



Read.md



윷장
윷짝놀이



✨ 프로젝트 진행 기간

2023.04.10 (월) ~ 2023.05.19 (금) - (6주간 진행)

SSAFY 8기 2학기 자율 프로젝트 - 윷짝놀이

✨ 팀원 소개



김소희 - FrontEnd

- 웹 소켓 구조 설계 및 구현
- 대기방 및 준비 구현
- UI/UX 설계



김정은 - BackEnd

- Kafka 환경 구축
- 웃 던지기 API 구현
- 실시간 채팅 API 구현



박소연 - BackEnd

- 서버 및 CI/CD 구축
- 게임 로직 및 설계 구현
- 이벤트 API 설계, 구현



박재희 - FrontEnd

- 스토리북 설정
- 게임 로직 설계 및 구현
- 게임 화면, 게임 설정 화면 구현



이준 - BackEnd

- 웹 소켓 통신 구축
- DB 설계
- 웃놀이 기능 구현



이진우 - FrontEnd

- 버튼 공통 컴포넌트 제작
- 웃 던지기 기능 구현
- UI/UX 설계

✨ 기획 배경

대한민국 문화에 대한 소비가 증가하는 추세입니다. K-Bueaty, K-Drama, K-Pop에 대한 관심이 해외에서 열기가 식을 줄 모르고 활활 타오르고 있으며 저희 또한 문화 소비가 활발한 추세입니다. 또한, 해외에는 이토록 자국 문화가 지켜지고 있는 것이 많지는 않은 편이라고 합니다. 하지만, 대한민국의 전통에 대해서는 잊혀져 가는 분위기입니다. 저희는 대한민국에 문화에 대한 관심이 높은 이 시점이 한국의 전통놀이를 알리기 적합하다고 생각하였습니다.

그러던 중, 2022.11.11을 기준으로 웃놀이가 무형 문화재에 등록되었다는 것을 알게 되었고 전통 놀이의 대표 중 하나인 웃놀이를 선택하여 다시금 주목 받고 알리고 싶었습니다.

✨ 서비스 개요

웃짝놀이는 명절에 다 같이 모였을 때만 즐기는 웃놀이가 아닌 평소에도 접근하기 쉽게 웹 브라우저 환경에서 제공하는 온라인 웃놀이 게임입니다.

- 웃놀이
 - 웃 던지기, 말 이동, 말 잡기 말 동나기 등의 기본 웃놀이 게임 를 차용
- 개인전
 - 1 대 1 플레이 서비스를 제공하며 최대 4인까지 가능합니다.
- 랜덤 이벤트
 - 웃놀이 판에서 이벤트 칸을 제공하며 게임 시작 시 매번 이벤트 칸이 변경 되며, 해당 칸에 도착하면 랜덤으로 이벤트가 발생하여 게임의 재미 요소를 더하였습니다.

- 채팅 기능

게임을 하면서 즐거움과 훈수를 위한 게임 참가자들의 메세지를 실시간으로 통신하는 채팅 기능을 제공합니다.

✓ 기술 스택

▼ BackEnd

- Java 11
- SpringBoot 2.7.10
- Websocket
 - Stomp 2.3.4
 - SockJS 1.5.1
- Kafka
- Redis
- test

▼ FronEnd

- Node.js 18.15.0
- React 18.2.0
- Next.js 13.3.0
- Recoil 0.7.7
- styled-components 5.3.9
- sockjs-client 1.6.1
- stompjs 2.3.3
- storybook 7.0.3

▼ Server

- EC2
- Docker, Docker-Compose
- Jenkins
- Nginx
- 가비아 도메인

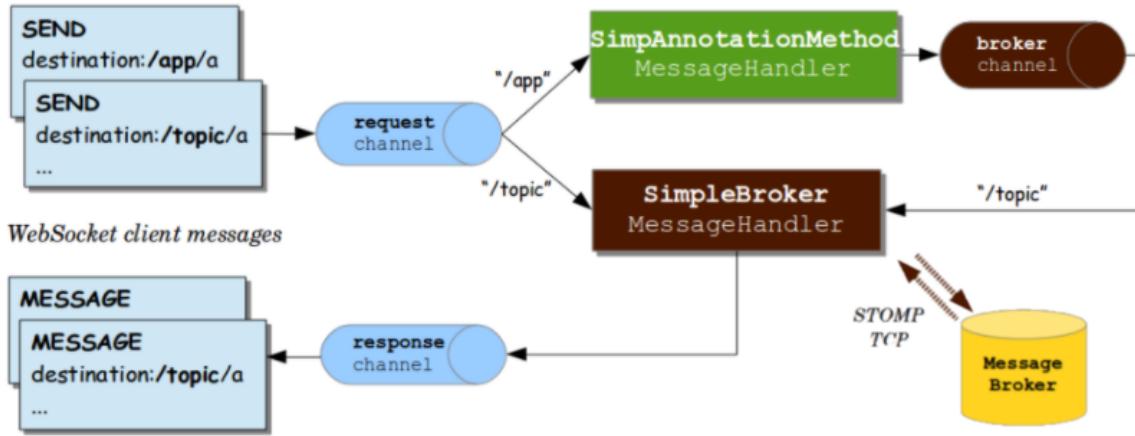
✓ 주요 기능

- NextJS 프레임 워크를 사용한 Front 구현

미리 렌더링된 페이지를 서버를 통해 가져와 사용자에게 빠른 결과 전달을 용이하게 해주는 프레임워크인 NextJS를 사용하여 성능과 개발 생산성을 향상

- 실시간 통신을 위한 Stomp 사용

WebSocket 위에서 동작하는 STOMP를 사용하여 메세지 형식을 정의하지 않고 발행(pub)과 구독(sub)을 통해 메세지 송수신이 가능하기 때문에 편리하고 정확한 통신 구축이 가능하다.



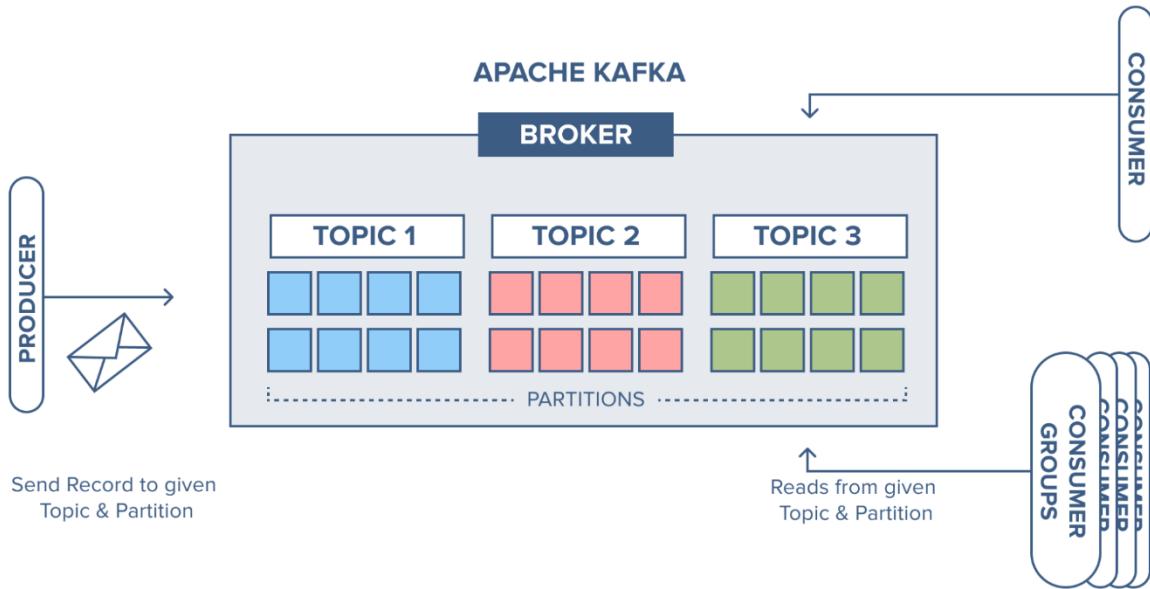
실시간으로 정보를 주고 받는 게임은 정확하게 정보를 특정한 사용자들에게 전달하고 받는 것이 중요하기 때문에 STOMP를 사용합니다.

- WebSocket 메세지 큐를 위한 Kafka 사용

Kafka는 메시지 큐 방식 기반으로 분산 메시징 시스템이다.

많은 양의 데이터를 partition하는 것에 초점을 맞춘 시스템이라 실시간으로 들어오는데 이터를 빠르게 처리할 수 있다.

하나의 메시지는 "topic"으로 분류 -> 하나의 토픽은 다수 개의 "partition"으로 분류된다.



- 게임의 실시간의 접근 시간 단축을 위한 Redis 사용

WebSocket을 통해 실시간으로 게임이 이루어져야 하는 만큼 인 메모리 캐시 데이터베이스인 Redis 사용합니다.

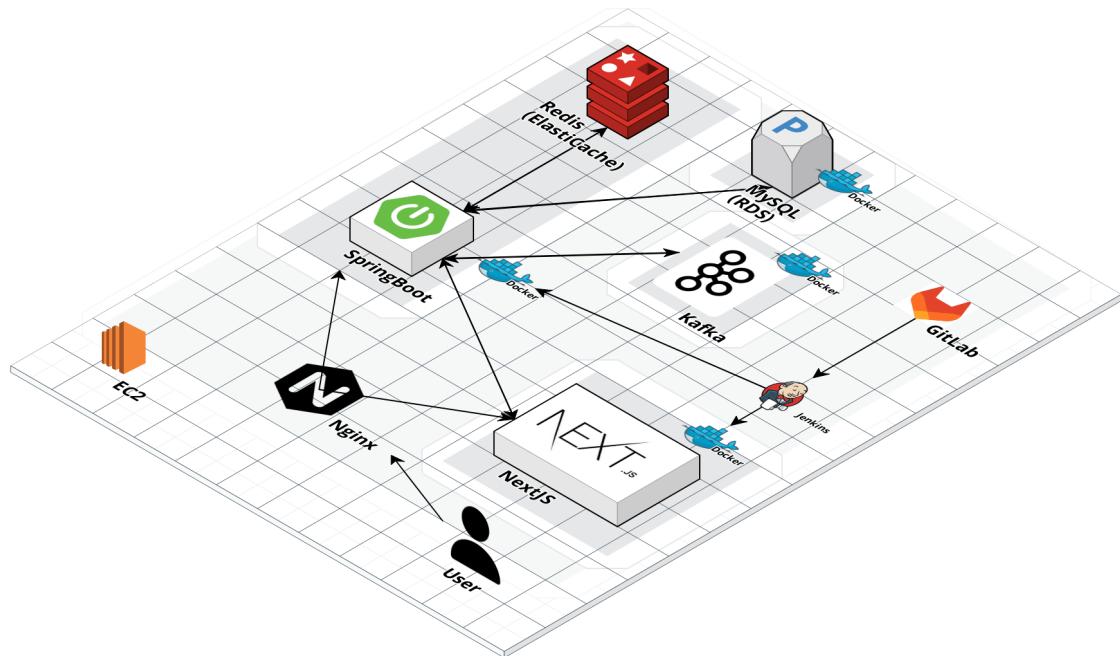
Redis는 성능이 뛰어난 인 메모리 캐시를 생성하여 액세스 지연 시간을 줄이고, 처리량을 늘리는 장점을 지니고 있습니다. 또한, Redis는 세션 관리 작업에 매우 적합합니다. Redis를 세션 키에 대한 적절한 TTL과 함께 빠른 키 값 스토어로 사용하면 간단하게 세션 정보를 관리할 수 있으며 이는 게임에 필요합니다. 이러한 Redis의 장점을 채택하여 WebSocket 통신에서 게임 정보를 저장하고 관리하는데 사용합니다.

✓ 프로젝트 파일 구조

✓ 프로젝트 산출물 및 결과물

- 산출물 exec 폴더 내 포팅 메뉴얼 및 서비스 시나리오 파일 참조

1 아키텍처 구조

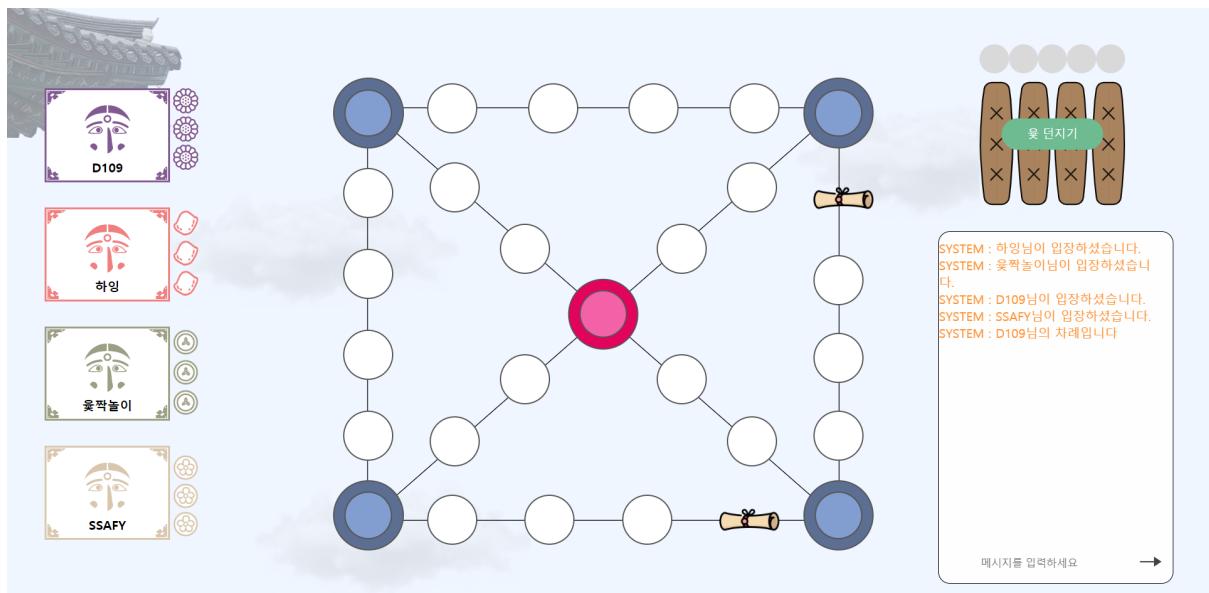


2 서비스 화면

대기 화면 및 준비 하기



게임 시작



웃 던지기 및 말 이동

