

Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Разработка программного обеспечения для мобильных платформ

Лабораторная работа №2

ТЕМА
«Платформа для поиска удаленной работы»

Студент: гр. 351001 Ушаков А. Д.

Проверила: Петровская В. В.

Минск 2026

1 Исправление бага из ЛР1

Проблема: при вводе текста в любое поле (add-job.tsx, edit-job/[id].tsx) клавиатура закрывалась после каждого нажатия.

Причина: компонент InputField был объявлен внутри тела экранного компонента.

```
// перемонтируется при каждом setState
export default function AddJobScreen() {
  const InputField = ({ ... }) => <View>...</View>; // ← здесь
  ...
}
```

React считает такую функцию новым компонентом при каждом рендеринге → TextInput размонтируется → клавиатура закрывается.

Решение: вынесен в отдельный файл components/InputField.tsx с React.memo.

```
// стабильная ссылка, нет лишних ремонтов
export const InputField = memo(function InputField({ ... }) {
  return <TextInput ... />;
});
```

2 Архитектура MVVM

Реализован паттерн MVVM (Model-View-ViewModel) через кастомные React Hooks.

Слой	Роль	Файлы
Model	Данные + сервисы	types/job.ts, services/jobService.ts
ViewModel	Бизнес-логика	viewmodels/JobsViewModel.ts, viewmodels/SavedJobsViewModel.ts
View	Только UI	app/(tabs)/index.tsx, app/(tabs)/saved.tsx

ViewModels:

useJobsViewModel() – управляет локальными и API вакансиями, поиском, фильтрацией, переключением вкладок, pull-to-refresh

useSavedJobsViewModel() – управляет сохранёнными вакансиями, перезагрузкой при фокусе экрана

Результат: Экраны стали «глупыми» view-компонентами — только рендеринг, никакой бизнес-логики.

3 Веб-API: Remotive

Подключён Remotive API – бесплатный публичный API вакансий удалённой работы, не требует ключа (но в моей локации доступен только через VPN).

Особенности:

Новый файл: services/remoteJobApiService.ts.

Endpoint: <https://remotive.com/api/remote-jobs?category=software-dev&limit=20>.

Маппинг API-ответа в внутренний тип Job

Автоматическое удаление HTML-тегов из описаний.

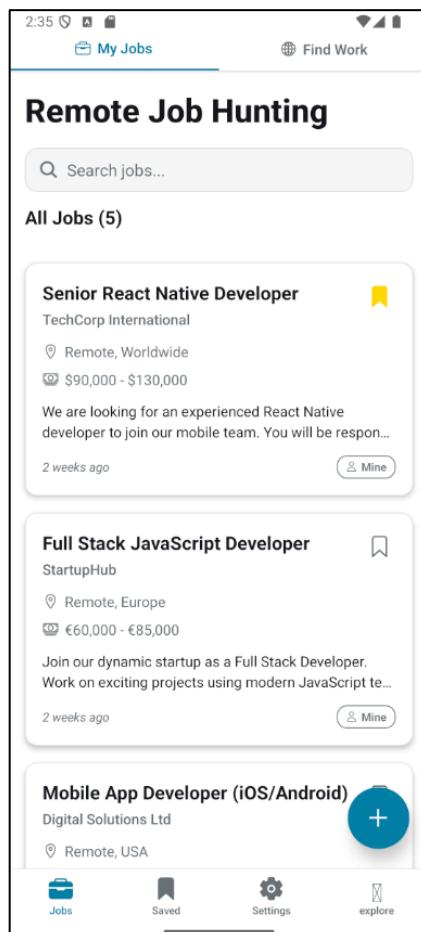
UI интеграция:

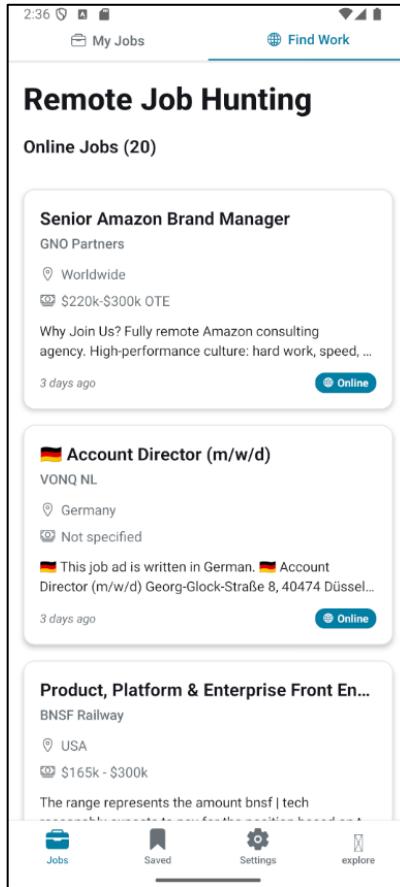
На главном экране появились две вкладки:

1 Мои вакансии – локальные из AsyncStorage (добавление, редактирование, удаление, избранное).

2 Найти работу – вакансии из Remotive API (только просмотр и ссылка «Откликнуться»).

Каждая карточка показывает бейдж [Online] или [Моя] для различия источника данных.





4 Мониторинг интернет-соединения

Установлен пакет `@react-native-community/netinfo`.

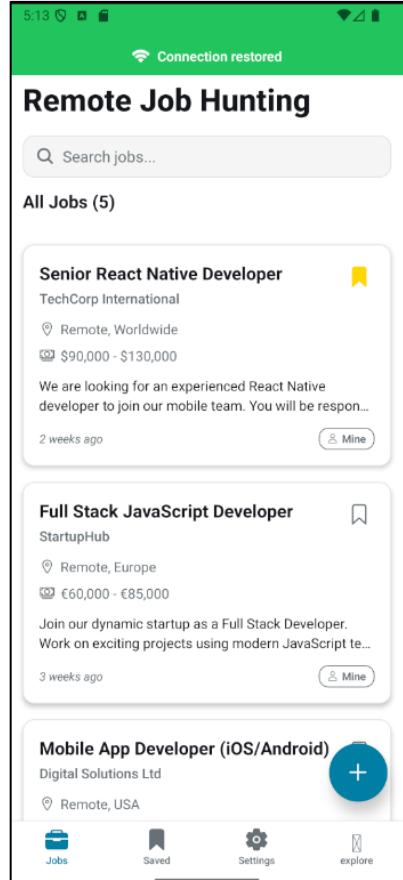
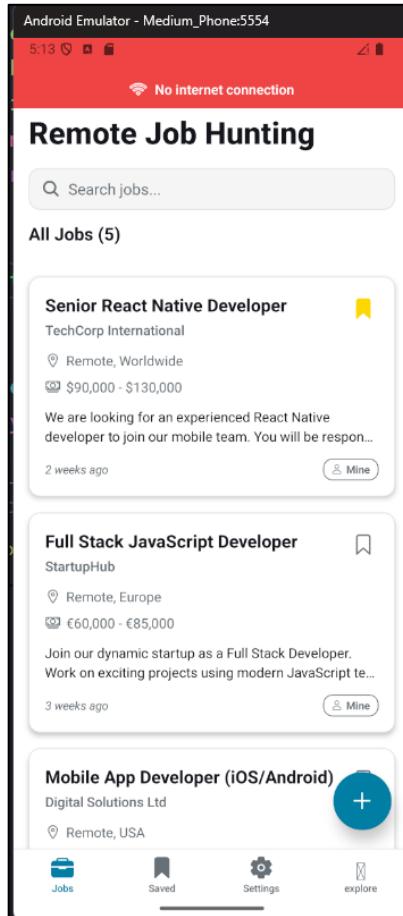
Хук: hooks/useNetworkStatus.ts

```
const { isConnected, isInternetReachable, connectionType } =  
useNetworkStatus();
```

Подписывается на изменения сети через `NetInfo.addEventListener`.

Компонент: components/NetworkBanner.tsx

Появляется при потере соединения: красный баннер «Нет подключения к интернету». При восстановлении: зелёный баннер «Соединение восстановлено». Автоматическое скрытие: Баннер (и красный, и зелёный) скрывается через 4 секунды после появления. Анимация: slide + fade (через Animated API). Расположение: корневой `app/_layout.tsx` – виден на всех экранах.



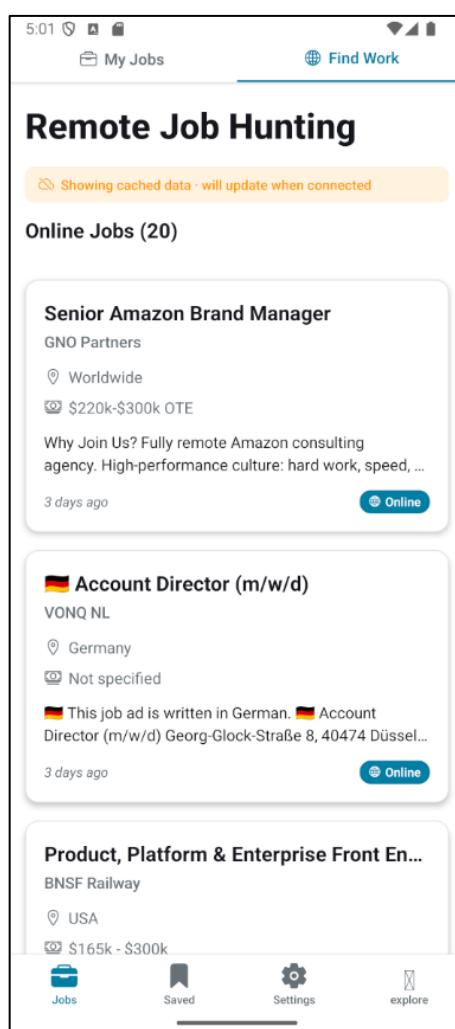
5 Оффлайн режим и кэширование

Кэш API в AsyncStorage:

При успешном запросе к Remotive API результат сохраняется в AsyncStorage с таймстемпом. TTL кэша: 30 минут (свежий кэш используется повторно без запроса к сети). При отсутствии сети: загружаются кэшированные данные (любой давности). Показывается уведомление «Данные из кэша обновятся при подключении»

Offline индикация:

При отсутствии сети на иконке вкладки «Найти работу» появляется красная точка. При восстановлении сети API-вакансии автоматически перезагружаются.



6 Пример вакансии из Remotive API