

LML

LOL-MODEL

“ 머신 러닝을 활용한 리그 오브 레전드
전적 검색 및 승부 예측 애플리케이션 ”

201611088 윤주석
201611065 김준영
201710473 이해진
201810928 김현민
201810988 임혁



#1 개발 동기

1. 주제 선정 배경
2. 필요성 및 기대효과



#2 프로젝트 소개

1. 팀원역할 소개
2. 개발 일정
3. 개발 환경



#3 애플리케이션 개요

1. 아키텍처
2. 화면흐름



#4 상세 기술

1. 모델설명
2. 머신러닝
3. 모듈 간 통신
4. 웹 서버



#5 시연 영상



#6 마무리

1. 느낀점
2. 출처

1. 개발 동기 - 주제 선정 배경

뉴스 > 이슈

세계 e스포츠 시장 규모 1조 육박...한국도 1,000억 규모로 성장

2019-01-16 09:56:19



HOME > 스포츠 > 일반/e스포츠

"e스포츠 시장규모, 2022년 30억달러 육박 예상"

서지영 / 기사승인 : 2020-07-24 10:53:24



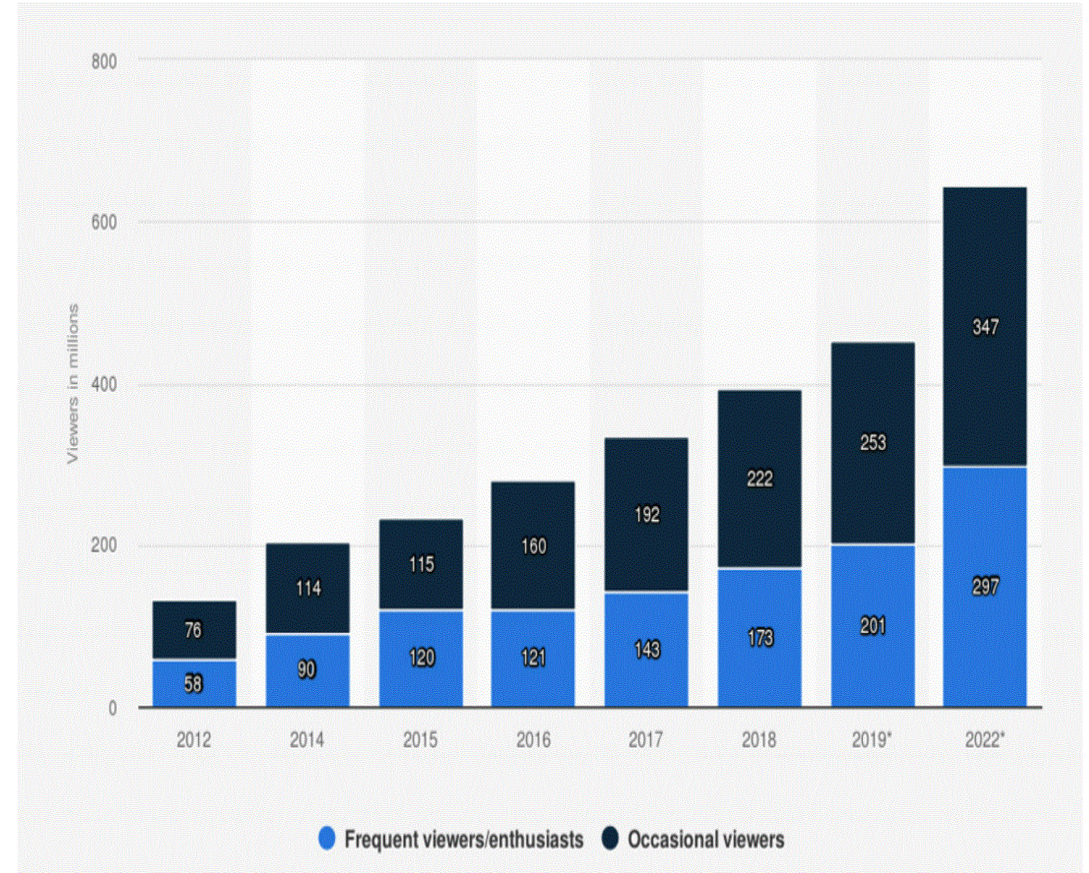
주요 e스포츠 온라인 중계 확대...시청자 수도 증가 전망

국내 e스포츠 산업 규모 1,398억 원... 전년 대비 22.8% 증가

게임뉴스 | 이두현 기자 | 댓글: 22개 | 11-23

1. 발전하는 E스포츠 시장

이스포츠 관객 규모 (2012~2022년*)



1. 개발 동기 - 주제 선정 배경

2. 룰의 점유율

디지털타임스

'리그 오브 레전드' 글로벌 흥행 동시접속자 최고 750만명 기록

12면 1단 | 기사일력 2014.04.17. 오후 9:15 | 기사원문 | 스크랩 | 본문듣기 · 설정

동시접속자 750만명 기록

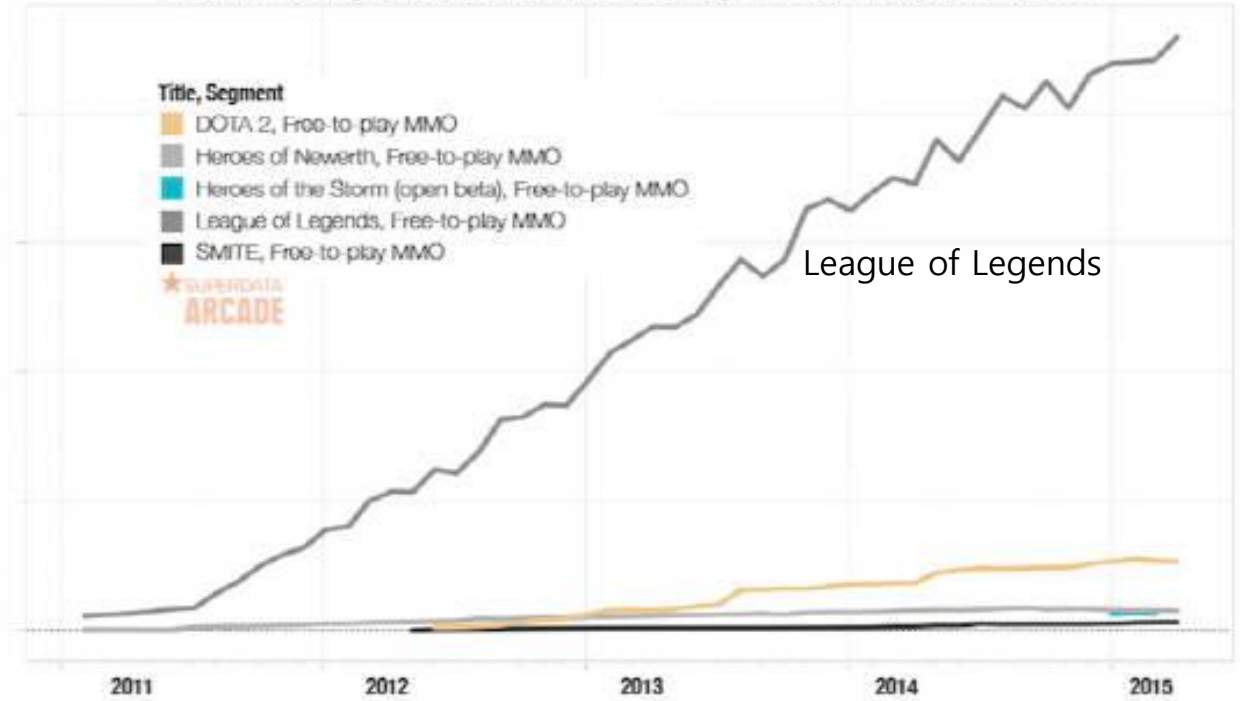
게임순위 2021년 9월 24일

순위	게임정보	점유율
1	 164주 1위 라이엇 게임즈 라이엇 게임즈 리그 오브 레전드	47.57%
2	서든어택	7.9%
3	배틀그라운드	6.13%
4	피파온라인4	5.58%
5	로스트아크	5.12%

(2021년 9월 기준)

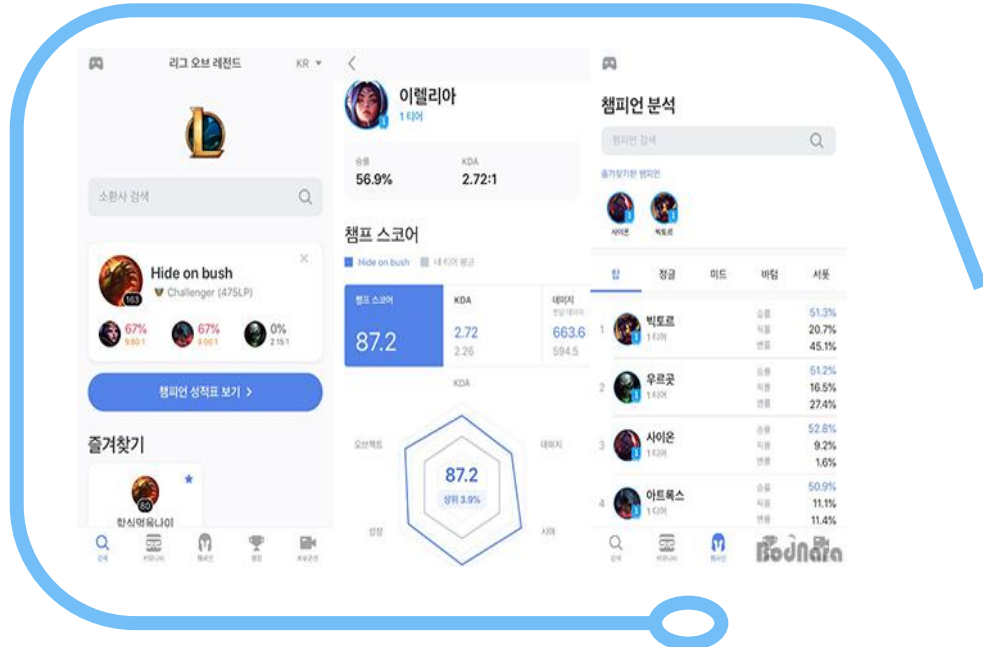
PC방 점유율 47.57%

Total monthly active user base for top MOBA titles, worldwide



최근 5년간 게임 시장 점유율 변화

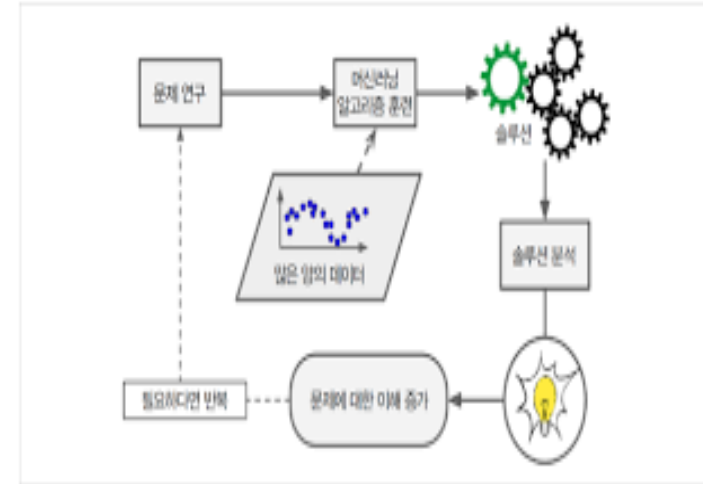
1. 개발 동기 - 필요성 및 기대효과



전적 검색 사이트

게임시장이 커지며 관련 편의 서비스들의 수요가 증가하고 있다.

그림 1-4 머신러닝을 통해 배웁니다.



머신러닝

Xgboost를 통해 산출된 자신 게임 승률을 확인함으로써 궁금증을 해결하고 그에 따른 대응적 플레이가 가능하다.

2. 프로젝트 소개 - 팀원 역할 소개



윤주석

모델 학습 및 테스트
전적 검색 기능개발



김준영

데이터 수집 및 전처리
웹 서버 개발



김현민

인게임 전적 기능 개발
전적 검색 기능개발
UI 개발



이혜진

문서 작성 및 관리
UI 개발
로고 디자인

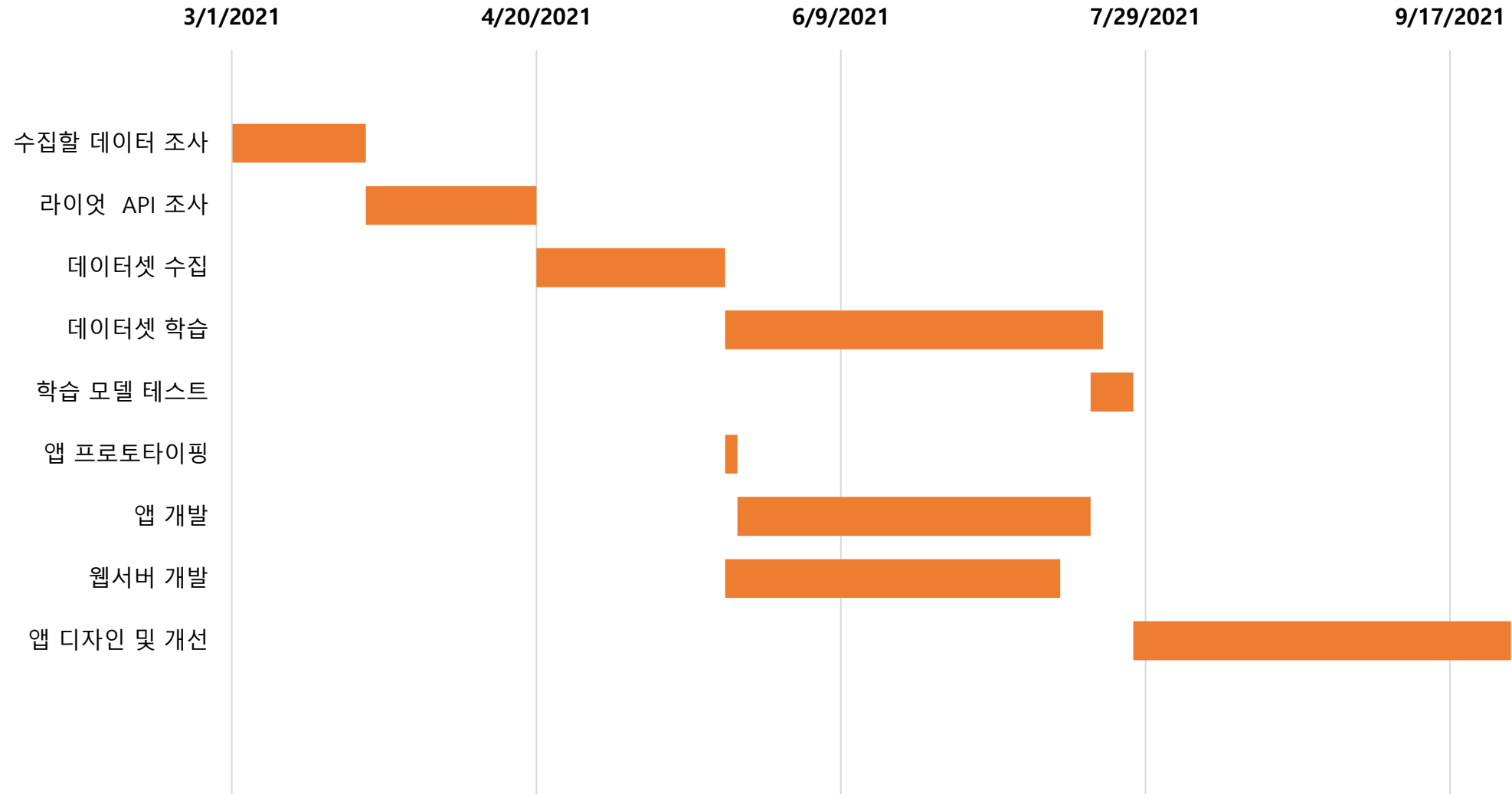


임혁

UI 개발
게시판 기능 개발
전적 검색 기능개발

2. 프로젝트 소개 - 개발 일정

일정 계획



2. 프로젝트 소개 - 개발 환경

- 애플리케이션

자바, 안드로이드 스튜디오



- 웹서버

파이썬, 플라스크, Amazon-ec2



Amazon EC2

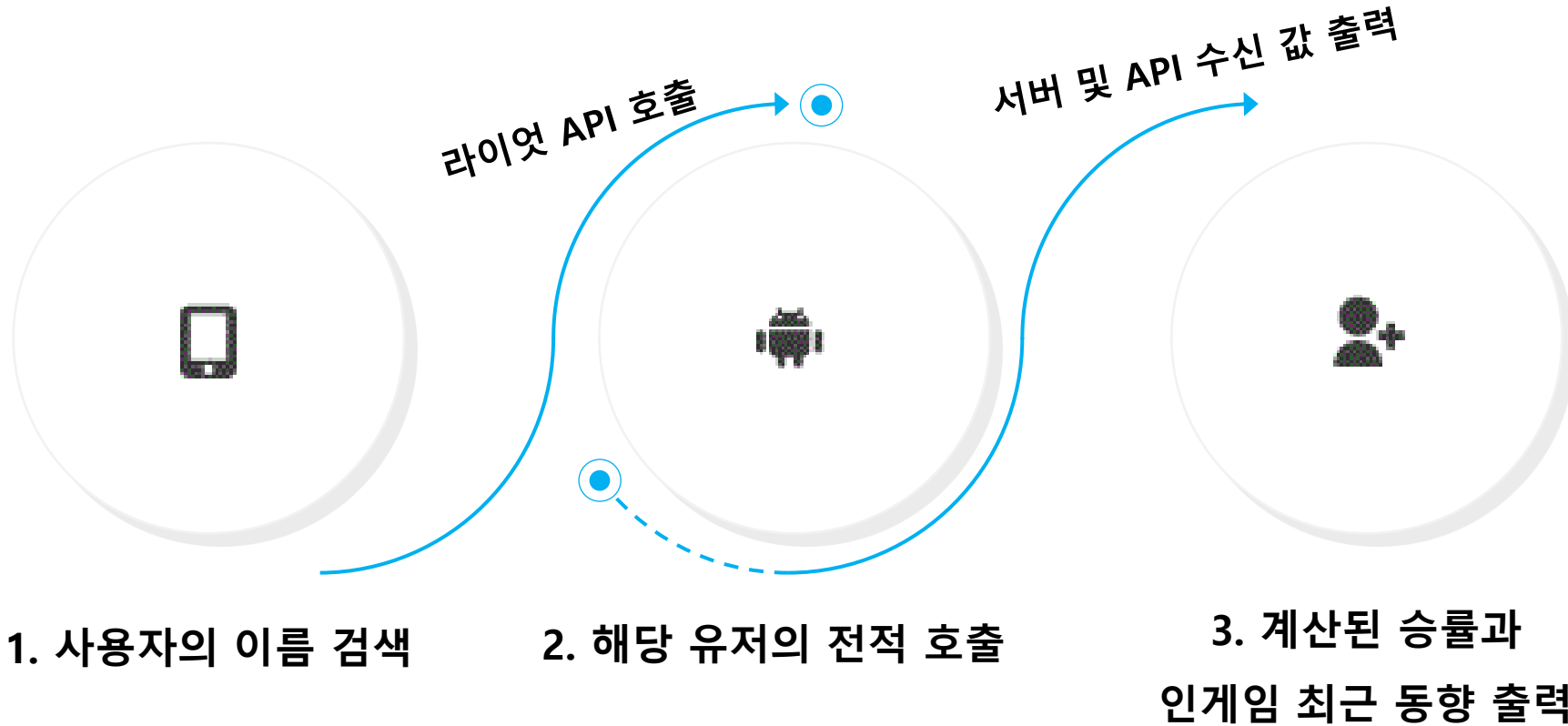
- 모델 트레이닝

Jupyter notebook, Xgboost

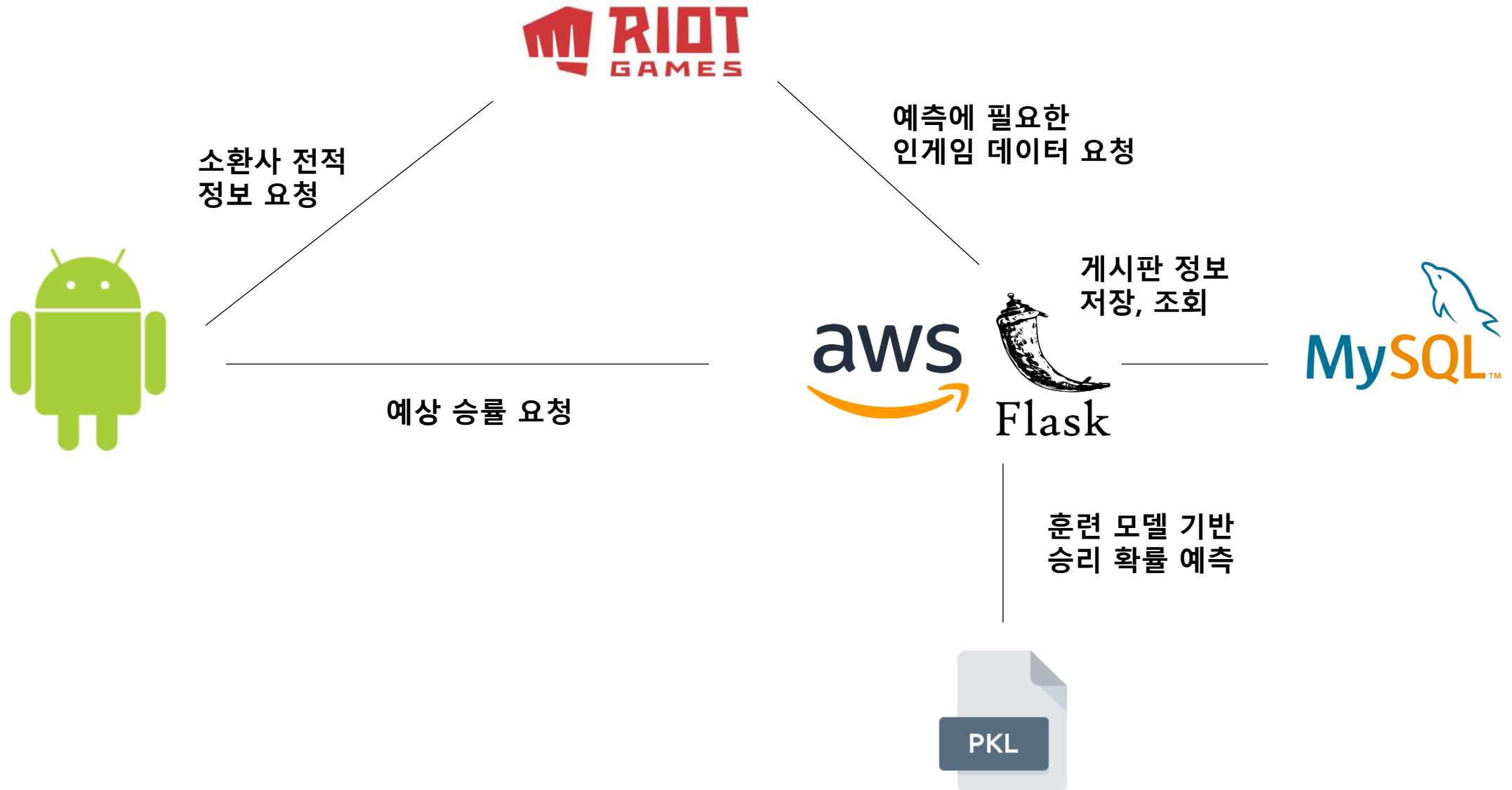


XGBoost

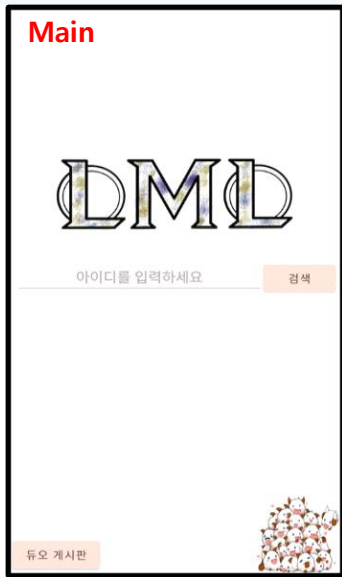
3. 애플리케이션 개요



3. 애플리케이션 개요 - 아키텍처

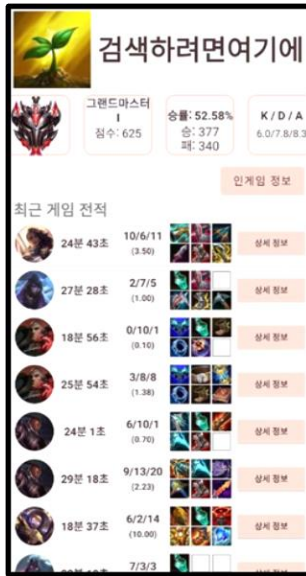


3. 애플리케이션 개요 - 화면 흐름



전적검색

사용자의
닉네임 입력



인게임

해당 유저의
인게임 전적 출력

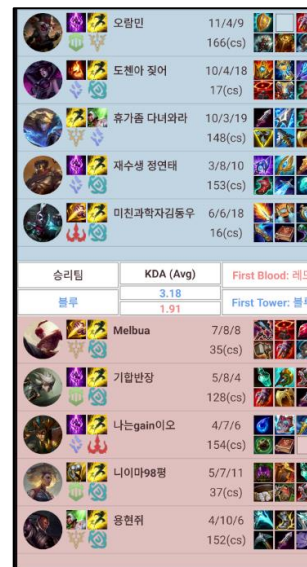


승률

진행중인 게임의
승률 출력



종료된 게임의
세부 정보를 출력



게시판 이동



같이 게임하고자 하는
유저의 개인 전적 검색 가능

듀오

상세정보

4. 상세기술 - 모델 설명

1. 수집 방법: <https://developer.riotgames.com/>에서 제공하는 API를 통한 데이터 수집.

// 수집 데이터는 데이터베이스에 저장

2. 수집한 데이터: p0~p9까지의 Kill, Deaths, Assists, gold, dealt, taken, vision의 10게임 간 평균값

id	blue_win	p0_kills	p0_deaths	p0_assists	p0_gold	p0_dealt	p0_taken	p0_vision
필터	필터	필터	필터	필터	필터	필터	필터	필터
5168549393	Win	1.0	5.4	14.8	273.458	238.934	581.464	47.4

The screenshot shows the Riot Games Developer API portal. At the top, there is a navigation bar with 'DEVELOPER' and links for 'APIS', 'DOCS', 'POLICIES', 'STATUS', and 'CHANNELS'. A user profile icon for '석주윤' is visible in the top right. The main content area displays a message: 'YOUR DEVELOPMENT API KEY HAS EXPIRED!' followed by a note: 'There are three routing values for account-v1; americas, asia, and europe. You can query for any account in any region. We recommend using the nearest cluster.' Below this, a list of endpoints for 'ACCOUNT-V1' is shown, including: 'GET /riot/account/v1/accounts/by-puuid/{puuid}', 'GET /riot/account/v1/accounts/by-riot-id/{gameName}/{tagLine}', 'GET /riot/account/v1/active-shards-by-game/{game}/by-puuid/{puuid}', and 'GET /riot/account/v1/accounts/me'. A sidebar on the left lists various API categories like 'ACCOUNT-V1', 'CHAMPION-MASTERY-V4', 'CHAMPION-V3', 'CLASH-V1', 'LEAGUE-EXP-V4', 'LEAGUE-V4', and 'LOL-STATUS-V3'.

4. 상세기술 - Feature 선정

Feature 선정:

p0_kills	p0_deaths	p0_assists	p0_gold	p0_dealt	p0_taken	p0_visic
필터	필터	필터	필터	필터	필터	필터
6.4	4.0	6.8	466.332	784.37	786.132	1

(p0~p9)

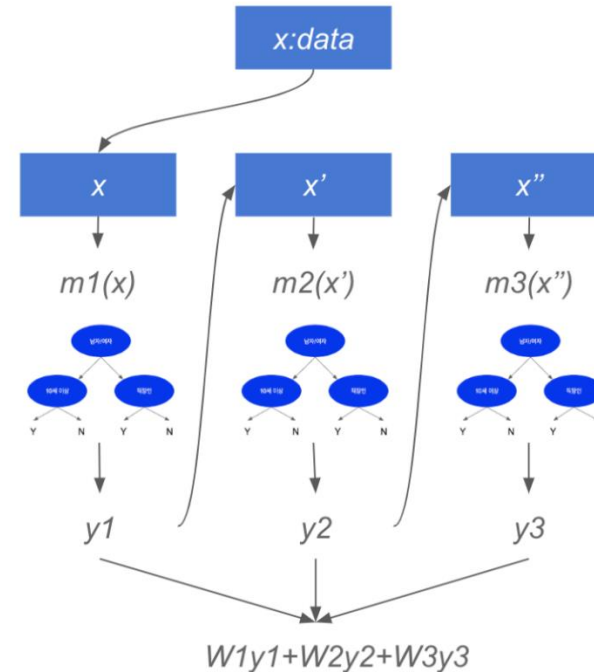
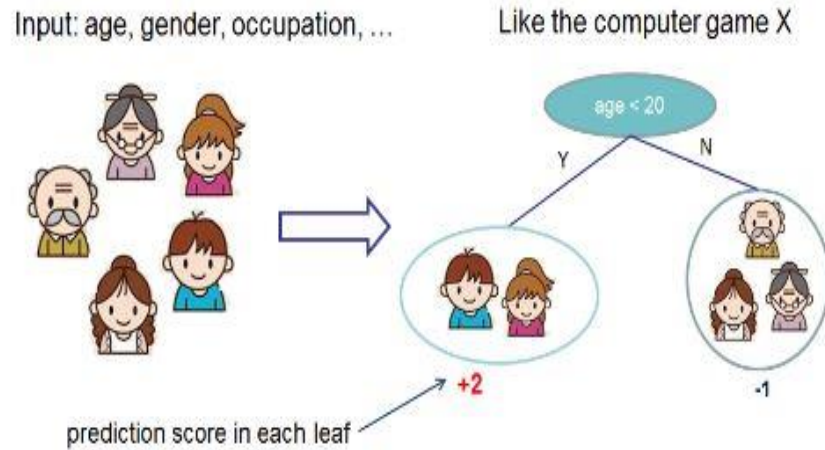
선정 방식

1. 게임의 결과에 큰 영향이 안가는 기본적인 스텟들 배제
ex) 넥서스 킬, 중립 몬스터 킬 등
2. 시간 별 스텟은 Duration 곱해서 단일화 (XpPerMindeltas와 같은 분당 스텟들)
3. Item 및 유저 개인정보에 해당하는 스텟 전부 삭제

4. 상세기술 - 머신러닝

XGBoost

1. Gradient Boosting 알고리즘을 분산환경에서도 실행할 수 있도록 구현해놓은 라이브러리
2. 여러개의 Decision Tree를 조합해서 사용하는 Ensemble 알고리즘



4. 상세기술 - 각 모듈 간 통신



```
private static RetroServerBuild instance = new RetroServerBuild();

public static RetroServerBuild getInstance() { return instance; }

private RetroServerBuild() { }

OkHttpClient okHttpClient = new OkHttpClient.Builder()
    .connectTimeout( timeout: 30, TimeUnit.SECONDS)
    .readTimeout( timeout: 30, TimeUnit.SECONDS)
    .writeTimeout( timeout: 30, TimeUnit.SECONDS)
    .build();

Retrofit retrofit = new Retrofit.Builder()
    .baseUrl("http://ec2-13-209-65-77.ap-northeast-2.compute.amazonaws.com:5000/")
    .addConverterFactory(GsonConverterFactory.create())
    .client(okHttpClient)
    .build();

GetServerService service = retrofit.create(GetServerService.class);
public GetServerService getService() { return service; }

public interface GetServerService {

    @POST("/board")
    Call<String> postBoard(@Body Map<String, String> data);

    @GET("/board")
    Call<List<Board>> getBoard();

    @GET("/LoL/ingame/{name}")
    Call<Ingame> getIngameData(@Path("name") String name);

    @GET("/LoL/{name}")
    Call<Player> getKDA(@Path("name") String name);
}
```

4. 상세기술 - 웹 서버

API ^{1.0}
[Base URL: /]
/swagger.json

default Default namespace ^

- GET /board
- POST /board
- GET /lol/ingame/{name}
- GET /lol/{name}

기능

1. 라이엇 API 호출을 통한 사용자의 최근 전적 평균값 계산(/lol/ingame/{name})
2. 사용자의 전적관련 정보 호출(/lol/{name})
3. 앱의 듀오 게시판과 연동하여 데이터를 주고받음(/board)

5. 시연 영상



6. 마무리 - 배운 점

1. 개발 환경과 개발 언어에 대한 이해도 증가
2. Restful API에 대한 이해도 증가
3. Git을 통한 형상 관리와 협업 능력 향상
4. 머신 러닝간 배운 점

6. 마무리 - 발표 자료 출처

국내 e스포츠 산업 규모 1,398억 원... 전년 대비 22.8% 증가
<https://www.inven.co.kr/webzine/news/?news=247623>

세계 e스포츠 시장 규모 1조 육박...한국도 1,000억 규모로 성장
http://www.dailyesports.com/view.php?ud=201901151816386443283f60ce7c_27

"e스포츠 시장규모, 2022년 30억달러 육박 예상"
<https://www.sportsw.kr/news/newsview.php?ncode=1065578033228520>

주요 e스포츠 온라인 중계 확대...시청자 수도 증가 전망

이스포츠 관객규모 (출처 : Statista, 2019)
<http://www.kocca.kr/trend/vol21/sub/s32.html>

<https://www.gamespot.com/articles/1100-6405485/>

<http://www.gametrics.com/>

<http://www.inews24.com/view/1349141>

감사합니다 :)