

April 19 Assignment

Walkthrough.tsx

```
import React from 'react';
import {
  StyleSheet,
  Text,
  View,
  TouchableOpacity,
  Image,
  Platform,
  TouchableHighlight,
  StatusBar,
} from 'react-native';
import {vh, vw} from '../constants/dimensions';
import images from '../constants/images';
import {
  useSafeAreaInsets,
  SafeAreaProvider,
} from 'react-native-safe-area-context';
const CustomStatusBar = ({backgroundColor}: any) => {
  const insets = useSafeAreaInsets();
  return (
    <View
      style={{
        height: insets.top,
        backgroundColor,
      }}>
      <StatusBar backgroundColor={backgroundColor} barStyle={'dark-content'} />
    </View>
  );
};
const Walkthrough = ({navigation}: any) => {
  return (
    <SafeAreaProvider>
      <View style={styles.Largecontainer}>
        <CustomStatusBar backgroundColor="rgba(248,247,247,1)" />
        <View style={styles.container}>
          <Image style={styles.logo} source={images.logo} />
          <Text style={styles.text}>Your Logo</Text>
        </View>
        { /* <View style={styles.border} /> */ }
        <Image style={styles.centerimage} source={images.financeLogo} />
        <Text style={styles.heading}>Banking Reimagined</Text>
        <Text style={styles.text1}>
          Save, Borrow, Spend & Grow your money,{'\n'} without leaving your
          bank!
        </Text>
      </View>
    </SafeAreaProvider>
  );
};
```

```

        </Text>
        <View style={styles.border2} />
        <TouchableHighlight
          style={styles.buttonBox}
          // activeOpacity={1}
          underlayColor="rgba(0, 53, 131,1)"
          onPress={() => {
            navigation.navigate('Phoneno');
          }}>
          <Text style={styles.buttonText}>Get Started</Text>
        </TouchableHighlight>
      </View>
    </SafeAreaView>
  );
};

const styles = StyleSheet.create({
  Largecontainer: {
    flex: 1,
    padding: 5,
  },
  logo: {
    width: vw(20),
    height: vh(20),
  },
  container: {
    flexDirection: 'row',
    alignItems: 'center',
    justifyContent: 'center',
    gap: 5,
    //marginTop: Platform.OS === 'ios' ? 5 : 0,
    width: vw(360),
    height: vh(60),
    borderBottomWidth: 1,
    borderBottomColor: 'rgba(0,0,0,0.1)',
  },
  text: {
    color: 'rgb(0,48,128)',
    fontWeight: 'bold',
  },
  border: {
    margin: 20,
    marginLeft: 0,
    borderBottomColor: '#D9DDE7',
    borderBottomWidth: 1,
    width: '100%',
  },
  border2: {
    margin: 30,
    marginLeft: 0,
    marginTop: 60,
  },
});

```

```

borderBottomColor: '#D9DDE7',
borderBottomWidth: 1,
width: '100%',
},
centerimage: {
  marginTop: 65,
  marginLeft: Platform.OS === 'ios' ? 20 : 25,
  height: vh(300),
  width: vw(300),
},
buttontext: {
  marginTop: Platform.OS === 'ios' ? 2 : 1,
  padding: 10,
  borderRadius: 100,
  // marginLeft: Platform.OS === 'ios' ? 5 : 10,
  height: Platform.OS === 'ios' ? vh(40) : vh(44),
  width: vw(328),
  color: 'white',
  textAlign: 'center',
  fontSize: Platform.OS === 'ios' ? 14 : 13,
},
heading: {
  color: 'rgba(24,25,31,1)',
  marginTop: 60,
  marginHorizontal: 80,
  fontSize: 24,
  fontWeight: '600',
},
text1: {
  marginTop: 15,
  textAlign: 'center',
  fontSize: 13,
  color: Platform.OS === 'ios' ? 'rgba(0,0,0,0.5)' : 'rgba(0,0,0,0.6)',
  lineHeight: 20,

  letterSpacing: Platform.OS === 'ios' ? -0.8 : -0.5,
},
buttonBox: {
  marginLeft: Platform.OS === 'ios' ? 12 : 12,
  height: Platform.OS === 'ios' ? vh(40) : vh(44),
  width: vw(330),
  //borderWidth: 1,
  marginTop: Platform.OS === 'ios' ? -20 : -20,
  borderRadius: 100,
  backgroundColor:
    Platform.OS === 'ios' ? 'rgba(0, 53, 131,1)' : 'rgb(0,53,131)',
},
});
export default Walkthrough;

```

Phoneno.tsx

```
import React, {useState} from 'react';

import {
  StyleSheet,
  Text,
  View,
  TouchableOpacity,
  SafeAreaView,
  Image,
  TextInput,
  Platform,
  KeyboardAvoidingView,
  Keyboard,
  TouchableWithoutFeedback,
} from 'react-native';
import {vh, vw} from '../constants/dimensions';
// import fonts from '../constants/fonts';

import images from '../constants/images';

const Phoneno = ({navigation}: any) => {
  const [mobileNumber, setMobileNumber] = useState('');
  const [buttonColor, setButtonColor] = useState('grey');
  const [isValid, setIsValid] = useState(false);
  const check = /^[6-9]\d{9}$/;
  const mobileNumberChange = (text: any) => {
    const numericValue = text.replace(/[^0-9]/g, '');

    setMobileNumber(numericValue);

    if (text.length === 10 && check.test(text)) {
      setButtonColor('rgb(0,53,131)');
      setIsValid(true);
    } else {
      setButtonColor('grey');
      setIsValid(false);
    }
  };

  return (
    <KeyboardAvoidingView
      style={styles.Largecontainer}
      behavior={Platform.OS === 'ios' ? 'padding' : 'height'}>
      { /* <SafeAreaView style={styles.Largecontainer}> */ }
      <View style={styles.header}>
        <TouchableOpacity
          onPress={() => {
            navigation.goBack();
          }}
        />
      </View>
    </KeyboardAvoidingView>
  );
};
```

```

    }}>
    <Image style={styles.headerButton} source={images.back} />
  </TouchableOpacity>
  <Image style={styles.headerButton} source={images.questionMarks} />
</View>
<TouchableWithoutFeedback onPress={Keyboard.dismiss}>
  <View>
    <View style={styles.mainContent}>
      <View style={styles.parentOfHeading}>
        <Text style={styles.maintextHeading}>
          Hi! What's your mobile{'\n'}number?
        </Text>
        <Text style={styles.maintextsubHeading}>
          We'll use this to verify your account
        </Text>
      </View>
      <View style={styles.contactNo}>
        <Text style={styles.mobileNo}> Mobile Number</Text>
        <View style={styles.inputNumber}>
          <Text style={styles.countryCode}>+91 </Text>

          <TextInput
            style={styles.inputContact}
            keyboardType="numeric"
            onChangeText={mobileNumberChange}
            value={mobileNumber}
            maxLength={10}
            //autoFocus={true}
          />
        </View>

        <Text style={styles.tip}>
          Pro Tip: Use the number linked with your Aadhaar Card
        </Text>
      </View>
    </View>

    {/* </SafeAreaView> */}
  </View>
</TouchableWithoutFeedback>
<View style={styles.buttonBox}>
  <Text style={styles.conditions}>
    By continuing, you agree to the given T&C and Privacy Policy
  </Text>
  <TouchableOpacity
    style={[styles.button, {backgroundColor: buttonColor}]}
    disabled={isValid ? false : true}
    onPress={() => {
      navigation.navigate('Phoneverification');
    }}>
    <Text style={styles.buttontext}>Send OTP</Text>
  </TouchableOpacity>
</View>

```

```

        </TouchableOpacity>
      </View>
    </KeyboardAvoidingView>
  );
};
const styles = StyleSheet.create({
  Largecontainer: {
    flex: 1,
    padding: 5,
  },
  header: {
    marginTop: Platform.OS === 'ios' ? 50 : 0,
    width: vw(360),
    height: vh(60),
    flexDirection: 'row',
    gap: 288,
    borderBottomWidth: 1,
    borderBottomColor: 'rgb(234,234,234)',
  },
  headerButton: {
    width: 20,
    height: 20,
    marginLeft: Platform.OS === 'ios' ? 15 : 12,
    marginTop: 20,
  },
  mainContent: {
    marginTop: 16,
    height: vh(260),
    width: vw(360),
    gap: 16,
    marginLeft: Platform.OS === 'ios' ? 15 : 12,
  },
  parentOfHeading: {
    marginLeft: 5,
    gap: 8,
  },
  maintextHeading: {
    fontWeight: '600',
    fontSize: 24,
    lineHeight: 32,
    color: 'black',
    //fontFamily: fonts.ibm_bold,
  },
  maintextsubHeading: {
    fontWeight: Platform.OS === 'ios' ? '300' : '400',
    fontSize: 12,
    lineHeight: 20,
    color: 'rgba(0,0,0,0.6)',
    letterSpacing: Platform.OS === 'ios' ? -0.3 : 0,
  },
  countryCode: {

```

```
fontWeight: '600',
color: 'black',
marginTop: Platform.OS === 'ios' ? 10 : 8,
marginLeft: 5,
fontSize: 18,
},
inputContact: {
  fontSize: 18,
  fontWeight: '600',
  marginLeft: Platform.OS === 'ios' ? 45 : 40,
  marginTop: Platform.OS === 'ios' ? -21 : -35,
},
contactNo: {
  marginTop: 20,
  marginHorizontal: 5,
  height: vh(152),
  width: vw(360),
},
mobileNo: {
  fontWeight: Platform.OS === 'ios' ? '400' : '500',
  fontSize: 12,
  lineHeight: 20,
  color: 'rgba(0,0,0,0.6)',
  marginBottom: 8,
},
inputNumber: {
  width: Platform.OS === 'ios' ? vw(315) : vw(320),
  height: 44,
  borderRadius: 8,
  borderWidth: 1,
  borderColor: 'rgba(0,0,0,0.1)',
},
tip: {
  marginTop: 8,
  fontWeight: Platform.OS === 'ios' ? '300' : '400',
  fontSize: 12,
  lineHeight: 20,
  color: 'rgba(0, 0, 0, 0.6)',
  letterSpacing: Platform.OS === 'ios' ? 0 : 0.5,
},
buttonBox: {
  height: vh(129),
  width: vw(360),
  marginTop: 480,
  position: 'absolute',
  bottom: Platform.OS === 'ios' ? 5 : 0,
  borderTopWidth: 1,
  borderTopColor: 'rgb(234,234,234)',
},
buttontext: {
  color: 'white',
```

```

paddingVertical: 6,
marginLeft: 150,
marginTop: Platform.OS === 'ios' ? 8 : 6,
borderRadius: 100,
fontSize: Platform.OS === 'ios' ? 14 : 13,
textAlign: 'center',

position: 'absolute',
},
conditions: {
  marginTop: 8,
  fontSize: 10,
  textAlign: 'center',
  color: 'rgba(0,0,0,0.6)',
  fontWeight: Platform.OS === 'ios' ? '400' : '500',
},
button: {
  marginLeft: Platform.OS === 'ios' ? 17 : 17,
  height: Platform.OS === 'ios' ? vh(40) : vh(44),
  width: vw(330),
  // borderWidth: 1,
  //marginTop: Platform.OS === 'ios' ? -10 : -15,
  borderRadius: 100,
  position: 'absolute',
  bottom: Platform.OS === 'ios' ? 65 : 50,
},
});
export default Phoneno;

```

Phoneverification.tsx

```

import React, {useRef, useState} from 'react';

import {
  StyleSheet,
  Text,
  View,
  TouchableOpacity,
  SafeAreaView,
  Image,
  TextInput,
  Platform,
  Keyboard,
  TouchableWithoutFeedback,
  KeyboardAvoidingView,
} from 'react-native';
import {vh, vw} from '../constants/dimensions';

```



```

import images from '../constants/images';
const Phoneverification = ({navigation}: any) => {
  const otpBoxes = Array.from({length: 6}, () => useRef());
  const [enable, setEnable] = useState(false);

  const [buttonColor, setButtonColor] = useState('grey');

  const handleChangeText = (text, index) => {
    text = text.replace(/^[^0-9]/g, '');

    if (text.length === 1 && index < otpBoxes.length - 1) {
      otpBoxes[index + 1].current.focus();
    } else if (text.length === 0 && index > 0) {
      otpBoxes[index - 1].current.focus();
    }
    if (index === otpBoxes.length - 1) {
      setButtonColor('rgb(0,53,131)');
      setEnable(true);
    } else {
      setButtonColor('grey');
      setEnable(false);
    }
  };

  const handleKeyPress = (e, index) => {
    if (e.nativeEvent.key === 'Backspace' && index > 0) {
      otpBoxes[index - 1].current.focus();
    } else if (e.nativeEvent.key === '' && index < otpBoxes.length - 1) {
      otpBoxes[index + 1].current.focus();
    }
  };

  return (
    <KeyboardAvoidingView
      behavior={Platform.OS === 'ios' ? 'padding' : 'height'}
      style={styles.Largecontainer}>
      <View style={styles.header}>
        <TouchableOpacity
          onPress={() => {
            navigation.goBack();
          }}>
          <Image style={styles.headerButton} source={images.back} />
        </TouchableOpacity>
        <Image style={styles.headerButton} source={images.questionMarks} />
      </View>
      <TouchableWithoutFeedback onPress={Keyboard.dismiss}>
        <View>
          <View style={styles.container}>
            <Text style={styles.headerText}>Enter Verification Code</Text>
            <Text style={styles.subtext}>
              We have sent a verification code to{' ' }
            <Text style={styles.number}>+91 7904086979</Text>
          </View>
        </View>
      </TouchableWithoutFeedback>
    </KeyboardAvoidingView>
  );
};

```

```

        </Text>
      </View>
      <View style={styles.OTP}>
        <View style={styles.EnterOTP}>
          {otpBoxes.map((ref, index) => (
            <TextInput
              key={index}
              ref={ref}
              style={styles.inputOtp}
              keyboardType="numeric"
              maxLength={1}
              pattern="[0-9]*"
              onChangeText={text => handleChangeText(text, index)}
              onKeyPress={e => handleKeyPress(e, index)}
            />
          ))}
        </View>
        <Text style={styles.OTPtimer}>Didn't receive the code? 00:29</Text>
      </View>
    </View>
  </TouchableWithoutFeedback>
  <View style={styles.buttonBox}>
    <TouchableOpacity
      style={[styles.button, {backgroundColor: buttonColor}]}
      disabled={enable ? false : true}
      onPress={() => {
        navigation.navigate('Emailid');
      }}
    >
      <Text style={[styles.buttonText]}>Verify OTP</Text>
    </TouchableOpacity>
  </View>
</KeyboardAvoidingView>
);
};

const styles = StyleSheet.create({
  Largecontainer: {
    flex: 1,
    padding: 5,
  },
  header: {
    marginTop: Platform.OS === 'ios' ? 50 : 0,
    width: vw(360),
    height: vh(60),
    flexDirection: 'row',
    gap: 288,
    borderBottomWidth: 1,
    borderBottomColor: 'rgba(234,234,234,1)',
  },
  headerButton: {
    width: 20,
    height: 20,

```

```
    marginLeft: Platform.OS === 'ios' ? 15 : 12,
    marginTop: 20,
  },
  container: {
    marginTop: 16,
    height: vh(60),
    width: vw(350),
    // borderWidth: 1,
    marginLeft: Platform.OS === 'ios' ? 10 : 5,
  },
  headerText: {
    fontWeight: '600',
    fontSize: 24,
    color: 'rgba(24, 25, 31, 1)',
    marginLeft: 12,
  },
  subtext: {
    marginTop: 10,
    fontWeight: Platform.OS === 'ios' ? '300' : '400',
    fontSize: 12,
    marginLeft: 12,
    color: 'rgba(0,0,0,0.6)',
    letterSpacing: Platform.OS === 'ios' ? -0.4 : 0.03,
  },
  number: {
    fontWeight: Platform.OS === 'ios' ? '400' : '500',
    color: 'black',
  },
  OTP: {
    width: vw(350),
    height: vh(104),
    marginLeft: Platform.OS === 'ios' ? 24 : 20,
    marginTop: Platform.OS === 'ios' ? 16 : 26,
    //borderWidth: 1,
  },
  EnterOTP: {
    flexDirection: 'row',
    width: vw(304),
    height: vh(44),
    marginVertical: 10,
    marginHorizontal: Platform.OS === 'ios' ? 6 : 7,
    gap: 8,

    //borderWidth: 1,
  },
  inputOtp: {
    width: Platform.OS === 'ios' ? vw(43) : vw(44),
    height: Platform.OS === 'ios' ? vh(43) : vh(44),
    borderRadius: 8,
    borderWidth: 1,
    borderColor: 'rgba(0,0,0,0.1)',
  },
}
```

```

    fontSize: 18,
    fontWeight: 'bold',
    textAlign: 'center',
  },
  OTPtimer: {
    fontSize: 12,
    fontWeight: Platform.OS === 'ios' ? '300' : '400',
    color: 'rgba(0,0,0,0.6)',
    marginHorizontal: 90,
    letterSpacing: Platform.OS === 'ios' ? 0 : 0.4,
  },
  buttonBox: {
    height: vh(129),
    width: vw(360),
    position: 'absolute',
    bottom: Platform.OS === 'ios' ? 0 : -15,
    borderTopWidth: 1,
    //marginLeft: 18,
    borderTopColor: 'rgba(0,0,0,0.1)',
  },
  buttontext: {
    color: 'white',
    paddingVertical: 12,
    borderRadius: 100,
    fontSize: Platform.OS === 'ios' ? 14 : 13,
    textAlign: 'center',
  },
  button: {
    marginLeft: 18,
    height: Platform.OS === 'ios' ? vh(40) : vh(44),
    width: vw(328),
    //borderWidth: 1,
    marginBottom: Platform.OS === 'ios' ? 16 : 12,
    borderRadius: 100,
    position: 'absolute',
    bottom: Platform.OS === 'ios' ? 70 : 60,
  },
});
export default Phoneverification;

```

Emailid.tsx

```

import React, {useState} from 'react';

import {
  StyleSheet,
  Text,

```

```

View,
TouchableOpacity,
Image,
TextInput,
Keyboard,
TouchableWithoutFeedback,
Platform,
KeyboardAvoidingView,
} from 'react-native';
import {vh, vw} from '../constants/dimensions';
import images from '../constants/images';
const Emailid = ({navigation}: any) => {
  const [email, setEmail] = useState('');
  //const [modalVisible, setModalVisible] = useState(false);
  const [valid, setValid] = useState(false);
  const [buttonColor, setButtonColor] = useState('grey');
  const emailValidation = /^\\w+([\\.-]?\\w+)*@\\w+([\\.-]?\\w+)*(\\.\\w\\w+)+$/;
  const emailChange = text => {
    setEmail(text.toLowerCase());
    if (emailValidation.test(email)) {
      setButtonColor('rgb(0,53,131)');
      setValid(true);
    } else {
      setButtonColor('grey');
      setValid(false);
    }
  };
  return (
    <KeyboardAvoidingView
      style={styles.Largecontainer}
      behavior={Platform.OS === 'ios' ? 'padding' : 'height'}>
      /* <SafeAreaView style={styles.Largecontainer}> */
      <View style={styles.header}>
        <TouchableOpacity
          onPress={() => {
            navigation.goBack();
          }}>
          <Image style={styles.headerButton} source={images.back} />
        </TouchableOpacity>
        <Image style={styles.headerButton} source={images.questionMarks} />
      </View>
      <TouchableWithoutFeedback onPress={Keyboard.dismiss}>
        <View>
          <View style={styles.container}>
            <Text style={styles.headerText}>What's your Email ID?</Text>
            <Text style={styles.subtext}>
              We'll only share important financial information here
            </Text>
          </View>
          <View style={styles.Email}>
            <View style={styles.EnterEmail}>

```

```

        <Text style={styles.emailId}>Email ID</Text>
        <TextInput
          style={styles.inputEmail}
          onChangeText={emailChange}
          value={email}
        />
      </View>
    </View>
  </View>
</TouchableWithoutFeedback>
{ /* <Design9
  isVisible={modalVisible}
  setVisible={setModalVisible}
  navigation={navigation}
/> */ }
<View style={styles.buttonBox}>
  <TouchableOpacity
    style={[styles.button, {backgroundColor: buttonColor}]}
    disabled={valid ? false : true}
    onPress={() => {
      navigation.navigate('Emailverification');
    }}>
    <Text style={styles.buttonText}>Confirm</Text>
  </TouchableOpacity>
  <TouchableOpacity onPress={() => navigation.navigate('Roadmap')}>
    <Text style={styles.SkipContinue}>Skip and Continue</Text>
  </TouchableOpacity>
</View>
</KeyboardAvoidingView>
);
};
const styles = StyleSheet.create({
  Largecontainer: {
    flex: 1,
    padding: 5,
  },
  header: {
    marginTop: Platform.OS === 'ios' ? 50 : 0,
    width: vw(360),
    height: vh(60),
    flexDirection: 'row',
    gap: 288,
    borderBottomWidth: 1,
    borderBottomColor: 'rgba(234,234,234,1)',
  },
  headerButton: {
    width: 20,
    height: 20,
    marginLeft: 15,
    marginTop: 20,
  },

```

```
container: {
  marginTop: 16,
  height: vh(60),
  width: vw(350),
  //borderWidth: 1,
  marginLeft: Platform.OS === 'ios' ? 0 : 0,
},
headerText: {
  fontWeight: '600',
  fontSize: 24,
  color: 'rgba(24, 25, 31, 1)',
  marginLeft: 12,
},
subtext: {
  marginTop: 10,
  fontWeight: Platform.OS === 'ios' ? '300' : '400',
  fontSize: 12,
  marginLeft: 12,
  color: 'rgba(0,0,0,0.6)',
},
buttonBox: {
  height: vh(129),
  width: vw(360),
  position: 'absolute',
  bottom: Platform.OS === 'ios' ? 20 : 0,
  borderTopWidth: 1,
  borderTopColor: 'rgba(0,0,0,0.1)',
},
buttontext: {
  color: 'white',
  paddingVertical: 12,
  height: vh(44),
  width: vw(328),
  borderRadius: 100,
  textAlign: 'center',
},
Email: {
  height: vh(152),
  width: vw(350),
  //borderWidth: 1,
  paddingHorizontal: 11,
  paddingVertical: 16,
  marginLeft: Platform.OS === 'ios' ? 0 : 0,
},
EnterEmail: {
  height: vh(120),
  width: vw(328),
},
emailId: {
  fontSize: 12,
  fontWeight: Platform.OS === 'ios' ? '500' : '500',
```

```

    marginBottom: 7,
    marginLeft: 6,
    color: 'rgba(0,0,0,0.6)',
  },
  inputEmail: {
    width: Platform.OS === 'ios' ? vw(320) : vw(320),
    height: vh(44),
    borderWidth: 1,
    paddingHorizontal: 8,
    paddingVertical: 10,
    borderColor: 'rgba(0,0,0,0.1)',
    borderRadius: 8,
    marginLeft: 6,
  },
  SkipContinue: {
    fontWeight: '500',
    fontSize: 12,
    color: 'rgba(0,60,146,1)',
    textAlign: 'center',
    marginVertical: Platform.OS === 'ios' ? 16 : 15,
    //marginLeft: 10,
  },
  button: {
    marginLeft: 18,
    height: Platform.OS === 'ios' ? vh(40) : vh(44),
    width: vw(328),
    // borderWidth: 1,
    marginTop: 10,
    borderRadius: 100,
    // position: 'absolute',
    // bottom: Platform.OS === 'ios' ? 70 : 60,
  },
});
export default Emailid;

```

Emailverification.tsx

```

import React, {useRef, useState} from 'react';

import {
  StyleSheet,
  Text,
  View,
  TouchableOpacity,
  SafeAreaView,
  TouchableWithoutFeedback,
  Keyboard,
  Image,
  TextInput,

```



```

Platform,
KeyboardAvoidingView,
} from 'react-native';
import {vh, vw} from '../constants/dimensions';
import images from '../constants/images';
const Emailverification = ({navigation}: any) => {
  const otpBoxes = Array.from({length: 6}, () => useRef());
  const [buttonColor, setButtonColor] = useState('grey');
  const [enable, setEnable] = useState(false);
  const handleChangeText = (text, index) => {
    text = text.replace(/[^0-9]/g, '');
    if (text.length === 1 && index < otpBoxes.length - 1) {
      otpBoxes[index + 1].current.focus();
    } else if (text.length === 0 && index > 0) {
      otpBoxes[index - 1].current.focus();
    }
    if (index === otpBoxes.length - 1) {
      setButtonColor('rgb(0,53,131)');
      setEnable(true);
    } else {
      setButtonColor('grey');
      setEnable(false);
    }
  };
  const handleKeyPress = (e, index) => {
    if (e.nativeEvent.key === 'Backspace' && index > 0) {
      otpBoxes[index - 1].current.focus();
    }
  };
  return (
    // <SafeAreaView style={styles.Largecontainer}>
    <KeyboardAvoidingView
      behavior={Platform.OS === 'ios' ? 'padding' : 'height'}
      style={styles.Largecontainer}>
      <View style={styles.header}>
        <TouchableOpacity
          onPress={() => {
            navigation.navigate('Emailid');
          }}>
          <Image style={styles.headerButton} source={images.back} />
        </TouchableOpacity>
        <Image style={styles.headerButton} source={images.questionMarks} />
      </View>
      <TouchableWithoutFeedback onPress={Keyboard.dismiss}>
        <View>
          <View style={styles.container}>
            <Text style={styles.headerText}>We have sent you an OTP</Text>
            <Text style={styles.subtext}>
              Enter 6 digit OTP sent to{' '}
              <Text style={styles.email}>mahesh.s@gmail.com</Text>
            </Text>

```

```

    </View>
    <View style={styles.OTP}>
      <View style={styles.EnterOTP}>
        {otpBoxes.map((ref, index) => (
          <TextInput
            key={index}
            ref={ref}
            style={styles.inputOtp}
            keyboardType="numeric"
            maxLength={1}
            onChangeText={text => handleChangeText(text, index)}
            onKeyPress={e => handleKeyPress(e, index)}
          />
        ))}
      </View>
      <Text style={styles.OTPtimer}>Didn't receive the code? 00:29</Text>
    </View>
  </View>
</TouchableWithoutFeedback>
<View style={styles.buttonBox}>
  <TouchableOpacity
    style={[styles.button, {backgroundColor: buttonColor}]}
    disabled={enable ? false : true}
    onPress={() => {
      navigation.navigate('Pandetails');
    }}>
    <Text style={styles.buttonText}>Verify OTP</Text>
  </TouchableOpacity>
</View>
</KeyboardAvoidingView>
);
};

const styles = StyleSheet.create({
  Largecontainer: {
    flex: 1,
    padding: 5,
  },
  header: {
    marginTop: Platform.OS === 'ios' ? 50 : 0,
    width: vw(360),
    height: vh(60),
    flexDirection: 'row',
    gap: 288,
    borderBottomWidth: 1,
    borderBottomColor: 'rgba(234,234,234,1)',
  },
  headerButton: {
    width: 20,
    height: 20,
    marginLeft: 12,
    marginTop: 20,
  },

```

```

},
container: {
  marginTop: 16,
  height: vh(60),
  width: vw(350),
  marginLeft: Platform.OS === 'ios' ? 5 : 2,
},
headerText: {
  fontWeight: '600',
  fontSize: 24,
  color: 'rgba(24, 25, 31, 1)',
  marginLeft: 12,
},
subtext: {
  marginTop: 10,
  fontWeight: '400',
  fontSize: 12,
  marginLeft: 12,
  color: 'rgba(0,0,0,0.6)',
  letterSpacing: Platform.OS === 'ios' ? -0.3 : -0.1,
},
email: {
  fontWeight: Platform.OS === 'ios' ? '400' : '500',
  color: 'black',
},
OTP: {
  width: vw(350),
  height: vh(104),
  marginLeft: Platform.OS === 'ios' ? 24 : 20,
  marginTop: 16,
},
EnterOTP: {
  flexDirection: 'row',
  width: vw(304),
  height: vh(44),
  marginVertical: 10,
  marginHorizontal: 8,
  gap: 8,
},
inputOtp: {
  width: vw(44),
  height: Platform.OS === 'ios' ? vh(40) : vh(44),
  borderRadius: 8,
  borderWidth: 1,
  borderColor: 'rgba(0,0,0,0.1)',
  fontSize: 18,
  fontWeight: 'bold',
  textAlign: 'center',
},
OTPtimer: {
  fontSize: 12,

```

```

fontWeight: Platform.OS === 'ios' ? '300' : '400',
color: 'rgba(0,0,0,.6)',
marginHorizontal: 90,
letterSpacing: Platform.OS === 'ios' ? 0 : 0.4,
},
buttonBox: {
  height: vh(129),
  width: vw(360),
  position: 'absolute',
  bottom: -20,
  borderTopWidth: 1,
  borderTopColor: 'rgba(0,0,0,0.1)',
},
buttontext: {
  color: 'white',
  paddingVertical: 12,
  paddingHorizontal: 24,
  // marginTop: 10,
  // height: vh(44),
  // width: vw(328),
  // borderRadius: 100,
  // backgroundColor: 'rgb(0,53,131)',
  textAlign: 'center',
  marginHorizontal: 15,
},
button: {
  marginLeft: 18,
  height: Platform.OS === 'ios' ? vh(40) : vh(44),
  width: vw(328),
  // borderWidth: 1,
  marginTop: 10,
  borderRadius: 100,
  // position: 'absolute',
  // bottom: Platform.OS === 'ios' ? 70 : 60,
},
});
export default Emailverification;

```

Output:-













