Engineer Software

Cliente macOS nativo en Swift 6

Desarrollé un cliente de mensajería nativo para macOS en Swift 6 y SwiftUI — 11.141 líneas en 53 archivos — sobre la interfaz C de un núcleo Rust enlazado como biblioteca estática desde una revisión fijada.

agosto 2026

11,141

líneas de Swift sobre un núcleo Rust

Situación. El protocolo de mensajería tenía un núcleo escrito en Rust y bien probado y clientes en varias plataformas, pero la experiencia de escritorio en macOS era una interfaz multiplataforma: no parecía una aplicación de Mac, no se comportaba como tal y llevaba un entorno de ejecución que una aplicación nativa no necesita. El núcleo expone sus capacidades mediante una interfaz C, lo que significa que cualquier lenguaje capaz de llamar a C puede construir sobre él, y Swift puede hacerlo.

Tarea. Había que construir un cliente nativo directamente sobre esa interfaz C, en el lenguaje y el framework de interfaz actuales de la plataforma, con el núcleo enlazado dentro en lugar de distribuido al lado.

Acción. La aplicación consta de 11.141 líneas de Swift repartidas en cincuenta y tres archivos del destino de compilación de la aplicación, construida sobre SwiftUI con el núcleo enlazado como archivo estático compilado desde una revisión fijada del proyecto de origen. Fijar la revisión en lugar de seguir una rama es la decisión que hace reproducible la compilación: la interfaz C es el contrato, y un núcleo cambiante modifica ese contrato sin cambiar una línea de Swift. La separación en capas mantiene pequeña la superficie insegura: una envoltura ligera es responsable de cada puntero y cada cadena que atraviesan el límite, los modelos superiores son valores ordinarios de Swift y la capa de interfaz nunca ve un puntero sin procesar. Todo lo que devuelve la interfaz C tiene una regla de propiedad, y equivocarse en una provoca una fuga de memoria o un fallo abrupto sin que medie ninguna advertencia del compilador, por lo que en la envoltura reside toda la disciplina de memoria del proyecto. La compilación se dirige

mediante un ejecutor de tareas con cincuenta y seis objetivos que cubren la compilación del núcleo, la de Swift, las baterías de pruebas, los analizadores y el empaquetado.

Resultado. Un cliente nativo que desde el inicio es una aplicación de Mac, usa los materiales y controles de la propia plataforma y no lleva un navegador integrado. Hay que declarar lo que no se ha hecho: esta no es una versión publicada. No hay distribución notarizada, ni canal de actualizaciones, y la interfaz tiene un solo idioma, así que es una aplicación funcional, no un producto que utilicen otras personas.

Aplique esta experiencia a su proyecto

Utilice «Cliente macOS nativo en Swift 6» como punto de partida para un reto similar en su organización. Los servicios que respaldan este trabajo están disponibles para su proyecto. La estructura del sistema se define primero sobre el papel para que las piezas encajen y los costes se conozcan antes de escribir código.

Explore nuestros servicios Póngase en contacto →

Mire más allá del resultado

Lea la situación, las decisiones y el resultado de este ejemplo. Podemos hablar de qué partes se adaptan a sus restricciones y cuáles necesitan otro enfoque. Cliente macOS nativo en Swift 6 — 11,141 líneas de Swift sobre un núcleo Rust

Explore nuestros servicios Póngase en contacto →

¿Necesita una aplicación nativa real, más allá de una página web encapsulada?

Converse sobre su proyecto con la ingeniera responsable de la entrega. Acordamos por escrito el alcance, la base del precio y el traspaso antes de empezar a trabajar.

Contexto de la función: Ingeniera de aplicaciones macOS. Se apoya en Arquitectura de soluciones, Ingeniería de frontend, Desarrollo integral, API e integración, Sistemas de diseño e interfaces y Diseño UX/UI. Y ya que está aquí, también se nos da bien Arquitectura de plataformas y soluciones, Desarrollo de frontend, Desarrollo integral de producto y Diseño UI/UX y sistemas de diseño.

Póngase en contacto Explore nuestros servicios

Una parte de **una trayectoria más amplia** — y toda ella está en este sitio.

Cada entrada sigue la misma estructura: la situación, la tarea, lo que hicimos y lo que cambió.

Vea la trayectoria completa

Defina la mejora que necesita

Describa el flujo de trabajo, las interfaces existentes y los usuarios que necesitan el cambio. Podemos definir una entrega comprobable y un traspaso que permita a su equipo mantener lo entregado.

Vea cuánto cuesta Póngase en contacto →

¿Necesita una aplicación nativa real, más allá de una página web encapsulada?

Pregunte por las decisiones detrás de «Cliente macOS nativo en Swift 6». Hablará con las personas que pueden explicar el trabajo y valorar su propio proyecto.

Explore nuestros servicios

welcome@engineer.company +45 50 10 61 84 Copenhague, Dinamarca

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

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

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://software.engineer.company/es/portfolio/built-a-native-macos-client-in-swift-6/