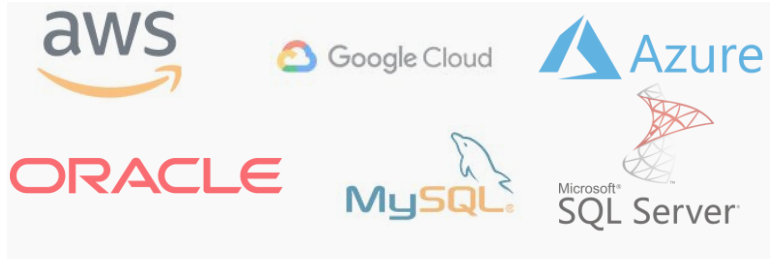


INGENIERÍA DE SOFTWARE Y REQUERIMIENTOS

FACULTAD DE INGENIERÍA EN SISTEMAS

MAESTRÍAS EN INGENIERÍA DE SISTEMAS



Ingeniería en Informática

Software

Computación

Videojuegos

Seguridad Informática

CyberSeguridad

Especializaciones en
Ing. de Sistemas

Sistemas de Información

Tecnologías de la Información

Inteligencia Artificial

Ciencia de Datos



AUTOPSY
DIGITAL FORENSIC



ANALOGÍA A CONSTRUIR SOFTWARE PROFESIONAL



Sin ingeniería

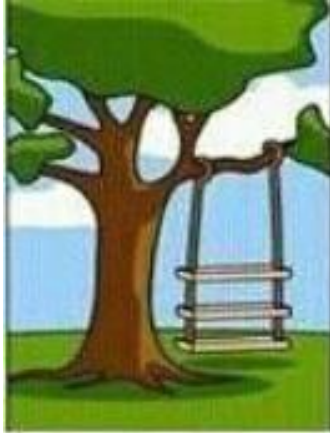


Con ingeniería

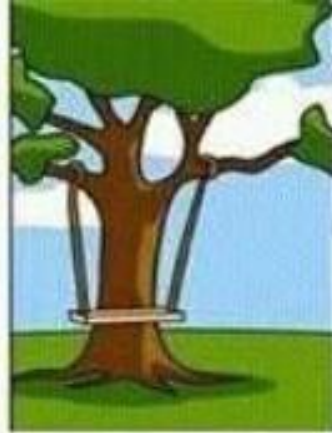
CONSTRUCCIÓN DEL SOFTWARE



SDLC



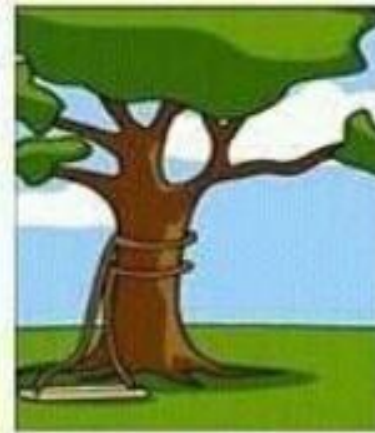
1. Como lo explicó el cliente.



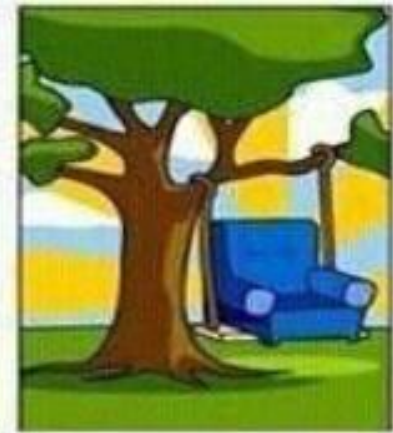
2. Como lo entendió el líder de proyecto.



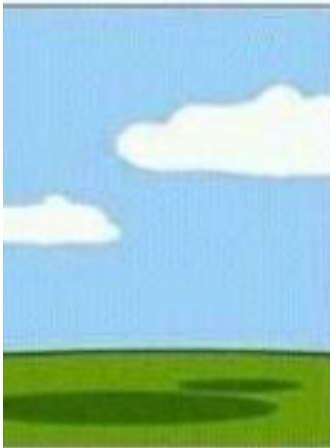
3. Como fue diseñado por el analista.



4. Como fue programado el producto.



5. Como fue descrito por el consultor de negocio.



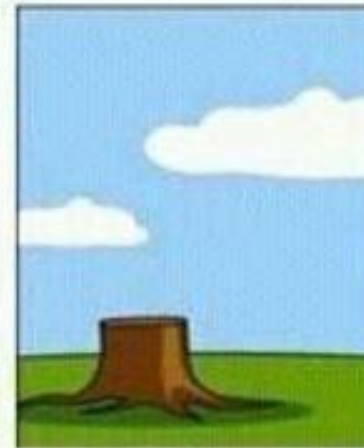
6. Lo documentado del producto y del proyecto.



7. Lo que se instaló para la operación.



8. Lo que se facturó.

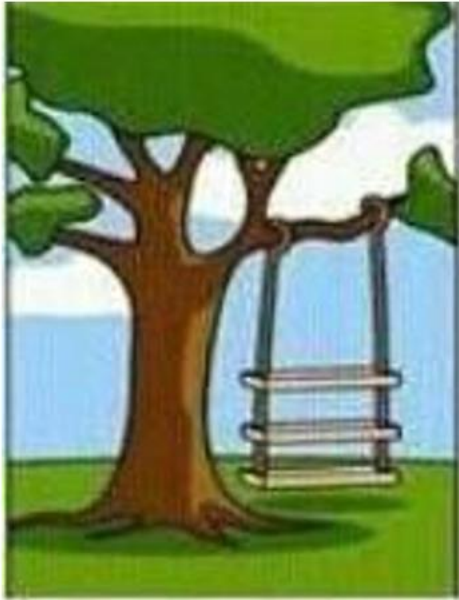


9. El soporte al producto.



10. Lo que en realidad necesitaba el cliente.

SDLC



1. Como lo
explicó el
cliente.



10. Lo que en
realidad
necesitaba el
cliente.

Generaciones y Estructuras

Baby Boomers

1946-1964

Generación X

1964-1976

Generación Y

1977-1992

Generación Z

1993-...

Organizacional

Fiel a su profesión

Emprendedores digitales

Multicarrera



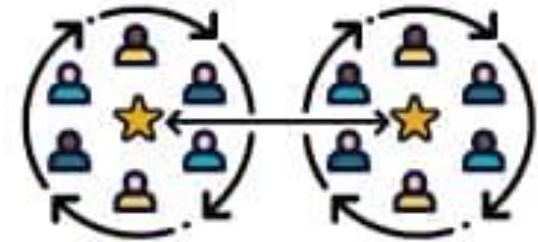
Jerarquía



Horizontal



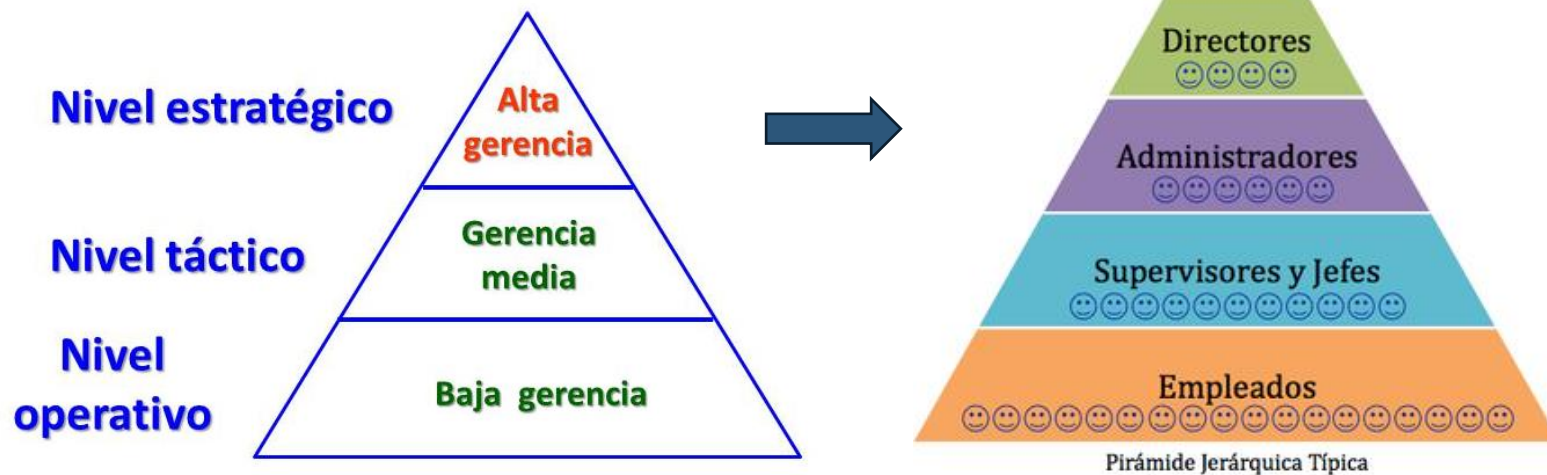
Plano



Holocracia

ENTERPRISE LEVELS

Gráfico 1. La pirámide organizacional



QUE SON LOS “HARD SKILLS” Y “SOFTS KILLS”

- Las habilidades duras (“hard skills”, en inglés) son las destrezas que adquirimos gracias a nuestra formación académica y nuestra experiencia profesional.
- Soft skills o habilidades blandas son las características y competencias que configuran el comportamiento individual de los profesionales en el ámbito de las empresas.

EJEMPLOS DE SOFTSKILLS



CURSO

Lenguaje no verbal para líderes

1 h 54 m



CURSO

Mejora tu competencia en conflictos

1 h 41 m



CURSO

Cómo tener conversaciones difíciles

1 h 12 m



CURSO

Comunicación interpersonal

Por: Rudi Bruchez y Elena Compte Tordesillas

72.909 visualizaciones · Lanzamiento: 20 de sept. de 2016

1 h 52 m



CURSO

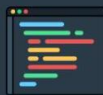
Fundamentos de la resolución de conflictos

Por: Lorena Díaz Quijano

64.302 visualizaciones · Lanzamiento: 1 de oct. de 2018

1 h 35 m

DEVELOPER LEVELS



JUNIOR

- Necesita supervisión.
- Conocimientos básicos sobre software y hardware.
- Conoce al menos un lenguaje de programación.
- Colabora en la planificación inicial del proyecto.
- Trabaja en funciones y herramientas internas de software.



SEMI SENIOR

- Capacidad técnica de realizar tareas con menos supervisión.
- Conoce las etapas del desarrollo: análisis, desarrollo, prueba, implementación, documentación, etc.
- Configura un ambiente de desarrollo por sí mismo.
- Detecta errores de código y lo hace más eficiente.
- Crea y escribe pruebas unitarias simples.

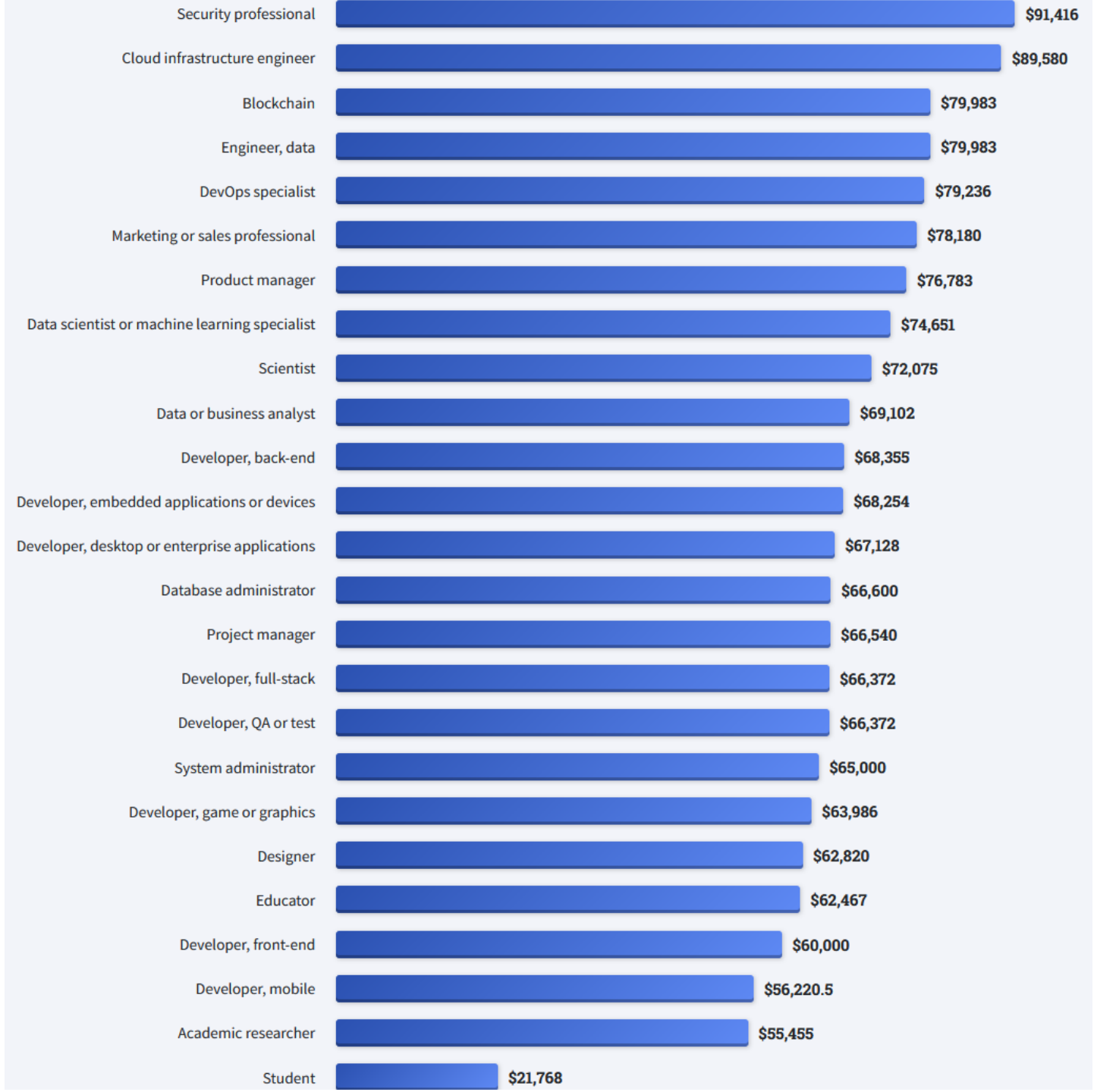


SENIOR

- Es capaz de supervisar y dirigir equipos.
- Comprende el alcance de un proyecto y plantea métodos para desarrollar, probar, implementar y mantener el proyecto.
- Asesora a desarrolladores junior y semi senior.
- Hace revisiones periódicas de código.
- Mejora la calidad y estructura del código.

Path de carrera – bandas salariales

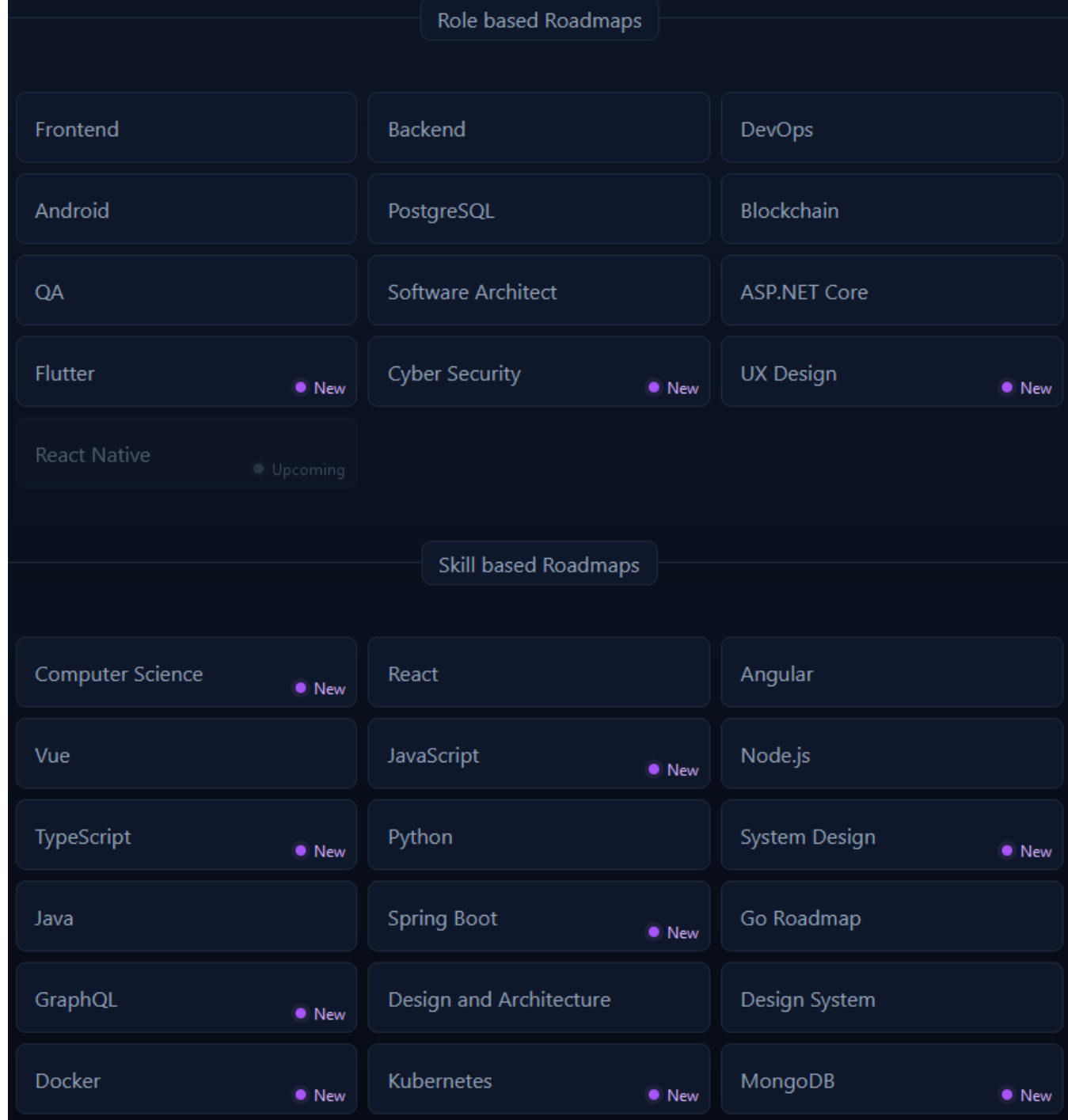
[Work | 2024 Stack Overflow Developer Survey](#)



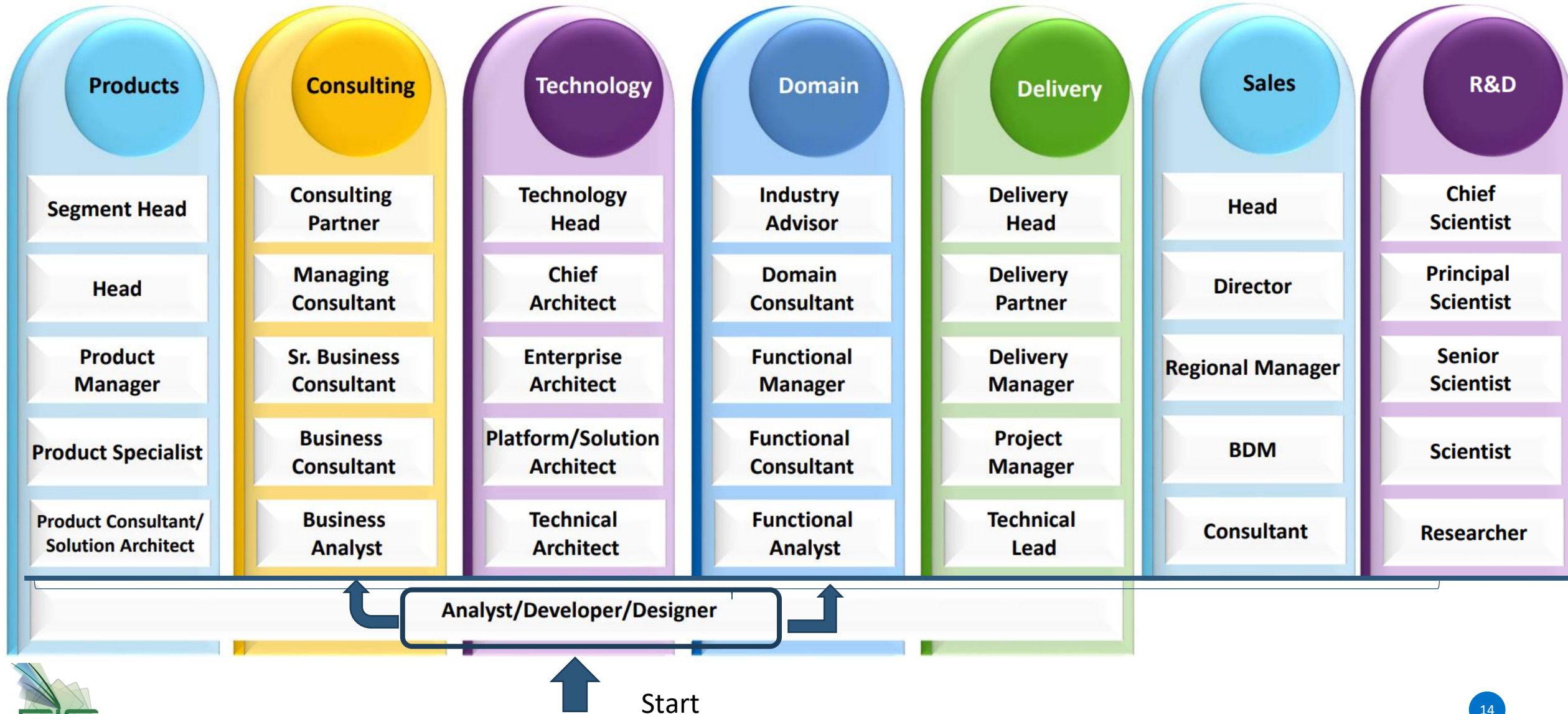
ROADMAP DEL DESARROLLADOR



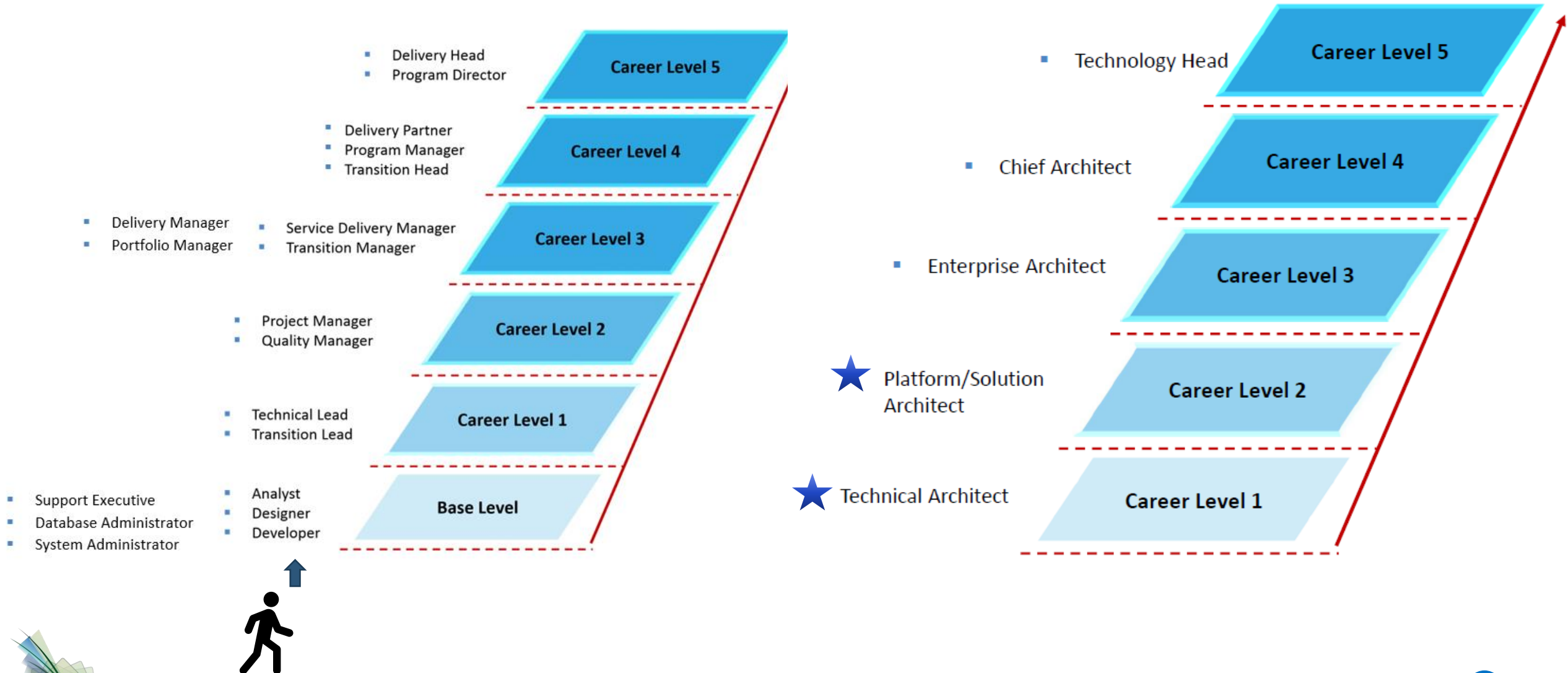
<https://roadmap.sh/>



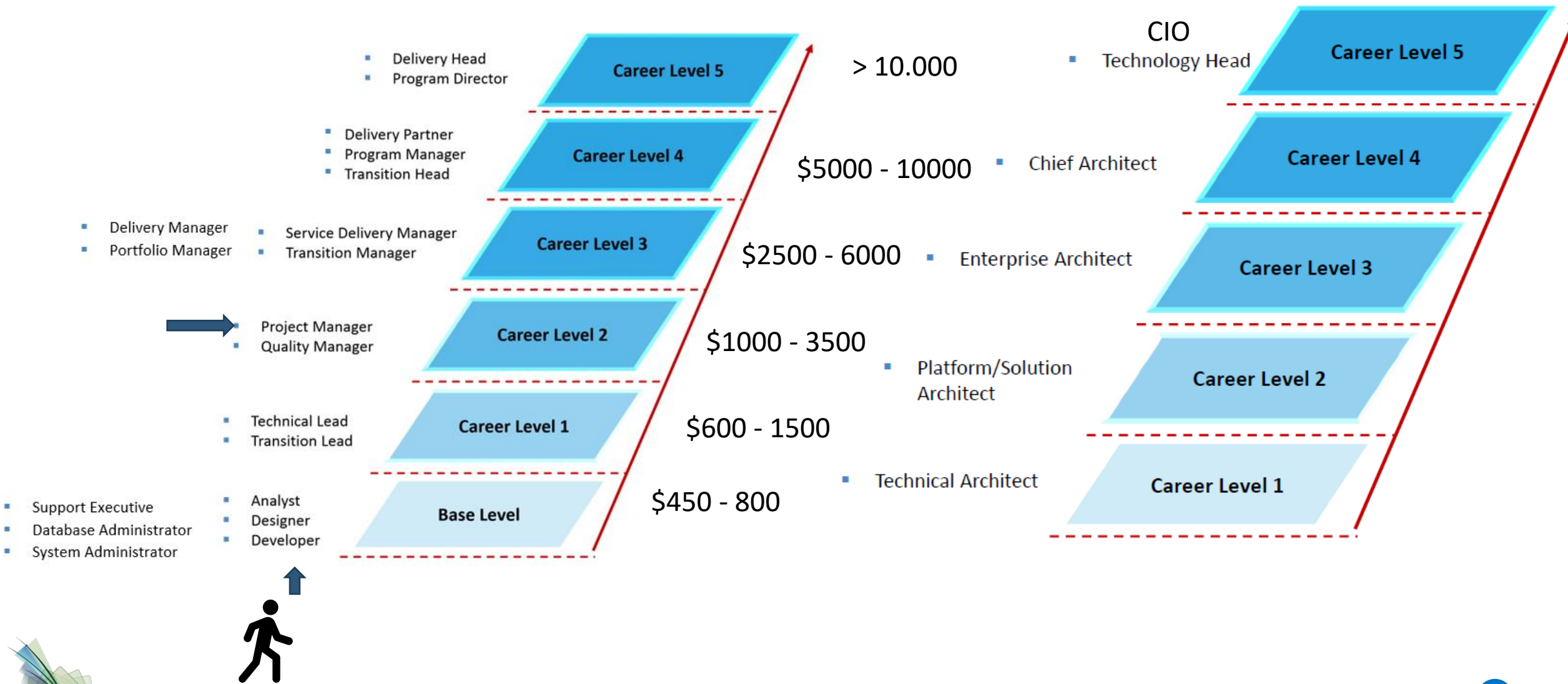
PATH DE CARRERA



PATH DE CARRERA



PATH DE CARRERA



PATH DE CARRERA

Hard Skills/
Habilidades
duras



Soft Skills/
Habilidades
blandas



Roles	Technical Lead	Transition Lead
Prerequisites	<ul style="list-style-type: none"> Should have spent a minimum of 3 years in the base roles (IT/Support) Should be well versed with SDLC and the different methodologies. Should hold the below mentioned competencies PI/Lean Six Sigma – E0; Knowledge Management – E0; Software Security - E0; Project Management – E0; Service Management – E0; Software Estimation - E0; Quality Management - E1 	
Key Responsibilities	Interacts with customers/users on operational aspects of project related to solution/service delivery, issues etc.	Prepares Pre-Due Diligence Questionnaire, collates pre- Due Diligence information and identifies gaps; Creates detailed transition plan, tracks progress and reports status on daily/weekly basis
	Leads the technical implementation and functions as a point of escalation for resolution of technical matters	Identifies training gaps and ensures completion of training activities. Escalates resource, delayed project phases and other planning issues to management on time.
	Ensures quality of requirement analysis, design, development and overall solutions and services being delivered; Ensures quality control through reviews, inspections and testing of the work items	Tracks progress and reports status of milestone tasks/secondary support activities to Transition Manager.; Identifies & implement process & technical improvements
Behavioural Attributes	Interpersonal Skill Customer Orientation Analytical Ability	Customer Orientation Coaching Analytical Ability

SOFTWARE DEVELOPMENT LIFE CYCLE

¿Cómo lo mejoro?

Ingenieros de mantenimiento



Producción y Mantenimiento del SW

INICIO

Requerimientos

¿Qué lo necesita?

Cliente



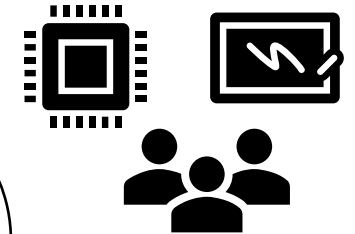
¿Cómo lo quiere?

Ing. Funcional



Análisis

¿Cómo lo diseño?



Arquitectos de Software/Líderes técnicos

Diseño

¿Cómo lo construyo?

Desarrolladores de Software



Codificación

Pruebas

¿Qué pruebo?

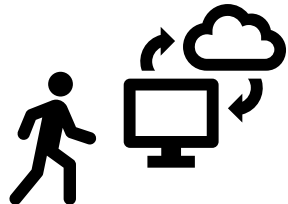
Ingenieros de QA Funcional y Técnico



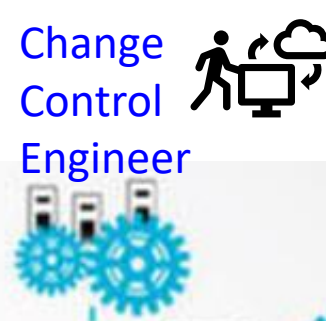
Despliegue

¿Cómo lo despliego?

Ingenieros de gestión y control de cambio



SDLC



IT Administrator
System Administrators
Database Administrators
Cloud infrastructure engineer

Client

- Business Analyst
- Functional Consultant

- Solution Architect
- Technical Leader

Developer

QA Engineer

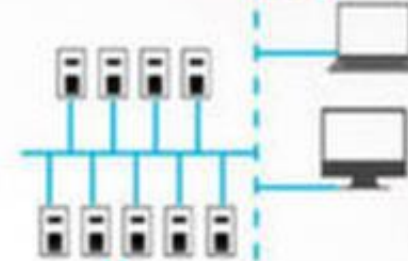
Change Control Engineer



Project Manager

Start

Finish



Project Kick-off

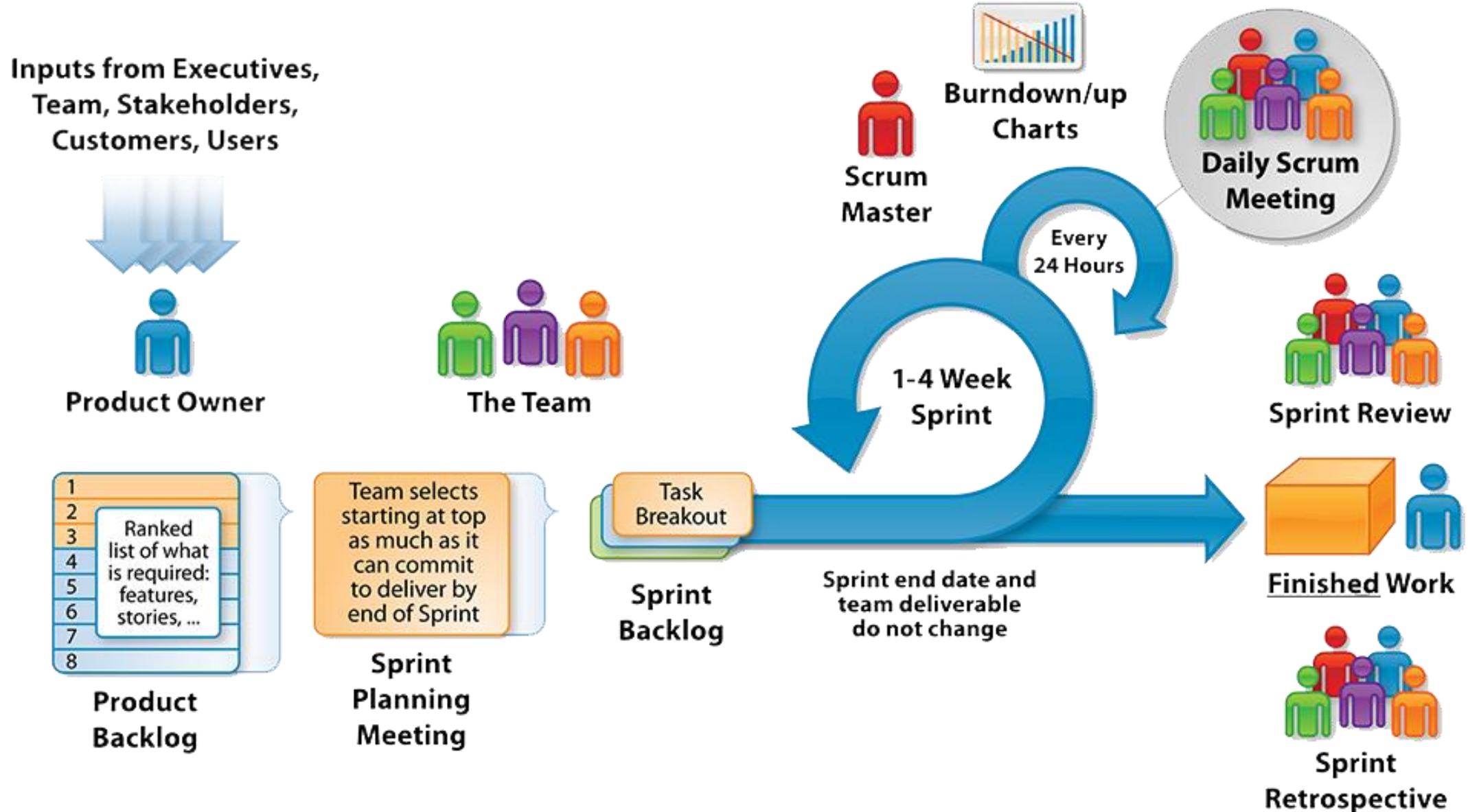
Project Closure



Project Planning

Timelines

AGILE WITH SCRUM FRAMEWORK



4 Agile Manifesto values

Individuals and interactions

over

Processes and tools

Working software

over

Comprehensive
documentation

Customer collaboration

over

Contract negotiation

Responding to change

over





Following a plan

TRADITIONAL VS AGILE

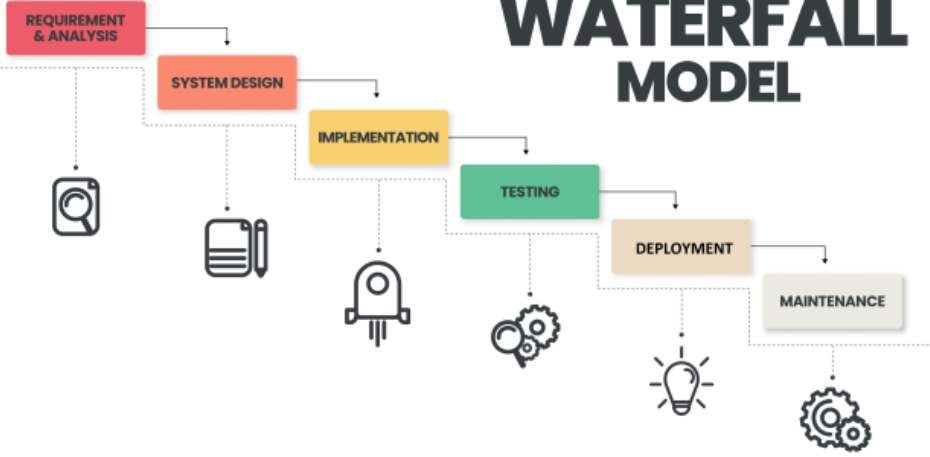
Parameter	Traditional Methods	Agile Methods
Ease of Modification	Hard	Easy
Development Approach	Predictive	Adaptive
Development Orientation	Process Oriented	Customer Oriented
Project Size	Large	Small or Medium
Planning Scale	Long Term	Short Term
Management Style	Command and Control	Leadership and Collaboration
Learning	Continuous Learning while Development	Learning is secondary to Development
Documentation	High	Low
Organization Type	High Revenue	Moderate and low Revenue
Organization's Number of Employees	Large	Small
Budget	High	Low
Number of Teams	Multiple	One
Team Size	Medium	Small

Waterfall vs Agile vs DevOps

demandsimplified.marketing

	 Waterfall	 Agile	 DevOps
Approach	Waterfall model provides a linear sequential approach to managing software projects. Each phase depends on deliverables from the previous one.	Agile is a highly dynamic and iterative approach where you do not need the complete set of requirements to start with. You can develop some features and check customer response before next steps	DevOps is an Agile methodology encompassing Development (Dev) and Operations (Ops). It enables end-to-end lifecycle delivery of features, fixes, and updates at frequent intervals.
Year	1970	2001	2009
Scope and Schedule	Adjust schedule to preserve scope. Fixed requirements. Limited flexibility.	Adjust scope to preserve schedule. Iterative approach allows for prioritization	Adjust scope to preserve schedule. Highly responsive to business needs
Time to Market	Slow	Rapid	Continuous
Automation	Low	Varied	High
Customer Feedback	End of Project	After every sprint	Continuous
Quality	Low. Issues are identified only at testing stage	Better. Issues are identified after every sprint	High. Automated unit testing during development
Collaboration	Teams operate in silos	Multiple teams are involved, but not all	All stakeholders are involved from start to finish
Variations	V-model	Scrum, XP, LeSS, SAFe	CI/CD, DevSecOps

WATERFALL MODEL



AGILE methodology

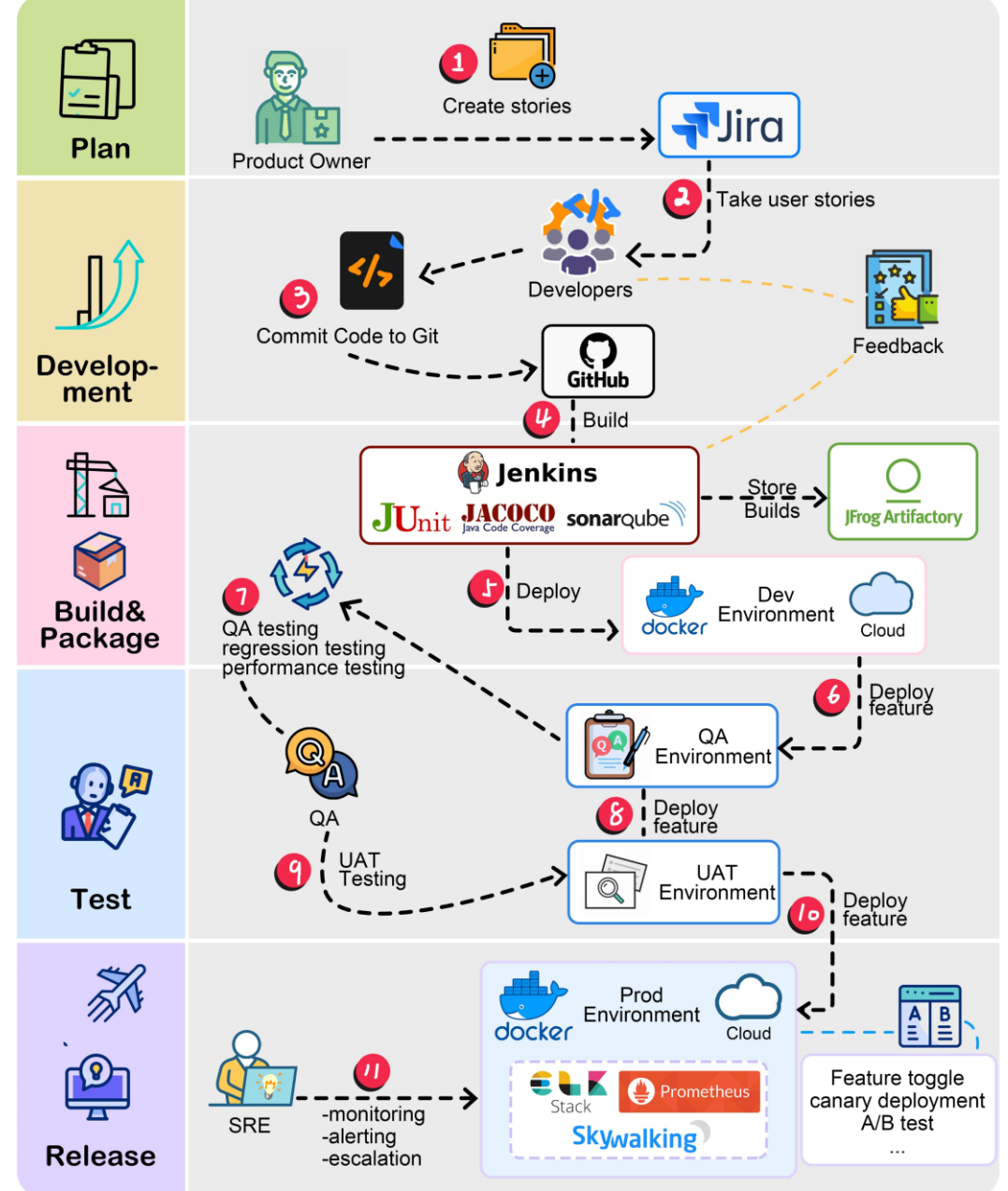


DevOps

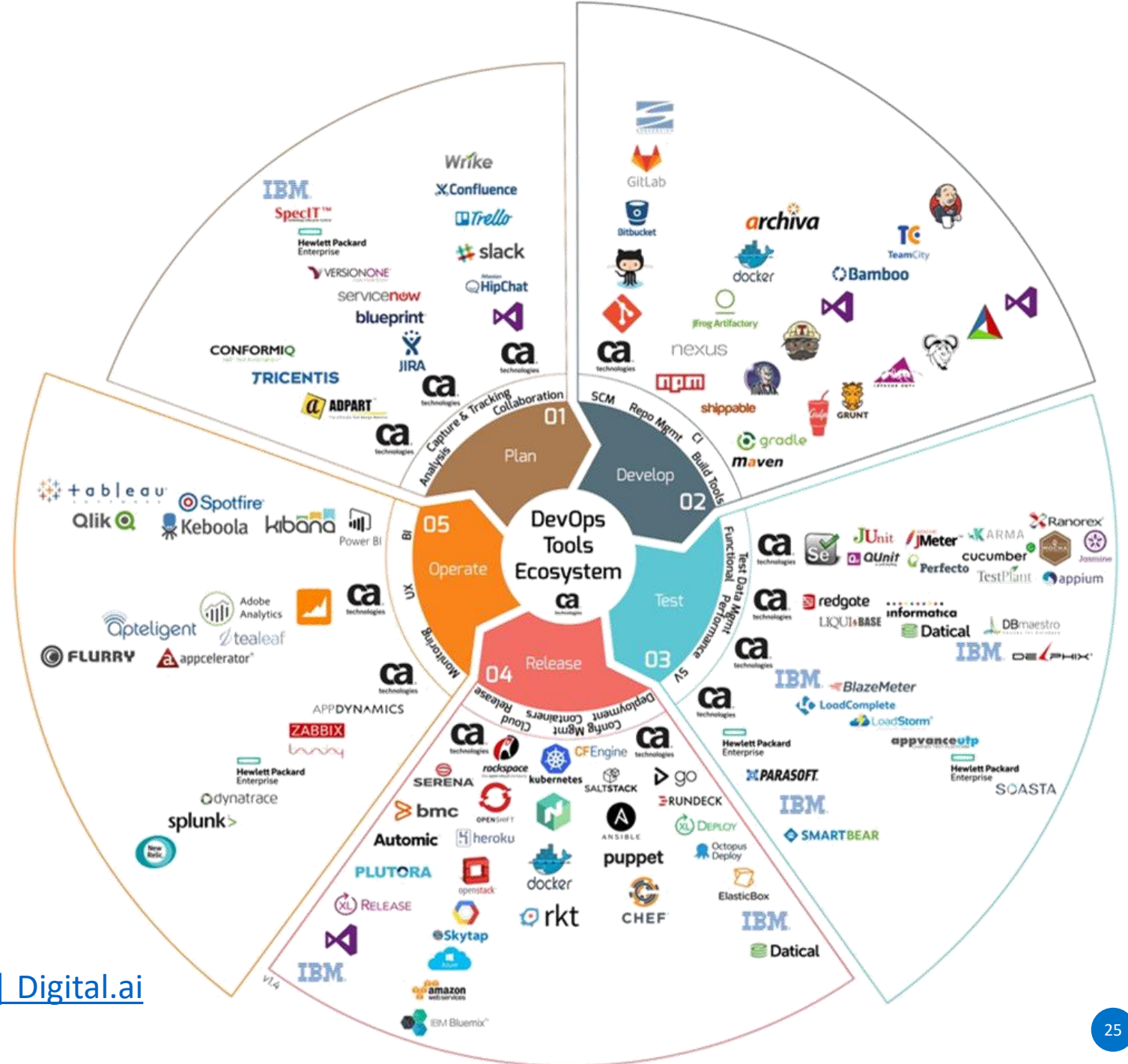


How Companies Ship Code To Production

blog.bytebytego.com

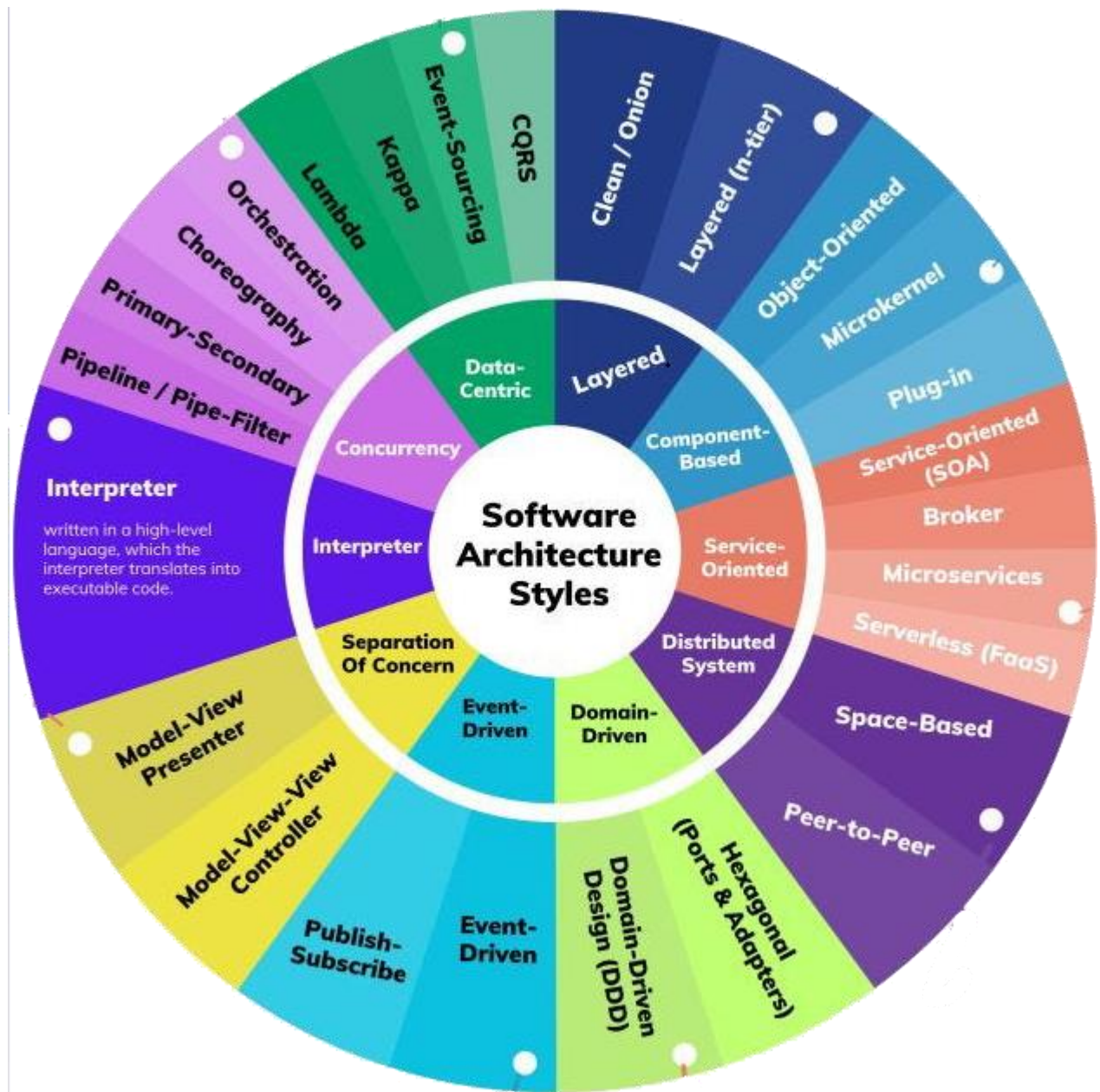
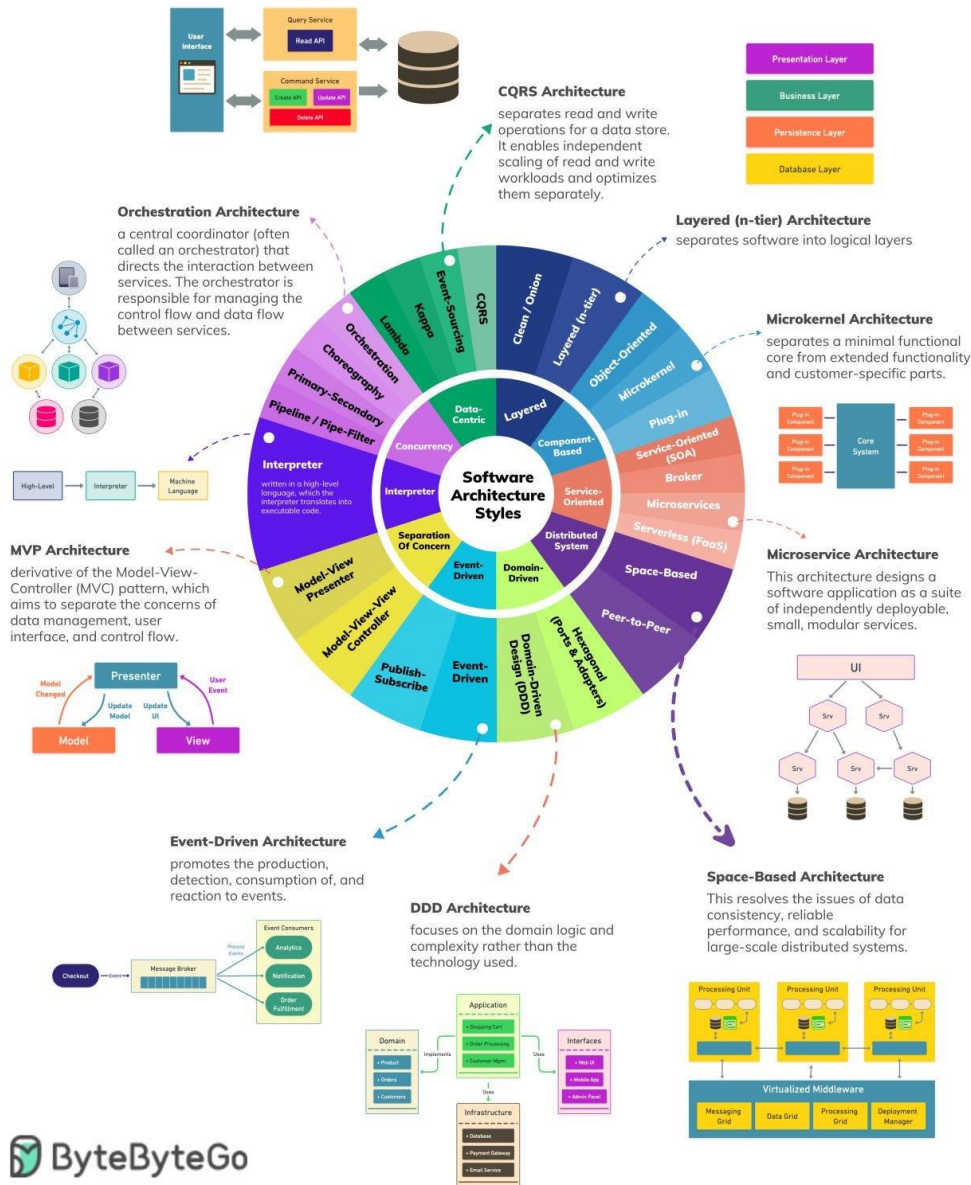


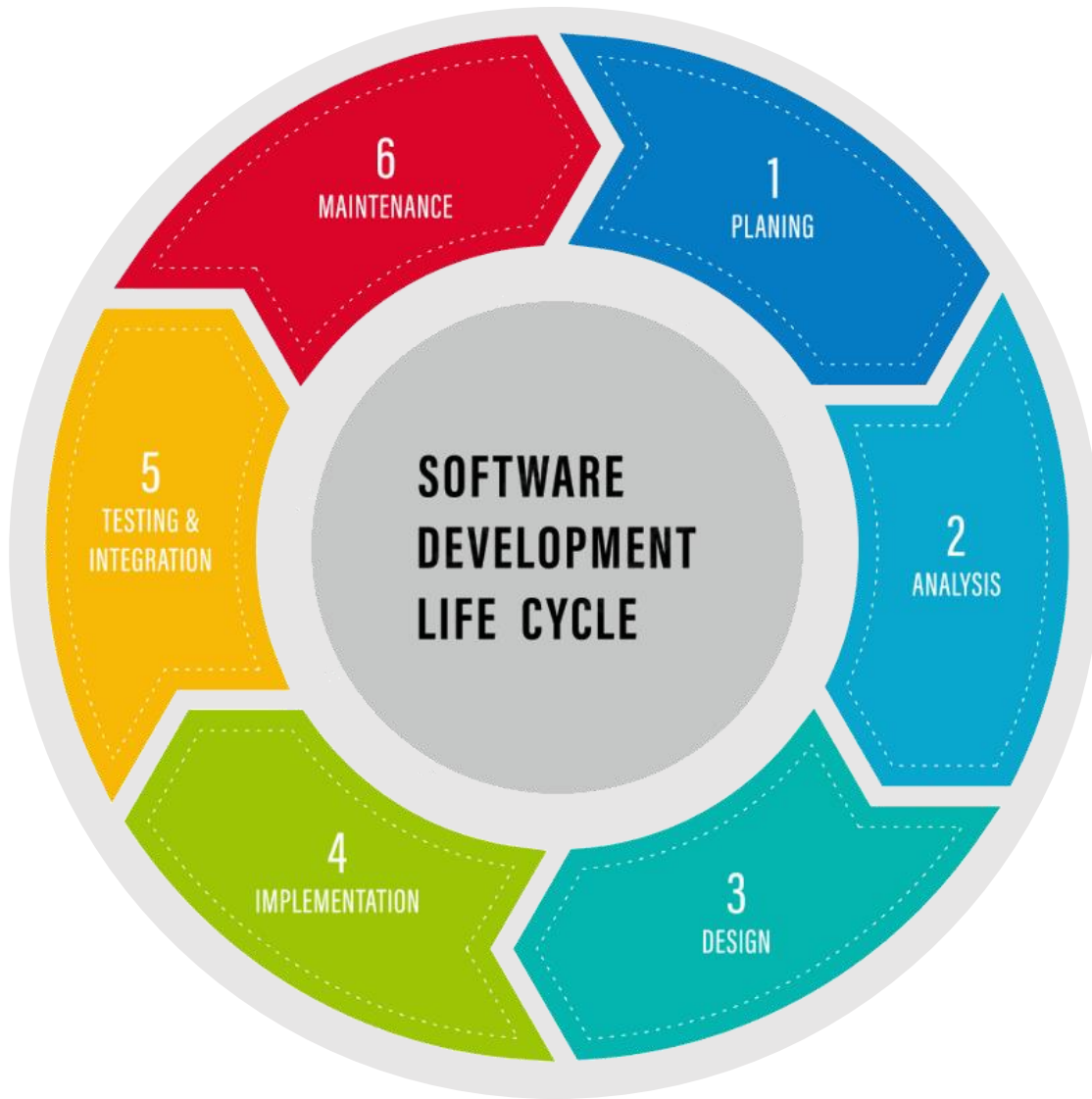
ECOSISTEMA DE HERRAMIENTAS



[Periodic Table of DevSecOps Tools | Digital.ai](#)

Software Architecture Styles





THANK YOU



VICTOR.VELEPUCHA@EPN.EDU.EC

