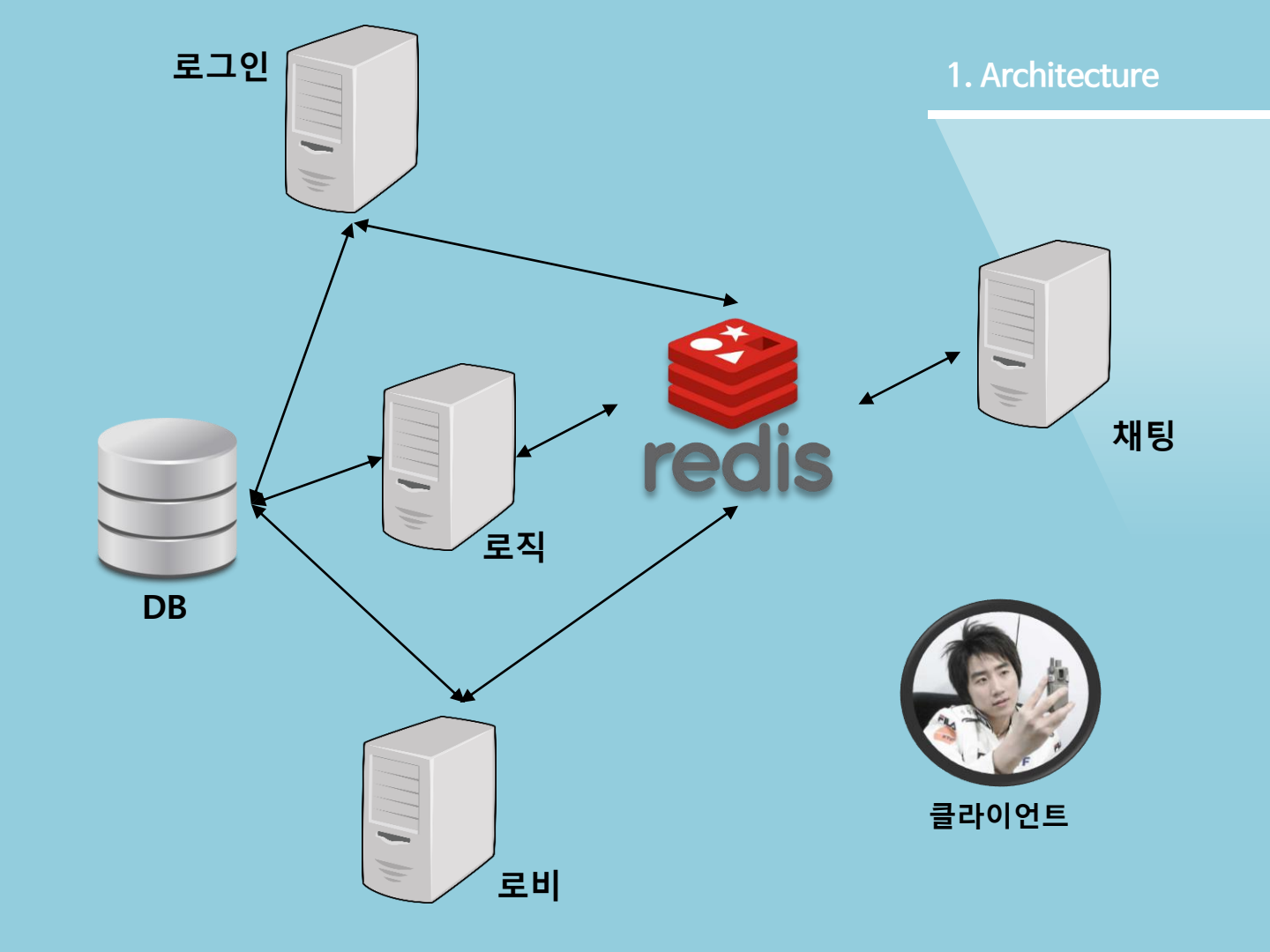
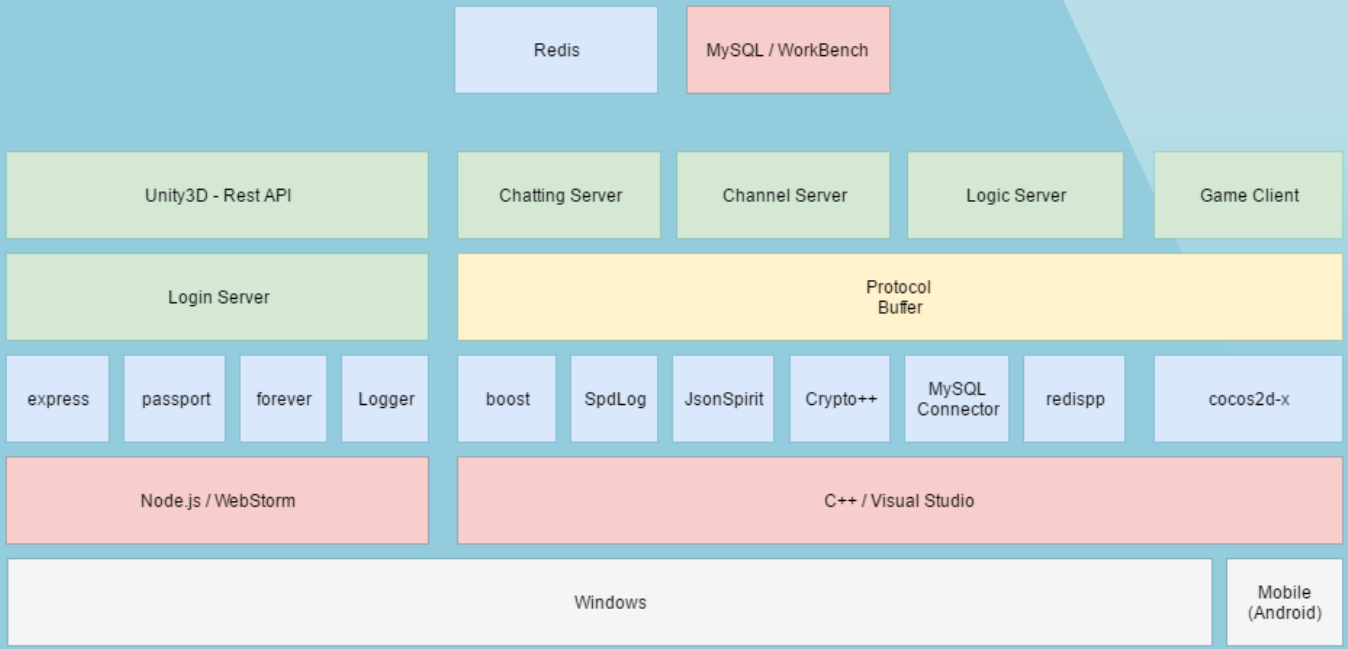
Game Server

Togater


로그인
redis
채팅
로직
DB
로비
클라이언트

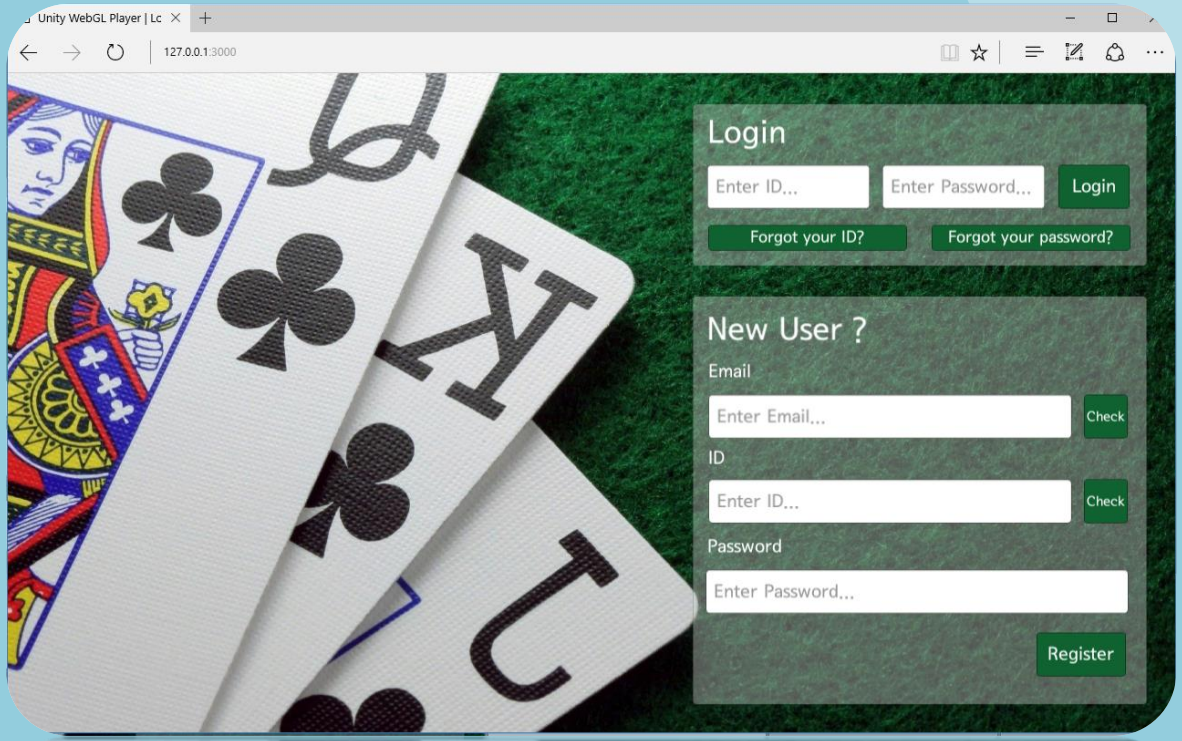
1. Architecture



2. Process

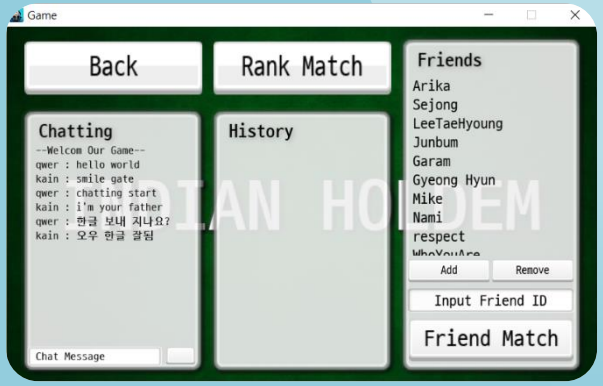

아키텍처
설계
개발 관련
스터디
담당 서버 핵심
기능 개발
클라이언트
핵심 기능 개발
로직 서버
연동
로그인 서버
연동
채팅 서버
연동
로비 서버
연동
핵심 기능
테스트
추가 기능 개발
성능 테스트
개발 완료

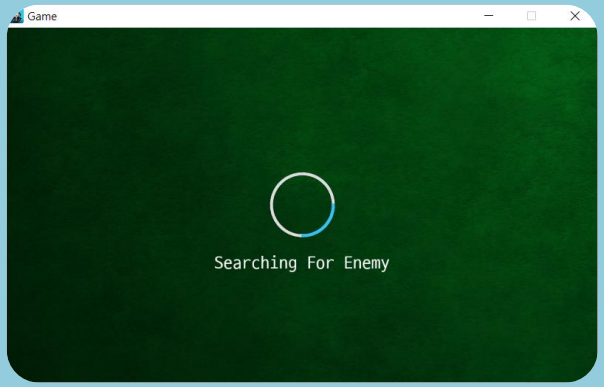
3. Layout


Unity WebGL Player | Lc
127.0.0.1:3000
Login
Enter ID...
Enter Password...
Login
Forgot your ID?
Forgot your password?
New User ?
Email
Enter Email...
Check
ID
Enter ID...
Check
Password
Enter Password...
Register

3. Layout


Game
Togater
INDIAN HOLDEM
ID
ID
PW
Password
LOGIN


Game
Back
Rank Match
Friends
Arika
Sejong
LeeTaeHyoung
Junbum
Garam
Gyeong Hyun
Mike
Nami
respect
Add
Remove
Input Friend ID
Friend Match
Chatting
--Welcom Our Game--
qwer : hello world
kain : smile gate
qwer : chatting start
kain : i'm your father
qwer : 한글 보내 지나요?
kain : 오우 한글 잘됨
Chat Message
History


Game
Searching For Enemy


Game
Chatting
5
Opponent Card
Public Card
1
Public Card
BET
Server Camp
Give Up
Input Chat Here


Done
Doing
To Do

Login Server

Done

기능	설명
로그인 웹 페이지	신속한 개발을 위해 Unity3D로 개발되었으며 웹에 배포하기 위해 WebGL 빌드
로그인 API	POST 데이터를 바탕으로 DB에 질의하여 로그인 인증, 엑세스 토큰을 생성하여 클라이언트로 전달, 레디스에 세션 정보 저장
아이디 찾기 API	DB에 질의하여 이메일을 기준으로 아이디 검색
비밀번호 찾기 API	DB에 질의하여 ID를 기준으로 비밀번호 검색
회원가입 API	DB에 질의하여 ID, 비밀번호, 이메일을 DB에 삽입

Done

기능	설명
회원가입 / 이메일 중복 체크 API	DB에 질의하여 이메일 중복 체크
회원가입 / ID 중복 체크 API	DB에 질의하여 ID 중복 체크
레디스 연결	Node.js 서버와 레디스가 통신하여 토큰, 세션 저장
MySQL 연결	Node.js 서버와 MySQL이 통신하여 각종 정보 질의

Doing

기능	설명
로비 웹 페이지	랭킹, 나의 전적, 친구 리스트, 유저 전적 찾기, 채팅 기능 등 페이지 디자인
랭킹	DB에 질의하여 게임의 승수를 기준으로 랭킹을 보여줌
나의 전적	DB에 질의하여 나의 전적을 보여줌
친구 리스트	DB에 질의하여 나의 친구를 리스트로 보여줌
유저 전적 찾기	DB에 질의하여 아이디를 기준으로 유저의 전적을 보여줌

To Do

기능	설명
OAuth 기능	페이스북과 같은 외부 프로바이더 인증
채팅 기능	Socket.IO 모듈을 통하여 웹 페이지에 접속한 유저끼리 채팅 기능 제공
회원탈퇴 기능	회원탈퇴를 원하는 사용자 탈퇴 제공
NGINX 프록시	보안, 정적 파일 제공을 위해 NGINX 웹 서버 프록시
Cluster 모듈	CPU 코어별로 클러스터 하기위해 Cluster 모듈 사용
HAProxy	로드밸런싱을 위한 HAProxy 연결

Chatting Server

Done

기능	설명
사용자 인증 확인	클라이언트가 보낸 쿠키와 아이디를 Redis에 있는 값과 일치하는지 확인한다. 일치하지 않는 경우에는 클라이언트와의 연결을 종료한다.
Redis	Redispp 라이브러리를 이용하여 채팅 서버에서 redis 서버로 접근한다.
프로토콜 버퍼	Protocol Buffer 라이브러리를 이용하여 직접 패킷을 정의하고 사용한다.
일반 채팅	사용자가 입력한 채팅을 다른 사용자에게 broadcasting 한다.

Doing

기능	설명
룸 채팅	게임이 매칭된 경우, 사용자를 로비 채팅에서 제외하며 게임 상대와 채팅을 할 수 있도록 한다.
예외 처리	현재 발생하는 모든 예외를 처리한다.

To Do

기능	설명
귓속말	특정 사용자에게만 채팅을 할 수 있도록 한다.
공지 메세지	관리자를 위해, 로비에 있는 사용자와 게임 중인 사용자 모두에게 채팅을 할 수 있도록 한다.
서버 효율 개선	현재 작성된 비효율적 코드를 성능 향상을 위해 개선한다.
로그 출력	spd_log를 사용하여 기간 및 채팅 그리고 매칭에 대한 개별 로그를 남기는 기능을 개발.

Lobby Server

Done

기능	설명
친구 찾기/삭제/추가	1. 클라이언트에서 추가,찾기,삭제하고자하는 대상의 id를 서버로 전송 2. 서버는 클라이언트의 DB에 id 저장,삭제, 읽기
랭크 게임	1. 랭크 점수로 랭크 구간을 나누어 각 구간마다 deque를 생성 2. 랭크 매치 요구가 들어올 때마다 점수에 적합한 deque에 삽입 3. Deque에 1명 이상의 유저가 있을 시 즉시 방을 개설하고 매칭 4. 개설된 Room 정보는 Redis에 저장 {Room번호:0}
친구와 게임하기	1. 클라이언트에서 게임하고자 하는 대상의 id를 서버로 전송 2. 서버는 대상의 id를 확인하고 로비에 접속되어 있는지 확인 후 대결 신청 메시지를 relay 신청에 대한 수락/ 거절 패킷을 수신 If(수락) 개설된 Room 정보는 Redis에 저장 {Room번호:0} Else 거절 메시지 relay
로그 아웃	1. 클라이언트는 로그아웃 통지 패킷을 서버로 전송 후 종료 2. 서버는 Redis에 저장된 클라이언트의 토큰 값을 삭제
로비 접속	1. 클라이언트는 로그인 서버에서 인증 받은 토큰 값을 로비 서버에 전송 2. Redis에서 토큰 값 유효성 체크 후 클라이언트에게 세션 할당

Doing

기능	설명
랭크 게임 매칭 개선	랭크 게임을 원하는 유저가 적고, 랭크점수에서 큰 편차가 있을 경우 매칭이 잡히지 않고 무한 대기하는 상태가 발생 따라서 일정 시간을 대기할 때 마다 하위 또는 상위 랭크 유저와 게임을 할 수 있도록 개선
비정상 종료에 대한 처리	로비 서버에서 관리하는 세션은 크게 4가지 상태로 관리된다. 로그인, 로그아웃, 매칭 신청, 매칭 완료 로그아웃을 제외한 상태에서 접속 종료 발생시 각 상태에 알맞은 처리가 필요

To Do

기능	설명
서버 통합 및 테스트	로그인 서버, 로비 서버, 채팅 서버, 게임 로직 서버 모두 연동하여 게임이 정상 동작을 하는지 확인
큐를 사용한 성능개선	DB 쿼리로 인한 병목현상이 있을 수 있으므로 DB쿼리가 필요한 기능들의 응답을 비동기로 처리하기 위해 큐를 사용
로그	로비 접속/해제, 게임 매칭, 친구 관리에 관한 로그를 남긴다. File I/O의 오버헤드를 고려하여 로그를 메모리에 저장하였다가 일정 개수가 넘어가면 Write 수행
DB 연동	로비에 접속하게 되면 접속한 유저의 전적, 친구리스트 정보를 가져오기 위해 DB서버에 접근 유저의 친구 추가,삭제,찾기 요청을 찾기하기 위해 DB 서버에 접근

Logic Server

Done

기능	설명
다중 접속	boost::asio를 이용하여 비동기 서버 개발을 함. 패킷 통신을 위해 protocol buffer를 사용하였음.
게임 로직 처리	게임의 방을 생성한 후, 유저와 통신을 하여 실제 게임 로직을 실시간으로 처리함.
환경 설정	Json_spirit을 이용하여 Json파일을 통해 서버 config을 관리함.
로그 처리	spd_log를 이용하여 system 및 matching log를 생성함.
인증 처리	Redis에 존재하는 유저&방 쿠키를 통해 접속에 대한 인증을 처리함.

Doing

기능	설명
버그 체크	서버 게임 로직이 원작 게임의 룰을 어긋나는 부분이 있는지 확인하는 중
코드 정리	코드 최적화를 위해 리팩토링을 진행 중

To Do

기능	설명
전적 갱신	게임 완료 시, 승&패, 레이팅, 게임 내역을 DB에 갱신함.
비정상 종료 체크	비정상 종료 시, 유저&방 인증 쿠키 제거와 DB에 접근하여 전적을 수정
성능 테스트	다량의 인원이 원활한 게임을 진행하는 환경을 구축함.
서비스 프로그램 작성	윈도우 서비스 형태로 서버가 동작하도록 수정함.

Client

Done

기능	설명
Login Scene	로그인 관련 기능을 수행함. 로그인 서버와 HTTP 통신을 통해 인증 절차를 거침.
Lobby Scene	채팅 서버와 인증 절차를 거침. (로비 서버는 추가 예정). 로비 채팅을 수행하고 테스트 매칭을 수행 중.
Loading Scene	게임 상대방 매칭을 시도함. 현재는 테스트 매칭으로 개발되어있음.
Game Scene	로직 서버와 인증 절차를 거침. 로직 서버와 TCP 통신을 통해 게임을 진행함.

Doing

기능	설명
Logic Scene GUI 정리	새로운 GUI를 추가하는 중. (상대방 카드 정보, 현재 배팅 정보, 채팅 버튼 등)
코드 개선	현재 빠른 프로토타입을 위한 코드를 조금 더 가독성 있고 성능상 최적화된 형태로 수정중.

To Do

기능	설명
로비 서버 연동	로비와 연동하여 표현할 기능들을 개발 합니다.
친구 추가 GUI	시간이 된다면 친구 관련 Scene을 추가로 개발합니다.
안드로이드 Build	안드로이드 플랫폼 App으로 빌드합니다.

서버	2월 1주차			
날짜	31일	1일	2일	3일
로그인	- 전적 웹 페이지 완성	- 랭킹 - 전적 확인 - 친구 리스트		- MySQL DB 구조 완성
채팅	- 룸 채팅 기능 추가		- 귓속말, 공지 기능 추가	
로비	- 랭크 게임 매칭 개선		- 비정상 종료에 대한 처리	
로직	- 게임 로직 관련 버그 수정		- 게임 종료 시, 인증 쿠키 제거 기능 추가	

서버	2월 2주차				
날짜	6일	7일	8일	9일	10일
로그인	- 회원 탈퇴 기능 추가	- 웹 채팅 기능 추가		- Cluster를 통해 멀티코어 프로그래밍 시도 - NGINX 프록시 -> 보안 및 정적 파일 속도 향상	
채팅	- 로그 기능 추가	- 효율 개선		- 예외 처리 추가	
로비	- DB 연동	- 로그 기능 추가	- 큐를 이용한 성능 개선		- 예외 처리 개선
로직	- 비정상 종료 체크 - 전적 갱신		- 서비스 프로그램 작성	- 코드 리팩토링	

서버	2월 3주차				
날짜	13일	14일	15일	16일	17일
로그인	- HAProxy를 통해 로드 밸런싱		- 회원 서비스 종합 점검		
채팅	- 전체 서버 Failover 기능 개발 / 연구 / 세미나 준비 / 각 보기				
로비	- 서버 통합 및 테스트		- 로드 밸런싱 관련 연구 / 세미나 준비 / 개발		
로직	- 클라이언트 최종 수정 (더 이상 UI, 기능 변경 없음)		- 암호화 관련 기능 개발 / 연구 / 세미나 준비		

시연

TOGATER

감 사 합 니 다

TOGATER