최종 보고서

생동감 있는 VR 체스게임

텍스트, 엠블럼, 로고, 원이(가) 표시된 사진

자동 생성된 설명]

**팀: 컴공이면 게임해야조**

**제출 일시: 2024 - 06 - 22**

목차

1. 개요

1.1 기획 배경

1.2 관련 동향

1.3 필요성 및 기존 연구/서비스의 한계점

1.4 연구/서비스의 소개

1.4.1 목표

1.4.2 기여

1.4.3 예상 결과물

1.5 프로젝트 주요 기능 및 특징

2. 기능적 요구사항

2.1 Top Level Use Case Diagram

2.2 각 기능별 동작 시나리오

3. 비기능적 요구사항

3.1 사용 편리성

3.2 신뢰성

3.3 성능

3.4 유지관리

3.5 구현

3.6 인터페이스

4. 실험 및 평가 방법

5. 현재 진행 상황과 향후 계획

5.1. 현재 진행 상황

5.2. 향후 계획 일정

6. 팀의 구성 및 팀원의 역할

7. 참고문헌

1. 개요

1.1 기획 배경

체스와 게임에 흥미 있는 학우들끼리 모여 프로젝트에 관해 생각하던 중, 다른 사람들도 이런 체스의 재미를 느낄 수 있도록, 교두보가 될 수 있는 생동감 있는 게임을 VR기기를 통해 만들어보고자 하여 이 프로젝트를 시작하게 되었습니다.

1.2 관련 동향

VR 게임은 아직 초기 단계이므로 기기가 비싸 아직까지는 대중화되기에는 진입장벽이 높습니다. 하지만 기술발전에 따라 기기 가격은 점점 저렴해질 것이며 VR게임에 대한 소비자 인지도 또한 올라갈 것입니다.

더불어 5G의 등장으로 더 빠른 네트워크 처리로 복잡하고 상세한 VR환경을 가능하게 하고 있습니다. 이로 인해 더욱 실감나고 게임 경험을 제공함에 있어 VR을 사용해보고자 하는 유저들이 늘어나고 있습니다.

또한 소니, 에픽 게임즈 등 유명 게임사들이 지속적으로 VR게임 시장에 투자하고, 여러 VR게임이 출시를 앞두고 있습니다. 따라서 많은 시간과 자본이 투자되고 있는 VR게임의 성장세는 가속화되리라 예상됩니다.

1.3 필요성 및 기존 연구/서비스의 한계점

저희 게임은 체스를 기반으로 한 2 : 2 게임으로, 서로와의 협동을 통해 유대감을 형성할 수 있습니다. 일반적인 체스 보다는 VR을 통해 색다른 느낌의 체스를 플레이 해보고 싶은 사람이라면 도움이 될 것입니다.

또한 기존 연구/서비스는

1.4 연구/서비스의 소개

1.4.1 목표

VR을 착용하고 움직임에 이질감이 들지 않는 게임을 구현합니다. 1인칭으로 플레이하는 플레이어에게는 체스 판 내에서의 충분한 공포감을 느끼게 해주고, 명령을 내리는 플레이어에게는 그걸 관전하는 재미를 줄 수 있는 것이 목표입니다.

1.4.2 기여

두 플레이어가 처한 상황이 달라 벌어지는 상황들을 잘 대처해 나가면 서로 간의 친밀감과 유대감 형성에 기여할 것입니다.

1.4.3 예상 결과물

2 : 2로 진행되는 VR 공포체험 체스게임 입니다. 한 명은 체스게임을 하며 VR을 쓴 플레이어에게 이동명령을 내리고, 한 명은 그 명령을 받아 이동하며 체스게임의 말이 되는 체험을 하게 됩니다.

체스 말이 되는 사용자는 공포감 자체로 재미를 느낄 것입니다. 명령을 내리는 플레이어는 체스 플레이 자체의 재미에 더불어 장기말이 된 플레이어가 주는 변수들을 전략으로 극복해 나가는 재미를 느낄 것입니다. 또한 두 플레이어 간의 의견 충돌로 인해 발생하는 돌발적인 재미있는 상황들이 연출 될 것입니다.

1.5 프로젝트 주요 기능 및 특징

1.5.1 기물 이동

체스를 플레이하는 사용자(이하 1p)가 이동시킬 기물을 정하면 명령을 듣고 움직이는 사용자(이하 2P)의 시점이 옮겨지고 앞에 명령에 따른 문구가 보입니다. 2P는 명령을 참고하여 자유롭게 기물을 이동시킬 수 있습니다.

이동 중, 또는 이동 완료 후 도착했다는 신호를 보낼 때 기물이 이동될 수 있는 칸인지 확인하고 가능하다면 이동이 종료됩니다. 가능하지 않다면 이동 전 위치로 되돌아 가게 됩니다.

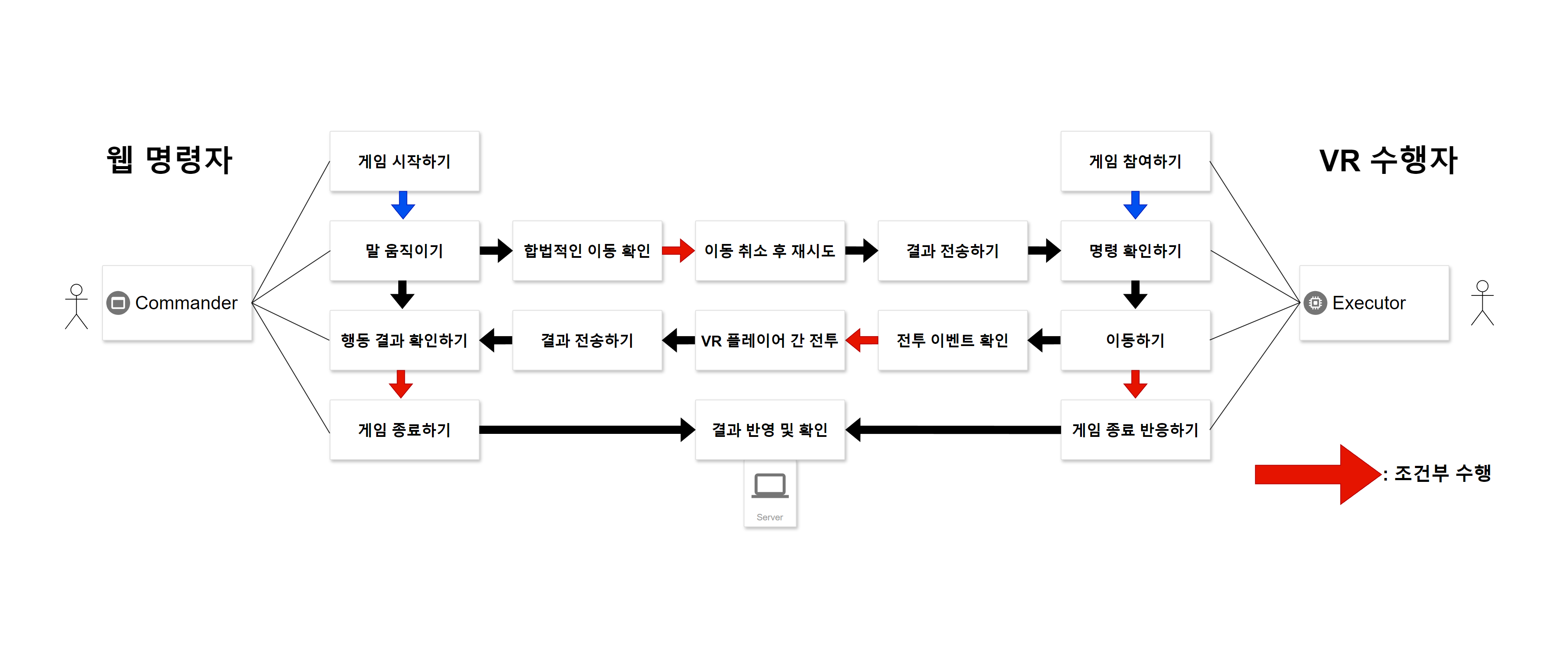
1.5.2 전투 기능

기물 이동 중 상대 기물과 마주치거나, 기물 이동 완료 후 도착 칸에 상대 기물이 존재한다면 전투가 발생합니다. 전투 시 순간적으로 나오는 선을 따라 공격하고 공격을 얼마나 잘 했는지에 따른 점수가 부여됩니다.

전투 승리 조건은 상대 사용자와 전투를 진행했을 경우, 점수가 더 높은 쪽이 승리합니다. 사용자가 아닐 경우에는 특정 점수 이상이면 승리합니다.

2. 기능적 요구사항

Use Case Diagram



2:2 팀 게임으로 진행됩니다. A팀과 B팀이 있다고 했을 때 A팀과 B팀에는 각각 명령자와 수행자가 존재합니다.

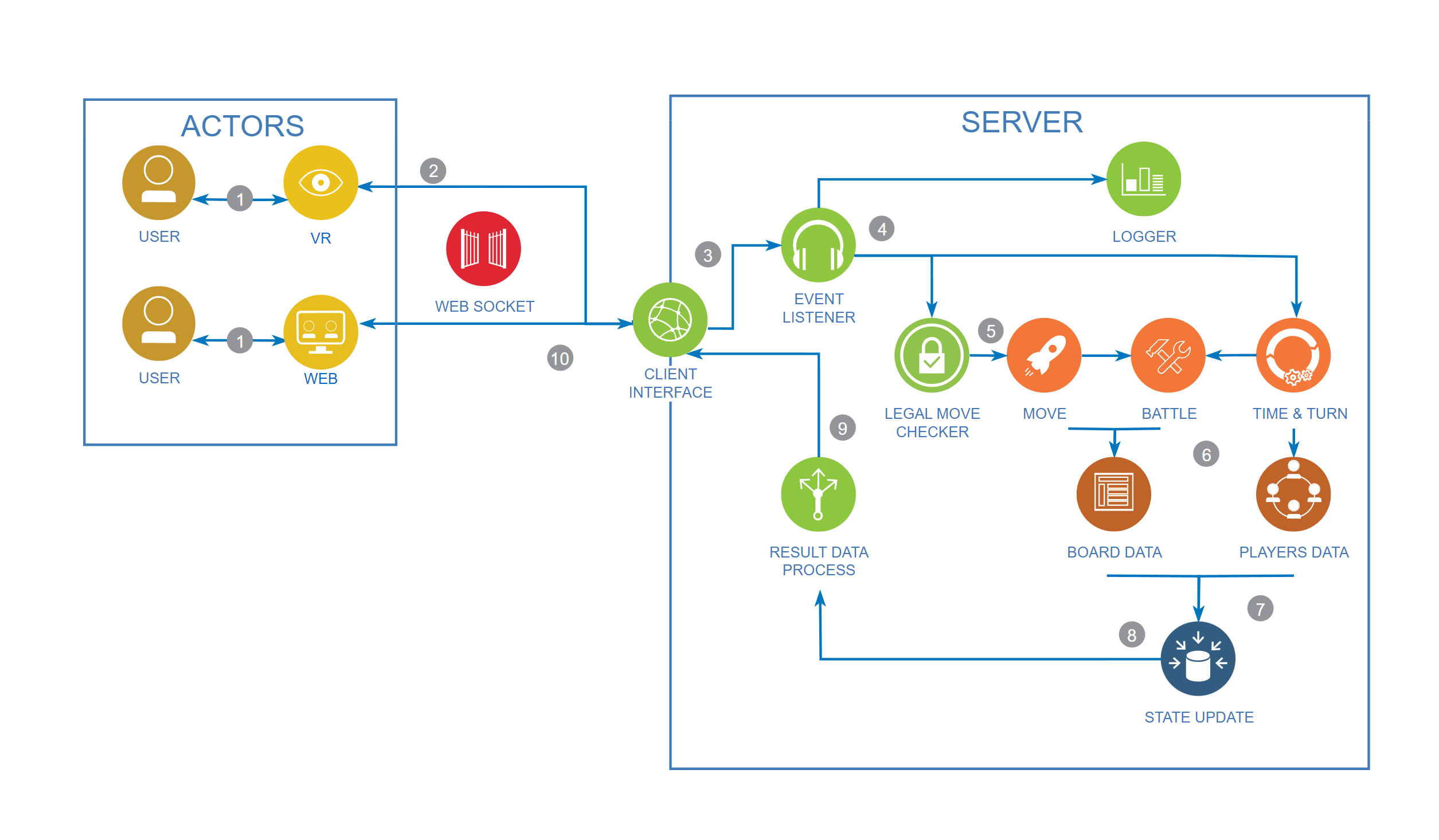
A팀의 명령자가 기물을 움직이면 서버에 전달되고, 움직임 정보는 A팀의 행동자에게 전달됩니다. A팀의 행동자는 명령을 받으면 기물이 되어서 움직입니다.

이후 B팀의 명령자와 행동자도 동일한 방식으로 차례를 진행합니다.

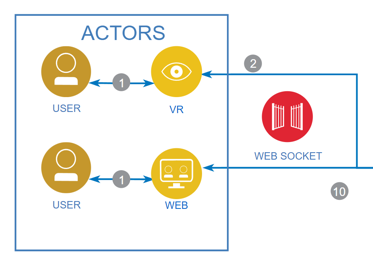
요약하면, A팀 명령자 움직임 -> A팀 행동자에게 전달 -> A팀 행동자 움직임 -> B팀 명령자 움직임 -> B팀 행동자에게 전달 -> B팀 행동자 움직임 게임이 끝날 때까지 이 과정을 반복합니다.

Figure 1은 한 팀 내에서 명령자와 행동자의 Usecase를 나타내는 다이어그램입니다.

Component Diagram



- **Component Diagram** 설명



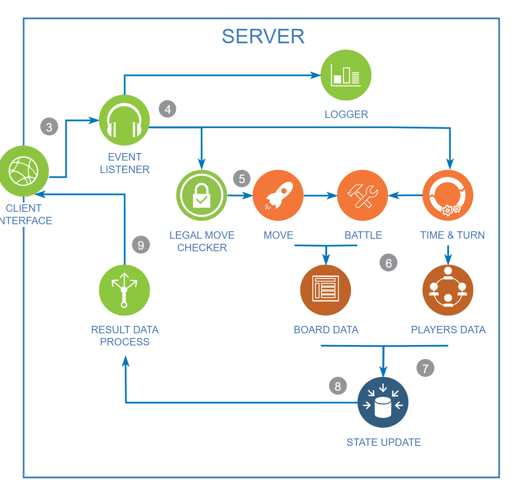
**▶User :**

VR 사용자와 웹 사용자

각각의 인터페이스를 통해 상호작용

**▶Web Socket :**

VR 및 웹 인터페이스 모두 서버와의 통신에 웹 소켓 연결



**▶Move :**

입력 : 이동 검사 결과, 아이템 사용 여부

출력 : 전투 발생 판별, 이동 후 기물 좌표

**▶Battle :**

입력 : 이동 검사 결과, 아이템 사용 여부

출력 : 전투 결과

**▶Time & Turn :**

입력 : 웹 및 VR 클라이언트 명령

출력 : 웹 및 VR 클라이언트의 턴, 남은 시간 및 아이템

**▶Board Data :**

입력 : 이동 및 전투 결과

출력 : 현 보드 상태

**▶Players Data :**

입력 : 남은 시간, 턴 및 사용한 아이템

출력 : 시간, 턴 및 아이템 갱신 결과

**▶State Update :**

입력 : 보드와 플레이어 데이터

출력 : 보드와 플레이어 데이터

**▶Result Data Process :**

입력 : 보드와 플레이어 데이터

출력 : 클라이언트에 전송하기 위해 분류 및 패키징 한 메시지

2.2 각 기능별 동작 시나리오

- 기물 이동

|  |  |
| --- | --- |
| 기물 이동 | |
| Name of Use Case | 기물 이동 |
| Actors | 1P(명령자), 2P |
| Flow of Events | 1. 1P : 이동명령 , 2P : 이동명령 확인  2. 2P : 이동  3. 2P : 도착 신호 보냄  4. 위치 전송  5. 위치정보 확인  6. 턴 종료 |
| Exceptions | 5에서 상대 기물 확인되면 전투  5에서 갈 수 있는 위치에서 벗어났다면 2번으로 회귀 |

- 전투 기능

|  |  |
| --- | --- |
| 전투 기능 | |
| Name of Use Case | 기물 이동 |
| Actors | 2P(수행자) |
| Flow of Events | 1. 상대 기물에 나타난 선 확인  2. 선을 따라 공격  3. 전투 스코어 및 시간 서버에 전송  4. 공격 성공/실패 판정 전송  5. 상대 기물 파괴 후 칸 차지 |
| Exceptions | 4에서 공격실패 판정 시 본인이 파괴됨 |

3. 비기능적 요구사항

3.1 사용 편리성

체스를 플레이하는 사용자에게 게임 진행 상황을 조금 더 직관적으로 살펴보며 플레이할 수 있도록 인터페이스 구성.

체스 말이 되는 사용자에게 게임 내 움직임을 적응할 수 있도록 VR 기기 체화를 위한 튜토리얼 제공.

기존 체스게임과의 차이점을 인지시키고 새로운 규칙을 설명하기 위한 설명서 제공.

3.2 신뢰성

플레이에 불편함이 없도록 99.999%의 가용성 보장.

그래픽 깨짐 현상으로 인한 멀미 유발 방지를 위한 그래픽 퍼포먼스 최적화.

3.3 성능

턴 제 게임이기 때문에 불편하지 않을 정도의 지연시간 (30~40ms) 유지.

사용자가 전투 시 실시간으로 진행되는 듯한 느낌을 받도록 100ms 응답시간으로 전투 실행 결과 확인.

3.4 유지관리

사용자 피드백을 빠르게 반영하여 오류 수정.

추후 업데이트에 대비한 설계.

3.5 구현

Unreal Engine 5를 이용한 클라이언트 개발. 자바를 이용한 서버 구축.

Epic Game Store에서 판매하는 에셋 사용.

3.6 인터페이스

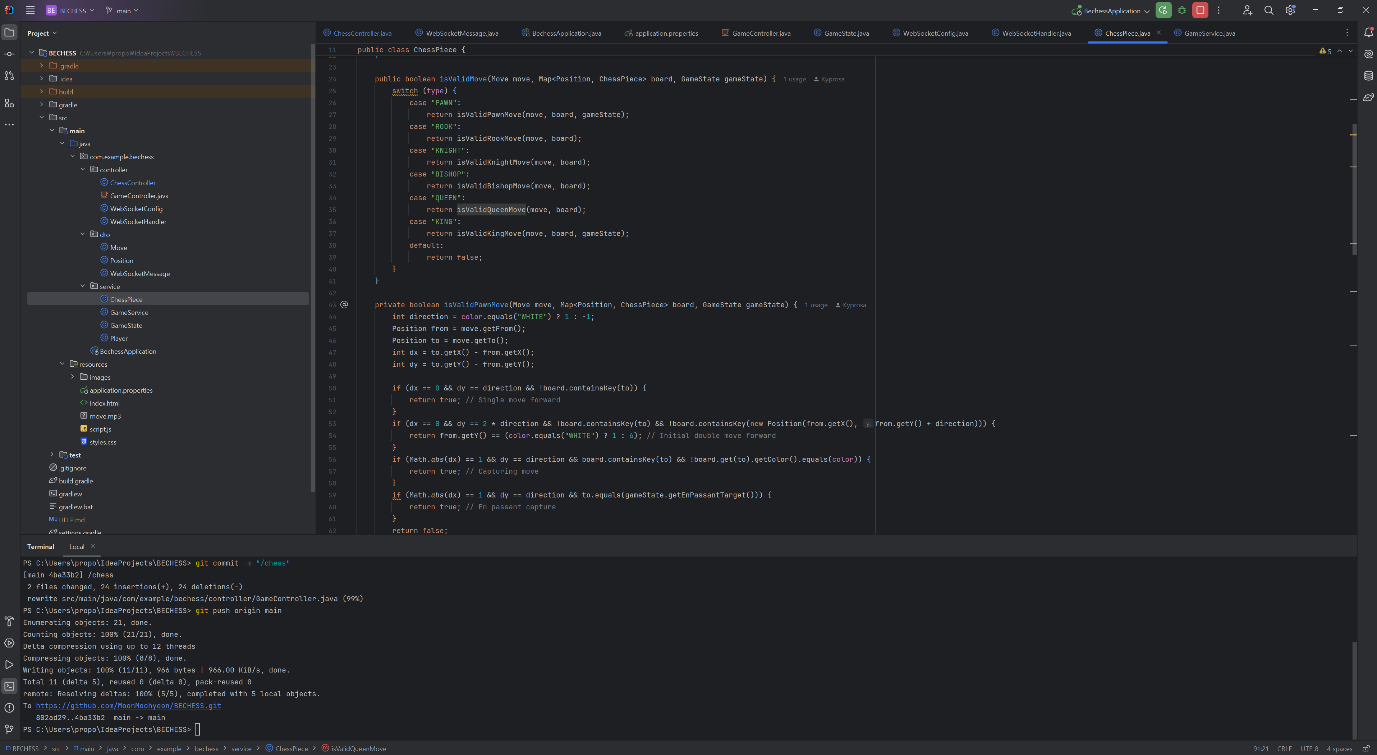
RESTful API 활용하여 클라이언트와 서버 연결.

4. 실험 및 평가 방법

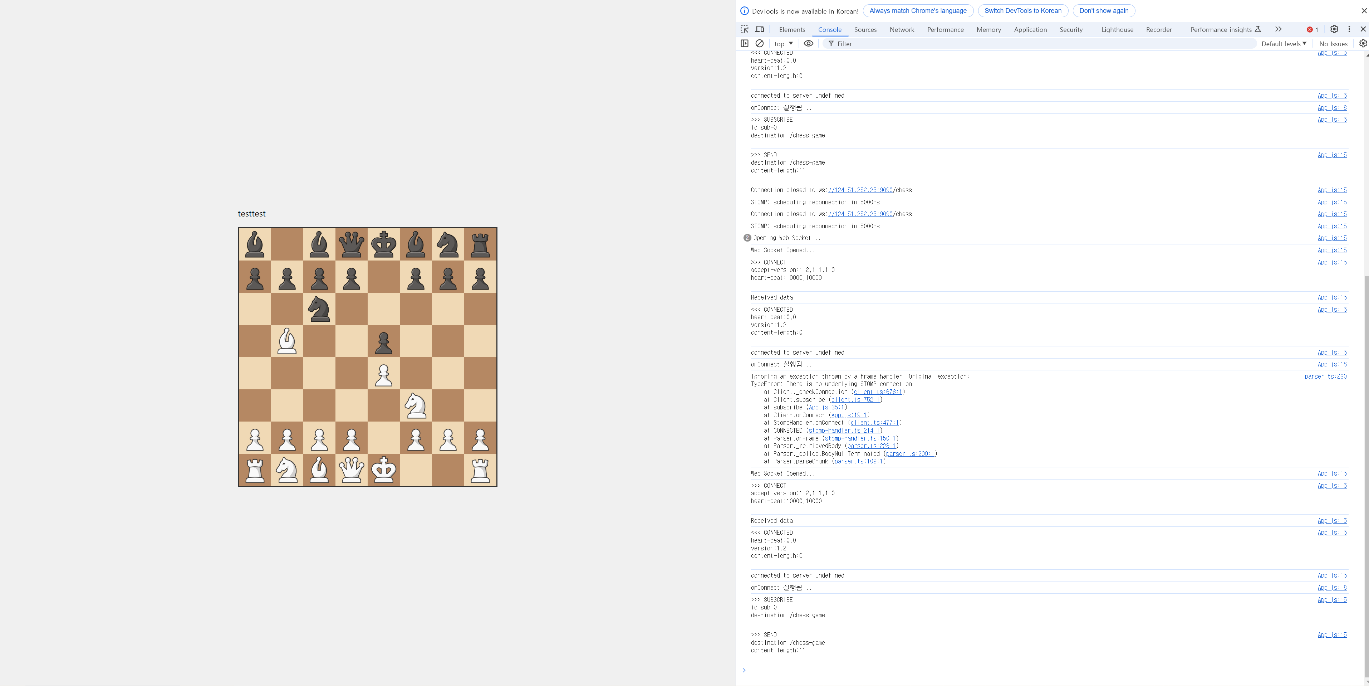
5. 현재 진행 상황과 향후 계획

5.1. 현재 진행 상황

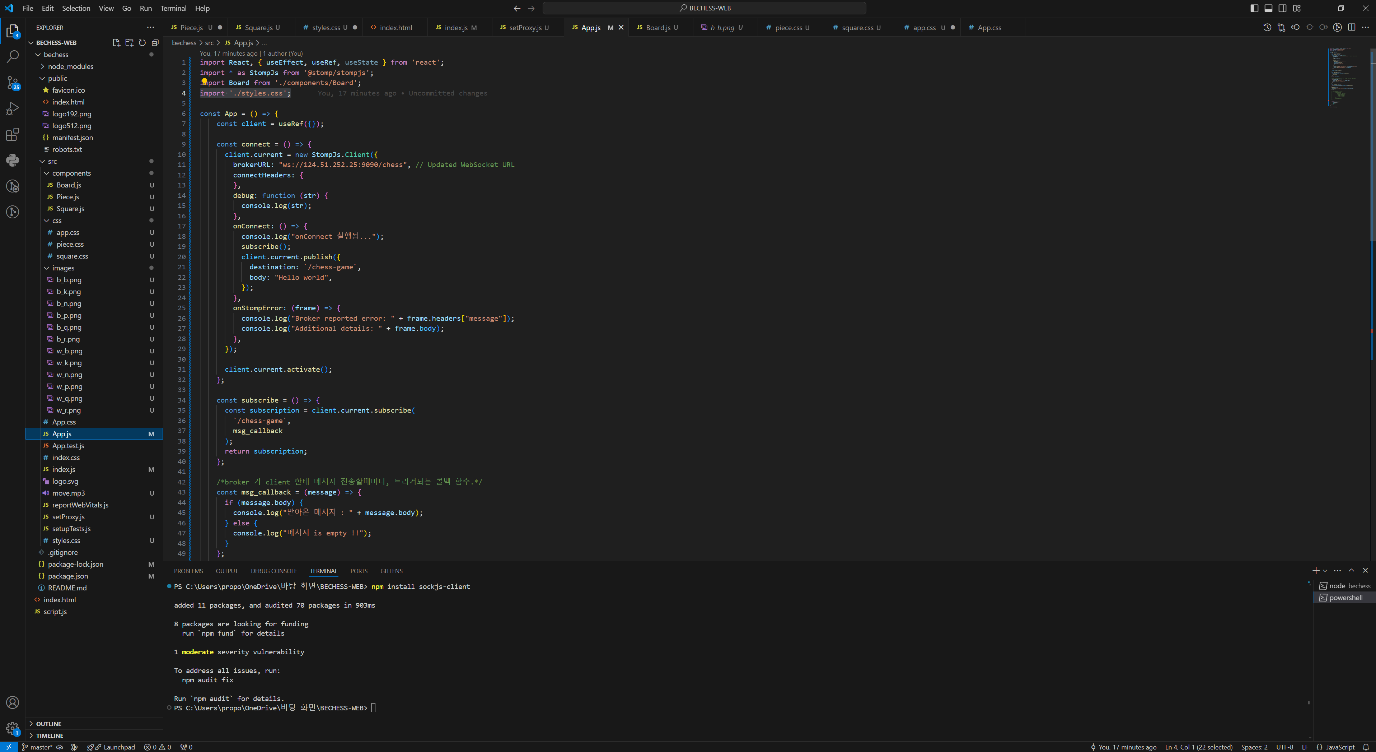
서버 진행 상황



기물 움직임에 대한 로직은 스프링 부트 서버에서 구현하였습니다.

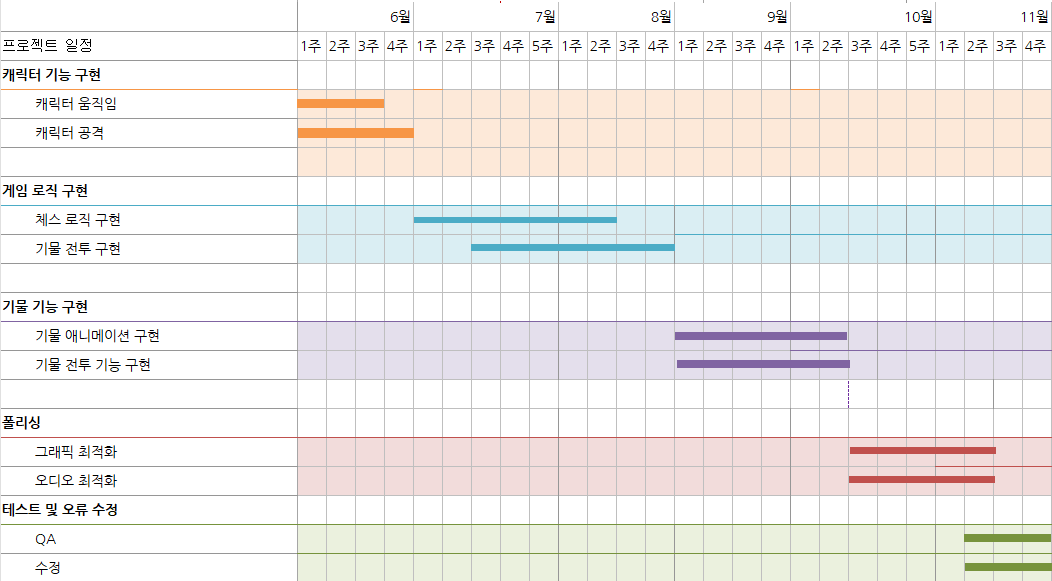


웹 클라이언트의 보드와 기물을 구현하고 웹소켓으로 서버와 연결하는 작업을 완료하였습니다.



서버는 배포 예정이 없으므로 문무현 개인 컴퓨터를 사용할 예정입니다. IP 주소는 125.51.252.25이며, 9090번 포트를 사용합니다.

5.2. 향후 계획 일정



6. 팀의 구성 및 팀원의 역할

- 프론트 엔드 팀

정성록: 기획, 모션 및 클라이언트 설계, 인 게임 모션 구현

서지우: 기획, 모션 및 클라이언트 설계, 인 게임 모션 구현

- 백 엔드 팀

문무현: 기획, 체스 로직 및 아이템 기능 구현, 서버 구현

전성윤: 기획, 체스 로직 및 아이템 기능 구현, 서버 구현

7. 참고문헌