

# ARQUITECTURA DE SOFTWARE

Potenciando la industria del software

Ing. Zacarías Monzón

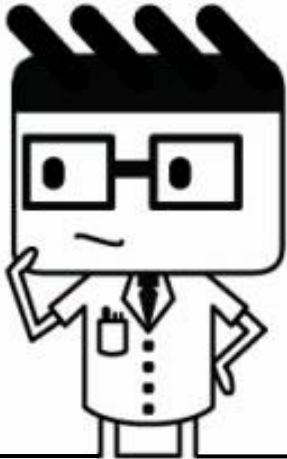
# ¿QUÉ ES LA ARQUITECTURA?



“Arquitectura es el conjunto de estructuras necesario para comprender un sistema, sus componentes, las relaciones entre ellos y las propiedades de ambos.”

# ¿QUE ES LA ARQUITECTURA?

“Architecture



is architecture

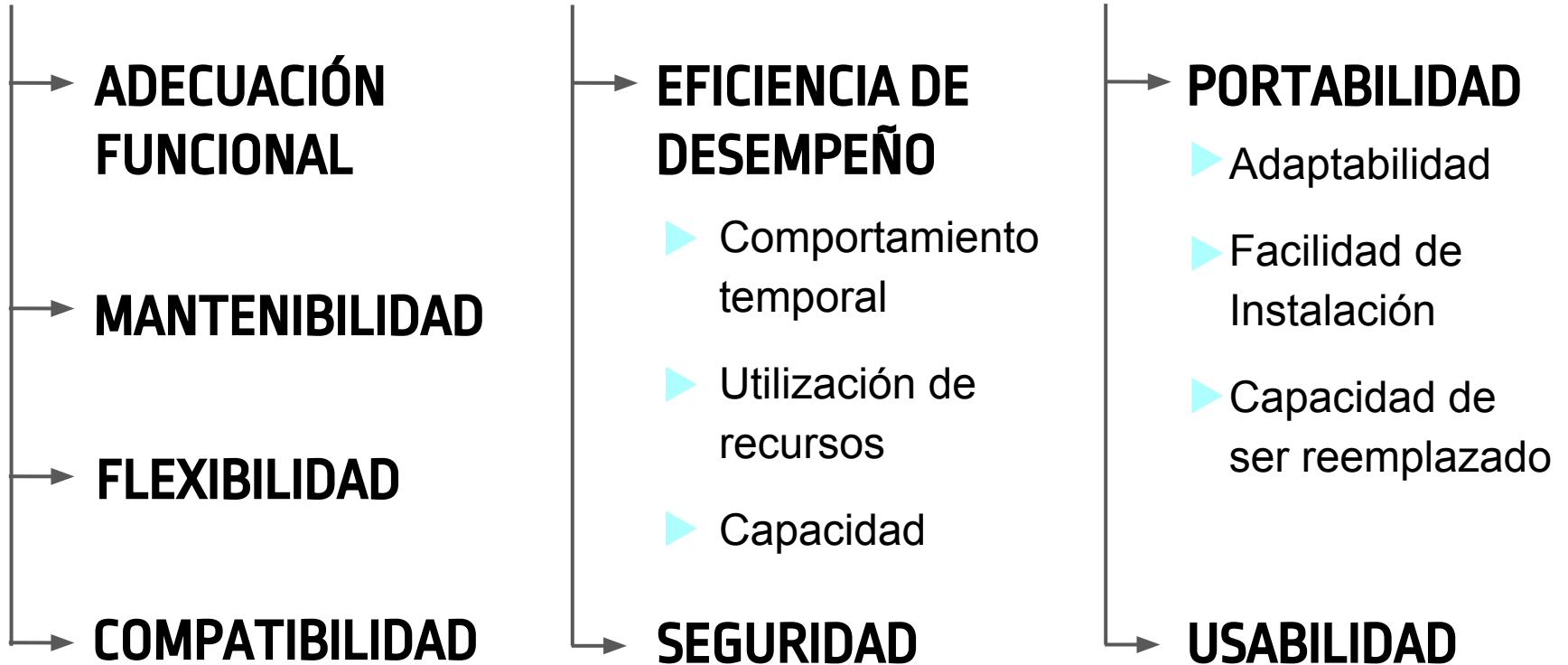


is architecture”

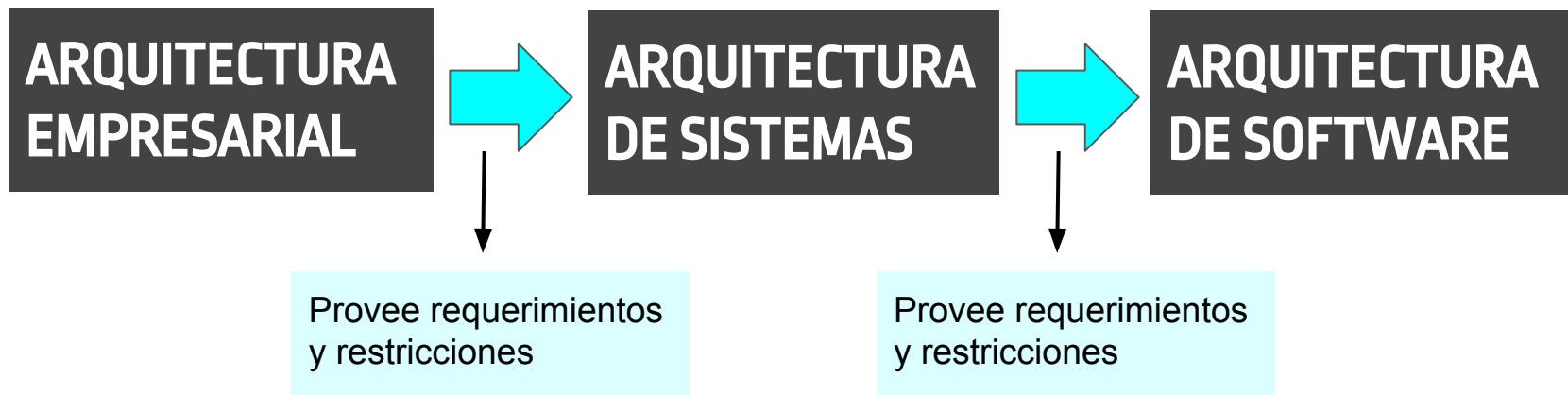


JOHN  
ZACHMAN

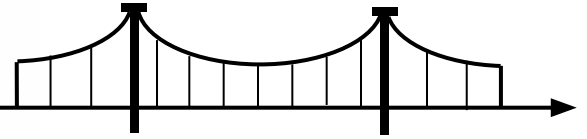
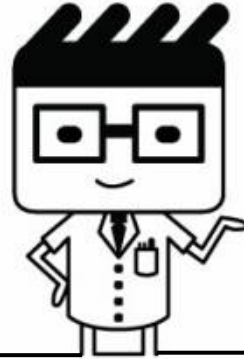
# ▶ ATRIBUTOS DE CALIDAD



# ¿QUÉ ES LA ARQUITECTURA?



# ¿QUÉ ES LA ARQUITECTURA?



“Los sistemas son construidos para satisfacer las necesidades de negocio. La arquitectura es un puente entre esos objetivos de negocio y el sistema resultante.”



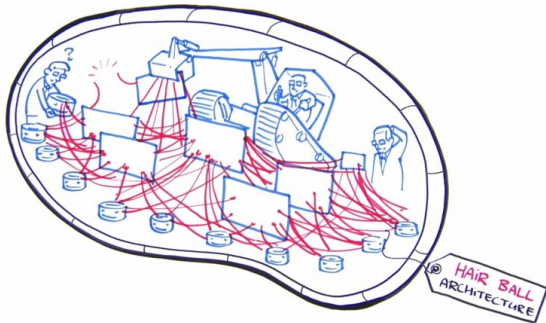
# ¿QUÉ ES LA ARQUITECTURA?

- Visión
- Estrategia
- Adquisiciones
- Requerimientos
- Arquitectura de negocios
- Arquitectura de servicios
- Arquitectura técnica
- Diseños
- Implementación
- Test
- Deploy



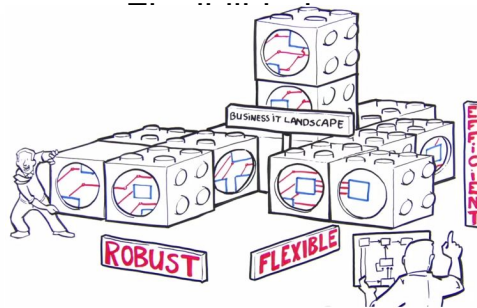
# SABORES DE ARQUITECTURA

## VERTICAL DE INDUSTRIA



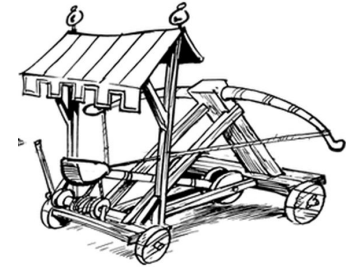
Orientado a la integración

## DESARROLLO DE PRODUCTOS



Orientado a perdurar en el tiempo, calidad, diversidad de clientes.

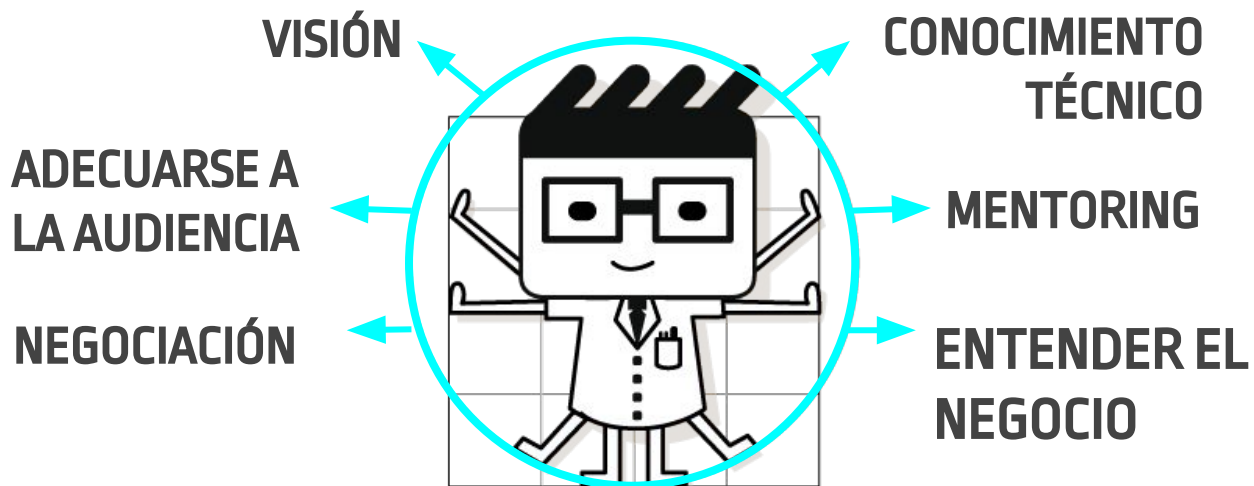
## SERVICIOS TECNOLÓGICOS



Responder con los compromisos adquiridos con el cliente, mantener la marca, reducir los costos



# HABILIDADES DEL ARQUITECTO



ARQ. EMPRESARIAL

ARQ DE SISTEMAS

ARQ DE SOFTWARE

VERTICAL DE INDUSTRIA - DESA. DE PRODUCTOS - SERV. TECNOLÓGICOS

# PRESENTANDO LA ARQUITECTURA

- Habilita o inhabilita atributos de calidad
- Brinda soporte a decisiones de alto impacto
- Facilita la gestión del cambio
- Reduce la curva de aprendizaje
- Predicción temprana de atributos de calidad

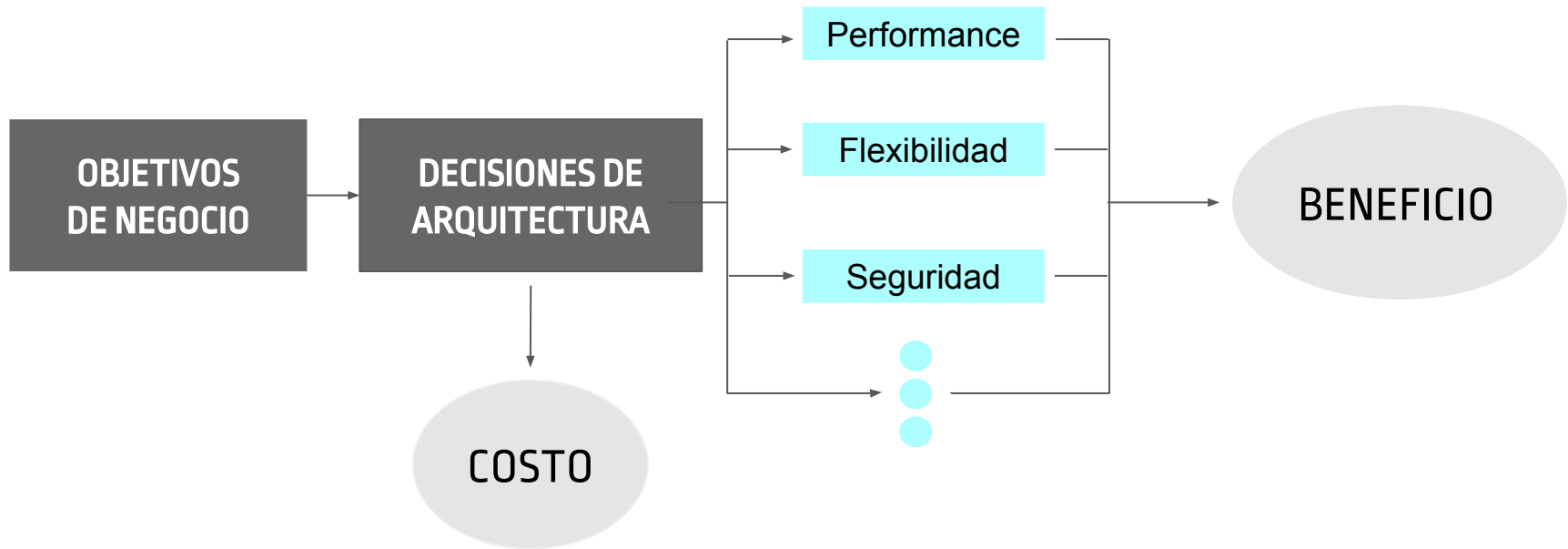
- Reduce la complejidad
- Es la base de la línea de producción
- Permite una mejor gestión de costo y tiempo
- Define restricciones para el diseño
- Facilita la comunicación entre stakeholders

*“Amateurs want to be right.  
Professionals want to make money.”*

*Alan Greenspan*



# ANALISIS ECONOMICO DE LA ARQUITECTURA



# ALGUNAS VARIABLES

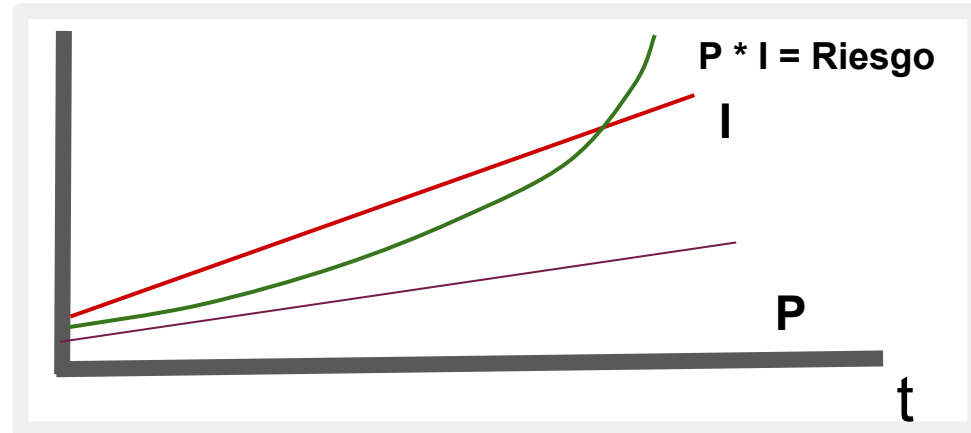
Reducción de errores  $x\%$

Esfuerzos de arquitectura  $\$l$

Reutilización en futuros proyectos  $\$z$  / proyecto.

Impacto en los tiempos de desarrollo  $\pm t\%$

Costo de oportunidad asociado  $\$c$





# DEUDA TÉCNICA

## Integración por JDBC a aplicación legada


- **Motivo:** velocidad de desarrollo, falta de recursos.
- **Riesgo:** acoplamiento de modelos, reimplementación ante nueva demanda.

## Desarrollo de adecuaciones no estándares

- **Motivo:** velocidad de desarrollo, falta de conocimiento tecnológico
- **Riesgo:** costo de migración a nuevas versiones.



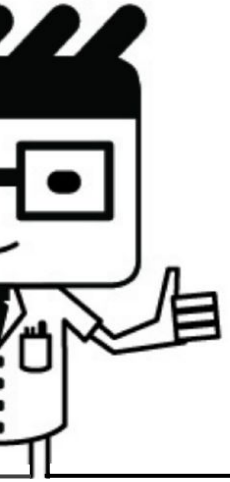
# SOFTWARE PRODUCT LINES



Pertencen al mismo dominio de aplicaciones.

Comparten una arquitectura

Basados en componentes y servicios.

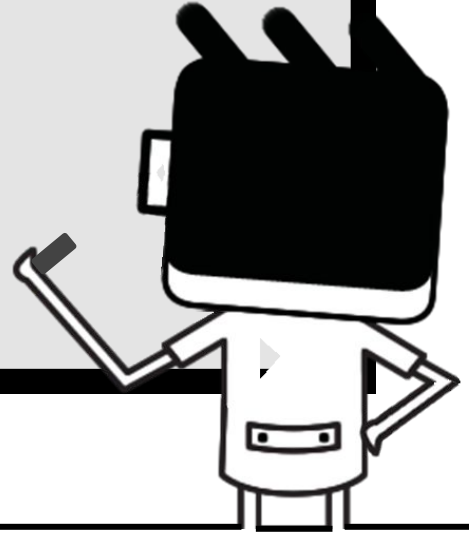


Obtienen una ventaja económica en los elementos comunes.

Comparten el impacto



# La ciencia detrás del significado

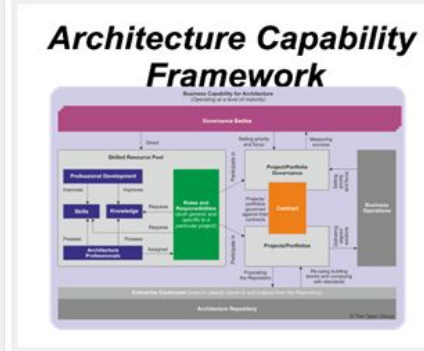
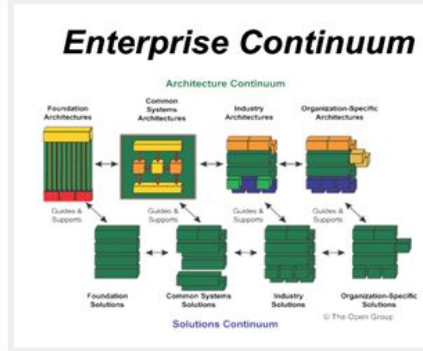
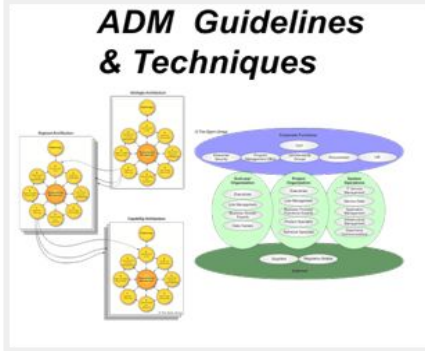
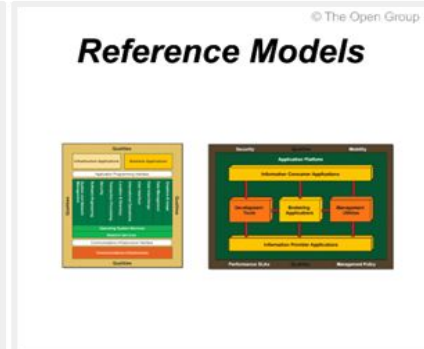
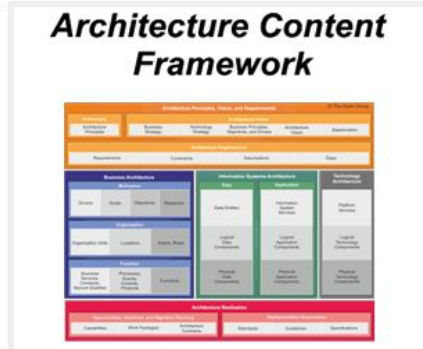




# Zachman

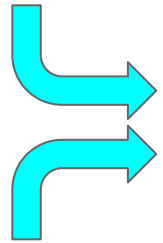
	Why	How	What	Who	Where	When
Contextual	Goal List	Process List	Material List	Organisational Unit & Role List	Geographical Locations List	Event List
Conceptual	Goal Relationship	Process Model	Entity Relationship Model	Organisational Unit & Role Relationship Model	Locations Model	Event Model
Logical	Rules Diagram	Process Diagram	Data Model Diagram	Role Relationship Diagram	Locations Diagram	Event Diagram
Physical	Rules Specification	Process Function Specification	Data Entity Specification	Role Specification	Location Specification	Event Specification
Detailed	Rules Details	Process Details	Data Details	Role Details	Location Details	Event Details

# TOGAF



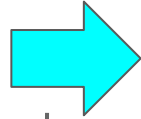
# ANÁLISIS Y DISEÑO DE ARQUITECTURA

Business Drivers



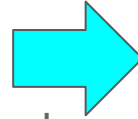
QAW

Architectural Plan



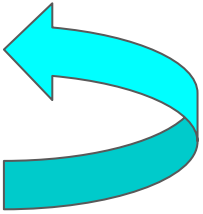
ADD

Quality Attributes



ATAM

Architecture design



Risk

# Tácticas

## DISPONIBILIDAD

### DETECTAR FALLAS

- ▶ Ping / Echo
- ▶ Heartbeat
- ▶ TimeStamp
- ▶ Sanity Checking
- ▶ Voting
- ▶ Exception detection
- ▶ Auto-Test

### RECUPERACIÓN DE FALLAS

- ▶ Redundancia activa
- ▶ Redundancia pasiva
- ▶ Spare
- ▶ Exception handling
- ▶ Rollback
- ▶ Retry
- ▶ Non-stop forwarding

### PREVENIR FALLAS

- ▶ Transacciones
- ▶ Modelo predictivo
- ▶ Incrementar conjunto de competencias



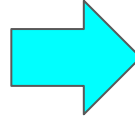
# CBAM

Determinar el Value for Cost ( $VFC = B_i / C_i$ ) utilizando:

- Curvas de utilidad
- Peso de escenarios
- Efectos colaterales
- Normalización de beneficios por escenario

# Arquitectura & Agile

~~Agile vs Architecture~~



¿Cuánta arquitectura?

Fase preliminar

Creación de backlog

Fase inicial de arquitectura

Sprint

Planificación

Construcción

Implementación

Retrospectiva

QAW

ADD

ATAM

ADD

ATAM

“La tecnología es una pasión pero estamos acá porque también es un negocio.”



Decisiones - Errores - Eficiencia

***Muchas gracias!!!***



@zacariasmonzon