Engineer

App

ein Programm, mit dem Menschen eine Aufgabe erledigen

Was es ist

Eine App ist ein Programm, das ein Mensch direkt öffnet und benutzt. Es hat ein Startsymbol, einen eigenen Bildschirm oder ein Fenster und eine erkennbare Aufgabe wie Nachrichten senden, einen Fahrplan prüfen oder eine Spesenabrechnung einreichen. „App“ ist die Kurzform von Application, und beide bedeuten dasselbe. Die Langform bleibt in formeller Sprache und Zusammensetzungen wie Webanwendung oder Desktopanwendung erhalten. Entwickler nennen eine App häufig Client, wenn sie das sichtbare Ende eines anderswo laufenden Dienstes ist.

Eine App ist Anwendungssoftware, im Gegensatz zur Systemsoftware, die den Rechner selbst am Laufen hält. Das NIST-Glossar sammelt mehrere Definitionen und merkt an, dass „Application und Software Application häufig synonym verwendet werden“. Eine Definition zählt auch Lohnabrechnungs- und Buchhaltungssysteme dazu. Eine Bibliothek liegt auf der anderen Seite der Grenze: Code, den andere Programme aufrufen und den kein Mensch öffnet.

Apps gehören zu drei großen Familien:

- **Native Apps** werden für eine Plattform mit deren eigener Sprache und Werkzeugen geschrieben, etwa Swift und SwiftUI auf dem Mac, und aus einem Store oder einem signierten Download installiert.
- **Web-Apps** laufen im Browser und sind über einen Link erreichbar. Ein Web-App-Manifest, eine kleine JSON-Datei mit Name, Symbolen, Start-URL und bevorzugtem Anzeigemodus, erlaubt die Installation auf dem Startbildschirm oder im Dock und ein eigenes Fenster. Ein Service Worker, ein zwischen Seite und Netzwerk laufendes Browserskript, ermöglicht den Offlinebetrieb. Beides zusammen heißt meist Progressive Web App, kurz PWA.
- **Plattformübergreifende Wrapper** verpacken eine Codebasis für mehrere Betriebssysteme, häufig mit einer in die App eingebetteten Browserengine.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist eine App?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. Eine schnelle, auffindbare Website, die Ihre Mitarbeitenden selbst bearbeiten können, ohne für jede Textänderung eine Rechnung zu bekommen.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird. 100 Webanwendungen mit HTML/HTML5 entwickelt — 100 entwickelte Webanwendungen

100 Webanwendungen mit HTML/HTML5 entwickelt Kontakt aufnehmen →

Warum es wichtig ist

Hinter den meisten App-Projekten steht die Frage nach der passenden Familie. Sie betrifft vor allem Nutzer und Verteilung, weniger den Code.

Eine Web-App benötigt eine Codebasis und erreicht über einen Link jeden mit Browser. Eine Aktualisierung erscheint mit der des Servers und wird beim nächsten Neuladen übernommen. Keine Store-Prüfung steht zwischen einer Korrektur und den darauf Wartenden. Der Preis ist die Tiefe: Ein Browser stellt weniger Gerätefunktionen bereit als das Betriebssystem, und jeder Browser bietet die Installation anders an. Unter iOS erscheint sie nie als Vorschlag. Wer seine Web-App auf Startbildschirmen sehen möchte, muss den Weg erklären.

Eine native App erhält die Bedienelemente, Tastaturabläufe, Barrierefreiheitsunterstützung und Leistung der Plattform sowie Zugriff auf deren Funktionen. Sie kostet eine Codebasis pro Plattform, Menschen mit dem jeweiligen Wissen und einen Verteilungsprozess: Store-Prüfung oder außerhalb des Stores Codesignatur und unter macOS Notarisierung. Seit macOS 10.15 verlangt Apple diese für mit Developer ID verteilte Software. Außerdem muss jemand den Aktualisierungskanal betreiben, solange die App existiert.

Ein plattformübergreifender Wrapper liegt dazwischen: eine Codebasis und eine installierbare App, bezahlt mit einer gebündelten Laufzeit und einer Oberfläche, die der Plattform ähnelt, sich aber nicht ganz so verhält.

Meist entscheidet, wo Nutzer die App öffnen. Ein Mensch am Schreibtisch mit Kabelverbindung verzeiht viel. Ein Mensch mit einem Mobiltelefon bei schwachem Empfang verzeiht wenig; für ihn ist Offlineverhalten eine

Anforderung und kein Feinschliff. Die andere entscheidende Frage ist, wer im dritten Jahr Aktualisierungen ausliefert. Die am billigsten gebaute App kann die teuerste im Weiterbetrieb sein.

Wie es in der Praxis aussieht

Auf einer Plattform arbeiten viele Nutzer als Seeleute oder im Außendienst mit dem Mobiltelefon, häufig ganz ohne Empfang. Dort wurde die Web-App installierbar gemacht. Vollständige Unterstützung für Progressive Web Apps liefert Startbildschirmsymbole in den verlangten Größen, einen Splashscreen und Themenfarben. Der Service Worker wird über next-pwa eingerichtet. Workbox liefert Schriftarten, Bilder, Audio, Video und CDN-Dateien zur Laufzeit bevorzugt aus dem Cache, zusammen mit HTTP-Cache-Control-Headern. Installation und Offlineverhalten wurden über HTTPS auf echten Geräten getestet. Nutzer können die App installieren, ohne Netz weiterarbeiten und bei späteren Besuchen aus dem Cache laden.

Ein Nachrichtenprotokoll mit einem plattformübergreifenden macOS-Desktopclient ging den umgekehrten Weg. Ein nativer macOS-Client in Swift 6 und SwiftUI umfasst 11.141 Zeilen in 53 Dateien. Der Rust-Kern des Protokolls ist als statisches Archiv aus einer festgeschriebenen Revision eingebunden. Er startet als Mac-Anwendung, verwendet native Bedienelemente und enthält keinen Browser. Die Seite benennt auch die Grenzen des Arbeitsstands: Eine veröffentlichte Version liegt nicht vor. Es gibt weder einen notarisierten Build noch einen Aktualisierungskanal, und die Oberfläche spricht eine Sprache.

Auch diese Website lässt sich als App installieren. Die Installationsseite erklärt den Weg für jeden Browser: kein Store, kein Konto, kein separates App-Paket zum Herunterladen, nur ein Symbol und ein eigenes Fenster. Safari auf dem iPhone bietet es unter „Teilen“ an. Firefox unter Windows kann Websites ab Version 143 als eigene Web-App-Fenster einrichten, bei Installationen aus dem Microsoft Store ab Version 150. Mozilla bietet diese Funktion weder unter macOS noch unter Linux an. Die Seite nennt die Grenze ebenfalls: Offlinebetrieb braucht einen Service Worker. Die Website liefert kein ausführbares JavaScript und registriert keinen Service Worker. Deshalb braucht die installierte App genauso eine Verbindung wie ein Browsertab.

Wo es schiefgeht

Der Bildschirm wird weiß, wenn der Empfang abreißt

Nutzer im Außendienst melden ein leeres Fenster oder die Offline-Seite des Browsers. Die App wurde auf der schnellen Verbindung im eigenen Gebäude entworfen und getestet. Jeder Bildschirm lädt seine Daten beim Öffnen. Entscheiden Sie, welche Ansichten offline funktionieren müssen, speichern Sie diese samt Ressourcen über einen Service Worker und testen Sie auf echten Geräten mit abgeschaltetem Netz statt in einer Desktop-Simulation.

Nutzer installieren eine Web-App und erwarten eine Offlinekopie

Jemand fügt die Website zum Startbildschirm hinzu, öffnet sie im Zug und sieht denselben Fehler wie im Browsertab. Ein Manifest macht installierbar; Offlineverhalten kommt nur vom Service Worker. Beides sind getrennte Entscheidungen. Liefern Sie entweder den Worker aus oder sagen Sie in der Installationsanleitung deutlich, dass die App eine Verbindung braucht.

Eine Veröffentlichung erscheint, Nutzer behalten die alte

Nach einer Bereitstellung läuft bei manchen noch tagelang der Code der Vorwoche. Der Service Worker liefert Seiten zuerst aus dem Cache, ohne eine neue Version zu kennzeichnen, oder ein neuer Worker wartet, bis sämtliche Tabs des alten geschlossen sind. Behalten Sie Cache-first für Ressourcen bei, die sich nicht pro Anfrage ändern, etwa Schriften, Medien und versionierte Dateinamen. Laden Sie Dokumente bevorzugt aus dem Netz und bieten Sie Neuladen an, sobald eine neue Version bereitsteht.

Der Wrapper fühlt sich auf dem Desktop fremd an

Mac-Nutzer beanstanden Menüs, Tastenkürzel und Textauswahl. Die App braucht Speicher wie ein Browser, weil sie einen enthält. Eine gebündelte Laufzeit ahmt die Plattform nach, statt sie zu verwenden. Ist die Desktop-Erfahrung das Produkt, vergleichen Sie einen nativen Client mit dem Wrapper. Gemeinsame Logik bleibt in einer Kernbibliothek, die alle Clients über eine stabile Schnittstelle aufrufen.

Die Verteilung bleibt bis zum Schluss liegen

Auf allen Entwicklerrechnern funktioniert die App. Der erste macOS-Kunde sieht stattdessen eine Gatekeeper-Warnung. Eine Korrektur wird ausgeliefert, doch die Hälfte der Nutzer behält die defekte Version. Signieren, Notarisieren und

Aktualisieren galten als Verpackungsarbeiten der letzten Woche. Nehmen Sie Signatur und Notarisierung ab der ersten Veröffentlichung in den Build auf und entscheiden Sie, wie Updates Menschen erreichen, bevor jemand Version eins installiert.

Weiterführende Lektüre

- NIST-CSRC-Glossar: Application — die Definitionen der NIST-Veröffentlichungen, von Systemen zur Erfüllung bestimmter Anforderungen bis zu Lohnabrechnung und Buchhaltung.
- W3C Web Application Manifest — die JSON-Datei für Name, Symbole, Start-URL und Anzeigemodus einer Web-App.
- W3C Service Workers — das Skript zwischen Web-App und Netzwerk mit Cache- und Fetch-Ereignissen für den Offlinebetrieb.
- Apple Developer: macOS-Software vor der Verteilung notarisieren — die Prüfungen des Notardienstes und die Fälle, in denen Software außerhalb des Mac App Store sie bestehen muss.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Beschreiben Sie den Arbeitsablauf, die vorhandenen Schnittstellen und die Nutzer, die die Veränderung benötigen. Wir können eine prüfbare Lieferung definieren und die Übergabe so gestalten, dass Ihr Team die gelieferten Bestandteile warten kann.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist eine App?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

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

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/app/