

```
import React, { useEffect, useState } from 'react';
import { View, Text, TextInput, Button, StyleSheet, Alert, Image, Share } from 'react-native';
import { NavigationContainer } from '@react-navigation/native';
import { createNativeStackNavigator } from '@react-navigation/native-stack';
import { AdMobInterstitial, setTestDeviceIDAsync } from 'expo-ads-admob';
import * as Font from 'expo-font';
import { StatusBar } from 'expo-status-bar';
```

```
// 配置信息
const CONFIG = {
  APP_NAME: "医生问诊 - 医生问诊",
  DOCTOR_NAME: "医生 (医生)",
  ADMOB: {
    INTERSTITIAL_ID: "ca-app-pub-3940256099942544/1033173712", // 广告 ID
    BANNER_ID: "ca-app-pub-3940256099942544/6300978111", // 广告 ID
    MAX_ADS_PER_HOUR: 3
  }
};
```

```
// AdMob 初始化
let adCount = 0;
let windowStart = Date.now();
```

```
async function initAdMob() {
  try {
    await setTestDeviceIDAsync('EMULATOR');
```

```
    AdMobInterstitial.setAdUnitID(CONFIG.ADMOB.INTERSTITIAL_ID);
  } catch (e) {
    console.log("AdMob error", e);
  }
}
```

```
async function tryShowInterstitial() {
  const now = Date.now();
  if(now - windowStart >= 3600*1000) {
    windowStart = now;
    adCount = 0;
  }
  if(adCount >= CONFIG.ADMOB.MAX_ADS_PER_HOUR) return false;
```

```
  try {
    adCount++;
    await AdMobInterstitial.requestAdAsync({ servePersonalizedAds: false });
    await AdMobInterstitial.showAdAsync();
    return true;
  } catch(e) {
    console.log("Ad show error", e);
    return false;
  }
}
```

```
// 页面生命周期
```

```
function HomeScreen({ navigation }) {
```

```
return (
  <View style={styles.container}>
    <Image source={require('./assets/quack_logo.png')} style={styles.logo} />
    <Text style={styles.title}>{CONFIG.APP_NAME}</Text>
    <Text style={styles.subtitle}>👨‍⚕️: {CONFIG.DOCTOR_NAME}</Text>
    <View style={{ marginTop: 20, width: '60%' }}>
      <Button
        title="👨‍⚕️ 📋 📝"
        onPress={() => navigation.navigate('Symptom')}
        color="#1976D2"
      />
    </View>
  </View>
);
}

// 📝 📋 📝 📝 📝
function SymptomScreen({ navigation }) {
  const [name, setName] = useState("");
  const [age, setAge] = useState("");
  const [symptom, setSymptom] = useState("");

  const quackResponses = {
    fever: {
      medicine: "👨‍⚕️👨‍⚕️👨‍⚕️👨‍⚕️👨‍⚕️ 📋 📝📝📝 (📋 📝📝📝 📝📝)",
      advice: "👨‍⚕️👨‍⚕️👨‍⚕️ 📝📝 📝📝 📝📝📝, 📋 📝📝📝 📝📝 📝📝📝 📝📝📝📝 📝📝📝📝",
    },
  },

```

```
cough: {
  medicine: "□□□□□□□□□□□□ (□□□□□□□□ □□ □□□□□□)",
  advice: "□□□□ □□□□□□□□ □□□□ □□□□, □□□□-□□□□□□ □□ □□□□"
},
default: {
  medicine: "□□□□□□□□ □□□□□□□□□□□□",
  advice: "□□□□□□□□ □□□□□□□□□□□□ □□□□□□"
}
};
```

```
const analyze = async () => {
  if (!name || !age || !symptom) {
    Alert.alert('□□□□□□□□!', '□□□□, □□□□ □ □□□□□□□ □□□□□ □□□□□');
    return;
  }
}
```

```
await tryShowInterstitial();
```

```
let result = quackResponses.default;
if (symptom.includes('□□□□□')) result = quackResponses.fever;
if (symptom.includes('□□□□□')) result = quackResponses.cough;
```

```
navigation.navigate('Prescription', {
  name, age, symptom,
  medicine: result.medicine,
```

```

        advice: result.advice
    });
};

return (
    <View style={styles.container}>
        <Text style={styles.label}>이름을 입력하세요</Text>
        <TextInput value={name} onChangeText={setName} style={styles.input} placeholder="이름을 입력하세요" />

        <Text style={styles.label}>나이</Text>
        <TextInput value={age} onChangeText={setAge} style={styles.input} placeholder="나이를 입력하세요" keyboardType="numeric" />

        <Text style={styles.label}>증상을 입력하세요</Text>
        <TextInput
            value={symptom}
            onChangeText={setSymptom}
            style={[styles.input, { height: 120 }]}
            placeholder="증상을 입력하세요 (예: 두통, 메스꺼움)"
            multiline
        />

        <Button title="진단하기" onPress={analyze} color="#4CAF50" />
    </View>
);
}

// 테스트를 위한 코드

```

```

function PrescriptionScreen({ route }) {
  const { name, age, symptom, medicine, advice } = route.params;

  const prescriptionText = `
    🏠🏠🏠🏠 🏠🏠: ${name}
    🏠🏠: ${age}

    🏠🏠🏠🏠: ${symptom}

    🏠🏠🏠: ${medicine}

    🏠🏠🏠🏠🏠🏠: ${advice}

    🏠🏠🏠🏠: ${new Date().toLocaleDateString('bn-BD')}
    🏠🏠🏠🏠🏠🏠🏠🏠🏠 🏠🏠🏠🏠🏠: ${CONFIG.DOCTOR_NAME}

    🏠 🏠🏠🏠🏠🏠🏠: 🏠🏠 🏠🏠🏠🏠🏠🏠🏠 🏠🏠🏠🏠🏠🏠 🏠🏠🏠🏠🏠🏠। 🏠🏠🏠 🏠🏠🏠🏠🏠 🏠🏠🏠 🏠🏠🏠🏠🏠🏠🏠 🏠🏠🏠🏠🏠 🏠🏠🏠।
    `.trim();

  const onShare = async () => {
    await Share.share({ message: prescriptionText });
  };

  return (
    <View style={styles.container}>
      <Text style={styles.title}>🏠🏠🏠🏠🏠🏠🏠🏠🏠🏠</Text>
      <Text style={styles.prescription}>{prescriptionText}</Text>
    </View>
  );
}

```

```
    <Button title="📄📄📄📄 📄📄📄" onPress={onShare} color="#FF7043" />
  </View>
);
}
```

```
// 📄📄📄📄📄📄 📄📄📄📄
const Stack = createNativeStackNavigator();
```

```
function MainStack() {
  return (
    <Stack.Navigator initialRouteName="Home">
      <Stack.Screen name="Home" component={HomeScreen} options={{ title: '📄📄📄📄📄 📄📄📄📄📄' }} />
      <Stack.Screen name="Symptom" component={SymptomScreen} options={{ title: '📄📄📄📄 📄📄📄📄' }} />
      <Stack.Screen name="Prescription" component={PrescriptionScreen} options={{ title: '📄📄📄📄📄📄📄📄📄' }} />
    </Stack.Navigator>
  );
}
```

```
// 📄📄📄 📄📄📄📄 📄📄📄📄📄📄📄📄
export default function App() {
  const [fontLoaded, setFontLoaded] = useState(false);
```

```
  useEffect(() => {
    initAdMob();
    loadFonts();
```

```
}, []);
```

```
const loadFonts = async () => {  
  await Font.loadAsync({  
    'SolaimanLipi': require('./assets/fonts/SolaimanLipi.ttf'),  
  });  
  setFontLoaded(true);  
};
```

```
if (!fontLoaded) return null;
```

```
return (  
  <NavigationContainer>  
    <MainStack />  
    <StatusBar style="auto" />  
  </NavigationContainer>  
);  
}
```

```
// 스타일시트
```

```
const styles = StyleSheet.create({  
  container: { flex: 1, padding: 20, backgroundColor: '#E3F2FD' },  
  logo: { width: 100, height: 100, marginBottom: 20, alignSelf: 'center' },  
  title: { fontSize: 22, fontWeight: '700', textAlign: 'center', fontFamily: 'SolaimanLipi' },  
  subtitle: { fontSize: 14, color: '#444', marginTop: 6, textAlign: 'center', fontFamily: 'SolaimanLipi' },
```



```
label: { fontSize: 16, marginBottom: 5, fontFamily: 'SolaimanLipi' },
input: {
  borderWidth: 1,
  borderColor: '#90CAF9',
  padding: 12,
  marginBottom: 15,
  borderRadius: 8,
  backgroundColor: 'white',
  fontFamily: 'SolaimanLipi'
},
prescription: {
  marginBottom: 20,
  lineHeight: 24,
  fontFamily: 'SolaimanLipi',
  backgroundColor: 'white',
  padding: 15,
  borderRadius: 8
}
});
```