# System requirements specifications

## Objective

시스템 Requirements specification 단계는 Functional Requirements 하고 Non-functional Requirements의 요구사항을 구체적으로 명시하는 것이다. 구조는 한 기능별 이름, 설명, 입력, 출력, 처리, 조건 6 부분으로 나눌 수 있다.

## Functional Requirements

1. Login Function

|  |  |
| --- | --- |
| 이름 | Login Function |
| 기능 | 카카오톡이나 네이버로 로그인 |
| 설명 | ‘Eat it’ 처음에 한국에서만 나와서 한국인들이 보통 다 있는 카카오톡이나 네이버로 로그인 기능이다. 이 기능을 통해 사용자들이 더 빠르고 쉽게 ‘eat it’의 서비스를 이용할 수 있는 목적이다. |
| 입력 | 카카오톡이나 네이버의 API를 이용하여 사용자의 카카오 톡이나 네이버 아이디를 얻는다. 처음에 로그인 시에 닉네임, 나이, 성별 등을 추가로 입력할 수 있다. |
| 출력 | 로그인이 정상적으로 되면 마이페이지 화면으로 이동한다 |
| 처리 | 사용자의 카카오톡이나 네이버 내부 아이디라는 정보가 디비에 없으면 처음에 로그인 후에, 시스템은 사용자의 정보를 디비에서 저장한다. |
| 조건 | 사용자가 자기의 정보를 제공하는 걸 동의해야 함. |

1. Explore First Evaluation Function

|  |  |
| --- | --- |
| 이름 | Explore First Evaluation Function |
| 기능 | 처음에 로그인 후에 사용자의 취향평가 |
| 설명 | 사용자는 처음에 로그인 한 후에 음식에 대한 취향의 정보가 아무도 없으니 사용자의 좋아하는 요리, 또는 식품을 통하여 기본적인 취향 분석 |
| 입력 | 사용자는 좋아하는 요리와 음식 |
| 출력 | 구매할 식품과 메뉴 추천 |
| 처리 | 사용자가 입력한 정보를 받고 분석하여 선호할 가능성이 높은 요리와 식품이 나온다. |
| 조건 | 사용자는 좋아하는 음식 입력 |

1. Preference Analysis Function

|  |  |
| --- | --- |
| 이름 | Preference Analysis Function |
| 기능 | 사용자의 취향을 machine learning으로 분석 |
| 설명 | 사용자는 ‘eat it’ 쓰면서 구매한 식품, 또는 검색한 메뉴, 좋아하는 음식 등을 정보를 수집하고 사용자의 취향을 더 완벽하게 분석한다. |
| 입력 | 사용자는 ‘eat it’ 쓰는 과정에서 수집된 메뉴, 식품, 음식 정보 |
| 출력 | 사용자의 입맛, 좋아하는 요리 스타일 |
| 처리 | 사용자가 선호한 요리와 식품을 바탕으로 사용자의 입맛과 식사 습관을 분석한다. |
| 조건 | 사용자는 앱을 한 번 쓰거나 계속 씀 |

1. Main Menu Recommendation Function

|  |  |
| --- | --- |
| 이름 | Main Menu Recommendation Function |
| 기능 | 사용자에게 메뉴 추천 |
| 설명 | 사용자의 입맛, 좋아하는 식품, 식사 습관 등 정보를 다 machine learning으로 분석하여 받아 가지고 이 정보들을 바탕으로 사용자에게 메뉴 추천한다. |
| 입력 | ‘추천받다’라는 버튼 |
| 출력 | 사용자가 좋아할 수 있는 메뉴 있는 화면 |
| 처리 | 사용자의 입맛, 좋아하는 식품, 식사 습관 등 정보를 받고 machine learning으로 사용자가 좋아할 수 있는 메뉴를 분석하고 출력한다. |
| 조건 | 사용자의 입맛, 좋아하는 식품, 식사 습관 등 정보가 있어야 함 |

1. Menu Search Function

|  |  |
| --- | --- |
| 이름 | Menu Search Function |
| 기능 | 메뉴를 검색하는 기능 |
| 설명 | 사용자가 메뉴의 이름, 들어가는 식품, 맛 등 요소로 메뉴를 검색할 수 있다. |
| 입력 | 메뉴에 관한 요소 |
| 출력 | 메뉴 정보를 담는 검색 결과 화면 |
| 처리 | 메뉴의 요소를 받고 디비에서 관한 메뉴를 찾아서 출력한다. |
| 조건 | 메뉴에 관한 요소 입력해야 함. |

1. Send to Cart Function

|  |  |
| --- | --- |
| 이름 | Send to Cart Function |
| 기능 | 상품을 장바구니에서 담는다. |
| 설명 | 사용자는 관심이 있는 상품을 상품 옆에 있는 ‘장바구니’ 버튼으로 장바구니에서 넣을 수 있다. |
| 입력 | 상품 옆에 있는 ‘장바구니’ 버튼 |
| 출력 | 이 상품을 장바구니에서 담는다. |
| 처리 | 관한 상품의 정보를 사용자의 장바구니에 보낸다. 디비 업데이트. |
| 조건 | 같은 상품이 장바구니에 없고 장바구니에 담을 수 있는 상품 제한 초과하지 않는다. |

1. Display Shopping Cart Function

|  |  |
| --- | --- |
| 이름 | Display Shopping Cart Function |
| 기능 | 장바구니 확인 |
| 설명 | ‘장바구니’ 버튼을 통하여 사용자는 자기의 장바구니에 무엇이 있는지 확인할 수 있다. |
| 입력 | ‘장바구니’ 버튼 |
| 출력 | 장바구니 화면 |
| 처리 | 사용자가 ‘장바구니’ 버튼을 누르면 디비에 사용자의 장바구니 정보를 받고 시스템으로 화면 생성 |
| 조건 |  |

1. Edit and Submit Shopping Cart Function

|  |  |
| --- | --- |
| 이름 | Edit and Submit Shopping Cart Function |
| 기능 | 장바구니를 수정하고 장바구니에 있는 상품을 주문 요청 |
| 설명 | 장바구니에 상품의 수량 수정, 삭제 등 동작을 편집하고 장바구니에 상품을 주문하는 것이다. |
| 입력 | 수량 수정, 삭제, 상품 선택 |
| 출력 | 다시 편집된 장바구니 |
| 처리 | 사용자의 동작에 따라 장바구니 디비 업데이트 |
| 조건 | 장바구니 빈 상태 아님. |

1. Order Process Function

|  |  |
| --- | --- |
| 이름 | Order Process Function |
| 기능 | 상품을 주문하는 기능 |
| 설명 | 장바구니에 있는 상품을 주문하거나 상품을 직접 주문해도 된다. 주문할 요청을 하고 배송지와 받는 사람 등을 설정한 다음에 결제한다. |
| 입력 | ‘주문하다’ 버튼, 배송지와 받는 사람의 정보, 결제수단 선택 |
| 출력 | 결제완료 화면, 또는 주문 내역 |
| 처리 | ‘주문하다’ 다음에 배송지와 받는 사람의 정보 입력 화면으로 이동, 다음에 결제 화면으로 이동, 결제 완료된 후에 디비 업데이트 |
| 조건 |  |

1. Display My Page Function

|  |  |
| --- | --- |
| 이름 | Display My Page Function |
| 기능 | 마이페이지 화면을 확인하는 기능 |
| 설명 | 마이페이지에 화면에 사용자 자신의 여러 가지 정보를 볼 수 있고 수정할 수 있는 부분을 수정해도 된다. 예를 들면, 사용자의 닉네임 수정, 주문 내역 조회. |
| 입력 | ‘마이페이지’ 버튼 |
| 출력 | 마이페이지 화면으로 이동 |
| 처리 | 사용자가 ‘마이페이지’ 버튼을 누르면 시스템이 디비에 있는 사용자의 정보로 마이페이지 화면 생성 |
| 조건 |  |

1. Order Management Function

|  |  |
| --- | --- |
| 이름 | Order Management Function |
| 기능 | 주문내역 확인하는 기능, 수취확인 기능, 또는 반품이나 환불기능 |
| 설명 | 사용자가 주문한 상품의 정보를 확인할 수 있고, 상품을 정상적으로 받은 후에 수취확인, 받은 상품은 정상상태가 아닌 경우, 반품과 환불 기능 지원한다. |
| 입력 | ‘주문내역’ 버튼, ‘수취확인’ 버튼, ‘환불신청’버튼 |
| 출력 | ‘주문내역’ 버튼을 누르면, 주문내역 화면으로 이동, ‘수취확인’ 버튼을 누르면, 구매한 상품을 수취확인 된다. ‘환불신청’버튼을 누르면 환불 단계로 이동한다. |
| 처리 | 사용자 디비에 관련된 정보를 보여주고 업데이트 |
| 조건 | 주문한 상품이 있어야 함. |

1. Post Evaluation Function

|  |  |
| --- | --- |
| 이름 | Post Evaluation Function |
| 기능 | 상품을 평가하는 기능 |
| 설명 | 상품을 구매한 후에 상품에 대한 평가를 할 수 있다. |
| 입력 | 상품평가의 글, 또는 사진 |
| 출력 | 그 상품의 평가 게시판에서 구매자가 평가한 내용을 업데이트 |
| 처리 | 평가한 내용을 서버로 보낸다. |
| 조건 | 수취확인 된 상태에 평가 가능 |

## Scenario

1. Sign up/Login Scenario

A.1 처음에 이 어플을 사용하는 사용자들이 카카오톡이나 네이버 계정으로 sign up하고 로그인 할 수 있다. 그 이유는 이 어플이 한국에서 생겨서 대부분 한국 사람들 다 가지는 계정으로 하면 더 쉽고 편하게 어플을 쓸 수 있다.

A.2 앞으로 다시 앱을 사용할 때에는 이미 sign up된 카카오톡이나 네이버 계정으로 하면 된다.

A.3 처음에 sign up 한 다음에 사용자의 닉네임 다시 설정할 수 있고 카카오톡이나 네이버의 회원정보 그대로 이여 써도 된다.

1. First Menu Evaluation Scenario

B.1 처음에 로그인 한 다음에, 사용자의 기본적인 음식 취향과 식사습관 등 정보를 파악하기 위해 첫 번째 메뉴 평가를 해야 한다. 사용자가 자기의 일상 메뉴 정보를 입력하고, 또는 자기의 입맛과 식사 습관 등 정보를 입력한다.

B.2 이 정보들을 가지고 알고리즘을 통하여 사용자의 기본적인 음식 선호를 분석하고 결과를 출력한다.

1. Main Menu Recommendation Scenario

C.1 사용자가 ‘eat it’ 쓰면서 모든 입력되고 받을 수 있는 정보를 받는다.

C.2 가지는 정보를 통하여 사용자에게 맞춤형 메뉴를 추천한다. 예를 들면, 사용자가 구매한 식품, 좋아하는 요리, 검색한 요리법 등을 모든 정보를 분석하여 가능성이 높은 사용자는 좋아할 수 있는 메뉴를 사용자에게 추천하고 요리법도 같이 제공한다.

1. Menu Search Scenario

D.1 ‘eat ‘it’에 이미 동록된 메뉴를 검색할 수 있다. 메뉴의 이름, 들어가는 식품, 맛, 메뉴의 종류 (예: 중화요리) 등 메뉴와 관한 아무 요소를 입력하면 관련된 메뉴를 검색 결과로 나온다,

D.2 또는 사용자가 입맛이 변할 수도 있는 것 같은 원인들을 고려해서 검색하기 전에 사용자 지신의 취향 참고 여부 선택도 할 수 있다.

1. Add to Card and Order Scenario

E.1 사용자가 상품을 보면서 장바구니에서 담을 수 있다. 구매하고 싶은 상품을 다 담고 주문하고 싶으면 장바구니로 이동하고 주문할 상품을 선택하고 결제한다.

E.2 결제 단계에 먼저 배송지와 받는 사람의 정보를 입력하고 결제 수단을 선택한다. 결제 수단을 신용카드, payco, 카카오페이 등 여러 가지 결제 수단을 지원한다.   
E.3 결제 수단을 선택한 다음에 지불한다.

E.4 또는 장바구니 안 통하여 상품을 직접 주문해도 된다. 만약 상품을 직접 주문하면 바로 결제 단계로 이동한다.

1. After Order Evaluation Scenario

상품을 구매하고 수취확인 되면 상품에 대한 평가를 할 수 있다. 상품의 질, 배송 속도 등 모든 상품과 관련된 걸 다 평가할 수 있고, 평가한 내용을 상품 아래 상품평 내역에서 등록될 것이다.

Glossary

사용자: ‘eat it’ 서비스를 이용하는 사람.

API : 응용 프로그램이 컴퓨터 운영 체계나 데이터베이스 관리 시스템 따위의 다른 프로그램의 기능을 이용하기 위한 작용수단

업데이트 – 이미 있는 정보나 데이터를 수정, 변경하고 최신의 정보나 데이터를 추가하는 것.

서버: 서비스에 응답하는 프로세스와 하드웨어를 통괄하는 용어를 말한다.

디비: 데이터베이스를 말한다. 서버에서 서비스를 제공할 때, 필요한 정보들을 저장하고 있는 장소.

Machine learning: 컴퓨터가 학습할 수 있도록 하는 알고리즘과 기술을 개발한다.