

요구사항 정의서 (Requirement Specification)

<목차>

[문서 정보](#)

[프로젝트 개요](#)

[이해관계자](#)

[기능 요구사항\(Functional Requirements\)](#)

[비기능 요구사항\(Non-Functional Requirements\)](#)

[데이터 저장 방식\(파일 기반\)](#)

[폴더 구조](#)

[배포 및 유지 보수](#)

[Use Case 다이어그램](#)

문서 정보

- 문서명 : 지출 관리 프로그램 요구사항 정의서
- 버전 : v1.0.0
- 작성일 : 2025-09-05
- 작성자 : 최마리
- 검토자 : 담당 강사님

프로젝트 개요

- 프로젝트명 : 한눈에 지출(Moo Money)
- 목표 : Java 콘솔 + 파일 처리 기반으로, 사용자가 자신의 지출 내역을 쉽고 빠르게 등록하고 관리할 수 있는 CLI 프로그램을 개발한다.
- 범위
 - 포함 : 회원 관리(가입 / 로그인 / 로그아웃), 영수증 관리(등록/조회), 지출 내역 보고서(월별 / 카테고리별)
 - 제외 : 프리미엄 회원의 결제 로직

이해관계자

구분	설명	역할
비회원	프로그램 접근 가능하지만 계정 없음	프로그램 소개 확인, 회원가입 진행
일반 회원	로그인 한 사용자	영수증 기반의 지출 관리(등록 / 조회)
프리미엄 회원	로그인한 사용자(유료 결제 회원), 추가 기능을 제공함	영수증 기반의 지출 관리(등록 / 조회) + 월별 전체 지출 보고서, + 월별 카테고리별 지출 보고서

기능 요구사항(Functional Requirements)

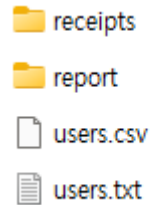
ID	요구사항	우선순위	수용 기준
REQ-01	사용자는 이메일/비밀번호로 회원 가입할 수 있다.	High	중복 이메일 불가, 비밀번호 암호화 저장
REQ-02	사용자는 로그인/로그아웃을 할 수 있다	High	올바른 ID/PW 입력 시 로그인 성공, 로그아웃 시 세션 종료
REQ-04	사용자는 영수증을 등록할 수 있다	High	날짜, 카테고리, 금액, 메모 입력 → txt, csv파일로 저장, 음수 금액 등록 불가
REQ-05	사용자는 본인의 영수증을 조회할 수 있다	High	날짜/카테고리 필터 가능, 모든 내역 출력
REQ-08	사용자는 월별/카테고리별 지출 보고서를 확인할 수 있다	High	총합 + 카테고리별 합계 출력

비기능 요구사항(Non-Functional Requirements)

ID	비기능 요구사항	설명	우선순위
NFR-02	보안	비밀번호 암호화 저장	High
NFR-03	안정성	사용자가 입력한 데이터에 대해 유효성을 검사, 실패 시 처리: 에러 메시지 출력 후 재입력 유도	High
NFR-04	사용성	콘솔 명령어 직관적, 페이지네이션 제공	Medium
NFR-05	확장성	csv/txt 외 향후 엑셀 확장 가능	Medium
NFR-06	유지보수성	팩토리 패턴, 추상 팩토리 패턴, 템플릿 메서드 적용	Medium

- 보안
 - 비밀번호 암호화 : SHA-256 해시 + 랜덤 Salt 적용 후 저장
 - CSV 파일에 직접 접근할 수 없도록 파일 권한 제한(읽기/쓰기 최소 권한) 설정
 - 회원 개인정보(이름/연락처)는 로그/보고서 출력 시 마스킹 처리
- 안정성
 - 사용자가 입력한 데이터에 대해 유효성을 검사, 실패 시 처리: 에러 메시지 출력 후 재입력 유도
 - 금액은 음수 불가, 날짜 형식은 yyyy-MM-dd, 문자열 길이 제한, 미래 날짜 입력 불가, 이메일 정규식으로 형식 체크
- 사용성
 - 콘솔 명령어는 직관적으로 설계
 - 데이터 조회 시 페이지네이션, 필터링(날짜/카테고리) 지원
 - 오류 메시지는 사용자 친화적인 안내 문구로 제공 (예: "날짜 형식이 잘못되었습니다. yyyy-MM-dd 형식으로 입력해주세요.")
- 확장성
 - 현재는 CSV/txt 파일 기반
 - 향후 Excel(.xlsx) 저장/읽기 기능 확장 가능하도록 I/O 모듈을 인터페이스로 분리
 - 데이터 접근 계층을 모듈화하여 DB 연동(MySQL, Oracle 등)으로 확장 가능
- 유지보수성
 - 팩토리 패턴, 추상 팩토리 패턴, 템플릿 메서드 적용

데이터 저장 방식(파일 기반)



- 회원 데이터
 - csv / txt
- 영수증 데이터
 - 월별로 폴더로 구분
 - csv/ txt
- 보고서
 - txt

폴더 구조

```
MooMoney/
├── data/                # 실행 중 생성/사용되는 데이터 파일
│   ├── users.csv
│   └── receipts.csv
├── src/
│   ├── main/
│   │   └── java/
│   │       └── io/github/choimari/moomoney/
│   │           ├── App.java        # 프로그램 진입점(main())
│   │           ├── model/          # User, Receipt 등
│   │           ├── dto/            # 컨트롤러 <-> 서비스 데이터 전달용
│   │           ├── repository/     # 파일 I/O 처리
│   │           ├── service/        # 업무 로직
│   │           ├── controller/     # 사용자 입력값 검증 + Service 호출
│   │           └── view/           # 출력 전담
```

```

├── util/                # 공용 기능 (InputReader)
├── test/
│   └── java/
│       └── io/github/choimari/moomoney/
└── lib/                # 필요한 외부 jar (JUnit, OpenCSV 등)

```

배포 및 유지 보수

- 배포: .jar 파일을 빌드하여 GitHub Release를 통해 배포
- 업데이트: GitHub Release 태깅(v1.0.0 형식)으로 버전 관리 **(계획없음)**
- 유지보수: GitHub Issues를 통한 버그 제보 및 개선 요청 접수 **(계획없음)**

Use Case 다이어그램

