



TFT 사무실

? 기획 배경

- 최근 제일 자주하고 간단히 즐기기 좋은 게임인 **Teamfight Tactics(전략적 팀 전투)** 관련 정보 사이트가 있긴 했지만 내가 원하는 정보는 없는 경우가 있었고, 있는 경우에도 다른 정보와 같이 있어 보기 힘든 경우도 있었다.
- 내가 알고싶은 **정보와 추천 영상**같은 메타 관련 영상을 모아서 보고싶었다.

👉 해결 컨셉 : Easy One For All

1) 나에게 필요한 정보만 **골라서 직관적**으로 편하게 보자

→ 복잡한 정보와 다양한 기능은 쓰던 앱으로 하면 된다.

2) 간편하게 **모바일**로 볼 수 있는 웹 어플리케이션

👤 목표: 기대 효과(USER)

여러 포털 사이트 / 앱을 사용하지 않아도 원하는 정보를 한 사이트에서 확인할 수 있다

원하지 않는 정보는 보지 않도록 모듈화된 기능을 선택하여 나만의 입맛에 맞는 페이지를 구성한다.

→ 최종 총 5개의 페이지 각 페이지 별로 다른 정보 제공.

Repository / Branch Manangement

- 본 프로젝트는 메인 + 기능별 4개, 총 5개 페이지로 구성. HTML, CSS, JavaScript만을 이용한 간단한 페이지
- 1인 제작

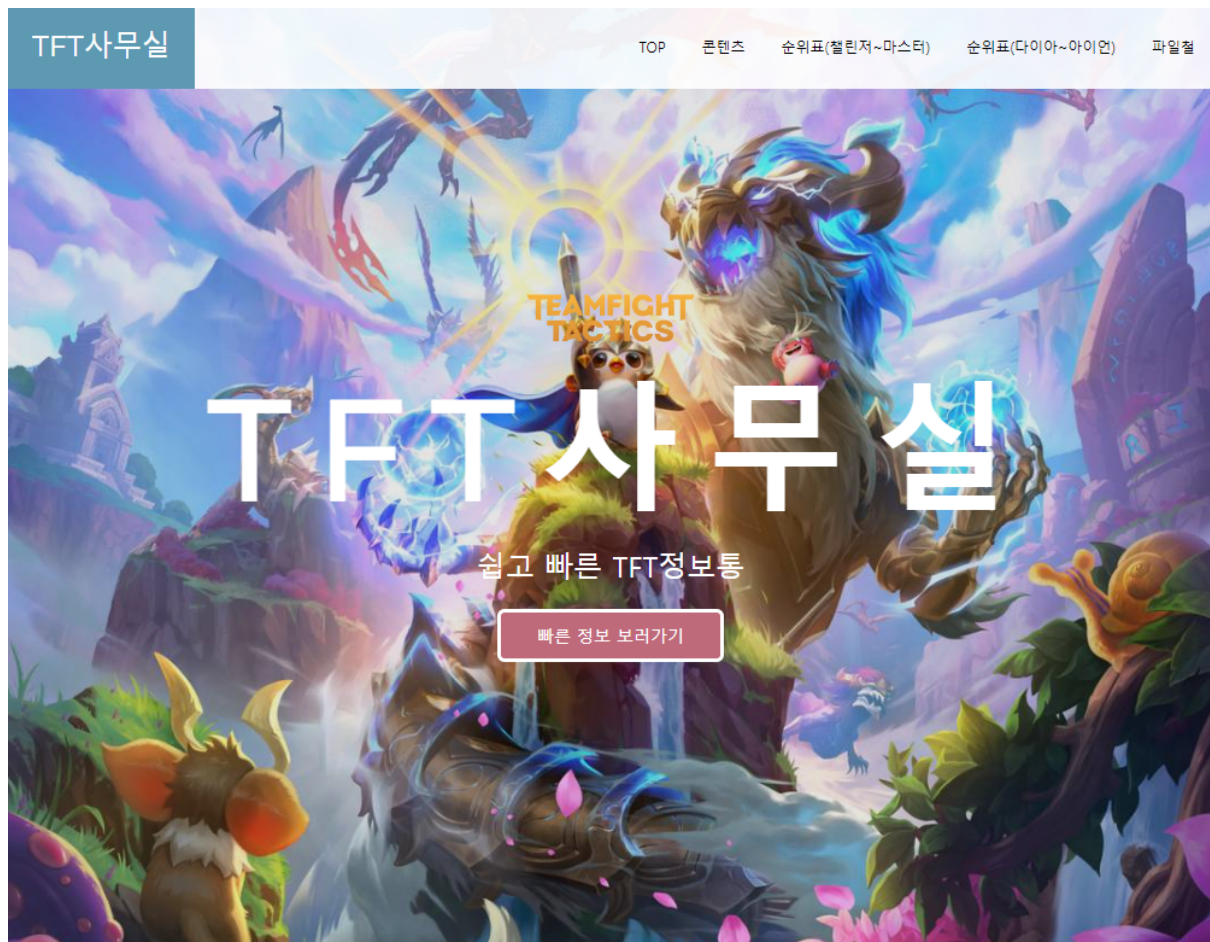
기술 스택

FrontEnd HTML JavaScript CSS

프로젝트 주요기능

메인페이지

총괄 페이지로 실행시 보이는 화면이고 이 페이지를 기준으로 다른 페이지로 이동이 가능하다.



사이드 페이지

- 콘텐츠
 - 해당 시즌의 챔피언, 증강체, 시너지 등 게임에 필요한 정보를 얻을 수 있다.
- 순위표

- 상위 티어의 순위와 하위티어의 순위를 확인할 수 있다.
- 리그오브레전드에서 제공하는 api를 가져와서 활용하기에 주기적인 api 업데이트는 필요하지만 실시간으로 변화하는 순위를 확인 가능하다.
- 파일철
 - 추천메타,패치노트,통계등 원하는 것들을 확인할 수 있다.
 - 추가적으로 필요한 정보는 하이퍼링크를 통해 각종 페이지로 이동 가능하다.

회고 및 트러블 슈팅

Front-End

- **HTML , CSS** 구성
 - 첫 개인 프로젝트이다 보니까 손수 처음부터 모든 걸 제작하는 것에 대해서 굉장히 많은 시간이 들었었다. 프로젝트 기간이 대략 8일정도였음에도 4일이상을 HTML,CSS를 구성하는데 들었던 것이 완성도가 모자랐던 이유 중에 하나였다고 생각합니다.
 - 딱히 특별한 해결 방법이 있었던건 아니었고 구글링을 통한 모든 수치를 직접 입력해보며 필요한 수치를 확인하여 입력했습니다.
- **API 오류**
 - API 제공 사이트에서 API를 가져오기 위해 JSON과 XML중 JSON 코드를 사용하는데 이해가 부족했어서 처음부터 사용법을 익히는데 생각보다 오랜 시간이 걸렸었다는 점
 - JSON을 사용하여 API키와 API를 수용할 url을 구성해서 데이터를 받아 조건부 렌더링했습니다.

느낀 점

```
<script>
  var api_key = "RGAPI-31e2934f-3f6e-4d0e-a99e-ab558fefa3e7";

  function removeTable(){
    document.getElementById('demoJSON').removeChild(document.getElementById('demoJSON').childNodes[0]);
  }
</script>
```

```

function loadData() {

    let url, xhttp;

    url = `https://kr.api.riotgames.com/tft/league/v1/challenger?api_key=${api_key}`;

    xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function(){
        if(this.readyState == 4 && this.status == 200){
            // alert(this.responseText);
            let jsonObj = JSON.parse(this.responseText);
            parseJSON(jsonObj);
        }
    };
    xhttp.open("GET", url, true);
    xhttp.send();

} // end loadData()

```

// 기본적인 API를 활용하는 프로젝트였지만 해당 프로젝트를 통해 수많은 데이터들을 내가 원하는 것을 골라 활용할 수 있는 방법을 알게 되는 좋은 시간이었습니다. 비록 프로젝트로써 완성도는 부족할 수도 있지만 이번 프로젝트를 통해 웹페이지의 여러가지를 만들 수 있다는 개념을 얻을 수 있었습니다.