

**ANKARA ÜNİVERSİTESİ**

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**BLM4538 - IOS İle Mobil Uygulama Geliştirme II**

**Proje Raporu**

**Taş Kağıt Makas Oyunu**

**Refik Can ÖZTAŞ**

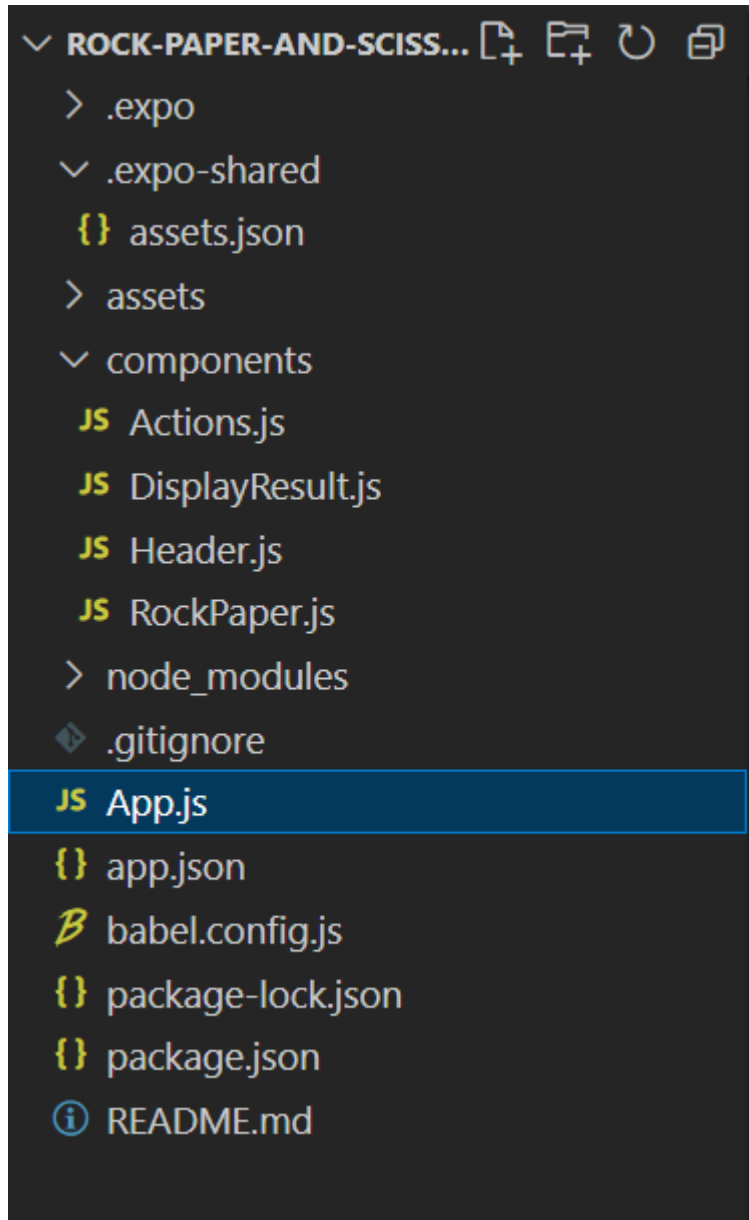
**19290266**

**5/6/2023**

BLM4538 için React Native üzerinde Cross-Platform bir Taş Kağıt Makas oyunu yazılmıştır.

Kodların incelenmesi aşağıdadır, oyun içi görüntüler code review kısmından sonradır.

github linki: <https://github.com/canoztas/RockPaperScissorsGame>



Projede yandaki gibi bir yapı kurulmuştur.


Proje expo üzerinde geliştirilip test edilmiştir. Bu sayede cross-platform olmuştur.

```

eustache on ➤ D:/projects/Rock-Paper-And-Scissors-main
# $env:NODE_OPTIONS="--openssl-legacy-provider"
eustache on ➤ D:/projects/Rock-Paper-And-Scissors-main
# npm start

> rock@1.0.0 start
> cross-env NODE_OPTIONS="--openssl-legacy-provider" && expo start

WARNING: The legacy expo-cli does not support Node +17. Migrate to the new local Expo CLI: https://blog.expo.dev/the-new-expo-cli-f4250d8e3421.
> Port 19000 is
  Use port 19001 instead? ... yes
Starting project at D:\projects\Rock-Paper-And-Scissors-main
Starting Metro Bundler



> Metro waiting on exp://192.168.1.4:19001
> Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

> Press a | open Android
> Press w | open web

> Press r | reload app
> Press m | toggle menu

> Press ? | show all commands

Logs for your project will appear below. Press Ctrl+C to exit.
> Opening on Android...

```

Kodlar:

```

import * as React from 'react';
import { AppBar } from 'react-native-paper';

const Header = () => {

  return (
    <AppBar.Header style={{backgroundColor: '#a60bff'}}>
      <AppBar.Content title="Rock - Paper - Scissors" color="black" style={{alignItems: 'center', transform: [{s
    </AppBar.Header>
  );
};

export default Header;

```

Sayfa headeri

```

1 import React from "react";
2 import { StyleSheet, Text, View } from "react-native";
3 import { FontAwesome5 } from "@expo/vector-icons";
4
5 const ICONS = ["hand-rock", "hand-paper", "hand-scissors"];
6
7 export default function DisplayResult({ userChoice, computerChoice }) {
8   return (
9     <View style={styles.column}>
10       <FontAwesome5
11         name={ICONS[userChoice - 1]}
12         size={64}
13         color="#0000ff"
14         solid
15         style={userChoice === 3 ? styles.scissorsLeftIcon : styles.leftIcon}
16       />
17       <Text style={styles.playerName}>You</Text>
18     </View>
19
20     <View style={styles.column}>
21       <FontAwesome5
22         name={ICONS[computerChoice - 1]}
23         size={64}
24         color="#ff0000"
25         solid
26         style={
27           computerChoice === 3 ? styles.scissorsRightIcon : styles.rightIcon
28         }
29       />
30       <Text style={styles.playerName}>Computer</Text>
31     </View>

```

run Terminal Help DisplayResult.js - Rock-Paper-And-Scissors-main - Visual Studio Code

... JS DisplayResult.js x {} package.json JS Actions.js JS Header.js README.md JS RockPaper.js

components > JS DisplayResult.js > styles > playerName > color

```

35 }
36
37 const styles = StyleSheet.create({
38   column: {
39     flex: 1,
40     justifyContent: "center",
41     alignItems: "center",
42   },
43   playerName: {
44     color: "#000000",
45     fontSize: 16,
46     marginTop: 16,
47   },
48   leftIcon: {
49     transform: [{ rotateZ: "80deg" }],
50   },
51   scissorsLeftIcon: {
52     transform: [{ rotateZ: "180deg" }, { rotateX: "180deg" }],
53   },
54   rightIcon: {
55     transform: [{ rotateZ: "-80deg" }, { rotateY: "180deg" }],
56   },
57   scissorsRightIcon: {
58     transform: [
59       { rotateZ: "180deg" },
60       { rotateY: "180deg" },
61       { rotateX: "180deg" },
62     ],
63   },
64 });
65

```

```
JS DisplayResult.js  {} package.json  JS Actions.js  JS Header.js  README.md  JS RockPaper.js
components > JS Actions.js > Actions
5  const Actions = ({play, canPlay}) => {
6    return (
7      <View style={styles.actions}>
8        <TouchableOpacity // Rock
9          disabled={!canPlay}
10         style={styles.actionButton}
11         onPress={() => play(1)}
12       >
13         <FontAwesome5 name='hand-rock' size={32} color='#a0a0a0' />
14       </TouchableOpacity>
15
16       <TouchableOpacity // paper
17         disabled={!canPlay}
18         style={styles.actionButton}
19         onPress={() => play(2)}
20       >
21         <FontAwesome5 name='hand-paper' size={32} color='#0000ff' />
22       </TouchableOpacity>
23
24       <TouchableOpacity // scissors
25         disabled={!canPlay}
26         style={styles.actionButton}
27         onPress={() => play(3)}
28       >
29         <FontAwesome5
30           name='hand-scissors'
31           size={32}
32           color='#ff0000'
33           style={{ transform: [{rotate: '67deg'}]}}
34         />
35       </TouchableOpacity>
36     )
  }
```

Ekrandaki taş kağıt makas sembollerinin ve scoreun ayarlanması

```
components > JS RockPaper.js > styles > content > backgroundColor
1  import React, { useState, useRef } from 'react';
2  import { StyleSheet, SafeAreaView, Text, View, Animated } from 'react-native';
3  import Constants from 'expo-constants';
4  import Actions from './Actions';
5  import DisplayResult from './DisplayResult';
6  import Header from './Header';
7
8  win=0;
9  lose=0;
10 export default function RockPaper(){
11     const [userChoice, setUserChoice] = useState(0);
12     const [computerChoice, setComputerChoice] = useState(0);
13     const [result, setResult] = useState("");
14     const [scoreboard, setScoreboard] = useState("");
15     const [canPlay, setPlay] = useState(true);
16
17     // For Animation
18     const fadeAnimation = useRef(new Animated.Value(1)).current;
19
20     function play(choice){
21         // We have 3 choice
22         // 1 = rock
23         // 2 = paper
24         // 3 = scissors
25
26         const randomComputerChoice = Math.floor(Math.random() * 3) + 1;
27         let resultString = "";
28
29         if (choice === 1) {
30             resultString = randomComputerChoice === 3 ? "WIN" : "LOSE";
31         }
```

components > RockPaper.js > styles > content > backgroundColor

```
35     else {
36         resultString = randomComputerChoice === 2 ? "WIN" : "LOSE";
37     }
38
39     if (choice === randomComputerChoice){
40         resultString = "DRAW";
41     }
42
43     if (resultString=="WIN"){
44         win++;
45     }
46     else if(resultString=="LOSE"){
47         lose++;
48     }
49
50     setUserChoice(choice);
51     setComputerChoice(randomComputerChoice);
52
53     // Wait animation hide old result
54     setTimeout(() => {
55         setResult(resultString);
56         setScoreboard(win + " win / " + lose + " lose");
57     }, 300);
58
59     // Animation hide old result and show new result
60     Animated.sequence([
61         Animated.timing(fadeAnimation, {
62             toValue:0,
63             duration:300,
64             useNativeDriver: true,
65         }),
66         Animated.timing(fadeAnimation, {
```

```

return(
  <SafeAreaView style={styles.container}>
    <Header/>
    <View style={styles.content}>
      <View style={styles.result}>
        <Animated.Text
          style={[styles.resultText, {opacity: fadeAnimation}]}
        >
          {result}
        </Animated.Text>
        <Animated.Text
          style={[styles.scoreboardText, {opacity: fadeAnimation}]}
        >
          {scoreboard}
        </Animated.Text>
      </View>
      <View style={styles.screen}>
        {!result ? (
          <Text style={styles.readyText}>Let's Play</Text>
        ) : (
          <DisplayResult
            userChoice={userChoice}
            computerChoice={computerChoice}
          />
        )}
      </View>
      <Actions play={play} canPlay={canPlay} />
    </View>
  </SafeAreaView>
);

```



```
12
13 const styles = StyleSheet.create({
14   container: {
15     flex:1,
16     paddingTop: Constants.statusBarHeight,
17   },
18   content: {
19     flex:1,
20     marginBottom:5,
21     backgroundColor: '#006666'
22   },
23   result : {
24     height:100,
25     justifyContent:'flex-end',
26     alignItems: "center",
27   },
28   resultText:{
29     fontSize: 48,
30     fontWeight:"bold",
31   },
32   scoreboard : {
33     height:100,
34     justifyContent:'flex-end',
35     alignItems: "center",
36   },
37   scoreboardText :{
38     fontSize: 24,
39     fontWeight:"normal",
40     color:"#0bf8ff"
```

Asıl oyunun oynandığı, kazananın belirlendiği ve scoreun tutulduğu kısım ve bunların viewi.

Oyun içi görüntüler:

10:14 LTE

# Rock - Paper - Scissors

Let's Play



10:14 LTE

# Rock - Paper - Scissors

## DRAW

0 win / 0 lose



You



Computer





10:14



LTE



# Rock - Paper - Scissors

## LOSE

2 win / 4 lose



You



Computer





10:14



LTE



# Rock - Paper - Scissors

## DRAW

2 win / 7 lose



You



Computer

