Term Project Report

유민석 2023312106

박정휴 2023311073

1. Introduction

저희가 만든 프로젝트는 소켓 통신을 통한 대결 게임입니다. 총 두명이서 게임을 진행하고, 한명 당 하나의 토개피(포켓몬)이 주어집니다.

대결은 두 토개피 중 어느 토개피의 체력이 0이 되거나, 도망갈 경우, 종료됩니다.

토개피는 매 턴마다 다음 4가지의 행동중 하나를 할 수 있습니다.

첫번째 행동은 1. 손가락 흔들기 입니다. 이는 각 유저의 토개피가 프로그램 상에 존재하는 random스킬을 상대유저의 토개피에게 사용하는 행동입니다. 손가락 흔들기를 통한 공격은 해당 random 스킬의 type과 상대 토개피의 type에 따라 서로 다른 데미지 및 상태이상 효과를 입히는 기술입니다.

두번째 행동은 2. 상처약 사용하기 입니다. 이는 자신의 토개피의 현재 체력을 50만큼 올립니다

세번쨰 행동은 3. 스텟 보기 입니다. 이는 현재 상황에서, 자신의 토개피의 능력치 및 상태를 보여줍니다. 스텟 보기를 택할 경우, 이는 직접적인 행동으로 간주되지 않으며 이후 다른 행동을 취할 수 있습니다.

네번쨰 행동은 4. 항복하기 입니다. 이는 대결에서 항복하는 것으로, 항복하기를 고른 토개피의 유저는 게임에서 패하게 됩니다.

1. Class소개

이 게임에서 사용된 class는 총 7개 입니다.

1. Server class

Server.java 안에 있는 class입니다. Server class의 기능은 다음과 같습니다.

1. Client 접속 받기

Server class의 main함수는 method인 start 함수를 호출합니다. 이 start 함수의 역할은 통신을 위한 socket을 열고, 이를 통해 client의 접속을 받습니다. Client가 연결되면 해당 클라이언트에 해당하는 ReceiveThread를 생성하고 이를 run합니다.

1. Client의 개수 세기

이 게임에 참여하는 client는 총 2개입니다. 따라서, 접속된 client의 수가 2개가 될 시에, 연결된 두개의 client에게 게임이 시작되었다는 메시지(문구)와 타입을 입력하라는 메시지를 전달합니다.

1. ReciveThread

Server.java안에 있는 ReciveThread class의 주된 기능은 다음과 같습니다.

1. Client로부터 입력 받기.

ReciveThread는 각각 하나의 client와 연결이 돼있습니다.

While-loop을 통해서 client로부터 끊임없이 입력을 받습니다. 하지만 게임의 특성상 서버에 연결된 두 클라이언트는 매 턴 마다 한번씩 입력을 해야합니다. 따라서 모든 ReciveThread 객체가 공유하는 static 변수 c(여태까지 client로부터 들어온 입력의 개수를 위한 변수)를 도입해서, 모든 입력이 들어올떄마다 c를 알맞게 조절하고, 두 클라이언트로부터 입력이 완료된다면, game\_manager 을 통해 두 토개피의 대결이 이루어지고, 그렇지 않을 경우에는 입력이 완료된 client에게만 ‘입력대기’라는 문구를 띄웁니다.

1. Game\_manager 호출 후, 대결 진행 화면을 각각의 client에게 보내주기.
2. 에서 말했듯이, 기본적으로 이 프로그램은 두 클라이언트로부터 입력이 완료되는 것을 한 turn의 개념으로 생각합니다. 따라서 두 클라이언트로부터 각각 토개피의 행동이 입력되면, game\_manger 객체가 해당 행동들과 두 토개피의 상태를 바탕으로 대결을 진행합니다. GameManager는 대결진행 상황을 해당객체의 public으로 처리된 member 변수에 저장합니다. 따라서 이를 참조하여 게임 진행 상황 및 결과를 각 client에게 보내줍니다.
3. GameManager class

GameManager class의 기능은 다음과 같습니다.

1. 선공권 정하기

각 대결의 turn마다, 어떤 토개피의 행동이 먼저되어야하는가에 대한 기준은 다음과 같습니다. 토개피 오브젝트는 각각의 speed를 지니는데, speed가 빠른 토개피가 먼저 행동할수 있는 선공권을 가집니다. Speed가 같을 경우엔 random으로 선공권을 정합니다.

1. 대결 진행상황을 저장하기

선공권이 정해진 뒤에는, 순서에 맞게 두 토개피가 client로부터 입력된 행동을 개시합니다.(pocketmon class의 move함수를 통해서). 이떄, 각각의 토개피의 상태와 행동의 설명은 ouput1,2 변수에, 행동의 결과는 middleScene1,2 변수에 저장됩니다. (output1,2 middleScene1,2는 모두 GameManager의 멤버 public member 변수 입니다. 저장된 이들의 정보는 RecieveThread에서 참조됩니다.)

1. Pokemon class

Client의 아바타, 즉 토게피에 대한 정보가 담긴 클래스입니다. 다음은 주요 기능 설명입니다.

* 1. 주요 프로퍼티 설명  
       
     \* 이름: 포켓몬의 이름을 저장합니다.  
     \* 능력치 및 초기 능력치 : HP, AD(공격력), AP(특수공격력), DF(방어력), MR(특수방어력), SP(speed, 스피드) / 각각 초기 능력치 200, 100, 100, 50, 50, 100  
     \* 상태 : 토게피의 현재 상태를 나타냅니다. 정상, 화상, 독, 맹독, 마비, 잠듦, 얼음 의 상태를 각각 대응되는 int로 저장합니다.  
     \* 타입 1, 2: 포켓몬의 타입입니다. 포켓몬의 타입은 상대방의 공격을 받을 때, 해당 공격의 타입(기술 타입, 2-e 참고)과 비교하여, damage값을 조정합니다.  
     \* skillList: 포켓몬이 사용할 수 있는 기술을 담은 리스트입니다. 해당 포켓몬이 행동할 때(d-2), 사용자 input에 따라서 무작위로 리스트 요소 중 하나가 사용될 수 있습니다. 스킬에 대해서는 2-e에서 설명합니다.  
     \* enemy: 상대 포켓몬 객체를 저장하는 프로퍼티 입니다. 해당 프로퍼티를 통해 상대방 포켓몬의 저장 값들을 엑세스합니다.  
     \* typeList: 포켓몬이 가질 수 있는 모든 타입들이 String으로 저장되어 있습니다.

1. Constructor  
     
   Client의 입력에 따라, 토게피의 이름 및 첫 번째 타입을 설정합니다. (타입에 대한 정보는 아래에서 설명) 이후에, 토게피가 가지는 두 번째 타입은 무작위로 설정됩니다.
2. 포켓몬의 행동 결정  
     
   사용자의 입력 중, 1-손가락 흔들기, 2-상처약 사용하기 에 따른 포켓몬의 행동을 결정합니다.   
   1번의 경우, 해당 토게피가 사용하게될 기술을 무작위로 결정합니다. 그 이후, 해당 기술이 가진 정보들 및 자신의 정보(타입, 능력치 등)을 통하여 상대에게 줄 damage, condition 및 부가효과에 대한 계산을 수행합니다.   
   2번의 경우, 상처약을 사용합니다.(3-c-2)
3. 스탯 전송  
     
   해당 포켓몬의 스텟을 String의 형태로 반환해주는 메서드를 호출할 수 있습니다.
4. Skill class

Skill class는 포켓몬이 사용하는 기술에 대한 정보를 담고 있습니다. 기술들은 각각 고유 이름, 타입, 데미지, 상태 등이 상이하게 포함됩니다. 기술을 그 종류에 따라, AttackSkill(공격 기술), ConditionalSkill(상태이상 기술), Barrier(방어 기술)이 존재하며, 각각 Skill class의 파생 클래스의 형태로 존재합니다.   
\* AttackSkill은 타입, 데미지 및 부과효과를 설정할 수 있고 반환할 수 있습니다.  
\* ConditionalSkill은 포켓몬의 상태를 결정합니다.  
\* Barrier는 상대 포켓몬의 공격의 damage를 0으로 만듭니다.

1. Multy\_Client class

Multy\_Cleint class의 주요한 기능은 다음과 같습니다.

1. 서버와 연결

Client는 sever와 연결되어있어야 합니다. 따라서 소켓을 통해 server와의 연결을 진행합니다.

1. Server로부터 데이터 전송 받기

While-loop은 서버로부터 데이터를 전송받고, 전송 받은 데이터를 해당 클라이언트의 콘솔에 스트링형식으로 띄웁니다.

1. SendThread 호출하기.

각각의 클라이언트는 서버에게 데이터를 전송받는것만 하는게 아니라, 데이터를 전송할 수 있어야합니다. 따라서 sendThread라는 오브젝트를 생성하고 run해주어서 server에게 데이터를 보낼 수 있도록 합니다.

1. sendThread

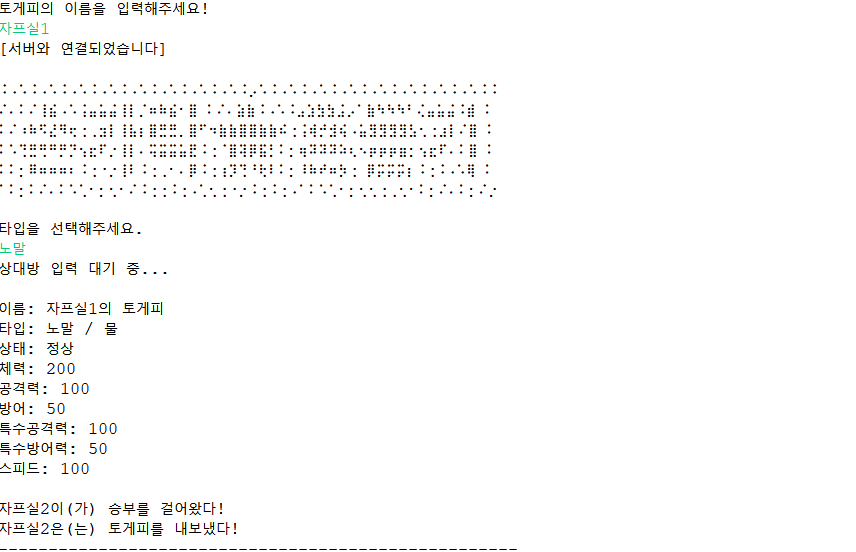
SendThread class의 주요한 기능은 다음과 같습니다.

1. 서버에게 데이터 보내기

SendThread class의 기능은 단 한가지 입니다. Client에서 sever로 데이터를 전달하는 것입니다.

1. In-Game\_Scene
2. Game-start Scene.

Multy\_client를 두번 실행하고, 각각의 터미널에 토개피 이름을 입력한다.

그러면 서버에 연결되고 아래와 같은 게임이름 ‘손가락 흔들기’ 라는 문구가 뜨고, 자신의 토개피의 타입을 설정한다(올바르지 않은 입력은 노말로 처리된다.) 이후 대결 시작 문구가 콘솔에 출력되고, 게임이 시작된다.텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

(자프실 1 클라이언트 콘솔) (자프실 2 클라이언트 콘솔)

1. In-game Scene
2. 사용자로부터 입력받기

각 클라이언트는 행동을 하기 전, 자신의 토개피가 해야할 행동을 입력해야 한다. 아래의 사진은 각 클라이언트에게 행동 입력을 기다리는 창이다. (현재 토개피들의 상황과, 행동테이블이 각각의 클라이언트에게 전달된다,)텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명  
(자프실 1 클라이언트 콘솔) (자프실 2 클라이언트 콘솔)

1. 행동 수행하기.

B단계에서 각각의 토개피들의 행동을 입력 받았다. 이후에는 토개피들의 행동을 진행과 결과를 각각의 클라이언트에게 보내준다.

1. 1. 손가락 흔들기

랜덤 기술을 실행한다.

(자프실 1 클라이언트 콘솔) (자프실 2 클라이언트 콘솔)텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명 \* 이후의 행동 결과(이 화면은 두개의 행동을 모두 마친 후 각각의 클라리언트에게 그 다음 행동을 받는 화면이다. 손가락 흔들기 이후의 상황을 보여주기 위한 자료이다.)

1. 상처약 사용하기

위에서는 1. 손가락 흔들기를 선택했기 떄문에, 사용한 기술에 대한 설명이 출력된다면, 2. 상처약 사용하기는 아래와 같이, 간단하게 현재 토개피의 상태와, 체력이 증가되었다는 문구만을 띄운다.  
텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

1. 스텟보기

스텟보기를 선택할 경우, 아래와 같이 현재 자신의 토개피의 정보를 보여준다. 이후에는 다시 다른 행동을 선택할 수 있도록 입력을 다시 받을 수 있다.(스텟보기는 턴이 넘어가지 않는 행동으로 간주된다.)텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진

자동 생성된 설명

1. 항복하기

항복하기를 선택하면, 항복하기를 입력한 클라이언트의 토개피는 패배, 다른 토개피는 승리하게 된다.