```javascript
import React, { Component } from 'react';
import {
    StyleSheet,
    View,
    Text,
    Alert
} from 'react-native';
import firebase, { database } from 'firebase';
import { Icon } from 'react-native-elements';
import Geocoder from 'react-native-geocoding';
import { FlatList, ScrollView } from 'react-native-gesture-handler';

import AsyncImage from '../../components/AsyncImage';
import ButtonComponent from '../../components/ButtonComponent';

class ReviewClaim extends Component {

    constructor(props) {
        super(props)

        this.state = {
            key: this.props.navigation.getParam("key"),
            uid: this.props.navigation.getParam('uid'),
            claim_info: null,
            loading: true
        }
    }

    async componentDidMount(){
        console.log(this.state)
        try {
            //EVENT: firebase call
            //fetch claim /users/<uid>/claims/claim_id
            const ref = firebase.database().ref(`/users/${this.state.uid}/claims
/${this.state.key}/`)
            const claim_info = await ref.once('value')
                .then(snapshot => {
                    return snapshot.val()}
                ).catch( error => console.log(error))
            this.setState({claim_info: claim_info, loading: false})
        } catch (error) {
            console.warn("Error fetching data -------------------------- ", error)
        }
    }

    //update firebase db
    //triggered after review decision
    async update(outcome){
        const {claim_info} = this.state
        const ref = firebase.database().ref(`/users/${this.state.uid}/claims
/${this.state.key}/`)

        const day = new Date().getDate()
        const month = new Date().getMonth()
        const year = new Date().getFullYear()
        const date = day + '/' + month + '/' + year

        await ref.set({
            complete: true,
```

```
                outcome: outcome,
                contact_list: claim_info.contact_list,
                date_submitted: claim_info.date_submitted,
                date_reviewed: date,
                description: claim_info.description,
                quote_image_id: claim_info.quote_image_id,
                report_image_id: claim_info.report_image_id
        }).then(() => {
                this.props.navigation.navigate('ClaimsScreen')
        }).catch(e => console.log(e))

    }

    render() {
        const { loading, claim_info } = this.state
        console.log(claim_info)
        if(!loading){
            return(
                <View style={styles.container}>
                    <Text style={styles.logo}>Claim Details</Text>
                    <Text>Submitted on {claim_info.date_submitted}</Text>
                    <Text>Requested amount: €{claim_info.amount}</Text>
                    <ScrollView>
                            <Text style={styles.contentHeader}>Contacts</Text>
                            <FlatList
                                data={claim_info.contact_list}
                                renderItem={({item, index}) =>
                                    <View>
                                        <View style={{paddingLeft: '10%',
borderBottomWidth: 1, borderBottomColor: '#2E6CB5', paddingBottom: 10}}>
                                            <Text style={{fontSize: 18, fontWeight:
'bold'}}>{item.role}</Text>

                                            <Text style={{fontSize: 18}}>Name:
{item.name}</Text>

                                            <Text style={{fontSize: 18}}>Phone Number:
{item.phone}</Text>

                                            <Text style={{fontSize: 18}}>Email:
{item.email}</Text>

                                        </View>
                                    </View>
                                }
                                keyExtractor={(item, index) => index.toString()}

                            />

                            <Text style={styles.contentHeader}>Description</Text>
                            <Text style={styles.content}>{claim_info.description}</Text>

                            <Text style={styles.contentHeader}>Quote</Text>
                            <AsyncImage
                                id={claim_info.quote_image_id}
                                style={{width: '100%', height: 400, resizeMode:
'contain'}}

                                refresh={this.state.refresh}
                                uid={this.state.uid}
                            />

                            <Text style={styles.contentHeader}>Report</Text>
                            <AsyncImage
```

```
                                    id={claim_info.report_image_id}
                                    style={{width: '100%', height: 400, resizeMode:
'contain'}}
                                    refresh={this.state.refresh}
                                    uid={this.state.uid}
                            />
                    </ScrollView>
                    <ButtonComponent
                        icon=''
                        type='antdesign'
                        onPress={() => {
                            Alert.alert(
                                'Approve Claim',
                                'Are you sure you would like to approve this claim?',
                                [
                                {
                                    text: 'Cancel',
                                    style: 'cancel',
                                },
                                {
                                    text: 'Confirm',
                                    onPress: () => this.update('Approved'),
                                },
                                ],
                            );
                        }}
                        text='Approve Claim'
                    />

                    <ButtonComponent
                        icon=''
                        type='antdesign'
                        onPress={() => {
                            Alert.alert(
                                'Reject Claim',
                                'Confirm you would like to reject this claim?',
                                [
                                {
                                    text: 'Cancel',
                                    style: 'cancel',
                                },
                                {
                                    text: 'Confirm',
                                    onPress: () => this.update('Rejected'),
                                },
                                ],
                            );
                        }}
                        text='Reject Claim'
                    />
                    <ButtonComponent
                        icon='arrowleft'
                        type='antdesign'
                        onPress={() => this.props.navigation.goBack()}
                        text='Back'
                    />
                </View>
            )
        } else {
```

```
            return(
                <View style={styles.container}>
                    <View>
                        <Text style={styles.logo}>Loading Claim...</Text>
                    </View>
                </View>
            )
        }

    }


}

const styles = StyleSheet.create({
    container: {
        flex: 1,
        width: '100%'
    },
    logo:{
      fontWeight:"bold",
      fontSize:35,
      color:"#2E6CB5",
      marginBottom:10,
      paddingTop: '10%'
    },
    contentHeader: {
        color: '#2E6CB5',
        fontSize: 28,
        fontWeight: 'bold',
        paddingTop: 20,
        paddingLeft: '5%',
        paddingBottom: 10
    },
    content: {
        fontSize: 20,
        paddingHorizontal: '5%'
    },
})


export default ReviewClaim
```