

Néstor Rubén Marsollier

Contact Information

- **Email:** nmarsollier@gmail.com
- **LinkedIn:** linkedin.com/in/nmarsollier
- **GitHub:** github.com/nmarsollier
- **Phone:** +54 261 503 5118
- **Address:** Villa Nueva, Guaymallén, Mendoza, Argentina

Personal Information

- **Languages:** Spanish (native), English (advanced)
- **Birth Date:** 10-15-1976

Education

- **Degree:** B.S. Computer Systems Engineering
- **Institution:** UTN Universidad Tecnológica Nacional, Mendoza, Argentina

Professional Summary

Experienced Android and Backend developer with a strong background in teaching and creating tutorials. Skilled in various programming languages and frameworks, with a focus on Android development since 2011 and Kotlin since 2012.

Skills

Personal Projects

Golang Microservices for E-Commerce

- 5 Microservices
- Different architectures styles from functional up to clean
- DB Polymorphism: Redis, PostgreSQL, Mongo
- Different storage techniques: Documental, Relational, Event Sourcing
- Restful api with gin
- GraphQL apollo server to integrate microservices
- RabbitMQ for async communication
- High integration coverage with gomock.
- CorrelationID trace using Logrus for logging, storing logs into MongoDB.

Golang Tutorial

- With crazy ideas about coding techniques

Mobile

- Open source apps and code samples with MVI architecture and Compose multi platform.
- Open source apps and code samples on React Native.

Backend Developer

- **Experience:** Since 1995
- **Experience:** Senior backend developer with expertise in monolith, SOA, and microservices architectures, including DDD when supported by the language.
- **Technologies:** GoLang, Java, Kotlin, Typescript, Javascript
- **Frameworks:** JSP, EJB 2 and 3, Spring stack, REST applications
- **Databases:** SQL (MySQL, PostgreSQL, Oracle), NoSQL (MongoDB), Key-Value (Redis)
- **Key Skills:** High-performance monolith applications, direct access, various ORMs
- **Code Samples:**
 - Many Microservices in Go and other languages,
 - Golang Tutorial

Android Native Developer

- **Experience:** Since 2011
- **Technologies:** Android SDK, Kotlin, Jetpack Compose, Kotlin Multiplatform, Espresso, JUnit4, SonarQube, Jacoco, Apollo (GraphQL), MVI architecture, Coroutines with flows, Dagger/Hilt, Navigation framework, DataDog, Jira, Figma, Confluence, Slack
- **Key Skills:** Kotlin, Koin, Google Services, GraphQL, REST, Retrofit, Room Database/SQLite, JUnit, Espresso, Jacoco, Hilt, Dagger, Reactive Paradigm, MVVM, MVC, MVI
- **Roles:** Technical Leader for Mobile at MercadoPago, Eventbrite
- **Architect:** MercadoPago
- **Methodologies:** Agile, Scrum, Jira, Figma, Confluence, Slack, Jenkins
- **Companies:** Dept, PayPal, MercadoPago, Mercadolibre, Eventbrite, Athos
- **Code Samples:** GitHub - Fit Fat

Teacher

- **Institution:** UTN (University)
- **Courses:** Microservices Architecture
- **Code Samples:** Microservices Architecture

Professional Experience

AirBnb

Sr Software Developer

Duration: 2024 - Present

Project: Viaduct Open Source

- Working on software refactoring to make Viaduct available Open Source.
- **Methodologies:** Watherfall.
- **Architecture:** Open Source Framework.
- **Languages:** Kotlin - Java.
- **Libraries:** Apollo Federation.

Dept Argentina

Principal II React Native Developer

Duration: 04/2024 - Present

Project: JOAN

- Successfully integrated Salesforce with React Native applications.
- Developed and refactored software using React Native with Expo Go.
- Leveraged Zustand for state management and Axios for data fetching.
- **Methodologies:** Agile.
- **Architecture:** Component-based architecture.
- **Languages:** JavaScript, TypeScript.
- **Libraries:** Zustand, React Query.
- **Tools:** Expo Go.

Principal II Android Developer

Duration: 08/2023 - 04/2024

Project: Prismacolor+ **Achievements:**

- Caught up within 6 months on an iOS project that had started 3 months earlier.
- Full refactor to apply a solid app architecture.
- Led the development of Prismacolor+, a customer acquisition project.
- Oversaw the architecture and development of the Android platform.
- Coordinated with cross-functional teams to ensure seamless integration and delivery of features.
- Applied best practices for code quality, testing, and deployment.
- Mentored junior developers and provided guidance on architectural decisions.
- **Methodologies:** Agile.
- **Architecture:** Eventual MVI.

- **Languages:** Kotlin.
- **Libraries:** Jetpack Compose, Kotlin MP, EXO Player.
- **Design Patterns:** MVI, Dependency Injection.
- **Tools:** Figma, Slack, Azure.

Backend Sr Developer

Duration: 08/2023 - 04/2024

Project: Financial software **Achievements:**

- Maintain and develop new features in.
- Applied best practices for code quality, testing, and deployment.
- **Methodologies:** Agile.
- **Architecture:** Eventual MVI.
- **Languages:** Golang.
- **Libraries:** Gin, Postgresql, Gomock.
- **Design Patterns:** SOLID, Dependency Injection.

PayPal

Sr Android Developer

Duration: 09/2021 - 08/2023

Project: Mastercard Credit Card Benefits **Achievements:**

- Reach 98% of code coverage.
- Built the Android UI for Mastercard Benefits.
- Created engagement screens, account summary, transaction details, and payment schedules.
- Implemented event-driven architecture using MVI, Coroutines with flows, and Dagger/Hilt for dependency injection.
- Collaborated with cross-functional teams to ensure seamless integration and delivery of features.
- Conducted code reviews and applied best practices for code quality and testing.
- **Methodologies:** Agile.
- **Architecture:** Event-driven, Model-View-Intent (MVI).
- **Languages:** Kotlin.
- **Libraries:** Android SDK, Espresso, JUnit4, SonarQube, Jacoco, Apollo (GraphQL), Navigation framework, Dagger/Hilt.
- **Design Patterns:** MVI, Dependency Injection.
- **Tools:** DataDog, Jira, Figma, Confluence, Slack.

Mercado Pago

Technical Leader

Duration: 05/2020 - 09/2021

Backend Project: Money In - Money Out Achievements:

- Fully refactor a microservice.
- Designed and developed money in flows (ask for money, get paid, request income).
- Created money out flows (send money, organize contacts, pay).
- Led technical solutions and implemented core functionalities.
- Programmed 2 microservices to enhance system performance.
- **Methodologies:** Agile.
- **Architecture:** Microservices, 3 layers of abstraction, MVC.
- **Languages:** JavaScript, TypeScript, GoLang.
- **Libraries:** Amazon AWS, EDS, Object Storage, KWS, Apache Kafka.
- **Design Patterns:** Functional, Clean.
- **Tools:** Amazon AWS, Apache Kafka, MySQL.

Android Project: Money In - Money Out Achievements:

- Lead 12 developers.
- Refactor a complex architecture.
- Coach Jr developers up to Sr level.
- Oversaw the architecture and development of the project.
- Designed and developed money in flows (ask for money, get paid, request income).
- Managed a team of 6 junior developers, coordinating the development of 4 modules and 3 libraries.
- Created money out flows (send money, organize contacts, pay).
- Led technical solutions and implemented core functionalities.
- Coached developers on technologies and architectures, ensuring adherence to best practices.
- **Methodologies:** Agile, Waterfall.
- **Architecture:** Event-driven, MVC, MVVM.

- **Languages:** Kotlin.
- **Libraries:** AndroidX Test, Espresso, Jetpack, OkHttp, Retrofit, ktx extensions, LiveData, Picasso, MockWebServer, PlayServices, Firebase, Gson, Google Maps.
- **Design Patterns:** Creational, Observers, State, Strategy, Command, Adapter, IoC.
- **Tools:** Jira, Bugsnag, Google Suite, Zeplin, CircleCI, GitHub, DataDog, Trello, Slack, Kafka.

Eventbrite

Senior Software Engineer II

Duration: 07/2016 - 05/2020

Project: Organizer App **Achievements:**

- I was the one person Android project against the job done by 12 iOS devs on the other app.
- Fully Refactor the app 2 times to adapt to new patterns. (java to kotlin, kotlin to MVVM reactive architecture)
- Responsible for all architectural and development decisions.
- Refactored the application twice to improve performance, maintainability, and scalability.
- Designed and implemented a fragment-based navigation system using MVC and MVVM architectures.
- Developed features using Java and Kotlin, ensuring high code quality and adherence to best practices.
- Applied event-driven architecture using MVVM, MVC, Repository, DAO (API), and various design patterns including Creational, Observers, State, Strategy, Command, Adapter, and IoC.
- Employed integrations like Jira, Bugsnag, Google Suite, Zeplin, GitHub, Slack, and Jenkins for project management and collaboration.
- Followed Agile and Waterfall methodologies to ensure timely delivery and high-quality software.
- **Architecture:** Fragment-based navigation, MVC, MVVM, Espresso, Robolectric.
- **Languages:** Java, Kotlin.
- **Libraries:** Jetpack, OkHttp, Retrofit, LiveData, Picasso, PlayServices, Firebase, Gson, AndroidX, PayPal Here, JUnit, Room, OrmLite, Analytics, Facebook Login.

- **Design Patterns:** Event-driven, MVVM, MVC, Repository, DAO (API), Creational, Observers, State, Strategy, Command, Adapter, IoC.
- **Tools:** Jira, Bugsnag, Google Suite, Zeplin, GitHub, Slack, Jenkins.
- **Methodologies:** Agile, Waterfall.

Atos Corp

Senior Developer, Architect

Duration: 01/2011 - 05/2014

Project: Redbus

- Acted as Architect and Senior Developer, leading a team of 2 developers.
- Designed and implemented web services to enhance system functionality.
- Utilized technologies including Android, JSP, Java Spring, Oracle, PostgreSQL, NFC, C++, REST.
- Managed and processed large amounts of data efficiently.
- Implemented data synchronization mechanisms to ensure data consistency across the system.
- Developed data consolidation strategies for accurate and reliable data aggregation.
- Ensured consistency control across various system components.
- Created features for card balance management.
- Implemented control over decentralized transaction processing to enhance system reliability.
- Applied object-oriented architecture using domain-driven design (DDD) principles to structure the system effectively.
- Ensured high code quality and adherence to best practices through code reviews.
- Provided mentoring and technical guidance to team members.
- **Architecture:** Object-oriented, domain-driven design (DDD).
- **Languages:** Java, JSP, C++.
- **Libraries:** Spring, NFC, REST.
- **Design Patterns:** Singleton, Factory, Observer, Strategy.
- **Tools:** Jira, GitHub, Jenkins.
- **Methodologies:** Agile, Waterfall.

Belatrix S.A.

Senior Developer

Duration: 03/2010 - 01/2011

Project: StoneRiver

- Acted as Senior Developer for the StoneRiver project.
- Developed and maintained applications using Spring, Mule, Java, and Flex.
- Worked with databases including DB2 and MySQL.
- Followed Agile methodologies and used Rally for project management.

- **Methodologies:** Agile.
- **Architecture:** Service-oriented architecture (SOA).
- **Languages:** Java.
- **Libraries:** Spring, Mule, Flex.
- **Design Patterns:** MVC, DAO.
- **Tools:** Rally, Eclipse, DB2, MySQL.

Senior Developer

Duration: 08/2009 - 03/2010

Project: Talent

- Acted as Senior Developer for the Talent project.
- Developed applications using Spring, Java, JSP, and Hibernate.
- Followed Agile methodologies and used Rally for project management.
- **Methodologies:** Agile.
- **Architecture:** Model-View-Controller (MVC).
- **Languages:** Java, JSP.
- **Libraries:** Spring, Hibernate.
- **Design Patterns:** MVC, DAO.
- **Tools:** Rally, Eclipse, MySQL.

Hewlett Packard Argentina

Specialist Developer

Duration: 02/2008 - 08/2009

Project: SABRE AACoRN System

- Acted as Senior Developer for the SABRE AACoRN System project.
- Developed applications using Java, Swing, EJB2, and Oracle.
- Worked with technologies including SABRE, Web Services, JSP, JaxWS, Axis 2, and Ant.
- Followed CMMI Level 3 standards and used tools like JIRA, Accurev, UML, StarTeam, and SAP.
- Designed and implemented core functionalities for the SABRE AACoRN System, ensuring high performance and reliability.
- Developed complex user interfaces using Swing for a seamless user experience.
- Integrated various web services and ensured smooth communication between different system components.
- Utilized EJB2 for building scalable and transactional business applications.
- Conducted code reviews and enforced best practices to maintain high code quality.
- Collaborated with cross-functional teams to gather requirements and deliver robust solutions.
- Provided technical support and training to team members and end-users.
- **Methodologies:** Agile.
- **Architecture:** Service-oriented architecture (SOA).

- **Languages:** Java, JSP.
- **Libraries:** Swing, EJB2, JaxWS, Axis 2.
- **Design Patterns:** MVC, DAO.
- **Tools:** JIRA, Accurev, UML, StarTeam, SAP.

Suris S.A.

Developer

Duration: 01/2007 - 02/2008

Project: Senectus / Vitae System

- Acted as Developer for the Senectus / Vitae System project.
- Developed applications using Java, Spring, ExtJS, and Hibernate.
- Designed and implemented REST APIs for seamless communication between client and server.
- Worked with SVN for version control, Eclipse as the development environment, and MySQL for database management.
- Ensured robust and scalable application architecture.
- Conducted unit testing and debugging to maintain code quality.
- Collaborated with cross-functional teams to gather requirements and deliver solutions.
- **Methodologies:** Agile.
- **Architecture:** RESTful architecture.
- **Languages:** Java.
- **Libraries:** Spring, ExtJS, Hibernate.
- **Design Patterns:** MVC, DAO.
- **Tools:** SVN, Eclipse, MySQL.

Imaquio S.A.

Technical Leader & Developer

Duration: 01/2002 - 01/2005

Project: Nuebus

- Acted as Architect & Technical Leader for the Nuebus project.
- Developed object-oriented applications with domain-driven design principles.
- Utilized technologies including Java, JSP, and Oracle.
- This was a system to sell long-distance tickets.
- Designed and implemented core functionalities for ticket booking and management.
- Ensured high availability and reliability of the system, which is still in operation.
- **Methodologies:** Waterfall.
- **Architecture:** Object-oriented architecture, domain-driven design (DDD).
- **Languages:** Java, JSP.
- **Libraries:** Oracle.

- **Design Patterns:** MVC, DAO.
- **Tools:** Eclipse, Oracle.

Technical Leader

Duration: 01/2005 - 01/2007

Project: Amadeus Client

- Acted as Technical Leader for the Amadeus Client project.
- Developed applications using EJB 2, Oracle, and Web Services.
- Developed JSP-based user interfaces for seamless user experience.
- This was a system to sell tickets for an airline.
- Designed and implemented core functionalities for ticket booking and management.
- Collaborated with cross-functional teams to integrate various services and APIs.
- Conducted performance tuning and optimization to enhance system efficiency.
- **Methodologies:** Waterfall.
- **Architecture:** Service-oriented architecture (SOA).
- **Languages:** Java, JSP.
- **Libraries:** EJB 2, Oracle, Web Services.
- **Design Patterns:** MVC, DAO.
- **Tools:** Eclipse, Oracle.

CRICYT

Developer, Designer, Analyst

Duration: 05/2006 - 06/2007

Project: Employee assistance control system

- Developed and designed the employee assistance control system.
- Used Java, Swing, and PostgreSQL.
- Implemented an Event Sourcing Database for reliable event tracking.
- Applied real object-oriented architecture with CQRS (Command Query Responsibility Segregation) to separate read and write operations.
- Maintained the event store for capturing application state changes.
- Projections to modify assistance motives in the past, ensuring accurate historical data.
- Provided technical support and training to users.
- **Methodologies:** Waterfall.
- **Architecture:** Object-oriented architecture, CQRS.
- **Languages:** Java.
- **Libraries:** Swing, PostgreSQL.
- **Design Patterns:** Event Sourcing, CQRS.
- **Tools:** Eclipse, PostgreSQL.

Senectus

Developer, Architect

Duration: 01/2000 - 01/2002

Project: Wine traceability system

- Engineered, conceptualized, and evaluated the wine traceability system.
- Leveraged technologies such as Fox Pro.
- Guaranteed system scalability and enhanced performance.
- Partnered with cross-functional teams to collect and refine requirements.
- Performed code inspections and enforced best practices for code integrity.
- Led a team of 3 developers.
- **Methodologies:** Waterfall.
- **Architecture:** Object-oriented architecture.
- **Languages:** Fox Pro.
- **Libraries:** N/A.
- **Design Patterns:** MVC, DAO.
- **Tools:** Fox Pro.

Altamed

Developer, Architect

Duration: 05/1998 - 01/2000

Project: Medicine insurance system

- Developed, designed, and analyzed the medicine insurance system.
- Utilized technologies including Fox Pro.
- Ensured system scalability and performance optimization.
- Collaborated with cross-functional teams to gather and refine requirements.
- Conducted code reviews and implemented best practices for code quality.
- Provided technical support and training to end-users.
- **Methodologies:** Waterfall.
- **Architecture:** Object-oriented architecture.
- **Languages:** Fox Pro.
- **Libraries:** N/A.
- **Design Patterns:** MVC, DAO.
- **Tools:** Fox Pro.

Banco de Prevision Social

Developer

Duration: 08/1995 - 05/1998

Project: Office supply tracking system

- Developed, designed, and analyzed the office supply tracking system.
- Acted as the architect for the refactor to convert the applications to Fox Pro.
- Utilized technologies including Clipper and Fox.

- Ensured system scalability and performance optimization.
- Collaborated with cross-functional teams to gather and refine requirements.
- Conducted code reviews and implemented best practices for code quality.
- Provided technical support and training to end-users.
- **Methodologies:** Waterfall.
- **Architecture:** Object-oriented architecture.
- **Languages:** Clipper, Fox Pro.
- **Libraries:** N/A.
- **Design Patterns:** MVC, DAO.
- **Tools:** Clipper, Fox Pro.

Previous Experience

- **Technologies:** Clipper, Fox, Fox Pro, Basic, Pascal, Java, JSP, EJB, HTTP, JavaScript, HTML, Oracle, MySQL, PostgreSQL, SQL Server.
 - **Clipper:** Used for developing database applications.
 - **Fox/Fox Pro:** Utilized for building data-centric applications with a focus on performance and scalability.
 - **Basic:** Employed for early-stage programming and scripting tasks.
 - **Pascal:** Applied for structured programming and algorithm development.
 - **Java:** Leveraged for building robust, high-performance applications.
 - **JSP:** Used for creating dynamic web pages and server-side applications.
 - **EJB:** Utilized for developing scalable, transactional, and multi-tier enterprise applications.
 - **HTTP:** Applied for web communication and data transfer.
 - **JavaScript:** Used for client-side scripting and enhancing user interfaces.
 - **HTML:** Employed for structuring web content.
 - **Oracle:** Utilized for managing large-scale databases.
 - **MySQL:** Used for relational database management and data storage.
 - **PostgreSQL:** Leveraged for advanced database management and complex queries.
 - **SQL Server:** Applied for enterprise-level database solutions and data warehousing.
- **Roles:** Various development and architecture roles since 1995.
 - **Developer:** Responsible for coding, debugging, and implementing software solutions.
 - **Designer:** Involved in creating software designs and architecture.
 - **Analyst:** Conducted system analysis and requirement gathering.
 - **Architect:** Designed and planned software architecture and system integration.
 - **Technical Leader:** Led development teams and provided technical

- guidance.
- **Mentor:** Provided training and support to junior developers.

UTN Universidad Tecnológica Nacional

Teacher

Duration: 2013 - Present

Course: Microservices Architecture

- **Technologies:** Java, Angular, Golang, Node.js, TypeScript, Kotlin, RabbitMQ, MongoDB, Redis, Kafka.
- **Responsibilities:** Developed and taught the Microservices Architecture course.
- **Study Plan:** Program and Study Plan.
- **Code Samples:** GitHub - nmarsollier.
- **Duration:** 2018 - Present.

Course: Advanced Programming

- **Technologies:** React, Node.js with TypeScript.
- **Responsibilities:** Developed and taught the Advanced Programming course.
- **Study Plan:** Study Plan.
- **Code Samples:** GitHub - nmarsollier.
- **Duration:** 2013 - 2023.

Repositories

- **Fit Fat Android App:** GitHub - Fit Fat
 - Controls fat and weight in different ways.
 - MVI Demo app in two versions: Android SDK and Jetpack Compose.
- **Fit Fat in React Native:** GitHub - Fit Fat React Native
 - Same functionality as the Android app but implemented in React Native.
- **Ecommerce Microservices:** GitHub - Ecommerce
 - Ecommerce microservices environment for educational purposes.
 - Includes 5 microservices written in Golang, TypeScript, Java, and Kotlin.
- **Golang Tutorial:** GitHub - Golang Tutorial
 - Advanced Golang ideas and techniques.
 - Includes innovative concepts and high-quality code.
- **Battery Switch Widget for Android:** GitHub - Battery Switch

- Switches the battery charger on/off based on battery percentage.
 - Supports Tasmota.
- **Volume Widget for Android:** GitHub - Volume Widget
 - A widget to control volume on Android devices.
- **My IP Widget for Android:** GitHub - My IP Widget
 - Displays the local IP address in a widget.
- **QR Barcode:** GitHub - QR Barcode
 - Generates barcodes from strings for demo purposes.
 - Implemented in Golang.
- **Semver Json Store:** GitHub - Semver Json Store
 - Downloads text resources in mobile apps while controlling versions.
 - Implemented in TypeScript/Node.js for educational purposes.
- **Document Store:** GitHub - Document Store
 - A simple document store implemented in Golang for educational purposes.
- **Card Game in React:** GitHub - Card Game
 - Backend for a card game implemented in React.
 - Designed for educational purposes.
- **4 in Line Game:** GitHub - 4 in Line
 - A simple game for educational purposes.
- **Crosswords Board Generator:** GitHub - Crosswords
 - Generates crossword boards.
 - Includes interesting problem-solving techniques.
 - Implemented in Java.
- **React Lessons:** GitHub - React Lessons
 - Educational repository with good and bad examples of React code.
- **JavaScript Lessons:** GitHub - JavaScript Lessons
 - Educational repository with good and bad examples of JavaScript code.
- **Node Lessons:** GitHub - Node Lessons
 - Educational repository with good and bad examples of Node.js code.

Quick Overview of my most relevant experience

Quick Overview