

React-native Weather app

초단기실황앱

CLI vs Expo

Expo

- 장점

- 편리하고 유용한 특성
- 배포가 편함. 쉬운 업데이트 버전 관리
- React Native를 처음 시작하는 개발자에게 편리함

- 단점

- Expo에서 제공하는 기능만 사용 가능
- 모듈을 만들어 사용 불가
- native 파일 제어 불가

CLI

- 장점

- 개발자가 원하는 기능, 모든 요소 제어 가능
- 다양한 라이브러리 사용 가능
- React Native를 많이 사용해본 전문가에게 편리함

- 단점

- expo에 비해 편리하지 않음
- 기본적 설정이 되어 있지 않음 (사용자 설정 필요)
- android studio, xcode를 다 설치하여 빌드 배포해야함.

왜 나는 CLI를 사용했을까?

앱을 처음 접해서 잘 모르기 때문에 Expo를 사용하는 것을 고민했었다. 하지만 한정된 라이브러리 및 Expo에서만 제공하는 기능만 쓰면 추후에 나 자신이 제대로 성장하지 못 할 것이라 생각했다.

그렇기에 개발 환경, 라이브러리 설치 및 오류 해결 등 경험을 직접 해보아야지만 성장 할 수 있다고 생각했다.

설치 참고 사이트

- <https://reactnative.dev/docs/environment-setup> –Document
- <https://adjh54.tistory.com/34>

에뮬레이터 환경

Platform	Device	OS
Android	Pixel 4	Android 12
IOS	iPhone 13	iOS 15.5

사용된 라이브러리

- **Axios**

- *npm install -save axios*

- **react-native-permissions**

- *npm install --save react-native-permissions*

- **react-native-permissions**

- *npm install --save react-native-geolocation-service*

Android 위치 정보 권한 받기

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.weather">

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <application
        android:name=".MainApplication"
        android:label="@string/app_name"
        android:icon="@mipmap/ic_launcher"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:allowBackup="false"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name"
            android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|screenSize|smallestScreenSize|uiMode"
            android:launchMode="singleTask"
            android:windowSoftInputMode="adjustResize"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

추가

android > app > main > AndroidManifest.xml

iOS 위치정보 권한 받기

```
<key>NSLocationWhenInUseUsageDescription</key>
<string>When In Use App?</string>
<key>NSLocationAlwaysAndWhenInUseUsageDescription</key>
<string>When asd In Use App?</string>
<key>UILaunchStoryboardName</key>
<string>LaunchScreen</string>
<key>UIRequiredDeviceCapabilities</key>
<array>
    <string>armv7</string>
</array>
<key>UISupportedInterfaceOrientations</key>
<array>
    <string>UIInterfaceOrientationPortrait</string>
    <string>UIInterfaceOrientationLandscapeLeft</string>
    <string>UIInterfaceOrientationLandscapeRight</string>
</array>
<key>UIViewControllerBasedStatusBarAppearance</key>
<false/>
```

추가

ios > app > Info.plist

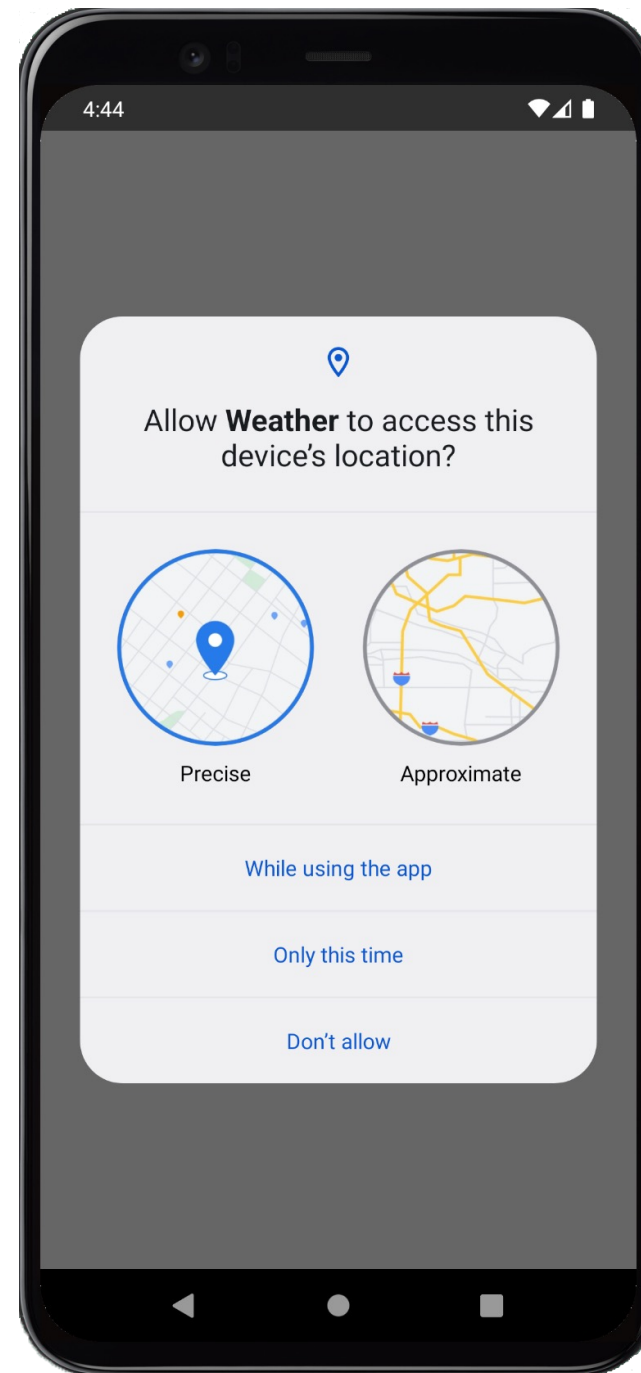
권한 받는 코드

```
requestLocationPermission = async () => {
  try {
    if(Platform.OS === 'android'){
      const granted = await PermissionsAndroid.request(
        PermissionsAndroid.PERMISSIONS.ACCESS_FINE_LOCATION
      );
      if (granted === PermissionsAndroid.RESULTS.GRANTED) {
        Geolocation.watchPosition(
          position => {

            const { latitude, longitude } = position.coords;
            this.fetchWeather(latitude,longitude);

            this.setState({
              latitude: latitude,
              longitude: longitude,
            })
          },
          error => {
            console.log(error.code, error.message);
          },
          {enableHighAccuracy : true, interval : 5000, distanceFilter : 5}
        );
      } else {
        console.log("Location permission denied");
      }
    }
    else{

      Geolocation.requestAuthorization("whenInUse");
      this.getGeoLocation();
    };
  } catch (err) {
    console.warn(err);
  }
}
```



날씨 정보 API

날씨 앱 제작 블로그 등 대부분 [OpenWeather](#) API 사용하는 것을 보았다. 이 API는 되게 편하게 잘 되어 있었지만, 대한민국의 날씨 정보는 기상청에서 받는게 더 좋다고 판단했으며, 또한 추후에 [공공데이터](#)에서 API받을 때 도움이 될 것이라 생각했습니다. 결과적으로 [기상청 단기예보](#) API를 사용했습니다

API요청 코드

```
fetchWeather = async (latitude, longitude) => {  
  try{  
    var toXY = dfsXyConv ('toXY',latitude, longitude)  
    const encodeURL = encodeURIComponent("") //API Key  
    const date = new Date().toISOString().split('T')[0].replace(/-/g,"")  
    const hour = new Date().getHours()  
    const minutes = new Date().getMinutes()  
    const time = String(hour-1)+ String(minutes)  
    const response = await axios.get("http://apis.data.go.kr/1360000/VilageFcstInfoService_2.0/getUltraSrtNcst",{  
      params: {  
        serviceKey: encodeURL,  
        pageNo: "1",  
        numOfRows: "15",  
        dataType: "JSON",  
        base_date: date,  
        base_time: time,  
        nx: toXY[0],  
        ny: toXY[1]  
      })  
    }  
    const json = response.data.response.body.items  
    const jsonstring = JSON.stringify(json.item,["category","obsrValue"])  
    const parsejson = JSON.parse(jsonstring)  
    console.log(parsejson)  
    this.setState({  
      response:json,  
      t1h:parsejson[3].obsrValue,  
      rn1: parsejson[2].obsrValue  
    })  
  }  
  catch (e){  
    console.log("Error : ", e)  
  }  
}
```

api 파라미터에 위,경도 값이 아닌 기상청의
격자 값을 받기 때문에 변환 함수

JSON 데이터의 key, value 값 추출

Response 데이터 포맷

초단기실황↵	T1H↵	기온↵	°C↵	10↵
	RN1↵	1시간 강수량↵	mm↵	8↵
	UUU↵	동서바람성분↵	m/s↵	12↵
	VVV↵	남북바람성분↵	m/s↵	12↵
	REH↵	습도↵	%↵	8↵
	PTY↵	강수형태↵	코드값↵	4↵
	VEC↵	풍향↵	deg↵	10↵
	WSD↵	풍속↵	m/s↵	10↵

기온, 강수량을 추출하여
사용했습니다

위, 경도 변환 함수

기상청에서는 위, 경도 값으로 위치 표시하지 않으며, 대한민국을 x , y 로 격자를 만들어서 데이터를 받고 있다. 기상청 홈페이지 내부에서 변환 소스가 있으며, 변환 부분만 발췌한 소스가 [github](#)에 있어 이를 사용했습니다.

Android Axios Network error 오류 해결

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <application
        android:usesCleartextTraffic="true"
        tools:targetApi="28"
        tools:ignore="GoogleAppIndexingWarning">
        <activity android:name="com.facebook.react.dev.support.DevSettingsActivity" android:exported="false" />
    </application>
</manifest>
```

추가

android > app > debug > AndroidManifest.xml

iOS Axios Network error 오류 해결

```
<key>NSAppTransportSecurity</key>
<dict>
  <key>NSAllowsArbitraryLoads</key>
  <true/>
  <key>NSExceptionDomains</key>
  <dict>
    <key>localhost</key>
    <dict>
      <key>NSExceptionAllowsInsecureHTTPLoads</key>
      <true/>
    </dict>
  </dict>
</dict>
</dict>
```

추가

Api 요청 시 https를
사용하지 않아서
발생하는 오류

ios > app > Info.plist

디자인 참고한 소스들

Response 값들 중 사용되는 값이 많지 않기에 직관적인 UI를 찾았으며, 다음 사진과 같은 디자인들을 참고하여 만들었습니다.



날씨 및 시간에 따라 바뀌는 디자인

```
render() {  
  const hour = new Date().getHours()  
  (() => {  
    if(this.state.rn1 > 0){  
      BackgroundTime = <View style={ styles.background_rain}></View>  
      WeatherIcon = <Image style={styles.img} source={require('./img/rain.png')}></Image>  
    }  
    else{  
      if (6 < hour && hour < 19){  
        BackgroundTime = <View style={ styles.background_afternoon}></View>  
        WeatherIcon = <Image style={styles.img} source={require('./img/sun.png')}></Image>  
      }  
      else{  
        BackgroundTime = <View style={ styles.background_night}></View>  
        WeatherIcon = <Image style={styles.img} source={require('./img/night.png')}></Image>  
      }  
    }  
  })()  
}
```

Rn1이 강수량에 관한 변수입니다. 강수량이 0mm
이상이면 비 아이콘으로 설정하게 하는 코드

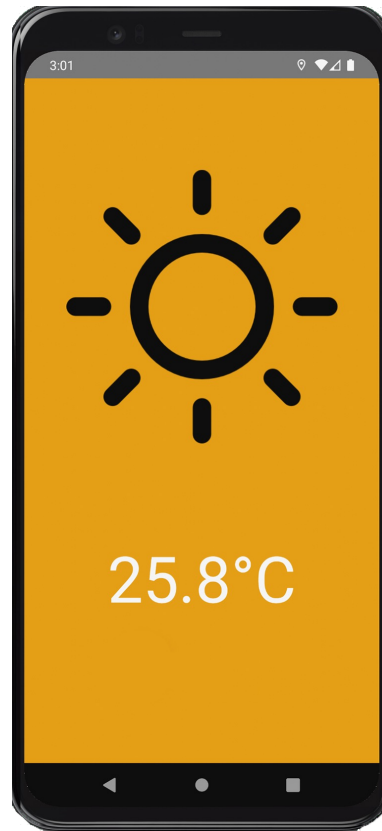
낮과 밤을 나누는 코드

07~18시는 낮 그 이외의 시간은 밤

날씨 및 시간에 따라 바뀌는 디자인

```
return (  
  
  <View style={ styles.Container}>  
    {this.state.response ? ( Response 값이 없으면 로딩 아이콘 띄우는 JSX구문  
      <>  
        {BackgroundTime}  
        <View style={styles.img_area}>  
          {WeatherIcon}  
        </View>  
        <Text style={styles.Label}>{this.state.t1h}°C</Text>  
        <Text style={styles.Label}>  
        </Text>  
      </>  
    ) : (  
      <ActivityIndicator style={styles.horizontal} size="large" color="black"/>  
    )}  
  </View>  
);
```

Android 결과물



iOS 결과물



느낀점

처음 프로젝트를 시작했을 때 하나하나가 정말 막막했었는데 계속 삽질하면서 차츰차츰 전체적인 흐름 등 이해해 나아갔습니다. 아직 앱에 대해 깊게 공부하지는 않았지만, 전체적으로 웹과 비슷한 느낌을 받았습니다. 또한 아직 권한 받는 기능에 대해 미숙하여 추가적으로 공부가 더 필요하다고 느낍니다.