

# Matías Mirabelli

## Desarrollador de software

Soy desarrollador de software y tengo casi veinte años de experiencia en la industria. Me especialicé en el desarrollo de aplicaciones web y sistemas distribuidos. Desarrollé sitios web estáticos; aplicaciones clásicas request/response; aplicaciones frontend con componentes reactivos; diferentes tipos de APIs; servicios de infraestructura como job schedulers y single sign-ons; herramientas de análisis y procesamiento de datos como ETLs y scrapers; y sistemas biométricos, entre otras cosas. Trabajé con diferentes arquitecturas y modelos de aplicaciones como web services (SOAP), microservicios (REST), web sockets, request/response clásicos (MVC, streaming), procesamiento batch, y CLIs, entre otras. Usé e integré diferentes orígenes de datos.

En desarrollo de backend, trabajé con varios stacks de frameworks en PHP, Java, Scala, Kotlin y NodeJS. Entre los problemas de backend que resolví, puedo mencionar autenticación (SAML, OIDC), manejo de sesiones, manejo de transacciones, acceso a datos con múltiples orígenes de datos, búsqueda de texto libre, gestión y compresión de recursos estáticos (JavaScript, CSS, imágenes, videos, etc.), soporte de múltiples idiomas, geolocalización, clusterización y procesamiento de datos, storage de contenido, seguridad de datos y aplicaciones, integración con servicios de terceros, integración de hardware biométrico, estrategias de escalabilidad y resiliencia, entre otras cosas.

En desarrollo frontend, valoro más el diseño de arquitectura que el conocimiento específico de frameworks. Entre los problemas de arquitectura comunes que resolví puedo destacar el diseño de componentes reutilizables, la optimización de recursos en favor de la experiencia del usuario, el manejo de comunicación cross-domain, la personalización, el desarrollo progresivo y activación de features por usuario, el manejo y la persistencia de estado, soporte de idiomas, entre otras cosas. Algunos de los frameworks y utilidades que usé son Vue 2/3, React, charts.js, jQuery, y lessCSS. Use herramientas de build como webpack, vite, rollup and babel. Desarrollé tests funcionales utilizando cypress, puppeteer y playwright.

Pienso que el trabajo en equipo y un proceso de desarrollo enfocado en la calidad son la clave para un software exitoso. Entre las buenas prácticas para garantizar la calidad del delivery mencionaría resolver problemas utilizando los patrones de desarrollo de software que están muy probados en la industria, fomentar buenas prácticas y convenciones, hacer reviews de código, construir conocimiento común a través del diálogo y el diseño de soluciones y adoptar metodologías de organización cuando es necesario. Pienso que poner a las personas sobre los procesos es la clave del éxito de cualquier proyecto.

Apoyo la filosofía del software libre. Participo en la comunidad y trato de usar y contribuir al software libre cuando es posible. En mi tiempo libre desarrollo herramientas sociales para fomentar la transparencia gubernamental, la organización social y la toma de decisiones colectivas. Estudié Antropología Social y Cultural en la Universidad de San Martín y desarrollé varias herramientas como parte de mis investigaciones sobre política y movimientos sociales.

## Links

Github: <https://github.com/seykron> y <https://github.com/f-nyx>

Gitlab: <https://git.rlab.be/seykron>

Resume: <http://seykron.github.io/cv>

# Idiomas

Español (idioma materno)

English (upper-intermediate)

Esperanto (progresanta)

# Experiencia en desarrollo de software

## ***Gobierno de la Provincia de Buenos Aires (2020 - 2023)***

Desde el 2020 trabajé como consultor para el Ministerio de Justicia de la Provincia de Buenos Aires. Me contrataron para construir un sistema de identificación biométrica para el Servicio Penitenciario. Construí un sistema AFIS para enrolar e identificar personas en las unidades penitenciarias. El sistema consiste en una aplicación web que permite el enrolamiento y la búsqueda de personas; una API pública para interoperar con otros sistemas; y un servicio standalone y multi-plataforma para manejar lectores de huellas digitales. Entre los desafíos para construir un sistema AFIS puedo mencionar el soporte multi-plataforma; el uso de protocolos y formatos estándares para interoperar con diferentes sistemas biométricos; la arquitectura de la información para guardar la información biométrica de forma segura; la capa de abstracción para soportar diferentes dispositivos de hardware usando una interfaz común; la búsqueda asincrónica y distribuida de personas por huella digital.

El backend de la aplicación web está construido en Kotlin, utilizando Spring Boot, Postgres, librerías libres de procesamiento de datos biométricos como SourceAFIS, RabbitMQ, MinIO, entre otras. El frontend está construido utilizando Vue4 y TypeScript, Jest para unit testing, y Storybook para documentar la librería de componentes. También trabajé con drivers propietarios, y realicé la investigación para definir qué lectores de huellas digitales eran viables para este proyecto.

*Tecnologías: Kotlin, TypeScript, Vue3; librerías como Spring Boot, Exposed; data sources como Postgres y Redis; otros servicios como RabbitMQ y systemd (linux).*

## ***SCV - WireWheel (2019-2023)***

WireWheel es una empresa que vende productos de privacidad para otras empresas, como la gestión de pedidos de datos, o la gestión de consentimiento de cookies. Los primeros dos años trabajé como desarrollador en el equipo de uno de los productos, y en febrero del 2022 me pidieron que lidere el desarrollo de un nuevo producto para la gestión de consentimiento.

Como líder técnico tengo dos tipos de responsabilidades: operativas y técnicas. Entre las responsabilidades operativas están darle soporte técnico al equipo de desarrollo; garantizar cierta predictibilidad en la velocidad del equipo; garantizar la calidad del delivery; trabajar con el equipo de producto en la planificación a corto y mediano plazo; organizar el proceso de release; tomar entrevistas para el ingreso de nuevas personas; y monitorear los servicios en producción.

Entre las responsabilidades técnicas están diseñar las soluciones en conjunto con la gente que lo va a implementar; realizar las tareas necesarias de configuration management para que el equipo pueda trabajar fácilmente como la dockerización del entorno local y la automatización del build en CI; definir e implementar las políticas de calidad como las métricas de cobertura de test y el análisis estático de código; definir en conjunto con el director de desarrollo las tecnologías que vamos a utilizar; implementar

las integraciones más complejas como el sistema de contención para tener alta disponibilidad en los servicios públicos; e implementar las herramientas de soporte que necesita el equipo de QA para su trabajo de automatización.

Como trabajamos con datos sensibles, también soy responsable de la seguridad a nivel aplicación, y trabajo con el equipo de devops para diseñar el esquema de seguridad entre los servicios y la infraestructura. También soy responsable de analizar el resultado de los pentests para resolver los problemas de seguridad que surjan.

*Tecnologías: NodeJS con TypeScript; frameworks como Fastify para el servidor web y Awilix para inyección de dependencias; bases de datos como Cassandra, Postgres y Snowflake; servicios de AWS como S3, SQS, SSM, SNS, y ECS.*

## **Despegar.com (2015 - 2019)**

Despegar.com es una agencia online de viajes. Trabajé exclusivamente en backend desarrollando microservicios.

Los primeros dos años trabajé en el sector de ventas de vuelos. Tuve la oportunidad de reescribir el motor de booking de vuelos en Scala. Los desafíos fueron aprender el lenguaje, orquestar el flujo de booking que es considerablemente complejo, y soportar el throughput de ventas.

Trabajé en el sector de postventa de vuelos desarrollando el sistema de detección y procesamiento de cambios involuntarios o reprogramaciones de vuelos. Mi trabajo principal fue la integración con proveedores de vuelos como Sabre, Amadeus, Worldspan, American Airlines, etc. Algunos desafíos técnicos fueron el manejo del flujo asíncrono (los proveedores notifican las reprogramaciones asíncronamente); el manejo de fallos frente al usuario; el análisis de datos de los proveedores (la consistencia es eventual ya que depende de las operaciones de las aerolíneas); generar las abstracciones para soportar múltiples proveedores de vuelos; manejar el volumen de tráfico en servicios que requieren alta disponibilidad; diseñar herramientas de soporte para testing ya que varios proveedores no contaban con ambientes de pruebas, y reescribir varias aplicaciones para migrarlas a una arquitectura de microservicios.

En el equipo de desarrollo éramos seis personas, y trabajábamos directamente con el equipo de producto.

*Tecnologías: GIT, Maven, Spring MVC, Java Spark Web, Play Framework, Scalatra, Finagle, Cassandra, RabbitMQ, SBT, Java, Scala, Kotlin.*

## **55Social (2013 - 2015)**

55social fue una empresa de marketing digital cuyo mercado principal eran las agencias digitales de Brasil. La aplicación consistía en un gestor de contenido para crear y publicar campañas en Facebook, un gestor de múltiples cuentas para community management, y un sistema de métricas que se integraba con Facebook y Google Analytics para analizar el impacto de las campañas.

Trabajé como desarrollador frontend y backend en el gestor de campañas y el motor de métricas respectivamente. En frontend el desafío fue generar un entorno de diseño muy personalizable para que agentes de marketing pudieran construir campañas en Facebook solamente configurando una serie de widgets. En cuanto al backend, el desafío fue generar un motor de métricas que pudiera extraer datos y procesarlos para mostrar diferentes tipos de gráficos que permitían analizar el impacto de las campañas.

Durante casi toda mi experiencia en Globant y 55social también fui entrevistador y dicté varios talleres y capacitaciones sobre desarrollo web.

Dejé 55social porque la empresa nunca llegó al break event y tuvo que cerrar.

*Tecnologías: GIT, Maven, Java EE, Hibernate, Spring MVC, Katari, Elastic Search, Freemarker, Social APIs (Facebook, Twitter, Google Plus), jQuery UI, Histone, OOCSS, AWS*

### **Globant (2009 - 2013)**

Mi siguiente trabajo fue en Globant, otra consultora de recursos humanos. Trabajé como desarrollador web en un proyecto de JWT, un cliente de la consultora dedicado a la comunicación y el marketing. El proyecto consistía en una reingeniería completa de un sistema de manejo de contenido. El sistema organizaba contenido digital como documentos, imágenes, y videos. Uno de los requisitos era que pudiera reutilizarse para diferentes clientes de la agencia, de manera que JWT pudiera revender el almacenamiento y la gestión de contenido a sus clientes. En este proyecto conocí un modelo de programación orientado a servicios, donde gran parte de la responsabilidad de la aplicación estaba delegada en el frontend.

En Globant también tuve mi primera experiencia coordinando un equipo de desarrollo para JWT. Comenzamos un nuevo proyecto que buscaba resolver el problema de la fragmentación entre las agencias de JWT. A través de una red social de proyectos de marketing, las agencias en distintas partes del mundo podían presentar ideas creativas para estos proyectos, interactuar entre ellas, compartir contenido y conocer el trabajo de las demás. Para fomentar la participación, las ideas eran votadas por los mismos usuarios de la comunidad de JWT, y la más votada era la idea que se realizaría. Mi rol antes de comenzar el proyecto consistió en estimar el esfuerzo total del proyecto, configurar la infraestructura base para que el equipo pudiera trabajar sobre el código y acordar con el cliente la modalidad de entrega y seguimiento del proyecto.

Luego de más de dos años de trabajar con JWT, comencé a trabajar con un nuevo cliente de la consultora: 55social. El desafío era construir una plataforma de social media para facilitar la gestión de contenido y campañas de marketing en redes sociales. Gracias a un pequeño y excelente equipo de desarrollo la primera versión del producto estuvo online en cuatro meses. Luego de dos años de desarrollo, la empresa se desvinculó de Globant y yo seguí trabajando para ellos.

*Tecnologías: SVN, GIT, Maven, Java EE, MySQL, JackRabbit, Hibernate, Spring MVC, jawr, Freemarker, YahooUI, AWS, ImageMagick, FFMPeg*

### **CDA (2007 - 2009)**

Luego de dejar W3 trabajé como programador Java en CDA, una consultora de recursos humanos. Participé en el desarrollo de sistemas administrativos internos relacionados con personas (altas, bajas, vacaciones, etc), por lo que aprendí bastante sobre los procesos de RRHH. El desafío fue integrar diferentes bases de datos legacy, garantizar la consistencia entre distintos orígenes de datos, y aprender Adobe Flex, ya que el frontend de estas aplicaciones tenía que desarrollarse en esta tecnología. El backend lo desarrollé utilizando Java 1.6, Struts MVC y web services en la capa de aplicación, e Ibatis en la capa de acceso a datos.

Tuve la oportunidad de desarrollar desde el comienzo un sistema de homologación para el Banco Francés, un cliente de la consultora. Este sistema de homologación era parte de los controles SOX que debía cumplir el banco para garantizar la trazabilidad de todos sus procesos. El desafío fue trabajar con Java 1.4, una base de datos sin soporte de integridad de datos (una versión antigua de Sybase), y un modelo de programación requerido por el banco basado en stored procedures. El frontend lo desarrollé en JSP. Aprendí sobre la gestión técnica de un proyecto de desarrollo, y resolví problemas de infraestructura

relativos a acceso a datos, almacenamiento, y empaquetado y distribución de aplicaciones Java, entre otros.

*Tecnologías: SVN, Ant, Java EE (v1.4/1.6 + Web Services), Sybase DB, Ibatis, Struts, JSP, HTML, CSS, JavaScript, Flex (ActionScript 3, Cairngorm), Websphere*

### **W3americas (2006 - 2007)**

Mi siguiente trabajo fue en W3, una consultora de sistemas dedicada al desarrollo y mantenimiento web para terceros. Trabajé en el mantenimiento progresivo del sitio de Citibank Brasil. Participé en el rediseño completo del sitio, principalmente en el desarrollo de componentes que requerían integración con servicios de terceros como tablas de cotizaciones del mercado bursátil y cotizaciones de moneda, entre otras cosas.

Fue mi primer contacto directo con clientes, ya que a menudo tenía que coordinar las tareas con las personas responsables en Brasil. Durante mi tiempo libre aprendía sobre diseño de sistemas y desarrollo de aplicaciones web en Java, lo que más tarde usaría en mi próximo trabajo. Dejé de trabajar en W3 porque se volvió monótono y prácticamente no había desarrollo, sino mantenimiento.

*Tecnologías: SourceSafe, Perl, HTML, CSS, JavaScript*

### **Netic infoservicios (2006)**

Luego de finalizar la escuela secundaria, comencé a trabajar como programador Visual Basic en una pequeña empresa que vendía un producto de SEO (posicionamiento en motores de búsqueda). En esta empresa yo era el único programador y con muy pocas herramientas tuve que resolver problemas de arquitectura y distribución. El producto era una suite de aplicaciones que interactuaban entre ellas para generar micrositos con diferentes contenidos, y se distribuía como un paquete instalable. Dejé este trabajo porque el dueño de la empresa me pidió reiteradamente que facture menos de lo que estaba cobrando.

*Tecnologías: Visual Basic, MS Access, NSIS*

### **Freelance (2004 - 2006)**

Mis primeros trabajos fueron como freelance. Entre los proyectos destacables estuvieron el sitio corporativo del laboratorio Labyes y un sistema de gestión de viajes para una remisería en Visual Basic. En esta misma época también desarrollé varios sitios utilizando Wordpress y Drupal.

*Tecnologías: PHP, HTML, CSS, JavaScript, Visual Basic, MS Access, Symfony 1.0*

### **Software libre**

Durante años desarrollé, mantuve y colaboré en distintos proyectos de software libre. La lógica de organización de una comunidad de software libre es diferente a la de las empresas, y la experiencia de gestionar estos proyectos me ayudó a desarrollar una visión del liderazgo más crítica y orientada a la resolución de problemas a través de la colaboración. Muchos de los proyectos que empecé ya no existen, pero hay otros tantos que están publicados en github y me gustaría hacer algunos comentarios sobre los más relevantes.

### **Tehanu**

Tehanu, una librería para desarrollar bots sobre distintos servicios de mensajería.

## Domino

Domino, un sistema de gestión colaborativa de finanzas que usamos varios años en orgas de Software Libre.

## Kotlin-Search

kotlin-search, una librería para manejar índices de búsqueda locales usando Lucene. Soporta índices en múltiples idiomas.

## afip-sdk-kotlin

afip-sdk-kotlin, una librería para facturar en la AFIP usando los web services. También tiene soporte para manejo de certificados.

## Importador de datos legislativos

ogov-importer obtiene todos los proyectos legislativos y los respectivos votos del Congreso Nacional Argentino desde el 2001 hasta la fecha, la lista de legisladores y las reuniones de comisión, entre otras cosas. El importador es genérico y está diseñado para hacer scrapping de sitios de forma rápida y eficiente.

*Tecnologías: Grunt, JavaScript (NodeJS), Cheerio*

## API para cliente de IRC

irc-client-api permite conectarse a un servidor IRC desde el browser a través de una API de bajo nivel. Soporta diferentes transportes, por lo que se puede adaptar a distintos dispositivos o ambientes.

*Tecnologías: Grunt, JavaScript (NodeJS), WebSockets*

# Educación formal

Universidad Nacional de San Martín (abandonado) - *Lic. en Antropología Social y Cultural*

E.N.E.T N°17 D.E. 20 (2001 - 2006) - *Técnico Electrónico*

Alfonsina Storni (Escuela Primaria)

# Libros

La siguiente selección son libros que considero una parte fundamental de mi formación profesional. Algunos son como manuales, siempre vuelvo a consultarlos cuando necesito tomar decisiones informadas.

**Martin Fowler**, *Patterns of Enterprise Application Architecture*, 2002.

**“The Gang of Four”**, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1994.

**Robert Martin**, *Clean Architecture: A Craftsman’s Guide to Software Structure and Design*, 2017.

**Robert Martin**, *Clean Code: A Handbook of Agile Software Craftsmanship*, Pearson, 2008.

**Steve McConnell**, *Code Complete: A Practical Handbook of Software Construction*, Microsoft Press, 2004.



**Mary and Tom Poppendieck**, *Implementing Lean Software Development: From Concept to Cash*, Addison-Wesley, 2006.

**Mary and Tom Poppendieck**, *Leading Lean Software Development: Results Are Not the Point*, Addison-Wesley, 2009.

**M. van Steen and A.S. Tanenbaum**, *Distributed Systems*, distributed-systems.net, 2017.

**A. Silberschatz, P. B. Galvin, G. Gagne**, *Operating System Concepts 10th edition*, Wiley, 2021.

**Daniel Bovet (Author), Marco Cesati**, *Understanding the Linux Kernel 3rd edition*, O'Reilly Media, 2010.

**Robert Love**, *Linux Kernel Development 3rd edition*, Addison-Wesley Professional, 2010.

## Contacto

Pueden contactarme por email a [matias.mi+resume@riseup.net](mailto:matias.mi+resume@riseup.net). Referencias personales a pedido.