객체지향설계프로젝트

141410 박현서
142032 최유빈
145191 정가인

프로그램 이름

숲 속의 캔버스

개발 동기

1. 평소에 추측성게임을 즐겨하면서 이 게임을 응용시켜 프로그램을 만들었으면 하는 생각을 했다.
2. 여러 장르를 브레인스토밍을 통해 회의한 결과 그림 퀴즈 형식의 게임을 선정.
3. 친구들과 소수로 즐길 수 있으며 많은 시간을 투자하지 않아도 되는 간단한 게임 선호.
4. 기존 형식의 게임 진행 시간이 길어 지루함을 느껴 좀 더 빠르고 많은 사람들이 그릴 수 있는 형식의 게임을 생각하게 됨.
5. 청소년을 타겟으로 한 아기자기한 게임을 만들고자 함.

제공할 기능

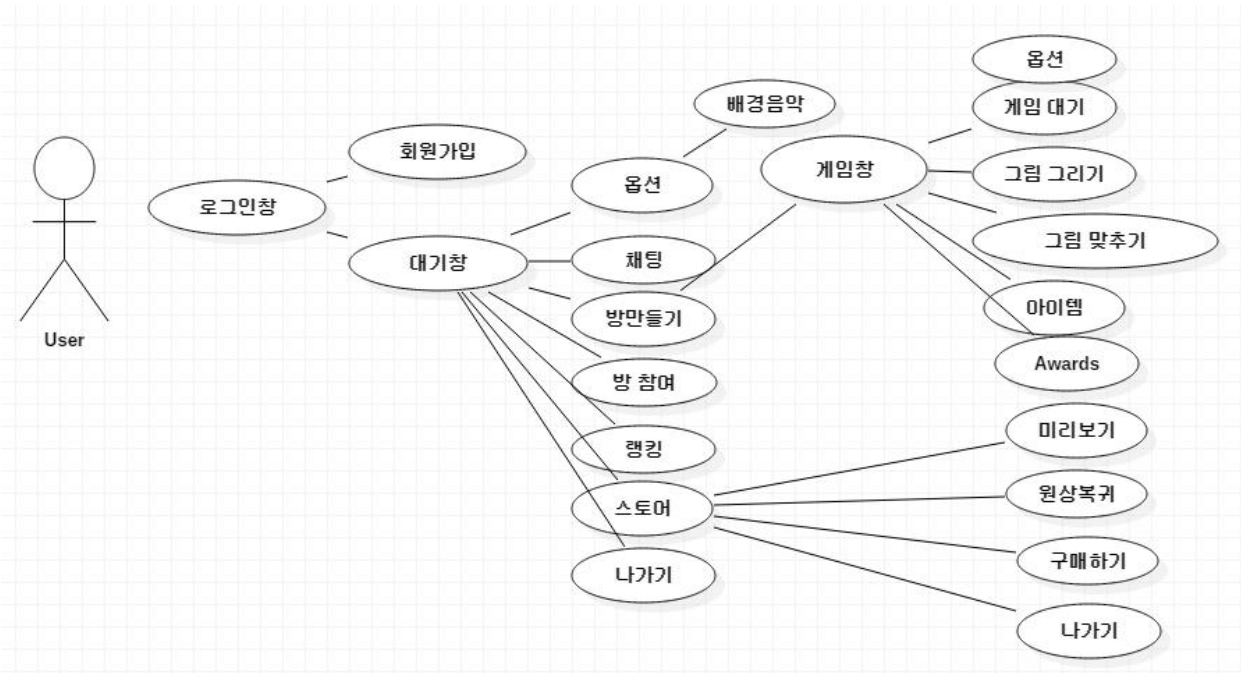
1. 클라이언트를 지원할 수 있는 서버 생성
2. 소켓을 이용하여 서버에 로그인
3. 4인용 게임 방 최대 6개까지 생성
4. 채팅
5. 배경음과 효과음 추가
6. 브로드캐스팅 이용한 공지
7. 제한시간 안에 그림 그리기
8. 정답자에 따른 점수 분배
9. 게임 진행 순서 정하는 미니 게임
10. 게임 아이템 생성 (시간 증가, 점수 두 배, 정답 창 금지)
11. 상점

12. 랭킹

13. 옵션

14. 초대

유즈케이스 다이어그램



세부 유즈케이스

1. 로그인 창에서 ID, IP를 입력하여 로그인

2. 메인 창에서 최대 6개까지의 게임 방 생성 또는 참여

3. 4명이 모였을 때 방장이 START 버튼을 누르면 게임 시작

4. 1~4까지 적힌 숫자 카드 게임을 통해 게임 순서 정하기

5.1 1번 : 문제를 제시 후 제한 시간 내(30초)에 일부 그림을 그린다.

2번 : 정답자가 없을 경우 제한 시간(20초) 동안 1번 그린 그림을 이어 그린다.

3번 : 정답자가 없을 경우 제한 시간(10초) 동안 2번 그린 그림을 이어 그린다.

5.2 1번이 그림을 그릴 때 2,3,4 정답 제출 가능

2번이 그림을 그릴 때 3,4 정답 제출 가능

3번이 그림을 그릴 때 4 정답 제출 가능

5.3 정답을 맞추지 못해 다음 차례로 넘어갈 경우 그림과 정답이 함께 전달

6. 사용자가 정한 라운드(5~10) 진행 후 점수 합산 및 발표

6.1. 한 라운드에 정답자가 없을 경우 다음 라운드 진행

6.2. 점수 배분 방법

6.2.1. 1번이 그릴 때 맞춘 경우 : 3점

- 1번의 점수 : 3점

6.2.2. 2번이 그릴 때 맞춘 경우 : 2점

- 2번의 점수 : 2점

6.2.3. 3번이 그릴 때 맞춘 경우 : 1점

- 3번의 점수 : 1점

7. 옵션에 배경 음악과 효과음 조절 기능

8. 유저에게 아이템 분배

8.1. 아이템의 종류

- 정답 제출 금지
- 시간 증가하기
- 점수 두 배

8.2. 게임 시작 후 모든 유저에게 모든 아이템을 1개씩 분배.

8.3. 아이템은 상시 사용 가능

9. 그림 도구

9.1. 색 상 : 빨강, 노랑, 초록, 파랑, 검정

9.2. 선 두께 조절

9.3. 지우개

10. 상점

10. 1 구매 기능

10. 2 미리 보기

11. 랭킹

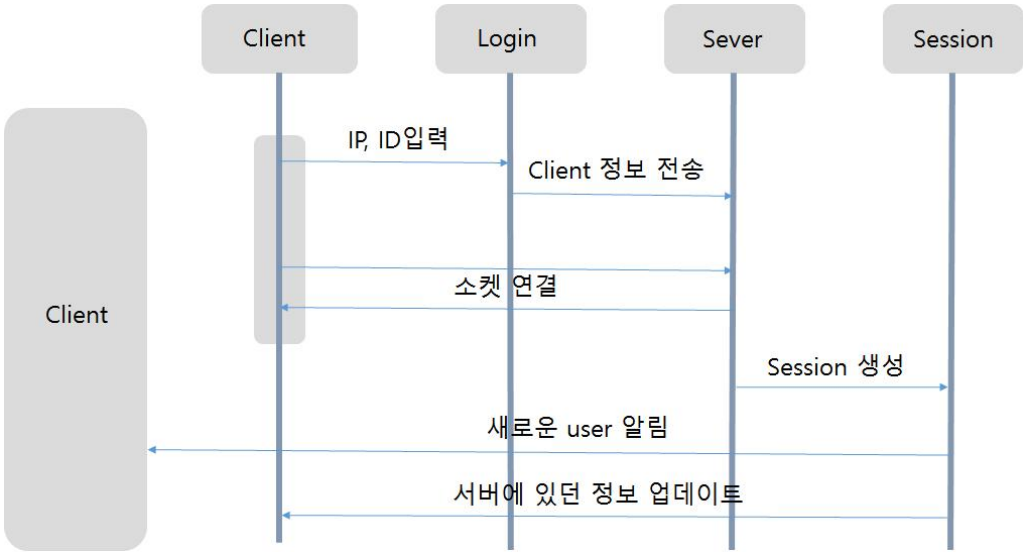
11.1 금, 은, 동메달 순위

요구 사항 리스트

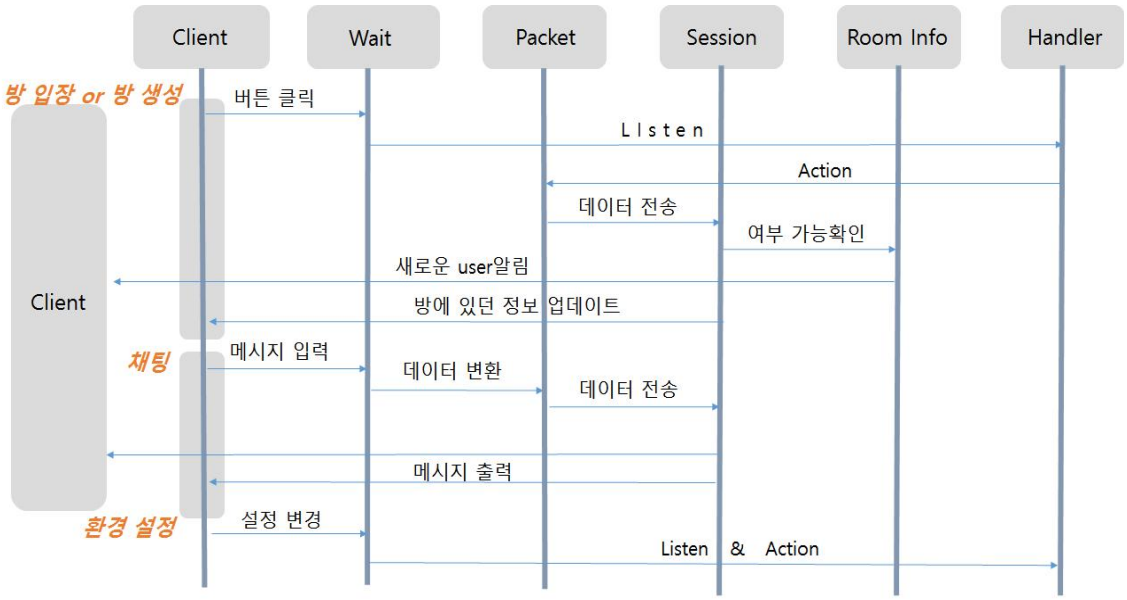
1. 유저 초대 기능 추가
2. 게임 분위기와 어울리는 GUI 꾸미기
3. 지우개 기능 심화
    - 3.1 전체 지우기
4. 옵션 기능 추가
    - 4.1 쪽지 수신 거부
    - 4.2 초대 거부
    - 4.3 배경음과 효과음 소리 조절 기능
5. 타이머 표시

시퀀스 다이어그램

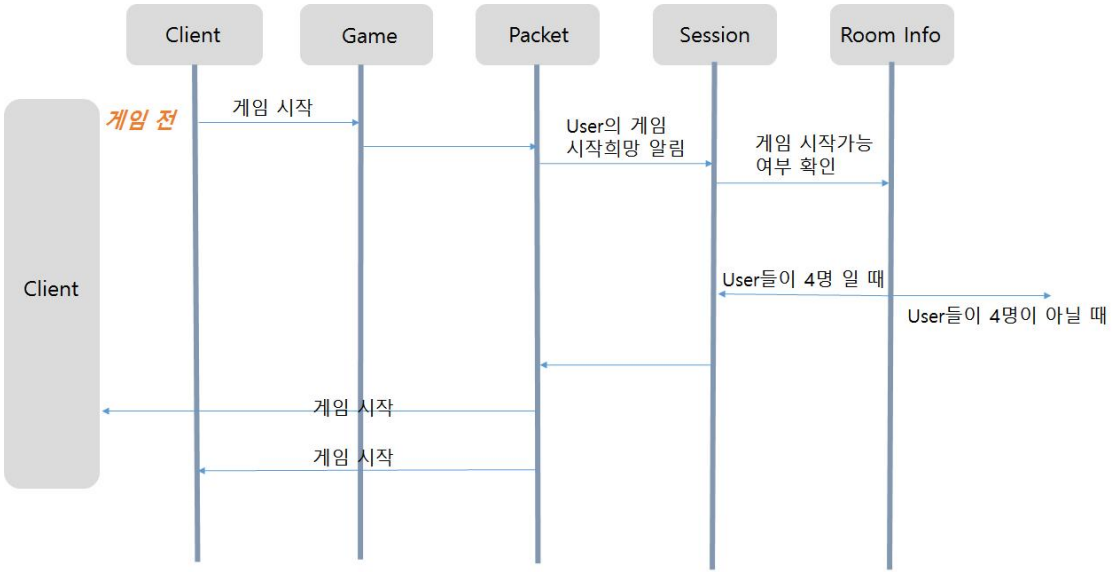
– 로그인창



– 대기실



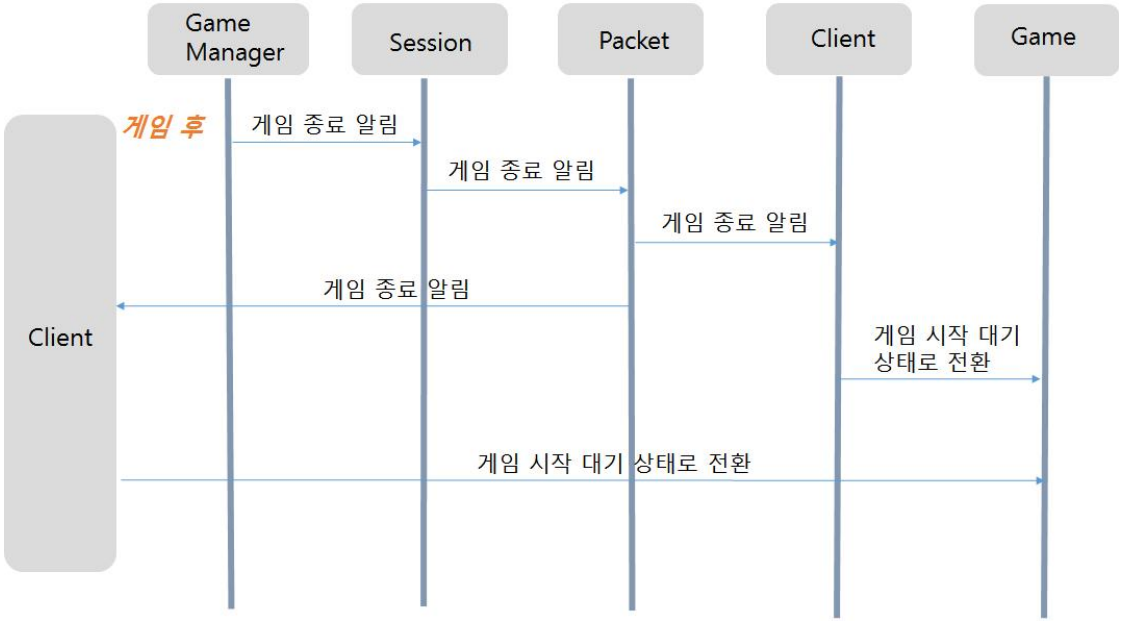
– 게임 전


Client
Game
Packet
Session
Room Info
게임 전
게임 시작
User의 게임
시작희망 알림
게임 시작가능
여부 확인
Client
User들이 4명 일 때
User들이 4명이 아닐 때
게임 시작
게임 시작

– 게임 중

GameManager
Minigame
Gaming
Session
게임 중
게임순서 정하기
순서 알림
게임 시작
Client
그림 그리는 상태 알림
스코어 알림

- 게임 종료 후

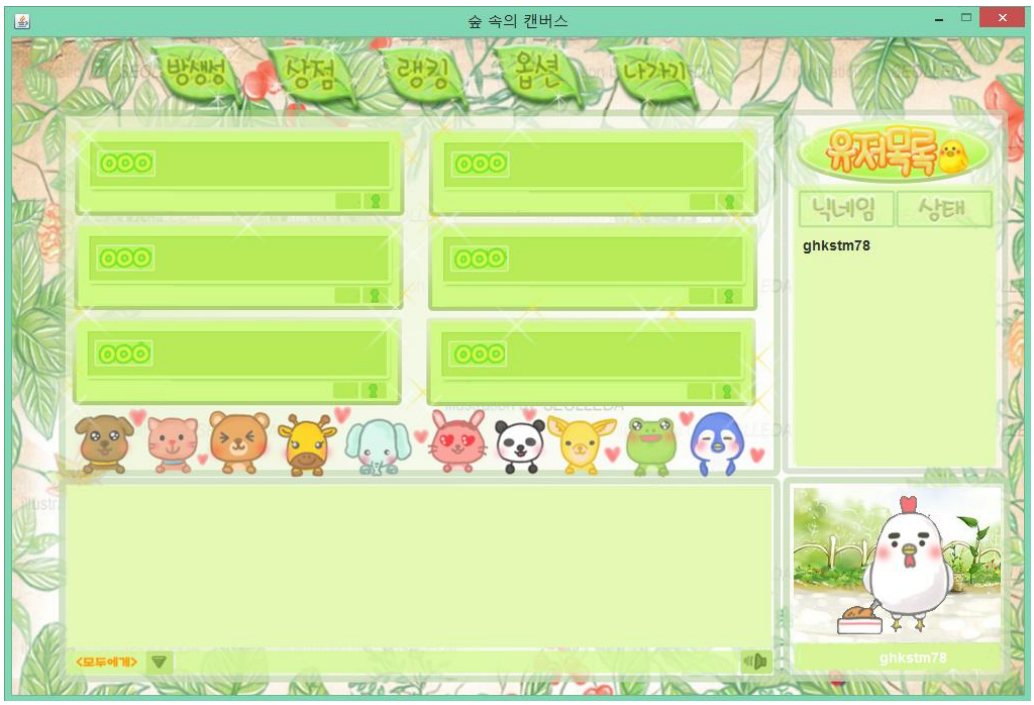


- Shop & Ranking

화면정의서

로그인 화면

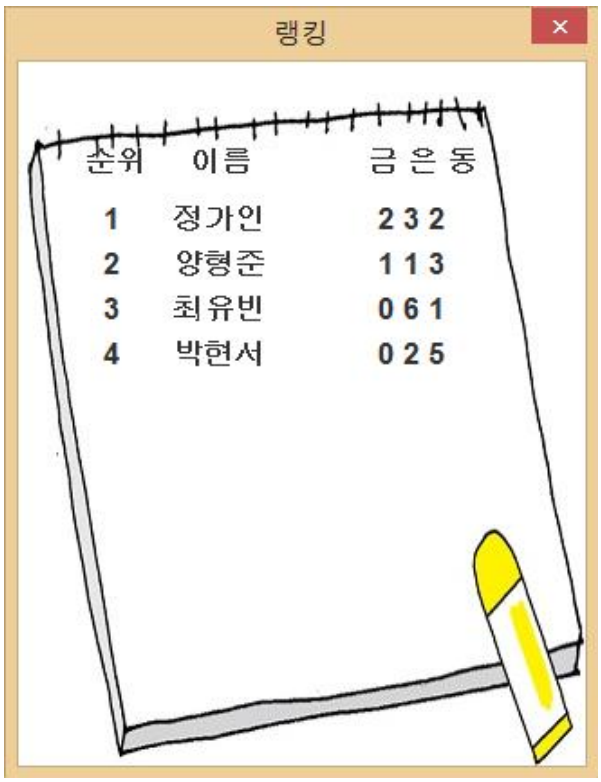
대기화면



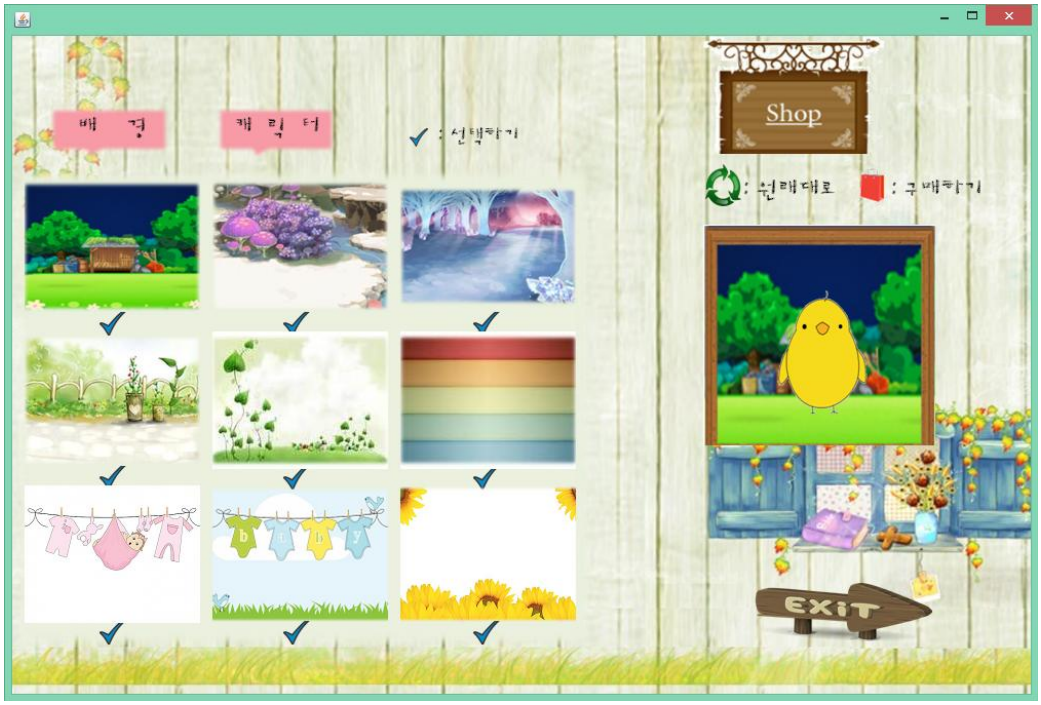
게임 창

숲 속의 캔버스
사람
EXIT
순위 이름 점수
2 박현서 0
3 최유빈 0
4 양형준 0
박현서
양형준
정가인
최유빈

랭킹

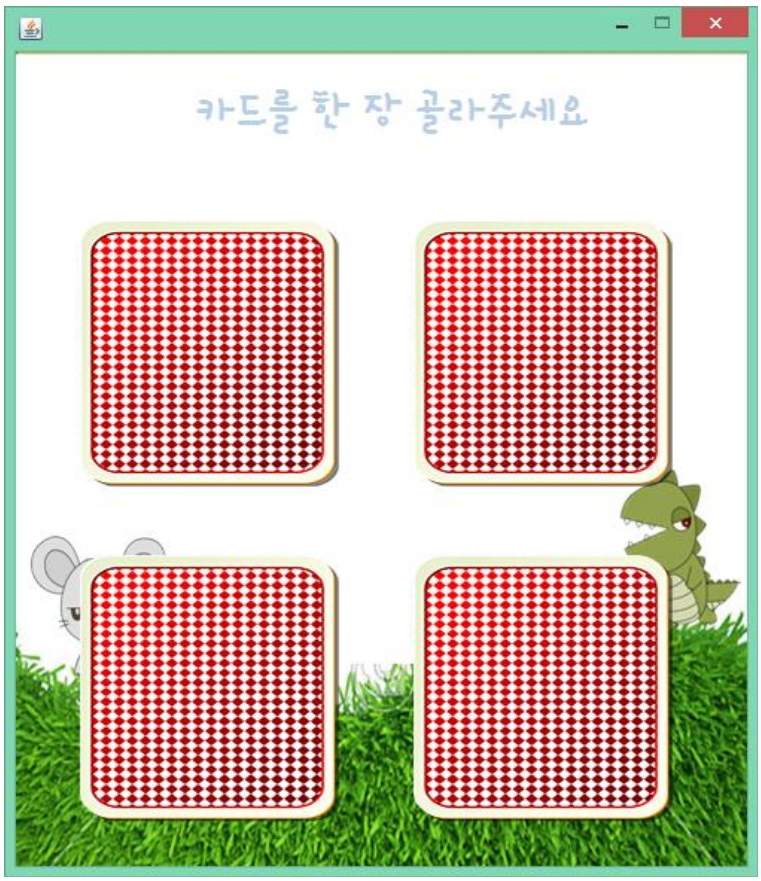

랭킹
순위 이름 금은동
1 정가인 232
2 양형준 113
3 최유빈 061
4 박현서 025

상 점

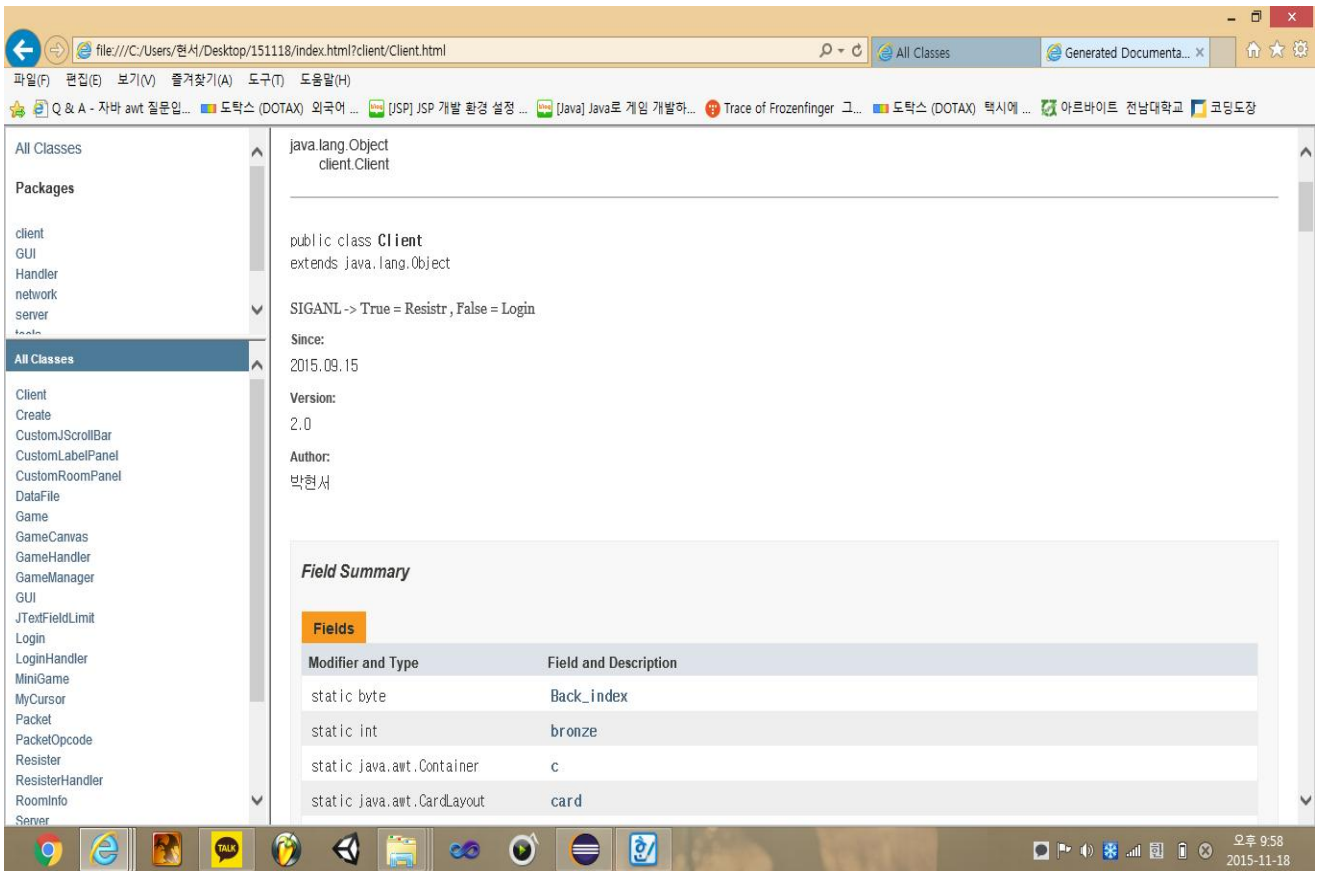


옵션

카드 뽑기

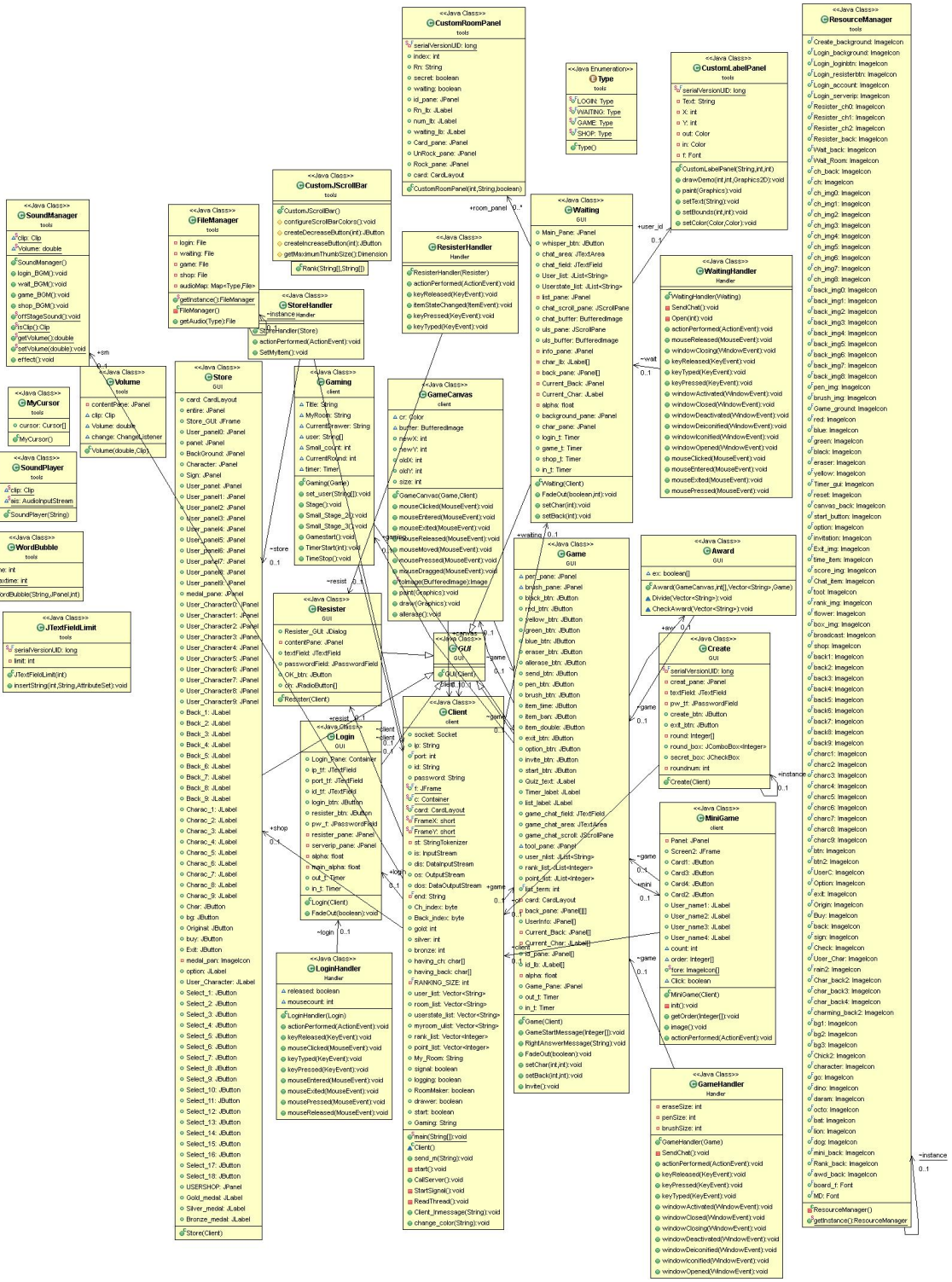


API 매뉴얼(Javadoc 사용) 초기페이지 스크린샷

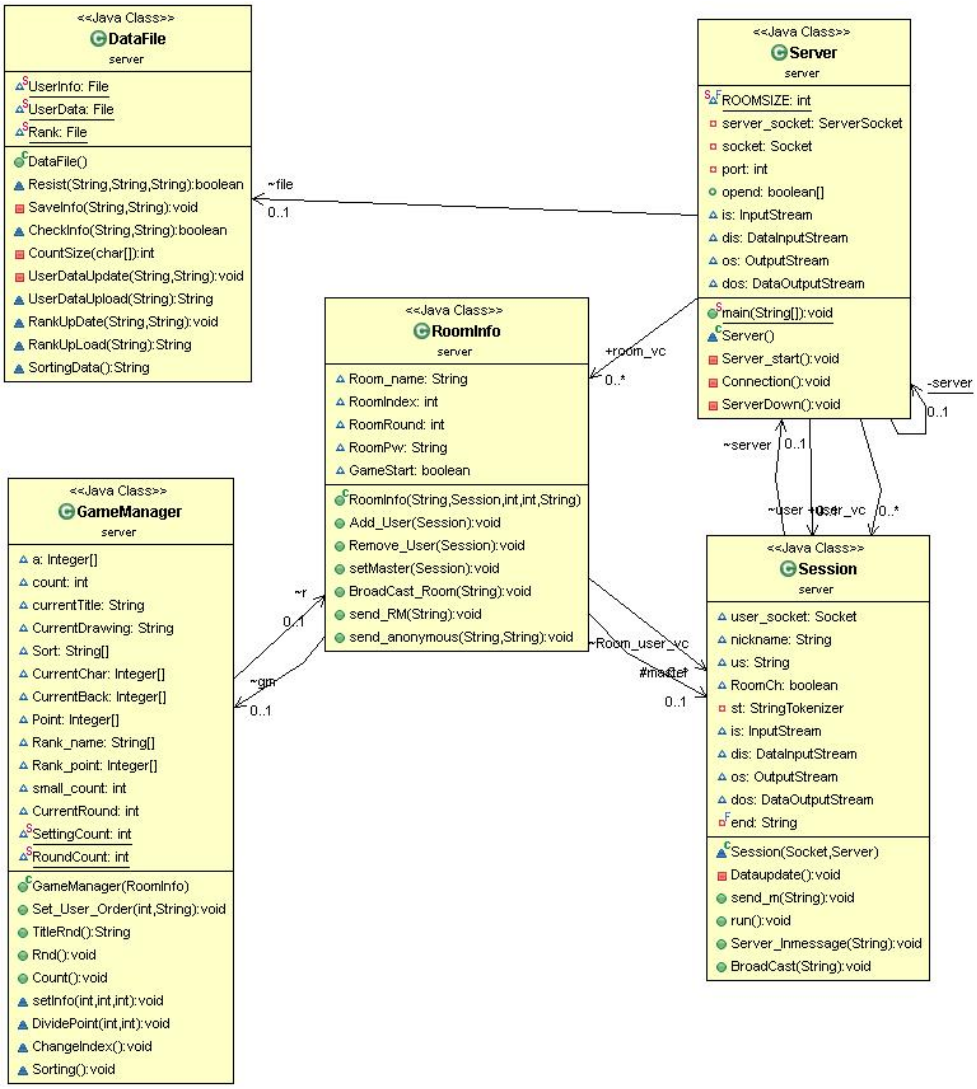


클래스 다이어그램

<Client>



<Server>


<<Java Class>>
DataFile
server
UserInfo: File
UserData: File
Rank: File
DataFile()
Resist(String,String,String):boolean
SaveInfo(String,String):void
CheckInfo(String,String):boolean
CountSize(char[]):int
UserDataUpdate(String,String):void
UserDataUpload(String):String
RankUpDate(String,String):void
RankUpLoad(String):String
SortingData():String
~file
0..1
<<Java Class>>
Server
server
ROOMSIZE: int
server_socket: ServerSocket
socket: Socket
port: int
opend: boolean[]
is: InputStream
dis: DataInputStream
os: OutputStream
dos: DataOutputStream
main(String[]):void
Server()
Server_start():void
Connection():void
ServerDown():void
-server
0..1
<<Java Class>>
RoomInfo
server
Room_name: String
RoomIndex: int
RoomRound: int
RoomPw: String
GameStart: boolean
RoomInfo(String,Session,int,int,String)
Add_User(Session):void
Remove_User(Session):void
setMaster(Session):void
BroadCast_Room(String):void
send_RM(String):void
send_anonymous(String,String):void
+room_vc
0..*
~server
0..1
~user_vc
0..*
<<Java Class>>
GameManager
server
a: Integer[]
count: int
currentTitle: String
CurrentDrawing: String
Sort: String[]
CurrentChar: Integer[]
CurrentBack: Integer[]
Point: Integer[]
Rank_name: String[]
Rank_point: Integer[]
small_count: int
CurrentRound: int
SettingCount: int
RoundCount: int
GameManager(RoomInfo)
Set_User_Order(int,String):void
TitleRnd():String
Rnd():void
Count():void
setInfo(int,int,int):void
DividePoint(int,int):void
ChangeIndex():void
Sorting():void
~r
0..1
~gm
0..1
<<Java Class>>
Session
server
user_socket: Socket
nickname: String
us: String
RoomCh: boolean
st: StringTokenizer
is: InputStream
dis: DataInputStream
os: OutputStream
dos: DataOutputStream
end: String
Session(Socket,Server)
Dataupdate():void
send_m(String):void
run():void
Server_Inmessage(String):void
BroadCast(String):void
~Room_user_vc
#master
0..1

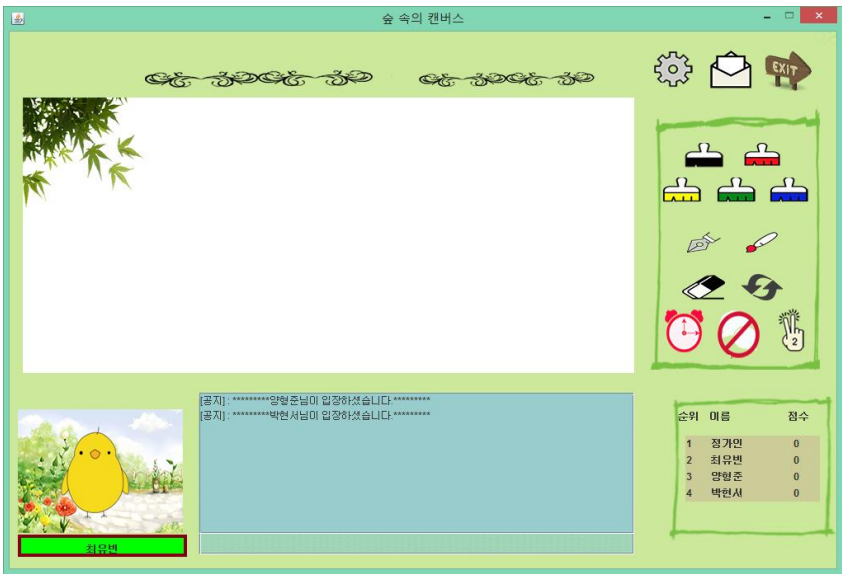
프로그램 구동 스크린 샷

1. 회원가입 및 로그인

2. 대기창

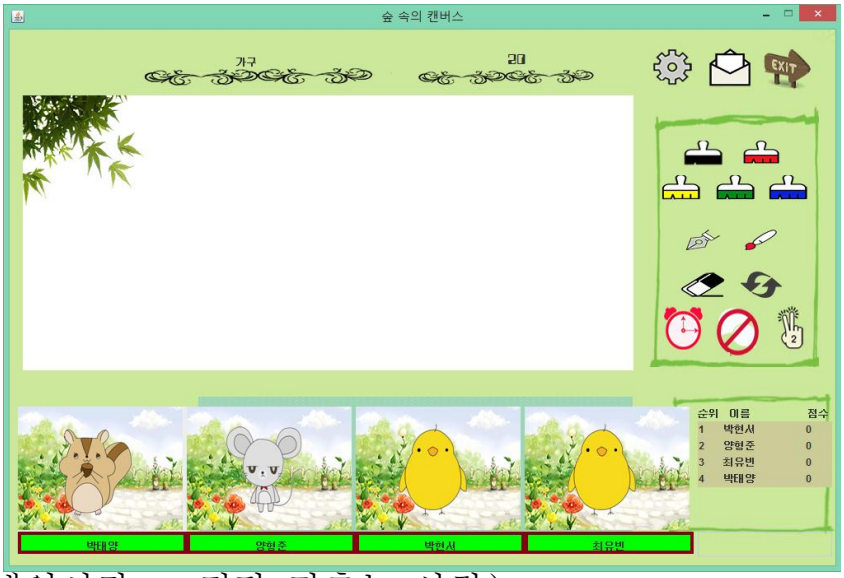
3. 상점

4. 게임 창 (게임 시작 전)

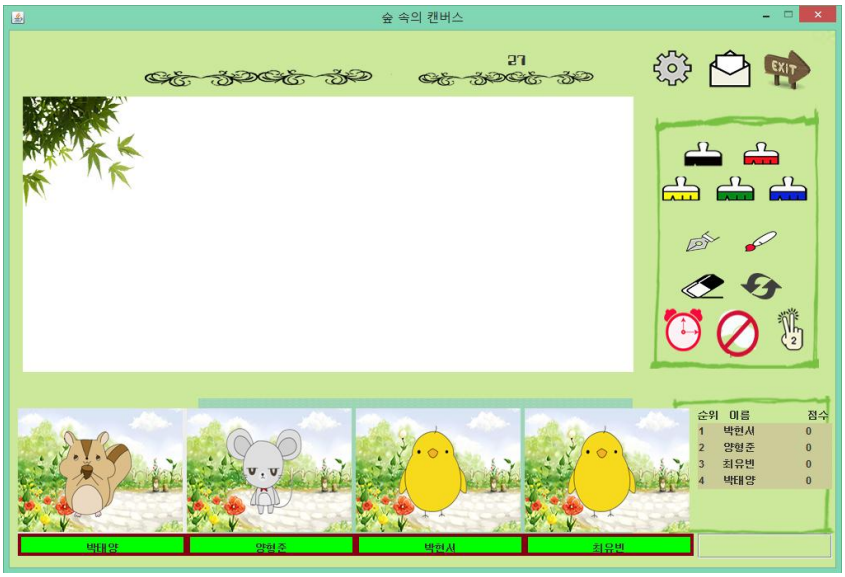


(게임 시작 후 – 유저 4명이 모이면)

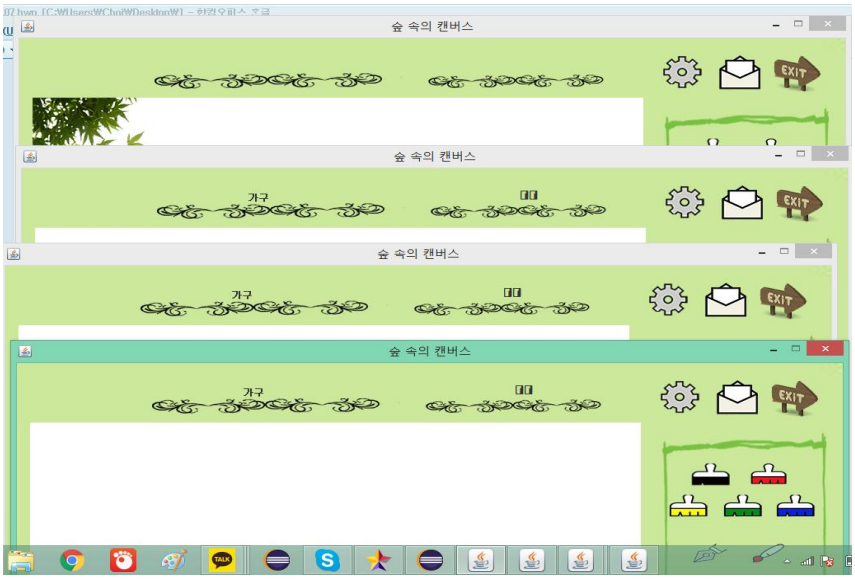
(게임 시작 후 – 그림 그리는 사람)



(게임시작 – 정답 맞추는 사람)



(게임 중 – 3번째 사람까지 그림)



5. 게임 종료 Awards

기능 구현 완료

1. 통신
    - 1.1 ServerSocket을 열고 accept() Thread 생성
    - 1.2 Client가 접속요청을 하고 확인이 되면 Session Thread(Client와 소통하는) 생성
    - 1.3 Input,output Stream 연결

2. 파일 저장
    - 2.1 UserInfo와 UserData디렉토리를 만든다.
    - 2.2 회원가입 시 아이디와 비밀번호를 UserInfo파일에 저장(같은 아이디가 존재 할 경우 중복메세지)
    - 2.3 로그인 할 때 UserData에 저장된 데이터를 받는다.

3. GUI <<Tool>>
    - 3.1 로그인 Scene
    - 3.2 대기실 Scene
    - 3.3 CustomClass
    - 3.5 게임 Scene
    - 3.6 스토어 Scene
    - 3.7 랭킹 Scene
    - 3.8 옵션 Scene
    - 3.9 Award Scene

4. Handler
    - 4.1 로그인 Handler
    - 4.2 대기실 Handler
    - 4.3 Resister
    - 4.4 Create(방 생성)
    - 4.5 게임 Handler
    - 4.6 스토어 Handler

5. Canvas
    - 5.1 색깔, 두께 조절가능
    - 5.2 지우개, 전체지우개
    - 5.3 게임이 시작하고 나서 그림을 그리기 때문에 BufferedImage를 통해 Client마다 그림을 그리는 순간 저장한다.

6. 말풍선 모듈
    - 6.1 Timer를 만들어 불 투명도를 시간이 증가함에 따라 감소하게 만듦

7. GameManager로 전체 게임 상황 총괄
    - 7.1 각 방마다 가지고 있다.
    - 7.2 각 방마다 유저 4명이 채워지면 게임 시작 버튼 생성

7.3 게임 시작 버튼 누르면 순서를 정하는 미니게임 창 실행
7.4 카드를 누름과 동시에 서버에 순서를 알려준다.
7.5 순서를 정한 후 제시어가 뜨고 동시에 타이머(30초) 실행
7.6 유저의 캐릭터와 배경을 소유

8. Mini_Game
8.1 카드 선택 시 순서와 유저이름 서버로 송신
8.2 카드 선택 한 개만 가능

9. Game Timer
9.1 Game Timer(int num)으로 30초, 20초, 10초 설정

10. sound manager
10.1 창이 전환할 때마다 그 창에 따른 배경음을 재생
10.2 배경음을 조절
10.3 효과음 생성

11. 마우스 이미지 변환
11.1 컴포넌트에 따른 마우스 이미지 전환 구현
(펜, 붓, 지우개, 기본)

작업 진행 결과(개 인 별)

<최유빈> - 작업 진행 결과 (2306 Line)

1. GUI
   - Game
   - Store
   - Awards
2. Handler
   - GameHandler
   - StoreHandler
3. GameManager
   - 각 방마다 유저 4명이 채워지면 게임 시작 버튼 생성 구현
   - 게임 시작 버튼 누르고 미니게임
   - 현재 그리는 사람과 제시어를 저장, 게임 당 유저 순서 등 게임에 필요한 정보 저장
   - 카운트를 두어서 미니게임 카드 누를 시 마다 카운트 증가.
4. MiniGame
   - 카드를 한 유저 당 하나만 뽑을 수 있고, 순서와 유저를 서버에 보낸다.
6. Gaming
   - set_user(String[] user)로 유저를 저장
   - Stage()는 타이머 30초. Small_Stage_2()는 20초, Small_Stage_3()은 10초
   - GameStart()로 미니게임 실행 후 2초 뒤 미니 게임 창 비활성화
   - TimerStart(num) 타이머 클래스
   - 타이머가 끝났을 시 다음 소 스테이지로 넘어가는 프로토콜 생성
7. 게임 진행
   - 미니게임 실행
   - 뽑은 순서대로 유저 정렬
   - 순서에 따라 그림을 그리는 사람만 그림 그리기, 채팅 금지 등 설정
   - 타이머가 끝나면 두 번째 사람이 제시어를 보고 20초간 그림을 그린다. 첫 번째 사람은 그림을 그릴 수 없고, 채팅도 할 수 없다.
   - 정답을 맞추면 다음 단계로 넘어간다. 맞추지 못하면 세 번째 사람이 그림을 그린다.

<정가인> - 작업 진행 결과 (314 Line)

1. sound
   - enume을 이용하여 소리 파일 저장
   - 소리 파일 리소스를 Map을 이용하여 관리하는 FileManager
   - 창이 전환할 때마다 그 창에 맞는 배경음 실행 메소드 실행
   - 배경음 조절 창을 이용하여 그 볼륨 값에 맞는 배경음 조절
   - 컴포넌트에 따른 효과음 파일 재생

2. Cursor
    - 기본, 붓, 연필, 지우개의 cursor image 설정
    - 각 컴포넌트에 따른 cursor 변환

<박현서> - 작업 진행 결과 (5632 Line)

1. 통신
    1.1 ServerSocket을 열고 accept() Thread 생성
    1.2 Client가 접속요청을 하고 확인이 되면 Session Thread(Client와 소통하는)생성
    1.3 Input,output Stream 연결
    1.4 현재 String객체를 보내고 readUTF메소드로 받는다.
    1.5 대기실 창을 나가면 Socket Close
2. 파일 저장
    2.1 UserInfo와 UserData디렉토리를 만든다.
    2.2 회원가입 시 아이디와 비밀번호를 UserInfo파일에 저장(같은 아이디가 존재 할 경우 중복메세지)
    2.3 로그인 할 때 UserData와 Rank에 저장된 데이터를 받는다.
    2.4 게임이 끝난 후 보상을 받거나 상점에서 물건을 구매하면 Rank(화폐가 들어있는 파일)이 업데이트된다.
3. GUI <<Tool>>
    3.1 로그인 Scene
    3.2 대기실 Scene
    3.3 랭크 Frame
    3.4 방 생성 Frame
    3.5 회원가입 Frame
    3.6 게임 완료후 보상 Frame
    3.3 CustomClass
4. Handelr(Game제외)
    4.1 로그인 Handler
    4.2 대기실 Handler
    4.3 Resister
    4.4 Create
5. Canvas
    5.1 MouseListener -> MouseDragged -> 클릭한 좌표 newX,Y -> MouseReleased-> newX,Y = oldX,Y-> DrawLine(old~new)
    5.2 색깔, 두께 조절가능
    5.3 지우개, 전체지우개
    5.4 게임이 시작하고 나서 그림을 그리기 때문에 BufferedImage를 통해 Client마다 그림을 그리는 순간 저장한다.
    5.5 그래픽 보완을 위해 Antialiasing
    5.6 통신 안전을 위해 Nagle's Algorithm
6. FadeIn FadeOut
    6.1 Timer를 만들어 불투명도를 시간이 증가함에 따라 감소하게

만듦
6.2 화면이 넘어갈 때 FadeIn/Out이 되어 부드럽게 진행
6.3 게임이 시작한뒤 캐릭터 위에서 말풍선이 뜨게 함

7. 게임 진행
7.1 Timer를 각 조그만 스테이지에 따라 시간을 다르게 실행
7.2 제한 시간 안에 정답을 맞추거나 시간이 다 지날 때 점수분배
7.3 다음 라운드로 넘어가면 그림그리는 사람이 바뀌거나 도구의 사용 가능 유무, 정답이 보이는지의 유무, 캐릭터의 위치변경등의 셋팅
7.4 모든 라운드가 끝난 후 시상식과 금메달 은메달 동메달 분배
7.5 게임 재시 작을 위한 설정
8. 예외처리
8.1 로그인 중복처리
8.2 방 이름 중복처리
8.3 게임 진행 중에 나가는 사람 처리
8.4 연결 오류 처리
9. 배경음악 제작 <운영체제 문제로 사용불가>
9.1 GameScene 배경음악