



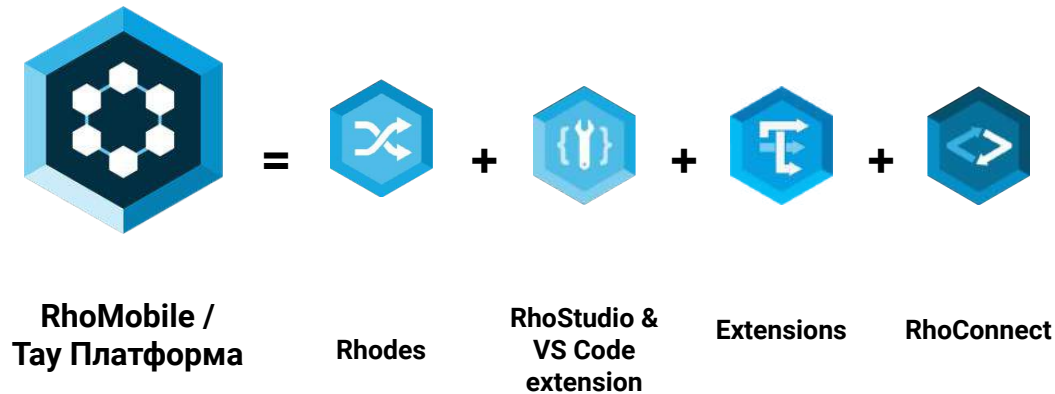
Разработка приложений для мобильных и настольных платформ с помощью единого инструмента.

RhoMobile

## Александр Епифанов

- Соучредитель и исполнительный директор в “Тау Технологиях”.
- В разработке ПО более 20 лет, начинал с мобильных приложений для Palm OS.
- Писал софт в геймдеве, телекоме, для встраиваемых систем.
- В разработке кроссплатформенных инструментов с 2012 года.
- “Играющий тренер” в Тау с 2015 года.
- Активный контрибьютор OSS.
- Контакты тут: <http://epifanoff.name/>.

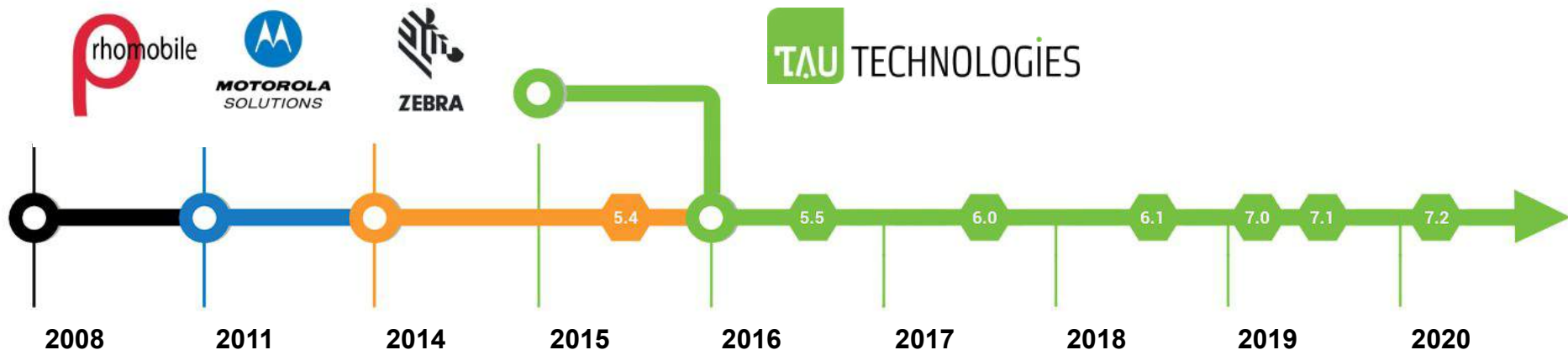




- Лицензия - MIT.
- Первый релиз - в 2008 году.
- Более 1000 “звёзд” на GitHub.
- Класс ПО - платформа для разработки приложений.
- Фокус - на бизнес приложения, доступна поддержка промышленных устройств и специальных API. Обеспечивает синхронизацию данных между приложением и корпоративным бэкендом.
- Модель распространения - свободно без ограничений.
- Модель продаж - техническая поддержка, разработка дополнительных функций, обучение.
- Тип приложений - гибридно-нативный.
- Предназначено для разработки переносимых приложений с использованием веб-технологий.
- Обеспечивает унификацию кода приложения на всех целевых ОС.
- Не noCode.
- Поддерживаются все актуальные мобильные и настольные ОС.
- Предоставляет разработчикам механизм реализации расширений.
- Можно разрабатывать на Windows, Linux, OS X.
- Включает инструментарий для разработчиков: IDE, отладчик, эмулятор.

## От стартапа в Калифорнии к российской импортозамещающей платформе:

- 2008** Адамом Блюмом ( Adam Blum ) основан стартап RhoMobile. Ментором проекта становится создатель языка Ruby Юкиhiro Мацумото.
- 2011** RhoMobile приобретен компанией Motorola Solutions.
- 2014** RhoMobile приобретен компанией Zebra Technologies как часть Motorola enterprise бизнеса.
- 2015** Несколько участников RhoMobile основали компанию Tau Technologies.
- 2016** Zebra перевела RhoMobile в open-source под MIT RMS 5.4 последняя версия выпущенная Zebra.  
Тау становится технологическим партнером Zebra.
- 2016-2022** Тау продолжает развивать проект.
- 2022** RMS 7.5 выпущена Тау, в настоящий момент проходит регистрацию в реестре отечественного ПО как “Тау Платформа”.



# Из чего состоит

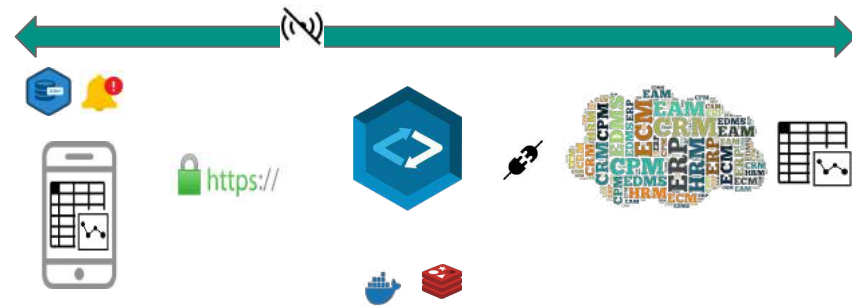
- Система сборки, собирающая родные артефакты, готовые к установке и запуску на целевой ОС.



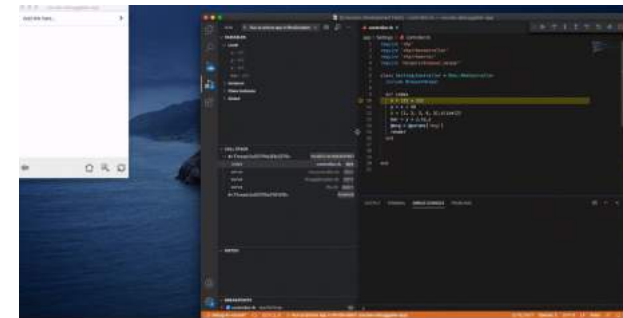
- Рантайм компоненты. Полный фреймворк для приложения, либо библиотека, плюс набор функциональных API.



- Серверные компоненты. Клиентское API с прозрачным ORM и push, и интеграционный сервер с возможностью подключения к любому типу бэкенда.



- IDE с отладчиком.
- Эмулятор. Даёт возможность запустить и проверить приложение на компьютере разработчика.





- Можно не (очень) разбираться в разработке мобильных и настольных приложений.
- Нужно уметь писать на Ruby или JavaScript, а также владеть HTML разметкой.
- Знания C/C++, Java/Kotlin, Objective C/Swift, C# не требуются.
- Можно перейти с серверного стека или из фронтенда.
- MVC архитектура приложения.
- Встроенный ORM с автоматической синхронизацией.
- Возможность переносить код из веб приложений или других гибридных платформ ( Cordova/PhoneGap ).
- Набор API для доступа к функциям устройств и ОС, поддающийся расширению.
- Поддерживает функциональность промышленных устройств с помощью независящего от вендора оборудования API.

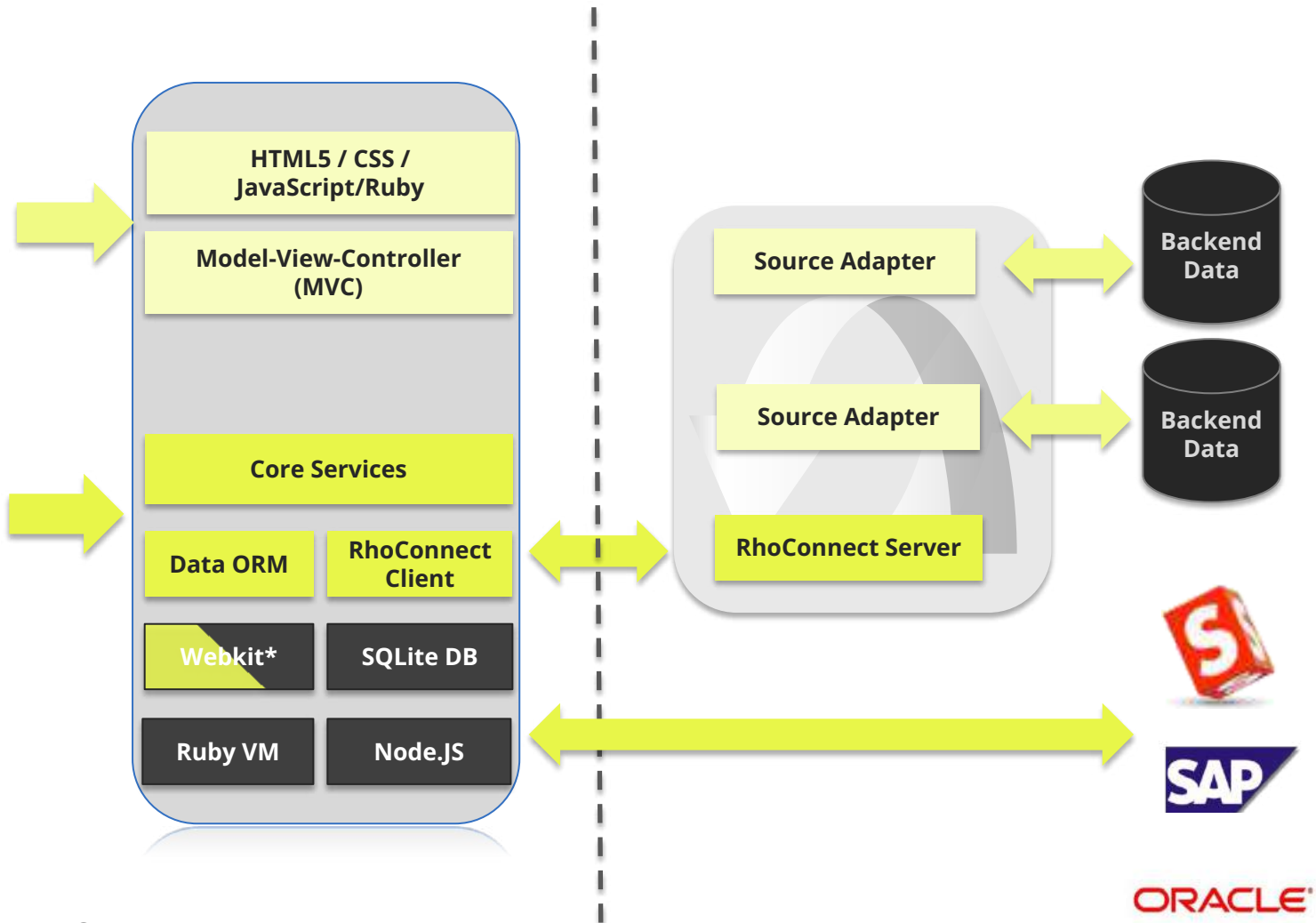


# Архитектура



You write the application

We provide the framework



\*Tau Webkit available on WinCE, Windows Mobile

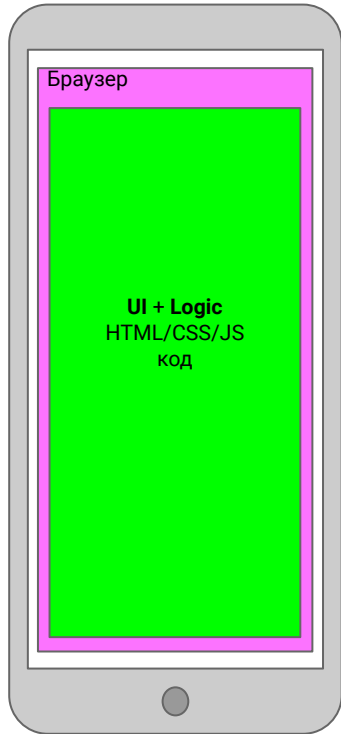
# Архитектуры кросс-платформенных приложений

Простое не кросс-платформенное нативное приложение



Windows: C#, C++  
 Linux: C++  
 Mac OS: Swift, ObjC, C++  
 iOS: ObjC, Swift, C++  
 Android: Java, C++  
 WinCE/WM: C#, C++

Web кросс-платформенное приложение



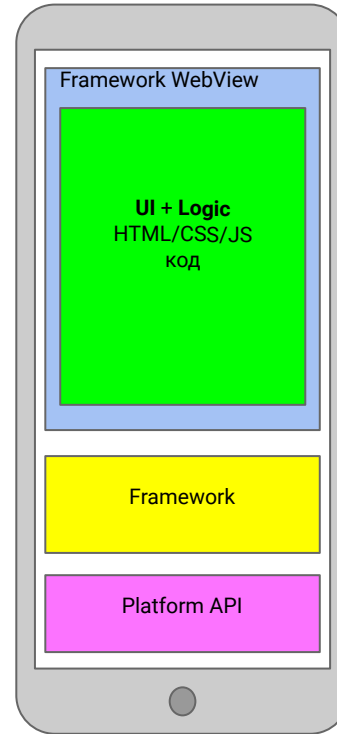
множество  
 HTML/CSS/JS фреймворков

Нативное кросс-платформенное приложение



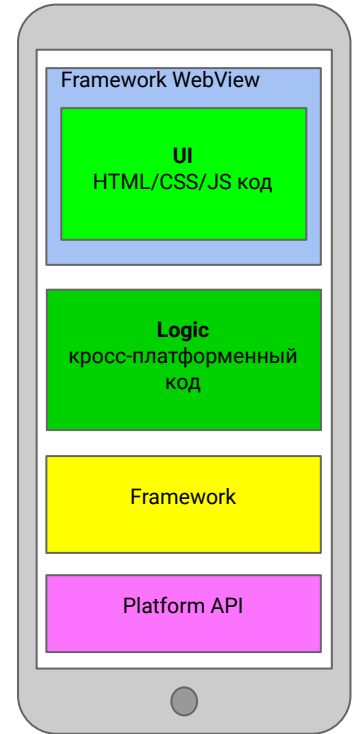
Mono, Xamarin (C#)  
 Appcelerator (JS)  
 React Native (JS)  
 NativeScript (JS)  
 QT (C++, QML)  
 Java  
 CodenameOne (Java)  
 Corona (Lua)  
 Flutter (Dart)

Гибридное кросс-платформенное приложение



Cordova/PhoneGap  
 RhoMobile, NW.js/Electron (в чисто гибридном режиме)  
 + а множество  
 HTML/CSS/JS фреймворков

Смешано-гибридное кросс-платформенное приложение

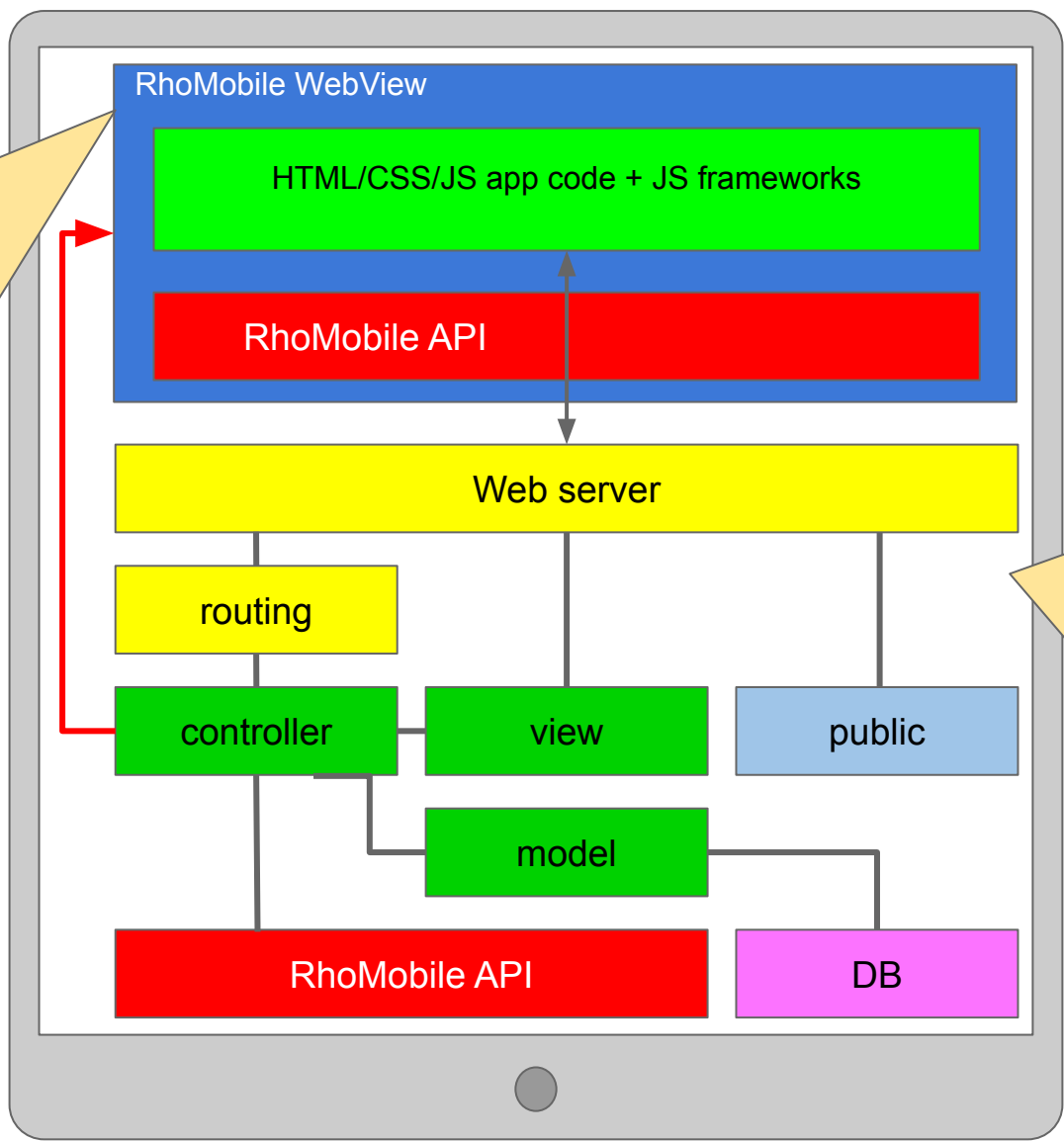


NW.js/Electron  
 RhoMobile (Ruby)  
 + множество  
 HTML/CSS/JS фреймворков

гибридные (web технологии)

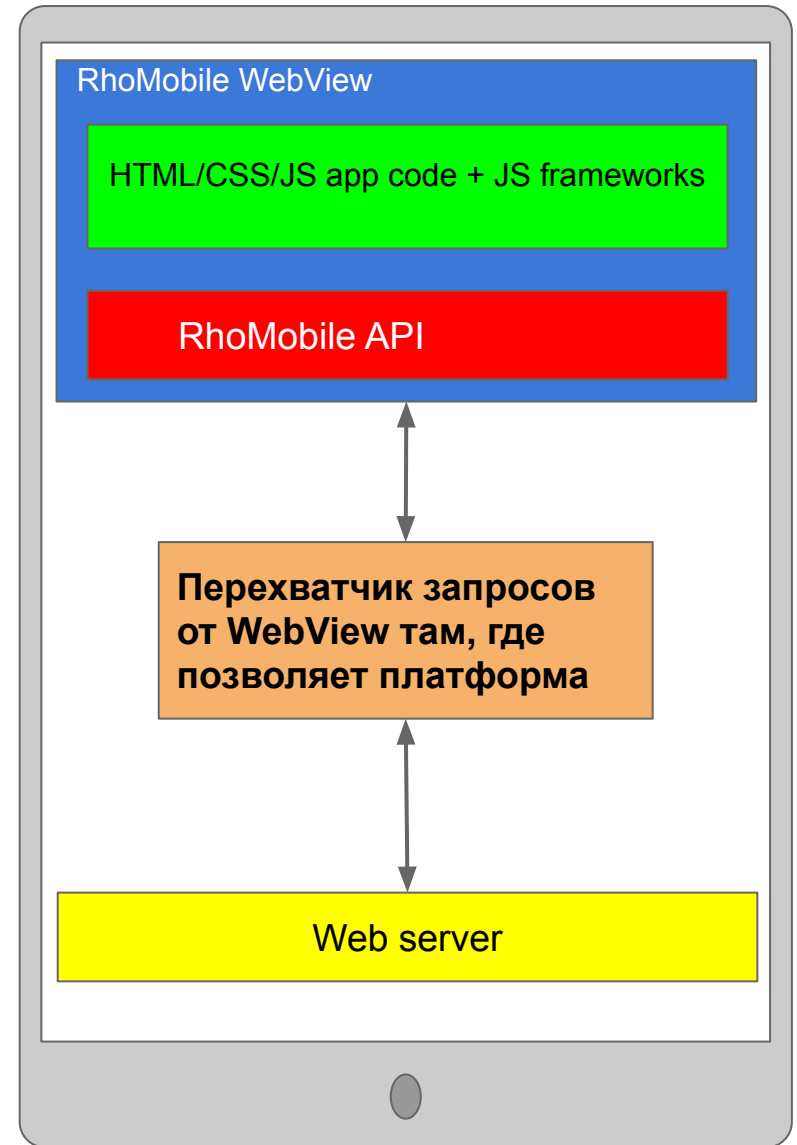
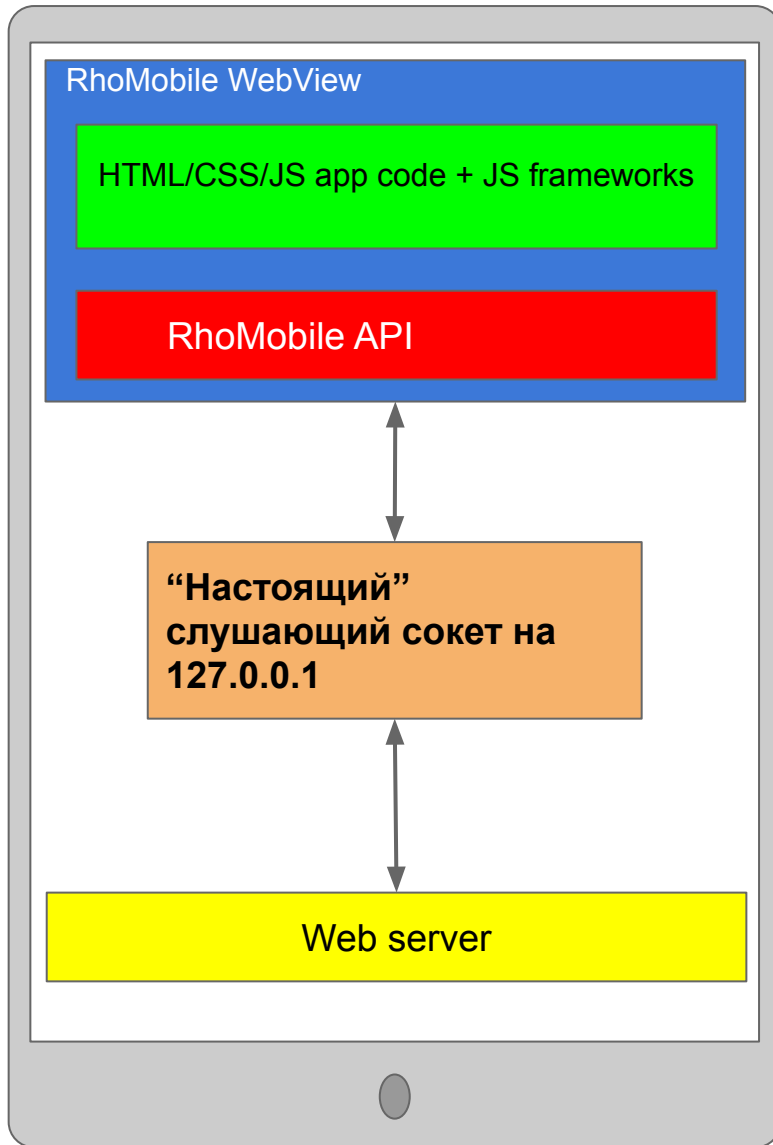
Гибридная архитектура. UI выполнен в WebView на базе веб технологий (HTML/CSS/JS ).

Разработчики могут свободно применять любые имеющиеся на рынке Javascript фреймворки.

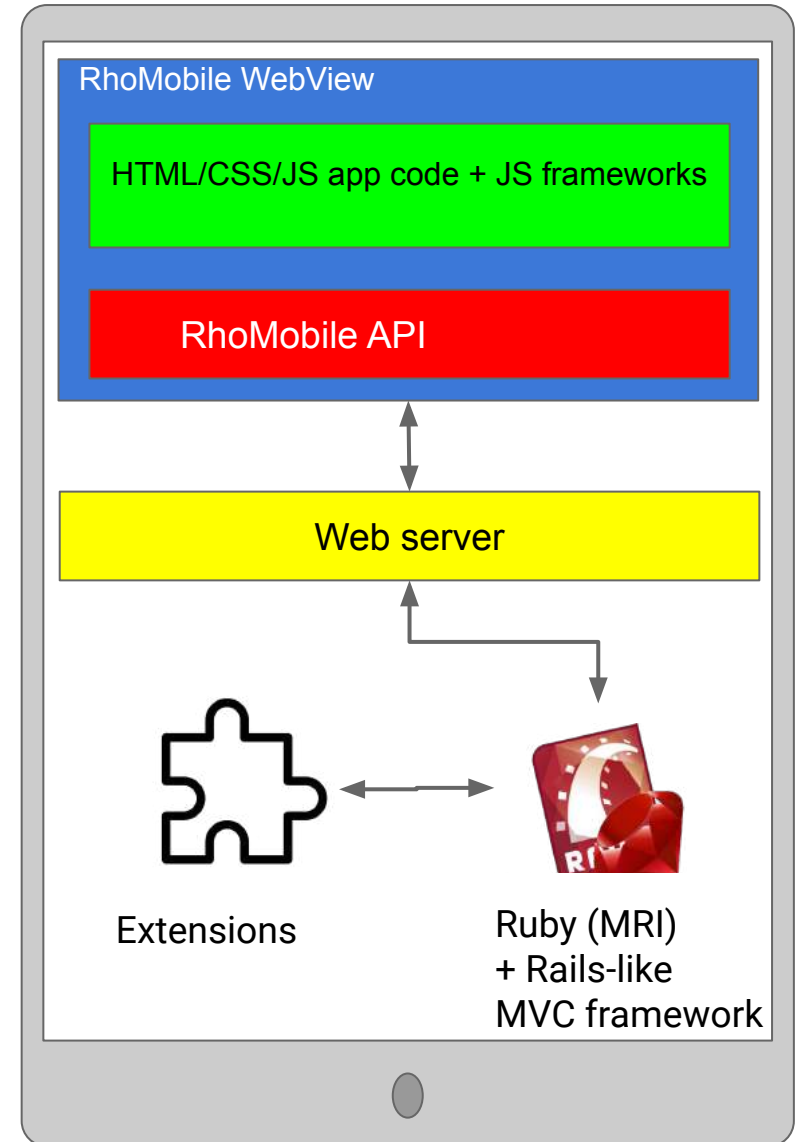


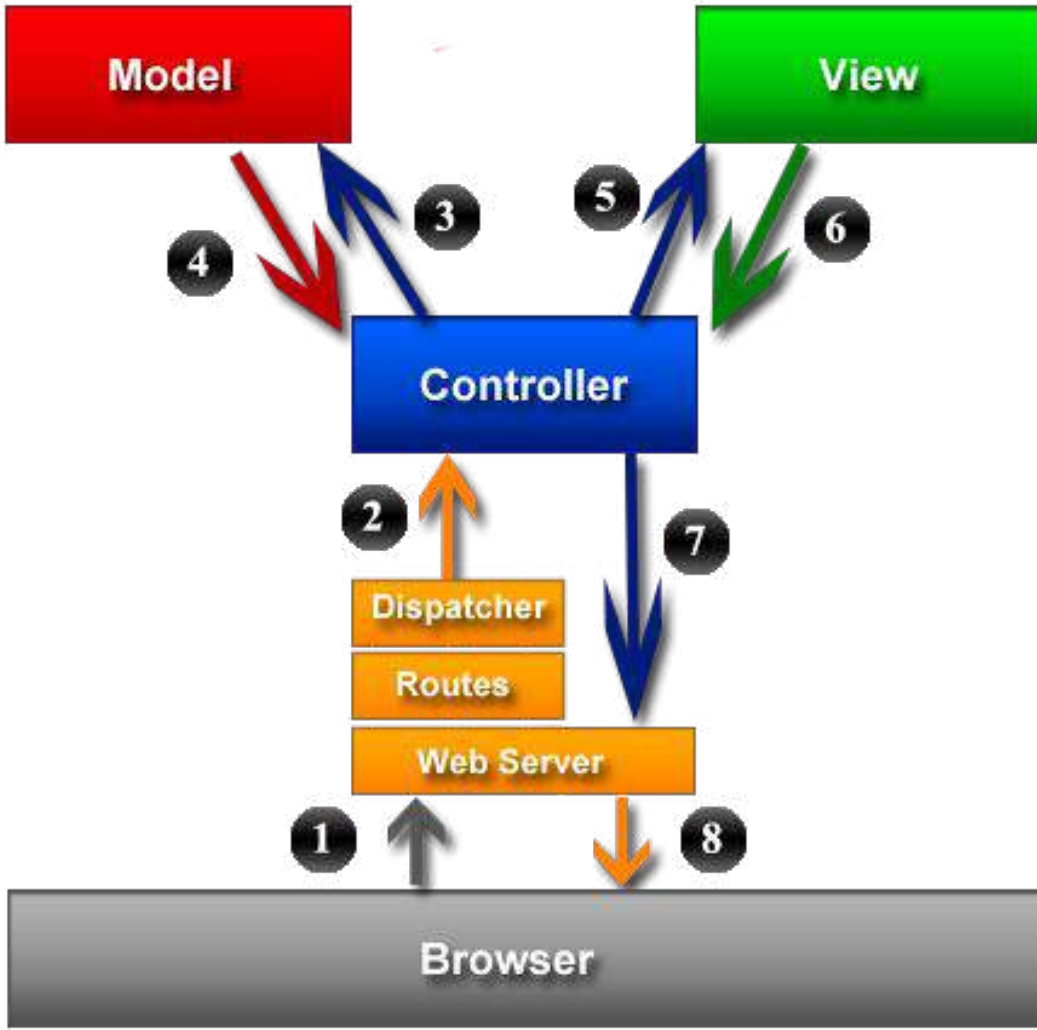
**RHODES**

Мы имеем локальный HTTP сервер с полноценной Ruby Виртуальной Машиной. Ruby код выполняется прямо на устройстве



- За веб-сервером работает машина Ruby в адаптированной реализации MRI.
- Код на Ruby при сборке компилируется в ISEQ.
- Принципиально возможно ( и реализовано ) разместить за веб-сервером любой другой рантайм ( Node, Lua, Python, C++ и т.д. ) .
- Ruby gems нельзя подключить “из коробки” - требуется адаптация в формат расширения RhoMobile.
- Из коробки есть набор базовых gems: json, bigdecimal, rexml, serialport, fileutils и др.





- CommonAPI: XML синтаксис для описания API расширений.
- Методы CommonAPI можно звать как на стороне Ruby контроллера, так и напрямую из WebView из JavaScript.
- Возможно вызывать API как синхронно, так и асинхронно.
- Нативный Extension manager позволяет гибко внедряться в функции платформы.

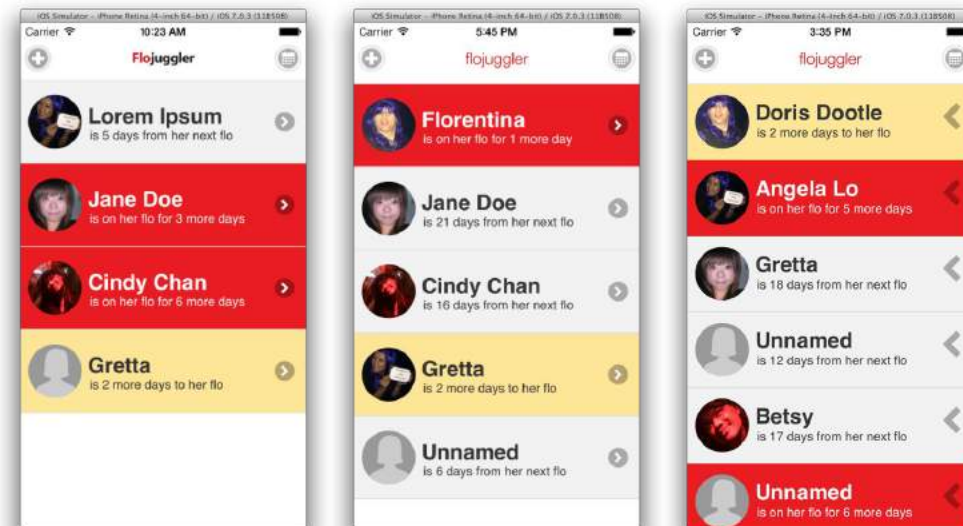
```
reapi > ext > system.xml
```

```
<METHOD name="zipFile">
  <DESC>Zip one file. Zip archive must placed to the folder where application can write
  <PARAMS>
    <PARAM name="localPathToZip" type="STRING">
      <DESC>The path to the specified zip. Application should have write permission
    </PARAM>
    <PARAM name="localPathToFile" type="STRING">
      <DESC>The path to the file to be zipped.</DESC>
    </PARAM>
    <PARAM name="password" type="STRING">
      <DESC>Password for the zip.</DESC>
      <CAN_BE_NIL/>
    </PARAM>
  </PARAMS>
</METHOD>
```

- Разделение UI части ( WebView ) и логики ( Ruby за веб сервером ) даёт возможность писать более нагруженные чем обычные JavaScript SPA приложения.
- Ресурсоёмкие части приложения можно вынести в нативный код.
- Код на Ruby предварительно компилируется в бинарный ISEQ, а не интерпретируется напрямую.
- Современные веб движки достаточно быстрые.



- Для UI можно выбрать любой доступный frontend фреймворк: bootstrap, vue, framework7 и пр.
- С помощью CSS можно добиться близкого к нативному look&feel приложения.
- Этого как правило достаточно для бизнес приложений.
- Для приложений, требующих использования всех возможностей визуальных компонентов целевой платформы, лучше использовать нативную архитектуру.



# Приложения работают на всех актуальных ОС

- Android ( 4.4.x - 11 ), с поддержкой ТСД различных вендоров, встраиваемых и пользовательских устройств.
- WinCE 6 & 7, собственный движок HTML5.
- Аврора / Sailfish.
- iOS всех версий.
- настольный Windows, от XP до 11.
- Linux, включая отечественные дистрибутивы Alt, РЕД, ROSA, Astra.

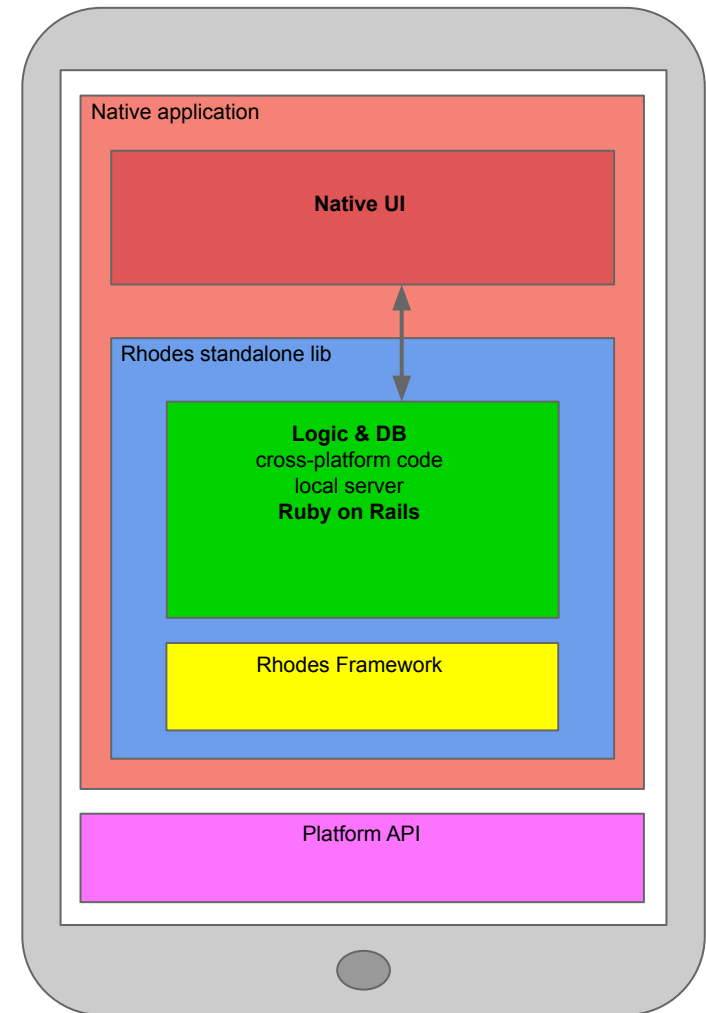


```
def do_take
  Rho::Barcode.take({}, url_for(:action => :take_callback))
end

def take_callback
  Rho::WebView.navigate url_for :action => :take_picture, :query => @params
end

<div class="result">
  <div class="alert"><%= @result %></div>
</div>
```

- Возможно использовать платформу не как фреймворк приложения, а просто как библиотеку.
- В режиме библиотеке можно подключить RhoMobile к нативному приложению как зависимость и использовать кроссплатформенную функциональность: исполнять код Ruby, вызывать API, использовать ORM и синхронизацию.
- Библиотека собирается в подключаемый артефакт для целевой платформы: Android AAR, iOS Framework.



## Внутренние:

- Ruby (MRI)
- SQLite
- CURL
- OpenSSL
- Node.JS ( при использовании Node бэкенда )

## Внешние:

- QT
- Redis ( для RhoConnect)
- Eclipse
- VS Code



# Особенности, связанные с применением в бизнес приложениях

- Защищенное хранилище - шифрование локальной БД.
- Шифрование исходного кода.
- Защищенный транспорт - продвинутый TLS с двусторонней аутентификацией.
- SSL pinning.
- Контроль целостности кода и данных.
- противодействие попыткам взлома и недобросовестного использования.



- Работа офлайн - данные можно хранить локально, и синхронизировать при появлении связи.
- Модификация серверного контента на лету.
- Бесшовная интеграция с помощью встроенного ORM и интеграционного сервиса RhoConnect.



{ REST }



## Унифицированное vendor-agnostic API для:

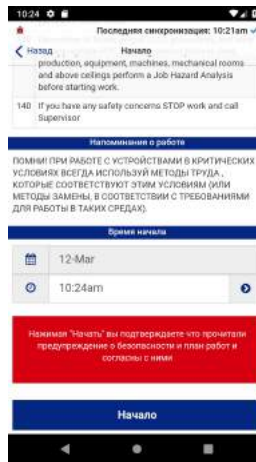
- RFID.
- Сканирование штрих и QR кодов.
- Bluetooth.
- Аппаратные кнопки и клавиатура.
- USB.
- Камера.
- Доступ к файловой системе.
- Запись и воспроизведение аудио/видео.
- Работа с адресной книгой и календарем.
- Push уведомления.
- Аппаратные сенсоры ( акселерометр, магнитометр и т.п. ).
- Геолокация.





## Сферы:

- мобильный сервис
- банкинг
- страхование
- энергетика
- промышленный браузер
- ритейл
- логистика



## Практика

напишем простое приложение (без синхронизации) на RhoMobile и запустим его на Android

Хост платформа: Windows 11

Целевая платформа: Android

```
PS C:\rho> ruby -v
ruby 2.7.2p137 (2020-10-01 revision 5445e04352) [x64-mingw32]
PS C:\rho> gem install .\rhodes-7.5.0.gem | Out-Null
PS C:\rho> rhodes-setup
We will ask you a few questions below about your dev environment.

JDK Path (blank to skip) (C:/Program Files/OpenJDK/openjdk-8u292-b10):
Android SDK path (blank to skip) (c:/android/sdk):
Android NDK path (blank to skip) (c:/android/android-ndk-r20): C:\android\sdk\ndk\23.1.7779620
```

Предустановки:

Ruby 2.7

Android SDK + NDK + Build tools

QT 5.13 ( для эмулятора требуется установка \$QTDIR )

## Создадим простое приложение

```
PS C:\rho> rhodes app helloworldapp
Generating with app generator:
C:/Ruby27-x64/lib/ruby/gems/2.7.0/gems/uuid-2.3.7/lib/uuid.rb:261: warning: Win32API
[ADDED] helloworldapp/rhoconfig.txt
[ADDED] helloworldapp/build.yml
[ADDED] helloworldapp/Podfile.example
[ADDED] helloworldapp/.gitignore
[ADDED] helloworldapp/app/application.rb
[ADDED] helloworldapp/app/index.erb
[ADDED] helloworldapp/app/layout.erb
[ADDED] helloworldapp/app/loading.html
[ADDED] helloworldapp/app/Settings/controller.rb
[ADDED] helloworldapp/app/Settings/err_sync.erb
[ADDED] helloworldapp/app/Settings/home.erb
[ADDED] helloworldapp/app/Settings/index.erb
[ADDED] helloworldapp/app/Settings/login.erb
[ADDED] helloworldapp/app/Settings/reset.erb
[ADDED] helloworldapp/app/Settings/wait.erb
[ADDED] helloworldapp/Rakefile
[ADDED] helloworldapp/app/loading.png
[ADDED] helloworldapp/AndroidManifest.erb
[ADDED] helloworldapp/app/helpers
[ADDED] helloworldapp/icon
[ADDED] helloworldapp/resources
[ADDED] helloworldapp/public
PS C:\rho>
```

```
PS C:\rho> cd .\helloworldapp\  
PS C:\rho\helloworldapp> rhodes model Hello name  
Generating with model generator:  
  [ADDED] app/Hello/index.erb  
  [ADDED] app/Hello/edit.erb  
  [ADDED] app/Hello/new.erb  
  [ADDED] app/Hello/show.erb  
  [ADDED] app/Hello/hello_controller.rb  
  [ADDED] app/Hello/hello.rb  
  [ADDED] app/test/hello_spec.rb  
PS C:\rho\helloworldapp>
```

## HELLOWORLDAPP

## app

## Hello

- edit.erb
- hello\_controller.rb
- hello.rb
- index.erb
- new.erb
- show.erb

## helpers

## Settings

## test

- application.rb

- index.erb

- layout.erb

- loading.html

- loading.png

## icon

## public

## resources

- .gitignore

- AndroidManifest.erb

- ! build.yml

- Podfile.example

- Rakefile

- rhoconfig.txt

Файлы модели:  
Код модели, контроллера и шаблонов представлений для CRUD методов

Точка входа в приложение

Настройки сборки

Рантайм настройки

## Внесём небольшие правки в код

```
app > Hello > show.erb
1 <div class="border d-flex justify-content-between">
2   <a href="<%= url_for :action => :index %>" class="m-1 btn btn-secondary btn-sm">Back</a>
3
4   <h3><%= @hello.name %></h3>
5
6   <a href="<%= url_for :action => :edit, :id => @hello.object %>" class="m-1 btn btn-primary btn-sm">Edit</a>
7
8 </div>
9
10 <div class="container-fluid p-1">
11
12   <ul class="list-group">
13
14     <li class="list-group-item">
15       <div class="row">
16         <div class="col-6">
17           <span><strong>Name</strong></span>
18         </div>
19         <div class="col-6">Hello, <%= @hello.name %>! This is RhoMobile application!</div>
20       </div>
21     </li>
22
23   </ul>
24
25 </div>
26
27
```

Слегка модифицируем шаблонный код

```
≡ rhoconfig.txt X
```

```
≡ rhoconfig.txt
```

```
1 # startup page for your application
2 start_path = '/app/Hello'
```

При запуске приложение запустит контроллер нашей модели, вызвав по умолчанию метод **index**

```
hello_controller.rb X
```

```
app > Hello > hello_controller.rb
```

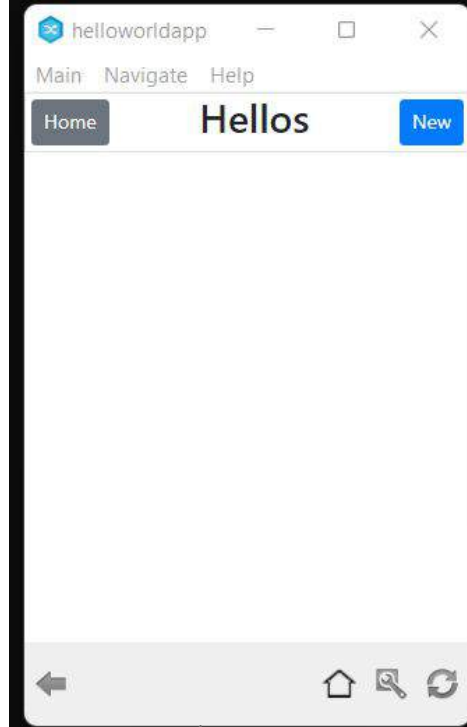
```
1 require 'rho/rhocontroller'
2 require 'helpers/browser_helper'
3
4 class HelloController < Rho::RhoController
5   include BrowserHelper
6
7   # GET /Hello
8   def index
9     @hellos = Hello.find(:all)
10    render :back => '/app'
11  end
```



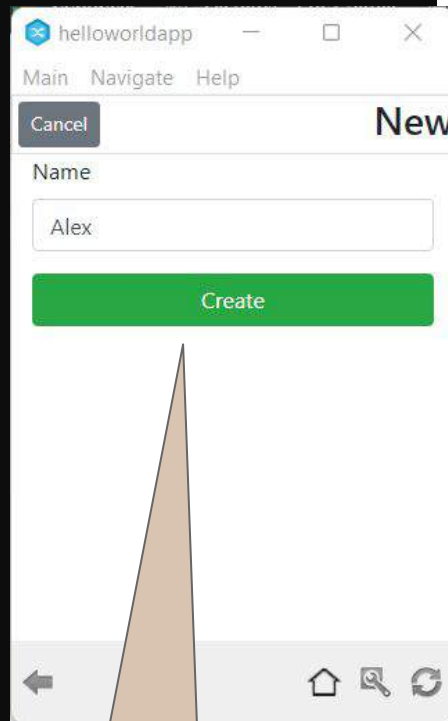
# Проверим логику приложения на встроенном эмуляторе

Команда запуска на эмуляторе

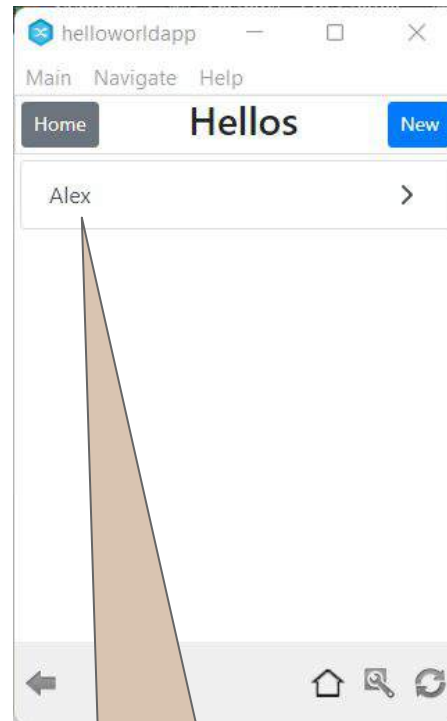
```
PS C:\rho\helloworldapp> rake run:android:rhosimulator
PS C:\rho\helloworldapp>
```



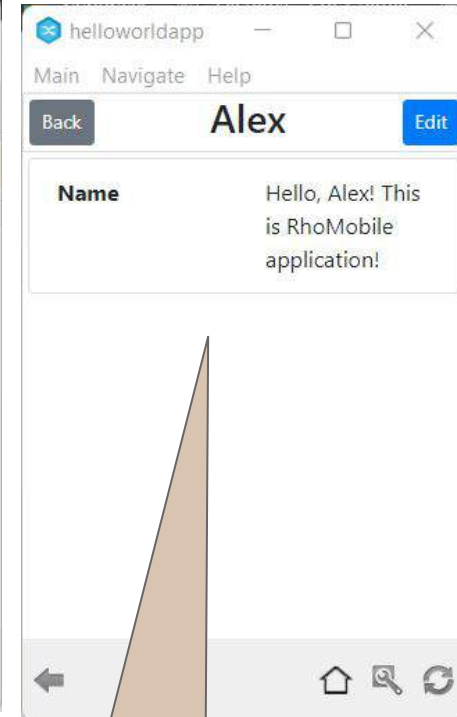
Стартовая страница



Создаем новый объект в модели

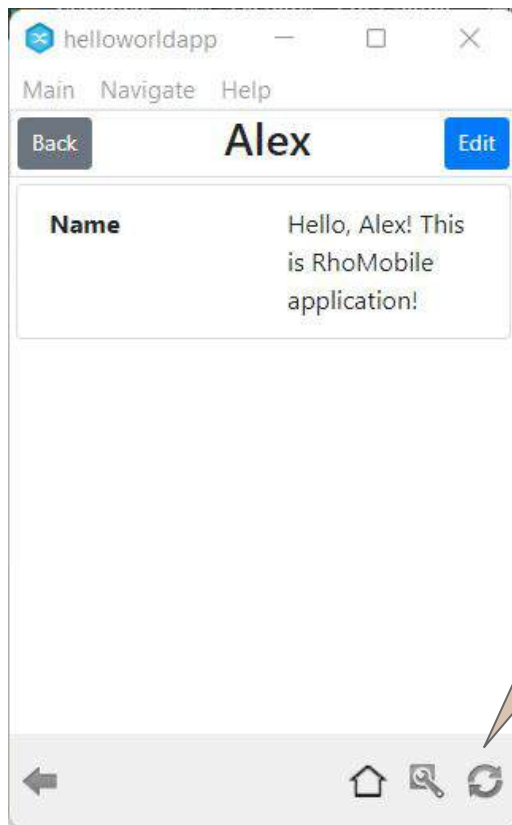


Видим его теперь в списке на главной странице

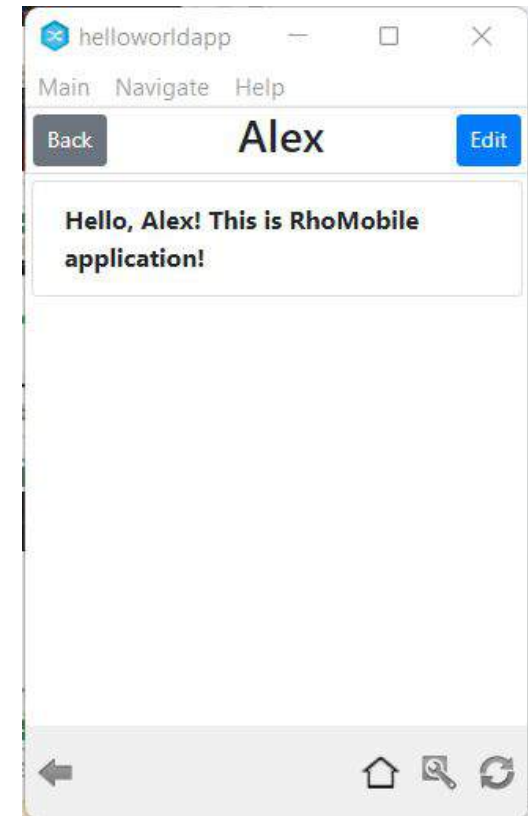


Просмотр объекта - метод **show** в контроллере

```
12 <ul class="list-group">
13 |
14 |   <li class="list-group-item">
15 |     <div class="row">
16 |       <div class="col-12"><b>Hello, <%= @hello.name %>! This is RhoMobile application!</b></div>
17 |     </div>
18 |   </li>
19 |
20 </ul>
```



Нажимаем кнопку обновления страницы - эмулятор сразу же отобразит изменения



Соберём и запустим приложение на настоящем устройстве

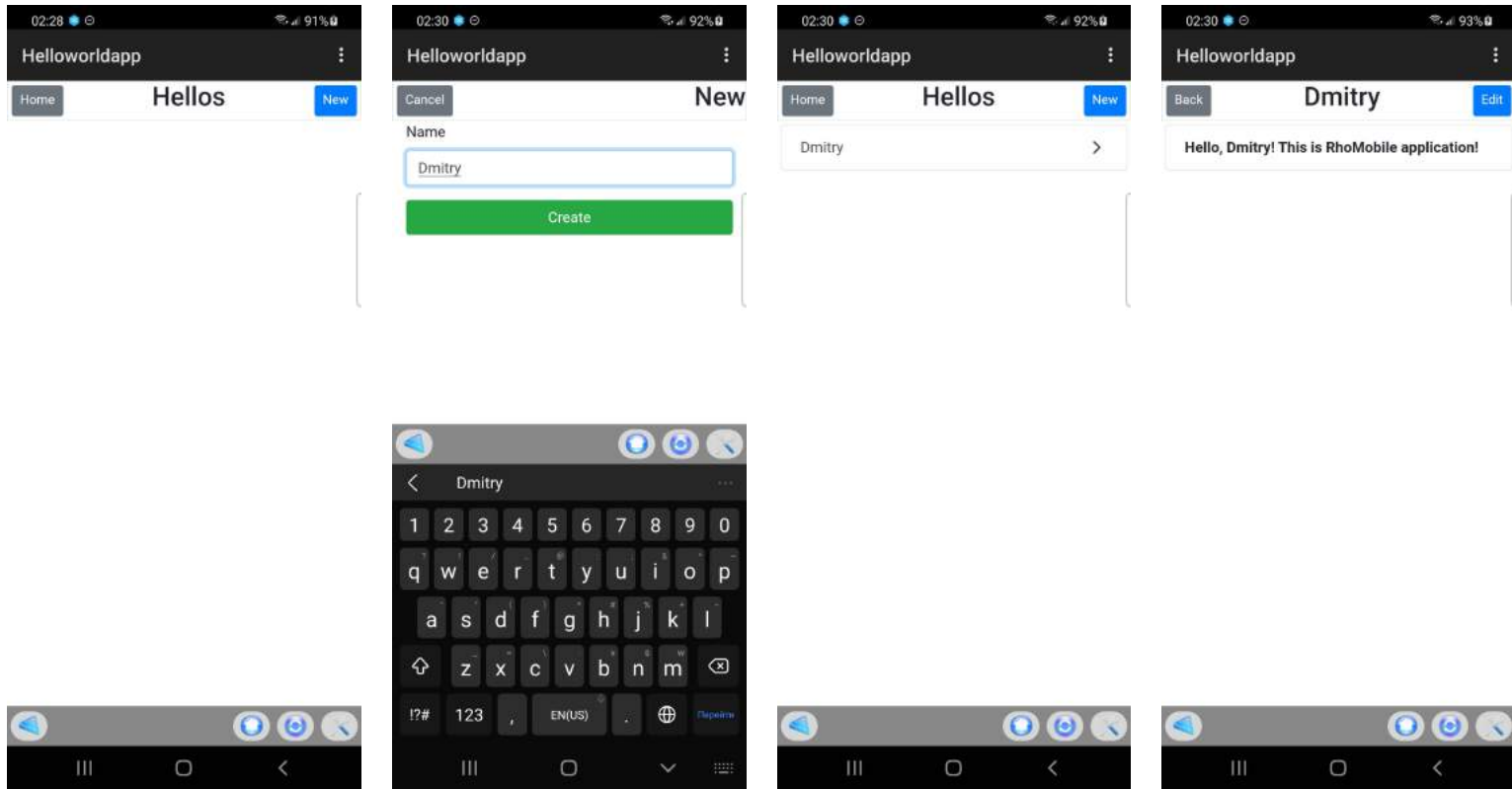
```
PS C:\rho\helloworldapp> rake run:android:device
```



A FEW  
MOMENTS LATER \*

\* Для Android - несколько минут для холодной сборки. Зависит от количества целевых ABI

# Приложение запущено на Android



```

    ✓ HELLOWORLDAPP
      > app
      ✓ bin
        ✓ target\android
          > debug
            ≡ app_info.txt
            ≡ helloworldapp-debug.apk
            ≡ helloworldapp-debug.apk.idsig
  
```

Артефакт сборки

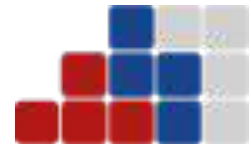
Блюм. А.  
Вовченко Е.  
Тарасов В.  
Солдатенков Д.  
Епифанов А.  
Рыбас К.  
Мусулайнен В.  
Прохоров Д.  
Вишневский А.  
Москальчук Д.  
Володарский С.  
Тимашев А.  
Тихвинский А.  
Малинин М.  
Бурджес Л.  
Шабалев Г.  
Шишкин А.

Сообщество разработчиков

Другие сотрудники Motorola  
Solutions и Zebra  
Technologies.



- Работаем с 2015 года
- Технологическая компания, основанная инженерами ПО с обширным опытом в различных направлениях разработки
- Создаём инструменты для разработки бизнес приложений, разрабатываем на заказ, консультируем
- Разрабатываем сложные и нестандартные решения, в основном в сферах мобильного и системного ПО. Проводим R&D
- Состоим в АРПП “Отечественный Софт”
- Работаем по всему миру. Разработка в основном находится в Санкт-Петербурге



## RhoMobile / Тау Платформа:

<https://tau-platform.com/ru/developers/downloads/>

<https://tau-platform.com/ru/products/rhobile/>

<https://tau-platform.com/ru/products/rhoconnect/>

## Репозитории

<https://github.com/rhobile/rhodes>

<https://github.com/rhobile/rhoconnect>

## RhoBrowser / Тау Браузер:

<https://tau-platform.com/ru/products/rhobrowser/>

[https://rhobrowser-demo-static.s3.us-east-1.amazonaws.com/rhobrowser\\_demo\\_signed\\_master\\_JDK1.8.apk](https://rhobrowser-demo-static.s3.us-east-1.amazonaws.com/rhobrowser_demo_signed_master_JDK1.8.apk)

## Документация:

<http://docs.tau-platform.com/en/7.4/home>

<http://docs.tau-platform.com/en/7.4/guide/apisummary>

<http://docs.tau-platform.com/en/7.4/guide/welcome>

## Примеры приложений :

<https://github.com/tauplatform/rhodes-system-api-samples>

<https://github.com/tauplatform/kitchensinkRuby>

<https://github.com/tauplatform/inventoryDemo-mobileApp>

<https://github.com/tauplatform/kitchensinkJS>

<https://github.com/tauplatform/universal-push-example>



## Контакты:

[info@tau-platform.com](mailto:info@tau-platform.com)

<http://alex.epifanoff.name>