

Mobil Programlama

Uygulama 1

React Native

- Download Node.js (<https://nodejs.org/en/>)
- `npm install expo-cli --global`
- `expo init Proje1`
- `cd Proje1`

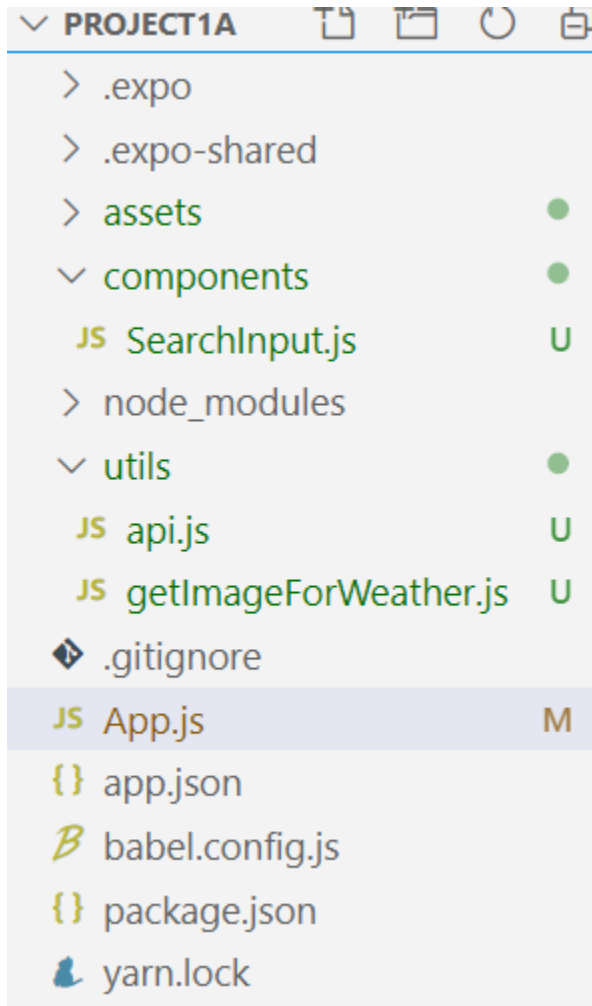
`expo start`

`yarn start`

`yarn ios`

`yarn android`

React Native Proje Yapısı



- **expo** /, Expo paketleyici ve cihaz bağlantı ayarları için konfigürasyonları tanımlayan dosyalar içerir.
- **assets** / varsayılan olarak birkaç görüntü varlığı içerir. Bunlar uygulama simgesi ve açılış ekranı görüntüleridir.
- **node_modules** / uygulamamızdaki tüm üçüncü taraf paketlerini içerir. Yeni bağımlılıklar ve geliştirme bağımlılıkları buraya gelir.
- **gitignore**, hangi dosyaların Git tarafından yok sayılacağını belirttiğimiz yerdir. Hem node_modules / hem de .expo / dizinlerinin zaten dahil edildiğini görebiliriz.
- **App.js**, uygulama kodumuzun bulunduğu yerdir.
- **app.json**, Expo uygulamamız hakkında bilgi eklememizi sağlayan bir yapılandırma dosyasıdır. Bu dosyaya dahil edilebilecek özelliklerin listesi dokümantasyonda²⁷ listelenmiştir. Burada değiştirilebilecek özellik örnekleri arasında uygulama simgesi ve açılış ekranı bulunur.
- **babel.config.js**, Babel²⁸'yi yapılandırmak için ön ayarları ve eklentileri tanımlamamıza olanak tanır. Daha önce de bahsettiğimiz gibi Babel, daha yeni deneysel JavaScript'i farklı platformlarla uyumlu kalması için eski sürümlere derleyen bir aktarıcıdır.
- **package.json**, uygulamanın bilgilerini paket yöneticimize sağladığımız ve tüm proje bağımlılıklarımızı belirlediğimiz yerdir.

<https://code.visualstudio.com/>

File Edit Selection View Go Run Terminal Help App.js - Project1a - Visual Studio Code

EXPLORER

> OPEN EDITORS

PROJECT1A

- > .expo
- > .expo-shared
- > assets
- components
 - JS SearchInput.js
- > node_modules
- utils
 - JS api.js
 - JS getImageForWeather.js
- .gitignore
- JS App.js
- app.json
- babel.config.js
- package.json
- yarn.lock

TIMELINE

NPM SCRIPTS

JS App.js

```
1 import React from 'react';
2 import {
3   StyleSheet,
4   View,
5   ImageBackground,
6   Text,
7   KeyboardAvoidingView,
8   Platform,
9   ActivityIndicator,
10  StatusBar,
11 } from 'react-native';
12
13 import { fetchLocationId, fetchWeather } from '../utils/api';
14 import getImageForWeather from '../utils/getImageForWeather';
15
16 import SearchInput from '../components/SearchInput';
17
18 export default class App extends React.Component {
19   constructor(props) {
20     super(props);
21
22     this.state = {
23       loading: false,
24       error: false,
25       location: '',
26       temperature: 0,
27       weather: '',
28     };
29   }
30 }
```

Ln 37, Col 1 Spaces: 2 UTF-8 LF JavaScript Prettier

master*+ 0 0

Type here to search

ENG TRQ 14:57 10.10.2020



EXPLORER



JS App.js X



> OPEN EDITORS

▼ PROJECT1A

▼ .expo

{ } packager-info.json

{ } settings.json

▼ .expo-shared

{ } assets.json

> assets

> node_modules

.gitignore

JS App.js M

{ } app.json

B babel.config.js

{ } package.json

yarn.lock

JS App.js > [] styles

```
1  import React from 'react';
2  import { StyleSheet, Text, View } from 'react-native';
3
4  export default function App() {
5    return (
6      <View style={styles.container}>
7        <Text>Merhaba React Native</Text>
8      </View>
9    );
10 }
11
12 const styles = StyleSheet.create({
13   container: {
14     flex: 1,
15     backgroundColor: '#fff',
16     alignItems: 'center',
17     justifyContent: 'center',
18   },
19 });
20
```

TERMINAL

PROBLEMS

OUTPUT

DEBUG CONSOLE

1: node



Running application on SM-N920C.

> TIMELINE

> NPM SCRIPTS



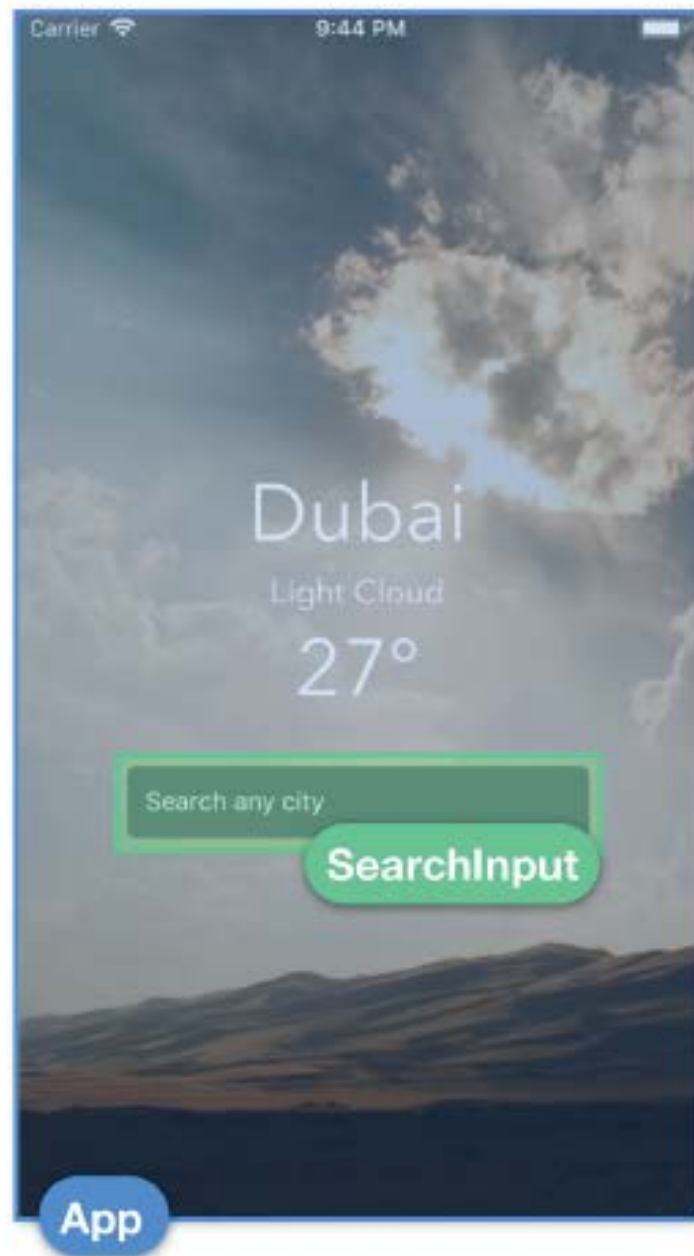
📱 DEVICE (1) - 11:37:54 PM

[Publish or republish project...](#)

Local

exp://192.168.1.34:19000

 METRO BUNDLER



Uygulama




```
import React from 'react';
import { StyleSheet, Text, View, Platform } from 'react-native';

export default function App() {
  return (
    <View style={styles.container}>
      <Text style={[styles.largeText, styles.textStyle]}>
        London
      </Text>
      <Text style={[styles.smallText, styles.textStyle]}>
        Light Cloud
      </Text>
      <Text style={[styles.largeText, styles.textStyle]}>24°</Text>
    </View>
  );
}
```

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#fff',
    alignItems: 'center',
    justifyContent: 'center',
  },
  textStyle: {
    textAlign: 'center',
    fontFamily:
      Platform.OS === 'ios' ? 'AvenirNext-Regular' : 'Roboto',
  },
  largeText: {
    fontSize: 44,
  },
  smallText: {
    fontSize: 18,
  },
});
```



```
import {StyleSheet, Text, KeyboardAvoidingView, Platform, TextInput,} from 'react-native';

export default class App extends React.Component {
  render() {
    return (
      <KeyboardAvoidingView
        style={styles.container}
        behavior="height"
      >
        <Text style={[styles.largeText, styles.textStyle]}>
          London
        </Text>
        <Text style={[styles.smallText, styles.textStyle]}>
          Light Cloud
        </Text>
        <Text style={[styles.largeText, styles.textStyle]}>24°</Text>

        <TextInput
          autoComplete={false}
          placeholder="Search any city"
          placeholderTextColor="white"
          style={styles.textInput}
          clearButtonMode="always"
        />
      </KeyboardAvoidingView>
    );
  }
}
```



```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#fff',
    alignItems: 'center',
    justifyContent: 'center',
  },
  textStyle: {
    textAlign: 'center',
    fontFamily:
      Platform.OS === 'ios' ? 'AvenirNext-Regular' : 'Roboto',
  },
  largeText: {
    fontSize: 44,
  },
  smallText: {
    fontSize: 18,
  },
  textInput: {
    backgroundColor: '#666',
    color: 'white',
    height: 40,
    width: 300,
    marginTop: 20,
    marginHorizontal: 20,
    paddingHorizontal: 10,
    alignSelf: 'center',
  },
});
```

▼ PROJECT1A

- > .expo expostart
- > .expo-shared
- > assets
- ▼ components ●
- JS SearchInput.js** U
- > node_modules
- ◆ .gitignore
- JS App.js M
- { } app.json
- B babel.config.js
- { } package.json

```
1 import React from 'react';
2 import { StyleSheet, TextInput, View } from 'react-native';
3
4 export default class SearchInput extends React.Component {
5   render() {
6     return (
7       <View style={styles.container}>
8         <TextInput
9           autoComplete={false}
10          placeholder={this.props.placeholder}
11          placeholderTextColor="white"
12          underlineColorAndroid="transparent"
13          style={styles.textInput}
14          clearButtonMode="always"
15        />
16      </View>
17    );
18  }
19 }
```

```
const styles = StyleSheet.create({
  container: {
    height: 40,
    width: 300,
    marginTop: 20,
    backgroundColor: '#666',
    marginHorizontal: 40,
    paddingHorizontal: 10,
    borderRadius: 5,
  },
  textInput: {
    flex: 1,
    color: 'white',
  },
});
```

PROJECT1A

> .expo

> .expo-shared

> components

JS SearchInput.js

{}

 assets.json

> assets

> node_modules

.gitignore

JS App.js

{}

 app.json

B

 babel.config.js

{}

👤

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

Text,

KeyboardAvoidingView,

Platform,

} from 'react-native';

import SearchInput from './components/SearchInput';

export default class App extends React.Component {

render() {

return {

<KeyboardAvoidingView

style={styles.container}

behavior="height"

>

<Text style={[styles.largeText, styles.textStyle]}>

London

</Text>

<Text style={[styles.smallText, styles.textStyle]}>

Light Cloud

</Text>

<Text style={[styles.largeText, styles.textStyle]}>24°</Text>

<SearchInput placeholder="Search any city" />

</KeyboardAvoidingView>

};

}

}



▼ assets	●
🖼️ clear.png	U
🖼️ favicon.png	
🖼️ hail.png	U
🖼️ heavy-cloud.png	U
🖼️ heavy-rain.png	U
🖼️ icon.png	M
🖼️ light-cloud.png	U
🖼️ light-rain.png	U
🖼️ showers.png	U
🖼️ sleet.png	U
🖼️ snow.png	U
🖼️ splash.png	
🖼️ thunder.png	U

> .expo	
> .expo-shared	
> assets	●
▼ components	●
JS SearchInput.js	U
> node_modules	
▼ utils	●
JS getImageForWeather.js	U
📄 .gitignore	
JS App.js	M

```

2
3  const images = {
4    Clear: require('../assets/clear.png'),
5    Hail: require('../assets/hail.png'),
6    'Heavy Cloud': require('../assets/heavy-cloud.png'),
7    'Light Cloud': require('../assets/light-cloud.png'),
8    'Heavy Rain': require('../assets/heavy-rain.png'),
9    'Light Rain': require('../assets/light-rain.png'),
10   Showers: require('../assets/showers.png'),
11   Sleet: require('../assets/sleet.png'),
12   Snow: require('../assets/snow.png'),
13   Thunder: require('../assets/thunder.png'),
14 };
15
16 export default weather => images[weather];
17

```

JS App.js > App > render

```
1 import React from 'react';
2 import {
3   StyleSheet,
4   View,
5   ImageBackground,
6   Text,
7   KeyboardAvoidingView,
8   Platform,
9 } from 'react-native';
10
11 import getImageForWeather from '../utils/getImageForWeather';
12
13 import SearchInput from '../components/SearchInput';
```

1

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#34495E',
  },
  imageContainer: {
    flex: 1,
  },
  image: {
    flex: 1,
    width: null,
    height: null,
    resizeMode: 'cover',
  },
  detailsContainer: {
```

3

```
<ImageBackground
  source={getImageForWeather('Clear')}
  style={styles.imageContainer}
  imageStyle={styles.image}
>
  <View style={styles.detailsContainer}>
    <Text style={[styles.largeText, styles.textStyle]}>
      London
    </Text>
    <Text style={[styles.smallText, styles.textStyle]}>
      Light Cloud
    </Text>
    <Text style={[styles.largeText, styles.textStyle]}>
      24°
    </Text>

    <SearchInput placeholder="Search any city" />
  </View>
</ImageBackground>
</KeyboardAvoidingView>
```

2



```

import { StyleSheet, TextInput, View } from 'react-native';
import PropTypes from 'prop-types';

export default class SearchInput extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      text: '',
    };
  }

  handleChangeText = text => {
    this.setState({ text });
  };

  handleSubmitEditing = () => {
    const { onSubmit } = this.props;
    const { text } = this.state;

    if (!text) return;

    onSubmit(text);
    this.setState({ text: '' });
  };

  render() {
    const { placeholder } = this.props;
    const { text } = this.state;

```

```

    const { text } = this.state;

    return (
      <View style={styles.container}>
        <TextInput
          autoComplete={false}
          value={text}
          placeholder={placeholder}
          placeholderTextColor="white"
          underlineColorAndroid="transparent"
          style={styles.textInput}
          clearButtonMode="always"
          onChangeText={this.handleChangeText}
          onSubmitEditing={this.handleSubmitEditing}
        />
      </View>
    );
  }
}

SearchInput.propTypes = {
  onSubmit: PropTypes.func.isRequired,
  placeholder: PropTypes.string,
};

SearchInput.defaultProps = {
  placeholder: '',
};

```

```
import React from 'react';
import {
  StyleSheet,
  View,
  ImageBackground,
  Text,
  KeyboardAvoidingView,
  Platform,
} from 'react-native';

import getImageForWeather from '../utils/getImageForWeather';

import SearchInput from '../components/SearchInput';

export default class App extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      location: 'London',
    };
  }

  handleUpdateLocation = city => {
    this.setState({
      location: city,
    });
  };
}
```



```
render() {
  const { location } = this.state;

  return (
    <KeyboardAvoidingView
      style={styles.container}
      behavior="height"
    >
      <ImageBackground
        source={getImageForWeather('Clear')}
        style={styles.imageContainer}
        imageStyle={styles.image}
      >
        <View style={styles.detailsContainer}>
          <Text style={[styles.largeText, styles.textStyle]}>
            {location}
          </Text>
          <Text style={[styles.smallText, styles.textStyle]}>
            Light Cloud
          </Text>
          <Text style={[styles.largeText, styles.textStyle]}>
            24°
          </Text>

          <SearchInput
            placeholder="Search any city"
            onSubmit={this.handleUpdateLocation}
          />
        </View>
      </ImageBackground>
    </KeyboardAvoidingView>
  );
}
```


PROJECT1A

> .expo

> .expo-shared

> assets

components

JS SearchInput.js

> node_modules

utils

JS api.js

JS getImageForWeather.js

.gitignore

JS App.js

{ } app.json

Babel.config.js

```
1 export const fetchLocationId = async city => {
2   const response = await fetch(
3     `https://www.metaweather.com/api/location/search/?query=${city}`,
4   );
5   const locations = await response.json();
6   return locations[0].woeid;
7 };
8
9 export const fetchWeather = async woeid => {
10   const response = await fetch(
11     `https://www.metaweather.com/api/location/${woeid}/`,
12   );
13   const { title, consolidated_weather } = await response.json();
14   const { weather_state_name, the_temp } = consolidated_weather[0];
15
16   return {
17     location: title,
18     weather: weather_state_name,
19     temperature: the_temp,
20   };
21 };
```

```
import {
  StyleSheet,
  View,
  ImageBackground,
  Text,
  KeyboardAvoidingView,
  Platform,
  ActivityIndicator,
  StatusBar,
} from 'react-native';

import { fetchLocationId, fetchWeather } from '../utils/api';
import getImageForWeather from '../utils/getImageForWeather';

import SearchInput from '../components/SearchInput';

export default class App extends React.Component {
  constructor(props) {
    super(props);

    this.state = {
      loading: false,
      error: false,
      location: '',
      temperature: 0,
      weather: '',
    };
  }
}
```

1

```
componentDidMount() {
  this.handleUpdateLocation('London');
}

handleUpdateLocation = async city => {
  if (!city) return;

  this.setState({ loading: true }, async () => {
    try {
      const locationId = await fetchLocationId(city);
      const { location, weather, temperature } = await fetchWeather(
        locationId,
      );

      this.setState({
        loading: false,
        error: false,
        location,
        weather,
        temperature,
      });
    } catch (e) {
      this.setState({
        loading: false,
        error: true,
      });
    }
  });
}
```

2

```
render() {
  const {
    loading,
    error,
    location,
    weather,
    temperature,
  } = this.state;
```

3

```
return (
  <KeyboardAvoidingView
    style={styles.container}
    behavior="height"
  >
    <StatusBar barStyle="light-content" />
    <ImageBackground
      source={getImageForWeather(weather)}
      style={styles.imageContainer}
      imageStyle={styles.image}
    >
      <View style={styles.detailsContainer}>
        <ActivityIndicator
          animating={loading}
          color="white"
          size="large"
        />

        {!loading && (
```

```
{!loading && (
  <View>
    {error && (
      <Text style={[styles.smallText, styles.textStyle]}>
        Could not load weather, please try a different
        city.
      </Text>
    )}

    {!error && (
      <View>
        <Text
          style={[styles.largeText, styles.textStyle]}
        >
          {location}
        </Text>
        <Text
          style={[styles.smallText, styles.textStyle]}
        >
          {weather}
        </Text>
        <Text
          style={[styles.largeText, styles.textStyle]}
        >
          `${Math.round(temperature)}°`
        </Text>
      </View>
    )}
  )}
```

4

```

        <SearchInput
          placeholder="Search any city"
          onSubmit={this.handleUpdateLocation}
        />
      </View>
    )}
  </View>
</ImageBackground>
</KeyboardAvoidingView>
);
}
}
const styles = StyleSheet.create({
  container: {
    flex: 1,
    height: 100,
    width: 100,
  },
});

```

5

