```
/ Elderly Tech Helper App
// A React Native application to help older adults with technology

// File structure:
// src/
//   - components/
//   - screens/
//   - services/
//   - assets/
//   - navigation/
//   - utils/
// App.js
// package.json
// README.md
// .gitignore

// package.json
{
  "name": "elderly-tech-helper",
  "version": "1.0.0",
  "description": "An app to help older adults with technology",
  "main": "node_modules/expo/AppEntry.js",
  "scripts": {
    "start": "expo start",
    "android": "expo start --android",
    "ios": "expo start --ios",
    "web": "expo start --web",
    "eject": "expo eject",
    "test": "jest"
  },
  "dependencies": {
    "@react-navigation/bottom-tabs": "^6.5.8",
    "@react-navigation/native": "^6.1.7",
    "@react-navigation/stack": "^6.3.17",
    "expo": "^48.0.0",
    "expo-av": "~13.2.1",
    "expo-speech": "~11.1.1",
    "expo-status-bar": "~1.4.4",
    "expo-video-thumbnails": "~7.2.1",
    "firebase": "^10.1.0",
    "react": "18.2.0",
    "react-native": "0.71.8",
    "react-native-gesture-handler": "~2.9.0",
    "react-native-paper": "^5.9.1",
    "react-native-safe-area-context": "4.5.0",
    "react-native-screens": "~3.20.0",
    "react-native-vector-icons": "^9.2.0",
    "react-native-webrtc": "^1.106.1"
  },
  "devDependencies": {
    "@babel/core": "^7.20.0",
    "@testing-library/react-native": "^12.1.2",
    "jest": "^29.2.1",
    "jest-expo": "^48.0.0"
  },
  "private": true
}

// App.js
import React, { useState, useEffect } from 'react';
import { NavigationContainer } from '@react-navigation/native';
import { StatusBar } from 'expo-status-bar';
import { SafeAreaProvider } from 'react-native-safe-area-context';
import { Provider as PaperProvider } from 'react-native-paper';
import { initializeApp } from 'firebase/app';
import { getAuth, onAuthStateChanged } from 'firebase/auth';

import AuthNavigator from './src/navigation/AuthNavigator';
import MainNavigator from './src/navigation/MainNavigator';
import LoadingScreen from './src/screens/LoadingScreen';

// Firebase configuration (replace with your own config)
const firebaseConfig = {
  apiKey: "YOUR_API_KEY",
  authDomain: "YOUR_AUTH_DOMAIN",
  projectId: "YOUR_PROJECT_ID",
  storageBucket: "YOUR_STORAGE_BUCKET",
  messagingSenderId: "YOUR_MESSAGING_SENDER_ID",
  appId: "YOUR_APP_ID"
};

// Initialize Firebase
initializeApp(firebaseConfig);

export default function App() {
  const [user, setUser] = useState(null);
  const [isLoading, setIsLoading] = useState(true);
  const auth = getAuth();

  useEffect(() => {
    const unsubscribe = onAuthStateChanged(auth, (user) => {
      setUser(user);
      setIsLoading(false);
    });
    return unsubscribe;
  }, []);

  if (isLoading) {
    return <LoadingScreen />;
  }

  return (
    <SafeAreaProvider>
      <PaperProvider>
        <NavigationContainer>
          {user ? <MainNavigator /> : <AuthNavigator />}
          <StatusBar style="auto" />
        </NavigationContainer>
      </PaperProvider>
    </SafeAreaProvider>
  );
}

// src/screens/LoadingScreen.js
import React from 'react';
import { View, ActivityIndicator, StyleSheet } from 'react-native';

const LoadingScreen = () => {
  return (
    <View style={styles.container}>
      <ActivityIndicator size="large" color="#0000ff" />
    </View>
  );
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },
});

export default LoadingScreen;

// src/navigation/AuthNavigator.js
import React from 'react';
import { createStackNavigator } from '@react-navigation/stack';
import LoginScreen from '../screens/auth/LoginScreen';
import RegisterScreen from '../screens/auth/RegisterScreen';
import ForgotPasswordScreen from '../screens/auth/ForgotPasswordScreen';

const Stack = createStackNavigator();

const AuthNavigator = () => {
  return (
    <Stack.Navigator initialRouteName="Login" screenOptions={{ headerShown: false }}>
      <Stack.Screen name="Login" component={LoginScreen} />
      <Stack.Screen name="Register" component={RegisterScreen} />
      <Stack.Screen name="ForgotPassword" component={ForgotPasswordScreen} />
    </Stack.Navigator>
  );
};

export default AuthNavigator;

// src/navigation/MainNavigator.js
import React from 'react';
import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';
import { Ionicons } from '@expo/vector-icons';

import HomeNavigator from './HomeNavigator';
import GuidesNavigator from './GuidesNavigator';
import VideoCallNavigator from './VideoCallNavigator';
import VoiceNavigator from './VoiceNavigator';
import SettingsNavigator from './SettingsNavigator';

const Tab = createBottomTabNavigator();

const MainNavigator = () => {
  return (
    <Tab.Navigator
      screenOptions={({ route }) => ({
        tabBarIcon: ({ focused, color, size }) => {
          let iconName;

          if (route.name === 'Home') {
            iconName = focused ? 'home' : 'home-outline';
          } else if (route.name === 'Guides') {
            iconName = focused ? 'book' : 'book-outline';
          } else if (route.name === 'Video Call') {
            iconName = focused ? 'videocam' : 'videocam-outline';
          } else if (route.name === 'Voice Help') {
            iconName = focused ? 'mic' : 'mic-outline';
          } else if (route.name === 'Settings') {
            iconName = focused ? 'settings' : 'settings-outline';
          }

          return <Ionicons name={iconName} size={size} color={color} />;
        },
        tabBarActiveTintColor: '#4f8ef7',
        tabBarInactiveTintColor: 'gray',
        tabBarLabelStyle: {
          fontSize: 12,
        },
        headerShown: false,
      })}
    >
      <Tab.Screen name="Home" component={HomeNavigator} />
      <Tab.Screen name="Guides" component={GuidesNavigator} />
      <Tab.Screen name="Video Call" component={VideoCallNavigator} />
      <Tab.Screen name="Voice Help" component={VoiceNavigator} />
      <Tab.Screen name="Settings" component={SettingsNavigator} />
    </Tab.Navigator>
  );
};

export default MainNavigator;

// src/navigation/HomeNavigator.js
import React from 'react';
import { createStackNavigator } from '@react-navigation/stack';
import HomeScreen from '../screens/HomeScreen';

const Stack = createStackNavigator();

const HomeNavigator = () => {
  return (
    <Stack.Navigator>
      <Stack.Screen name="HomeScreen" component={HomeScreen} options={{
title: 'Tech Helper' }} />
    </Stack.Navigator>
  );
};

export default HomeNavigator;

// src/navigation/GuidesNavigator.js
import React from 'react';
import { createStackNavigator } from '@react-navigation/stack';
import GuidesScreen from '../screens/guides/GuidesScreen';
import GuideDetailScreen from '../screens/guides/GuideDetailScreen';
import GuideStepScreen from '../screens/guides/GuideStepScreen';
import CreateGuideScreen from '../screens/guides/CreateGuideScreen';

const Stack = createStackNavigator();

const GuidesNavigator = () => {
  return (
    <Stack.Navigator>
      <Stack.Screen name="GuidesScreen" component={GuidesScreen}
options={{ title: 'How-to Guides' }} />
      <Stack.Screen name="GuideDetail" component={GuideDetailScreen}
options={({ route }) => ({ title: route.params?.title || 'Guide Details'
})} />
      <Stack.Screen name="GuideStep" component={GuideStepScreen}
options={{ title: 'Step-by-Step' }} />
      <Stack.Screen name="CreateGuide" component={CreateGuideScreen}
options={{ title: 'Create Custom Guide' }} />
    </Stack.Navigator>
  );
};

export default GuidesNavigator;

// src/navigation/VideoCallNavigator.js
import React from 'react';
import { createStackNavigator } from '@react-navigation/stack';
import VideoCallHomeScreen from
'../screens/videocall/VideoCallHomeScreen';
import FindHelperScreen from '../screens/videocall/FindHelperScreen';
import CallScreen from '../screens/videocall/CallScreen';
import HelperProfileScreen from
'../screens/videocall/HelperProfileScreen';

const Stack = createStackNavigator();

const VideoCallNavigator = () => {
  return (
    <Stack.Navigator>
      <Stack.Screen name="VideoCallHome" component={VideoCallHomeScreen}
options={{ title: 'Video Help' }} />
      <Stack.Screen name="FindHelper" component={FindHelperScreen}
options={{ title: 'Find a Helper' }} />
      <Stack.Screen name="Call" component={CallScreen} options={{ title:
'Tech Support Call', headerShown: false }} />
      <Stack.Screen name="HelperProfile" component={HelperProfileScreen}
options={({ route }) => ({ title: route.params?.name || 'Helper Profile'
})} />
```

```
    </Stack.Navigator>
  );
};

export default VideoCallNavigator;

// src/navigation/VoiceNavigator.js
import React from 'react';
import { createStackNavigator } from '@react-navigation/stack';
import VoiceHomeScreen from '../screens/voice/VoiceHomeScreen';
import VoiceTutorialScreen from '../screens/voice/VoiceTutorialScreen';

const Stack = createStackNavigator();

const VoiceNavigator = () => {
  return (
    <Stack.Navigator>
      <Stack.Screen name="VoiceHome" component={VoiceHomeScreen} options={{ title: 'Voice Tutorials' }} />
      <Stack.Screen name="VoiceTutorial" component={VoiceTutorialScreen} options={{ title: 'Voice Guide', headerShown: false }} />
    </Stack.Navigator>
  );
};

export default VoiceNavigator;

// src/navigation/SettingsNavigator.js
import React from 'react';
import { createStackNavigator } from '@react-navigation/stack';
import SettingsScreen from '../screens/settings/SettingsScreen';
import ProfileScreen from '../screens/settings/ProfileScreen';
import AccessibilityScreen from '../screens/settings/AccessibilityScreen';
import NotificationsScreen from '../screens/settings/NotificationsScreen';
import HelpScreen from '../screens/settings/HelpScreen';

const Stack = createStackNavigator();

const SettingsNavigator = () => {
  return (
    <Stack.Navigator>
      <Stack.Screen name="SettingsScreen" component={SettingsScreen} options={{ title: 'Settings' }} />
      <Stack.Screen name="Profile" component={ProfileScreen} options={{ title: 'My Profile' }} />
      <Stack.Screen name="Accessibility" component={AccessibilityScreen} options={{ title: 'Accessibility' }} />
      <Stack.Screen name="Notifications" component={NotificationsScreen}
options={{ title: 'Notifications' }} />
      <Stack.Screen name="Help" component={HelpScreen} options={{ title:
'Help & Support' }} />
    </Stack.Navigator>
  );
};

export default SettingsNavigator;

// src/screens/HomeScreen.js
import React from 'react';
import { View, Text, StyleSheet, ScrollView, TouchableOpacity, Image }
from 'react-native';
import { Card, Title, Paragraph, Button } from 'react-native-paper';
import { Ionicons } from '@expo/vector-icons';
import { SafeAreaView } from 'react-native-safe-area-context';

const HomeScreen = ({ navigation }) => {
  const recentGuides = [
    { id: 1, title: 'How to Send an Email', category: 'Email', icon:
'mail' },
    { id: 2, title: 'Using Video Calls', category: 'Communication', icon:
'videocam' },
    { id: 3, title: 'Smartphone Basics', category: 'Mobile', icon:
'phone-portrait' },
  ];

  const quickHelp = [
    { id: 1, title: 'Connect to WiFi', icon: 'wifi' },
    { id: 2, title: 'Install an App', icon: 'download' },
    { id: 3, title: 'Take a Photo', icon: 'camera' },
    { id: 4, title: 'Make a Call', icon: 'call' },
  ];

  return (
    <SafeAreaView style={styles.container}>
      <ScrollView>
        <View style={styles.header}>
          <Text style={styles.welcomeText}>Welcome back, Susan!</Text>
          <Text style={styles.subtitle}>What would you like help with
today?</Text>
        </View>

        <View style={styles.quickHelpSection}>
          <Text style={styles.sectionTitle}>Quick Help</Text>
          <View style={styles.quickHelpGrid}>
```

```
            {quickHelp.map(item => (
              <TouchableOpacity
                key={item.id}
                style={styles.quickHelpItem}
                onPress={() => navigation.navigate('Guides', {
                  screen: 'GuideDetail',
                  params: { id: item.id, title: item.title }
                })}
              >
                <View style={styles.iconContainer}>
                  <Ionicons name={item.icon} size={24} color="#4f8ef7" />
                </View>
                <Text style={styles.quickHelpText}>{item.title}</Text>
              </TouchableOpacity>
            ))}
          </View>
        </View>

        <View style={styles.section}>
          <Text style={styles.sectionTitle}>Need Immediate Help?</Text>
          <Card style={styles.videoCallCard}>
            <Card.Content>
              <Title>Talk to a Live Helper</Title>
              <Paragraph>Get real-time assistance from our friendly
support team</Paragraph>
            </Card.Content>
            <Card.Actions>
              <Button
                mode="contained"
                onPress={() => navigation.navigate('Video Call')}
                style={styles.videoCallButton}
              >
                Start Video Call
              </Button>
            </Card.Actions>
          </Card>
        </View>

        <View style={styles.section}>
          <View style={styles.sectionTitleContainer}>
            <Text style={styles.sectionTitle}>Recent Guides</Text>
            <TouchableOpacity onPress={() =>
navigation.navigate('Guides')}>
              <Text style={styles.seeAllText}>See All</Text>
            </TouchableOpacity>
          </View>
          {recentGuides.map(guide => (
            <TouchableOpacity
              key={guide.id}
              onPress={() => navigation.navigate('Guides', {
                screen: 'GuideDetail',
                params: { id: guide.id, title: guide.title }
              })}
            >
              <Card style={styles.guideCard}>
                <Card.Content style={styles.guideCardContent}>
                  <View style={styles.guideIconContainer}>
                    <Ionicons name={guide.icon} size={24} color="#4f8ef7"
/>
                  </View>
                  <View style={styles.guideTextContainer}>
                    <Title>{guide.title}</Title>
                    <Paragraph>{guide.category}</Paragraph>
                  </View>
                  <Ionicons name="chevron-forward" size={24} color="#ccc"
/>
                </Card.Content>
              </Card>
            </TouchableOpacity>
          ))}
        </View>

        <View style={styles.section}>
          <Text style={styles.sectionTitle}>Try Voice Commands</Text>
          <Card style={styles.voiceCard}>
            <Card.Content style={styles.voiceCardContent}>
              <View style={styles.voiceTextContainer}>
                <Title>Voice-Controlled Help</Title>
                <Paragraph>Just say what you need help with</Paragraph>
              </View>
              <TouchableOpacity
                style={styles.micButton}
                onPress={() => navigation.navigate('Voice Help')}
              >
                <Ionicons name="mic" size={32} color="white" />
              </TouchableOpacity>
            </Card.Content>
          </Card>
        </View>
      </ScrollView>
    </SafeAreaView>
  );
};
```

```
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#f5f5f5',
  },
  header: {
    padding: 20,
    backgroundColor: '#4f8ef7',
  },
  welcomeText: {
    fontSize: 24,
    fontWeight: 'bold',
    color: 'white',
  },
  subtitle: {
    fontSize: 16,
    color: 'white',
    marginTop: 5,
  },
  section: {
    padding: 20,
  },
  quickHelpSection: {
    padding: 20,
    backgroundColor: 'white',
  },
  sectionTitle: {
    fontSize: 18,
    fontWeight: 'bold',
    marginBottom: 15,
  },
  sectionTitleContainer: {
    flexDirection: 'row',
    justifyContent: 'space-between',
    alignItems: 'center',
    marginBottom: 15,
  },
  seeAllText: {
    color: '#4f8ef7',
    fontWeight: '500',
  },
  quickHelpGrid: {
    flexDirection: 'row',
    flexWrap: 'wrap',
    justifyContent: 'space-between',
  },
  quickHelpItem: {
    width: '48%',
    backgroundColor: '#f0f7ff',
    borderRadius: 10,
    padding: 15,
    marginBottom: 15,
    flexDirection: 'row',
    alignItems: 'center',
  },
  iconContainer: {
    width: 40,
    height: 40,
    borderRadius: 20,
    backgroundColor: '#e6f0ff',
    justifyContent: 'center',
    alignItems: 'center',
    marginRight: 10,
  },
  quickHelpText: {
    fontWeight: '500',
  },
  videoCallCard: {
    borderRadius: 10,
    elevation: 2,
  },
  videoCallButton: {
    backgroundColor: '#4f8ef7',
  },
  guideCard: {
    marginBottom: 10,
    borderRadius: 10,
    elevation: 2,
  },
  guideCardContent: {
    flexDirection: 'row',
    alignItems: 'center',
  },
  guideIconContainer: {
    width: 40,
    height: 40,
    borderRadius: 20,
    backgroundColor: '#e6f0ff',
    justifyContent: 'center',
    alignItems: 'center',
    marginRight: 15,
  },
  guideTextContainer: {
    flex: 1,
  },
  voiceCard: {
    borderRadius: 10,
    elevation: 2,
    backgroundColor: '#f0f7ff',
  },
  voiceCardContent: {
    flexDirection: 'row',
    alignItems: 'center',
  },
  voiceTextContainer: {
    flex: 1,
  },
  micButton: {
    width: 60,
    height: 60,
    borderRadius: 30,
    backgroundColor: '#4f8ef7',
    justifyContent: 'center',
    alignItems: 'center',
  },
});

export default HomeScreen;

// src/screens/guides/GuidesScreen.js
import React, { useState } from 'react';
import { View, Text, StyleSheet, ScrollView, TouchableOpacity, TextInput }
from 'react-native';
import { Card, Chip, Searchbar, FAB } from 'react-native-paper';
import { Ionicons } from '@expo/vector-icons';
import { SafeAreaView } from 'react-native-safe-area-context';

const GuidesScreen = ({ navigation }) => {
  const [searchQuery, setSearchQuery] = useState('');
  const [selectedCategory, setSelectedCategory] = useState(null);

  const categories = [
    { id: 1, name: 'All', icon: 'apps' },
    { id: 2, name: 'Email', icon: 'mail' },
    { id: 3, name: 'Social Media', icon: 'people' },
    { id: 4, name: 'Internet', icon: 'globe' },
    { id: 5, name: 'Smartphone', icon: 'phone-portrait' },
    { id: 6, name: 'Computer', icon: 'desktop' },
  ];

  const guides = [
    { id: 1, title: 'How to Send an Email', category: 'Email', difficulty: 'Easy', steps: 5, icon: 'mail' },
    { id: 2, title: 'Using Facebook', category: 'Social Media', difficulty: 'Medium', steps: 8, icon: 'logo-facebook' },
    { id: 3, title: 'Setting Up Gmail', category: 'Email', difficulty: 'Easy', steps: 4, icon: 'mail' },
    { id: 4, title: 'Browsing the Internet Safely', category: 'Internet', difficulty: 'Medium', steps: 6, icon: 'globe' },
    { id: 5, title: 'Smartphone Basics', category: 'Smartphone', difficulty: 'Easy', steps: 7, icon: 'phone-portrait' },
    { id: 6, title: 'How to Download Apps', category: 'Smartphone', difficulty: 'Easy', steps: 3, icon: 'download' },
    { id: 7, title: 'Using Video Calls', category: 'Computer', difficulty: 'Medium', steps: 5, icon: 'videocam' },
    { id: 8, title: 'Computer File Management', category: 'Computer', difficulty: 'Hard', steps: 10, icon: 'folder' },
  ];

  const filteredGuides = guides.filter(guide => {
    const matchesSearch = guide.title.toLowerCase().includes(searchQuery.toLowerCase());
    const matchesCategory = selectedCategory === null || selectedCategory === 'All' || guide.category === selectedCategory;
    return matchesSearch && matchesCategory;
  });

  return (
    <SafeAreaView style={styles.container}>
      <Searchbar
        placeholder="Search guides..."
        onChangeText={setSearchQuery}
        value={searchQuery}
        style={styles.searchBar}
      />

      <ScrollView horizontal style={styles.categoriesContainer} showsHorizontalScrollIndicator={false}>
        {categories.map(category => (
          <Chip
            key={category.id}
            icon={() => <Ionicons name={category.icon} size={16} color={selectedCategory === category.name ? 'white' : '#4f8ef7'} />}
            selected={selectedCategory === category.name}
            onPress={() => setSelectedCategory(category.name === 'All' ? null : category.name)}
            style={[
              styles.categoryChip,
              selectedCategory === category.name &&
styles.selectedCategoryChip
            ]}
            textStyle={[
              styles.categoryChipText,
              selectedCategory === category.name &&
styles.selectedCategoryChipText
            ]}
          >
            {category.name}
          </Chip>
        ))}
      </ScrollView>

      <ScrollView style={styles.guidesContainer}>
        {filteredGuides.map(guide => (
          <TouchableOpacity
            key={guide.id}
            onPress={() => navigation.navigate('GuideDetail', { id:
guide.id, title: guide.title })}
          >
            <Card style={styles.guideCard}>
              <Card.Content style={styles.guideCardContent}>
                <View style={styles.guideIconContainer}>
                  <Ionicons name={guide.icon} size={24} color="#4f8ef7" />
                </View>
                <View style={styles.guideTextContainer}>
                  <Text style={styles.guideTitle}>{guide.title}</Text>
                  <View style={styles.guideMetaContainer}>
                    <Chip style={styles.metaChip}
textStyle={styles.metaChipText}>{guide.category}</Chip>
                    <Chip style={styles.metaChip}
textStyle={styles.metaChipText}>{guide.difficulty}</Chip>
                    <Chip style={styles.metaChip}
textStyle={styles.metaChipText}>{guide.steps} steps</Chip>
                  </View>
                </View>
                <Ionicons name="chevron-forward" size={24} color="#ccc" />
              </Card.Content>
            </Card>
          </TouchableOpacity>
        ))}
      </ScrollView>

      <FAB
        style={styles.fab}
        icon="plus"
        onPress={() => navigation.navigate('CreateGuide')}
        label="Create Guide"
      />
    </SafeAreaView>
  );
};

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#f5f5f5',
  },
  searchBar: {
    margin: 16,
    borderRadius: 10,
  },
  categoriesContainer: {
    paddingHorizontal: 16,
    marginBottom: 10,
  },
  categoryChip: {
    marginRight: 8,
    backgroundColor: 'white',
    borderColor: '#4f8ef7',
    borderWidth: 1,
  },
  selectedCategoryChip: {
    backgroundColor: '#4f8ef7',
  },
  categoryChipText: {
    color: '#4f8ef7',
  },
  selectedCategoryChipText: {
    color: 'white',
  },
  guidesContainer: {
    padding: 16,
  },
  guideCard: {
    marginBottom: 12,
    borderRadius: 10,
    elevation: 2,
  },
  guideCardContent: {
    flexDirection: 'row',
    alignItems: 'center',
  },
  guideIconContainer: {
    width: 50,
    height: 50,
    borderRadius: 25,
    backgroundColor: '#e6f0ff',
    justifyContent: 'center',
    alignItems: 'center',
    marginRight: 15,
  },
  guideTextContainer: {
    flex: 1,
  },
  guideTitle: {
    fontSize: 16,
    fontWeight: 'bold',
    marginBottom: 5,
  },
  guideMetaContainer: {
    flexDirection: 'row',
    flexWrap: 'wrap',
  },
  metaChip: {
    marginRight: 5,
    marginBottom: 5,
    height: 24,
    backgroundColor: '#f0f0f0',
  },
  metaChipText: {
    fontSize: 10,
  },
  fab: {
    position: 'absolute',
    margin: 16,
    right: 0,
    bottom: 0,
    backgroundColor: '#4f8ef7',
  },
});

export default GuidesScreen;

// src/screens/guides/GuideDetailScreen.js
import React from 'react';
import { View, Text, StyleSheet, ScrollView, Image } from 'react-native';
import { Button, Divider, List, Card } from 'react-native-paper';
import { Ionicons } from '@expo/vector-icons';
import { SafeAreaView } from 'react-native-safe-area-context';
```

```
const GuideDetailScreen = ({ route, navigation }) => {
  const { id, title } = route.params;

  // Mock data for the guide
  const guide = {
    id,
    title,
    category: 'Email',
    difficulty: 'Easy',
    steps: [
      { id: 1, title: 'Open your email app', description: 'Find and tap the email app icon on your device' },
      { id: 2, title: 'Create a new email', description: 'Look for the compose or new message button' },
      { id: 3, title: 'Add a recipient', description: 'Type the email address of the person you want to contact' },
      { id: 4, title: 'Write your message', description: 'Enter a subject and compose your email message' },
      { id: 5, title: 'Send your email', description: 'Tap the send button to deliver your message' },
    ],
    description: 'Learn how to send an email step-by-step, from opening your email app to sending your message.',
    createdBy: 'Tech Helper Team',
    imageUrl: null,
  };

  return (
    <SafeAreaView style={styles.container}>
      <ScrollView>
        <Card style={styles.headerCard}>
          <Card.Content>
            <Text style={styles.title}>{guide.title}</Text>
            <View style={styles.metaRow}>
              <View style={styles.metaItem}>
                <Ionicons name="bookmark" size={16} color="#4f8ef7" />
                <Text style={styles.metaText}>{guide.category}</Text>
              </View>
              <View style={styles.metaItem}>
                <Ionicons name="speedometer" size={16} color="#4f8ef7" />
                <Text style={styles.metaText}>{guide.difficulty}</Text>
              </View>
              <View style={styles.metaItem}>
                <Ionicons name="list" size={16} color="#4f8ef7" />
                <Text style={styles.metaText}>{guide.steps.length} steps</Text>
              </View>
            </View>
            <Text style={styles.description}>{guide.description}</Text>
            {guide.imageUrl && (
              <Image source={{ uri: guide.imageUrl }} style={styles.image} />
            )}
            <Text style={styles.createdBy}>Created by: {guide.createdBy}</Text>
          </Card.Content>
        </Card>

        <View style={styles.section}>
          <Text style={styles.sectionTitle}>Steps Overview</Text>
```