

ТЕХНИЧЕСКИЙ РАЗБОР

React Native vs Flutter: как мы выбираем мобильный стек

Прототип-бенчмарк, New Architecture и Impeller, OTA-контур
и CI/CD — наша методология выбора



Ай-Ти Фреш

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

Суть проблемы

Заказчик тянет два нативных приложения (Swift + Kotlin): каждая фича пишется дважды, релизы расходятся на недели, мобильная команда дублируется. Мы решаем задачу выбора единого кроссплатформенного стека — React Native 0.86 или Flutter 3.44 — через 6-недельную прототип-фазу с бенчмарками, чтобы решение опиралось на замеры и стоимость владения, а не на моду.

Почему это важно бизнесу

- Две кодовые базы = двойной фонд оплаты труда мобильной команды и задержка фич на 2-4 недели между платформами
- Неверный выбор стека всплывает через год: переписывание приложения стоит дороже, чем вся прототип-фаза
- OTA-хотфиксы сокращают починку критичного бага с дней (ревью стор) до часов — это прямые потери выручки
- Стоимость найма под стек: JS/TS-разработчиков на рынке в разы больше, чем Dart, — это скорость роста команды

Ключевые параметры реализации

0.86

React Native stable; New Architecture по умолчанию с 0.76, legacy-архитектура заморожена с 0.80

по докам reactnative.dev

3.44

Flutter stable с Dart 3.12; рендер Impeller: iOS без Skia, Android — Vulkan с API 29+, на старых устройствах fallback OpenGL ES

по докам docs.flutter.dev

60 fps

порог плавности ленты 500+ карточек: FlashList v2 (без estimatedItemSize) + Reanimated 4 на UI-потоке

наш стандарт

≤ 2 с

порог холодного старта на среднем Android; меряем только release-сборку, 10 прогонов, усреднение

наш стандарт

-75%

снижение размера OTA-обновления за счёт диффов Hermes-байткода в EAS Update (по умолчанию с SDK 56)

по докам expo.dev

31.03.2025

дата отключения App Center CodePush — OTA-контур строим только на живых поддерживаемых сервисах

learn.microsoft.com



Прототип-бенчмарк: один экран на двух стеках

Что настраиваем

Лента 500+ карточек с параллаксом; iPhone, средний и бюджетный Android

Как мы это делаем

- 1 RN: включаем New Architecture (Fabric + TurboModules), список — FlashList v2, анимации — Reanimated 4 + react-native-worklets, babel-плагин react-native-worklets/plugin
- 2 Flutter 3.44: CustomScrollView + SliverChildBuilderDelegate, состояние Riverpod, изображения cached_network_image; рендер Impeller (Metal/Vulkan)
- 3 Меряем только release/profile-сборки: холодный старт, RAM, % jank-кадров; Android — Perfetto, iOS — Xcode Instruments, Flutter — DevTools Performance
- 4 10 прогонов на каждой из 3 моделей устройств, усредняем; пороги приёмки: старт ≤ 2 с, скролл 60 fps, jank $\leq 2\%$

РЕЗУЛЬТАТ

Решение принимается по цифрам, а не по симпатиям команды: за 6 недель получаем сравнимые метрики и реальную трудоёмкость обоих стеков на самом тяжёлом экране продукта.

КЛЮЧЕВОЙ НЮАНС

Debug-сборки лгут: Flutter в debug работает через JIT, RN — с dev-оверхедом. Любой бенчмарк — только release/profile, иначе один из стеков «медленнее» в разы на ровном месте.

ОТА-контур хотфиксов: EAS Update и Shorebird

Что настраиваем

Выкатка исправлений минуя ревью сторов; ветки RN и Flutter

Как мы это делаем

- 1 RN: expo-updates + EAS Update, каналы preview/production, постепенный rollout и rollback; диффы Hermes-байткода режут размер обновления до –75%
- 2 App Center CodePush отключён 31.03.2025, репозиторий code-push-server в архиве с 20.05.2025 — новые контуры на нём не строим
- 3 Flutter: Shorebird — команды shorebird release / shorebird patch; патчится только Dart-код, нативные изменения требуют полного релиза через стор
- 4 Политика: ОТА только для JS/Dart-хотфиксов в рамках правил сторов; новые нативные модули и permissions — обычный релиз с ревью

РЕЗУЛЬТАТ

Критичный баг чинится за часы вместо дней ожидания ревью; каналы и rollback позволяют откатить неудачный патч без пересборки приложения.

КЛЮЧЕВОЙ НЮАНС

ОТА не заменяет релизный цикл: любое изменение нативного кода всё равно едет через стор — закладываем это в план релизов и не обещаем бизнесу «всё за час».

CI/CD: GitHub Actions + Fastlane

Что настраиваем

Сборка и выкладка обеих платформ; раннеры ubuntu-latest и macos-14

Как мы это делаем

- 1 Стадия lint-and-test: setup-node (Node 20, cache yarn), yarn lint + yarn test --coverage; сборки стартуют только после зелёных тестов
- 2 Android: fastlane, gradle task bundle → upload_to_play_store, сначала track internal, продвижение в production вручную
- 3 iOS: macos-14, pod install, build_app + upload_to_app_store через App Store Connect API key; подписи — fastlane match из приватного git-репо
- 4 Секреты (PLAY_STORE_JSON_KEY, ASC_KEY) — только в GitHub Secrets; build number инкрементируется lane-ом, не руками

РЕЗУЛЬТАТ

Релиз обеих платформ — одна кнопка; выкладка store-обновления сокращается с дней ручной сборки до часов конвейера, воспроизводимо и без «магии на ноутбуке».

КЛЮЧЕВОЙ НЮАНС

Минуты macos-раннеров в разы дороже: линт и тесты гоняем на ubuntu, macos поднимаем только на финальную iOS-сборку — счёт за CI падает кратно.

Подводные камни

✗ Reanimated 4 на старой архитектуре

4.x работает только с New Architecture; на legacy-проектах держим ветку 3.x, при апгрейде меняем babel-плагин на react-native-worklets/plugin.

✗ estimatedItemSize в FlashList v2

В v2 проп deprecated — список сам измеряет элементы; v2 не запускается на old arch, для старых проектов остаётся ветка v1.

✗ Разные minor-версии worklets

reanimated, react-native-worklets и babel-плагин должны совпадать по minor-версии, иначе undefined behavior и падения на старте.

✗ Ставка на архивный CodePush

App Center закрыт 31.03.2025, code-push-server архивирован; новые OTA-контурсы строим на EAS Update (RN) или Shorebird (Flutter).

✗ Бенчмарк в debug-сборке

Flutter debug = JIT, RN dev-mode = лишний оверхед; сравнивать стеки можно только на release/profile-сборках, иначе выводы ложные.

✗ Фоновая геолокация на Android 14+

Нужен foreground service type location и учёт Doze; проверяем на реальных устройствах с включённым battery optimization — эмулятор это не ловит.

✗ Impeller на устройствах без Vulkan

На Android ниже API 29 или без поддержки Vulkan движок уходит в fallback OpenGL ES — графические артефакты ловим только тестом на реальном бюджетном парке устройств.

✗ Игнорирование стоимости найма

Dart-разработчиков на рынке в разы меньше, чем JS/TS; если команда растёт, стоимость стека — это ещё и стоимость людей под него.

Как правильно

МИНИМУМ

- Зафиксировать версии: RN 0.8x с New Architecture или Flutter 3.4x, без legacy
- Все замеры и сравнения — только release/profile-сборки на реальных устройствах
- Crash-репортинг (Crashlytics/Sentry) с первого релиза, цель crash-free $\geq 99\%$

НОРМАЛЬНО

- CI/CD: GitHub Actions + Fastlane, автосборка обеих платформ из одной ветки
- OTA-контур: EAS Update (RN) или Shorebird (Flutter) с каналами и rollback
- Матрица тест-устройств: флагман + средний + бюджетный Android, iPhone

ХОРОШО

- Прототип-фаза 4–6 недель: самый тяжёлый экран на обоих стеках + бенчмарки
- Монорепозиторий с веб-фронтом: шаринг типов и бизнес-логики до 35% (для RN)
- Перфоманс-гейты в CI: cold start, % jank и размер бандла на каждый PR



ПРОВЕРЬТЕ У СЕБЯ

Чек-лист самопроверки

- Включена ли New Architecture (Fabric + TurboModules) и убран ли legacy-код?
- Замеры старта и скролла делались на release-сборке и бюджетном устройстве?
- Есть ли OTA-канал хотфиксов с постепенным rollout и кнопкой rollback?
- Не завязан ли OTA-контур на закрытый App Center CodePush?
- Совпадают ли minor-версии reanimated / worklets / babel-плагина?
- Проверена ли фоновая геолокация на Android 14+ с battery optimization?
- Подписи iOS хранятся в fastlane match, а не на ноутбуке разработчика?
- Оценён ли рынок найма под выбранный стек на горизонт роста команды?

Если хотя бы на два вопроса ответ «нет» или «не знаю» — тема требует внимания.



Как поможет ITFresh

ITFresh — ИТ-аутсорсинг для юридических лиц до 50 рабочих мест в Москве и области. 15+ лет практики, собственная инфраструктура в дата-центре МТС (8 серверов Dell Xeon Platinum).

- Проводим прототип-фазу: один тяжёлый экран на RN и Flutter с бенчмарками и отчётом для руководства
- Разворачиваем CI/CD под ключ: GitHub Actions + Fastlane + подписи match, релиз одной кнопкой
- Строим OTA-контур хотфиксов на EAS Update или Shorebird с каналами и rollback
- Аудитируем существующее приложение: производительность, размер бандла, миграция на New Architecture

15+

лет в ИТ-поддержке

50

рабочих мест — наш профиль

МТС

дата-центр, Москва

КОНТАКТЫ

Обсудить вашу задачу

Сайт **itfresh.ru**

Телефон **+7 903 729-62-41**

Telegram **@ITfresh_Boss**

Бесплатно посмотрим вашу инфраструктуру по этому чек-листу и скажем, где тонко — без обязательств.



itfresh.ru

Техническая база

- 01** [React Native — Versions / New Architecture \(reactnative.dev — 0.86\)](#)
- 02** [Flutter release notes / Impeller \(docs.flutter.dev — 3.44\)](#)
- 03** [Dart changelog \(dart.dev — 3.12\)](#)
- 04** [Reanimated — Migration from 3.x / Compatibility \(docs.swmansion.com — 4.x\)](#)
- 05** [FlashList — What's new in v2 / Migration \(shopify.github.io — 2.x\)](#)
- 06** [Visual Studio App Center Retirement \(learn.microsoft.com — 2025\)](#)
- 07** [EAS Update / expo-updates \(docs.expo.dev — SDK 56\)](#)
- 08** [Shorebird Code Push — Overview \(docs.shorebird.dev — 2026\)](#)

Основано на официальной документации продуктов и нашей практике внедрения.

