학생 역량 관리 시스템

(Student Competency Management System)

프로젝트 수행 계획서

프로젝트명 : 학생 역량 관리 시스템 개발

Project Code : SCMS-2025-001

작성일 : 2025년 10월 27일

작성자 : 2팀 - 조수아

**프로젝트 개요**

**[현황 분석]**

대학 내 비교과 프로그램, 역량 개발, 진로 상담 등이 각 부서별로 산재되어 관리되고 있어 학생들의 역량 관리가 어려운 상황입니다.

**[문제점]**

* 비교과 프로그램 신청 절차 복잡 (수기 신청, 이메일 접수 등)
* 학생별 역량 진단 데이터 분산 관리
* 마일리지 적립 내역 실시간 확인 불가
* 상담 예약 시스템 부재로 학생 및 상담사 대기 시간 증가
* 취업 준비를 위한 포트폴리오 관리 기능 부족

**[해결 방안]**

통합 웹 플랫폼을 구축하여 학생- 교직원 간 원활한 소통과 체계적인 역량 관리를 지원합니다.

**[주요 목적]**

1. 비교과 프로그램 통합 관리 시스템 구축
2. 학생 역량 진단 및 분석 자동화
3. 마일리지 기반 역량 인증 체계 확립
4. 진로 설계 및 취업 지원 기능 제공
5. 통합 상담 예약 및 이력 관리 시스템 구현

**[정량적 목표]**

* 프로그램 신청 소요 시간 70% 단축 (10분 -> 3분)
* 역량 진단 참여율 50% 증가
* 상담 예약 대기 시간 60% 감소
* 시스템 동시 접속자 최소 100명 이상 지원

**[포함 범위]**

✅ 비교과 프로그램 관리 (등록, 조회, 신청, 참여 확인)

✅ 역량 진단 테스트 (설문 시스템, 결과 분석)

✅ 마일리지 관리 (적립, 조회, 역량 인증)

✅ 진로 설계 (포트폴리오, 채용공고 연동)

✅ 통합 상담 (예약, 승인, 상담 이력)

✅ 관리자 대시보드

✅ 반응형 웹 UI (PC/모바일)

**[기대효과]**

(학생 측면)

* 원스톱 역량 관리로 편의성 증대
* 체계적인 진로 설계 지원
* 마일리지 가시화를 통한 동기부여

(교직원 측면)

* 프로그램 운영 효율성 향상
* 학생 역량 데이터 통합 분석 가능
* 행정 업무 자동화로 업무 부담 감소

(대학 측면)

* 학생 역량 개발 성과 정량화
* 대학 평가 지표 개선
* 취업률 향상 기여

**팀 구성 및 역할 분담**

**[팀 구성]**

팀명 : 2팀

팀원 수 : 5명

개발 기간 : 4주 (기획 1주 + 개발 2주 + 테스트 1주)

| 이름 | 구분 | 담당 업무 | 상세 |
| --- | --- | --- | --- |
| 조수아 | 백엔드 | 비교과 프로그램 | - CRUD (등록, 조회, 수정, 삭제)  - 프로그램 신청/취소  - 참여 확인 처리  - 페이징 처리 |
| 알림 | 이메일 발송 | - 프로그램 신청 승인 시 알림  - 공통 이메일 모듈 (JavaMailSender) |
| 프론트엔드 | 비교과 화면 | - 프로그램 목록 화면  - 프로그램 상세 페이지  - 신청 화면  - 메인 대시보드 |
| 프로젝트 관리 | PM 업무 | - 일정 관리  - 회의 진행 |
| 공통 모듈 | 인증 및 보안 | - 로그인/회원가입  - JWT 인증 필터  - 공통 예외 처리  - 메인 대시보드  - 이메일 알림 공통 모듈  - 권한 관리  - URL 접근 제어 |
| 홍종학  역량 진단  +UI/UX  +보안문서 | 백엔드 | 역량 진단 시스템 | - 역량 진단 설문 CRUD  - 설문 응답 제출  - 역량 결과 분석/통계 |
| 프론트엔드 | 역량 진단 + 공통 UI | - 역량 진단 설문 화면  - 결과 분석 화면 (차트)  - 공통 헤더/푸터 CSS  - 에러 페이지 (404, 403, 500) |
| 추가 | ERD 및 보안 문서 | - ERD 작성 지원 (역량 테이블)  - XSS/SQL Injection 방지 문서 작성 |
| 지성현  마일리지  +DB 설계 | 백엔드 | 마일리지 시스템 | - 마일리지 적립 로직  - 마일리지 조회/이력  - 역량 인증 처리 |
| 프론트엔드 | 프론트엔드 | - 마일리지 조회 화면  - 적립 이력 화면  - 역량 인증 화면 |
| 추가 | DB 설계 및  샘플 데이터 생성 | - 전체 ERD 작성 총괄  - 테이블 정의서 작성  - MySQL DB 초기 설정(인덱스 설정\_학번, 프로그램 ID 등)  - 학생 더미 데이터 50, 비교과 프로그램 샘플 20, 역량 진단 문항 30개 |
| 황혜은  진로설계  +파일업로드  +외부API | 백엔드 | 진로 설계 시스템 | - 진로 탐색 설문  - 포트폴리오 CRUD  - 채용 공고 API 연동  - 포트폴리오 파일 업로드 및 다운로드 (보안 검증 포함) |
| 프론트엔드 | 진로 설계 화면 | - 진로 탐색 화면  - 포트폴리오 관리 화면  - 채용공고 목록 |
| 추가 | 파일 및 API 연동 | - 파일 업로드 공통 기능  - 외부 API 연동 문서 작성  - 파일 업로드 보안 가이드 (확장자 검증, 크기 제한 10MB) |
| 이서호  통합상담  +관리자 대시보드  +QA 총괄 | 백엔드 | 상담 + 관리자 기능 | - 상담 예약 CRUD  - 상담 승인/거부  - 상담 이력 조회 및 페이징  - 관리자 대시보드  - 사용자 관리 CRUD |
| 프론트엔드 | 상담 + 관리자 화면 | - 상담 예약 화면  - 상담 승인 화면 (관리자)  - 상담 이력 화면  - 관리자 대시보드  - 사용자 관리 화면 |
| 추가 | 품질 관리 총괄 | - 전체 QA 및 테스트  - API 문서화  - 버그 트래킹  - 사용자 매뉴얼 작성 (학생용/관리자용) |

**[공동 책임]**

* 일일 스크럼 참여 (매일 17:30)
* 코드 리뷰 (매주 토요일 09:00)
* 문서화 작업 (각자 담당 기능)
* 입력 폼 Validation (@Vaild)
* XSS 방지
* SQL Injection 방지

**개발 방법론 - 애자일(Agile) 스크럼**

**[선택 근거]**

1. 짧은 개발 기간에 적합한 반복적 개발
2. 요구사항 변경에 유연한 대응 가능
3. 소규모 팀(5명)에 최적화된 의사소통 구조
4. 주차별 결과물 확인을 통합 리스크 조기 발견

**[스프린트 계획]**

| sprint 0 (1주차) : 기획 및 설계   * 요구사항 정의 * ERD 설계 * 화면 설계 * 개발 환경 구축 |
| --- |

**[스크럼 이벤트 일정]**

(데일리 스크럼)

* 시간 : 매일 오후 17:30 (15분)
* 방식 : 대면
* 내용 : 오늘 한 일 / 내일 할 일 / 블로킹 이슈

(스프린트 계획 회의)

* 시간 : 매주 월요일 오후 17:00
* 방식 : 대면
* 내용 : 이번주 개발 목표 설정, 백로그 선택, 작업 분배

(스프린트 리뷰)

* 시간 : 매주 토요일 09:00
* 방식 : Discord 음성 채널 (비대면)
* 내용 : 개발 결과물 시연, 완료 기능 검증

(스프린트 회고)

* 시간 : 매주 토요일 10:00
* 방식 : Discord 음성 채널 (비대면)
* 내용 : KPT(Keep, Problem, Try)

**주요 일정 및 마일스톤**

**[전체 일정 개요]**

| 단계 | 기간 | 산출물 |
| --- | --- | --- |
| 기획 | 10/27 | 프로젝트 수행 계획서   * 프로젝트 개요 * 팀 구성 및 역할 분담 * 개발 방법론 선택 * 주요 일정 및 마일 스톤 * 리스크 관리 계획 * 커뮤니케이션 계획   팀 협업 문서   * Git 브랜치 전략 * 커밋 컨벤션 * 코드 리뷰 프로세스 * PR 규칙 * 코딩 컨벤션 * 일일 스크럼 시간 및 형식 * 산출물 저장 위치   Git 저장소 초기 설정 |
| 10/28 | 요구사항 정의서   * 기능별 요구사항 ID 정의 * 요구사항 분류 * 상세 설명 및 우선순위 선정 * 입력 데이터/출력 데이터 설정 * 비기능 요구사항   기능 분류 체계  User Story   * 페르소나 정의 |
| 10/29 | 시스템 아키텍처 문서   * 시스템 구성도 * 외부 API 연동 계획 * 배포 환경 * 보안 전략   기술 스택 선정 보고서  사이트맵  ERD  테이블 정의서 |
| 10/30 | 화면 설계서   * 주요 화면 별 와이어프레임 * 화면 ID 부여 * 화면 구성 요소 * 입력 항목 및 버튼 위치 * 반응형 여부 표시 * 필수 화면 목록 작성   화면 정의서   * 화면 ID, 화면명, URL 경로 * 화면 설정 및 접근 권한 * 입력 필드 목록 * 버튼 동작 정의 * 오류 메시지 정의   업무 프로세스 플로우차트   * 비교과 프로그램 신청 프로세스 * 역량 진단 프로세스 * 마일리지 적립 프로세스 * 상담 예약 및 승인 프로세스 * 예외 처리 흐름(정원 초과, 중복 신청 등)   API 명세서 초안   * 요청/응답 데이터 구조 * 상태 코드 * 인증방식 |
| 10/31 | WBS   * 작업 분해 구조 * 작업 항목별 담당자, 시작일, 종료일 작성 * 선후행 작업 관계 정의   일정표   * WBS 기반 시각적 일정표 * 마일 스톤 표시 * 주차별 목표 설정 * 팀원별 업무 부하 시각화   벤치마킹 보고서   * 유사 시스템 3개 이상 분석 * 주요 기능 비교표 * UI/UX 분석   기술 검토 보고서   * 성능 목표 * 보안 요구사항 * 외부 API 연동 가능성 조사 |
| 개발 | 11/01~11/14 (2주) | 소스 코드   * 단위 테스트 * API 문서 |
| 테스트/배포 | 11/15~ | 테스트 결과 보고서   * 배포된 시스템 * 사용자 매뉴얼 |

**리스크 관리 계획**

**[리스크 내용]**

| **리스크 ID** | **리스크 내용** | **확률** | **영향도** | **대응전략** | **담당자** |
| --- | --- | --- | --- | --- | --- |
| RISK-001 | - MySQL 연동 실패  - DB 접속 오류 | 중 | 상 | - 로컬 DB 테스트 선행  - 연결 풀 설정 문서화 | 지성현 |
| RISK-002 | - 요구사항 변경 빈번 | 상 | 중 | - 10/31까지 요구사항 동결  - 변경 시 변경 관리 절차  - 우선순위 재조정 | 조수아 |
| RISK-003 | - 팀원 개인 사정  (질병, 긴급상황) | 중 | 상 | - 페어 프로그래밍  - 코드 리뷰 철저  - 20% 버퍼 타임 확보 | 조수아 |
| RISK-004 | - 외부 API 연동 실패  (채용 공고 API) | 중 | 중 | - Mock 데이터 준비  - API 대체재 조사  - 최소 기능으로 축소 | 황혜은 |
| RISK-005 | - 성능 이슈  (100명 동시 접속 등) | 하 | 상 | - 페이징 처리  - 인덱스 최적화 | 전원 |
| RISK-006 | - Git 충돌 빈번 | 상 | 중 | - 작은 단위 커밋  - 매일 develop 동기화 | 전원 |
| RISK-007 | - 일정 지연 | 중 | 상 | - 주간 진척도 체크  - 핵심 기능 우선 개발 | 조수아 |
| RISK-008 | - 테스트 시간 부족 | 중 | 중 | - 개발과 테스트 병행  - 단위 테스트 작성  - 마지막 주 테스트 집중 | 이서호 |
| RISK-009 | - 데이터 손실 | 하 | 상 | - 정기적 백업 (매일)  - Git 버전 관리 활용 | 전원 |

**[리스크 모니터링]**

* 주기 : 매주 금요일 회고 시
* 담당 : 조수아
* 보고 : 리스크 관리 대장 업데이트

**[최악의 시나리오 대비]**

* 팀원 이탈 시 : 기능 축소
* 일정 심각하게 지연 시 : 필수 기능만 구현
* DB 완전 손실 시 : Git 백업 복구, 샘플 데이터 재생성

**커뮤니케이션 계획**

**[정기 회의]**

| 회의명 : 데일리 스크럼  주기 : 매일 (월~금)  시간 : 17:30  참석 : 전원 필수  목적 : 진행 상황 공유, 블로킹 이슈 해결 |
| --- |

| 회의명 : 스프린트 계획 회의  주기 : 매주 월요일  시간 : 17:00  참석 : 전원 필수  목적 : 주간 목표 설정, 백로그 선택 |
| --- |

| 회의명 : 스프린트 리뷰 & 회고  주기 : 매주 토요일  시간 : 09:00  참석 : 전원 필수  목적 : 완료 기능 시연, 개선점 도출 |
| --- |

**[의사소통 도구]**

| **도구** | **용도** | **응답 시간** |
| --- | --- | --- |
| GitHub | * 코드 리뷰 * 이슈 트래킹 * Wiki (회의록, 기술 문서) | 24시간 이내 |
| Discord | * 화면 공유 코딩 * 스프린트 리뷰 및 회고 | 즉시 |
| Kakao Talk | * 일일 스크럼 * 긴급 문의 | 즉시 |
| Google Drive | * 발표 자료 * ERD 파일 * 화면 설계서 |  |

**[보고 체계]**

(진행 상황 보고)

* 일일 : 데일리 스크럼 구두 보고
* 주간 금요일 스프린트 리뷰 (시연 및 코드 설명)
* 긴급 : 카카오톡을 통한 즉시 보고

(문서 버전 관리)

* 산출물 파일명 : 문서명\_v1.0\_날짜.확장자
* 예 : 요구사항정의서\_v1.2\_20251027.xlsx
* 최종본 : \_final 접미사 추가