

Mihaela Mihaljević Jakić

Senior iOS Architect | Swift Server & OpenAPI | AI Tooling

Experienced iOS engineer with a deep commitment to modular architecture and elegant code. I specialize in OpenAPI-driven development and AI-augmented tooling, building robust, multi-platform Swift applications that scale gracefully. From MCP servers that bring Apple documentation to AI assistants, to native AI-powered apps, I bridge the gap between Apple platform engineering and modern AI workflows. My work is defined by clean structure, extreme attention to testability, and the kind of modularization that makes features easy to extend, reason about, and maintain. I take pride in shipping polished, production-grade software that balances design clarity with engineering precision.

Contact:

- Email: mihaelamj@me.com
- Phone: +385995491157
- Location: Zagreb, Croatia
- [LinkedIn](#)
- [GitHub](#)
- [Website](#)

Education:

- M.Sc. in Information and Speech Science
- University of Zagreb, Faculty of Humanities and Social Sciences

EXPERIENCE

Maurer Electronics (Sep 2025 - Present) - Senior iOS Architect

ID Document Systems

- Architecting and developing iOS applications for secure ID document capture, personalisation, and verification
- Building mobile solutions for Bundesdruckerei GmbH's identity systems platform
- Designing modular, scalable architectures for security-critical mobile applications
- Remote from Zagreb, Croatia

Technologies: Swift, SwiftUI, UIKit, Swift Package Manager, Structured Concurrency, Unit Testing

Code Weaver (May 2024 - Jun 2025) - Senior iOS Developer

Anonymous Big EU Project

- Refactored client backend OpenAPI specs and built a Swift networking package
- Reorganized and clarified existing OpenAPI specifications to better reflect backend capabilities and client requirements. Ensured consistency and accuracy across endpoints and models
- Created a modular Swift package to encapsulate networking logic, designed for clean separation of concerns, testability, and reuse across projects
- Integrated Swift OpenAPI Generator to automatically generate type-safe API client code from OpenAPI specs, enabling seamless updates and minimizing manual maintenance
- Streamlined the build process to regenerate client code automatically upon spec changes, aligning with CI/CD pipelines and reducing the risk of contract drift

Technologies: Swift, SwiftUI, REST, OpenAPI, Swift Package Manager, Swift Server, Unit Testing, UI Testing, Structured Concurrency

Everliv AI Fitness Chat

- Developed comprehensive OpenAPI YAML specifications defining REST and Server-Sent Events endpoints for real-time chat functionality
- Built mock Swift Vapor server implementation with REST API and SSE streaming capabilities for development and testing workflows
- Integrated Swift OpenAPI Generator to automatically generate type-safe server code from YAML specifications, enabling seamless API contract enforcement
- Generated modular Swift networking layer from OpenAPI specs, providing clean separation of concerns and automatic client code updates
- Developed cross-platform SwiftUI chat clients for macOS and iOS with real-time messaging, error handling, and responsive user interfaces
- Implemented comprehensive preview and testing systems with mock data scenarios, ensuring robust development workflow and UI validation across platforms
- Engineered a custom Server-Sent Events (SSE) client with fine-grained streaming management, async/await handling, and resilient error propagation for real-time interactions
- Refactored chat architecture using actors, modular components, and adaptive UI to ensure a scalable, maintainable, and clean codebase
- Designed and implemented a rich ActionCard system supporting multiple types (nutrition, goal, measurement, etc.) with expandable content and platform-optimized UX
- Enhanced user experience with image upload workflows, previews, adaptive layouts, and cross-platform interaction consistency

Technologies: Swift, SwiftUI, REST, OpenAPI, Swift Package Manager, Unit Testing, UI Testing, Structured Concurrency, SSE, OpenAPI, Swift Server

iOLAP (Nov 2022 - Mar 2025) - Senior iOS Developer

ZUMIEZ Stash App

- Worked on the Zumiez Stash app as a Senior iOS Engineer
- Created Swift network client code generator from OpenAPI specs
- Refactored legacy applications and unit-tested critical components
- Implemented a custom lightweight GraphQL client without external dependencies, featuring type-safe request handling, efficient query utilities, custom GraphQL schema parsing, and sophisticated error management
- Designed and implemented a modular filtering system with dynamic UI components, supporting complex data sorting and real-time updates
- <https://www.zumiez.com>

- <https://apps.apple.com/app/apple-store/id1163863113>

Technologies: Swift, REST, OpenAPI, GraphQL, UIKit, UI in code, Swift Package Manager, Unit Testing

AI Chat SDK

- Developed a cross-platform chat SDK for iOS and macOS using Swift and SwiftUI for an AI chatbot
- Implemented real-time messaging, speech recognition, and UI components with modern architecture patterns
- Designed using modular architecture with 10+ local packages handling UI styling, message management, socket communication, and speech recognition
- Built with Combine for reactive programming, protocol-oriented design, and comprehensive testing infrastructure
- Implemented platform-specific components with shared protocols ensuring consistent behavior

Technologies: WebSockets, Swift, UIKit, Artificial Intelligence, Swift Package Manager, macOS, Unit Testing, AppKit

iRobot App

- Added features for the onboarding process
- Automated localization

Technologies: Swift, SwiftUI, Swift Package Manager, Objective-C, UIKit, UI in code, Unit Testing

Cherishing Studio (Jul 2019 - Oct 2022) - Senior iOS Developer

Wheels Up

- Modernized and refactored legacy Objective-C codebase to Swift
- Improved maintainability and fixed bugs
- Simplified login flow to enhance user experience and reduce authentication friction
- Developed custom UIKit components to improve app performance and user interface
- <https://wheelsup.com>
- <https://itunes.apple.com/us/app/wheels-up/id956615077>

Technologies: Objective-C, Swift, UIKit, UI in code, Push Notifications, REST, AFNetworking, OpenAPI, Unit Testing

Birch Finance

- Refactored a large iOS app's UI and networking layer using modern Swift
- Implemented MVVM architecture with a clean separation of concerns
- Built a comprehensive networking library with complete test coverage
- Organized code into feature-based modules
- Improved code quality through formatting, linting, and testing infrastructure
- <https://www.birchfinance.com>

Technologies: Swift, REST, UIKit, CoreAnimation, UI in code, Unit Testing, UI Testing

Budtz Innovation

- Led the refactoring of a legacy iOS app's UI and networking stack using modern Swift paradigms
- Reorganized the codebase into scalable, feature-driven modules to improve team velocity and onboarding

Technologies: Swift, REST, UIKit, UI in code

Token (Jun 2014 - Aug 2019) - Senior iOS Developer

Kindergarten (Cancelled)

- iOS app for a Croatian company, for managing kindergartens
- Developed 90% of the app
- Designed REST APIs and implemented them in the app
- Implemented custom UI design code

Technologies: Objective-C, UIKit, UI in code, AFNetworking, AutoLayout, REST, Carthage, Unit Testing, CocoaPods

Whatt - social network

- iOS app for a social network
- Developed custom expandable TextView with links and tagging

Technologies: Objective-C, UIKit, UI in code, AFNetworking, AutoLayout, REST, Carthage, Unit Testing, CoreAnimation, CocoaPods

Christian Resources

- Universal iOS app (iPhone and iPad), for a client through oDesk
- Developed the application (100%)
- Created design in Sketch and implemented it in the application, with dynamic UI for iPhone 4, iPhone 5, iPhone 6, iPhone 6 Plus, and iPad
- Used Core Animation, Auto Layout, AFNetworking, remote audio streaming
- Client changed parts of the UI afterward
- <https://itunes.apple.com/us/app/bible-study-tools-christian/id600610494>

Technologies: Objective-C, UIKit, UI in code, AFNetworking, AutoLayout, REST, Carthage, Unit Testing, AVFoundation, CoreGraphics

ServicePal

- Added custom features using Objective-C, UIKit
- Dynamic Content Framework Developer - designed and implemented a template-based dynamic content creation system that allowed for real-time content generation and updates without requiring app redeployment
- Increased development efficiency by 35%
- <https://servicepal.com>

Technologies: Objective-C, UI in code, UIKit, CoreData, REST, Unit Testing

Breck World

- Built app from scratch in Swift 4+

- Built UI in code using UIKit and Auto Layout
- Implemented networking layer using URLSession
- ARKit integration

Technologies: Swift, UIKit, UI in code, CoreData, AutoLayout, REST, ARKit

Masinerija (Nov 2018 - May 2019) - Senior iOS Developer

Birthdayrama App

- A fun way to share birthday wishes with friends and family
- Refactored the entire app
- Created a form factory for the app which uses many screens with input fields
- Used the latest Auto Layout best practices
- <https://www.producthunt.com/products/birthdayrama>

Technologies: Swift, UIKit, UI in code, CoreData, AutoLayout

Huxly App

- Newsreader reimaged
- Added many features and new screens to the app
- Login / Sign up / Forgot password screens (dynamic form screens)
- Demographics screens (dynamic radio control screens using Core Animation)
- Home and dynamic filter screens
- Sharing functionality

Technologies: Swift, UIKit, UI in code, CoreData, AutoLayout, REST, CocoaPods

Purch (Mar 2015 - Oct 2018) - iOS Developer

Consumr App

- Adapted existing client application for iOS 8+
- Performed detailed gap analysis for the application, including iOS frontend and REST API backend differences
- Implemented custom UI design code (from design sheets)
- REST API implementation
- Used Core Animation, Auto Layout, AFNetworking, Push Notifications
- <https://x.com/purchxapp>

Technologies: Objective-C, UIKit, UI in code, Push Notifications, CoreAnimation, REST, Unit Testing, AutoLayout, Barcodes & QR Codes

Shopsavvy App

- App for shopping and barcode scanning
- Developer in Swift, UIKit, Auto Layout
- Worked on parts of the app including barcode scanning, QR code scanning and creation, and general bug fixing
- <https://itunes.apple.com/us/app/shop-savvy-barcode-scanner/id338828953>

Technologies: Swift, UIKit, UI in code, CoreAnimation, REST, Unit Testing, AutoLayout, Barcodes & QR Codes, Swift Package Manager

QR Code Reader and Scanner App

- Objective-C, custom UI in code
- QR code scanning and creation
- Added custom sharing and feedback functionality
- <https://itunes.apple.com/us/app/qr-code-reader-and-scanner/id388175979>

Technologies: Objective-C, UIKit, UI in code, CoreAnimation, REST, Unit Testing, AutoLayout, Barcodes & QRcodes

Undabot (Sep 2013 - Jun 2014) - Junior iOS Developer

Tito

- iOS (iPad) book application about the life of Josip Broz Tito
- Objective-C, CocoaPods
- Implemented custom UI in code (from design sheets)
- CoreText custom page layouts

Technologies: Objective-C, UIKit, UI in code, CoreText, CoreGraphics, CoreAnimation, Carthage, In-App Purchase

wogibtswas.at

- wogibtswas.at, Austria's biggest "what's on offer" portal
- Objective-C, AFNetworking, REST services, CocoaPods
- Custom UI design in code
- <https://itunes.apple.com/de/app/wogibtswas.at-aktionen-angebote/id771962700>
- <https://www.wogibtswas.at>

Technologies: Objective-C, UIKit, UI in code, AFNetworking, AutoLayout, REST, Carthage

Blade Soho App

- Custom app for one of the leading London hair salons
- Objective-C, AFNetworking, REST services, CocoaPods
- Implemented custom UI in code (from design sheets)
- <https://www.bladesoho.co.uk>

Technologies: Objective-C, UIKit, UI in code, AFNetworking, AutoLayout, REST, Carthage, Unit Testing

Coachlette App

- Custom app for personal trainers and coaches
- Objective-C, AFNetworking, REST services, CocoaPods
- Implemented custom UI in code

Technologies: Objective-C, UIKit, UI in code, AFNetworking, AutoLayout, REST, Carthage, Unit Testing

CONFERENCES

iOSKonf26 (May 2026) - Speaker

- "Teaching AI to Read Apple Docs: Building Cupertino, an MCP Server in Swift"
- iOSKonf26, Skopje, North Macedonia
- [iOSKonf](#)

NSSpain XIII (Sep 2025) - Speaker

- "API-Driven Development Refactored: Scaling Server and Client Code in Highly Modular Architectures"
- NSSpain XIII, Logroño, Spain
- [NSSpain](#)
- [Talk recording](#)

OPEN SOURCE

Cupertino

- A local Apple Documentation crawler and MCP server, written in Swift
- CLI tool providing offline access to Apple developer documentation for AI assistants
- [GitHub](#)

Technologies: Swift, Swift Package Manager, MCP

iRelay

- Apple-native personal AI assistant, pure Swift
- Cross-platform macOS and iOS application
- [GitHub](#)

Technologies: Swift, SwiftUI, AppKit

SKILLS

AFNetworking | AI | AI Tooling | ARKit | AVFoundation | AppKit | Artificial Intelligence | AutoLayout | Barcodes & QR Codes | Carthage | CocoaPods | CoreAnimation | CoreData | CoreGraphics | CoreText | GraphQL | In-App Purchase | MCP | Objective-C | OpenAPI | Push Notifications | REST | SSE |

Structured Concurrency | Swift | Swift Package Manager | Swift Server | SwiftUI | UI Testing | UI in code | UIKit | Unit Testing | WebSockets | macOS

Created with [CVBuilder](#)