Engineer Software

Har bygget en native macOS-beskedklient i Swift 6

Har bygget en native macOS-beskedklient i Swift 6 og SwiftUI — 11.141 linjer fordelt på 53 filer — oven på C-grænsefladen til en Rust-kerne linket som et statisk arkiv fra en fastlåst revision.

august 2026

11,141

linjers Swift oven på en Rust-kerne

Situation. Beskedprotokollen havde en velafprøvet kerne skrevet i Rust og klienter på flere platforme, men skrivebordsoplevelsen på macOS var en platformsuafhængig skal — den så ikke ud som et Mac-program, opførte sig ikke som et og slæbte rundt på et runtime-miljø, som et native program ikke har brug for. Kernen stiller sine funktioner til rådighed gennem en C-grænseflade, hvilket betyder, at ethvert sprog, der kan kalde C, kan bygge på den, og det kan Swift.

Opgave. En native klient skulle bygges direkte på den C-grænseflade, i platformens nuværende sprog og nuværende UI-framework, med kernen linket ind og ikke leveret ved siden af.

Handling. Programmet er 11.141 linjer Swift fordelt på treoghalvtreds filer i app-målet, bygget på SwiftUI med kernen linket ind som et statisk arkiv, kompileret fra en fastlåst upstream-revision. At fastlåse revisionen i stedet for at følge en gren er den beslutning, der gør buildet reproducerbart: C-grænsefladen er kontrakten, og en kerne i bevægelse ændrer kontrakten uden at ændre en linje Swift. Lagdelingen holder den usikre overflade lille — en tynd indpakning ejer hver pointer og hver streng, der krydser grænsen, modellerne over den er almindelige Swift-værdier, og grænsefladelaget ser aldrig en rå pointer. Alt, hvad C-grænsefladen returnerer, har en ejerskabsregel, og en enkelt fejl i dem giver enten en lækage eller et nedbrud uden nogen advarsel fra oversætteren undervejs, så indpakningen er der, hvor hele projektets hukommelsesdisciplin bor. Buildet drives af en task runner med seksoghalvtreds mål, der dækker kompileringen af kernen, Swift-buildet, testsuiterne, linterne og pakningen.

Resultat. En native klient, der starter som et Mac-program, bruger platformens egne materialer og kontroller og ikke har nogen indlejret browser. Det, der ikke er gjort, bør siges: dette er ikke en udgivet version. Der er ingen notariseret distribution, ingen opdateringskanal og ét sprog i grænsefladen, så det er et fungerende program og ikke et produkt, som andre bruger.

Brug erfaringen i jeres projekt

Brug »Har bygget en native macOS-beskedklient i Swift 6« som udgangspunkt for en lignende udfordring i jeres organisation. Ydelserne bag arbejdet kan også bruges i jeres projekt. Systemets form lagt fast på papir først, så delene passer sammen, og prisen er kendt, før nogen skriver kode.

Se vores ydelser Kontakt os →

Se bag resultatet

Læs om situationen, beslutningerne og resultatet i eksemplet. Vi kan drøfte, hvad der passer til jeres vilkår, og hvad der kræver en anden tilgang. Har bygget en native macOS-beskedklient i Swift 6 — 11,141 linjers Swift oven på en Rust-kerne

Se vores ydelser Kontakt os →

Brug for en rigtig native app frem for en pakket webside?

Tal om jeres projekt med den ingeniør, der har ansvaret for leverancen. Vi aftaler omfang, prisgrundlag og overdragelse skriftligt, før arbejdet begynder.

Rolle i opgaven: macOS-applikationsudvikler. Det trækker på Løsningsarkitektur, Frontend-udvikling, Full stack-udvikling, API'er & integration, Designsystemer & UI og UX/UI-design. Og når I alligevel er her, er vi ret dygtige til Platform- & løsningsarkitektur, Frontend-udvikling, Full stack-produktudvikling og UI/UX-design & designsystemer.

Kontakt os Se vores ydelser

Ét stykke af **en længere fortegnelse** — og det hele ligger her på sitet.

Hver post er skrevet på samme måde: situationen, opgaven, hvad vi gjorde, og hvad der ændrede sig.

Se hele fortegnelsen

Definér den ønskede forbedring

Beskriv arbejdsgangen, de eksisterende grænseflader og de brugere, der har brug for ændringen. Vi kan definere en testbar leverance og en overdragelse, der gør teamet i stand til at vedligeholde løsningen.

Se, hvad det koster Kontakt os →

Brug for en rigtig native app frem for en pakket webside?

Spørg til beslutningerne bag »Har bygget en native macOS-beskedklient i Swift 6«. I taler med mennesker, der kan forklare arbejdet og vurdere jeres eget projekt.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af D.

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

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://software.engineer.company/da/portfolio/built-a-native-macos-client-in-swift-6/