ПРИВЕТ мы с тобой делаем прототип мобильного агрегатора . давай в код который у нас уже есть добавим элементы при нажатии кнопки Вулканы у нас должна открываться новая страница с карточками активностей с тегом вулканы ,   
при нажатии кнопки Термы у нас должна открываться новая страница с карточками активностей с тегом Термальные источники.

И такие карточки Мы создадим для Всех активностей.

и нам необходимо еще создать ети карточки в стиле маркетплейса скажем озон или валбериз  
  
 import React, { useState, useEffect, useRef } from 'react'; import { View, Text, StyleSheet, ImageBackground, ScrollView, TouchableOpacity, Dimensions, Modal, Image, TextInput, Keyboard } from 'react-native'; const { width, height } = Dimensions.get('window'); const HomeScreen = () => { const [activeTab, setActiveTab] = useState('home'); const [modalVisible, setModalVisible] = useState(false); const [selectedImage, setSelectedImage] = useState(null); const [price, setPrice] = useState(''); const [showSearchBlock, setShowSearchBlock] = useState(false); const [keyboardHeight, setKeyboardHeight] = useState(0); const [searchFilters, setSearchFilters] = useState({ type: '', dates: '', people: '', price: '' }); useEffect(() => { const keyboardDidShowListener = Keyboard.addListener( 'keyboardDidShow', (e) => { setKeyboardHeight(e.endCoordinates.height); } ); const keyboardDidHideListener = Keyboard.addListener( 'keyboardDidHide', () => { setKeyboardHeight(0); } ); return () => { keyboardDidShowListener.remove(); keyboardDidHideListener.remove(); }; }, []); const activities = [ { id: '1', icon: '🗻', title: 'Вулканы' }, { id: '2', icon: '♨️', title: 'Термы' }, { id: '3', icon: '🎣', title: 'Рыбалка'}, { id: '4', icon: '🐻', title: 'Природа' }, { id: '5', icon: '🚁', title: 'Вертолёты' }, { id: '6', icon: '⛵', title: 'Морские' }, { id: '7', icon: '🚗', title: 'Джип-Туры' }, { id: '8', icon: '🛶', title: 'Сплавы' } ]; const navItems = [ { id: 'auth', icon: '👤', label: 'Авторизация' }, { id: 'map', icon: '🗺️', label: 'Карта леса' }, { id: 'home', icon: '🏠', label: 'Главная' }, { id: 'favorites', icon: '❤️', label: 'Избранное' }, { id: 'search', icon: '🔍', label: 'Поиск' }, { id: 'menu', icon: '☰', label: 'Меню' }, ]; const partnerPhotos = [ { id: '1', url: 'https://i.pinimg.com/736x/0a/c2/86/0ac28605a4e075019f0ab47123461bf8.jpg', title: 'Рыбалка на Камчатке', source: 'fishingkam.ru' }, { id: '2', url: 'https://i.pinimg.com/originals/8d/e5/ea/8de5ea9c3f38b8601474d54a7379db5b.jpg', title: 'Туры по Камчатке', source: 'mestechkokam.ru' }, { id: '3', url: 'https://optim.tildacdn.com/tild6136-3038-4630-a135-633937346462/-/format/webp/noroot.png', title: 'Северное сияние', source: 'tildacdn.com' }, { id: '4', url: 'https://media-cdn.tripadvisor.com/media/photo-s/10/15/73/91/caption.jpg', title: 'Морская рыбалка', source: 'fishingkam.ru' }, { id: '5', url: 'https://kam-travel.ru/wp-content/uploads/2022/10/na-vertolete-na-kurilskoe-ozero.jpg', title: 'Вулканы Камчатки', source: 'mestechkokam.ru' }, { id: '6', url: 'https://i.pinimg.com/originals/69/95/fb/6995fbb8cfbfe78efd7b1fd851d5fdba.jpg', title: 'Медведи Камчатки', source: 'САЙТ' }, ]; const doublePartnerPhotos = [...partnerPhotos, ...partnerPhotos]; const scrollViewRef = useRef(null); const currentOffset = useRef(0); useEffect(() => { const timer = setInterval(() => { if (scrollViewRef.current) { currentOffset.current += 1; scrollViewRef.current.scrollTo({ x: currentOffset.current, animated: false }); const maxScroll = (width \* 0.65 / 4 + 8) \* partnerPhotos.length; if (currentOffset.current >= maxScroll) { currentOffset.current = 0; scrollViewRef.current.scrollTo({ x: 0, animated: false }); } } }, 30); return () => clearInterval(timer); }, []); const openImage = (image) => { setSelectedImage(image); setModalVisible(true); }; const handleNavPress = (id) => { setActiveTab(id); setShowSearchBlock(id === 'search'); }; const handleSearch = () => { console.log('Search filters:', searchFilters); // Здесь будет логика поиска }; const handleFilterChange = (name, value) => { setSearchFilters(prev => ({ ...prev, [name]: value })); }; return ( <ImageBackground source={{ uri: 'https://cs8.pikabu.ru/post\_img/big/2017/05/12/4/1494568737139342763.jpg' }} style={styles.background} resizeMode="cover" > <View style={styles.overlay} /> <ScrollView contentContainerStyle={styles.container} keyboardShouldPersistTaps="handled" > <View style={styles.header}> <Text style={styles.mainTitle}>Здесь начинается Россия</Text> <Text style={styles.subTitle}>Камчатка - земля огня и льда</Text> </View> {showSearchBlock && ( <View style={styles.tourBlock}> <View style={styles.searchHeader}> <Text style={styles.gridTitle}>Найти тур</Text> <Text style={styles.searchSubtitle}>Открой для себя дикую природу Камчатки</Text> </View> <View style={styles.searchRow}> <TextInput style={styles.searchInput} placeholder="Куда хотите поехать?" placeholderTextColor="#a0aec0" onChangeText={(text) => handleFilterChange('query', text)} returnKeyType="search" onSubmitEditing={handleSearch} /> <TouchableOpacity style={styles.searchButton} onPress={handleSearch} > <Text style={styles.searchButtonText}>Найти</Text> </TouchableOpacity> </View> <View style={styles.filtersGrid}> <View style={styles.filterItem}> <Text style={styles.filterLabel}>Тип тура</Text> <TextInput style={styles.filterInput} placeholder="Поход, экскурсия, рыбалка..." placeholderTextColor="#a0aec0" value={searchFilters.type} onChangeText={(text) => handleFilterChange('type', text)} /> </View> <View style={styles.filterItem}> <Text style={styles.filterLabel}>Даты</Text> <TextInput style={styles.filterInput} placeholder="Выберите даты" placeholderTextColor="#a0aec0" value={searchFilters.dates} onChangeText={(text) => handleFilterChange('dates', text)} /> </View> <View style={styles.filterItem}> <Text style={styles.filterLabel}>Количество людей</Text> <TextInput style={styles.filterInput} placeholder="Сколько человек?" placeholderTextColor="#a0aec0" value={searchFilters.people} onChangeText={(text) => handleFilterChange('people', text)} keyboardType="numeric" /> </View> <View style={styles.filterItem}> <Text style={styles.filterLabel}>Бюджет</Text> <TextInput style={styles.filterInput} placeholder="Максимальная сумма" placeholderTextColor="#a0aec0" value={searchFilters.price} onChangeText={(text) => handleFilterChange('price', text)} keyboardType="numeric" /> </View> </View> <TouchableOpacity style={styles.applyButton} onPress={handleSearch} > <Text style={styles.applyButtonText}>Найти подходящий тур</Text> </TouchableOpacity> </View> )} {/\* Активности Камчатки - скрываем только при поиске \*/} {!showSearchBlock && ( <View style={styles.activitiesContainer}> <Text style={styles.sectionTitle}>Активности Камчатки</Text> <View style={styles.activitiesGrid}> {activities.map((activity) => ( <TouchableOpacity key={activity.id} style={[ styles.activityCard, activeTab === activity.id && styles.activeActivityCard ]} onPress={() => setActiveTab(activity.id)} > <Text style={[styles.activityIcon, styles.whiteIcon]}>{activity.icon}</Text> <Text style={styles.activityTitle}>{activity.title}</Text> </TouchableOpacity> ))} </View> </View> )} {/\* Камчатка глазами путешественников - показываем всегда \*/} <View style={styles.carouselContainer}> <Text style={styles.sectionTitle}>Камчатка глазами путешественников</Text> <View style={styles.carouselWrapper}> <ScrollView ref={scrollViewRef} horizontal showsHorizontalScrollIndicator={false} contentContainerStyle={styles.carousel} scrollEventThrottle={16} > {doublePartnerPhotos.map((photo, index) => ( <TouchableOpacity key={`${photo.id}-${index}`} onPress={() => openImage(photo)} activeOpacity={0.8} > <Image source={{ uri: photo.url }} style={styles.carouselItem} resizeMode="cover" /> </TouchableOpacity> ))} </ScrollView> </View> </View> </ScrollView> <View style={[styles.navBar, { bottom: keyboardHeight > 0 ? keyboardHeight + 10 : 70 }]}> {navItems.map((item) => ( <TouchableOpacity key={item.id} style={[ styles.navButton, activeTab === item.id && styles.activeNavButton ]} onPress={() => handleNavPress(item.id)} activeOpacity={0.7} > <Text style={[ styles.navIcon, activeTab === item.id ? styles.activeNavIcon : styles.inactiveNavIcon ]} > {item.icon} </Text> </TouchableOpacity> ))} </View> <Modal animationType="fade" transparent={true} visible={modalVisible} onRequestClose={() => setModalVisible(false)} > <View style={styles.modalContainer}> {selectedImage && ( <> <Image source={{ uri: selectedImage.url }} style={styles.fullImage} resizeMode="contain" /> <View style={styles.imageInfo}> <Text style={styles.modalTitle}>{selectedImage.title}</Text> <Text style={styles.modalSource}>Источник: {selectedImage.source}</Text> </View> <TouchableOpacity style={styles.closeButton} onPress={() => setModalVisible(false)} > <Text style={styles.closeButtonText}>✕</Text> </TouchableOpacity> </> )} </View> </Modal> </ImageBackground> ); }; const styles = StyleSheet.create({ background: { flex: 1, width: '100%', height: '100%', }, overlay: { ...StyleSheet.absoluteFillObject, backgroundColor: 'rgba(0, 0, 0, 0.3)', }, container: { flexGrow: 1, padding: 15, paddingBottom: 100, }, header: { marginTop: height \* 0.04, marginBottom: 20, alignItems: 'center', }, mainTitle: { fontSize: 28, fontWeight: 'bold', color: '#FFF', textAlign: 'center', textShadowColor: 'rgba(0, 0, 0, 0.7)', textShadowOffset: { width: 1, height: 1 }, textShadowRadius: 5, marginBottom: 8, }, subTitle: { fontSize: 16, color: '#FFF', textAlign: 'center', textShadowColor: 'rgba(0, 0, 0, 0.5)', textShadowOffset: { width: 1, height: 1 }, textShadowRadius: 3, }, activitiesContainer: { marginVertical: 15, }, sectionTitle: { fontSize: 20, fontWeight: '600', color: '#FFF', marginBottom: 15, textAlign: 'center', textShadowColor: 'rgba(0, 0, 0, 0.5)', textShadowOffset: { width: 1, height: 1 }, textShadowRadius: 3, }, activitiesGrid: { flexDirection: 'row', flexWrap: 'wrap', justifyContent: 'space-between', paddingHorizontal: 5, }, activityCard: { width: '24%', alignItems: 'center', marginBottom: 10, padding: 4, backgroundColor: 'rgba(255, 255, 255, 0.15)', borderRadius: 10, borderWidth: 1, borderColor: 'rgba(255, 255, 255, 0.2)', }, activeActivityCard: { backgroundColor: 'rgba(255, 165, 0, 0.4)', borderColor: 'rgba(255, 165, 0, 0.7)', }, activityIcon: { fontSize: 20, marginBottom: 2, }, whiteIcon: { color: '#FFF', textShadowColor: 'rgba(0, 0, 0, 0.5)', textShadowOffset: { width: 1, height: 1 }, textShadowRadius: 2, }, activityTitle: { fontSize: 10, fontWeight: '500', color: '#FFF', textAlign: 'center', textShadowColor: 'rgba(0, 0, 0, 0.5)', textShadowOffset: { width: 1, height: 1 }, textShadowRadius: 2, }, carouselContainer: { marginTop: 15, }, carouselWrapper: { height: 50, }, carousel: { flexDirection: 'row', alignItems: 'center', }, carouselItem: { width: width \* 0.65 / 4, height: 180 / 4, marginRight: 8, borderRadius: 4, backgroundColor: '#ccc', }, navBar: { position: 'absolute', left: '5%', right: '5%', backgroundColor: 'rgba(120, 120, 120, 0.4)', flexDirection: 'row', justifyContent: 'space-around', alignItems: 'center', borderRadius: 22, padding: 6, borderWidth: 1, borderColor: 'rgba(255, 255, 255, 0.2)', shadowColor: '#000', shadowOffset: { width: 0, height: 2 }, shadowOpacity: 0.2, shadowRadius: 4, elevation: 3, height: 40, }, navButton: { width: 32, height: 32, borderRadius: 16, justifyContent: 'center', alignItems: 'center', backgroundColor: 'transparent', }, activeNavButton: { backgroundColor: 'rgba(255, 255, 255, 0.9)', }, navIcon: { fontSize: 18, textAlign: 'center', }, activeNavIcon: { color: '#000', }, inactiveNavIcon: { color: 'rgba(255, 255, 255, 0.8)', }, // КАМЧАТСКИЙ СТИЛЬ ДЛЯ ПОИСКА ТУРА tourBlock: { backgroundColor: 'rgba(26, 32, 44, 0.85)', borderRadius: 16, padding: 20, marginVertical: 15, borderWidth: 2, borderColor: '#e2b93b', shadowColor: '#000', shadowOffset: { width: 0, height: 4 }, shadowOpacity: 0.3, shadowRadius: 8, elevation: 8, }, searchHeader: { marginBottom: 20, alignItems: 'center', }, gridTitle: { fontSize: 24, fontWeight: 'bold', color: '#e2b93b', marginBottom: 8, textAlign: 'center', textShadowColor: 'rgba(0, 0, 0, 0.5)', textShadowOffset: { width: 1, height: 1 }, textShadowRadius: 2, }, searchSubtitle: { fontSize: 14, color: '#a0aec0', textAlign: 'center', }, searchRow: { flexDirection: 'row', marginBottom: 20, }, searchInput: { flex: 1, height: 50, backgroundColor: 'rgba(255, 255, 255, 0.1)', borderRadius: 12, borderWidth: 1, borderColor: '#4a5568', paddingHorizontal: 20, color: '#fff', fontSize: 16, fontWeight: '500', }, searchButton: { width: 100, height: 50, backgroundColor: '#e2b93b', borderRadius: 12, marginLeft: 10, justifyContent: 'center', alignItems: 'center', shadowColor: '#e2b93b', shadowOffset: { width: 0, height: 2 }, shadowOpacity: 0.4, shadowRadius: 4, elevation: 4, }, searchButtonText: { color: '#1a202c', fontSize: 16, fontWeight: 'bold', }, filtersGrid: { flexDirection: 'row', flexWrap: 'wrap', justifyContent: 'space-between', marginBottom: 20, }, filterItem: { width: '48%', marginBottom: 15, }, filterLabel: { color: '#e2b93b', fontSize: 14, fontWeight: '600', marginBottom: 8, }, filterInput: { height: 50, backgroundColor: 'rgba(255, 255, 255, 0.1)', borderRadius: 12, borderWidth: 1, borderColor: '#4a5568', paddingHorizontal: 15, color: '#fff', fontSize: 16, }, applyButton: { height: 50, backgroundColor: '#3182ce', borderRadius: 12, justifyContent: 'center', alignItems: 'center', shadowColor: '#3182ce', shadowOffset: { width: 0, height: 2 }, shadowOpacity: 0.4, shadowRadius: 4, elevation: 4, }, applyButtonText: { color: '#fff', fontSize: 16, fontWeight: 'bold', }, modalContainer: { flex: 1, backgroundColor: 'rgba(0, 0, 0, 0.9)', justifyContent: 'center', alignItems: 'center', }, fullImage: { width: width, height: height \* 0.7, }, imageInfo: { position: 'absolute', bottom: 40, left: 20, right: 20, backgroundColor: 'rgba(0, 0, 0, 0.6)', padding: 15, borderRadius: 10, }, modalTitle: { color: '#FFF', fontSize: 20, fontWeight: 'bold', marginBottom: 5, }, modalSource: { color: 'rgba(255, 255, 255, 0.7)', fontSize: 16, }, closeButton: { position: 'absolute', top: 40, right: 20, backgroundColor: 'rgba(0, 0, 0, 0.5)', width: 40, height: 40, borderRadius: 20, justifyContent: 'center', alignItems: 'center', borderWidth: 1, borderColor: 'rgba(255, 255, 255, 0.3)', }, closeButtonText: { color: '#FFF', fontSize: 24, fontWeight: 'bold', }, }); export default HomeScreen; Напиши исправленный код полностью