프로젝트 이름

팀원 : 1901314 황용주, 1701091 김희수

개발 환경

운영체제: Windows10, Mac Os

언어: 파이썬 3.7.0, Js

라이브러리: channels, redis

(웹서버가 있다면) 웹서버: React Web Server, Django WebServber, redis server

(데이터베이스가 있다면) 데이터베이스: SQLite3

Django FrameWork 사용

프로젝트 개요 (기획서 참조해서 최초 목표 기준으로 설명)

이 프로젝트는 루미큐브 게임입니다.

루미큐브는 혼자서 할 수 없는 놀이인데 코로나로 인해서 만나지 못하는 점을 고려하여 네트워크 상으로 진행하며 어떨가 싶어 기획하게 됨

웹소켓을 활용하여 유저간에 통신을 활용하여 데이터를 주고 받도록 설계

기본 유저데이터나 방 데이터 는 DB에 저장하여 유저가 참여할 위치를 배정해줌

게임 진행시 숫자를 체크하고 규칙에 맞도록 진행 하도록 함

유저간에 턴 게임을 활용하여 자신의 턴이 아니면 이용할수 없음

프로젝트 구현도(최초 기획 대비 구현 한 것, 못한 것 정리해서 적기)

방생성 -> 자동생성 및 참여로 변경

게임 인원은 4명이 최대로 되도록 구현

호스트 클라이언트 구분 완료

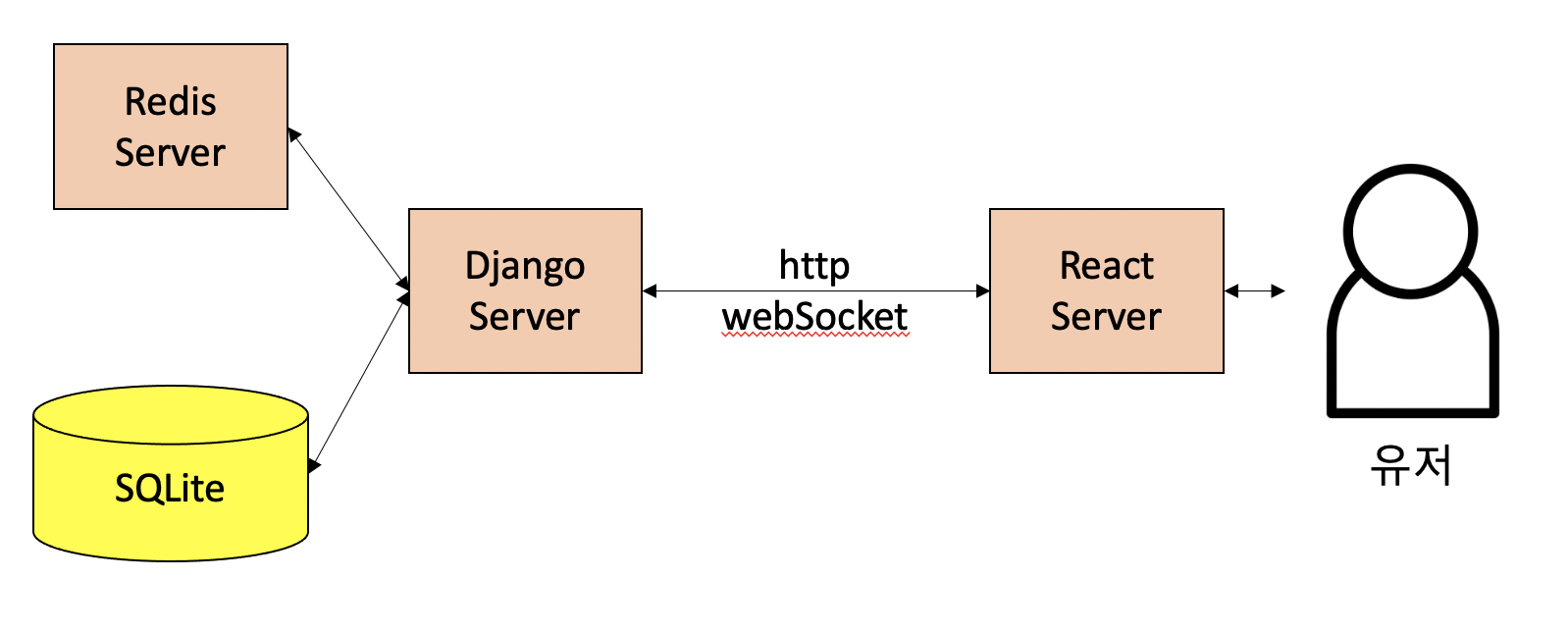
호스트 가 게임시작 시 전체 게임 시작 구현 완료

랜덤으로 카드 뿌리기 완료

Django로 channels 라이브러리 활용 webSokcet 연결 완료

유저 데이터, 방 정보 DB에 저장

시스템 구성 (서버나 DB, 파일 등을 사용하지 않는 프로젝트는 이 부분 생략)



유저가 React Server로 페이지 요청

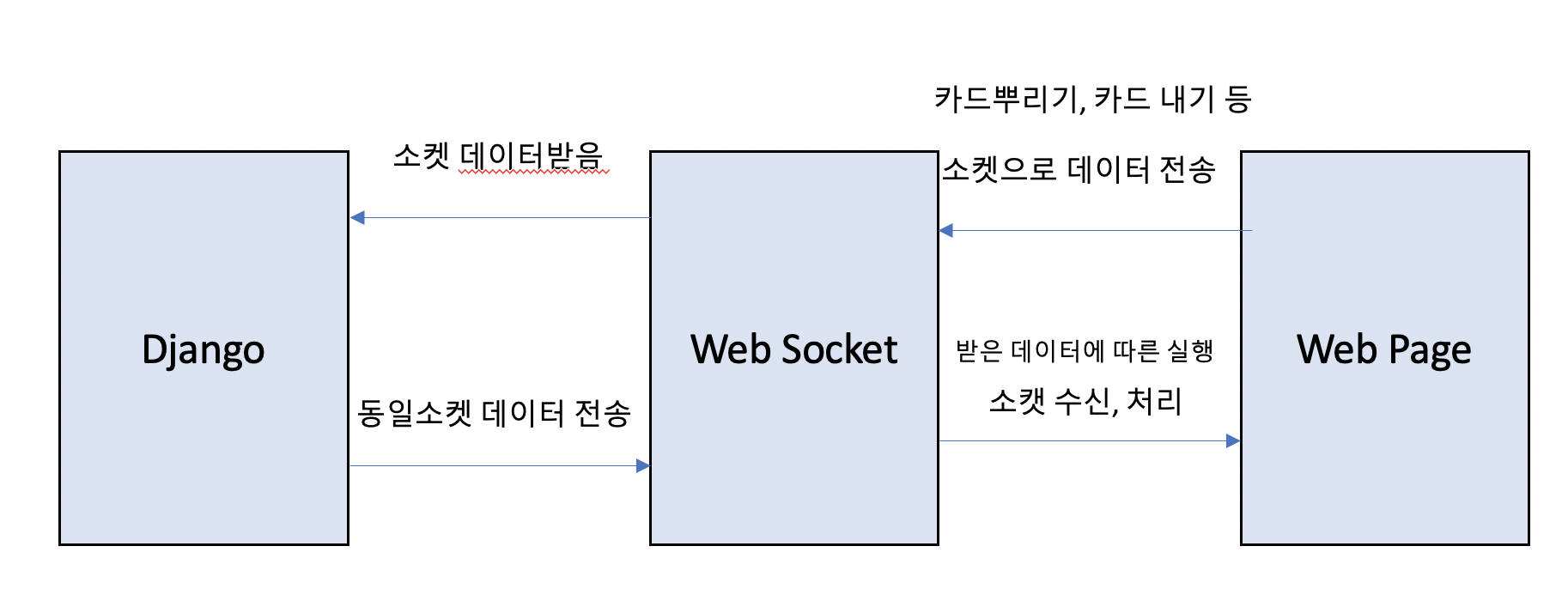
접속이 Django Server를 활용하여 데이터를 주고받음

Redis Server를 통하여 유저간 webSocket 방 관리

SQLite 에 유저 방, 유저정보 등을 저장

Django에서 webSocket으로 받은 message를 동일한 webSocket 으로 전송

세부 모듈 (Application 내의 모듈 구조를 설명)



Web Page -> 사용자 View 밑 동작

Web Socket 사용자 동작에 대한 값을 해당 방의 유저에게 전송하기 위함

Django 동일한 Link를 가진 WebSocket으로 데이터를 전송해줌

참고 코드 (구현에 참고한 코드가 있다면 다 적기)

<https://docs.djangoproject.com/ko/3.1/intro/> -> 장고 관련

<https://velog.io/@matisse/Django-advanced-channel-socket> -> 장고활용 channel , websocket 통신

대표 화면

텍스트이(가) 표시된 사진

자동 생성된 설명