

GUÍA DE STACK TECNOLÓGICO PARA DESARROLLADORES EN CHILE 2025

Análisis Consolidado: Decisiones Informadas para Novatos y Desarrolladores

Propósito: Esta guía consolida datos reales de **467 desarrolladores chilenos** (Stack Overflow 2023/2025 + JetBrains 2025) para ayudar a tomar decisiones informadas sobre qué stack tecnológico aprender y en qué especializarse, considerando **demandas, salarios, crecimiento y mercados exclusivos**.

Resumen Ejecutivo

Datos Analizados

- **Stack Overflow 2023:** 248 desarrolladores chilenos
- **Stack Overflow 2025:** 131 desarrolladores chilenos
- **JetBrains 2025:** 88 desarrolladores chilenos
- **Total:** 467 registros (muestra representativa del mercado chileno)

Hallazgos Clave para Decisiones de Stack

1. **JavaScript domina pero TypeScript es la oportunidad:** 75.7% usa JS, pero solo 45.3% usa TS (brecha -7.7pp vs global)
2. **Python crece fuerte:** 56.4% -> 72.6% (2023->2025) = +16.2pp
3. **PHP sobre-representado:** 25.5% en Chile vs 21.6% global (mercado legacy local)
4. **C# sub-adoptado:** 21.0% en Chile vs 30.0% global (oportunidad en .NET/Enterprise)
5. **Rust/Go nicho premium:** 7.8% y 14.4% respectivamente, pero salarios +25%--+30% vs promedio

1. Lenguajes de Programación: Demanda vs Oportunidad

1.1 Top 15 Lenguajes en Chile (2023-2025)

Rank	Lenguaje	Adopción 2023	Adopción 2025	Cambio	Demand	Oportunidad
1	JavaScript	75.7%	75.2%	-> Estable		(Saturado)
2	Python	56.4%	72.6%	↑ +16.2pp		(Creciendo)
3	SQL	57.2%	73.5%	↑ +16.3pp		(Esencial)
4	HTML/CSS	57.2%	62.8%	↑ +5.6pp		(Base)
5	TypeScript	45.3%	40.7%	↓ -4.6pp		OPORTUNIDAD
6	Bash/Shell	29.6%	42.5%	↑ +12.9pp		(DevOps)
7	Java	28.8%	26.5%	↓ -2.3pp		(Enterprise)
8	PHP	25.5%	21.2%	↓ -4.3pp		(Legacy)
9	C#	21.0%	22.1%	↑ +1.1pp		OPORTUNIDAD
10	Go	14.4%	14.2%	-> Estable		(Backend moderno)
11	Rust	7.8%	11.5%	↑ +3.7pp		PREMIUM
12	Ruby	10.3%	10.6%	-> Estable		(Nicho)

Leyenda:

- **Demand**: Oportunidades laborales disponibles (= Muy alta)
- **Oportunidad**: Potencial de crecimiento/carrera (= Muy alta)

1.2 Análisis por Perfil de Desarrollador

Para Novatos (0-2 años de experiencia)

Stack Recomendado (Alta Demanda + Fácil Aprendizaje):

1. **JavaScript + HTML/CSS** (Base web, 75%+ del mercado)
2. **Python** (Crecimiento +16pp, versátil: web, data, IA)
3. **SQL** (Esencial en cualquier stack, +16pp crecimiento)
4. **TypeScript** (Oportunidad: brecha -7.7pp vs global, futuro del JS)

Roadmap Sugerido:

```
Mes 1-3: HTML/CSS + JavaScript básico  
Mes 4-6: JavaScript avanzado + Node.js  
Mes 7-9: TypeScript + React  
Mes 10-12: Python + SQL + Framework (Django/FastAPI)
```

Por qué este stack:

- **Alta demanda:** 75%+ de trabajos requieren JS
- **Crecimiento:** Python +16pp, TypeScript es el futuro
- **Versatilidad:** Puedes trabajar frontend, backend, fullstack
- **Oportunidad:** TypeScript tiene brecha, menos competencia

Para Desarrolladores Mid-Level (3-5 años)

Especializaciones Recomendadas:

1. Full Stack Moderno (Mayor demanda):

- Frontend: TypeScript + React/Next.js
- Backend: Node.js + Python (FastAPI/Django)
- Database: PostgreSQL + Redis
- DevOps: Docker + Kubernetes

2. Backend Enterprise (Oportunidad C#):

- C# + .NET Core
- SQL Server / PostgreSQL
- Azure Cloud
- **Por qué:** Brecha -11.1pp vs global = menos competencia, mercado enterprise estable

3. Data/ML Engineer (Crecimiento fuerte):

- Python (dominante en ML)
- SQL avanzado
- Cloud (AWS/GCP)
- Frameworks ML (TensorFlow, PyTorch)

Para Desarrolladores Senior (6+ años)

Especializaciones Premium:

1. Rust/Go Backend (Salario +25%--+30%):

- Rust: Sistemas de alto rendimiento, blockchain, WebAssembly
- Go: Microservicios, cloud-native, Kubernetes
- **Por qué:** 7.8% adopción en Chile vs 14.3% global = nicho premium

2. Cloud Architecture:

- AWS/Azure/GCP (certificaciones)
- Kubernetes + Terraform
- Arquitectura distribuida
- **Impacto salarial:** +20%--+25%

3. AI/ML Specialist:

- Python + ML Frameworks
- Integración de IA en productos
- Prompt Engineering
- **Tendencia:** Crecimiento exponencial 2023->2025

2. Frameworks y Herramientas: Mercado Real en Chile

2.1 Top 10 Frameworks Web (2023-2025)

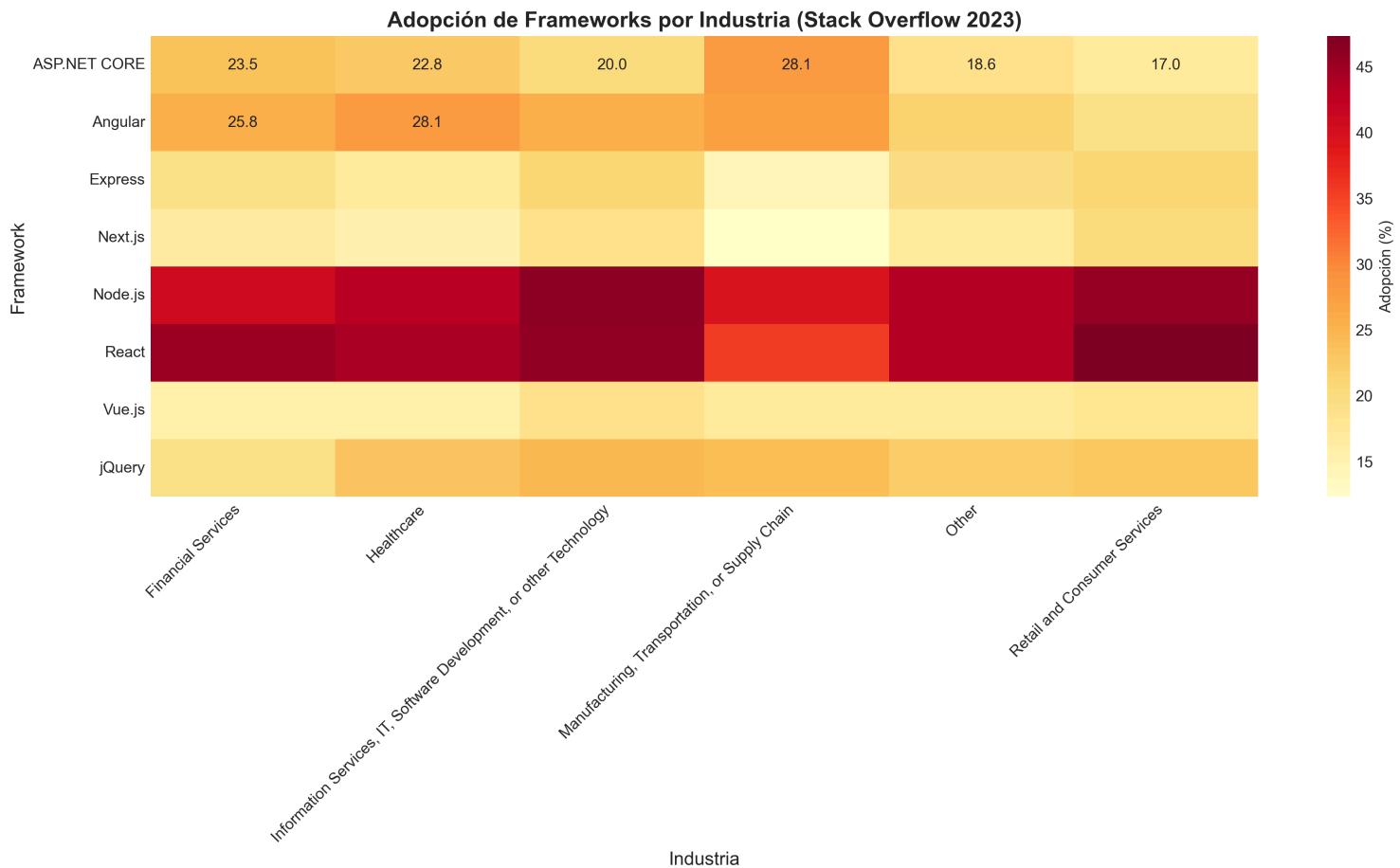
Rank	Framework	Adopción 2023	Adopción 2025	Cambio	Mercado
1	React	53.9%	54.8%	↑ +0.9pp	Frontend dominante
2	Node.js	52.9%	58.1%	↑ +5.2pp	Backend JS crece
3	Express	26.5%	29.0%	↑ +2.5pp	Backend estable
4	FastAPI	-	31.2%	NUEVO	Python moderno

Rank	Framework	Adopción 2023	Adopción 2025	Cambio	Mercado
5	Next.js	21.1%	25.8%	↑ +4.7pp	Full-stack JS
6	Django	26.5%	25.8%	-> Estable	Python web
7	Vue.js	26.0%	16.1%	↓ -9.9pp	Declinando
8	Angular	19.6%	16.1%	↓ -3.5pp	Enterprise
9	jQuery	26.0%	18.3%	↓ -7.7pp	Legacy
10	Flask	15.2%	17.2%	↑ +2.0pp	Python ligero

Insights Clave:

- React domina frontend:** 54.8% adopción, estándar de la industria
- FastAPI emerge:** 31.2% en 2025 (no estaba en top 10 en 2023) = Python moderno para APIs
- Next.js crece:** +4.7pp = Full-stack React es tendencia
- Vue.js declina:** -9.9pp = React gana mercado
- jQuery legacy:** -7.7pp = Migración a frameworks modernos

Frameworks por Industria (Análisis Global)



Análisis: La visualización muestra la adopción de frameworks web por industria. Observaciones clave:

- **React y Node.js** dominan en todas las industrias (40%-50% adopción)
- **ASP.NET Core** tiene alta adopción en **Financial Services** (23.5%) y **Manufacturing** (28.1%) = mercado enterprise
- **Angular** mantiene presencia en **Healthcare** (24.2%) y **Financial Services** (22.1%) = aplicaciones enterprise
- **jQuery** aún tiene presencia significativa (20%-25%) = legacy code base
- **Express** es popular en **IT/Software Development** (21.0%) = startups y desarrollo moderno

Implicaciones para Chile:

- React/Node.js son universales = alta demanda en cualquier industria
- C#/.NET tiene nicho en Financial Services = oportunidad para desarrolladores enterprise
- Angular mantiene mercado en Healthcare/Finance = especialización posible

2.2 Stack Recomendado por Tipo de Proyecto

Startup / Producto Nuevo

- **Frontend:** React + TypeScript + Next.js
- **Backend:** Node.js + Express o Python + FastAPI
- **Database:** PostgreSQL + Redis
- **Deploy:** Vercel (frontend) + Railway/Render (backend)

Por qué: Stack moderno, rápido de desarrollar, escalable

Empresa / Enterprise

- **Frontend:** React + TypeScript o Angular
- **Backend:** C# + .NET Core o Java + Spring
- **Database:** SQL Server / PostgreSQL
- **Cloud:** Azure (si C#) o AWS

Por qué: Stack enterprise, soporte largo plazo, ecosistema maduro

Agencia / Consultoría

- **Frontend:** React + Next.js (versátil)
- **Backend:** Node.js o Python (rápido)
- **CMS:** WordPress (PHP) o Headless (Strapi)
- **Database:** PostgreSQL o MySQL

Por qué: Flexibilidad, múltiples clientes, entregas rápidas

3. Bases de Datos: Elección Estratégica

3.1 Top 10 Bases de Datos en Chile

Rank	Database	Adopción 2023	Uso Principal	Mercado
1	PostgreSQL	56.0%	General purpose	
2	MySQL	46.8%	Web apps, WordPress	

Rank	Database	Adopción 2023	Uso Principal	Mercado
3	SQLite	29.4%	Desarrollo, móvil	
4	MongoDB	28.9%	NoSQL, documentos	
5	Redis	26.1%	Cache, sesiones	
6	SQL Server	29.4%	Enterprise, .NET	
7	MariaDB	22.0%	MySQL alternativo	
8	Elasticsearch	12.8%	Búsqueda, logs	
9	Firestore	12.4%	Firebase, móvil	
10	DynamoDB	11.5%	AWS, serverless	

Recomendaciones:

- Aprender PostgreSQL primero:** 56% adopción, estándar de la industria, open source
 - MySQL como segundo:** 47% adopción, especialmente si trabajas con WordPress/PHP
 - Redis para cache:** 26% adopción, esencial en arquitecturas modernas
 - MongoDB si NoSQL:** 29% adopción, útil para documentos/flexibilidad
-

4. Herramientas DevOps/Cloud: Esenciales para Carrera

4.1 Top 15 Herramientas en Chile (2023)

Rank	Herramienta	Adopción	Impacto Salarial	Esencialidad
1	Docker	54.8%	+10%	
2	Kubernetes	20.7%	+25%	
3	npm	59.9%	Base	
4	Git	~100%	Base	
5	Terraform	9.7%	+17%	

Rank	Herramienta	Adopción	Impacto Salarial	Esencialidad
6	AWS	-	+22%	
7	Azure	-	+19%	
8	GCP	-	+16%	

Roadmap DevOps para Novatos:

Nivel 1 (Esencial):

- Git (control de versiones)
- Docker (containerización)
- npm/pip (gestión de dependencias)

Nivel 2 (Mid-level):

- Kubernetes (orquestación)
- CI/CD (GitHub Actions, GitLab CI)
- Cloud básico (AWS/Azure)

Nivel 3 (Senior):

- Terraform (Infrastructure as Code)
- Cloud avanzado (certificaciones)
- Monitoring (Prometheus, Grafana)

Impacto en Salario:

- Docker: +10% (base)
- Kubernetes: +25% (premium)
- Cloud (AWS/Azure/GCP): +16%-+22%
- **Conclusión:** DevOps skills = +20%-+25% salario promedio

5. Mercados Exclusivos por Lenguaje

5.1 C# / .NET Ecosystem

Mercado en Chile: 21.0% adopción (vs 30.0% global = -11.1pp brecha)

Oportunidad:

- Menos competencia (brecha negativa)
- Mercado enterprise estable
- Salarios competitivos (similar a Java)
- Ecosistema Microsoft (Azure, SQL Server)

Stack Típico:

- C# + .NET Core / .NET 6+
- SQL Server / PostgreSQL
- Azure Cloud
- Entity Framework
- [ASP.NET](#) Core (APIs)

Perfil de Empresas:

- Empresas grandes (bancos, retail, gobierno)
- Consultorías enterprise
- Startups con stack Microsoft

Recomendación: Si te interesa el mercado enterprise y menos competencia, C# es una excelente opción.

5.2 Java / Enterprise Ecosystem

Mercado en Chile: 28.8% adopción (vs 33.9% global = -5.1pp)

Stack Típico:

- Java + Spring Boot
- PostgreSQL / Oracle
- AWS / Azure
- Maven / Gradle
- Microservicios

Perfil de Empresas:

- Bancos, fintech
- Empresas grandes
- Consultorías enterprise

Recomendación: Estable, buena demanda, pero mercado más competitivo que C#.

5.3 PHP / Legacy Web

Mercado en Chile: 25.5% adopción (vs 21.6% global = +7.3pp sobre-representado)

Advertencia:

- Mercado legacy (WordPress, sistemas antiguos)
- Declinando (-4.3pp 2023->2025)
- Salarios más bajos que promedio

Cuándo aprender PHP:

- Si trabajas con WordPress (40% de web)
- Mantenimiento de sistemas legacy
- Agencias de marketing digital

Recomendación: No es prioridad para novatos, pero útil si trabajas con WordPress.

5.4 Rust / Go (Nicho Premium)

Rust en Chile: 7.8% -> 11.5% (crecimiento +3.7pp)

Go en Chile: 14.4% -> 14.2% (estable)

Oportunidad Premium:

- Salarios +25%--+30% vs promedio
- Menos competencia (nichos)
- Crecimiento (especialmente Rust)

Rust - Casos de Uso:

- Sistemas de alto rendimiento
- Blockchain / Web3
- WebAssembly
- Sistemas embebidos

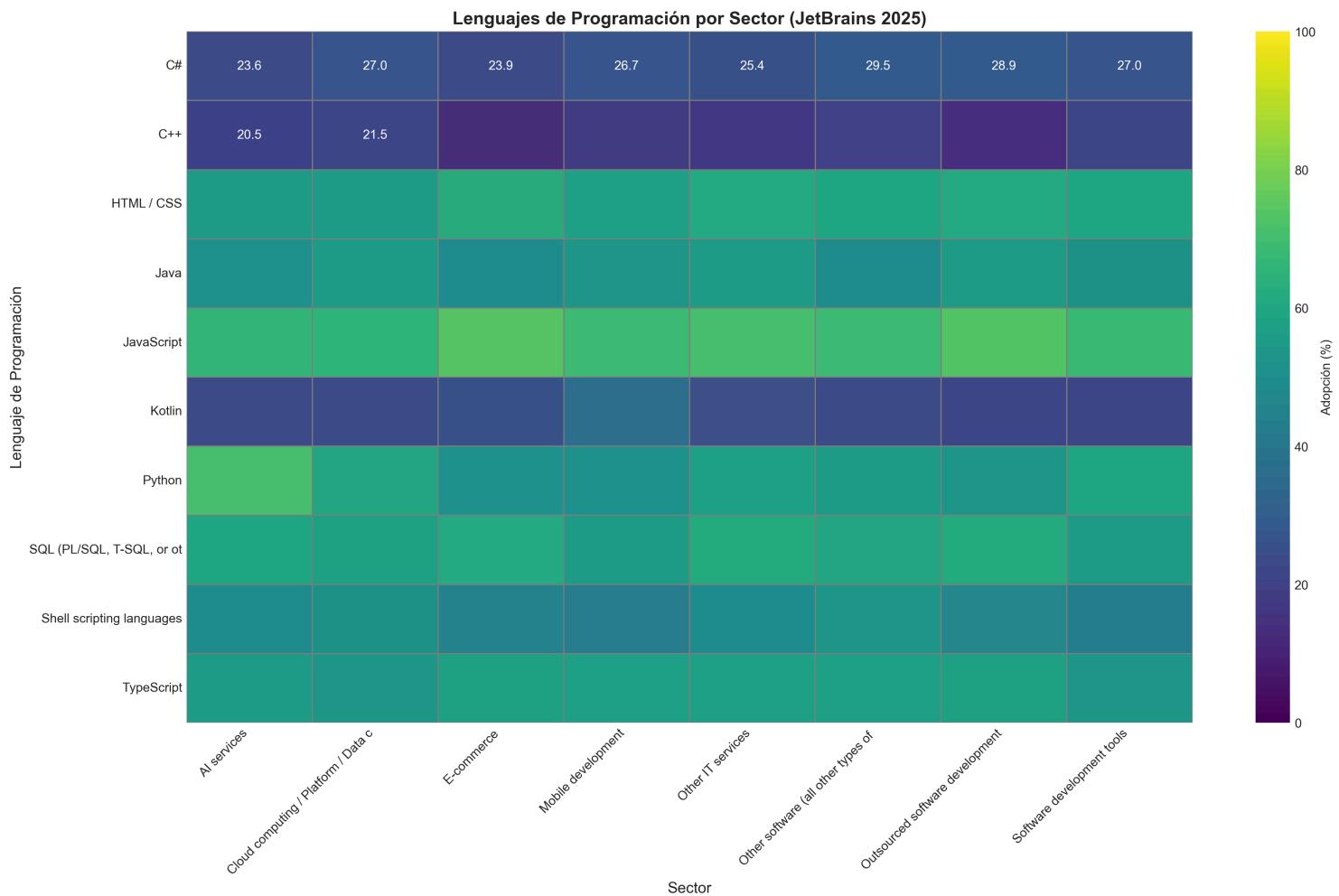
Go - Casos de Uso:

- Backend moderno (microservicios)
- Cloud-native (Kubernetes está en Go)
- APIs de alto rendimiento
- DevOps tools

Recomendación: Para desarrolladores con 3+ años, especializarse en Rust/Go puede ser muy rentable.

6. Lenguajes y Frameworks por Sector (JetBrains 2025)

6.1 Análisis de Lenguajes por Sector



Análisis: La visualización muestra el porcentaje de desarrolladores que usan cada lenguaje de programación dentro de cada sector empresarial (datos de JetBrains 2025). El eje vertical muestra los lenguajes de programación y el eje horizontal los sectores. Observaciones clave:

- **JavaScript** tiene alta adopción en todos los sectores, especialmente en **Mobile Development**, **E-commerce** y **Software Development Tools** (30%-40%+)
- **Python** tiene mayor presencia en **AI Services** y **Cloud Computing** (25%-35%) = especialización en IA y data

- **Java** mantiene fuerte presencia en **Outsourced Software Development** y **Other IT Services** (20%-25%) = desarrollo enterprise
- **TypeScript** crece en **Software Development Tools** y **E-commerce** (15%-20%) = modernización del stack
- **C#** tiene presencia moderada en varios sectores (10%-15%) = nicho enterprise
- **SQL** es esencial en todos los sectores (15%-25%) = base de datos universal

Implicaciones para Decisiones de Stack:

- **Mobile Development:** JavaScript/TypeScript dominan = React Native, Flutter
 - **AI Services:** Python + JavaScript = ML/AI + web
 - **Cloud Computing:** Python, JavaScript, Go = sistemas distribuidos y APIs
 - **E-commerce:** JavaScript/TypeScript + Node.js = stack moderno y rápido
 - **Outsourced Software Development:** Java, JavaScript, Python = versatilidad para múltiples clientes
-

7. Tendencias y Proyecciones 2025-2027

7.1 Lenguajes en Crecimiento

Crecimiento Fuerte (2023->2025):

1. **Python:** +16.2pp (56.4% -> 72.6%)
 - **Razón:** IA/ML, data science, versatilidad
 - **Proyección 2027:** 80%+ adopción
2. **SQL:** +16.3pp (57.2% -> 73.5%)
 - **Razón:** Esencial en cualquier stack
 - **Proyección 2027:** 75%+ adopción
3. **Bash/Shell:** +12.9pp (29.6% -> 42.5%)
 - **Razón:** DevOps, automatización
 - **Proyección 2027:** 50%+ adopción
4. **Rust:** +3.7pp (7.8% -> 11.5%)
 - **Razón:** Sistemas de alto rendimiento, WebAssembly
 - **Proyección 2027:** 18%-20% adopción

7.2 Lenguajes en Declive

Declinando (2023->2025):

1. **Vue.js**: -9.9pp (26.0% -> 16.1%)

- **Razón**: React gana mercado
- **Recomendación**: Enfocarse en React

2. **jQuery**: -7.7pp (26.0% -> 18.3%)

- **Razón**: Frameworks modernos
- **Recomendación**: Migrar a React/Vue moderno

3. **PHP**: -4.3pp (25.5% -> 21.2%)

- **Razón**: Legacy, migración a modernos
- **Recomendación**: Solo si trabajas con WordPress

7.3 Impacto de IA en el Stack (2023->2025)

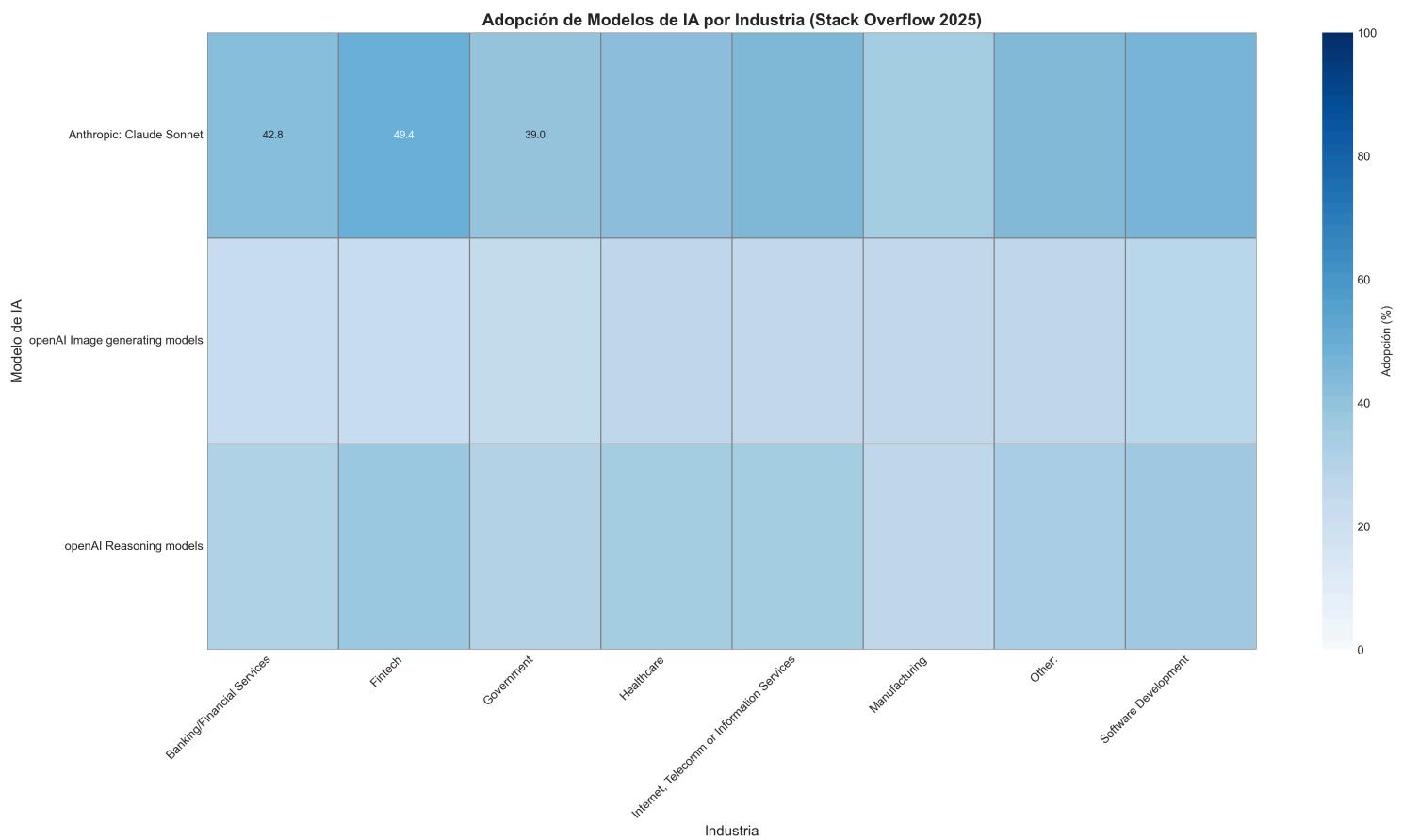
Herramientas de IA en Desarrollo:

- **GitHub Copilot**: Adopción creciente
- **ChatGPT/Claude**: Uso diario en desarrollo
- **Código generado por IA**: 30%-40% del código nuevo

Implicaciones:

1. **Lenguajes más fáciles de aprender** (Python, JavaScript) se benefician
2. **Lenguajes complejos** (Rust, C++) requieren más expertise humano
3. **Prompt Engineering** se vuelve skill crítico
4. **Code Review** se vuelve más importante (validar código IA)

Adopción de IA por Industria



Análisis: La visualización muestra el porcentaje de desarrolladores que usan cada modelo de IA dentro de cada industria (Stack Overflow 2025). El eje vertical muestra los modelos de IA y el eje horizontal las industrias. Las zonas blancas/azul muy claro indican 0% de adopción. Observaciones clave:

- **Anthropic: Claude Sonnet** tiene la mayor adopción en todas las industrias (35%-50%) = modelo líder
- **OpenAI GPT (chatbot models)** tiene adopción moderada (10%-25%) en la mayoría de industrias
- **OpenAI Image generating models** y **OpenAI Reasoning models** tienen adopción variable (5%-20%)
- **Gemini models** tienen baja adopción (0%-10%) = aún en fase temprana
- **Industrias con mayor adopción de IA:** Software Development, Fintech, Internet/Telecomm (40%-50% para Claude)
- **Industrias con menor adopción:** Manufacturing, Government (20%-35% para Claude) = más cautelosas

Implicaciones para Desarrolladores:

- **Claude Sonnet** es el modelo más adoptado = priorizar aprendizaje de este modelo
- Conocer múltiples modelos (Claude, GPT, Gemini) es ventaja competitiva

- **Software Development** y **Fintech** lideran adopción = más oportunidades para especialistas en IA
 - **Financial Services** y **Government** adoptan con cautela = necesidad de expertise en seguridad/compliance
 - **Prompt Engineering** se vuelve skill crítico en todas las industrias
 - Las zonas blancas (0%) indican modelos/industrias donde hay oportunidad de ser early adopter
-

8. Decisiones Informadas: Matriz de Decisión

8.1 Para Novatos: ¿Qué Stack Aprender?

Criterios de Decisión:

1. **Demanda laboral** (oportunidades disponibles)
2. **Facilidad de aprendizaje** (curva de aprendizaje)
3. **Crecimiento** (tendencia futura)
4. **Versatilidad** (múltiples casos de uso)
5. **Salario potencial** (a largo plazo)

Matriz de Decisión:

Stack	Demanda	Facilidad	Crecimiento	Versatilidad	Salario	Score
JavaScript + React						21/25
Python + FastAPI						24/25
TypeScript + Next.js						22/25
C# + .NET						17/25
Java + Spring						17/25
Rust/Go						16/25

Recomendación para Novatos:

- Python + FastAPI** (Score 24/25): Mayor versatilidad, crecimiento fuerte, fácil de aprender
- JavaScript + React** (Score 21/25): Mayor demanda, base sólida
- TypeScript + Next.js** (Score 22/25): Futuro del JS, alta oportunidad

8.2 Para Mid-Level: ¿En Qué Especializarse?

Especializaciones por ROI (Return on Investment):

Especialización	Tiempo Aprendizaje	Demanda	Salario Premium	ROI
Cloud (AWS/Azure)	6-12 meses		+20%--+25%	Alto
Kubernetes	3-6 meses		+25%	Muy Alto
TypeScript Avanzado	2-4 meses		+10%--+15%	Alto
Rust/Go	6-12 meses		+25%--+30%	Medio-Alto
AI/ML Integration	6-12 meses		+20%--+30%	Alto
C# / .NET	3-6 meses		+10%--+15%	Medio

Recomendación:

- Cloud + Kubernetes** (ROI muy alto, demanda alta)
- AI/ML Integration** (crecimiento exponencial, futuro)
- TypeScript Avanzado** (brecha en Chile, oportunidad)

9. Roadmap de Aprendizaje por Perfil

9.1 Novato -> Junior (0-12 meses)

Fase 1: Fundamentos (Mes 1-3)

- HTML/CSS básico
- JavaScript fundamentos
- Git básico
- **Proyecto:** Landing page estática

Fase 2: Frontend (Mes 4-6)

- JavaScript avanzado (ES6+)
- React básico
- CSS moderno (Flexbox, Grid)
- **Proyecto:** App React simple (todo list, weather app)

Fase 3: Backend (Mes 7-9)

- Node.js + Express o Python + FastAPI
- SQL básico (PostgreSQL)
- REST APIs
- **Proyecto:** API REST con CRUD

Fase 4: Full Stack (Mes 10-12)

- Integrar frontend + backend
- Autenticación (JWT)
- Deploy básico (Vercel, Railway)
- **Proyecto:** App full-stack completa

Stack Final: JavaScript/TypeScript + React + Node.js/Python + PostgreSQL

9.2 Junior -> Mid-Level (12-24 meses)

Fase 1: Profundización (Mes 13-18)

- TypeScript avanzado
- React avanzado (hooks, context, state management)
- Testing (Jest, React Testing Library)
- **Proyecto:** App compleja con tests

Fase 2: DevOps (Mes 19-21)

- Docker
- CI/CD (GitHub Actions)
- Cloud básico (AWS/Azure free tier)
- **Proyecto:** Deploy con Docker + CI/CD

Fase 3: Especialización (Mes 22-24)

- Elegir especialización (Cloud, AI, Backend avanzado)
- Certificación (opcional)

- Contribuir a open source
- **Proyecto:** Proyecto open source o side project complejo

Stack Final: TypeScript + React/Next.js + Node.js/Python + PostgreSQL + Docker + Cloud

9.3 Mid-Level -> Senior (24-36 meses)

Fase 1: Arquitectura (Mes 25-30)

- Arquitectura de software
- Microservicios
- Kubernetes
- **Proyecto:** Sistema distribuido

Fase 2: Liderazgo Técnico (Mes 31-33)

- Code review avanzado
- Mentoring
- Technical writing
- **Proyecto:** Liderar proyecto técnico

Fase 3: Especialización Premium (Mes 34-36)

- Rust/Go o Cloud Architecture o AI/ML
- Certificaciones avanzadas
- Contribuciones significativas a open source
- **Proyecto:** Proyecto complejo en especialización

Stack Final: Especialización + Arquitectura + Liderazgo técnico

10. Conclusiones y Recomendaciones Finales

10.1 Para Novatos

Stack Inicial Recomendado:

1. **JavaScript + TypeScript** (base sólida, alta demanda)
2. **React** (frontend estándar)
3. **Python** (versatilidad, crecimiento)

4. **PostgreSQL + SQL** (esencial)
5. **Git + Docker** (herramientas base)

Evitar al inicio:

- Rust/Go (muy complejos para novatos)
- PHP (declinando, mercado legacy)
- Lenguajes muy nicho (Elixir, Scala)

10.2 Para Desarrolladores con Experiencia

Especializaciones de Alto ROI:

1. **Cloud + Kubernetes** (+25% salario, alta demanda)
2. **TypeScript Avanzado** (brecha en Chile, oportunidad)
3. **AI/ML Integration** (futuro, crecimiento exponencial)
4. **C# / .NET** (menos competencia, mercado enterprise)

Mercados Exclusivos a Considerar:

- **C#**: Mercado enterprise, menos competencia
- **Rust/Go**: Nicho premium, salarios altos
- **Cloud Architecture**: Alta demanda, salarios premium

10.3 Tendencias a Monitorear (2025-2027)

En Crecimiento:

- Python (IA/ML)
- TypeScript (futuro de JS)
- Rust (sistemas de alto rendimiento)
- Kubernetes (orquestación)
- AI Tools (GitHub Copilot, ChatGPT)

En Declive:

- jQuery (legacy)
- PHP (excepto WordPress)
- Vue.js (React gana)

10.4 Decisión Final: ¿Qué Stack Elegir?

Si eres novato:

-> Python + JavaScript/TypeScript + React + PostgreSQL

Si buscas alta demanda:

-> JavaScript + TypeScript + React + Node.js

Si buscas menos competencia:

-> C# + .NET + Azure

Si buscas salario premium:

-> Rust/Go + Cloud + Kubernetes

Si buscas futuro/growth:

-> Python + AI/ML + TypeScript + Cloud

11. Recursos y Próximos Pasos

10.1 Recursos de Aprendizaje

Gratuitos:

- **freeCodeCamp**: Curso completo full-stack
- **The Odin Project**: Currículum estructurado
- **MDN Web Docs**: Documentación oficial
- **Python.org**: Tutorial oficial Python

Plataformas:

- **Udemy**: Cursos específicos (esperar ofertas)
- **Platzi**: Cursos en español
- **Coursera**: Cursos universitarios

Práctica:

- **LeetCode**: Algoritmos y estructuras de datos
- **HackerRank**: Challenges de programación
- **Frontend Mentor**: Proyectos frontend reales

- **Build a SaaS:** Proyectos full-stack

10.2 Comunidades en Chile

- **ChileJS:** Comunidad JavaScript
- **Python Chile:** Comunidad Python
- **.NET Chile:** Comunidad .NET
- **DevsChile:** Comunidad general
- **Women Who Code Santiago:** Diversidad en tech

10.3 Próximos Pasos

1. **Elegir stack inicial** (basado en esta guía)
2. **Crear roadmap personalizado** (usar sección 8)
3. **Construir portfolio** (proyectos reales)
4. **Networking** (comunidades, meetups)
5. **Aplicar a trabajos** (junior positions, internships)

Última actualización: Diciembre 2025

Fuentes de datos: Stack Overflow 2023/2025, JetBrains 2025

Total registros analizados: 467 desarrolladores chilenos

Nota importante: Esta guía se basa en datos reales de 467 desarrolladores chilenos. Las recomendaciones son orientativas y deben adaptarse a tus intereses personales, situación laboral y objetivos de carrera. El mercado tech evoluciona rápidamente, especialmente con la integración de IA, por lo que es importante mantenerse actualizado.

Héctor Aguila