

PWA вместо App Store

Опыт замены нативного iOS-приложения и технические ограничения

Константин Шкурко

РСХБ.Цифра (РСХБ-Автоматизация)



Константин Шкурко

Разработчик мобильных приложений с 10+ годами опыта. Разрабатываю под iOS и Android (нативно и кроссплатформенно). Умею в веб-разработку и бэк на Python и Golang. Имею черный пояс по пальцевому кунг-фу в области мануального и не только тестирования.

Что же делать?



PWA ≠ mobile web

Установка

Иконка

Offline

Push

- 1999** Microsoft .hta (HTML App)
- 2007** Стив Джобс Web Clip (Ajax)
- 2015** Google PWA (дизайнер Фрэнсис Бэрримэн и инженер Алекс Расселл)



2015

- ✓ Manifest
- ✓ Service worker
- ✓ Push-notifications



2018

- ✓ ✗ Manifest
- ✓ Service worker
- ✗ Push-notifications

manifest



HTTPS

service worker

manifest.json



Основной
документ PWA

Как браузер понимает что это PWA?

```
<link rel="manifest" href="/manifest.json">
```

```
{
  "name": "My App",
  "short_name": "My App",
  "start_url": "/",
  "display": "standalone",
  "icons": [
    {
      "src": "/icons/icon-192.png",
      "sizes": "192x192",
      "type": "image/png"
    },
    {
      "src": "/icons/icon-512.png",
      "sizes": "512x512",
      "type": "image/png"
    }
  ]
}
```

Минимальная версия
манифеста





Обычный manifest.json + meta-тэги

```
<link rel="manifest" href="/manifest.json">  
<link rel="apple-touch-icon" href="/icons/apple-touch-icon.png">  
<meta name="apple-mobile-web-app-capable" content="yes">  
<meta name="apple-mobile-web-app-title" content="My App">
```

name и short_name



short_name

Экран
установки
PWA

name

Иконки приложения

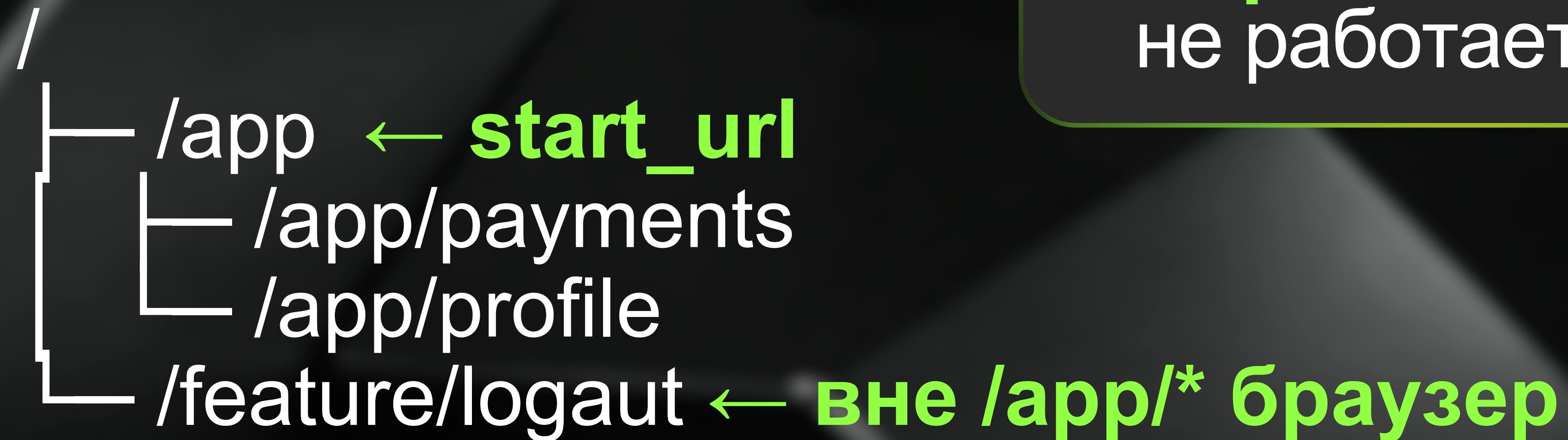
48
x
48

192
x
192

512
x
512

start_url и scope

Возврат из браузера
обратно в scope
не работает в iOS



display

- ✓ browser
- ✓ minimal-ui
- ✓ standalone
- ✓ fullscreen



- ✓ browser
- ✗ minimal-ui
- ✓ standalone
- ✗ fullscreen



theme_color и background_color



работают не стабильно и
перекрываются meta-тэгами

```
<meta name="theme-color" content="#0f172a">  
<meta name="apple-mobile-web-app-status-bar-style"  
content="black-translucent">
```

Все поля `manifest.Json` и их поддержка



**Почему же Apple использует
архаичные meta-тэги и игнорит
ключи в manifest.json?**



Service Worker

сеть

кэш

offline

пуши



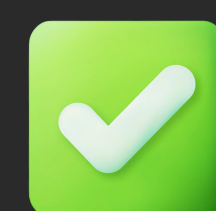
Жизненный цикл Service Worker

```
if ('serviceWorker' in navigator) {  
  window.addEventListener('load', async () => {  
    try {  
      const registration = await navigator.serviceWorker.register('/sw.js', {  
        scope: '/',  
      });  
  
      console.log('SW registered:', registration.scope);  
    } catch (error) {  
      console.error('SW registration failed:', error);  
    }  
  });  
}
```

**register > parsed
Service Worker**

Жизненный цикл Service Worker

Install - идеальное место для первого оффлайн запуска



Из кэша

- ИКОНКИ
- js
- css
- html



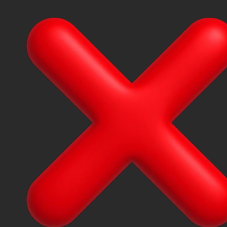
- ЧИСТКА кэша
- миграции indexedDB
- бизнес логика

Жизненный цикл Service Worker

waiting - это ещё не активный воркер



показать UI
об обновлении, если
нужно обновить SW



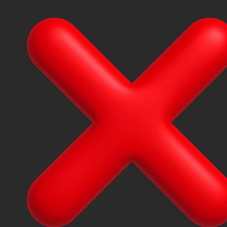
тихое обновление
SW (моргание экрана
для пользователя)

Жизненный цикл Service Worker

activating - идеальное место для чистки и миграций



- удаление старого кэша
- миграции данных



- держат длинные операции

Жизненный цикл Service Worker

activated - воркер становится текущим «контролёром» для страниц в своём scope и начинает получать функциональные события: `fetch`, `push`, `sync`, `message`



определение стратегии кэширования

`cache-first`, `network-first`, `stale-while-revalidate`, `offline fallback`



кэшировать все подряд без стратегии

Service Worker

Cache-first - сначала берём ответ из кэша и только если его нет — идём в сеть

Network-first - сначала пробуем сеть и кладём успешный ответ в кэш, сеть недоступна - отдаём последний кэш

Stale-while-revalidate - сразу отдаём кэш, параллельно делаем сетевой запрос и обновляем данные

Offline fallback - сначала network-first или cache-first, а если сеть и кэш не дали ответа, подсовываем заранее подготовленную заглушку

Предназначение service worker

Скорость

Устойчивость

Offline

Уведомления

Проблемы `service worker`

Залипание
кэша

Битые
offline-сценарии

Сложность
дебага

Безопасность

HTTPS



- WebAuth
- Push
- Геолокация
- Bluetooth

с учетом ограничений на iOS

Secure context

localhost — поблажка для разработки

install-prompt



Manifest
HTTPS
Показ prompt без
действий
пользователя



Вобще не отобразится,
пользователь должен сам
установить приложение
Без SW, manifest и HTTPS
иконка - просто закладка
браузера

**Без HTTPS ваш кэш, куки, токены,
localStorage занимают очередь
для атаки злоумышленников**

Доступность API на платформах

	Android	iOS
Камера/микрофон	✓	✓
Геолокация	✓	✓ ✗
Оффлайн	✓	✓
Хранение	✓	✓ ✗
Web Push	✓	✓
Background Sync	✓	✗
Bluetooth/NFC	✓	✗
Deep link	✓	✓ ✗

Биометрия WebAuthn

Приложение
само
предложит



Только если
пользователь
совершит
действие



Хранение данных

Всё доступно,
можно
запрашивать
устойчивое
хранение



Local storage, indexedDB,
кэш, куки, но с
ограничения из-за
Intelligent Tracking
Prevention в Safari



Web Push

Отлично
работают
уже давно



Появились только с версии ios 16.4 и работают только с явного согласия пользователя. Ограничения по количеству

Нет возможности проверить есть ли разрешение пользователя. Re-subscribe на каждом старте. После долгого неиспользования пропадает разрешение



Deep links

Работают без
нареканий



Доступны только из пушей

В PWA не перейти по
ссылке из СМС или по
ссылке с сайта из-за
контейнеров



Web Share

Можно делиться
из PWA

Можно
зарегистрировать
PWA как target
share



Можно делиться
из PWA

Нельзя
зарегистрировать
PWA как target
share



manifest

HTTPS

SW

PWA

СПАСИБО ЗА ВНИМАНИЕ!



Константин Шкурко
РСХБ.Цифра (РСХБ-Автоматизация)