Engineer

Застосунок

програма, якою люди користуються, щоб виконати завдання

Що це таке

Застосунок — це програма, яку людина відкриває, щоб безпосередньо в ній працювати: у неї є значок для запуску, власне вікно чи екран і зрозуміле користувачеві призначення, як-от надіслати повідомлення, переглянути розклад чи подати звіт про витрати. Англійське app — скорочення від application. Українською усталився термін «застосунок», у розмові часто кажуть «додаток», а у формальних текстах трапляється «прикладна програма». Розробники кажуть «клієнт», коли застосунок є видимим кінцем сервісу, що працює деінде.

Застосунок належить до прикладного програмного забезпечення, на відміну від системного, яке забезпечує роботу самої машини. Глосарій NIST зводить кілька визначень і зазначає, що терміни application і software application часто вживають як синоніми; одне з визначень зараховує до застосунків також системи нарахування зарплати й бухгалтерського обліку. По інший бік межі лежить бібліотека: код, який викликають інші програми і який людина ніколи не відкриває сама.

Застосунки бувають трьох основних видів:

- **Нативні** пишуть для однієї платформи її власною мовою та інструментами, скажімо Swift і SwiftUI для Mac, і встановлюють із магазину або як підписаний файл.
- **Вебзастосунки** працюють у браузері, і відкривають їх за посиланням. Маніфест вебзастосунку, невеликий JSON-файл, що повідомляє браузеру назву, значки, початкову адресу й бажаний режим показу, дає змогу додати вебзастосунок на головний екран чи в Dock і відкривати його в окремому вікні. Service worker, скрипт, який браузер запускає між сторінкою та мережею, дає йому працювати без підключення. Таке поєднання зазвичай називають прогресивним вебзастосунком, або PWA.

- **Кросплатформні оболонки** загортають одну кодову базу для кількох операційних систем, часто вбудовуючи в застосунок рушій браузера.

Застосуйте поняття до рішення

Застосуйте «Що таке застосунок?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Швидкий сайт, який легко знайти в пошуку і який ваші працівники редагують самі, — без рахунку за кожну зміну в тексті.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Розробила 100 вебзастосунків на HTML/HTML5 — 100 вебзастосунків побудовано

Розробила 100 вебзастосунків на HTML/HTML5 Зв'язатися з нами →

Чому це важливо

Більшість проєктів, що мають завершитися застосунком, починаються з вибору виду, і цей вибір залежить переважно від користувачів і способу розповсюдження, а не від коду.

Вебзастосунок вимагає однієї кодової бази й доходить до кожного, хто має браузер, через посилання. Оновлення виходить разом із сервером: наступне перезавантаження сторінки його підхоплює, і між виправленням та людьми, які на нього чекають, немає перевірки в магазині. Платити доводиться глибиною. Браузер відкриває менше можливостей пристрою, ніж операційна система, а кожен браузер пропонує встановлення по-своєму. На iOS встановити сайт як застосунок ніколи не пропонують автоматично, тож команда, яка хоче бачити свій вебзастосунок на головних екранах, мусить сама пояснити людям, як це зробити.

Нативний застосунок отримує власні елементи керування платформи, її поведінку клавіатури, підтримку доступності, швидкодію й доступ до всього, що пропонує система. Він вимагає окремої кодової бази для кожної платформи, людей, які знають кожну з них, і процесу розповсюдження: перевірки в магазині або, поза ним, підписування коду, а на macOS ще й нотаризації, якої Apple починаючи з macOS 10.15 вимагає від ПЗ, що розповсюджується з Developer ID. Хтось також має підтримувати канал оновлень, доки застосунок існує.

Кросплатформна оболонка стоїть посередині: одна кодова база й застосунок, який можна встановити, ціною вбудованого середовища виконання та інтерфейсу, що схожий на платформу, але поводиться не

зовсім як вона.

Зазвичай усе вирішує те, де будуть користувачі, коли відкриють застосунок. Людина за столом із дротовим підключенням пробачає багато. Людина з телефоном і слабким сигналом не пробачає майже нічого, і для неї робота без мережі — вимога, а не прикраса. Друге вирішальне питання — хто випускатиме оновлення на третій рік, бо найдешевший у розробці застосунок може виявитися найдорожчим в утриманні.

Як це виглядає на практиці

На одній платформі багато користувачів — моряки й польові працівники, які працюють із телефонів і часто зовсім без зв'язку. Відповіддю став вебзастосунок, який можна встановити: повноцінна підтримка Progressive Web App дає йому значки для головного екрана в розмірах, яких вимагають платформи, заставку під час запуску й кольори теми. Service worker налаштовано через плагін next-pwa, кешування Workbox під час роботи віддає шрифти, зображення, аудіо, відео та ресурси з CDN за принципом cache-first разом із HTTP-заголовками керування кешем, а встановлення й роботу без мережі перевіряли через HTTPS на справжніх пристроях. Люди можуть встановити застосунок, користуватися ним далі, коли мережа зникає, а повторні відвідини отримують із кешу.

Протокол обміну повідомленнями, чий клієнт для macOS був кросплатформною оболонкою, пішов протилежним шляхом: нативний клієнт для macOS на Swift 6 і SwiftUI налічує 11 141 рядок у 53 файлах, а ядро протоколу на Rust вбудовано в нього як статичний архів, зібраний із закріпленої ревізії. Клієнт запускається як програма для Mac, використовує власні елементи керування платформи й не містить вбудованого браузера. Межі зробленого сторінка теж називає: до публічного випуску клієнт не доведено, нотаризованої збірки й каналу оновлень немає, а інтерфейс має одну мову.

Цей сайт теж можна встановити як застосунок, і сторінка про встановлення пояснює, як це зробити в кожному браузері: без магазину й облікового запису, без завантаження окремого пакета застосунку, лише значок і окреме вікно. Safari на iPhone пропонує це через «Поділитися». Firefox у Windows дає змогу додавати сайти, щоб відкривати їх у власних вікнах вебзастосунків, починаючи з версії 143 або з версії 150, якщо Firefox встановлено з Microsoft Store; Mozilla не пропонує цієї функції в macOS чи Linux. Там само сказано й

про обмеження: для роботи без мережі потрібен service worker, сайт не постачає виконуваного JavaScript і не реєструє service worker, тому встановленому застосунку потрібне підключення, як і звичайній вкладці.

Де виникають проблеми

Екран біліє, щойно зникає сигнал

Користувачі в польових умовах бачать біле вікно або сторінку браузера про відсутність мережі. Застосунок проєктували й тестували на швидкому дротовому з'єднанні, і кожен екран завантажує потрібне в момент відкриття. Визначте, які екрани мусять працювати без мережі, кешуйте їх разом із ресурсами через service worker і перевіряйте на справжніх пристроях із вимкненою мережею, а не в симуляції настільного браузера.

Вебзастосунок встановили й чекають від нього офлайн-копії

Людина додає сайт на головний екран, відкриває його в потягу й бачить ту саму помилку, що й у вкладці браузера. Маніфест робить сайт придатним до встановлення, а роботу без мережі дає тільки service worker, і це два окремі рішення. Або постачайте service worker, або прямо напишіть в інструкції зі встановлення, що застосунку потрібне підключення.

Новий випуск вийшов, а користувачі лишаються на старому

Після розгортання частина людей кілька днів працює з кодом минулого тижня. Service worker віддає сторінки застосунку за принципом cache-first, і нову версію ніщо не позначає, або новий worker чекає, доки закриються всі вкладки, якими керує старий. Залиште cache-first для ресурсів, однакових для кожного запиту, як-от шрифти, медіа й файли з версією в назві, документи беріть спершу з мережі й показуйте пропозицію перезавантажити, коли нова версія готова.

Оболонка на комп'ютері здається чужою

Користувачі Mac скаржаться на меню, комбінації клавіш і виділення тексту, а застосунок споживає пам'ять, як браузер, бо всередині в нього браузер і є. Вбудоване середовище виконання імітує платформу замість того, щоб нею користуватися. Там, де досвід на комп'ютері і є продуктом, порівняйте нативний клієнт з оболонкою, а спільну логіку тримайте в бібліотеці-ядрі, яку всі клієнти викликають через стабільний інтерфейс.

Розповсюдження відкладають на кінець

Застосунок працює на машині кожного розробника, а перший клієнт на macOS бачить замість нього попередження Gatekeeper. Виправлення виходить, а половина користувачів і далі сидить на зламаній версії. Підписування, нотаризацію й оновлення вважали дрібними пакувальними справами на останній тиждень. Вбудуйте підписування й нотаризацію в збирання з першого випуску і вирішіть, як оновлення доходитимуть до людей, ще до того, як хтось установить першу версію.

Що почитати далі

- NIST CSRC Glossary: application — визначення, якими користуються публікації NIST, від будь-якої системи, створеної під певний набір вимог, до систем нарахування зарплати й бухгалтерії.
- W3C Web Application Manifest — JSON-файл, що дає вебзастосунку назву, перелічує його значки й початкову адресу та задає режим, у якому він відкривається.
- W3C Service Workers — скрипт між вебзастосунком і мережею, з подіями кешу й запитів, завдяки яким можлива робота без підключення.
- Apple Developer: Notarizing macOS software before distribution — що перевіряє служба нотаризації і коли ПЗ, поширюване поза Mac App Store, мусить її пройти.

Словник

Обирайте за фактичними умовами

Опишіть робочий процес, наявні інтерфейси та користувачів, яким потрібна зміна. Ми можемо визначити перевірюваний результат і спосіб передачі, який дасть вашій команді змогу підтримувати його.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке застосунок?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/app/