**2306008 김명훈 프로젝트계획서**

**주제: 개선된 학사 관리 시스템**

**작성자: 2306008 김명훈**

**학년/반: 3-A**

**팀명: EMAS(Enhanced Academic Management System)**

**프로젝트 참여자: 2106117 김태연, 2306007 김수혁, 2306008 김명훈**

**제출일자: 2025년 4월 18일**

**목차**

|  |
| --- |
| **1 개요**   * 1. **프로젝트 개요**   2. **프로젝트 선정 동기**   3. **프로젝트로 인한 효과**   **1.4 프로젝트의 산출물**  **2 자원 및 일정 예측**  **2.1 자원**  **2.1.1 인력**  **2.1.2 비용**  **2.2 일정**  **3 조직 구성 및 인력 배치**  **3.1 조직 구성**  **3.2 직무 기술**  **4 WBS 차트**  **5 기술관리 방법**  **5.1 변경 관리**  **5.2 위험 관리**  **5.3 비용 및 진도관리**  **5.4 문제점 해결 방안**  **6 표준 및 개발 절차**  **7 검토 회의**  **7.1 검토회 일정**  **7.2 검토회 진행 방법**  **7.3 검토회 후속 조치**  **8 개발 환경(언어)**  **9 성능 시험 방법**  **10 문서화**  **11 유지보수**  **12 설치, 인수**  **13 참고문헌 및 부록** |

**1. 개요**

**1.1 프로젝트 개요**

**해당 프로젝트는 기존의 학사 관리 시스템에서 발생하는 문제점들을 개선하여 새로운 학사 관리 시스템을 개발하는 팀 프로젝트이다.**

**1.2 프로젝트 선정 동기**

**현재 사용되는 학사관리 시스템의 기본적으로 두가지 문제점이 존재하며 첫**

**번째 문제점은 기능들이 어디에 위치해 있는지 찾는 것이 어렵다는 문제이**

**존재한다.**

**두 번째 문제점은 메신저의 부재로 학생들이나 교수들이 학사 관리 시스템안**

**에서 소통할 창고인 메신저가 굉장히 빈약하다는 문제점이 존재한다.**

**이러한 두가지 문제점은 충분히 개선할 수 있는 문제점이라고 봤기에 해당 프로젝트를 계획했다.**

**1.3 프로젝트로 인한 효과**

**첫 번째 문제점은 디자인 적인 부분에서 많은 개선이 필요한 문제로, 타 학교 포털 사이트와 여러 웹사이트의 디자인을 참고하여 기존 디자인보다 기능들의 위치를 더 직관적으로 배치하며, 또한 검색 기능에 연관 검색어 기능을 추가하여 사용자가 입력한 단어와 관련된 기능을 쉽게 확인할 수 있도록 함으로 원하는 기능의 위치를 보다 쉽게 찾을 수 있을 것을 기대한다.**

**현재 학사 관리 시스템에는 메신저 기능이 존재하지만, 사실상 거의 사용되지 않는 상황이다. 이에 따라, 단체 방 기능과 파일 업로드 기능 등 다양한 기능을 갖춘 새로운 메신저를 개발하고, 이를 학생들과 교수들이 자주 방문하는 포털 사이트의 메인 페이지에 배치한다면, 메신저를 더 자주 접하게 되어 사용 빈도가 증가할 것으로 기대된다.**

**1.4 프로젝트의 산출물**

**이 프로젝트는 웹사이트 기반의 프로젝트로 이 프로젝트 산출물은 웹사이트로만 작동하는 학사 관리 시스템과 그 학사 관리 시스템의 기능들이 어떤 식으로 동작하게 할 지 결정한 회의록이 본 프로젝트의 산출물로 해당 산출물들은 깃허브에 공개된다.**

**2. 자원 및 일정 예측**

**2.1 자원**

**2.1.1 인력**

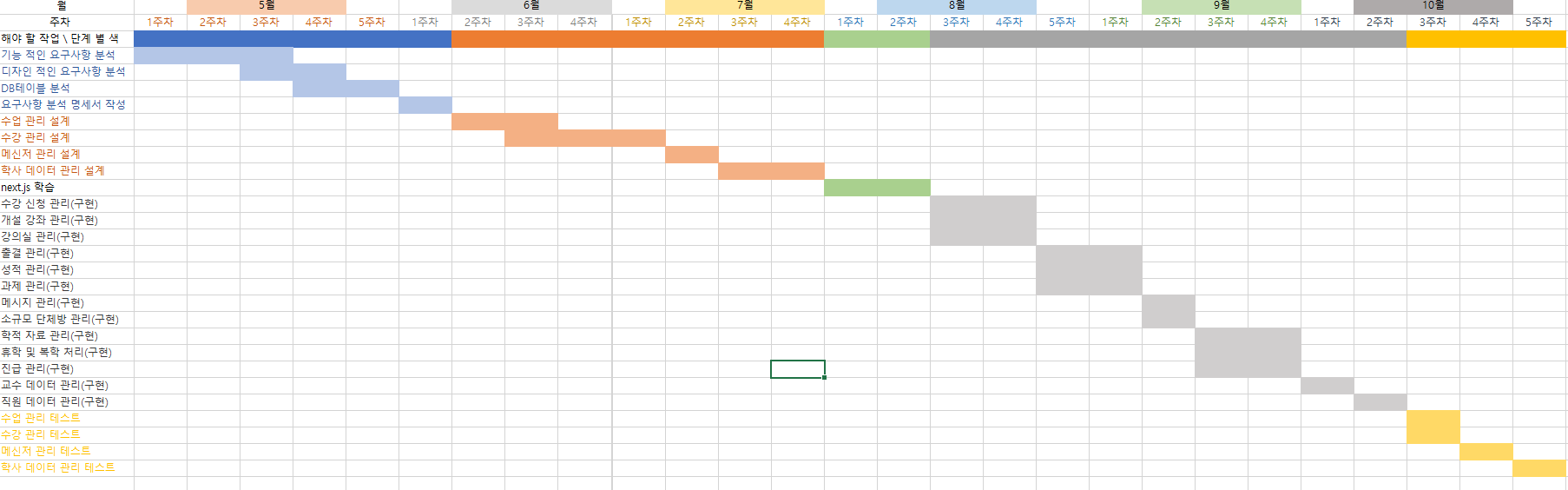
**대학생 3명이서 진행하는 프로젝트로 약 2년간에 배운 것을 바탕으로 진행하기에 추가적인 스터디나 애기가 필요해 보인다.**

**2.1.2 비용**

**248(총 기능 점수)\*210000(점수 단 단가)\*0.9(언어)\*1.0(어플리케이션 유형)\*0.6(규모)= 32,367,766** **원**

**총 비용 32,367,766** **원 한 개의 기능을 구현하는데 총 하루의 3시간을 투자한다고 했을 때 총 7일의 시간이 걸리며 급여는 최저시급(10000원) 으로 했을 때 나온 점수 당 단가인 210000원에 총 기능 점수 248점과 각종 보정 계수를 곱해서 나온 점수이다.**

**2.2 일정**

****

**파란색은 분석단계를 의미하며 주황색은 설계단계를, 회색은 구현단계 노란색은 테스트 단계를 의미한다.**

**일정의 5월부터 11월 말까지 이며 해당 간트 차트에는 wbs차트에서 분석한 작업들을 각단계에서 언제 수행할지 명시되어 있다.**

**3 조직 구성 및 인력 배치**

**3.1 조직 구성**

**조직 구성 3명의 팀원으로 구성되며 모두가 다 같이 모든 파트를 전부 참여하기에 구분이 거의 없지만 각 파트의 파트장은 존재한다. 백 엔드에 경우 김명훈씨가 파트장이며, 프론트 엔드에 경우 김수혁씨가 파트장이며, 디자인파트의 경우 김태연씨가 파트장이다. 파트장들은 각자 파트에서는 발언권이 높으며 해당 파트는 대체로 파트장에 의견에 따라 구현하게 된다.**

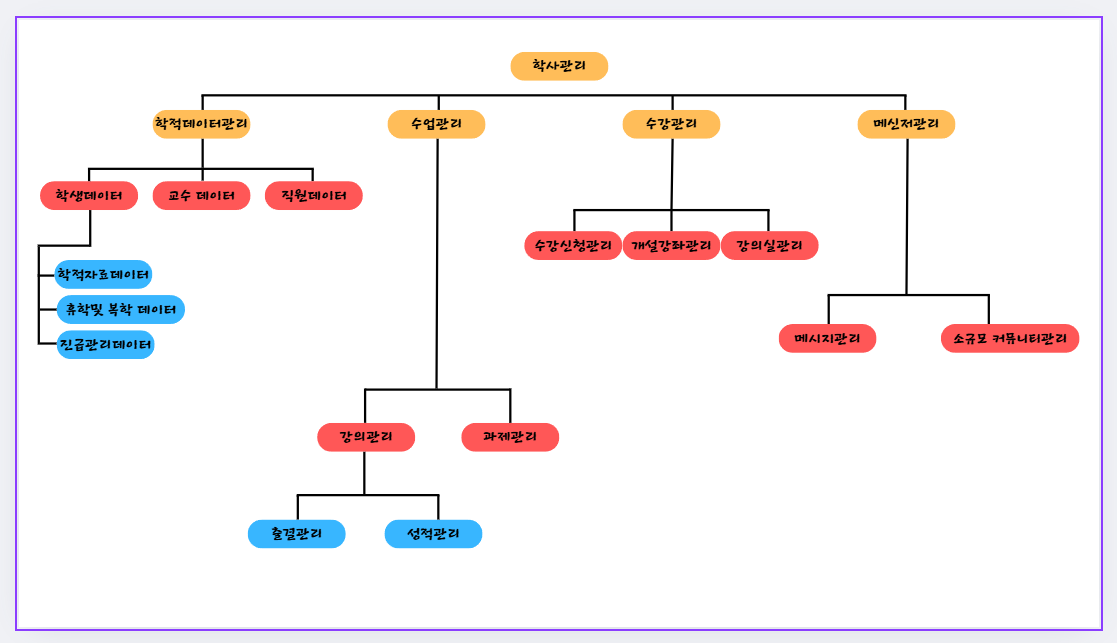
**우리 팀은 총 3명으로 구성되어 있으며, 모든 팀원이 각 파트에 고르게 참여하기 때문에 업무 구분은 거의 없다. 다만, 각 파트별로 파트장이 정해져 있어 해당 분야에서의 방향성과 의사결정에 중심적인 역할을 맡고 있습니다. 백엔드 파트는 김명훈 씨, 프론트엔드 파트는 김수혁 씨, 디자인 파트는 김태연 씨가 각각 파트장을 맡고 있으며, 각 파트에서는 파트장의 의견이 중요한 기준이 되어 이에 따라 구현 방향이 결정된다.**

**3.2 직무 기술**

* **DBSQL문 관련 지식**
* **정규화**
* **뷰와 인덱스 객체 관련 지식**
* **ERD 설계**
* **테이블 관계 설정 관련 지식**
* **자바 스크립트 이벤트 처리**
* **자바 스크립트 기본 문법**
* **객체 지향 개념**
* **HTML 및 CSS 지식**
* **깃 관련 지식**
* **NEXT.JS 프레임워크 관련지식**
* **React.js라이브러리 지식**
* **Prisma 라이브러리 지식**

**현재 모든 인력들은 NEXT.JS 프레임워크 관련 지식과 깃 관련 지식을 제외한 모든 기술들을 거의 다 습득하였다.**

**4 WBS 차트**

****

**우선 wbs차트를 보면 학사관리를 네가지 작업으로 세분화 시켰다.**

**학적 데이터관리는 실제로 데이터가 들어가는 작업들을 나타낸다.**

**실제로 관리할 데이터들은 세가지로 학생 데이터, 교수 데이터, 직원 데이터 세가지이기에 학생 데이터 관리, 교수 데이터 관리, 직원 데이터 관리 이렇게 세가지의 기능으로 분리한다.**

**학생데이터 관리에는 학생 인적 사항 그리고 학적 기록이 관리할 수 있도록 하며**

**복학 및 휴학 처리 기능과 특정 조건이 되면 자동으로 진급처리 되는 기능 이렇게 3가지 기능이 존재한다.**

**수업관리는 성적입력 출석관리, 과제 관리 등 교수들이 수업들을 관리할 때 사용하는 기능들을 모아 놓은 것이다.**

**수강 관리는 수강 신청과 관련된 작업들이 들어간다. 수강관련 작업에는 교수들이 강좌를 개설하는 것과 학생들이 개설된 강좌들을 신청하는 수강 신청이 존재한다.**

**메신저 관리는 직접적인 메시지를 보내는 메시지 기능과 자유 게시판과 같은 소규모 커뮤니티 개설 관리하는 소규모 커뮤니티 관리 두가지로 이루어져 있다.**

**5 기술관리 방법**

**5.1 변경 관리**

**만일 설계대로 구현이 불가능하다고 판단될 시 검토회에서 설계를 검토한다, 이때 변경된 설계에서 해당 요구사항을 구현하기 위해 추가적인 기능이 많이 요구될 시 사용자의 편의를 위한 기능들이 존재할 경우 구현의 필수적인 기능들이 아닌 편의를 위한 기능들을 전부 삭제를 하며 모든 기능이 요구사항을 구현하기 위한 필수적인 기능일 시 몇 개의 기능들로 대체할 수 있는지 검토해본다, 만일 대체가 불가능 할 시 추가적으로 필요한 기능들을 추가하여 구현한다.**

**5.2 위험 관리**

* **파일 삭제: 깃허브에 저장하여 팀원 컴퓨터의 파일이 없어지게 되도 깃허브에 미리 파일들을 보관하여 사라져도 불러올 수 있게 한다.**
* **소통부재: 팀원 간에 소통이 안되어 긴밀하게 연결되는 파트에서 제대로 작동이 안되는 문제가 발생할 수 있다. 이를 방지하고자 기능을 구현 했을시 어떤 식으로 구현했는지 정규 회의에서 말하는 시간을 갖는다.**

**5.3 비용 및 진도관리**

**일정이 미뤄졌을 경우, 검토 회의에서 지연된 이유를 파악하고, 필요하면 업무를 다시 재분배해서 새로운 일정을 짠다.**

**새로운 요구사항이 추가되었을 때, 해당 요구사항을 구현하는 데 큰 비용 이 들지 않는다면(기능 점수 8이하), 가능한 한 추가한다.**

**하지만 너무나 많은 기능이 추가되면 안되기에 최대 기능점수는 260으로 잡는다.**

**만일 해당 최대 기능점수를 넘는 기능을 추가해야 할 경우 기존의 기능 중에 삭제할 기능을 검토한다.**

**5.4 문제점 해결 방안**

**지출: 금전적인 문제가 발생하여서 지출이 필요할 경우 각자 1/n 지불하여**

**문제를 해결한다.**

**장시간 팀원의 부재: 인원이 부재 될 경우 따로 회의를 현재 인원의 맞게 파트를 재분배하여 일정을 조정한다.**

**6 표준 및 개발 절차**

**이 프로젝트는 전통적인 폭포수(Waterfall) 모델을 기반으로 하되, 설계 단계에서 프로토타입 개발을 병행하여 설계가 적절한지 판단해 설계의 완성도를 높인다.**

**이는 이미 계획단계에서 요구사항이 어느정도 구체적으로 뽑혔기에 분석과정에서 프로토타입을 진행하여 요구사항을 구체화하는 것 보다 설계단계에서**

**포로토타입 모델을 진행하여 요구사항을 분석하여서 나온 기능들이 정확한지**

**검증함과 동시에 프로토타입은 디자인 설계하는 과정에서도 많은 도움이 되기에 설계단계에서 프로토타이핑을 진행하여 전체적인 완성도를 높인다.**

**또한 소스코드를 작성할 때 표준적인 절차는 클래스명은 파스칼 스타일로 작성하며 함수(또는 메서드)와 변수명은 캐멀 스타일로 작성한다.**

**if문과 for문을 사용할 때는 가능한 주석을 쓰며 클래스와 메서드를 선언할 시 무슨 클래스와 메서드인지 설명하는 주석을 반드시 작성해야 한다.**

**7 검토 회의**

**7.1 검토회 일정**

**매주 수요일마다 검토회의를 진행한다.**

**7.2 검토회 진행 방법**

**메인 회의 때 결정한 작업들이 잘 진행되었나 확인하며 추가로 안건들에**

**대해서도 회의한다.**

**7.3 검토회 후속 조치**

**만일 검토회를 통해 진행하여 결정된 사항은 특별한 경우가 아니면 무조건 이행한다.**

**8 개발 환경**

**HTML및 CSS와 자바 스크립트, next.js 프레임워크를 통해 진행한다.**

**IDE는 비주얼 스튜디오 코드로 하여 소스코드를 작성한다.**

**Next.js프레임워크를 통해 프론트엔드 및 백엔드를 전부 개발하고자 한다.**

**Next.js는 틱톡, 나이키 홈페이지, twitch 등 많은 기업들에서 사용하고 있는 프레임워크로 자바 스크립트를 통해 많은 작업들이 가능하며 간단하게**

**웹사이트 개발이 가능하기에 next.js를 통해 개발한다.**

**마지막으로 DB는 MariaDB를 사용하여서 개발하고자 한다.**

**9 성능 시험 방법**

**다른 학사 관리 시스템의 포털사이트 현재 개발중인 포털사이트와 비교하여**

**추가적인 기능이나 문제점이 없는지 검토하며 다 팀에게 테스트하게 하여**

**문제점이 있는지 물어본다.**

**10 문서화**

* **회의기록 및 결정사항들은 모두 워드파일로 저장하여 나중에 확인할 수 있도록 한다.**
* **발표의 사용된 ppt와 각 단계에서 산출한 문서들은 word,excel 파일로 작성되어 확인 가능하게 한다.**
* **프로젝트 개발에 사용된 모든 소스코드 파일들을 깃허브에 업로드하여 관리한다.**

**11 유지사항**

* **건의사항 포럼을 만들어서 해당 건의사항을 바로 작성할 수 있게 한다.**
* **신속하게 처리할 수 있는 건의사항(ex: 포털 메뉴 위치 변경)은 빠르게 처리한다.**
* **신속하게 처리할 수 없는 건의사항(ex: 도서관 홉페이지 같은 신기능 추가)의 경우 비용 산정한 후 많은 비용이 들지 않을 경우에 추가를 진행하며 만일 비용이 많은 드는 기능일 경우 다른 기능으로 대체 가능한지 찾는다.**
* **개발자와 직접적인 1대 1 문의를 원할 시 이메일로 문의한다.**

**12 설치, 인수**

**웹사이트이기에 다른 설치 과정이 필요가 없으며 인수과정에 경우 깃허브를 통해 전달하며 만일 인수 과정에서 오류가 생길 경우 깃허브에 문의한다.**

**13 참고문헌 및 부록**

**본 보고서는 쉽게 배우는 소프트웨어 공학을 참고하여서 작성되었다.**