horizontal line

시세 App 테크 스펙

(React-Navite 시세 APP with TDD)

**작성자: 조영수**

**조직: 플랫폼 개발**

**마지막 수정일: 2022-06-27**

# 요약 (Summary)

사용자들이 많이 찾는 게임을 대상으로 15일간의 시세와 등락 폭, 거래 순위 등을 볼 수 있는 어플리케이션 개발

세부적인 정보의 출력은 웹앱으로 이동하며, 간단한 기능들 만이 존재하는 어플리케이션

game\_list에서 게임명을 가져와 순서대로 출력해주고, 세부 정보는 사용자가 게임을 선택했을 때 server\_list에서 게임의 서버 목록이 출력

서버를 선택했을 때 서버별로 15일 간의 시세 정보를 출력

서버별 15일간의 시세 정보에는 전일 시세와 비교해 등락 폭과 아이콘을 출력

집계된 순위를 이용해 이전 순위와 비교해 등락을 출력

관심게임 등록을 통해 사용자가 자주 확인하는 서버의 시세 정보에 빠른 접근을 지원

세부 페이지에서는 뒤로가기를 이용해 이전 페이지로 이동가능

# 계획 (Plan)

[구조]

index.js에서는 App.js를 연결

App.js에서 react-router-native를 이용해 모든 페이지를 연결, 새로고침해도 유지해야 할 State는 App.js에서 선언 후 전달

모든 페이지에서 공통으로 보여지는 HeaderTop 하단에 Outlet을 이용해 다른 페이지를 출력

아이콘 부분은 전부 IconBox.js 에서 관리하고, 템플릿 부분은 Common.js 에서 관리한다.

Common.js 에서는 item1, item2, IconWrap 등을 관리?

[index.js]

import App.js

[App.js]

[tab, setTab] : 최상단인 App.js에서 State를 관리해 뒤로가기 시 이전 tab을 읽어올 수 있도록 만들기

- 동작하지 않을 경우 localStorage에 저장하고 읽어오는 방향으로 작성

[loading, setLoading]

로딩 완료시 setLoading을 이용해 Loading.js > Main.js 출력

import 출력할 모든 페이지

loading이 true일 때 <Loading />출력

false 일 때

<NativeRouter>

<Route path"/" component={HeaderTop}>

<Route index component={Main} />

<Route path="/games:gameName" component={Games}>

<Route path="/games/:gameName/servers/:serverName" component={Servers} />

</Route>

<Route path="/settings" component={Settings}>

<Route path="/privateInfo" component={PrivateInfo} />

</Route>

<Route>

<NoMatch />

</Route>

<Route>

</NativeRouter>

[테스트]

jest & ReactTestingLibrary를 이용해서 테스트

description을 이용해 비슷한 테스트끼리 묶어서 출력