

**Diet In Body Application**

**과목명 소프트웨어공학**

**담당교수님 김상철교수님**

**201904458 이준용**

**202001574 박찬근**

**201901940 송경민**

목차

1. 개요

1.1. 시스템의 개요

1.2. 범위

1.3. 정의, 약어

2. 기능적 요구

2.1. 외부 인터페이스 요구

2.1.1. 사용자 인터페이스

2.1.2. 하드웨어 인터페이스

2.1.3. 소프트웨어 및 통신 인터페이스

2.2. 요구 기능

2.2.1. 기능 #1(식단 추천 기능 #1)

2.2.2. 기능 #2(음식 검색 및 등록 #2)

2.2.3. 기능 #3(다이어리 기능 #3)

2.2.4. 기능 #4(영양정보 확인 #4)

2.2.5. 기능 #5(마이페이지 기능 #5)

2.2.6. 기능 #6(후기추천평가 기능#6)

2.2.7. 기능 #7(식단 피드백 기능#7)

3. 기타 요구 및 제약 사항

3.1. 성능 요구

3.2. H/W 요구

3.3 예외 조건 및 이의 처리

3.4 자원, 인력에 대한 제약 조건

4. 인수 조건

4.1. 기능 시험 및 성능 시험

5. 참고 자료

**1. 개요**

**1.1. 시스템 개요**

|  |
| --- |
| 올바른 식생활은 우리의 건강한 삶을 위해 필수적인 요소입니다. 올바른 식생활은 우리의 신체적, 정신적 건강을 유지하는 데 도움을 주며, 영양 정보들은 우리가 올바른 식단을 선택하고 필요한 영양소를 충분히 섭취할 수 있도록 도와줍니다. 우리는 식단 다이어리 작성, 영양 정보 라벨 확인 등의 실천 방법을 통해 건강한 식생활을 실천할 수 있지만 여러 현실적인 이유로 혼자 지속하기 힘든 경우가 많습니다. 그래서 이를 좀 더 쉽게 할 수 있도록 도움을 드릴 수 있는 어플을 만들고자 저희는 “Diet In Body Application”을 계획하게 되었습니다. 우리가 삶에서 쉽게 볼 수 있는 많은 음식의 영양 정보들이 데이터화 되어 있어 각 음식의 영양정보와 하루 섭취한 칼로리 및 영양소를 매일매일 쉽게 체크할 수 있고, 맞춤 식단을 추천해주는 기능도 있으며 체중 추적 기능을 통해 체중변화를 확인해 볼 수 있습니다. 또한, 달마다 제공하는 Body 리포트를 받아보며 그에 따라 목표 설정할 수 있도록 하는 기능도 포함시킬 계획입니다. |

**1.2. 범위 목적**

|  |
| --- |
| 이 프로젝트의 범위는 웹 크롤링을 통해 식단을 작성하여 직접 고객들에게 지원하는 반응형 웹 기반 시스텝입니다. 제품 광고, 재고 관리 및 계정 청구는 이 프로젝트의 일부가 아닙니다.  또한 현재 데이터베이스의 논리적 및 물리적 설계에 변경이 예상됩니다. 새로운 데이터베이스 시스템의 실제 구현은 이 프로젝트의 일부가 아닙니다. 웹 검색 엔진과 언어 번역기는 새로운 시스템을 위해 구매된 구성 요소로 얻을 것입니다. 그들의 내부 세부 사항은 이 프로젝트의 일부가 아닙니다. 웹 사이트 보안 문제는 사이트 내에서의 암호 보호를 제외하고 이 프로젝트의 일부가 아닙니다. |

**1.3. 정의, 약어**

|  |
| --- |
| -Diet In Body Application : 우리가 개발할 어플리케이션 이름. 사용자들이 다이어리 작성, 영양 정보 라벨 확인 등의 실천 방법을 통해 건강한 식생활을 실천할 수 있도록 도와준다.  -Body 리포트 : 한달 주기로 사용자들의 식단에 따른 체중변화에 대한 보고서.  -웹 크롤링 : 컴퓨터 소프트웨어 기술로 웹 사이트들에서 원하는 정보를 추출하는 것을 의미한다. |

**2. 기능적 요구**

**2.1. 외부 인터페이스 요구**

|  |  |  |
| --- | --- | --- |
| **인터페이스 요구사항 ID** | **명칭** | **설명** |
| **IR –001** | **DB 연결** | **DB 서버와 웹을 연결한다.** |
| **IR –002** | **영양정보 데이터베이스** | **영양정보 저장 및 관리, 외부데이터베이스 API로 가져옵니다.** |
| **IR –003** | **사용자 계정 및 데이터 동기화** | **개인 식단 및 체중 추적을 위한 계정 생성 및 저장합니다.** |
| **IR –004** | **음식 데이터소스** | **음식 항목과 그에대한 영양정보를 실시간으로 가져오는 외부음식 데이터소스가 필요합니다.** |
| **IR –005** | **사용자 위치 기반 서비스** | **지리적 위치를 기반으로 지역 특화된 식단 추천을 제공합니다.** |
| **IR –006** | **보안 및 개인 정보 보호** | **사용자 데이터는 안전하게 보호되어야합니다. 암호화 서비스가 필요합니다.** |

**2.1.1. 사용자 인터페이스**

|  |
| --- |
| 텍스트, 소프트웨어, 스크린샷, 디자인이(가) 표시된 사진  자동 생성된 설명  **그림 1. 로그인 페이지**  텍스트, 번호, 폰트, 소프트웨어이(가) 표시된 사진  자동 생성된 설명  **그림 2. 회원가입 페이지 및 필수 작성 항목에 빈칸으로 회원가입 버튼 누를 때 에러 화면**  텍스트, 스크린샷, 소프트웨어, 컴퓨터 아이콘이(가) 표시된 사진  자동 생성된 설명텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진  자동 생성된 설명  **그림 3. (a) 로그인 후 메인 대시 보드 페이지 (b) 좌측 상단 버튼 눌렀을 때 사이드바 오버레이 된 화면**  텍스트, 스크린샷, 포스트잇 노트, 다채로움이(가) 표시된 사진  자동 생성된 설명텍스트, 스크린샷, 번호, 디스플레이이(가) 표시된 사진  자동 생성된 설명  **그림 4. (a) 메인 대시 보드 페이지에서 상단 카테고리로 검색 버튼 누른 후 화면 (b) 사이드 바에서 마이 페이지를 누른 후 마이페이지로 이동한 화면**  텍스트, 웹 페이지, 웹사이트, 스크린샷이(가) 표시된 사진  자동 생성된 설명  **그림 5. 메인 대시 보드 페이지와 마이 페이지에서 검색한 내용에서 식단 클릭 시 상세 페이지로 이동** |

**2.1.2. 하드웨어 인터페이스**

|  |
| --- |
| “Diet In Body Application” 은 웹 크롤링을 통해 식단을 작성하여 직접 고객들에게 지원하는 반응형 웹 기반 시스템입니다. |

**2.1.3. 소프트웨어 및 통신 인터페이스**

|  |
| --- |
| -사용자 인터페이스 (UI): 웹은 사용자 친화적이고 직관적인 사용자 인터페이스를 제공해야 합니다. 이 인터페이스는 사용자가 음식 데이터를 입력하고 식단을 관리하며, 체중 추적과 리포트 확인을 쉽게 할 수 있도록 설계되어야 합니다.    -음식 데이터베이스 및 API: 웹은 음식의 영양 정보를 저장하고 업데이트해야 합니다. 음식 데이터베이스는 다양한 음식 항목과 그에 대한 영양 정보를 포함해야 하며, API를 통해 새로운 음식 정보를 가져올 수 있어야 합니다.    -사용자 계정 및 관리: 사용자는 개인 계정을 생성하고 사용자 정보를 관리하므로 이를 통해 사용자의 식단과 체중 데이터를 저장하고 동기화할 수 있습니다.    -체중 추적 장치 및 센서: 웹은 사용자의 체중을 추적하려면 체중 측정 장치와 연동해야하기 때문에 이를 위해 Bluetooth 또는 USB 연결을 지원할 수 있어야 합니다.    -데이터 동기화 및 백업: 사용자의 데이터는 안전하게 서버에 동기화되고 백업되어야 하므로 사용자가 여러 기기 또는 플랫폼을 사용할 때 데이터의 일관성을 유지해야 합니다.    -알고리즘 및 추천 엔진: 웹은 사용자의 목표와 식단 정보를 기반으로 맞춤 식단을 추천하는 알고리즘을 포함해야 합니다. 이를 위해 기계학습 및 데이터 분석 기술을 사용할 수 있어야 합니다.    -체중 변화 추적 및 리포트: 웹은 사용자의 체중 변화를 추적하고 매월 Body 리포트를 생성해야 합니다. 이 리포트는 사용자에게 목표 설정과 진전 상황에 대한 정보를 제공해야 합니다.    -푸시 알림 및 알람: 사용자에게 식단 입력, 목표 달성, 리포트 확인 등의 중요 이벤트에 대한 푸시 알림을 제공할 수 있어야 하며, 필요한 경우 사용자가 설정한 알람을 지원해야 합니다.    -보안 및 개인 정보 보호: 사용자 데이터는 암호화되어 안전하게 저장되어야 하며, GDPR 및 기타 개인 정보 보호 규정을 준수해야 합니다.    -다양한 플랫폼 지원: 웹은 주요 모바일 플랫폼 (iOS, Android)를 지원해야 하며, 필요한 경우 웹 버전도 고려해야 합니다.    -외부 서비스 통합: 사용자에게 편리한 서비스를 제공하기 위해 위치 기반 서비스, 음식 주문 및 배달 서비스, 건강 관련 정보 서비스 등을 통합할 수 있는 API를 활용해야 합니다.    -사용자 피드백 및 지원: 웹은 사용자 피드백을 수집하고 지원 서비스를 제공할 수 있어야 합니다. 사용자의 문의나 의견에 신속하게 응답할 수 있는 기능이 필요합니다. |

**2.2 기능요구**

텍스트, 스크린샷, 도표, 라인이(가) 표시된 사진

자동 생성된 설명

**2.2.1. 기능 #1(식단 추천 기능)**

|  |  |
| --- | --- |
| 구분 | 내용 |
| 요구사항명 | 식단 추천 기능 |
| 액터명 | 사용자 |
| 개요 및 설명 | 사용자는 대시보드에서 식단을 선택하면 여러 옵션을 선택 후 식단 추천을 받을 수 있다. |
| 사전 조건 | 식단 추천 버튼 누르기 전 상태여야 한다. |
| 기본 흐름 | 1. (액터) 사용자가 식단 추천 버튼을 누른다. 2. (시스템) 사용자가 검색한 식단이 서버에 전송되고 추천 시스템을 통해 다시 액터에게 전송된다. 3. (시스템) 사용자에게 추천 식단을 보여준다. 4. (액터) 추천 식단을 확인할 수 있고, 추천된 식단을 저장할 수 있다. 5. (시스템) 추천된 식단에 대한 정보를 회원의 마이페이지에 전송하고 저장한다. |
| 예외 흐름 | E1. 식단을 선택하지 않고 추천버튼 누를경우  식단을 클릭해주세요라는 메시지를 띄우고 다시 요청한다. |
| 대안 흐름 |  |
| 사후조건 | 사용자가 추천받은 식단을 저장할 경우 해당 추천 식단이 유저 마이페이지 정보에 저장된다. |

**2.2.2. 기능 #2(음식 검색 및 등록)**

|  |  |
| --- | --- |
| 구분 | 내용 |
| 요구사항명 | 음식 검색 및 등록 |
| 액터명 | 관리자 |
| 개요 및 설명 | 액터는 음식을 검색하고 등록한다. |
| 사전 조건 | 관리자는 관리 권한을 가진 ID로 로그인한다. |
| 기본 흐름 | 1. (시스템) 음식 최상위 카테고리 목록을 관리자에게 보여준다.  2. 액터) 관리자가 조회하고자 하는 카테고리를 선택한다.  3. (시스템) 선택된 카테고리의 하위 카테고리 목록을 보여준다.  4. (액터) 등록할 음식 명과 영양정보를 입력하고 저장한다.  5. (시스템) 음식 등록 후 카테고리 목록을 갱신하여 보여준다. |
| 예외 흐름 | E1. 입력한 음식명이 이미 존재할 경우  E1-1 동일한 이름이 존재한다는 메시지 출력 |
| 대안 흐름 | - |
| 사후 조건 | 등록한 음식이 포함된 하위 카테고리 목록을 보여준다. |

**2.2.3. 기능 #3(다이어리 기능)**

|  |  |
| --- | --- |
| 구분 | 내용 |
| 요구사항명 | 다이어리 |
| 액터명 | 사용자 |
| 개요 및 설명 | 사용자는 마이페이지의 다이어리에서 오늘 먹은 음식과 영양정보를 등록 및 확인한다. |
| 사전 조건 | 사용자 아이디로 로그인한다. |
| 기본 흐름 | 1. (액터) 사용자가 마이페이지 버튼을 누른다. 2. (액터) 사용자가 다이어리 버튼을 누른다. 3. (시스템) 맨 위에 총 섭취 칼로리를 보여주고 아침, 점심, 저녁, 간식/기타 카테고리를 보여준다. 4. (액터) 사용자가 해당하는 카테고리로 들어가 음식을 검색 후 등록한다. 5. (시스템) 해당 카테고리에 등록된 음식의 칼로리와 맨 위에 총 섭취 칼로리를 보여준다. |
| 예외 흐름 | E1. 음식을 검색했는데 음식이 등록이 안 되어 있을 경우  E1-1 검색한 음식이 존재하지 않는다는 문구 출력한다  E1-2 음식 등록 요청 버튼을 보여준다 |
| 대안 흐름 | - |
| 사후 조건 | 다이어리에 총 섭취 칼로리, 아침, 점심, 저녁에 먹은 식단을 보여준다. |

**2.2.4. 기능 #4(영양정보 확인)**

|  |  |
| --- | --- |
| 구분 | 내용 |
| 요구사항명 | 영양정보 확인 |
| 액터명 | 사용자 |
| 개요 및 설명 | 사용자는 다이어리에서 오늘 먹은 음식의 각종 영양정보를 확인할 수 있다. |
| 사전 조건 | 사용자 아이디로 로그인한다. |
| 기본 흐름 | 1. (액터) 사용자가 마이페이지 -> 다이어리 -> 영양정보 확인 버튼을 누른다. 2. (시스템) 오늘 섭취한 총 칼로리, 각종 영양정보(지방, 콜레스트롤, 나트륨, 식이섬유, 탄수화물, 단백질, 당 등)을 보여준다. 3. (시스템) 오늘 섭취한 탄수화물, 단백질, 지방의 비율을 원형차트로 보여준다. |
| 예외 흐름 |  |
| 대안 흐름 |  |
| 사후조건 | 다이어리에서 오늘 섭취한 영양정보를 확인한다. |

**2.2.5. 기능 #5(마이페이지 기능)**

|  |  |
| --- | --- |
| 구분 | 내용 |
| 요구사항명 | 마이페이지 기능 |
| 액터명 | 사용자 |
| 개요 및 설명 | 사용자 마이페이지에서 저장한 추천 식단 열람, 삭제 및 수정 기능 |
| 사전 조건 | 사이드바에서 마이페이지 클릭 후 마이페이지로 이동해야 한다. |
| 기본 흐름 | 1. (액터) 사이드바에서 마이페이지 버튼 클릭 2. (시스템) 요청된 사항인 사용자의 DB에 접속하여 저장된 식단 내용 출력 3. (액터) 출력된 식단에 대해 열람, 삭제, 수정이 가능 4. (시스템) 삭제와 수정 요청이 있을 경우 해당 사항 삭제 및 수정 |
| 예외 흐름 | E1. 저장된 식단 개수가 0인데 개별 삭제가 아닌 전체 삭제 요청하는 경우  E1-1. 삭제할 식단이 없습니다 라는 메시지 출력 |
| 대안 흐름 |  |
| 사후조건 | 삭제 및 수정 된 식단의 메시지 출력한다. |

**2.2.6. 기능 #6(후기 추천 평가 기능)**

|  |  |
| --- | --- |
| 구분 | 내용 |
| 요구사항명 | 경험해 본 회원들이 그 식단에 대해서 후기나 추천 또는 평가할 수 있는 기능 |
| 액터명 | 사용자 |
| 개요 및 설명 | 사용자는 웹 내에서 자신이 시도한 식단에 대한 후기를 작성하거나 다른 사용자들에게 추천할 수 있다. |
| 사전 조건 | 웹에 로그인 되어 있어야 한다. |
| 기본 흐름 | 1.사용자는 웹 내의 “식단 후기 작성” 또는 “식단 추천” 섹션으로 이동한다.  2. 사용자는 작성하고자 하는 식단에 대한 후기를 작성하거나 추천 이유를 입력한다.  3. 사용자는 작성한 후기를 저장하고 웹 상에 공개하거나 비공개로 설정한다. |
| 예외 흐름 | 인증오류 또는 권한 오류 |
| 대안 흐름 | - |
| 사후조건 | 회원이 후기를 작성, 수정, 삭제한 경우 변경사항은 웹읲 데이터베이스에 반영되어 다른 사용자들의 열람 가능하며 추천을 작성한경우 확인하고 참고할 수 있다. |

**2.2.7. 기능 #7(유지칼로리 계산 식단 피드백 기능)**

|  |  |
| --- | --- |
| 구분 | 내용 |
| 요구사항명 | 정보 입력한 데이터로 유지칼로리 계산해서 오늘 식단 피드백 해주는  기능 |
| 액터명 | 시스템 또는 피드백 모듈 |
| 개요 및 설명 | 시스템은 사용자가 입력한 정보와 식단 데이터를 기반으로 해당 날짜의 유지칼로리르 계산하고 식단에 대한 피드백을 제공한다. |
| 사전 조건 | 사용자는 웹에 로그인 되어 있어야하며 개인프로필에 신체정보를 입력해야한다.  하루 섭취한 음식의 정보를 입력해야한다. |
| 기본 흐름 | 시스템은 사용자의 프로필 정보와 입력한 음식 데이터를 기반으로 해당 날짜의 유지칼로리를 계산한다.  계산된 유지칼로리와 사용자가 섭취한 칼로리를 비교하여 남은 칼로리 또는 초과한 칼로리를 계산한다.  시스템은 오늘의 식단에 대한 피드백을 제공한다. |
| 예외 흐름 | 프로필 정보 또는 음식 정보 미입력 |
| 대안 흐름 | - |
| 사후조건 | 시스템은 계산된 유지칼로리와 사용자가 섭취한 칼로리에 대한 피드백을 사용자에게 제공한다.  사용자는 피드백을 통해 더욱 신중하게 식단을 선택할 수 있다. |

**3. 기타 요구 및 제약 사항**

**3.1. 성능 요구**

|  |
| --- |
| • 반응시간 - 2초. 사용자의 요청에 따른 최소한의 반응 시간의 기준 |
| • 처리 소요 시간- 3초. 데이터베이스에서의 처리 시간을 기준 |
| • 처리율- 5건/시간 |

**3.2. H/W 요구**

|  |
| --- |
| * 장치 호환성   사용자는 다양한 장치(스마트폰, 태블릿, 랩톱, 데스크탑)에서 웹사이트에 접근할 수 있어야 합니다. 이를 위해 모든 장치에서 호환 가능한 반응형 디자인이 중요   * 스마트폰 및 태블릿   - 최신의 웹 브라우저(예: Chrome, Safari, Firefox)를 실행할 수 있는 운영 체제 (iOS, Android, Windows Phone)  - 최소 4인치 화면 크기, 권장 720p 이상의 해상도  - 터치스크린 기능을 포함한 기본 입력 장치   * 데스크탑 및 랩탑   - 최신의 웹 브라우저를 지원하는 운영 체제(예: Windows, macOS, Linux)  - 최소 1024x768 해상도, 권장 1080p 이상  - 키보드와 마우스 또는 트랙패드와 같은 표준 입력 장치   * 인터넷 연결   안정적인 인터넷 연결이 필수이며, 최소 3G/4G 연결을 권장합니다. 더 나은 사용자 경험을 위해 Wi-Fi 또는 고속 이더넷 연결을 사용하는 것이 좋습니다.   * 메모리 및 저장공간   웹사이트의 원활한 실행을 위해 최소 2GB의 RAM, 웹 브라우저 캐시 및 쿠키를 저장할 충분한 로컬 저장 공간.   * 프로세서   웹 크롤링과 데이터 처리에는 적절한 처리 능력이 요구됩니다. 중간 수준의 CPU가 권장되며, 모바일 장치의 경우 최신 프로세서를 사용하는 것이 좋습니다. |

**3.3 예외 조건 및 이의 처리**

|  |
| --- |
| • 입력 오류에 대한 시스템 처리 - 형식에 맞지 않는 자료를 입력하였을 경우 오류 메시지 표시하여 다 시 입력하도록 유도한다  • 오류로 인해 웹이 종료될 때 마지막 작업 상태로 복구한다.  • 보안 - 안전한 웹 애플리케이션 보안 구축을 통해 항상 안전한 상태를 유지한다.  - 외부 바이러스나 해킹 등과 같은 보안문제 처리를 위한 긴급처리센터를 구성한다.  • 웹에 대한 사용자의 요구사항과 개선할 점을 피드백 받기 위해 고객센터를 만들어 사용자와의 소통을 이어간다. |

**3.4 자원, 인력에 대한 제약 조건**

|  |
| --- |
| • 인력: 3명(팀장, 팀원)  • 대용량 이미지 및 텍스트 크롤링으로 인한 서버 과부하를 관리하는 문제  • 신속하게 정보가 전송되고 저장되는 백앤드 쿼리 처리 속도 개선 문제 및 화면 송출 속도 개선 문제 |

**4. 인수 조건**

**4.1. 기능 시험 및 성능 시험**

|  |
| --- |
| • 식단과 음식의 등록이 제대로 이루어지는지 확인한다  • 식단 추천과 음식 등록의 업데이트가 제대로 이루어지는지 확인한다  • 다이어리에서 음식을 등록했을 때 영양정보들이 제대로 보여지는지 확인한다  • 다이어트, 벌크업 등 각자의 목적에 맞게 식단을 설정하고 그에 맞춰서 식단을 진행했을 때 몸무게 및 골격근량의 예측치와 실제수치를 비교하고 피드백이 제대로 제공되는지 확인한다  • 사용자의 식단 추천 기능과 후기가 작성이 잘 되는지 확인한다  • 각각의 기능들이 오류 없이 수행되는지 시험한다 |

**5. 참고 자료**

|  |
| --- |
| 최은만 저, 새로 쓴 소프트웨어 공학, 정익사 출판  리액트 <https://ko.legacy.reactjs.org/> 사용자 인터페이스를 만들기 위한 JavaScript 라이브러리  리덕스<https://ko.redux.js.org/> 자바스크립트 앱을 위한 예측 가능한 상태 컨테이너  리코일 <https://recoiljs.org/ko/docs/introduction/getting-started/> React를 위한 상태 관리 라이브러리  셀레니움 <https://wikidocs.net/91474> 자바스크립트가 동적으로 만든 데이터를 크롤링 하기 위함 |