

2022-2학기 세종창의학기제 최종결과보고서

이름	홍성철	학과(전공)	컴퓨터공학과		
학번	18011489	학년	3		
연락처	010-6263-7649	e-Mail	tjd000222@naver.com		
과목명	자기주도창의전공I	분반	004	신청학점	6
	자기주도창의전공II		004		
학습기간	22.09.01 ~ 22.12.07	총학습주차	14	총학습시간	280H
창의 과제	국문	AI 기반 리그오브레전드 실시간 승률 예측 시스템			
	영문	AI based real-time League of Legends win rate prediction system			

1. 학습 목표

- GRU 모델을 통한 League of Legends 게임 경기 내 실시간 승률 분석 수치화된 실시간 승률 도출을 통해 게임 중계 시 새로운 Insight 제공, 스포츠 분야 중계 시 AI 승률 예측 활용가치 상승
- 프로 팀의 리플레이 피드백 상황에서 경기의 주요 승률 변동 원인 분석
- 최상위권 경기 데이터 기반 AI 모델을 일반 유저의 경기에 적용하여 실력 향상 도모

2. 학습 내용

- React를 이용하여 웹앱 및 데스크톱 클라이언트 제작.
- Docker를 활용하여 프로젝트 개발 환경 관리.
- Docker Compose를 활용하여 AWS EC2 인스턴스에 프로젝트 배포.
- Jenkins를 활용하여 프로젝트 CI/CD 관리.

3. 주차별 학습보고

주차	학습 목표 및 활동	학습방법	학습시간
1	- TypeScript props type 지정 형식 및 props전달 방식 파악. - 실시간 분석 페이지 및 분석완료 페이지에 대한 React 반응형 웹페이지 UI 제작.	공식문서 및 웹사이트 활용	20
2	- 컴포넌트별 css 파일 세분화. - chart-js 라이브러리를 활용하여 실시간 분석 및 분석완료 페이지에 사용되는 그래프 구현.	공식문서 및 웹사이트 활용	20
3	- API 로 받아오는 데이터와 같은 형식을 가지는 더미데이터를 chart-js에 적용하여 원하는 범위 내의 가로막대 차트 구현. - chart-js 세부 css 적용. - Riot 제공 API를 활용하여 실시간 분석 및 분석완료 페이지에 사용되는 이미지 구현.	공식문서 및 웹사이트 활용	20
4	- 데스크톱 클라이언트 앱 UI 구현. - 검증 코드 입력 및 분석에 대한 기능 및 애니메이션 구현. - 웹 페이지 더미데이터 적용.	공식문서 및 웹사이트 활용	20
5	- 검증 코드 자동 대문자 변환 구현.	공식문서 및	



	<ul style="list-style-type: none"> - 드래그 앤 드롭 파일 업로드 구현 - 코드가 존재하지 않는 경우 출력되는 화면 구현 - API를 적용하여 플레이어 10명에 대한 정보 받아오기 구현. 	웹사이트 활용	20
6	<ul style="list-style-type: none"> - 기존 파일업로드 라이브러리 지원한계에 따른 라이브러리 교체 - API 데이터 로딩에 걸리는 스켈레톤 애니메이션 페이지 구현 - 분석 완료시에 출력되는 페이지 구현. 	공식문서 및 웹사이트 활용	20
7	<ul style="list-style-type: none"> - 본인이 작성한 frontend 코드에 대한 1차 세부 리팩토링 진행 	공식문서 및 웹사이트 활용	20
8	<ul style="list-style-type: none"> - frontend 전체 코드에 대한 2차(최종) 세부 리팩토링 진행 	공식문서 및 웹사이트 활용	20
9	<ul style="list-style-type: none"> - 리팩토링 코드에 대한 팀원들 리뷰 반영. - 실시간 통신을 위한 socket 연결과 외부 API 적용하여 최종 기능 구현. 	공식문서 및 웹사이트 활용	20
10	<ul style="list-style-type: none"> - 수정된 API 적용 (데이터 예외처리 관련 수정). - OS에 따라 다르게 렌더되는 문제 해결. - frontend 세부사항 적용. 	공식문서 및 웹사이트 활용	20
11	<ul style="list-style-type: none"> - 프로젝트 개발환경 구축에 필요한 Docker 학습. 	공식문서 및 웹사이트 활용	20
12	<ul style="list-style-type: none"> - 학습한 내용을 기반으로 프로젝트에 Docker 적용. 	공식문서 및 웹사이트 활용	20
13	<ul style="list-style-type: none"> - AWS EC2에 Docker Compose 적용하여 프로젝트 배포. - Jenkins를 통한 CI/CD 구현. 	공식문서 및 웹사이트 활용	20
14	<ul style="list-style-type: none"> - 프로젝트 최종 flow 에 위반사항이 없도록 frontend 코드 검토 - AWS EC2 인스턴스가 다운되는 문제 해결 - CORS 보안 에러 해결. 	공식문서 및 웹사이트 활용	20

4. 최종결과물 및 성과

- League of Legends 실시간 게임플레이 시계열 데이터셋 수집

- 데이터 수집 스크립트 오픈소스 패키지 PyPI에 개발 및 배포 (DataNashor 0.4.5)
- Kaggle 플랫폼에 수집한 데이터셋 오픈소스 배포(12.21패치, 29,517경기, JSON 파일 19.49GB 분량)

- GRU 기반 AI 모델 개발

- 조기 승률 예측이 아닌 실시간 승률 평가 목적에 맞는 Scoring Metric 적용

- 15.gg 반응형 웹 서비스 개발 (웹, 모바일)

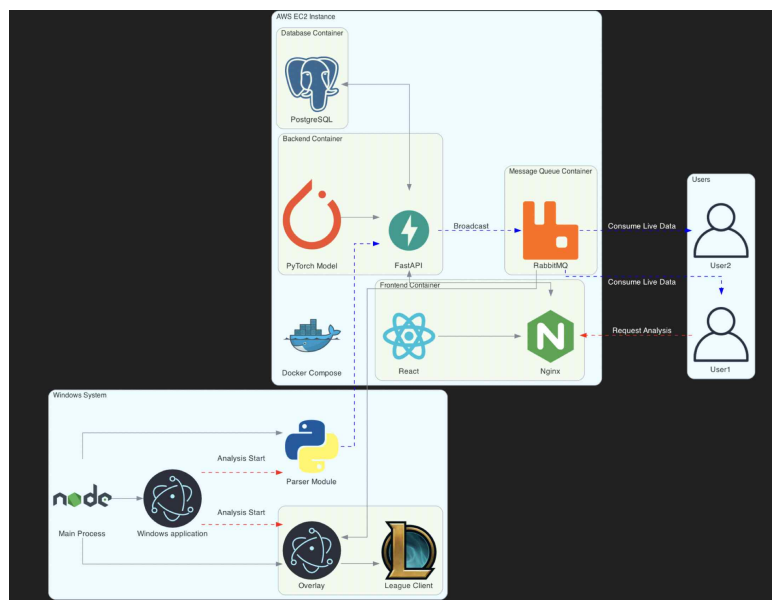
- Windows App과 LoL Game Overlay를 통한 실시간 연동 AI승률 및 게임 내 킬, 데스, 아이템 구매 등 이벤트 동기화해 제공
- 유저(소환사)별 스탯 및 승률 데이터 제공, 서비스를 통해 기 분석된 경기에 대한 데이터 제공
- OTP 생성으로 Windows App과 데이터 연동해 자동으로 분석 진행되는 프로세스 개발

- 15.gg Windows 클라이언트 개발

- Web 서비스에서 생성한 OTP, 리플레이 파일(.rofl)업로드를 통해 LoL 클라이언트 자동 시작 및 분석
- LoL 게임 실행시 자동으로 실시간 승률 정보 오버레이 형태로 실행

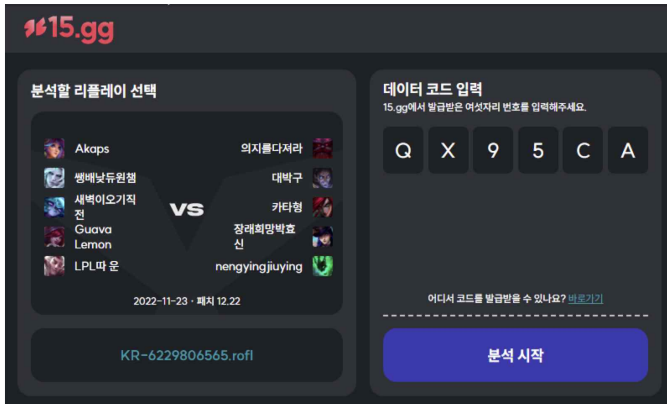
- 프로젝트 구조

- Windows App: Electron.js, TypeScript, Node.js, Python3
- Web Frontend: React.js, TypeScript, NGINX
- Web Backend: FastAPI, Uvicorn
- 데이터셋 파싱 모듈: Python3
- AI 모델: PyTorch
- 데이터 전처리: NumPy, Pandas
- DBMS: PostgreSQL
- Message Queue: RabbitMQ
- Infra: Docker Compose, AWS EC2

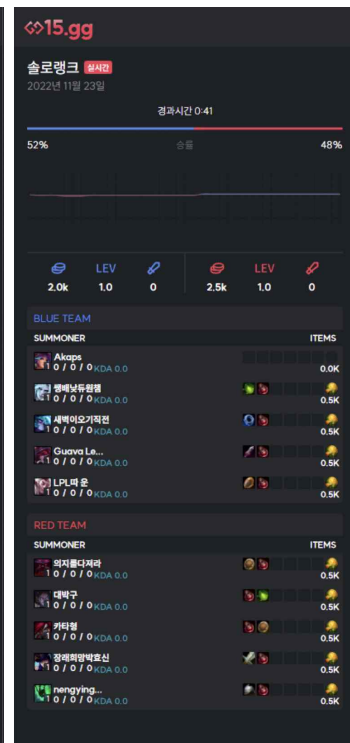
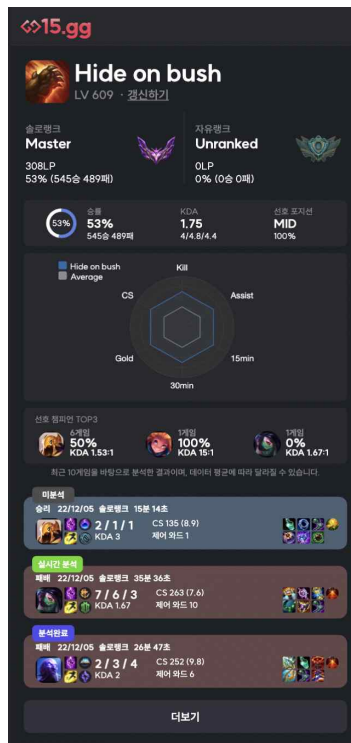


프로젝트 구조도

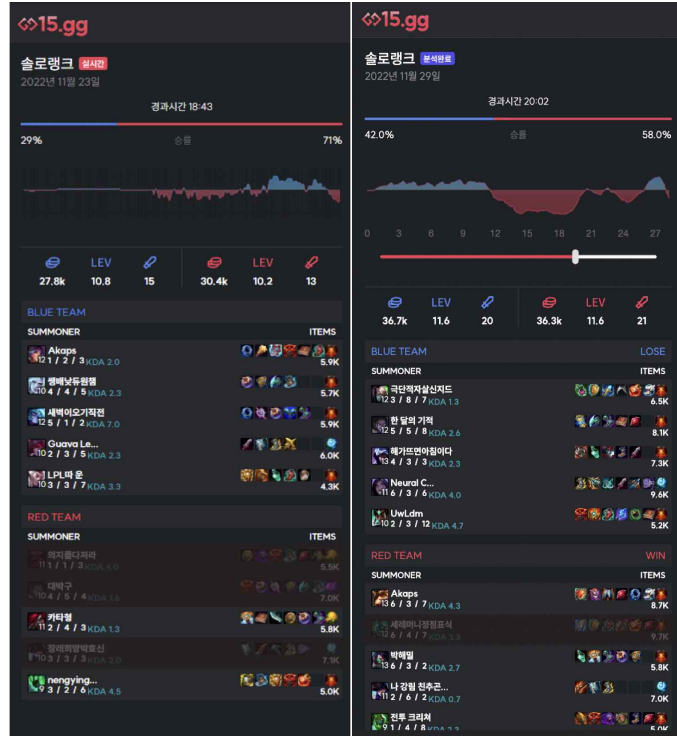
- 프로젝트 사진 (배포 후 서비스 사진)



15GG 데스크톱 클라이언트



15GG 웹 실시간 분석 페이지



15GG 웹 실시간 분석 페이지

15GG 웹 분석 결과 확인 페이지

5. 기대효과

- 추후 LCK(League of Legends Champions Korea) 중계 시 15GG AI 모델 및 서비스 활용 추진
- TEA (T1 Esports Academy), GGA (GenG Global Academy) 등 리그오브레전드 전문 프로게이머 육성기관 교육 및 피드백 자료로 활용 추진
- 일반 사용자 대상 Web, Windows 서비스 활용 방안 다각화

지도교수 : 최 유 경 (인)