

## 2. Analysis

# Refrigerator with U

## [ Revision history ]

Revision date	Version #	Description	Author
MM/DD/YYYY	0.00	Type brief description here	Author name
	0.1	Analysis 초안	

= Contents =

1. Introduction .....	4
2. Use case analysis .....	5
3. Domain analysis .....	21
4. User Interface prototype .....	22
5. Glossary .....	25
6. References .....	26

## 1. Introduction

### Summary

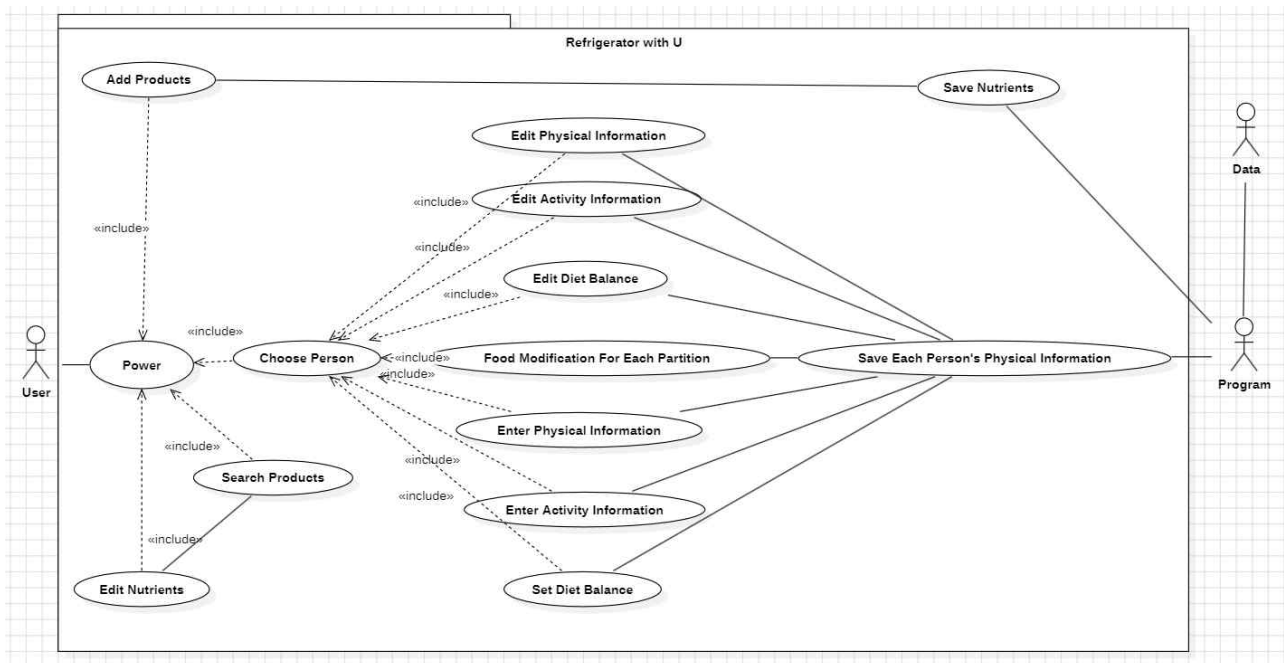
사람들은 식단을 계획하는 것을 어려워한다. 개개인의 목표에 따라 어느 정도의 식사량을 먹어야 하는지부터 냉장고의 식재료들을 어떻게 먹어야 할지까지 대부분이 모른다. 이러한 점들을 해소하기 위하여 만들게 된 시스템이 Refrigerator with U이다.

### Goal

Refrigerator with U 시스템은 고객들의 영양 섭취 개선 및 건강 개선을 목표로 한다. 식단을 수행하는 것만으로도 크고 작은 영양학적 질병이나 건강이 개선될 수 있다. 냉장고에 내장되어 있지만 스마트폰과의 연동으로 언제, 어디서나 식단을 추가/삭제 및 수행 정도를 확인할 수 있다. 이를 통해 특정 시간대에 과잉영양 섭취를 하게 되는 것을 예방할 수 있다. 결과적으로 전체적인 삶의 질을 향상시킬 수 있게 한다.

## 2. Use case analysis

### 1) Use case diagram



<그림 n> Use case diagram

Conceptualization에서 작성했던 System context diagram과 Use case list를 참조하여 작성하였다. Actor와 상호작용하는 Use case들은 실선 Association으로 연결해주었고, 특정 Use case를 실행하기 위해 반드시 실행해야 하는 Use case는 적절한 방향의 Include로 연결해주었다.

## 2) Use case description

Use Case #1 : Power	
GENERAL CHARACTERISTICS	
Summary	시스템을 시작하고 메인 함수를 호출하기 위해 사용한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	시스템이 실행되어야 한다.
Trigger	User가 시스템을 실행하고 첫 화면에서 접속을 입력할 때
Success Post Condition	시스템을 켜는 것에 성공하여 시스템 내의 기능들을 사용할 수 있게 된다.
Failed Post Condition	시스템을 켜는 것에 실패하여 시스템 내의 기능들을 사용할 수 없게 된다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자가 시스템을 시작하고자 할 때 시작한다.
1	사용자는 실행 여부를 묻는 화면에서 실행 여부를 입력한다.
2	실행하고자 하면 메인 메뉴를 띄우고, 그렇지 않다면 종료한다.
3	이 Use case는 종료할 때까지 계속 동작한다.
EXTENSION SCENARIOS	
step	Branching Action
2	2a. 2가지의 실행 여부 외의 선택을 입력한 경우 2a.1. 잘못 입력했다는 메시지를 띄우고, 다시 입력하게 한다. 2b. 실행 여부를 입력하지 않고 다음으로 넘어가려는 경우 2b.1. 실행 여부를 입력하지 않았다는 메시지를 띄우고, 다시 입력하게 한다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #2 : Add Products	
GENERAL CHARACTERISTICS	
Summary	User로부터 제품의 정보를 입력받고 txt 파일로 저장한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	시스템이 동작 중이어야 한다.
Trigger	메뉴에서 add products를 선택한다.
Success Post Condition	Food 폴더가 만들어지고 제품 txt 파일이 생성된다.
Failed Post Condition	제품 txt 파일이 저장되지 않는다.
MAIN SUCCESS SCENARIO	
step	Action
s	냉장고에 제품을 추가하고자 할 때 시작한다.
1	메뉴에서 1을 입력한다.
2	등록할 제품의 바코드를 입력한다.
3	등록할 제품의 제품명을 입력한다.
4	등록할 제품의 영양 정보를 입력한다.
5	제품의 txt 파일을 생성한다.
EXTENSION SCENARIOS	
step	Branching Action
2	2a. 바코드에 숫자가 아닌 문자가 들어가는 경우 2a.1. 잘못 입력했다는 메시지를 띄우고, 메뉴 화면으로 돌아간다. 2b. 바코드 입력을 하지 않고 넘어가려는 경우 2b.1. 바코드를 입력하라는 메시지를 띄우고, 메뉴 화면으로 돌아간다.
3	3a. 제품명 입력을 하지 않고 넘어가려는 경우 3a.1. 제품명을 입력하라는 메시지를 띄우고, 메뉴 화면으로 돌아간다.
4	4a. 영양 정보 입력을 하지 않고 넘어가려는 경우 4a.1. 영양 정보를 입력하라는 메시지를 띄우고, 메뉴 화면으로 돌아간다. 4b. 영양 정보를 덜 입력한 경우 4b.1. 기본값을 0으로 두고 자동으로 채워지게 한다. 4b.2. 다 입력하라는 메시지를 출력하고 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds

Frequency	Variable
Concurrency	None
Other	



Use Case #3 : Edit Nutrients	
GENERAL CHARACTERISTICS	
Summary	기존 제품의 영양 정보를 수정한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	시스템이 동작 중이어야 한다.
Trigger	메뉴에서 edit nutrients를 선택한다.
Success Post Condition	Food 폴더 속의 제품 txt 파일의 영양 정보가 수정된다.
Failed Post Condition	제품 txt 파일이 수정되지 않는다.
MAIN SUCCESS SCENARIO	
step	Action
s	냉장고에 저장된 제품의 영양 정보를 수정하고자 할 때 시작한다.
1	메뉴에서 2를 입력한다.
2	수정하고자 하는 제품의 바코드를 입력한다.
3	제품이 맞는지 확인 후 제품명, 영양 정보를 입력한다.
4	제품의 txt 파일을 수정한다.
EXTENSION SCENARIOS	
step	Branching Action
2	2a. 해당하는 제품이 없을 경우 2a.1. 해당하는 제품이 없다는 메시지를 출력 후, 메뉴 화면으로 돌아간다. 2b. 잘못 입력한 경우 2b.1. 해당하는 제품이 없다는 메시지를 출력 후, 메뉴 화면으로 돌아간다.
3	3a. 정보를 덜 입력하는 경우 3a. 모든 정보를 다 입력하라는 메시지를 출력 후, 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #4 : Search Products	
GENERAL CHARACTERISTICS	
Summary	입력받은 제품이 있는지 검색한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	시스템이 동작 중이어야 한다.
Trigger	메뉴에서 search products를 선택한다.
Success Post Condition	Food 폴더 속의 제품 txt 파일을 찾아서 영양 정보를 표시한다.
Failed Post Condition	제품을 찾는 것에 실패해서 영양 정보를 표시하지 않는다.
MAIN SUCCESS SCENARIO	
step	Action
s	냉장고에 입력된 제품을 저장하고자 할 때 시작된다.
1	메뉴에서 3을 입력한다.
2	검색할 제품의 바코드를 입력한다.
3	제품을 확인 후, 정보 수정 여부를 결정한다.
EXTENSION SCENARIOS	
step	Branching Action
2	2a. 해당하는 제품이 없을 경우 2a.1. 해당하는 제품이 없다는 메시지를 출력 후, 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #5 : Choose Person	
GENERAL CHARACTERISTICS	
Summary	사용자 프로필을 관리한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	시스템이 동작 중이어야 한다.
Trigger	메뉴에서 choose person을 선택한다.
Success Post Condition	choose person 클래스가 할 수 있는 기능을 표시한다.
Failed Post Condition	choose person 클래스가 할 수 있는 기능을 표시하지 못한다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자 프로필을 선택, 추가하고자 할 때 시작된다.
1	메뉴에서 4를 입력한다.
2	사용자 프로필을 선택하거나, 추가, 또는 이전 화면으로 되돌아간다.
EXTENSION SCENARIOS	
step	Branching Action
2	2a. 프로필을 삭제하고자 하는 경우 2a.1. (삭제 기능 추가 예정)
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #6 : Enter Physical Information	
GENERAL CHARACTERISTICS	
Summary	User에게 신체 정보를 입력하도록 요청한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	choose person 클래스가 실행 중이어야 한다.
Trigger	새로 만들기를 선택한다.
Success Post Condition	새로 만든 프로필의 신체 정보를 입력 받는다.
Failed Post Condition	새로 만든 프로필의 신체 정보를 입력 못 받는다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자가 프로필을 만들 때 시작된다.
1	Choose Person을 선택한 후, 2번을 입력한다.
2	프로필 이름을 입력한다.
3	키, 몸무게, 골격근량, 체지방률을 입력한다.
EXTENSION SCENARIOS	
step	Branching Action
3	3a. 신체 정보 입력을 하지 않고 넘어가려는 경우 3a.1. 신체 정보를 입력하라는 메시지를 띄우고, 메뉴 화면으로 돌아간다. 3b. 신체 정보를 덜 입력한 경우 3b.1. 다 입력하라는 메시지를 출력하고 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #7 : Edit Physical Information	
GENERAL CHARACTERISTICS	
Summary	입력받은 신체 정보를 수정하게 한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	프로필이 하나 이상 만들어져있어야 한다.
Trigger	프로필을 불러온 후 프로필 수정 중 신체 정보 변경을 입력한다.
Success Post Condition	입력된 신체 정보를 수정한다.
Failed Post Condition	입력된 신체 정보 수정에 실패한다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자가 자신의 프로필의 신체 정보를 수정할 때 시작된다.
1	프로필을 선택한다.
2	메뉴에서 4번을 입력 후 신체 정보 수정을 선택한다.
3	신체 정보를 수정한다.
4	프로필 txt 파일을 수정한다.
EXTENSION SCENARIOS	
step	Branching Action
3	3a. 신체 정보를 덜 입력한 경우 3a.1. 다 입력하라는 메시지를 출력하고 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #8 : Enter Activity Information	
GENERAL CHARACTERISTICS	
Summary	User에게 활동 정보를 입력하도록 요청한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	신체 정보를 입력 받은 직후
Trigger	신체 정보 입력 후 자동 입력
Success Post Condition	활동 정보를 입력 받는다.
Failed Post Condition	활동 정보 입력에 실패한다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자가 프로필을 만들고 신체 정보 입력이 끝난 뒤 시작된다.
1	주간 운동 횟수와 운동 시간을 입력한다.
EXTENSION SCENARIOS	
step	Branching Action
1	1a. 활동 정보 입력을 하지 않고 넘어가려는 경우 1a.1. 활동 정보를 입력하라는 메시지를 띄우고, 메뉴 화면으로 돌아간다. 1b. 활동 정보를 덜 입력한 경우 1b.1. 기본값을 0으로 두고 자동으로 채워지게 한다. 1b.2. 다 입력하라는 메시지를 출력하고 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #9 : Edit Activity Information	
GENERAL CHARACTERISTICS	
Summary	입력받은 활동 정보를 수정하게 한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	프로필이 하나 이상 만들어져있어야 한다.
Trigger	프로필을 불러온 후 프로필 수정 중 활동 정보 변경을 입력한다.
Success Post Condition	입력된 활동 정보를 수정한다.
Failed Post Condition	입력된 활동 정보 수정에 실패한다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자가 자신의 프로필의 활동 정보를 수정할 때 시작된다.
1	프로필을 선택한다.
2	메뉴에서 4번을 입력 후 활동 정보 수정을 선택한다.
3	활동 정보를 수정한다.
4	프로필 txt 파일을 수정한다.
EXTENSION SCENARIOS	
step	Branching Action
3	3a. 활동 정보를 덜 입력한 경우 3a.1. 다 입력하라는 메시지를 출력하고 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #10 : Set Diet Balance	
GENERAL CHARACTERISTICS	
Summary	User에게 식단 균형을 설정하도록 요청한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	활동 정보를 입력 받은 직후
Trigger	활동 정보 입력 후 자동 입력
Success Post Condition	식사 밸런스 정보를 입력 받는다.
Failed Post Condition	식사 밸런스 정보 입력에 실패한다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자가 프로필을 만들고 신체, 활동 정보 입력이 끝난 뒤 시작된다.
1	식사 밸런스 정보를 선택한다.
2	프로필 txt 파일을 생성한다.
EXTENSION SCENARIOS	
step	Branching Action
1	1a. 식사 밸런스 정보 입력을 하지 않고 넘어가려는 경우 1a.1. 식사 밸런스 정보를 입력하라는 메시지를 띄우고, 메뉴 화면으로 돌아간다. 1b. 식사 밸런스 정보를 덜 입력한 경우 1b.1. 기본값을 0으로 두고 자동으로 채워지게 한다. 1b.2. 다 입력하라는 메시지를 출력하고 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	



Use Case #11 : Edit Diet Balance	
GENERAL CHARACTERISTICS	
Summary	입력받은 식단 균형을 수정하게 한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	프로필이 하나 이상 만들어져있어야 한다.
Trigger	프로필을 불러온 후 프로필 수정 중 식사 밸런스 정보 변경을 입력한다.
Success Post Condition	입력된 식사 밸런스 정보를 수정한다.
Failed Post Condition	입력된 식사 밸런스 정보 수정에 실패한다.
MAIN SUCCESS SCENARIO	
step	Action
s	사용자가 프로필의 식사 밸런스 정보를 수정할 때 시작된다.
1	프로필을 선택한다.
2	메뉴에서 4번을 입력 후 식사 밸런스 정보 수정을 선택한다.
3	식사 밸런스 정보를 수정한다.
4	프로필 txt 파일을 수정한다.
EXTENSION SCENARIOS	
step	Branching Action
3	3a. 식사 밸런스 정보를 덜 입력한 경우 3a.1. 다 입력하라는 메시지를 출력하고 메뉴 화면으로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #12 : Food Modification for each Partition	
GENERAL CHARACTERISTICS	
Summary	특정 사용자 프로필에 대한 식사 정보를 입력/삭제하게 하고 요청 시 저장된 모든 식사 정보를 표시한다.
Scope / Level	Refrigerator with U / User Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	User
Preconditions	프로필이 하나 이상 만들어져있어야 한다.
Trigger	프로필을 선택한다.
Success Post Condition	아침, 점심, 저녁별로 식사 입력/삭제가 가능해진다.
Failed Post Condition	아침, 점심, 저녁별로 식사 입력/삭제가 불가능해진다.
MAIN SUCCESS SCENARIO	
step	Action
s	프로필에서 식사 정보를 입력/삭제하고자 할 때 시작된다.
1	프로필을 선택한다.
2	아침, 점심, 저녁 중 식사를 추가하고 싶은 시간의 메뉴를 선택한다.
3	냉장고에 있는 제품의 바코드를 입력한다.
4	먹은 식사를 저장한다.
EXTENSION SCENARIOS	
step	Branching Action
3	3a. 냉장고에 제품이 저장되어 있지 않은 경우 3a.1. 제품이 없습니다. 제품을 추가하겠습니까?처럼 추가 여부를 묻는 메시지를 출력한다. 3b. 바코드를 잘못 입력한 경우 3b.1. 잘못 입력했다는 메시지를 출력하고 이전 메뉴로 돌아간다.
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #13 : Save Nutrients	
GENERAL CHARACTERISTICS	
Summary	입력받은 영양 정보들을 Data로 저장한다.
Scope / Level	Refrigerator with U / Program Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	Program
Preconditions	시스템이 동작 중이어야 한다.
Trigger	제품의 영양 정보를 입력 또는 수정할 때
Success Post Condition	정상적으로 제품의 영양 정보가 저장된다.
Failed Post Condition	제품의 영양 정보 저장에 실패한다.
MAIN SUCCESS SCENARIO	
step	Action
s	제품의 영양 정보 추가/수정을 하고자 할 때 시작된다.
1	제품의 영양 정보 추가/수정을 하면 자동으로 txt 파일에 저장한다.
EXTENSION SCENARIOS	
step	Branching Action
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

Use Case #14 : Save Each Person's Information	
GENERAL CHARACTERISTICS	
Summary	입력받은 각 프로필의 정보들을 Data로 저장한다.
Scope / Level	Refrigerator with U / Program Level
Author	
Last Update	28/04/2024
Status	Analysis
Primary Actor	Program
Preconditions	시스템이 동작 중이어야 한다.
Trigger	프로필의 신체 정보를 입력 또는 수정할 때
Success Post Condition	정상적으로 프로필의 신체 정보가 저장된다.
Failed Post Condition	프로필의 신체 정보 저장에 실패한다.
MAIN SUCCESS SCENARIO	
step	Action
s	프로필을 생성/수정 하고자 할 때 시작된다.
1	프로필을 추가/수정을 하면 자동으로 txt 파일에 저장한다.
EXTENSION SCENARIOS	
step	Branching Action
RELATED INFORMATION	
Performance	< 3 Seconds
Frequency	Variable
Concurrency	None
Other	

### 3. Domain analysis

#### 1) Power

시스템 전원 제어 클래스

#### 2) AddProducts

시스템에 식품 정보를 추가하는 클래스

#### 3) EditNutrients

저장된 식품의 영양 정보를 수정하는 클래스

#### 4) SearchProducts

시스템에 저장된 식품을 검색하는 클래스

#### 5) ChoosePerson

사용자 프로필을 선택 및 관리하는 클래스

#### 6) FoodModificationForEachPartition

특정 사용자 프로필에 대한 식사 정보를 수정하는 클래스

#### 7) EnterPhysicalInformation

사용자의 신체 정보를 입력하는 클래스

#### 8) EditPhysicalInformation

입력받은 사용자의 신체 정보를 수정하는 클래스

#### 9) EnterActivityInformation

사용자의 활동 정보를 입력하는 클래스

#### 10) EditActivityInformation

입력받은 사용자의 활동 정보를 수정하는 클래스

#### 11) SetDietBalance

사용자의 식단 균형을 설정하는 클래스

#### 12) EditDietBalance

입력받은 사용자의 식단 균형을 수정하는 클래스

#### 4. User Interface prototype

```
프로그램을 실행하려면 1을, 종료하려면 0을 입력하세요.0
프로그램이 종료되었습니다.
```

```
프로그램을 실행하려면 1을, 종료하려면 0을 입력하세요.1
프로그램이 실행되었습니다.
```

```
=====메뉴를 선택하세요=====
```

1. Add Products
2. Edit Nutrients
3. Search Products
4. Choose Person
5. Exit

F5를 눌러 실행을 시키면 실행하려면 1, 종료하려면 0을 입력하는데 0을 누르면 종료, 1을 누르면 실행이 된다.

```
=====메뉴를 선택하세요=====
```

1. Add Products
2. Edit Nutrients
3. Search Products
4. Choose Person
5. Exit

```
1
바코드를 입력하세요:1234
제품명을 입력하세요:sample
탄수화물, 단백질, 지방, 나트륨의 값을 순서대로 입력하세요 (띄어쓰기로 구분):100 20 10 400
상품 정보가 성공적으로 저장되었습니다.
```

실행 후 메뉴에서 1을 입력하면 등록할 제품의 바코드, 제품명, 영양성분을 입력 할 수 있다. 입력 후 프로그램이 txt파일을 Food폴더에 자동적으로 저장한다. 이때 Food폴더가 없으면 생성 후 안에 저장한다.

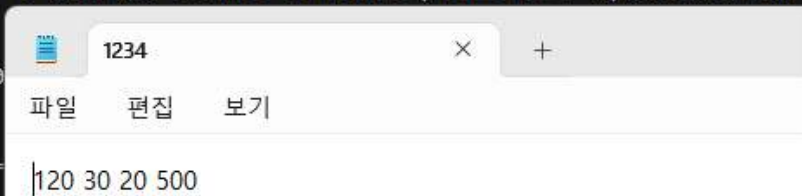
```
=====메뉴를 선택하세요=====
```

1. Add Products
2. Edit Nutrients
3. Search Products
4. Choose Person
5. Exit

```
2
수정하고자 하는 제품의 바코드를 입력하세요:1234
현재 상품 정보: 제품명: sample, 탄수화물: 100, 단백질: 20, 지방: 10, 나트륨: 400
새로운 제품명, 탄수화물, 단백질, 지방, 나트륨의 값을 순서대로 입력하세요 (띄어쓰기로 구분):120 30 20 500
상품 정보가 성공적으로 수정되었습니다.
```

```
프로그램을 실행하려면 1을, 종료하려면 0을 입력하세요.0
프로그램이 실행되었습니다.
```

```
=====메뉴를 선택하세요=====
```



메뉴에서 2번을 입력하면 수정하고자 하는 제품의 바코드를 입력할 수 있게 한다. Food폴더에 해당하는 바코드가 있다면 현재 상품 정보를 출력해주고, 없다면 출력하지 않고 되돌아간다. 입력해준 값으로 정상적인 수정이 완료된 사진이다.

```
=====메뉴를 선택하세요=====
1. Add Products
2. Edit Nutrients
3. Search Products
4. Choose Person
5. Exit
3
검색할 상품의 바코드를 입력하세요:1234
제품명: 120 30 20 500
수정하시겠습니까? (yes/no):yes
```

3번을 입력한다면 검색할 상품의 바코드를 입력받는데, 이때 상품이 있다면 제품명과 영양성분이 출력되고 수정할 것인지를 물어본다. 없다면 상품이 없다는 말을 출력하고 이전으로 되돌아간다. 사진의 경우는 제품명이 정상적으로 출력되지 않는 모습이다.

```
4
=====ChoosePerson=====
1. 불러오기
2. 새로 만들기
3. 취소하기
선택하세요:[]
```

4번을 입력한다면 1번을 눌러 저장된 프로필 txt파일을 불러올 수 있다. 2번을 누르면 새로운 프로필을 만들 수 있다. 3번을 누르면 이전 단계로 되돌아간다.

```

=====ChoosePerson=====
1. 불러오기
2. 새로 만들기
3. 취소하기
선택하세요:2
프로필 이름을 입력하세요:sample
키, 몸무게, 골격근량, 체지방률을 순서대로 입력하세요 (띄어쓰기로 구분):200 150 120 4
주당 운동 횟수와 운동 시간을 순서대로 입력하세요 (띄어쓰기로 구분):7 7

=====SetDietBalance=====
1. 근육성장
2. 유지
3. 다이어트
식단 균형을 선택하세요:1
프로필 정보가 성공적으로 저장되었습니다.
  
```



2번을 눌러 profile폴더에 저장된 프로필 txt파일을 생성해보자. profile폴더가 없다면 자동적으로 생성된다. 프로필 이름(파일명)을 입력하고 키, 몸무게, 골격근량, 체지방률을 입력받는다. 입력받은 후 주당 운동 횟수와 운동 시간을 순서대로 입력받는다. 사진은 신체정보와 활동정보, 식단 균형이 띄어쓰기로 잘 저장된 모습이다.

```

=====ChoosePerson=====
1. 불러오기
2. 새로 만들기
3. 취소하기
선택하세요:1
1. sample
2. 이름
프로필을 선택하세요:1
현재 프로필 정보: 키 200, 몸무게 150, 골격근량 120, 체지방률 4, 주당 운동 횟수 7, 운동 시간 7, 식단 균형 근육성장

=====profile=====
1. 아침 식사
2. 점심 식사
3. 저녁 식사
4. 프로필 수정
5. 취소
선택하세요:[]
  
```

1번을 누르면 profile폴더에 저장된 sample.txt(예시)를 선택 할 수 있게 된다. 프로필을 선택하면 생성할 때 입력했던 프로필 정보가 표시되고 그 프로필 파일로 어떤 것을 할 것인지 표시가 된다.

아침, 점심, 저녁 각각 먹은 식사를 저장할 수 있고 저장을 하면 총 영양소별 총 섭취량이 표시된다. 신체의 변화, 활동량의 변화로 인해 프로필을 수정해야 할 일이 있으면 4번을 입력해서 수정이 가능하다. 코드의 예러로 동작하는 사진은 넣지 못했다. 추후 업데이트가 되면 수정할 예정이다.



## 5. Glossary

Term	Description
영양 정보	탄수화물, 단백질, 지방, 나트륨을 표현하는 말
신체 정보	키, 몸무게, 골격근량, 체지방률을 표현하는 말
활동 정보	주당 운동 횟수, 운동 시간을 표현하는 말
식단 균형(식사 밸런스)	근성장, 유지, 다이어트처럼 식사 방향성을 표현하는 말

## 6. References

-강의 자료: 04\_Use case diagram I, 05\_Use case diagram II

-참고 자료: Always be there, 있고 없고, 전 세계의 음식들, Market Nearby You, Lost & Found App, Real Time Location System