requirements: Name, description, inputs, source, outputs, destination, action, requirements, pre-condition.

## 2.1 Functional Requirements – Frontend

1. Sign up

|  |  |
| --- | --- |
| Name | Sign up |
| Description | User can register their own account for using application. |
| Inputs | User input |
| Outputs | User can get the account for application |
| Action | When the user first opens the application, it requires user to login or sign up for the application with ID, name and password. |
| Requirements | User must not register before |
| Pre-condition | User should have email and certificate it. |
| Post-condition | - |

Table 1: Signup Function

1. Login

|  |  |
| --- | --- |
| Name | Login |
| Description | User can login to use the application after signing up or already have the account |
| Inputs | User input |
| Outputs | An object containing all details associated with one specific item. |
| Action | User use the email and password that signed up to login into the application. |
| Requirements | Must have the credential in the application system |
| Pre-condition | - |
| Post-condition | - |

Table 2: Show Item Detail

1. Add Product to Bookmark

|  |  |  |
| --- | --- | --- |
| Name | Add Item to Bookmark | |
| Description | Store a specific product in the bookmark list of the currently login user. | |
| Inputs | User ID | Login user id |
| Product ID | Product ID |
| Outputs | Product must be added to the bookmark list | |
| Action | User information and product information are obtained from the database by using the User ID and Product ID, respectively. Check whether the user has already bookmarked the product in the database. If the user already has a bookmark, bookmarked this product. | |
| Requirements | User ID, Product ID | |
| Pre-condition | The product does not exist in the bookmark of the user. | |
| Post-condition | The product is added to the user's bookmark, and the Add to Bookmarks button changes to the Remove from Bookmarks button. | |

Table 3: Add Product to Bookmark

1. Add Store to Bookmark

|  |  |  |
| --- | --- | --- |
| Name | Add Store to Bookmark | |
| Description | Store a specific Store in the bookmark list of the currently login user. | |
| Inputs | User ID | Login user id |
| Store ID | Store ID |
| Outputs | Product must be added to the bookmark list | |
| Action | User information and Store information are obtained from the database by using the User ID and Store ID, respectively. Check whether the user has already bookmarked the Store in the database. If the user already has a bookmark, bookmarked this product. | |
| Requirements | User ID, Store ID | |
| Pre-condition | The Store does not exist in the bookmark of the user. | |
| Post-condition | The Store is added to the user's bookmark, and the Add to Bookmarks button changes to the Remove from Bookmarks button. | |

Table 4: Add Store to Bookmark

1. Show Feed List

|  |  |
| --- | --- |
| Name | Show Feed List |
| Description | Show recommendation list based on user’s information |
| Inputs | User ID |
| Outputs | List of products and stores which match user body information |
| Action | When user open the application, automatically show products and store that matched with user body information. |
| Requirements | User ID |
| Pre-condition | User must input body information |
| Post-condition | - |

Table 5: Show Feed List

1. Add Review

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Add Review | | | |
| Description | User write a review about one product; it should include the photo of that product with the product information as well as user body information. | | | |
| Inputs | User ID | Product ID | Store ID | User review |
| Outputs | review product | | | |
| Action | User write a review of a product, included photos and many others information, then upload it to system, then user gain a Fitts point. | | | |
| Requirements | User must be logged in, a review must include body information, photos, … | | | |
| Pre-condition | None. | | | |
| Post-condition | None. | | | |

Table 6: Add Review

1. My page

|  |  |
| --- | --- |
| Name | My page |
| Description | Pages that display User information |
| Inputs | User ID |
| Outputs | Authenticated user information object |
| Action | Personalized page for each user. Through this page, user's order/delivery inquiry, purchase history, personal information management, and his/her review management can be processed. Therefore, this page is only accessible by the logged in user |
| Requirements | User must be logged in |
| Pre-condition | non-authorized user cannot access. |
| Post-condition |  |
| Side effects | Frontend will be changed into authorized version including specific user’s information. |

Table 7: My page

1. Shopping

|  |  |
| --- | --- |
| Name | Shopping |
| Description | This screen show up the products with price for user to buy and show many store in the application system . |
| Inputs | User body information |
| Outputs | Authenticated user information object |
| Action | User can go to this screen and choose which product that they want to buy or just take a look at the product for price, color, … reference or they also can go to each store in the system. User can bookmark product and store here. |
| Requirements | User body information |
| Pre-condition | User must input body information |
| Post-condition | - |
| Side effects | - |

Table 8: Shopping

## 2.2 Functional Requirements – Backend

1. Login

|  |  |  |
| --- | --- | --- |
| Name | Login Function | |
| Description | Authenticate and authorize the user | |
| Inputs | ID | Id string of user |
| Password | Password string of user |
| Source (Input) | User input | |
| Outputs | Authenticated user information object | |
| Destination (Output) | User terminal (web browser) | |
| Action | Search user information stored in the database by using the past ID. After hash the past password, compare it with the hash stored in the database. If correct, create a user object containing other user information (nickname, authority, …) and session ID, and return the object to the web browser. | |
| Requirements | Both Id and password must be specified. | |
| Pre-condition | User can access only non-authorized area. | |
| Post-condition | User can access all authorized area. | |
| Side effects | Frontend will be changed into authorized version including specific user’s information. | |

Table 9: Login Function

1. Review Collecting System

|  |  |  |
| --- | --- | --- |
| Inputs | Target Site List | 크롤링할 목표 사이트의 목록 |
| Source (Input) | Backend Application Server, Cron Scheduler | |
| Outputs | Parsed Item Review List | |
| Destination (Output) | Review Analysis System | |
| Action | 타겟 사이트의 리스트를 순회하며, 목표 상품들을 검색해 나오는 상품 페이지의 리뷰 리스트를 파싱한다. 제목과 작성자, 리뷰 내용, 사진을 분류해 목록화한다. | |
| Requirements | - | |
| Pre-condition | - | |
| Post-condition | - | |
| Side effects | - | |

Table 10: Review Collecting System

1. Review Analysis System

|  |  |  |
| --- | --- | --- |
| Inputs | Raw Reviews | 크롤링된 상품 리뷰 목록 |
| Source (Input) | Review Collection System, Backend Application Server | |
| Outputs | Analyzed Review List | |
| Destination (Output) | Review Database Updater | |
| Action | 크롤링된 상품 리뷰 목록의 리뷰 내용을 구글 클라우드 자연어 처리 API에 쿼리한다. 쿼리 결과에서 키워드 목록, 긍정 평가 정도를 따로 분류해 저장한다. 긍정 평가 정도와 상품 사진 게시 여부, 리뷰 길이, 키워드 종류를 평가해 리뷰의 중요도를 매긴 다음 긍정 평가 정도와 곱해 리뷰 점수에 반영한다. | |
| Requirements | - | |
| Pre-condition | 분석되지 않은 상품 리뷰 목록 | |
| Post-condition | 분석된 상품 리뷰 목록 | |
| Side effects | - | |

Table 11: Review Analysis System

1. Review Database Updater

|  |  |  |
| --- | --- | --- |
| Inputs | Review List | 분석 완료된 리뷰 리스트 |
| Source (Input) | Review Analysis System, Backend Application Server | |
| Outputs | - | |
| Destination (Output) | - | |
| Action | 리뷰 분석 시스템에서 분석을 완료한 리뷰의 리스트를 저장한다. 리뷰 제목, 작성자, 출처 사이트, 내용, 키워드, 긍정 평가 정도, 중요도를 포함해 데이터베이스에 기록한다. | |
| Requirements | - | |
| Pre-condition | - | |
| Post-condition | - | |
| Side effects | 리뷰 데이터가 업데이트될 경우 현재 표시되고 있는 상품 정보에 변동이 발생한다. 따라서 프론트엔드에서의 다음 요청시 업데이트된 상품 정보를 리턴해야 한다. | |

Table 12: Review Database Updater

## 2.3 Non-Functional Requirements

1. Product requirements
2. Performance

* It needs the same performance with others ecommerce application to buy the products.
* It also needs the good system to upload and download image for products. It needs to balance the network to make it fast enough when users need update feed.
* The Fitts point must be stable, reliable and still have the interesting for users.

1. Security

* Review must be reliable and must guarantee the anonymity of review writer. And each personal information must be protected from outer access and there for we need security. And, Id and passwords should be protected by saving it securely.

1. Usability

* The UI of the application should have a clear screen UI to provide important information and as least button as possible deal to the density of product will show up in the feed screen. The UX should be clear and easy to use for users.

1. External Requirements
2. Personal information Protection

* The system collects and stores user information when they sign up to the webpage. Only some basic information of users is gathered and of course every personal information should be confidential to get trusted from the users.

1. External System Policy

* As this system provides information reprocessing the review list of other systems, the system that prohibits unauthorized reproduction of data may violate the operation policy depending on the data collection method. Therefore, it is necessary to carefully check the operation policy of the information collection target system and collect data without violating the policy.

# 3. System Architecture

## 3.1 Frontend Diagram

## 3.2 Backend Diagram

## 3.3 Use-case Diagram