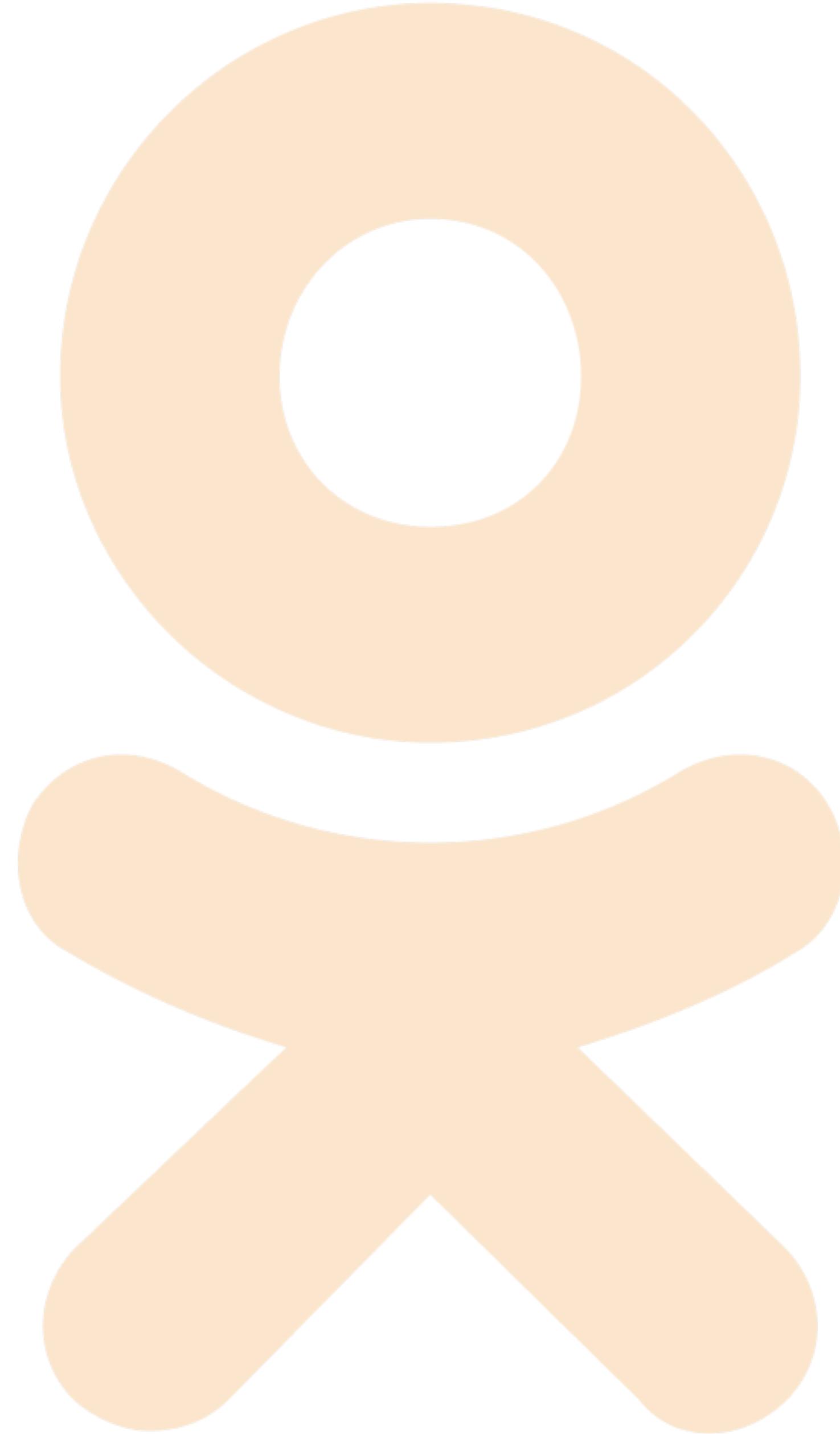


Что может iOS-симулятор

Антон Смолянин

Описание возможностей и ограничений iOS-симулятора



Смолянин Антон

Старший инженер по автоматизации тестирования в OK.RU

Настройка iOS фермы и оптимизация тестов

7 лет в Android и 3 года в iOS



xCode



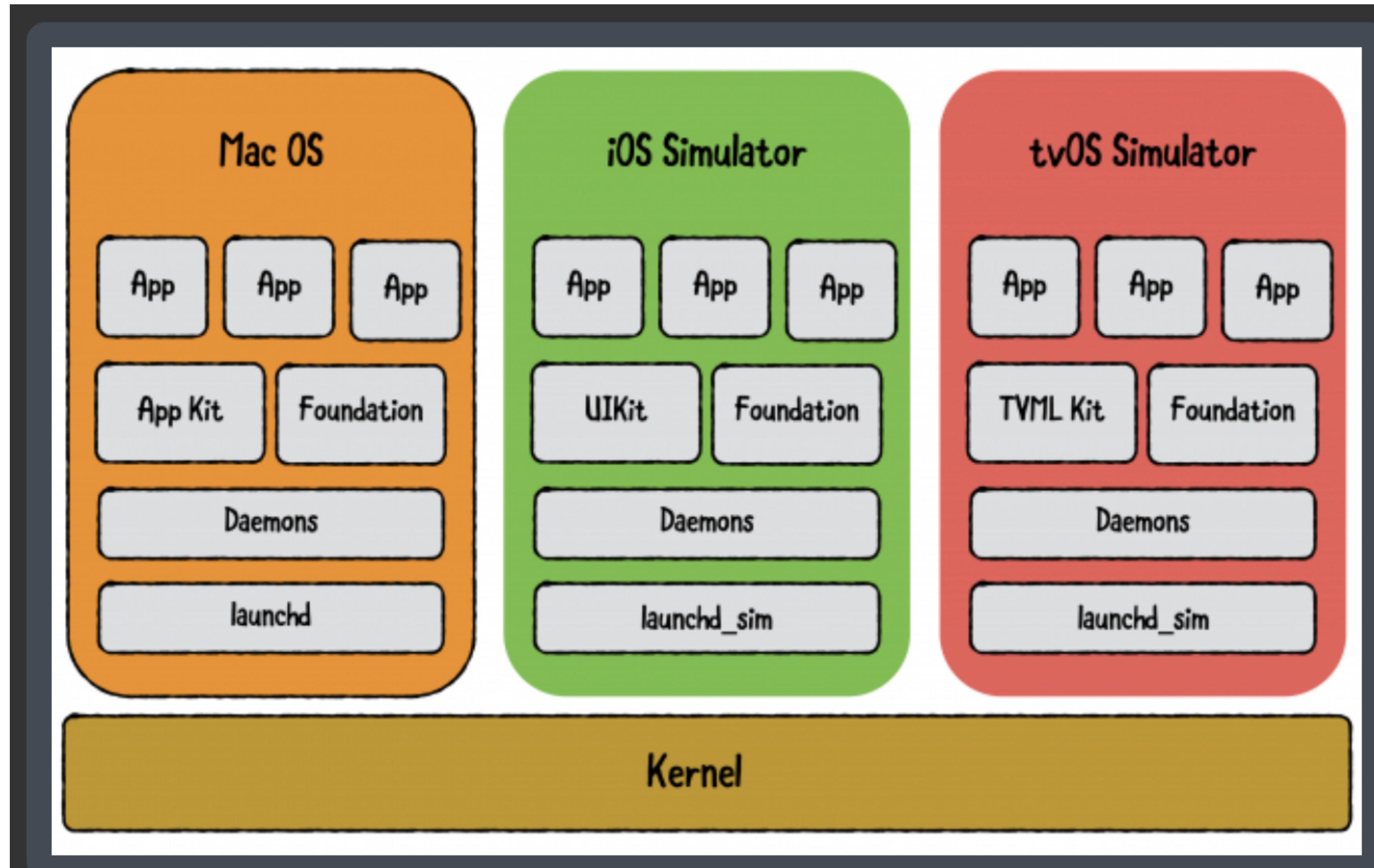
В докладе вы услышите

- Разница между симулятором и эмулятором
- Возможности iOS-симулятора
- CLI iOS-симулятора
- Ограничения iOS-симулятора
- Дополнительные инструменты и программы

Эмулятор и Симулятор



Симулятор	Эмулятор
Отдельный userspace	Отдельный экземпляр операционной системы внутри QEMU
Использует хранилище Host машины	Отдельная файловая система
Использует сетевой канал Host машины	Виртуальная сетевая карта и отдельная подсеть
Нет ограничений по памяти и процессору	Можно выставить ограничения по ядрам и количеству памяти



Симулятор

Отдельный userspace

Использует хранилище Host машины

Использует сетевой канал Host машины

Нет ограничений по памяти и процессору

Эмулятор

Операционная система внутри QEMU

Отдельная файловая система

Виртуальная сетевая карта и отдельная подсеть

Можно выставить ограничения по ядрам и количеству памяти

Симулятор	Эмулятор
Отдельный userspace	Операционная система внутри QEMU
Использует хранилище Host машины	Отдельная файловая система
Использует сетевой канал Host машины	Виртуальная сетевая карта и отдельная подсеть
Нет ограничений по памяти и процессору	Можно выставить ограничения по ядрам и количеству памяти

Симулятор

Отдельный userspace

Использует хранилище Host машины

Использует сетевой канал Host машины

Нет ограничений по памяти и процессору

Эмулятор

Операционная система внутри QEMU

Отдельная файловая система

Виртуальная сетевая карта и отдельная подсеть

Можно выставить ограничения по ядрам и количеству памяти

Возможности iOS-симулятора





Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

Работа с Геолокацией

Управление Accessibility

Переключение Локализации

Работа с уведомлениями (Notifications)

Поддержка Metal

Несколько симуляторов





Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

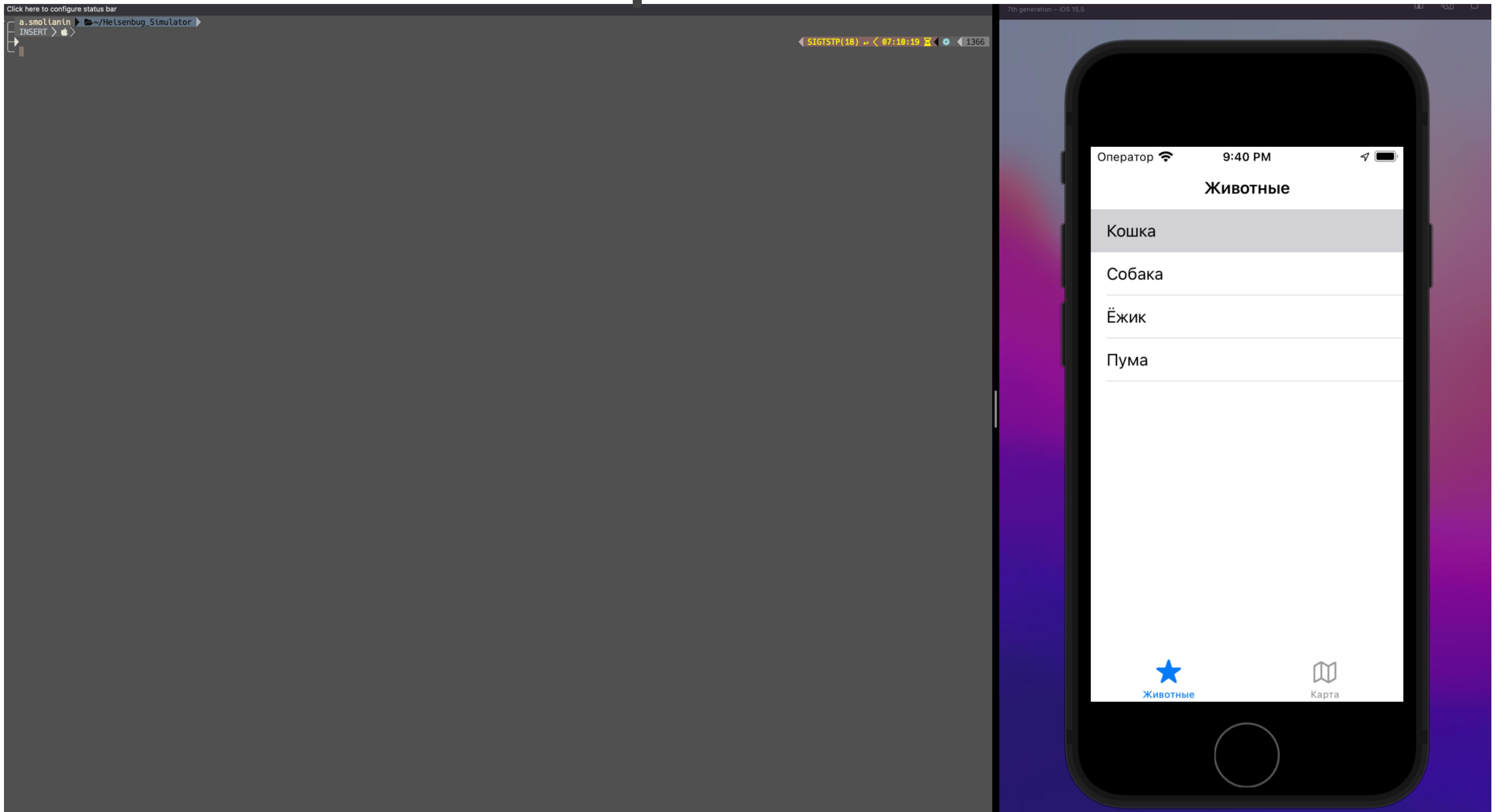
Работа с Геолокацией

Управление Accessibility

Переключение Локализации

Поддержка Metal

Как обычное приложение



Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

Работа с Геолокацией

Управление Accessibility

Переключение Локализации

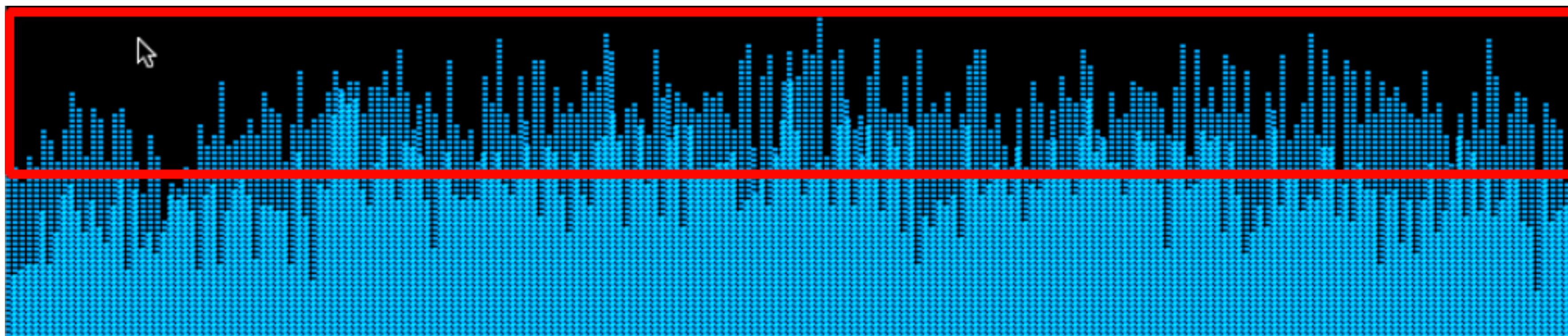
Поддержка Metal

Headless режим



Потребление ресурсов

- Headless режим меньше нагружает GPU
- Замеры на 16 ГБ M1 Mac Mini, 6 симуляторов





Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в `headless` режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

Работа с Геолокацией

Управление Accessibility

Переключение Локализации

Поддержка Metal

Параллельный запуск

- До Xcode 10 разные destinations
- До Xcode 9 через сторонние программы (pxctest)
- Клон симулятора на каждое ядро процессора
- Параллелизация по тестовым классам

Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

Работа с Геолокацией

Управление Accessibility

Переключение Локализации

Поддержка Metal

Что такое StoreKit

- Framework
- Поддержка покупок внутри приложения
- С Xcode 14 можно синхронизироваться с App Store Connect

StoreKit в симуляторе

- В Xcode 12 с iOS 14 можно тестировать без App Store Connect
- Внутренний дебаггер транзакций
- С Xcode 14 можно синхронизироваться с App Store Connect

HEISENBUG

2019 Piter

**Виктор Короневич
Владимир Солодов**

Badoo

Тестируем платные сервисы:
как перестать запускать
самолет, чтобы проверить
лампочку на приборной
панели





Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

Работа с Геолокацией

Управление Accessibility

Переключение Локализации

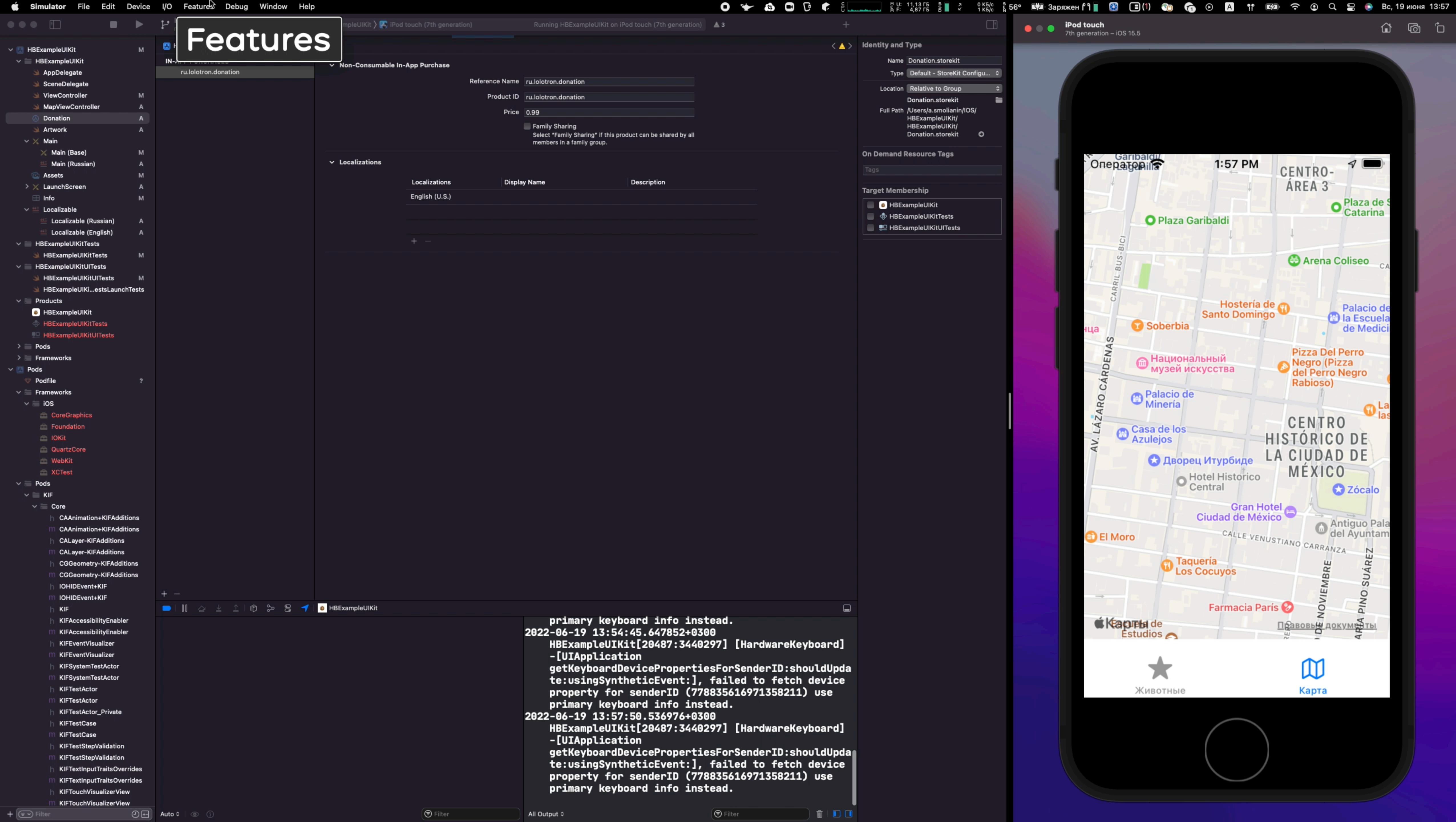
Поддержка Metal

Работа со Геолокацией

- Предопределенное место
- Кастомные координаты
- Симуляция движения

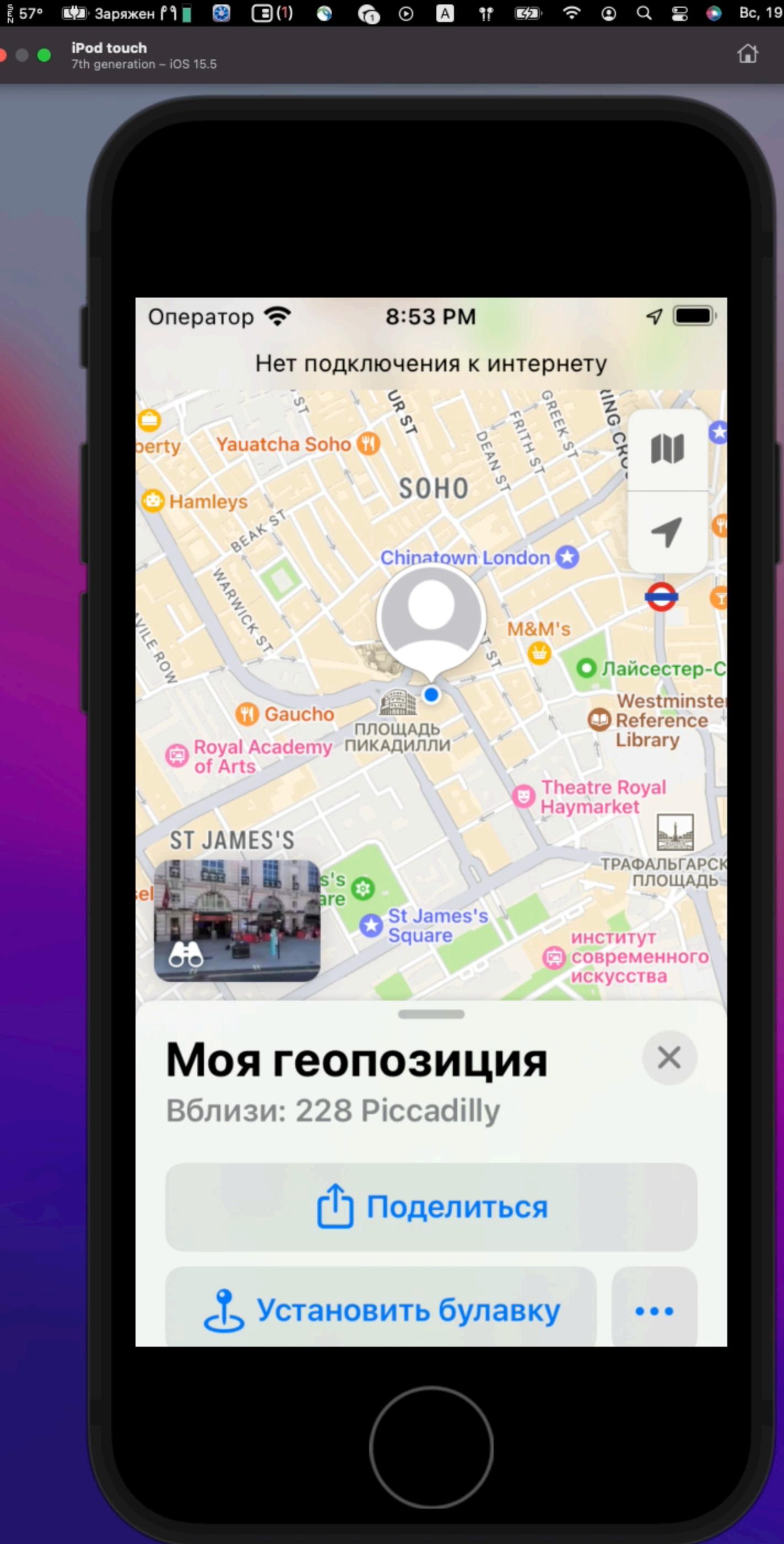
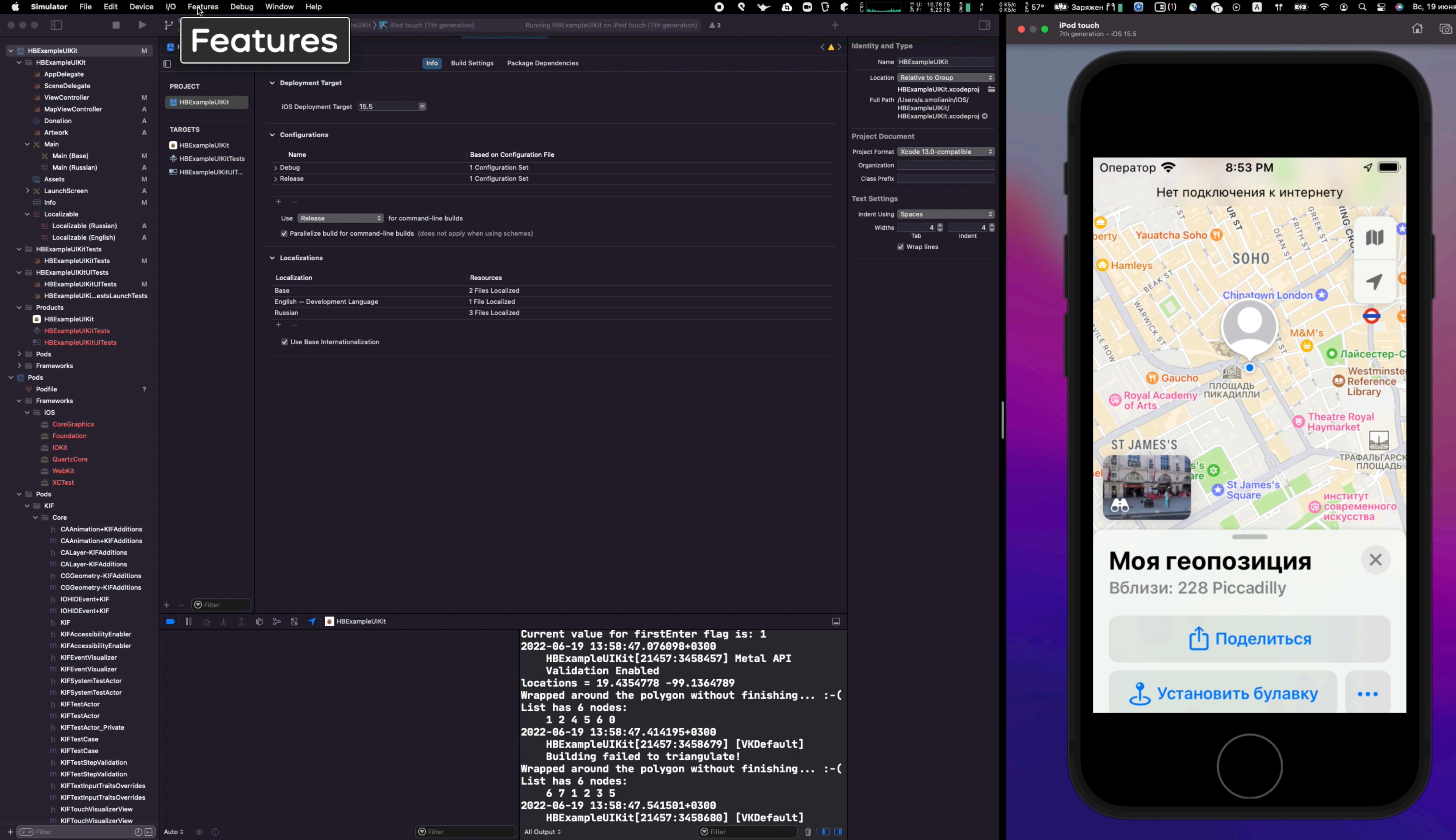
Предопределенное место

- Легче всего установить
- Доступ из Features меню
- Доступ из Scheme
- Доступ из интерфейса

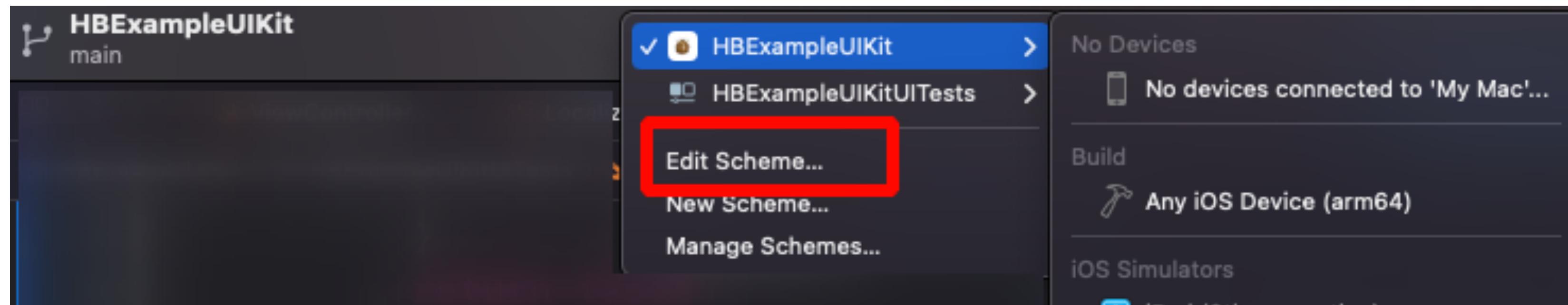


Кастомные координаты

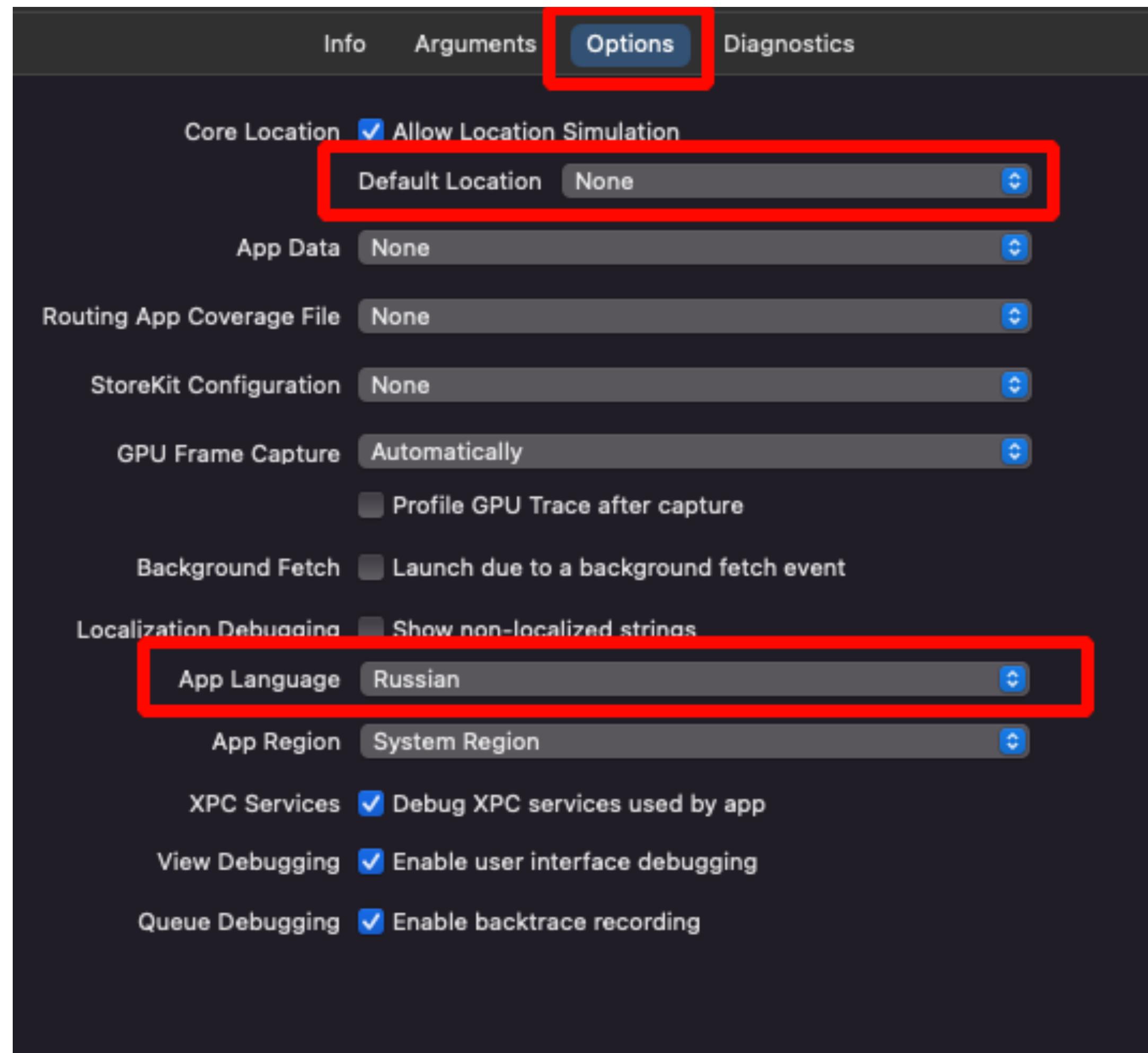
- Из меню
- GPX файл

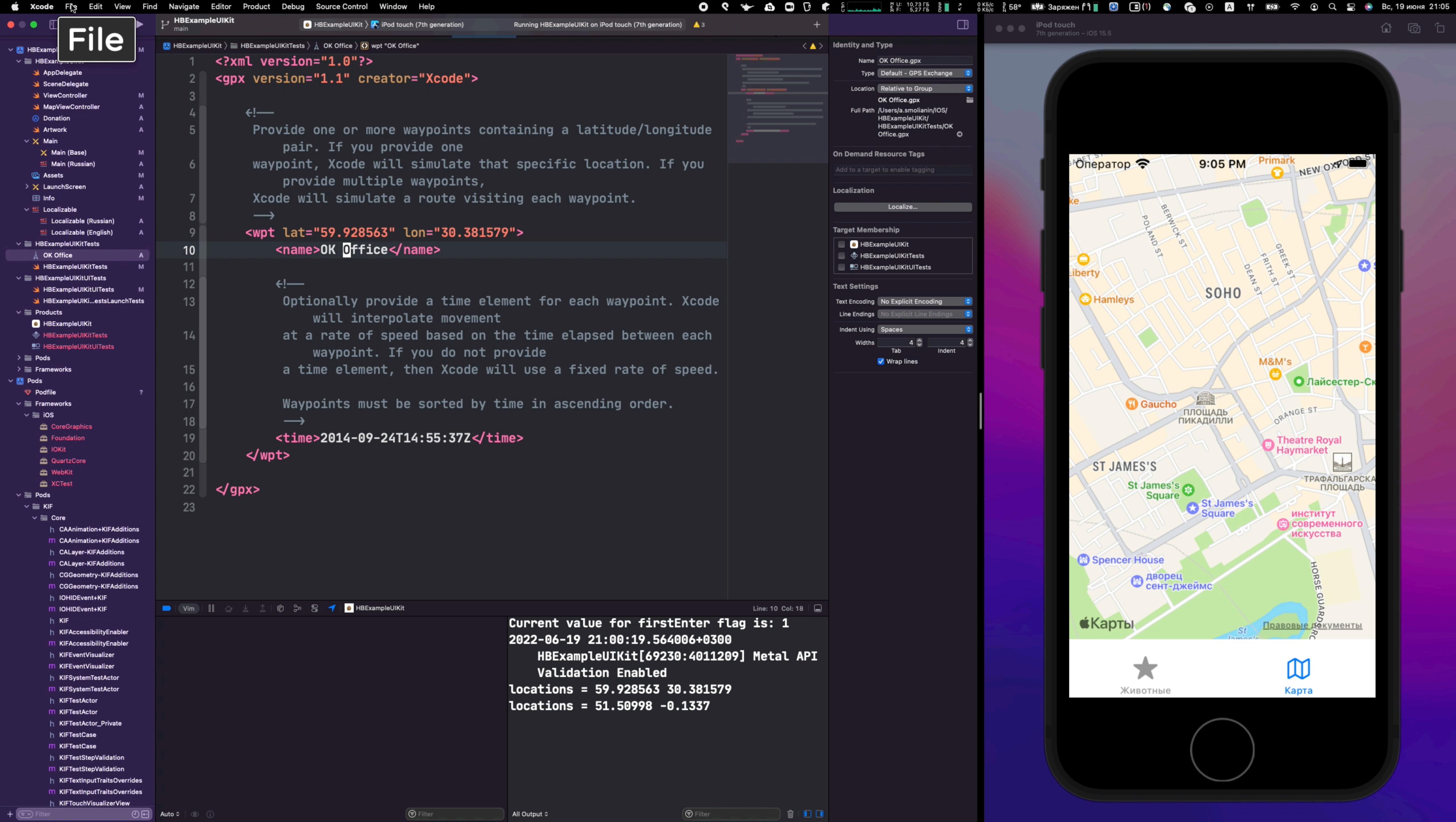


Схемы в XCode



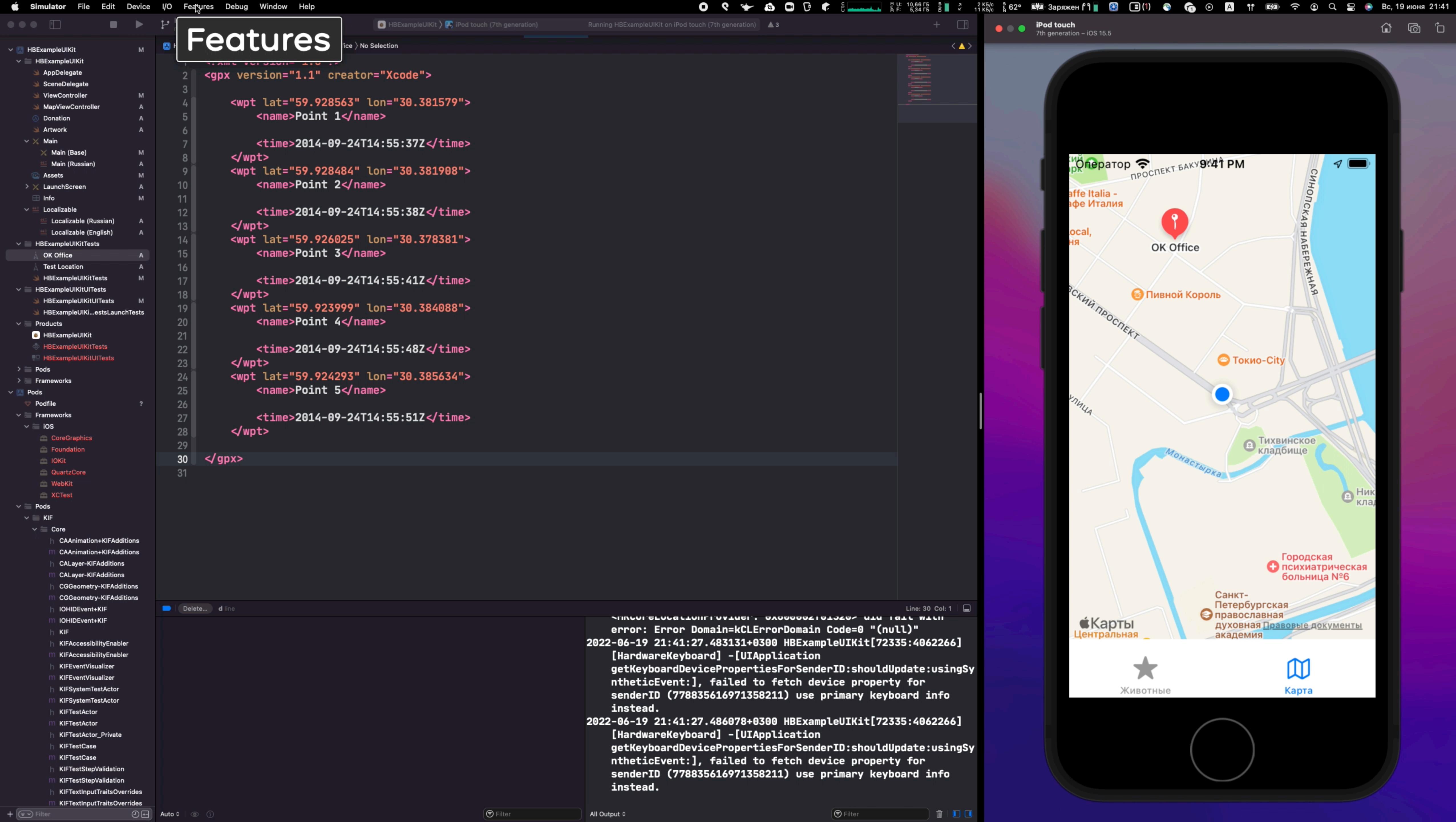
Настройки схемы





Симуляция движения

- Из меню
- GPX файл





Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

Работа с Геолокацией

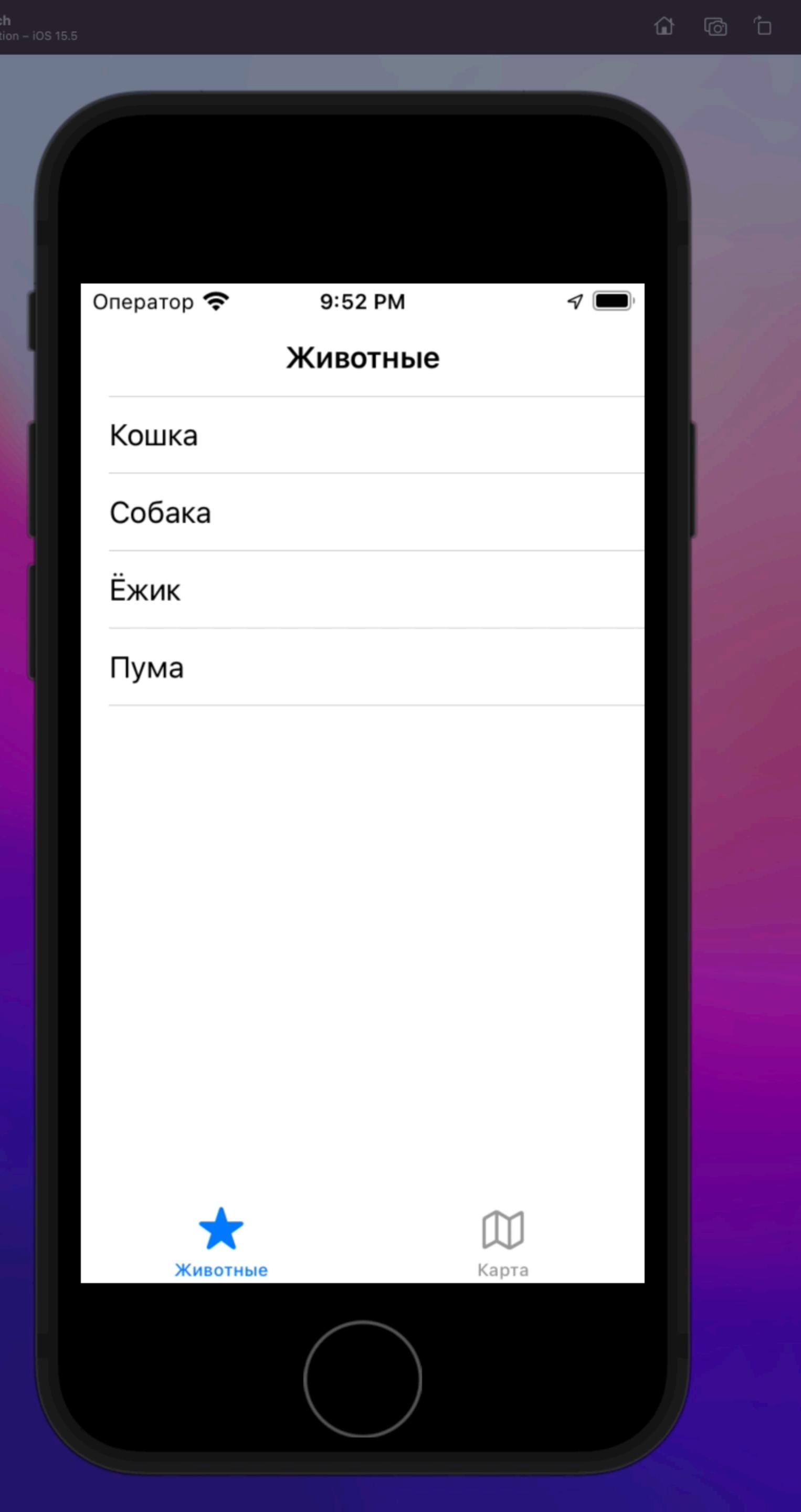
Управление Accessibility

Переключение Локализации

Поддержка Metal

Управление Accessibility

- Размер текста
- Светлая / Темная тема
- Контрастность
- Очертания кнопок





Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

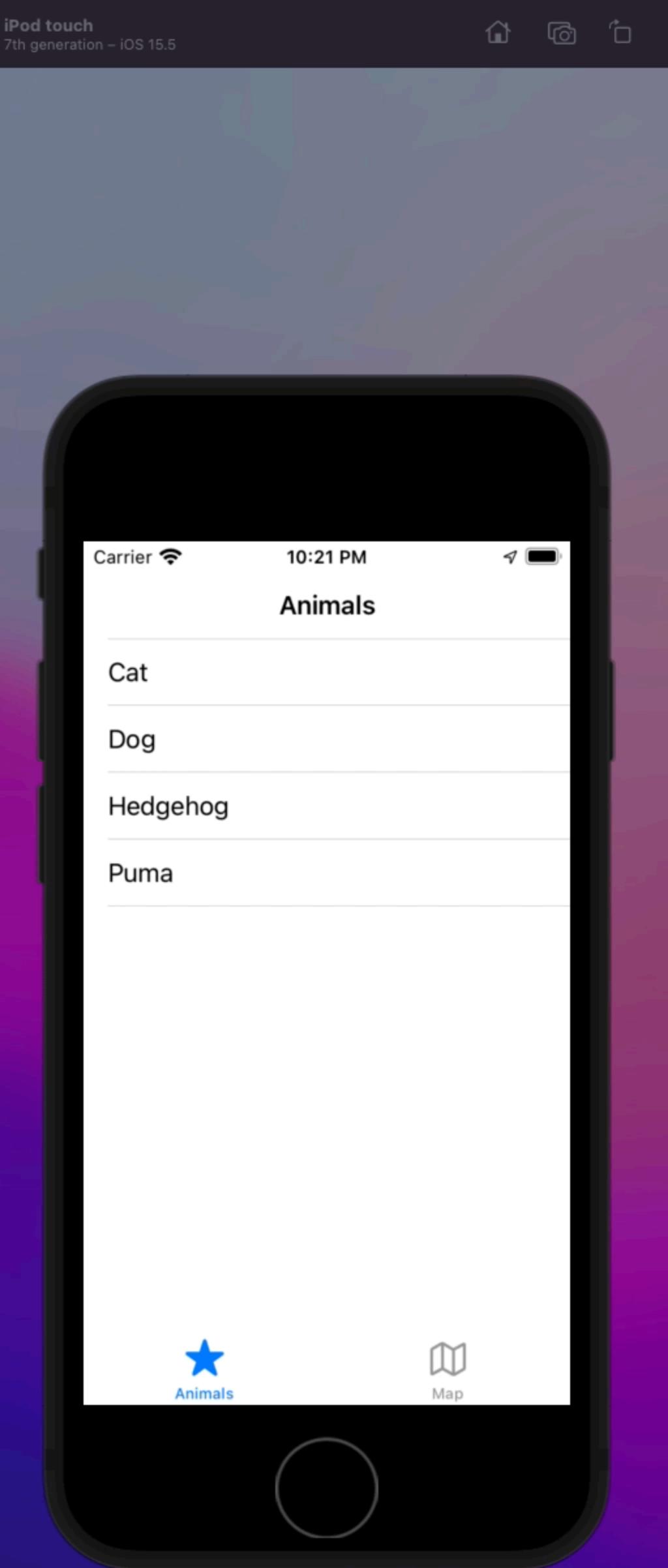
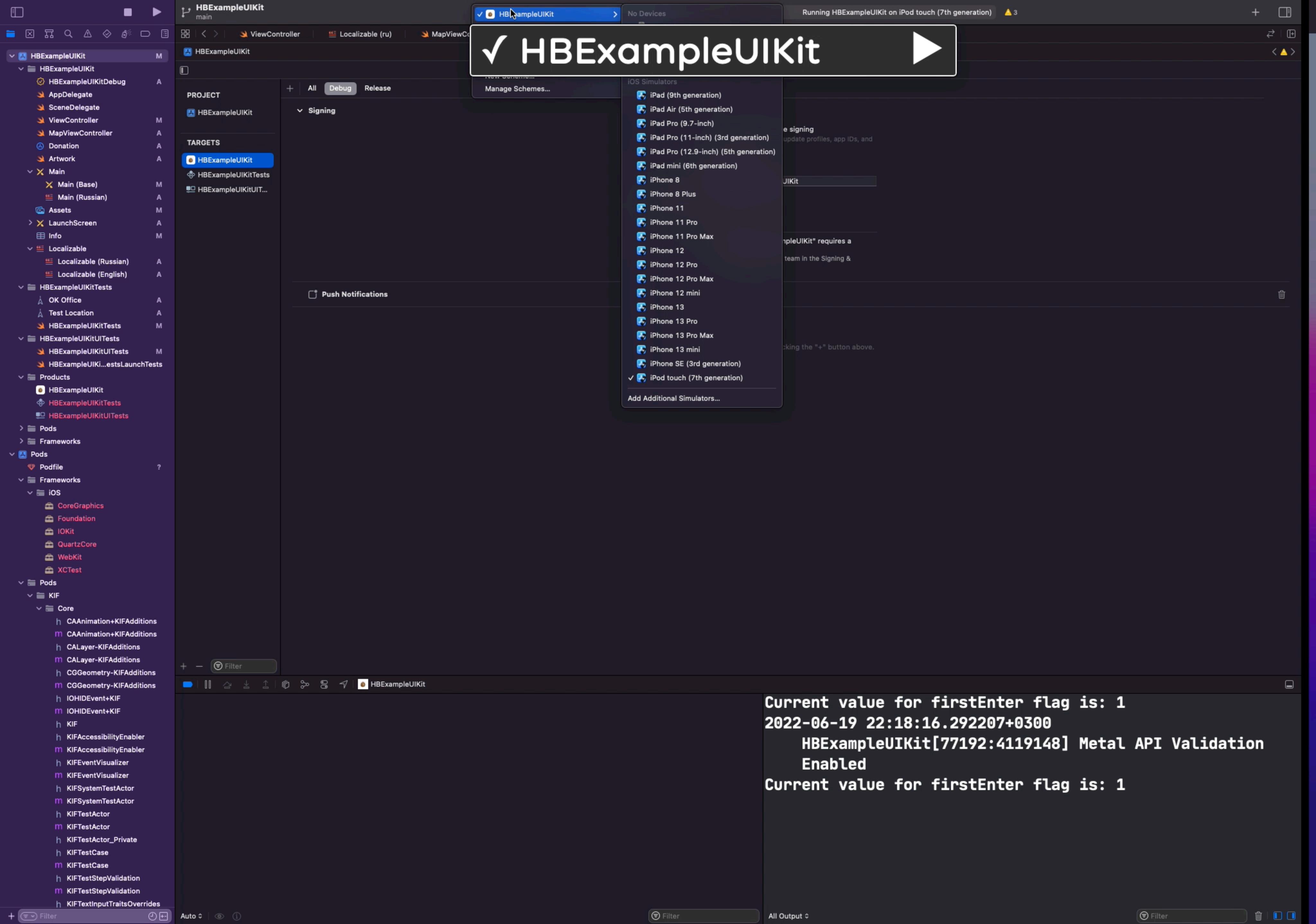
Работа со StoreKit

Работа с Геолокацией

Управление Accessibility

Переключение Локализации

Поддержка Metal





Симулятор

Запускать несколько симуляторов на одной машине

Взаимодействовать как с другими приложениями в системе

Запускать симуляторы в headless режиме для тестов

Параллельный запуск тестов

Работа со StoreKit

Работа с Геолокацией

Управление Accessibility

Переключение Локализации

Поддержка Metal

Metal



Metal

Accelerating graphics and much more.

Поддержка Metal

60 FPS

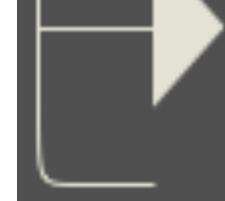
>90% потребление процессора

CLI
iOS-симулятора



Список симуляторов

Получение списка девайсов:



```
xcrun simctl list devices -j $searchTerm
```

Получение симулятора

```
xcrun simctl list devices -j $searchTerm
```

```
xcrun simctl list devices "iphone 13"
== Devices ==
-- iOS 14.1 --
-- iOS 14.3 --
-- iOS 15.5 --
iPhone 13 Pro (099F2C2B-A1B2-422F-A0E9-038B550A894A) (Shutdown)
iPhone 13 Pro Max (64F11B2B-DAE7-4CC7-8A8B-6FBB4DCCE6D9) (Shutdown)
iPhone 13 mini (73B40ABD-5C89-470C-96B0-3C90439498F9) (Shutdown)
iPhone 13 (F393FDCB-5B54-430C-9510-8EA8E570F91F) (Shutdown)
```

Информация о симуляторе

```
{  
  "dataPath": "/Users/a.smolianin/Library/Developer/CoreSimulator/Devices/F393FDCB-5B54-430C-9510-8EA8E570F91F",  
  "dataPathSize": 13316096,  
  "logPath": "/Users/a.smolianin/Library/Logs/CoreSimulator/F393FDCB-5B54-430C-9510-8EA8E570F91F.log",  
  "udid": "F393FDCB-5B54-430C-9510-8EA8E570F91F",  
  "isAvailable": true,  
  "deviceTypeIdentifier": "com.apple.CoreSimulator.SimDeviceType.iPhone-13",  
  "state": "Shutdown",  
  "name": "iPhone 13"  
}
```

Start / Stop симулятора

Запуск симулятора:

```
xcrun simctl boot F393FDCB-5B54-430C-9510-8EA8E570F91F
```

Остановка одного или всех симуляторов:

```
xcrun simctl shutdown F393FDCB-5B54-430C-9510-8EA8E570F91F | all
```

Список приложений

Получение списка приложений

```
xcrun simctl listapps booted
```

Информация о приложении

```
"ru.lolotron.HBExampleUIKit" = {  
    "BundleContainer" = "file:///Users/a.smolianin/Library/Developer/..."  
    "DataContainer" = "file:///Users/a.smolianin/Library/Developer/..."
```

Контейнеры приложения

Запрос пути до папки Data

```
xcrun simctl get_app_container booted <appId> data
```

Абсолютный путь до папки Data

```
/Users/a.smolianin/Library/Developer/CoreSimulator/Devices/81F...
```

Скриншоты и видео

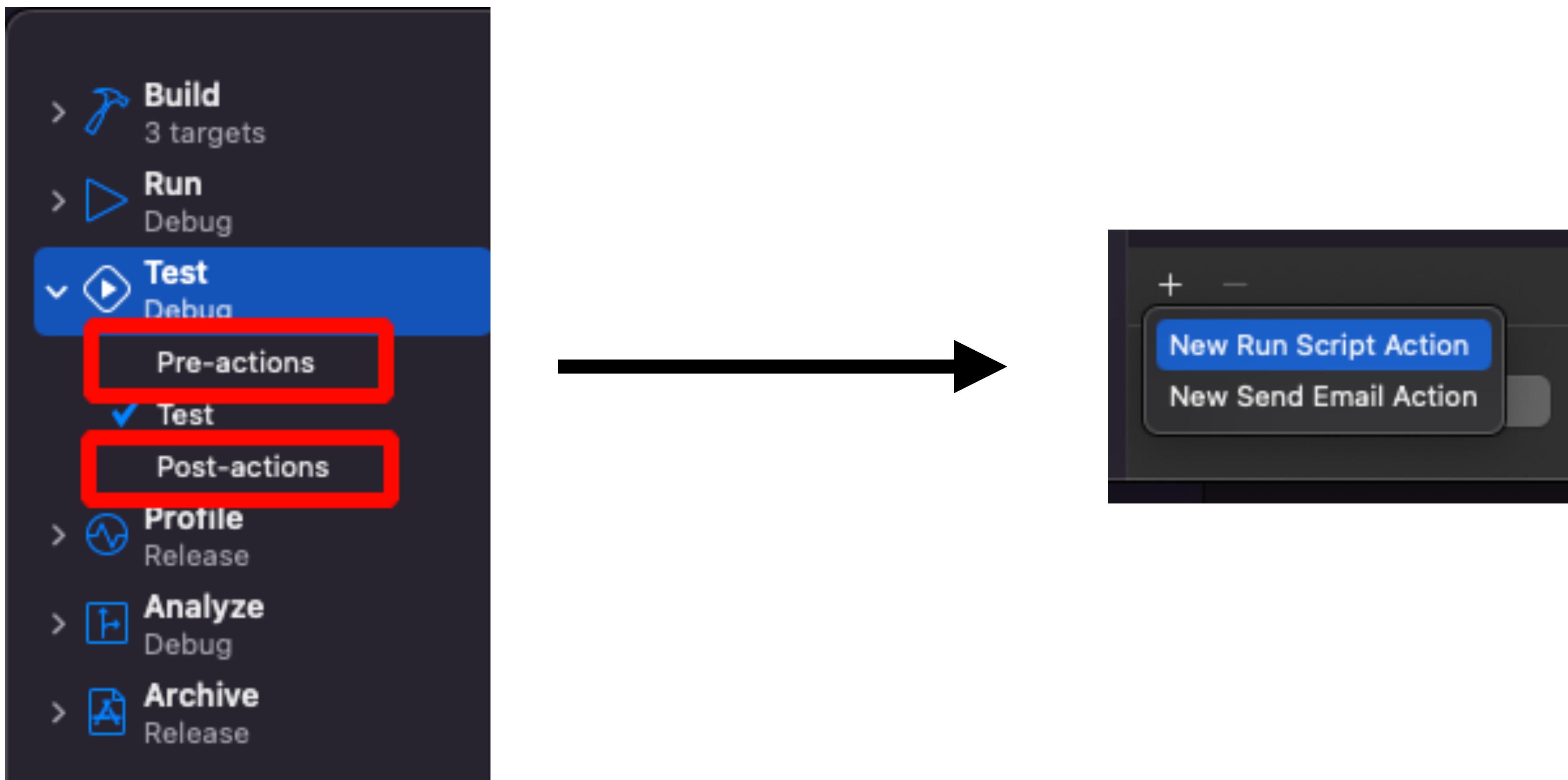
Запись видео:

```
xcrun simctl io booted recordVideo -f --codec=h264 ~/video.mp4
```

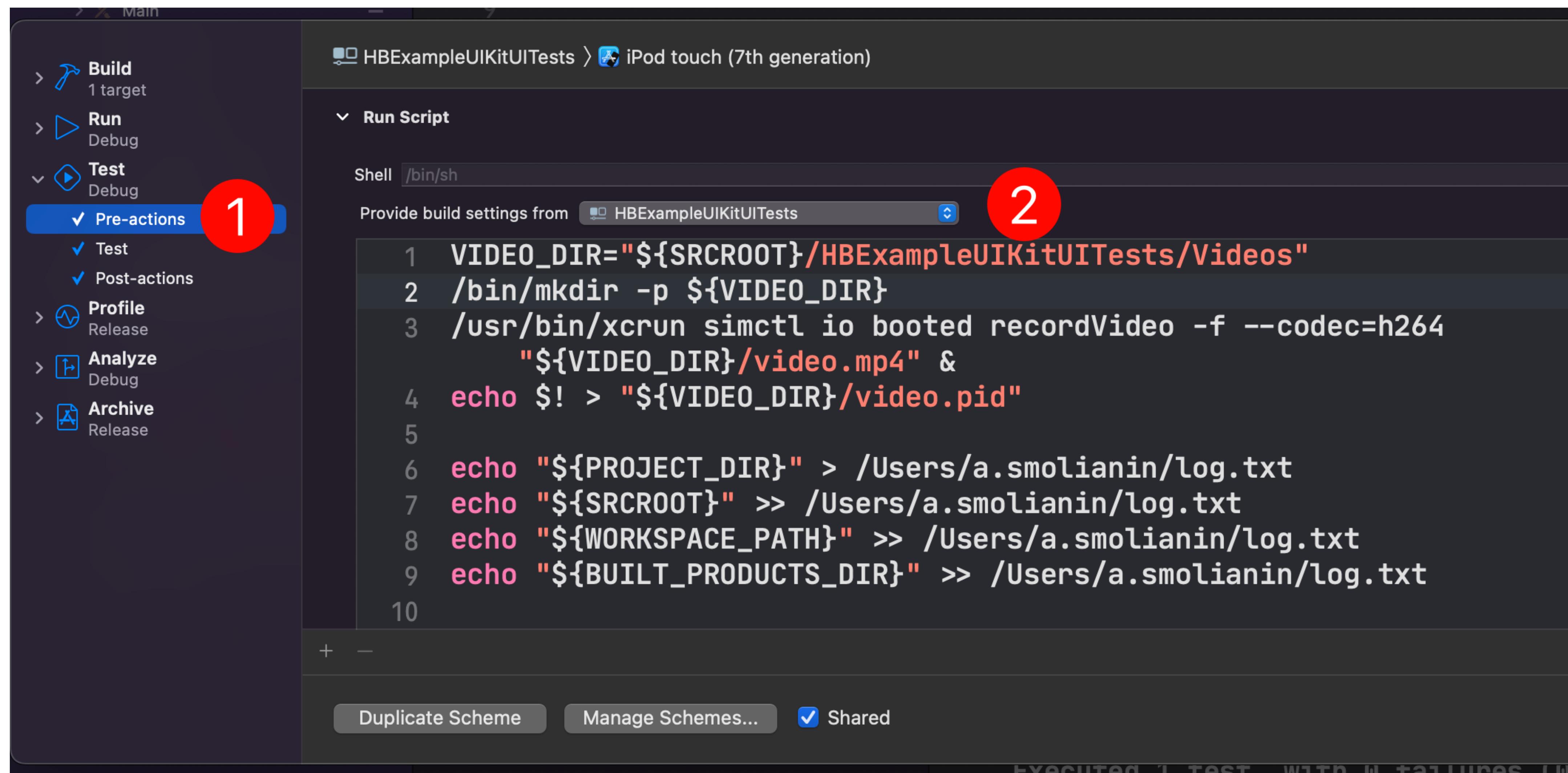
Снятие скриншота

```
xcrun simctl io booted screenshot screenshot.png
```

Дополнительная подготовка



Pre-action



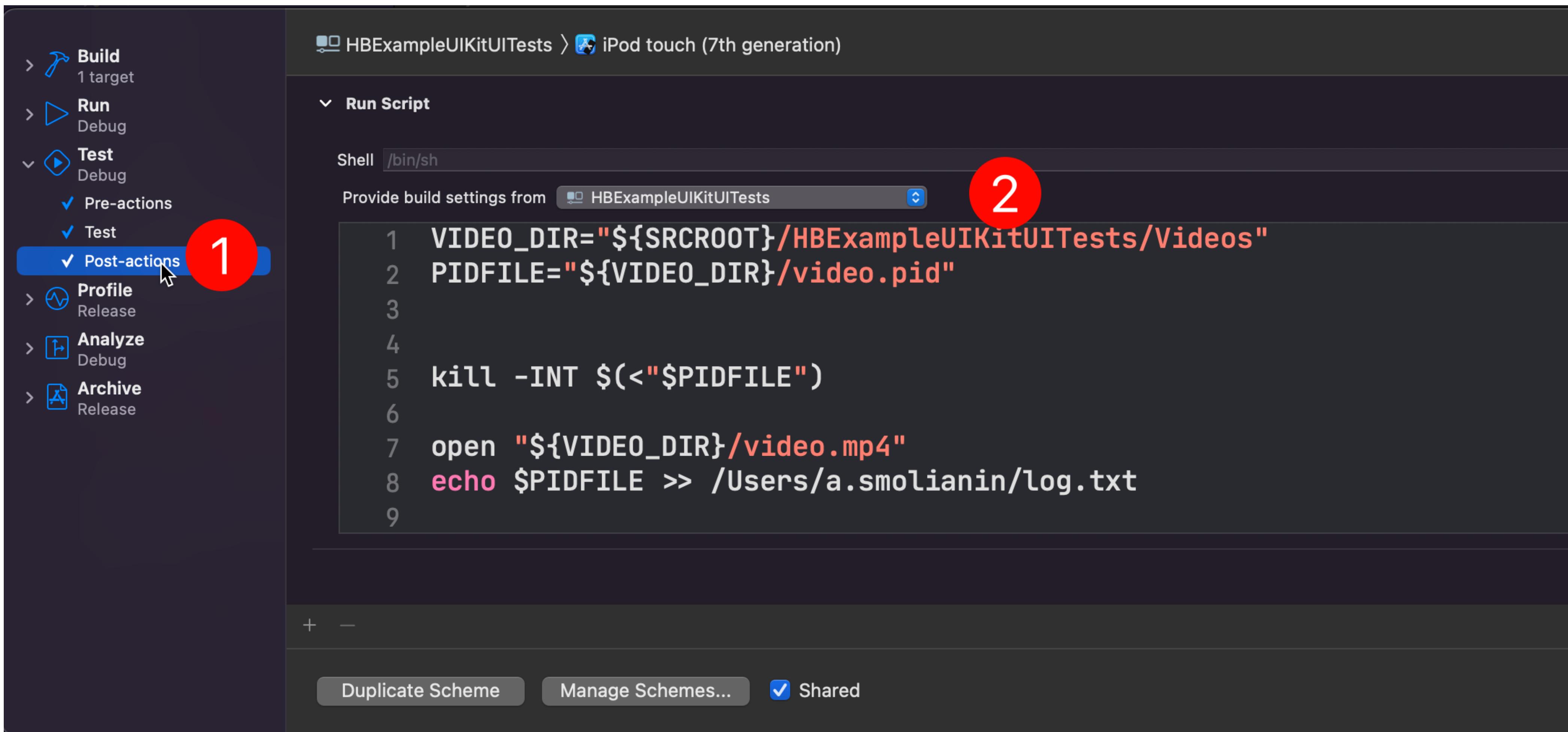
Pre-action

The screenshot shows the Xcode Scheme editor for the 'HBExampleUIKitUITests' scheme. The target is set to 'iPod touch (7th generation)'. In the 'Run Script' section, the 'Shell' is set to '/bin/sh' and the build settings are provided from 'HBExampleUIKitUITests'. The script contains the following code:

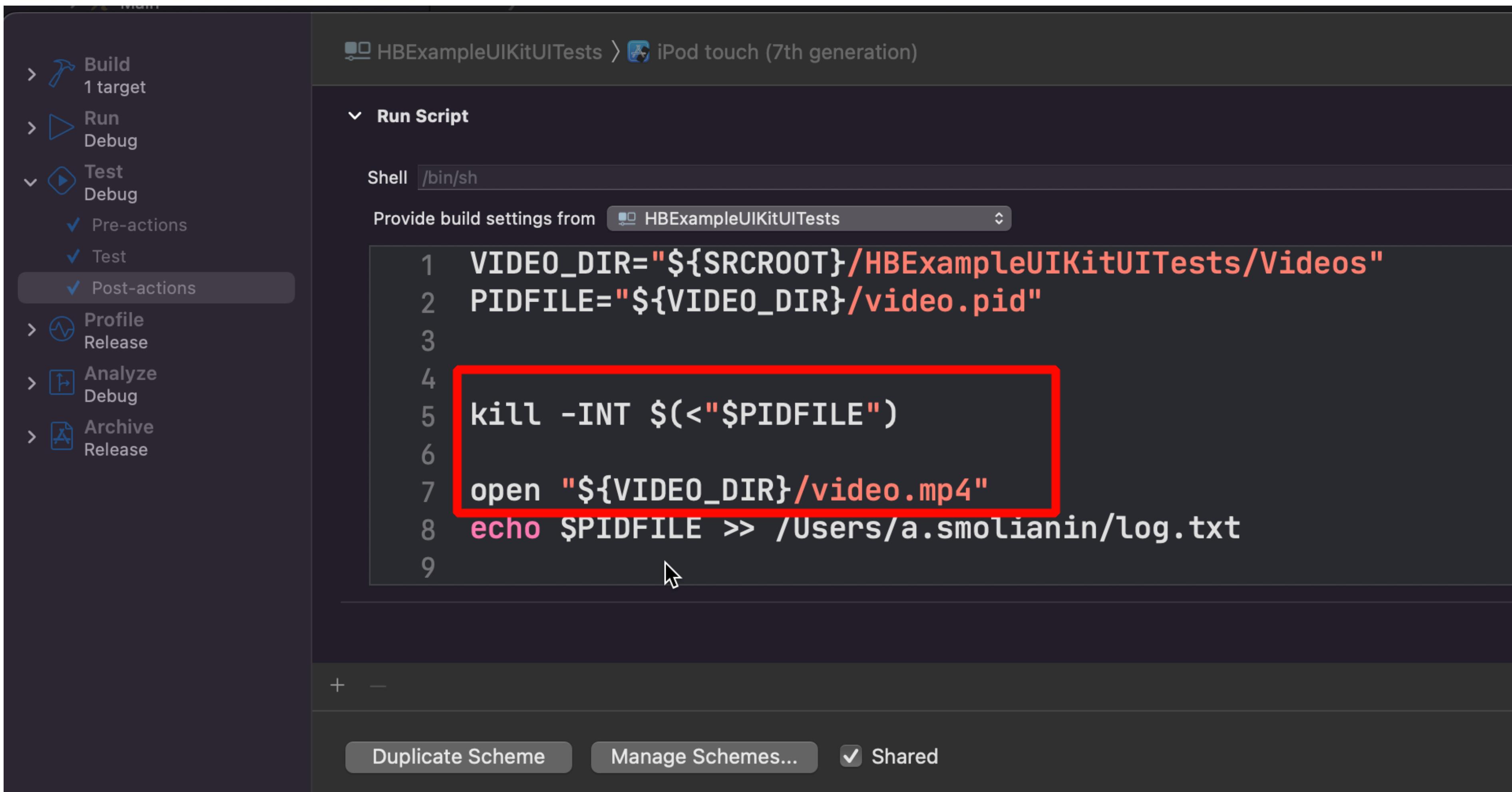
```
1 VIDEO_DIR="${SRCROOT}/HBExampleUIKitUITests/Videos"
2 /bin/mkdir -p ${VIDEO_DIR}
3 /usr/bin/xcrun simctl io booted recordVideo -f --codec=h264
   "${VIDEO_DIR}/video.mp4" &
4 echo $! > "${VIDEO_DIR}/video.pid"
5
6 echo "${PROJECT_DIR}" > /Users/a.smolianin/log.txt
7 echo "${SRCROOT}" >> /Users/a.smolianin/log.txt
8 echo "${WORKSPACE_PATH}" >> /Users/a.smolianin/log.txt
9 echo "${BUILT_PRODUCTS_DIR}" >> /Users/a.smolianin/log.txt
10
```

The line starting with '/usr/bin/xcrun simctl ...' is highlighted with a red rectangle.

Post-action



Post-action



The screenshot shows the Xcode Scheme editor for the 'HBExampleUIKitUITests' scheme. The left sidebar lists various build actions: Build, Run, Test (with sub-options Pre-actions, Test, and Post-actions selected), Profile, Analyze, and Archive. The main pane displays the 'Run Script' action for the 'iPod touch (7th generation)' target. The shell is set to '/bin/sh'. The script content is:

```
1 VIDEO_DIR="${SRCROOT}/HBExampleUIKitUITests/Videos"
2 PIDFILE="${VIDEO_DIR}/video.pid"
3
4
5 kill -INT $(cat $PIDFILE)
6
7 open "${VIDEO_DIR}/video.mp4"
8 echo $PIDFILE >> /Users/a.smolianin/log.txt
9
```

The lines from 5 to 7 are highlighted with a red rectangle.

At the bottom of the editor, there are buttons for 'Duplicate Scheme', 'Manage Schemes...', and a checked 'Shared' checkbox.

Надо помнить при настройке

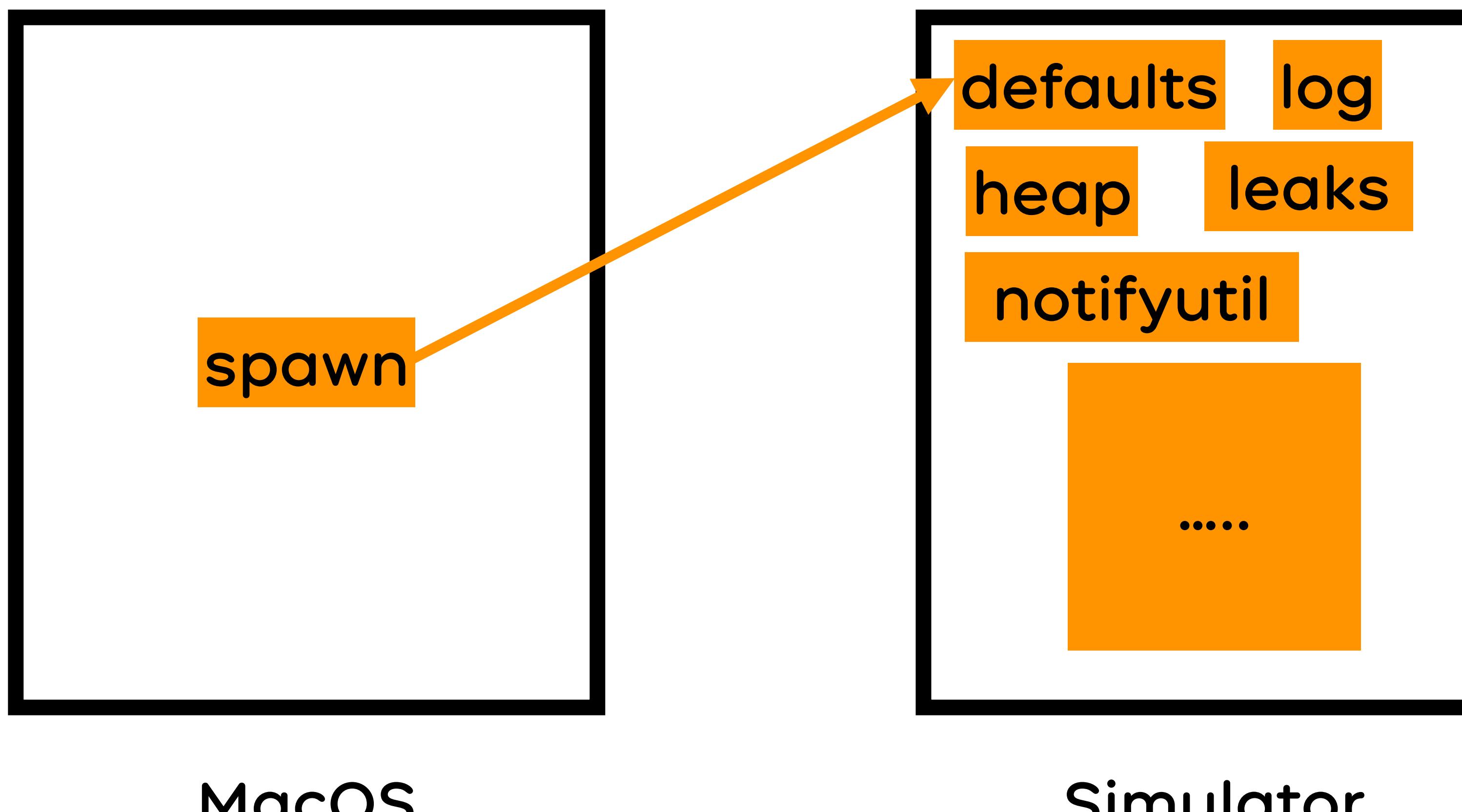
- Абсолютные пути до команд
- Указать приложение для Build Setting
- Логировать в файл
- Не работают с xcodebuild (из консоли)

A screenshot of an Xcode workspace. The left side shows the code editor with a file named 'HBExampleUIKitUITests.swift'. The code contains several test functions, including one for location access and another for a map feature. The right side shows a simulated iPhone screen displaying the home screen with various app icons. The top status bar indicates the device is an 'iPod touch' running '7th generation - iOS 15.5'.

Команда Spawn



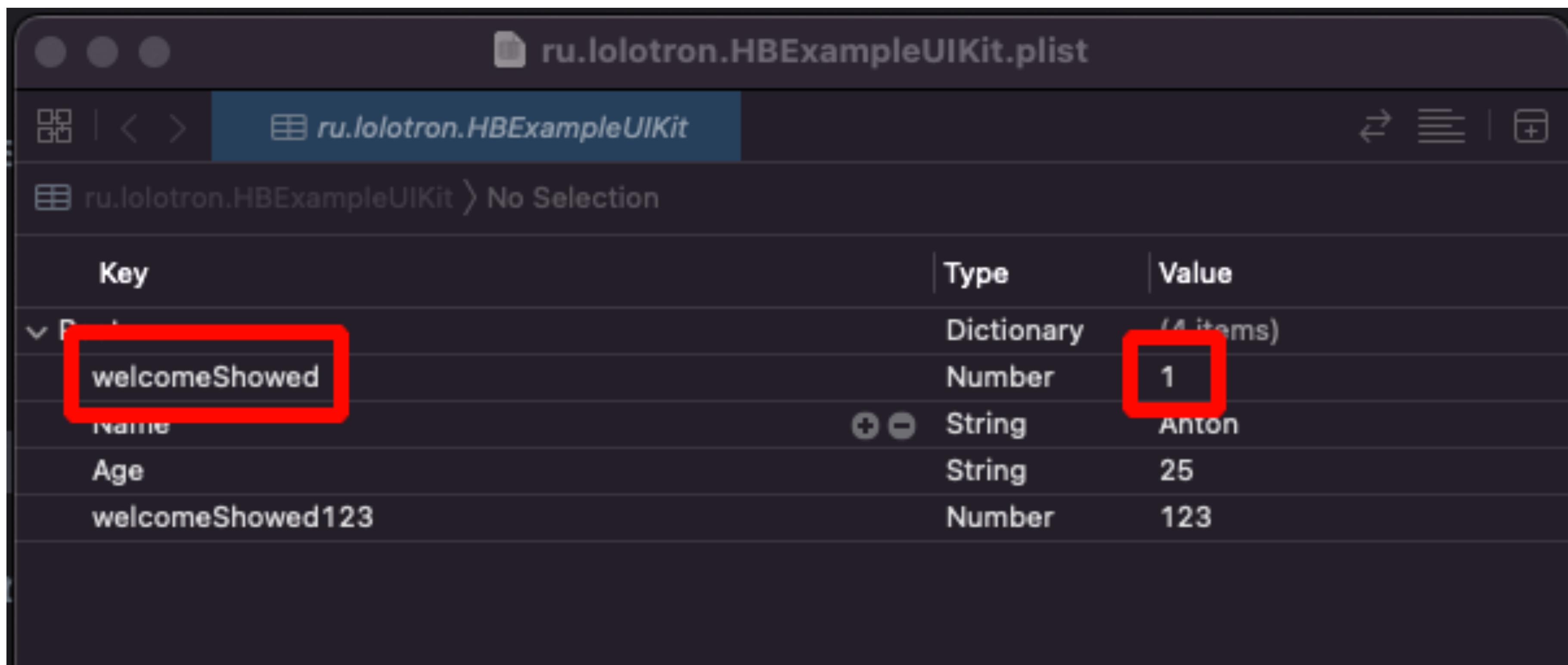
Команда Spawn



MacOS

Simulator

Как хранятся UserDefaults





```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```

```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```



```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```



```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```

```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```



```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```

```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```



```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выведем логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
```

Click here to configure status bar

```
1 BUNDLE_ID="ru.lolotron.HBExampleUIKit"
2
3 # Получим папку Data для приложения
4 DATA_FOLDER=$(xcrun simctl get_app_container booted $BUNDLE_ID data)
5
6 # Получим путь до файла с настройками
7 PATH_TO_DEFAULTS_FILE="$DATA_FOLDER/Library/Preferences/$BUNDLE_ID.plist"
8
9 # Завершим приложение
10 xcrun simctl terminate booted $BUNDLE_ID
11
12 # Импортируем все настройки из файла
13 xcrun simctl spawn booted defaults import $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
14 # Поменяем флаг на 0
15 xcrun simctl spawn booted defaults write $BUNDLE_ID welcomeShowed -integer 0
16 # Экспортируем настройки обратно в файл
17 xcrun simctl spawn booted defaults export $BUNDLE_ID $PATH_TO_DEFAULTS_FILE
18
19 # Запускаем приложение и выводим логи в консоль
20 xcrun simctl launch --console-pty booted $BUNDLE_ID
21
22
23
24
25
26
```

~

~

~

~

~

~

~

~

~

~

~

~

~

~

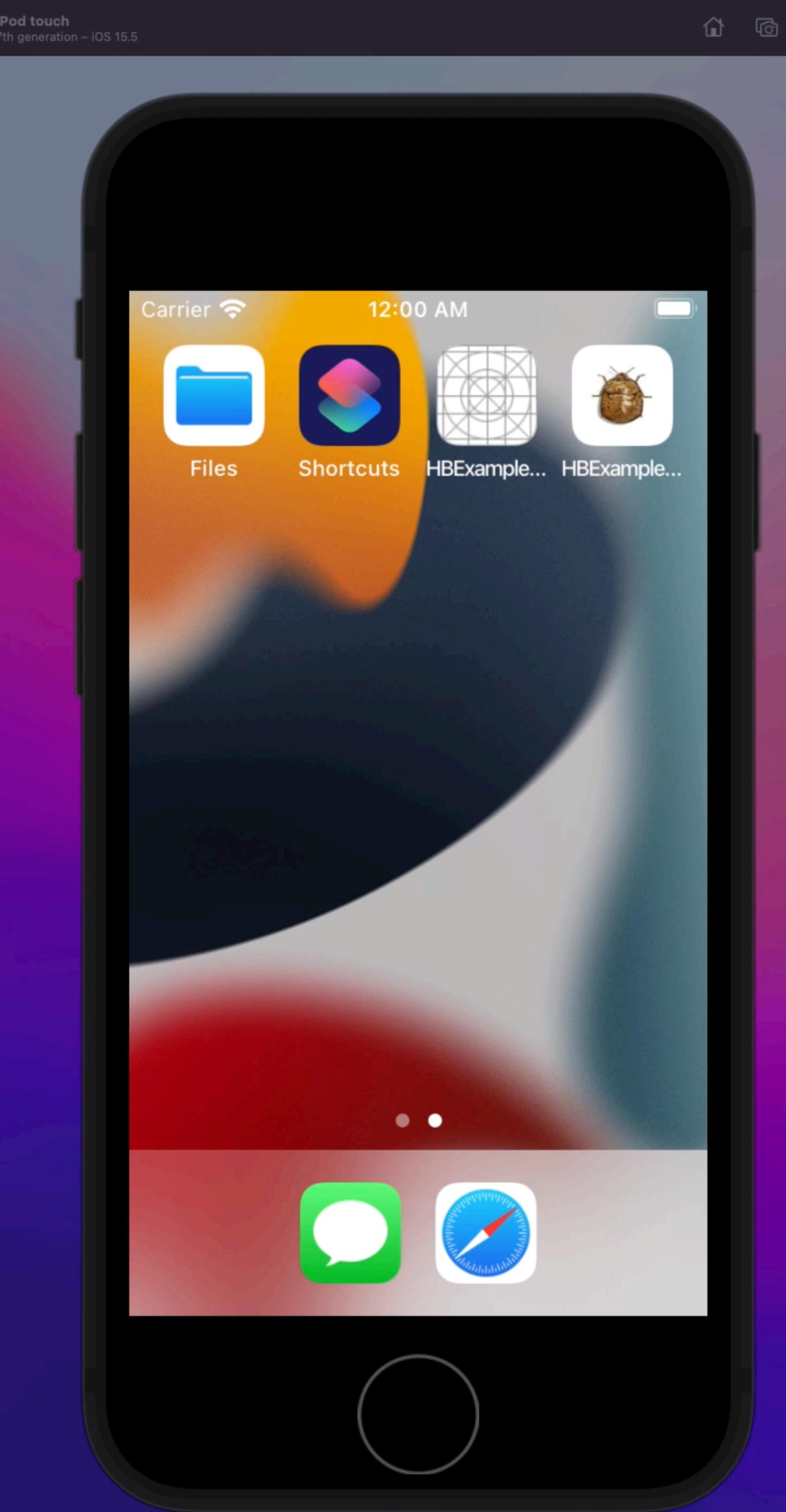
~

~

~

NORMAL defaults.sh

sh utf-8[unix] 92% In :24/26≡%:1



IDB

- Facebook
- Бывший FBSimCtl
- Физические устройства
- Доступ к скрытым возможностям XCode
- Без Accessibility framework

Ограничения Симулятора



Ограничения

- Камера
- Телефон
- Гироскоп
- APNS пушки
- Отключение сети
- Микрофон

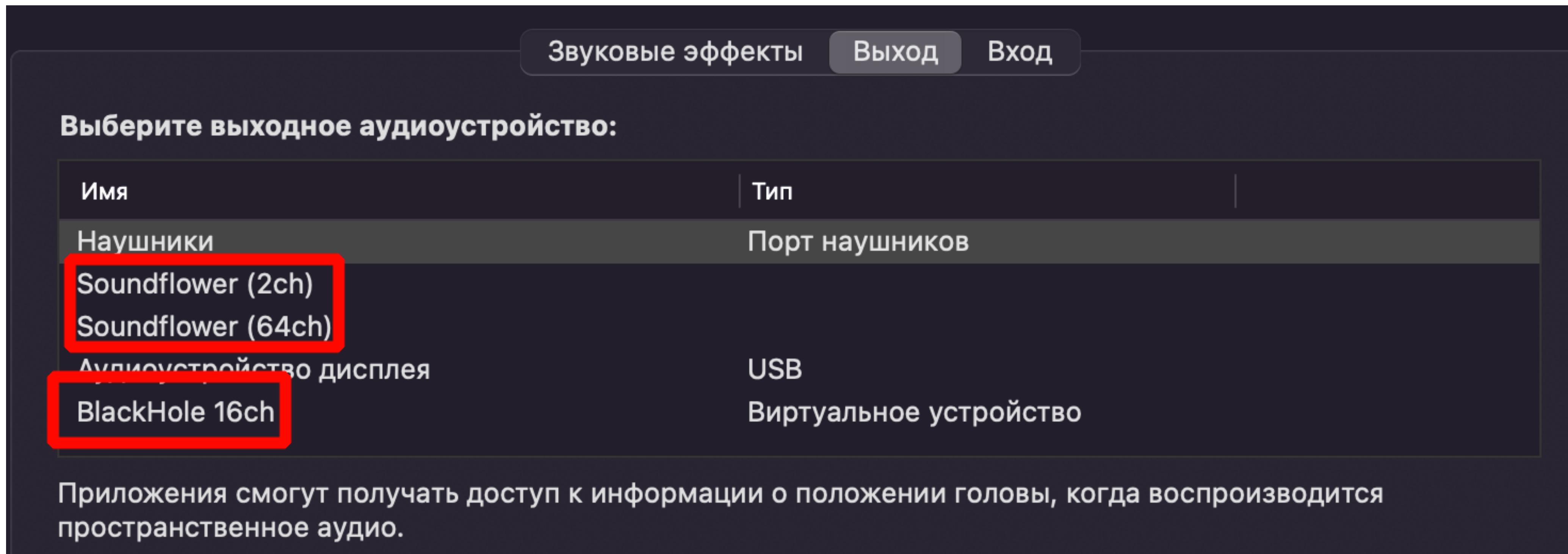
SBTUITestTunnel

- Позволяет мокать сеть
- Позволяет мокать уведомления (Notifications)

Микрофон

- Soundflower x86
- BlackHole M1

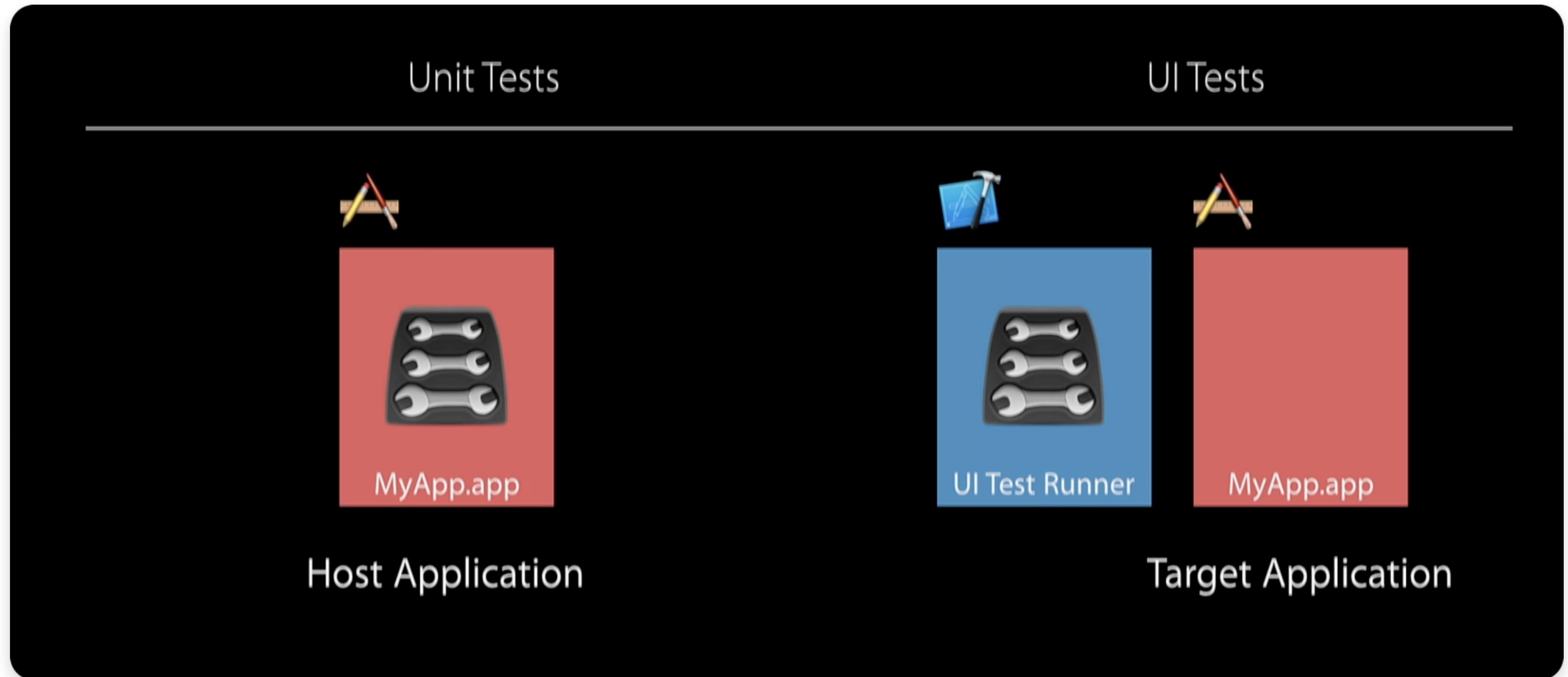
Отображение в системе



Проблемы

- Не выставить перемешены из Unit Test'ов
- М1 не работает буфер обмена

Виды тестов



Applesimutils

- Консольная утилита
- Выставляет перемещены
- Выставляет геолокацию

дополнительные инструменты



Control Room

- UI вокруг simctl
- OpenSource
- Нативное приложение

iPhone

- Default
- For auto-tests (14.1)
- iPhone 11 (15.5)
- iPhone 11 Pro (15.5)
- iPhone 11 Pro Max (1...)
- iPhone 11 Pro Max (1...)
- iPhone 12 (15.5)
- iPhone 12 Pro (15.5)
- iPhone 12 Pro Max (1...)
- iPhone 12 Pro Max (1...)
- iPhone 12 mini (15.5)
- iPhone 13 (15.5)
- iPhone 13 Pro (15.5)
- iPhone 13 Pro Max (1...)
- iPhone 13 mini (15.5)
- iPhone 8 (14.1)
- iPhone 8 (14.3)
- iPhone 8 (15.5)
- iPhone 8 Plus (14.1)
- iPhone 8 Plus (15.5)
- iPhone SE (3rd gene...)
- iPod touch (7th gene...)
- iPod touch (7th gene...)

iPad

- iPad (9th generation...)
- iPad Air (5th generat...)
- iPad Pro (11-inch) (3...)
- iPad Pro (12.9-inch)...)
- iPad Pro (9.7-inch) (...)
- iPad mini (6th gener...)

Apple TV

- Apple TV (15.4)
- Apple TV 4K (2nd ge...)
- Apple TV 4K (at 108...)

Apple Watch

- Apple Watch Series...

iPod touch (7th generation)
iOS 15.5 (19F70)

Shutdown

System App Battery Location Network Screen

Application: HBExampleUIKit – ru.lolotron.HBExampleUIKit

Show system apps

HBExampleUIKit
Version 1.0
Build 1

Launch Terminate Restart Uninstall

App Data

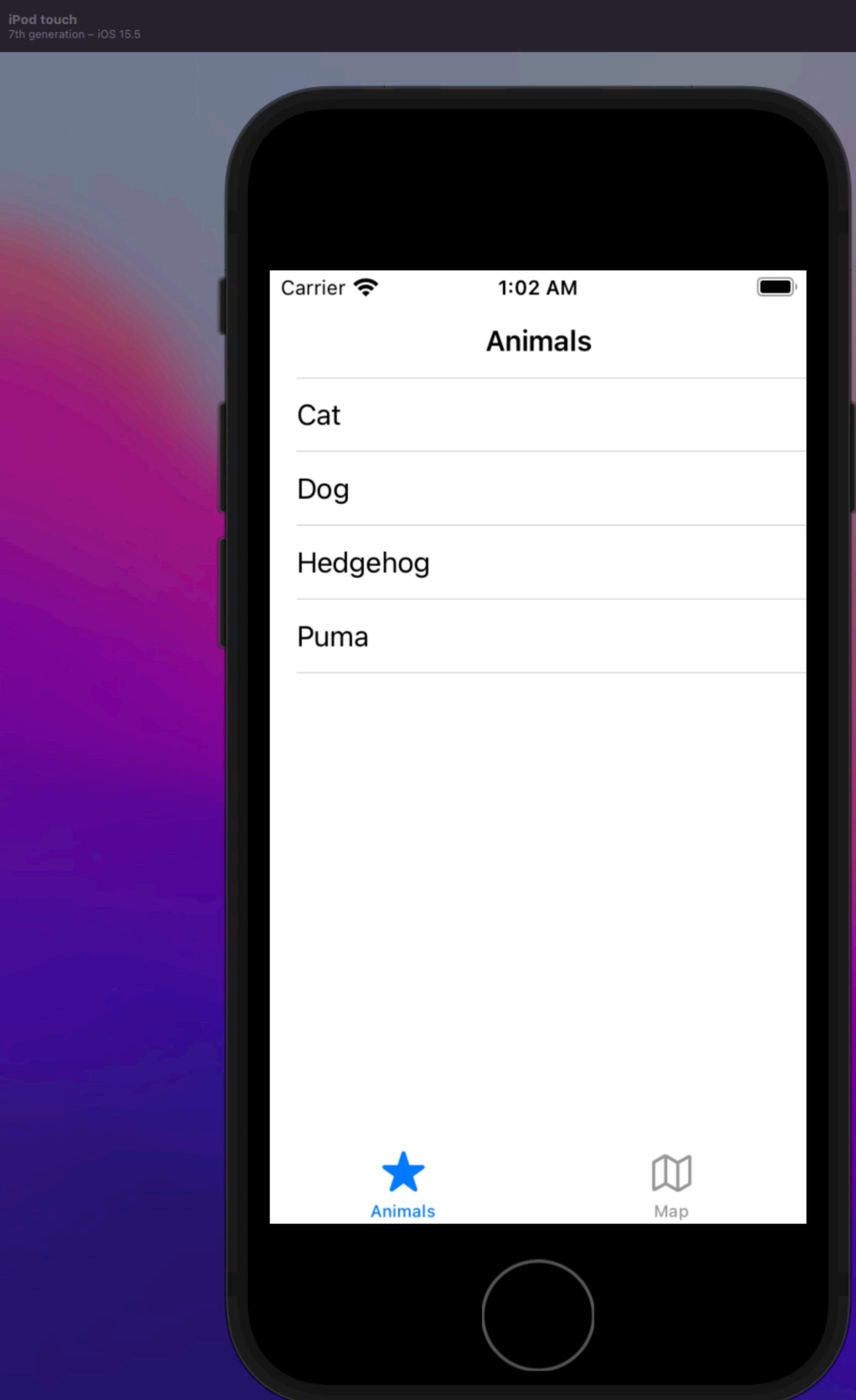
Permissions: All

Grant Revoke Reset

```
{  
    "ru.lolotron.HBExampleUIKit": {  
        "alert": {  
            "body": "Hello, World!",  
            "title": "From Control Room"  
        }  
    }  
}
```

Select Notification Templates

Open Notification Editor Send Push Notification



Выводы



Узнали много нового

- Познакомились с симулятором
- Научились автоматизировать подготовку перед тестом
- Хороший старт для нативного тестирования
- Узнали когда стоит использовать симулятор
- Узнали что симулятор не умеет
- Узнали как обходить ограничения

Спасибо за внимание

<https://github.com/gopito/Heisenbug-2022-Spring>

https://t.me/anton_smolianin

