**[ 10월 1주차 프로젝트 수행 일지 ]**

|  |  |  |  |
| --- | --- | --- | --- |
| **프로젝트 타이틀** | Websocket기반 실시간 스포츠 매칭 플랫폼 | | |
| **프로젝트 팀명** | Multi FC | 프로젝트 팀원 | 팀장 : 박태란  팀원 : 권성민, 문진우, 윤승근, 최준호, 홍예린 |

※ 프로젝트 수행 일지는 매 주 금요일 17시까지 최종본을 저장해주세요

**1. 프로젝트 수행 계획 및 현황**

|  |  |  |
| --- | --- | --- |
| **1. 프로젝트 수행 계획 및 현황** | | |
|  | **금주 프로젝트 수행 내용** | **차주 프로젝트 수행 계획** |
| **10/27** | **1. 프로젝트 주제 선정 및 자료조사**  - 주제 : Websocket기반 실시간 스포츠 매칭 플랫폼  - Websocket기반 실시간 스포츠 매칭 플랫폼 프로젝트 세부 주제 조사   |  |  |  | | --- | --- | --- | | 권성민 | 풋살 매칭 | 실시간으로 사용자 간의 매칭이 가능 한 풋살 플랫폼 | | 문진우 | 한국형 스포츠 매칭 | WebSocket을 사용하여 사용자들이 실시간으로 소통하며 함께 운동할 상대를 찾고 경기를 주선하는 플랫폼 | | 박태란 | 풋살 플랫폼 | WebSocket 기반 매칭 및 채팅 시스템과 더불어, 풋살에 대한 정보등을 커뮤니티 내에서 지속적으로 소통 가능할 수 있게 하는 풋살 플랫폼 | | 윤승근 | 실시간 풋살 매칭 | 사용자들이 실시간으로 함께 풋살할 인원를 찾고 채팅으로 소통하며 경기를 주선하는 매칭 플랫폼 | | 최준호 | 실시간 메시지 축제 만남 사이트 | 관광정보 OpenAPI(TourAPI)와 연동해 전국 축제, 관광지, 음식점 등 최신 관광 데이터를 실시간으로 제공 하여 사용자들이 축제 정보를 공유하고, 실시간으로 함께할 동반자나 그룹을 찾으며, 현장 상황에 따른 커뮤니케이션을 할 수 있는 플랫폼 | | 홍예린 | 실시간 야구 응원 배틀 웹 | 실시간으로 야구 팬들이 자신이 응원하는 팀을 선택해 응원 버튼을 누르면,  모든 사용자 화면에 팀별 응원 수가 즉시 반영되는 웹사이트 |     (조사한 자료 공유한 Notion 페이지 첨부/https://lavish-mountain-fb1.notion.site/29928b27a55981c69a8cdb9d024e92ed)  - 개인 사용자도 쉽게 팀을 구성하고 풋살 유저 간의 채팅을 통한 실시간 소통과 정보 공유를 할 수 있는 플랫폼을 주제로 선정  **2. 기능 구상**  - 사용자 프로필  - 개인별 매칭, 팀 별 배칭  - 신뢰도 기반 매칭 → 개인에 대한 후기  - 실시간 채팅 기능  - 커뮤니티 기능  - 카카오맵 기반 주변 풋살장 위치 찾기 | |  |  | | --- | --- | | 권성민 | 1. 웹페이지(로그인, 회원가입, 메인, 마이페이지, 정보수정, 구장검색, 실시간채팅, 커뮤니티 등등) 구현  2. 수행일지 작성 | | 문진우 | 1. 데이터베이스 생성 2. 메일을 통한 회원가입 구현 3. 소셜로그인을 통한 회원가입 구현 | | 박태란 | 1. DB스키마 확정 및 테이블 생성  2. 마이페이지 개인정보 수정 부분 구현  3. 일정 생성 및 삭제, 수정 기능 구현 | | 윤승근 | 1. 웹페이지(로그인, 회원가입, 메인, 마이페이지, 정보수정, 구장검색, 실시간채팅, 커뮤니티 등등) 구현  2. 수행일지 작성 | | 최준호 | 1. RDS 구성 후 로컬 작업 환경에 연동.  2. 채팅 기능 구현 ( 대화방 만들기 구현, 메세지 전송 기능 구현 ) | | 홍예린 | 구장 정보와 카카오맵 API를 연동하여, 사용자가 지도상에서 구장을 검색 및 조회할 수 있는 핵심 기능의 백엔드(API) 개발을 완료할 예정이다.  백엔드: 기반 환경 구축 및 핵심 DB 설계  1. DB 스키마 확정: 매칭 시스템의 근간이 되는 Stadium, Match\_Room, Match\_Participant 테이블 스키마를 최종 확정하고, 개발 DB에 생성한다.  2. 프로젝트 초기 설정: Spring Boot 등 백엔드 프로젝트 초기 환경을 구축한다.  3. API 키 발급: 카카오 개발자 콘솔에서 애플리케이션 등록 및 지도 연동에 필요한 API 키를 발급받는다.  백엔드: 구장(Stadium) API 개발  1. 구장 생성 : 관리자가 구장 정보를 등록하는 API를 개발한다.  2. 구장 조회 : 지도에 표시할 전체 구장 목록을 반환하는 API를 개발한다.  3. 구장 검색 및 상세 조회 : 사용자가 키워드(이름/지역)로 구장을 검색하거나, 특정 구장의 상세 정보를 조회하는 API를 개발한다. | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **10/28** | **1. 사이트맵 작성**  - 사이트를 구성 할 전체적인 사이트맵 작성  • **로그인 페이지**  ⌊  회원가입 페이지  아이디/비밀번호 찾기 페이지  • **메인페이지**  ⌊  구장 페이지– 매칭방 입장 – 그룹채팅 생성 – 후기  실시간채팅 페이지– 채팅목록 + 1:1/그룹 채팅방 생성  커뮤니티 페이지 – 공지, 자유, 팀원모집 등의 게시판  일정 – 내 매칭 일정 표시  **• 마이페이지**  ⌊  프로필 확인  일정 관리  정보수정  (사이트맵 구성 이미지 첨부)  **2. 역할분담**  - git관리를 위한 팀장 : 윤승근, 최준호   |  |  | | --- | --- | | Frontend – 권성민, 윤승근  Backend – 문진우, 박태란, 최준호, 홍예린 | | | 권성민 | 로그인페이지/ 회원가입페이지/아이디&비밀번호찾기 페이지/정보수정페이지 | | 윤승근 | 메인페이지/마이페이지 | | 문진우 | 데이터베이스 및 로그인 영역 | | 박태란 | 마이페이지 영역 + 일정표 | | 최준호 | 실시간 채팅 영역 | | 홍예린 | 카카오맵 연동 및 커뮤니티 영역 |   **3. 필요한 기술스택 논의**  - WebSocket, 카카오맵 API연동  - Database : MySQL  - 아키텍쳐 모델 : MVC모델 적용  - Frontend : Vanila, Html, Css, Js  - Backend : Spring Boot  \*\* 채팅방, 커뮤니티, 일정표, 카카오맵 기능 구현방법 찾아보기  \*\* Database로의 로그인 방법 외의 구글, 카카오등의 로그인 연동 기능 구현 고려하기  **4. 변경사항**  - 개인에 대한 후기 남기기 → 구장에 대한 후기 남기기로 변경 |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **10/29** | **1. WBS초안 작성**  - 10/29 에 정한 역할 부분에 필요한 기능 및 페이지 정리   |  |  | | --- | --- | | 권성민 | 로그인페이지/ 회원가입페이지/아이디&비밀번호찾기 페이지/정보수정페이지 | | 윤승근 | 메인페이지/마이페이지 | | 문진우 | 데이터베이스 및 로그인 영역에 필요한 페이지 정리 | | 박태란 | 마이페이지 영역 + 일정표 영역에 필요한 페이지 정리 | | 최준호 | 실시간 채팅 영역에 필요한 페이지 정리 | | 홍예린 | 카카오맵 연동 및 커뮤니티 영역에 필요한 페이지 정리 |     (WBS초안 중 일부 이미지 첨부)  **2. 담당 부분 별 기능 명세서 작성**   |  |  | | --- | --- | | 문진우 | - 데이터베이스 설계에 필요한 주요 엔티티 별 테이블 생성 스크립트 작성 및 필드명, 타입 설정  - 시작 및 사용자 인증 부분 입출력값, 예외처리 보안 부분 작성 | | 박태란 | 마이페이지 내 프로필 수정과 일정 관리에 필요한 기능 정리, 입출력값, 예외처리 보안 부분 작성 | | 최준호 | - 채팅 영역의 1:1, 그룹 채팅 기능 정리, 입출력값, 예외처리 보안 부분 작성 - 채팅방 생성 기능 정리, 입출력값, 예외처리 보안 부분 작성 | | 홍예린 | - 카카오맵 연동 구장 검색, 실시간 인원 반영, 매칭방 연동, 경기 생성 기능 정리, 입출력값, 예외처리 보안 부분 작성  - 커뮤니티 영역 게시판 구성과 게시판 상세, 글작성 삭제 등의 기능 정리, 입출력값, 예외처리 보안 부분 작성 |     (기능 명세서를 작성 한Notion 페이지 첨부 / https://lavish-mountain-fb1.notion.site/29928b27a55981928d87fe92a092635f)  **3. 변경사항**  - 경기 생성 기능 구현 부분 추가 (구장검색애서 구장 선택 후 보이는 상세페이지에서 경기 생성)  - 후기 시스템 (매칭방 생성 시 설정 한 시간 이후 일정 시간 지나면 참여자에게 후기 남기는 알림)  **4. 기타**  - 작성한 기능명세서 바탕으로 WBS 세부내역 확장 필요 |  |

|  |  |  |
| --- | --- | --- |
| **10/30** | **1. WBS 수정 ( 기능 별 세분화 및 수행 일정 작성)**  - 로그인 기능: JWT Access Token 기능, SSO 기능, 아이디/비밀번호 찾기 기능 (메일 번호 인증), 사용자 유효성 확인 기능 등으로 작업 세분화  -구장 검색 기능: 구장 검색 기능(카카오맵 API 연동), 구장 필터링 및 상세 팝업, 페이지 연동 기능, Websocket 서버 연동, 인원 수 제한 기능 등으로 작업 세분화  - 일정표 기능: 마이페이지 내 캘린더 매칭일정 연동, 일정 생성, 수정, 삭제 기능 구현, 필수입력 누락 시 예외처리, 다가오는 일정 없을 시 빈 리스트 반환 및 안내메세지 등으로 작업 세분화  - 채팅 기능: 채팅 모델/DB (메시지, 채팅방, 멤버) 설계, 채팅방 생성/초대/퇴장 로직 구현, 채팅 메시지 저장/조회/읽음 처리, 테스트 및 부하/스트레스 테스트 등으로 작업 세분화    (WBS 중 일부분 이미지 첨부)  **2. 요구사항 정의서 작성**  - 페이지별 회원/비회원에게 제공되는 서비스와 그에 따른 필요 기능, 기능에 대한 설명 작성    (요구사항 명세서 중 일부분 이미지 첨부)  **3. API명세서 초안 작성**  - 프론트와 백엔드가 공유하는 각 페이지, 기능 별 API 작성    (API명세서 중 일부분 이미지 첨부)  **4. 기획서 수정**  - 기획안 내용 확인  - 프로젝트 수행 도구 추가  • 배포 및 빌드: 사용자 이미지와 커뮤니티에 올라오는 이미지 파일 등을 저장하기 위해 S3추가  • Backend: RabbitMQ 추가  • DB: ERD작성을 위한 drwxip추가 |  |

|  |  |  |
| --- | --- | --- |
| **10/31** | **1. API명세서 수정**  - 메인페이지 누락된 Router추가    (수정 된 API명세서 이미지 첨부)  **2. 와이어프레임 작성(권성민,윤승근)**  - 구장 검색(구장 정보, 경기 생성, 경기 참여), 커뮤니티(커뮤니티 페이지, 게시판 글쓰기 페이지, 세부 페이지), 채팅방(현재 들어가 있는 채팅 페이지, 그룹 채팅방 페이지, 인원 확인 및 유저프로필 확인 페이지) 와이어 프레임 작성  **3. 유저 플로우 작성 (문진우, 박태란, 최준호, 홍예린)**  **-** 사용자가 사이트를 방문했을 때를 가정하여 예상되는 시나리오 작성  - 풋살 매칭을 위해 방문한 유저를 타겟으로 했을 때 예상되는 시나리오 작성    (User Flow 이미지 첨부)  **4. ERD 검토**  - ERD명세서 바탕으로 데이터베이스 테이블 설계 및 데이터 구조 정의    (ERD이미지 첨부) |  |

**2. 강사님 피드백**

|  |
| --- |
| WBS 초안까지 구성한 점 아주 좋습니다.  주제 선정 후 바로 역할을 나누고, 기능 명세서를 작성해 세부 단계를 고민한 모습이 인상적이었습니다.  아직 프로젝트 초반이라 세부 일정이 다소 포괄적이지만, 방향은 잘 잡혀 있습니다.  다음 주에는 각 기능별 작업 단계를 ‘설계 → 구현 → 테스트 → 검증’ 순서로 세분화하고, 담당자와 예상 소요 기간을 함께 표시해보세요.  특히 ‘채팅’, ‘커뮤니티’, ‘카카오맵 연동’ 같은 주요 기능은  상호 의존도가 높기 때문에 순서를 명확히 잡아두면 이후 충돌이나 일정 지연을 방지할 수 있습니다.  전반적으로 팀워크가 좋아 보여서, 구조만 조금 더 구체화되면 훨씬 안정적인 개발이 될 듯합니다. |