

2020 협동분산시스템(3205)

기말 프로젝트

< CM Attack : 실시간 FPS게임 > [최종보고서]



201511178 김근욱

201511181 김상호

201511191 문병준

201511208 유선규

201511211 윤지호

[목차]

1. 팀원 정보 및 역할

- a. 팀원정보
- b. 역할
- c. 기여도

2. 프로젝트 소개

- a. 프로젝트 목표
- b. 프로젝트 구조

3. 프로젝트 디자인

- a. 프로젝트 설계
- b. CM API

4. 프로젝트 결과

5. YouTube & GitHub URL

1. 팀원 정보 및 역할

a. 팀원 정보

- i. 컴퓨터공학부 201511178 김근욱
- ii. 컴퓨터공학부 201511181 김상호
- iii. 컴퓨터공학부 201511191 문병준
- iv. 컴퓨터공학부 201511208 유선규
- v. 소프트웨어학과 201511211 윤지호

b. 역할

- i. 김근욱 : 조장 및 서버 이벤트처리
- ii. 김상호 : 서버 총괄, 서버-클라이언트 간 각종 이벤트 처리, 이벤트 구조 설계 및 이벤트 처리 함수 개발
- iii. 문병준 : 기획 총괄, 프론트엔드 개발, 자바 스윙 다이얼로그 제작, 인 게임 그래픽 갱신 구현, 서버-클라이언트간 클래스 설계, 디자인 리소스 제작
- iv. 유선규 : CM API 분석, 백엔드 구조 설계 및 개발, 서버-클라이언트간 클래스 설계
- v. 윤지호 : 프론트엔드 이벤트 처리, 클릭 키보드 동작 처리

c. 기여도

김근욱 20%

문병준 20%

김상호 20%

유선규 20%

윤지호 20%

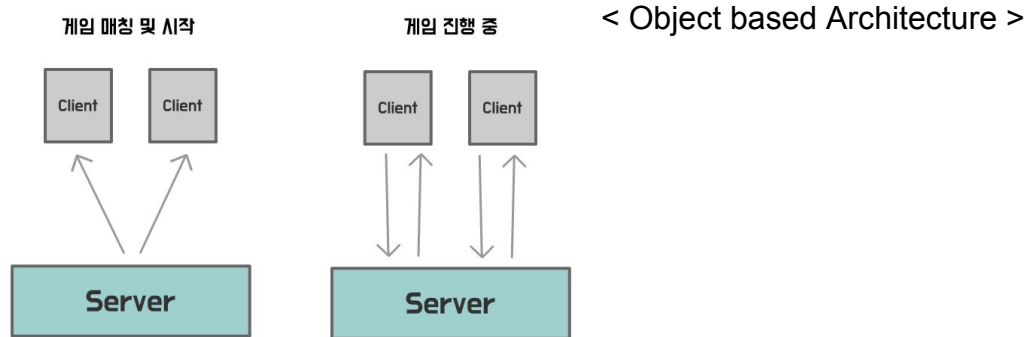
2. 프로젝트 소개

a. 프로젝트 목표

본 프로젝트는 클라이언트 - 서버 간의 통신을 이용하여 멀리 있는 사용자와 실시간으로 온라인 FPS(1인칭 슈팅) 게임을 즐길 수 있게 하는 것이 목표입니다. 각 플레이어는 주어진 체력에서 제한된 총알로 상대 플레이어의 체력을 모두 소진시켜 먼저 처치하면 승리하는 방식입니다.

기본적인 게임 방식은 2D화면상에 키보드를 이용 하여 좌우 이동 조작이 가능하도록 하고 마우스를 이용하여 총 쏘기가 가능하며 무기의 종류에 따라 연사속도, 데미지를 다르게 하려고 합니다. 그리고 상대방의 이동 정보, 사격 정보가 두 사용자 동시에 동작하도록 할 예정입니다.

b. 프로젝트 구조

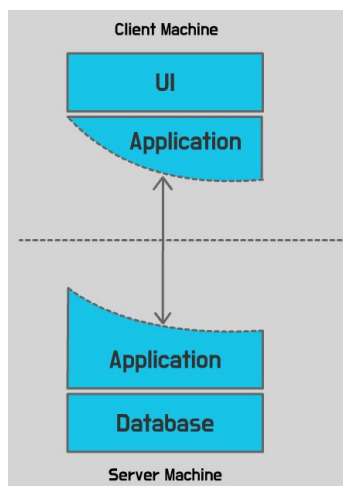


1. 게임 매칭 및 시작 시

- a. 두 개의 클라이언트가 게임에 접속 시, 서버가 최초로 한번 클라이언트의 정보(ip, 총기 종류, 닉네임 등)를 각 클라이언트들에게 전송해준다.

2. 게임 진행 중

- a. 실시간으로 클라이언트들은 인게임 정보(Player X좌표, Aim(총 조준점)좌표, HP, 생존여부)등을 서버에 전송하고 서버는 다른 클라이언트에게 그 정보를 전송한다.



<Centralized Arch : Multi-tiered architectures>

->클라이언트에서는 게임의 인터페이스(남은HP, 남은총알,총기 그래픽, 맵 이미지)등을 출력해주고, 총 사격 애니메이션, 저격총 줌인 등을 처리한다.

->서버에서는 각 Player의 X값 좌표, HP, 에임좌표등을 저장하며, 에임좌표와 상대방의 좌표를 계산하여 피격 판정을 전송하고, HP가 0이 될 시 사망여부를 전송한다.

<클라이언트 - 서버 최종 클래스 구조>

3. 프로젝트 디자인

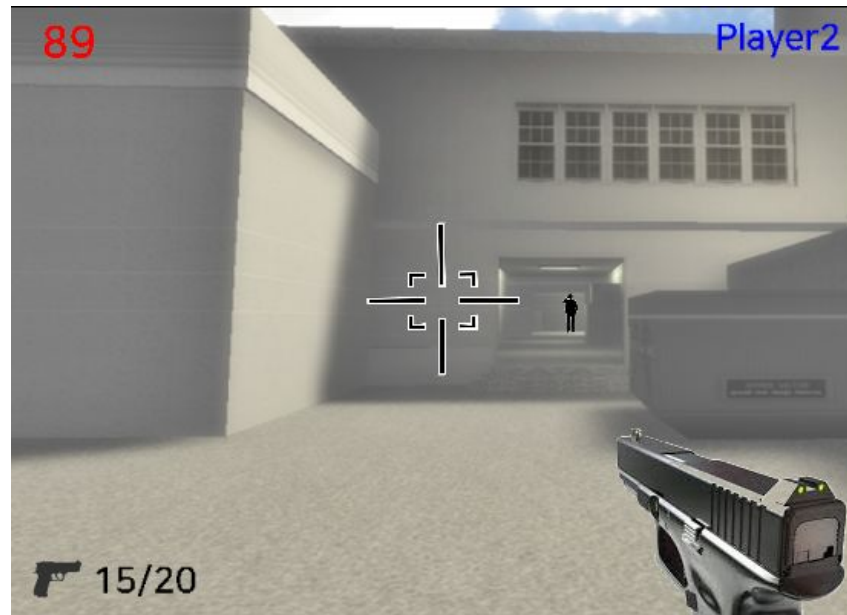
a. 프로젝트 설계

i. 게임방법

1. 해당 프로그램에 로그인하여 접속한다.
2. 2가지 총기(라이플, 저격총) 중 하나를 선택한다. 총 종류에 따라 연사속도와 데미지 다름
3. 총 고르기 이후 2명 모두 준비하기 버튼을 누르면 게임 자동매칭.
4. 이후 계속해서 접속하는 사용자에게 대해, 2명이 모일 때마다 게임 시작할 수 있도록 제공
5. 게임 화면으로 전환시 게임 시작.
 - 왼쪽 아래 남은 체력, 남은 총알 수 표시
 - 왼쪽 위 아이디 표시
6. 키보드, 마우스로 사용자 조작
 - A,D 키로 좌우 동작
 - 마우스 클릭 시 총알 발사

7. HP가 0이 되는 플레이어는 패배하고 상대는 승리하며 게임 종료.
8. 다시하기를 누르면 새로운 이름으로 로그인 가능하고 나가기를 누르면 로그아웃

ii. 게임화면(프로토 타입)



iii. 게임화면(최종)



iv. 프로그램 동작 과정

1. Server 실행.
2. Client 1 실행 및 Server 접속. Client 2 실행 및 Server 접속.
3. 접속된 양쪽 클라이언트 정보를 Server에서 관리.
 - a. Server 관리 정보 : 각 플레이어의 위치정보, HP정보
 - b. Client 관리 정보 : 보유 총알 수, 플레이어 HP
4. 사용자 입력 동작에 따라 Client 및 Server에서 관리 정보 연산 및 갱신.
 - a. 키보드 연산 : 키보드 조작이 발생하면 Server에서 사용자 좌표 값 변화 확인 후 갱신. 갱신된 좌표값에 따라 양쪽 플레이어에 보여지는 위치값 갱신.
 - b. 마우스 좌클릭 연산 : 해당 플레이어가 마우스 좌클릭시 서버에서 확인. Server는 전송된 현재 에임(총 조준점) 좌표와 상대 플레이어의 위치정보를 비교하여 타격 여부 확인 후 HP정보 갱신 및 총알 갯수 갱신.
5. Server에서 관리하는 Client 중 한 Client에서 HP 값이 0이 되면 게임 종료 메시지 출력.

b. 프로그램 구조

i. Client.java

1. Login() : 플레이어 접속시, Login화면인 Login다이얼로그 출력.
2. Game() : 플레이어 2명 매칭될 시, 게임화면인 InGame다이얼로그 출력.
3. GameOver() : 게임종료시, 결과화면인 Result다이얼로그 출력.

ii. CMClientEventHandler.java

1. processSessionEvent() : user_event의 ID에 따라 이벤트 처리.
2. updateGroupInfo() : 매칭 중인 client의 그룹정보 갱신.
3. updateHPInfo() : client의 hp정보 갱신 & 피격 사운드 출력.
4. gameOver() : client가 승자인지 패자인지 확인 후 result다이얼로그 출력하는 GameOver() 호출.
5. updateMove() : 각 client간의 이동정보에 따라 좌표 값인 PosX정보 갱신
6. attackSuccess() : 상대방을 피격할 시 피격 사운드 출력.

iii. Server.java

1. startCM() : 서버를 구동한다.
2. plusPlayerCount() : 현재 접속한 플레이어 수를 1 증가.
3. minusPlayerCount() : 현재 접속한 플레이어 수를 1 감소.

- iv. CMServerEventHandler.java
 - 1. processEvent() : session_event 및 user_event 처리.
 - 2. processEvent() : 게임플레이 관련 이벤트 처리하여 client에 이벤트 전송.
 - 3. joinplayer(): 새로운 플레이어가 로그인 했을 때 두 플레이어를 같은 그룹으로 매칭시켜준다.
 - 4. playGame(): 게임중 클라이언트의 이벤트(클릭, 이동)를 처리 해주는 함수
 - 5. endGame(): 게임이 끝났을때 플레이어들에게 이벤트 보내주는 함수
- v. GameManager.java
 - 1. removeGM() : 현재 GameManager에 null값을 넣는다
 - 2. getPM(): 플레이매니저 배열은 반환하는 함수
- vi. InGameDialog.java
 - 1. DrawStart(): 멀티스레드를 사용해 정해진 시간마다 화면을 업데이트 해주는 함수
 - 2. paint(): 화면에 게임에 필요한 이미지를 그려주는 함수
 - 3. KeyListener extends KeyAdapter : 키보드 버튼 입력 이벤트 리스너, a d 키에 따라 사용자 위치 정보를 전송
 - 4. MouseListener extends MouseAdapter : 마우스 클릭 입력 이벤트 리스너, 시간 값을 이용해서 연사속도를 조절하고 데미지 일정 범위에서 랜덤으로 발사, 총 반동 효과 주기 위해 총알 좌표 일정 범위 랜덤하게 바뀌어서 발사
 - 5. Play() : 음향 효과 재생함수
- vii. LoginDialog.java
 - 1. LoginDialog() : 로그인 화면을 보여주는 함수.
- viii. MatchDialog.java
 - 1. init(): 게임 매칭대기 화면 다이얼로그 초기화 및 표시
 - 2. Login(): Server에 로그인세션 이벤트 전송, 플레이어 객체 생성, 사용자 정보 담은 UserEvent 전송
- ix. ResultDialog.java
 - 1. actionPerformed : 재시작, 나가기를 결정하는 이벤트 리스너

c. CM API 사용

- i. CM_SESSION_EVENT : session이벤트가 로그인인지 로그아웃인지 확인.
- ii. CM_USER_EVENT : field의 그룹을 통해 게임 매치 전의 유저이벤트인지, 매치 후의 유저이벤트인지 확인.
- iii. send(CMEvent cme, String strTarget, int opt) : client와 server간의 통신에 사용. client는 server에게 이동, 사격 등의 정보를, server는 그 정보를 처리한 결과를 client에게 전달

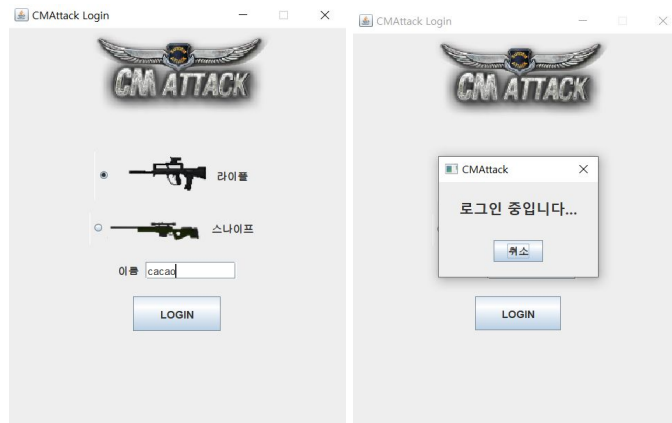
- iv. `getEventField` : client 혹은 server에서 전달 된 이벤트를 처리하기 위해 메시지를 해독
- v. `login` : 플레이어가 아이디와 충기를 선택하면 서버에 로그인.
- vi. `logout` : 플레이어가 게임을 그만두거나, 매치 중에 종료를 하면 서버에 로그아웃.

4. 프로젝트 결과

a. 서버실행

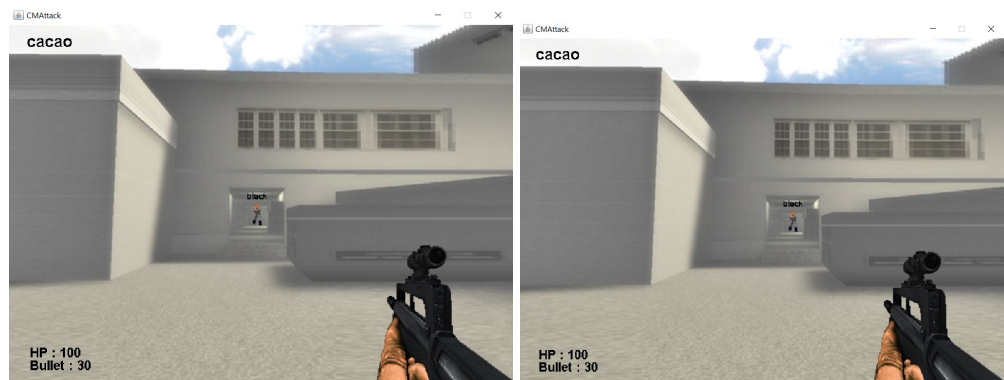
- 서버(Server.java)를 실행한다. 최소 2명의 접속자가 들어올 경우 게임 방을 생성하고 서로를 게임 방으로 매칭 시켜준다.
- 2명 뿐만 아니라 여러 명의 사용자가 접속하더라도 각각 매칭하여 게임을 진행시켜줄 수 있다. 즉, 2명 마다 방이 만들어져서 독립적으로 플레이 할 수 있다.

b. 로그인



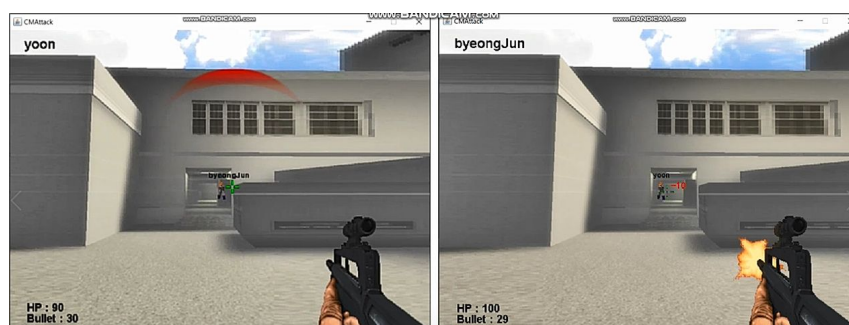
- 클라이언트(Client.java)를 실행한다. 한 명의 사용자가 로그인을 하면 다른 사용자를 기다리고 또 다른 사용자가 로그인하면 두 사람을 매칭하고 게임을 시작해준다. 무기의 종류를 선택할 수 있고, 자신의 이름을 적을 수 있다.

c. UI 및 좌우 이동



- 좌측 하단에 남은 체력과 총알이 표시되고, 좌측 상단에는 자신의 닉네임이 표시되며, 상대방의 머리 위에 상대방 닉네임이 출력된다. 커서는 조준점으로 바뀐다. a,d 키를 이용하여 좌우로 이동할 수 있고 나와 상대방에게 모두 움직임이 갱신된다.

d. 총 쏘기



- 마우스를 클릭하여 총을 쏘면 연사속도에 맞춰 총이 나가고 총알 수가 갱신된다. 총을 맞출 시 상대방에게 가해지는 데미지가 보인다. 총을 맞을 시에는 인디케이터 표시가 나오고 hp가 갱신된다.
- 총알의 데미지는 일정하지 않고 특정 범위 안에서 랜덤 값을 가진다.
- 반동이 존재하며 조준 점의 위치에서 조금 벗어나게 총알이 발사 될 수 있다.

e. 승패 알림 및 다시하기 기능



- 상대방의 HP를 먼저 0을 만들면 승리한다. 다시하기를 누르면 새로운 아이디와 무기타입을 정해서 다시 로그인을 할 수 있다.

f. 총 종류 별 데미지, 연사속도 변경



- 총 종류에 따라 데미지와 연사속도가 다르다.

5. Youtube and Git URL

- https://github.com/cacao518/CM_Attack
- https://www.youtube.com/watch?v=NT5deylGM_M