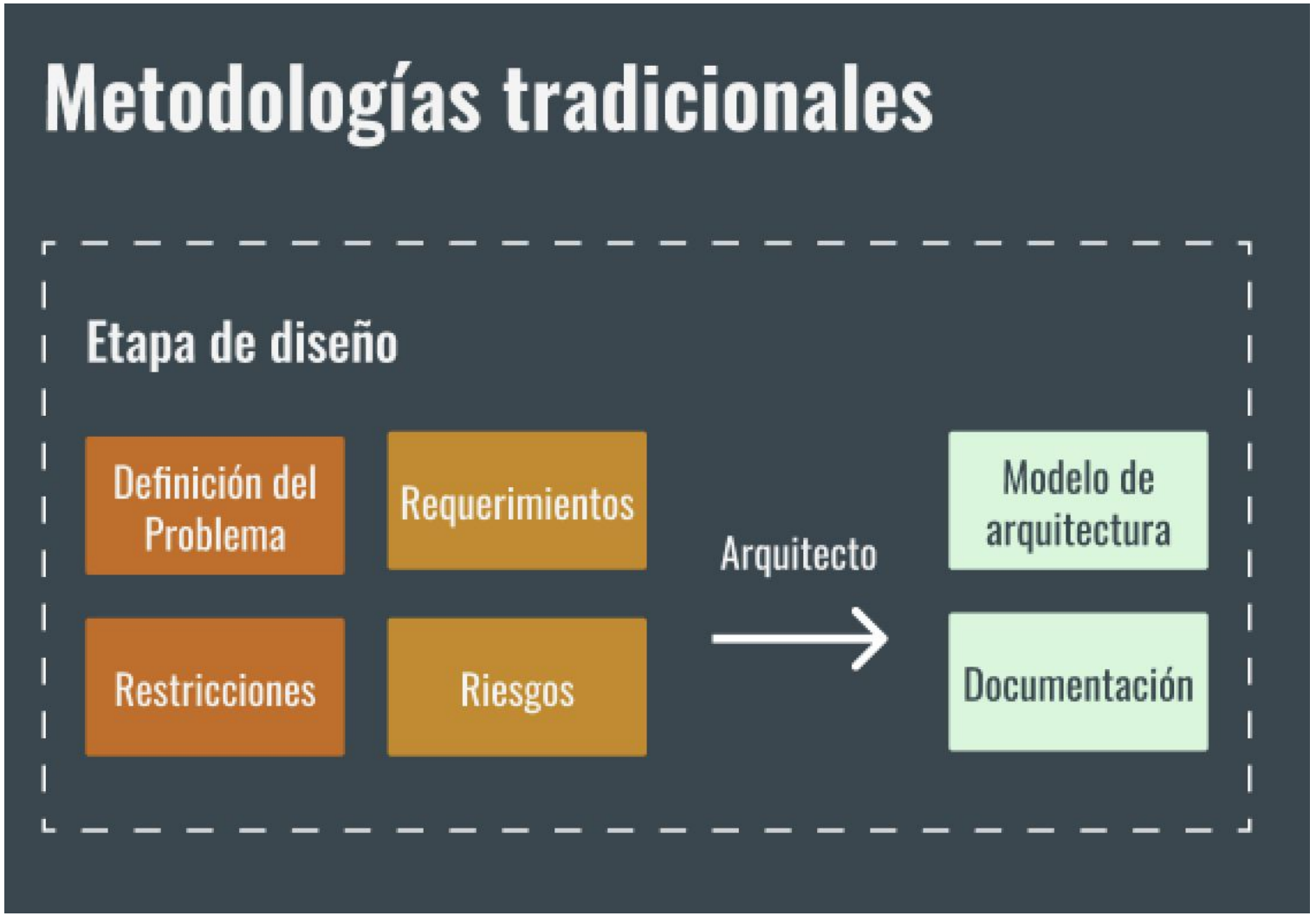
Atributos de Calidad Software Avanzado 2024

Marco Tulio Aldana Prillwitz

Metodología y Arquitectura

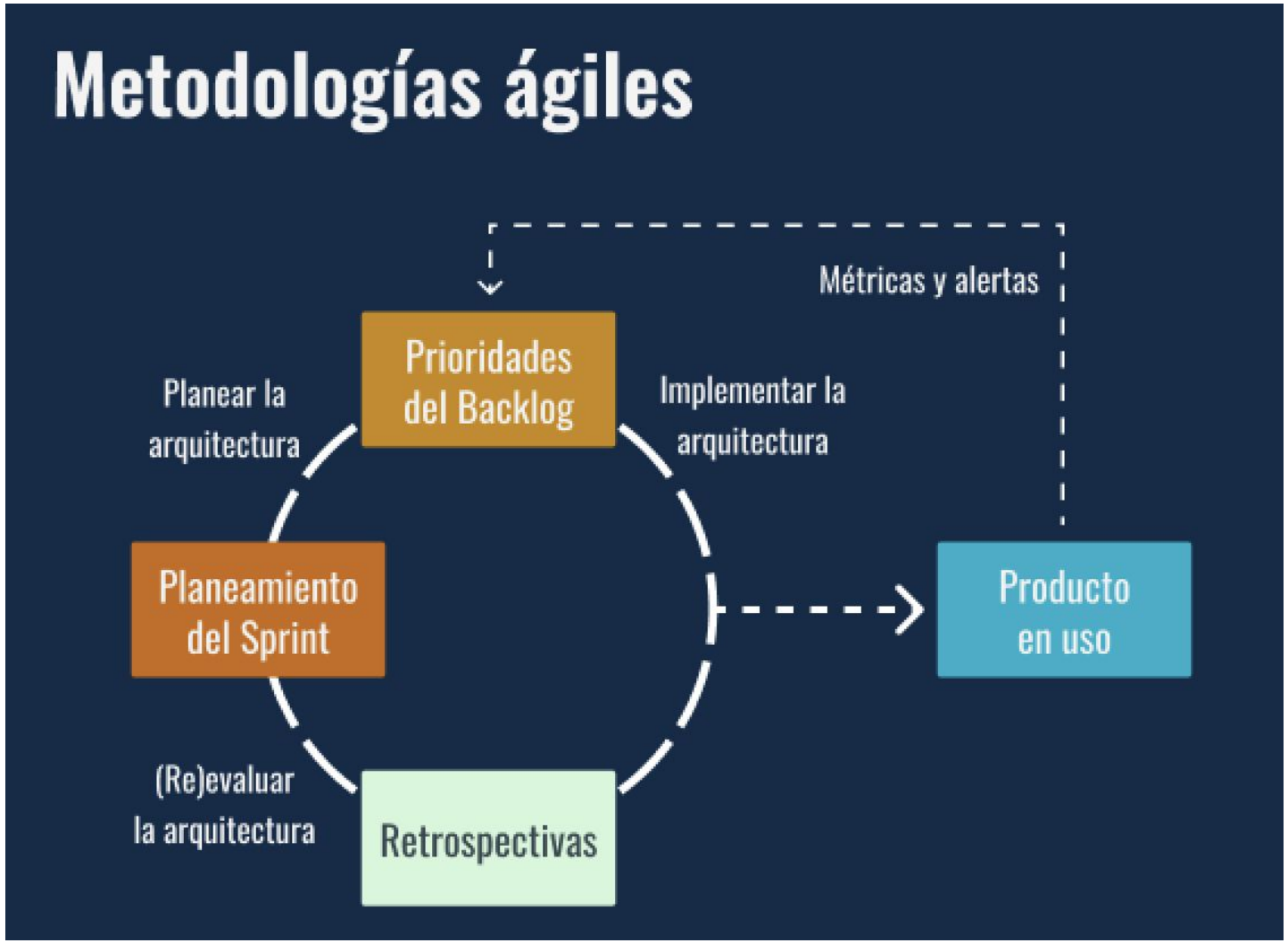
- El arquitecto diseña una solución basado en :
  - Requerimiento
  - Restricciones
  - Riesgos

- El objetivo es encontrar los problemas y diseñar una solución que minimice o elimine esos problemas, recuerden que en esta metodología aún no tenemos feedback del usuario.



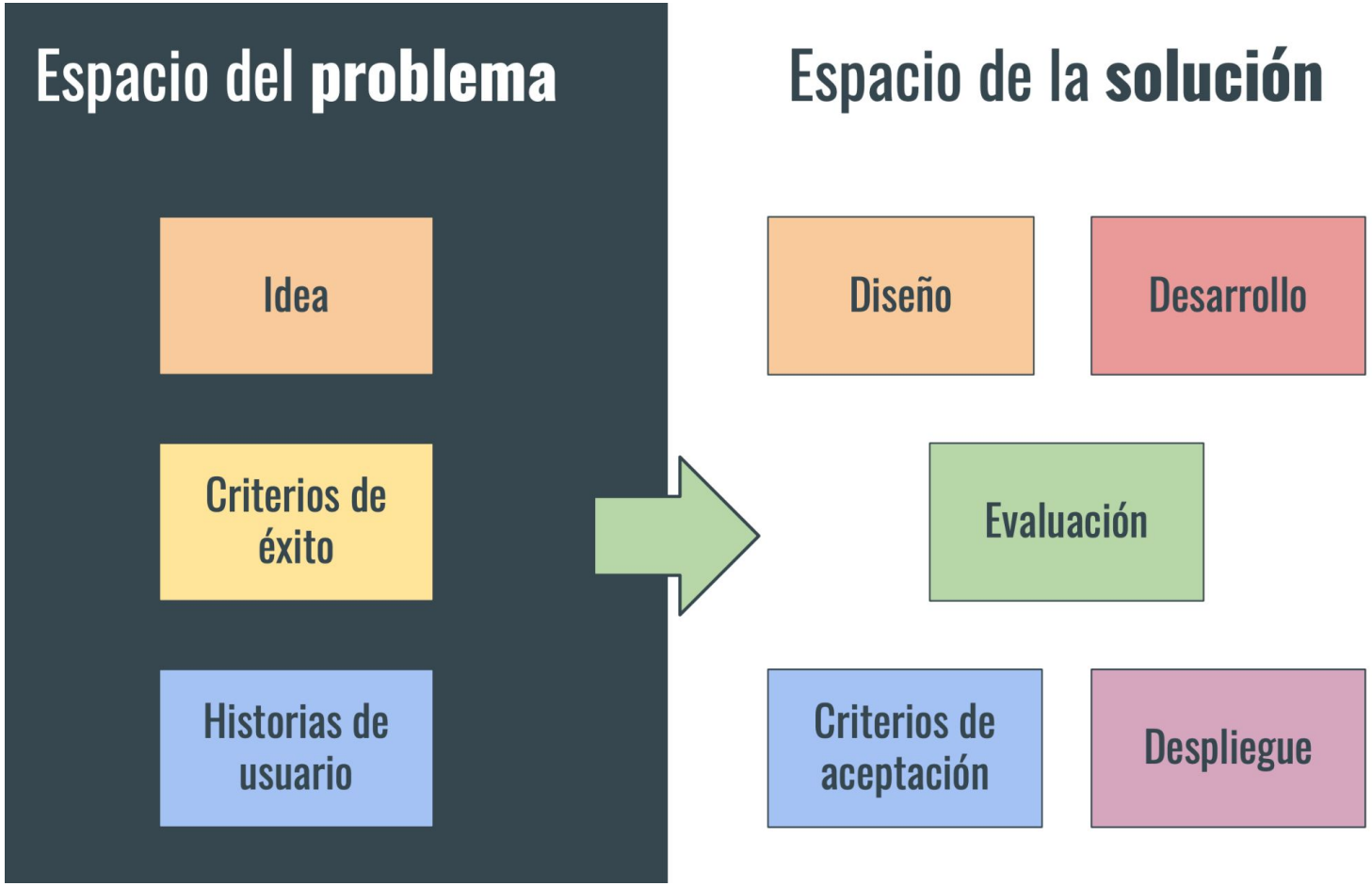
Metodología y Arquitectura

- La arquitectura emerge de un equipo autogestionado que realiza el diseño de una solución evolutiva de sprint a sprint lo que da lugar a reevaluar las decisiones tomadas.
- El objetivo es que nuestra arquitectura alcance el ideal dependiendo de nuestro esquema de requerimientos de forma evolutiva.



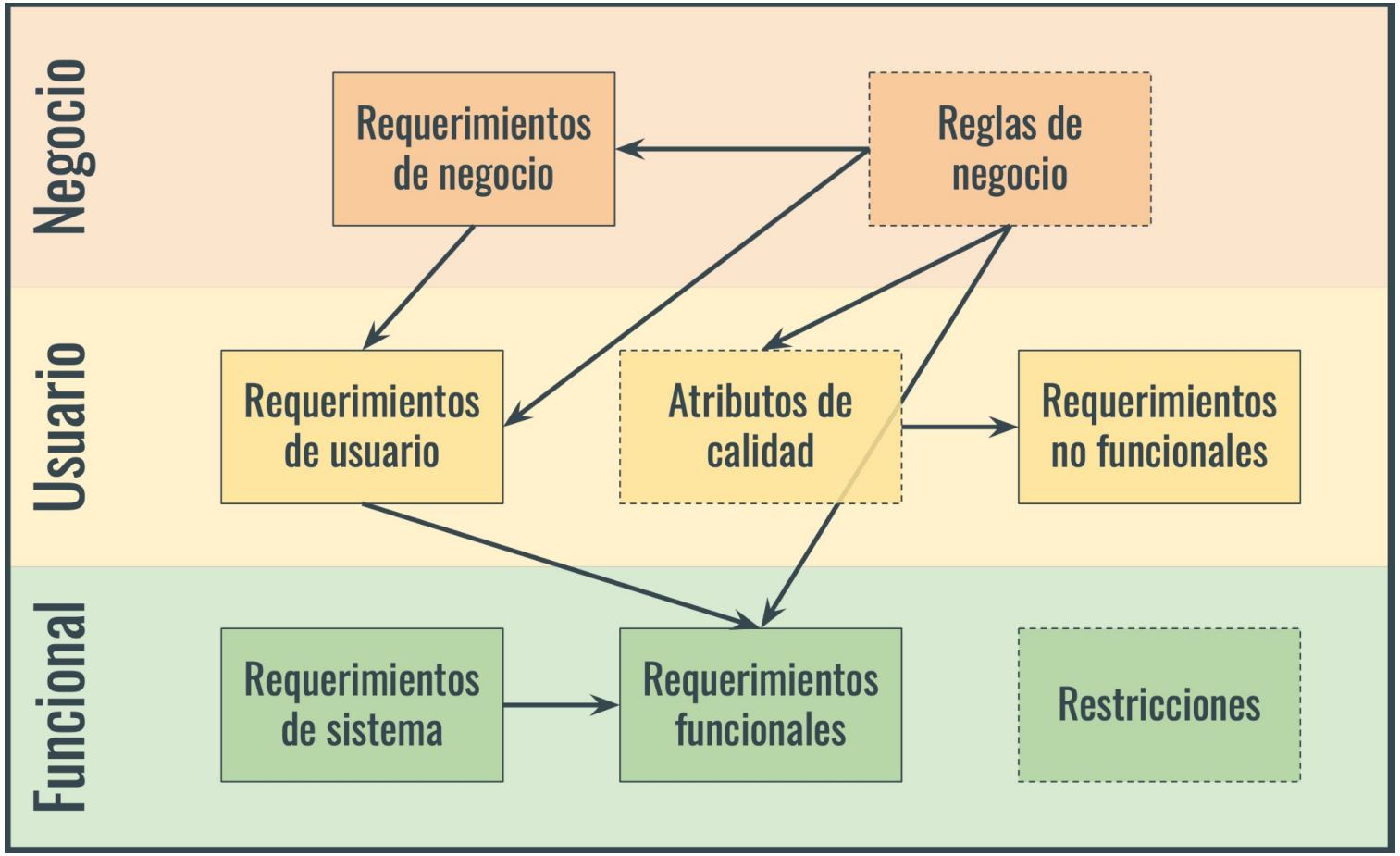
Entender el Problema

- El objetivo de entender el problema es obtener el conocimiento necesario para poder comunicarse con los stakeholders, comprender el negocio, las necesidades y modelar una solución adecuada.
- El objetivo de la solución es lograr un diagrama detallado que cubra todas las criterios de aceptación de los stakeholders.



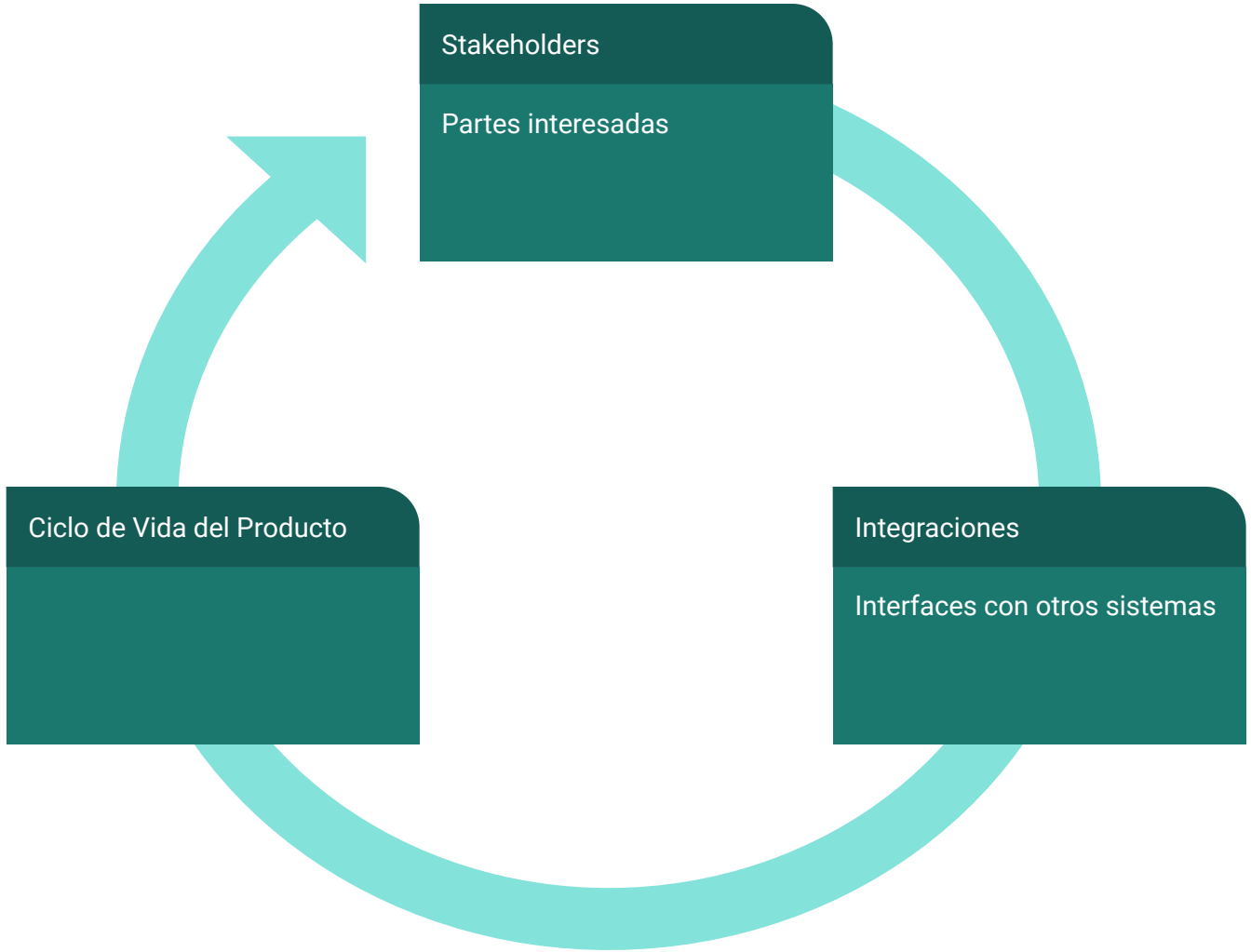
Gestión de Requerimientos

- Requerimientos de producto : dados por los stakeholders
- Requerimientos de proyecto:
  - Fechas de entrega
  - Planes
  - Equipos de trabajo



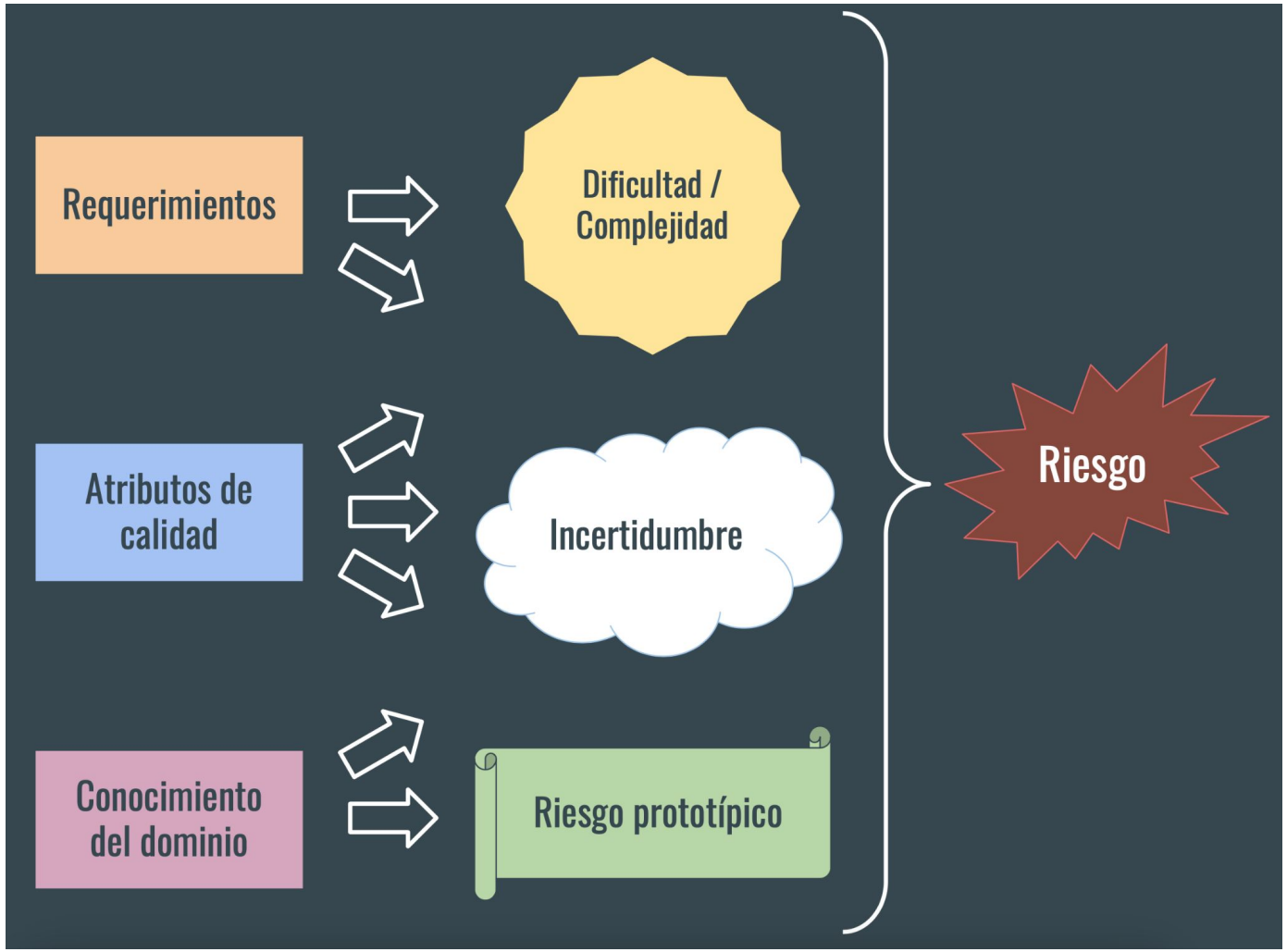
Gestión de Restricciones

- La arquitectura emerge de un equipo autogestionado que realiza el diseño de una solución evolutiva de sprint a sprint lo que da lugar a reevaluar las decisiones tomadas.



Gestión de Riesgos

- Gestión de riesgos :
  - Identificación de riesgos
  - Análisis de riesgos
  - Categorización de riesgos
  - Planificación de mitigación de riesgos
- Matriz de Impacto/Prioridad
  - Riesgo
  - Impacto
  - Prioridad



Atributos de Calidad

- También conocidos como características de calidad o requisitos no funcionales, son aspectos importantes que afectan la eficacia, la eficiencia y la confiabilidad de un sistema de software, pueden ser :
  - Usabilidad
  - Rendimiento
  - Disponibilidad
  - Confiabilidad
  - Mantenibilidad
  - Portabilidad
  - Seguridad
  - Eficiencia
  - Escalabilidad
  - Interoperabilidad

Usabilidad

- La facilidad con la que los usuarios pueden interactuar con el sistema. Incluye aspectos como la accesibilidad, la claridad de la interfaz de usuario y la facilidad de aprendizaje. Está regida por la ISO 9241-11
- Aspectos claves :
  - Facilidad de aprendizaje
  - Eficiencia de uso
  - Memorabilidad
  - Prevención de errores
  - Satisfacción de usuario
  - Diseño centrado en el usuario
  - Accesibilidad

Rendimiento

- Se refiere a la eficiencia y la capacidad de un sistema para responder a las solicitudes de los usuarios de manera rápida y eficaz, manteniendo un nivel adecuado de utilización de recursos como la CPU, la memoria y la red.
- Aspectos claves :
  - Tiempo de respuesta
  - Latencia
  - Carga máxima
  - Cache y almacenamiento en memoria.

Disponibilidad

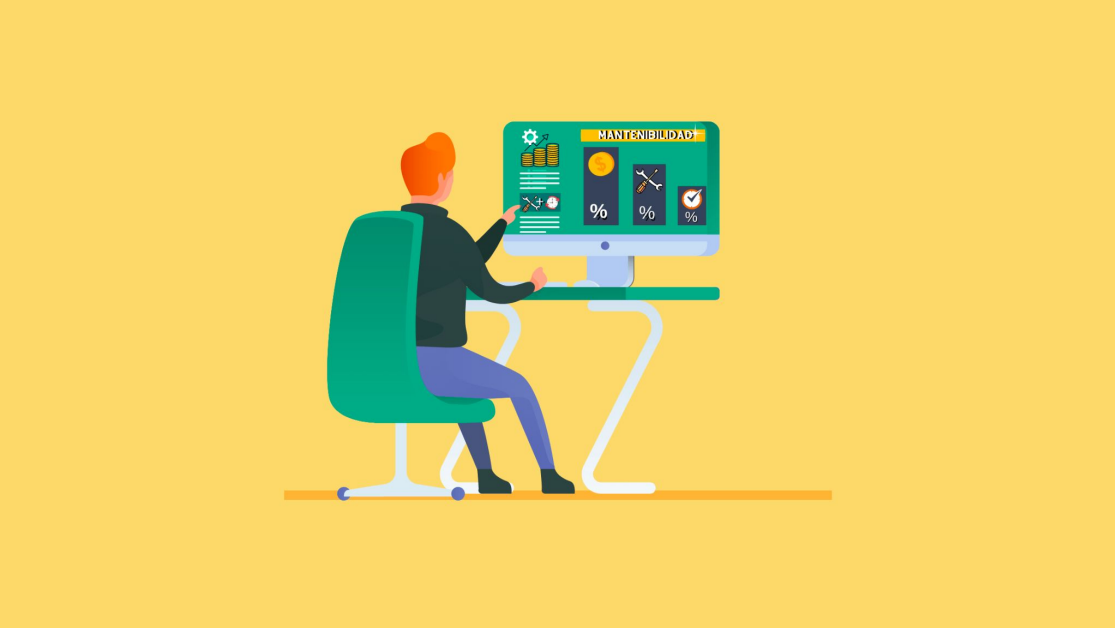
- Se refiere a la capacidad de un sistema para estar operativo y accesible cuando los usuarios lo necesitan.
- Este atributo mide la confiabilidad y la continuidad del servicio, asegurando que el software esté disponible y funcional incluso en situaciones adversas.
- Aspectos claves:
  - Tiempo de inactividad
  - Tolerancia a fallos
  - Respuesta a incidentes
  - Gestión de downtime

Confiabilidad

- Se refiere a la capacidad de un sistema para realizar sus funciones de manera precisa y consistente a lo largo del tiempo, bajo diversas condiciones y situaciones.
- La confiabilidad implica que el software no experimente fallas inesperadas, errores críticos o comportamientos no deseados, y que cumpla con las expectativas del usuario de manera consistente.
- Aspectos claves :
  - Prevención de errores
  - Manejo de excepciones
  - Recuperación ante fallos

Mantenibilidad

- Se refiere a la facilidad con la cual un sistema de software puede ser modificado, actualizado, reparado y ampliado. Un software fácilmente mantenible es crucial para reducir los costos de desarrollo a lo largo del tiempo, facilitar la introducción de nuevas características y corregir problemas.
- Aspectos claves :
  - Legibilidad del código, modularidad
  - Reusabilidad, facilidad de pruebas
  - Documentación, desacoplado tecnologicamente
  - Facilidad de depuración, gestión de cambios



Portabilidad

- Se refiere a la capacidad de un sistema de software para funcionar eficazmente en diferentes entornos, plataformas y configuraciones, sin requerir modificaciones significativas. Un software portátil es aquel que puede ser fácilmente transferido y ejecutado en diversos contextos sin pérdida de funcionalidad.
- Aspectos claves :
  - Compatibilidad
  - Independencia
  - Empaquetado y distribución
  - Configurabilidad

Seguridad

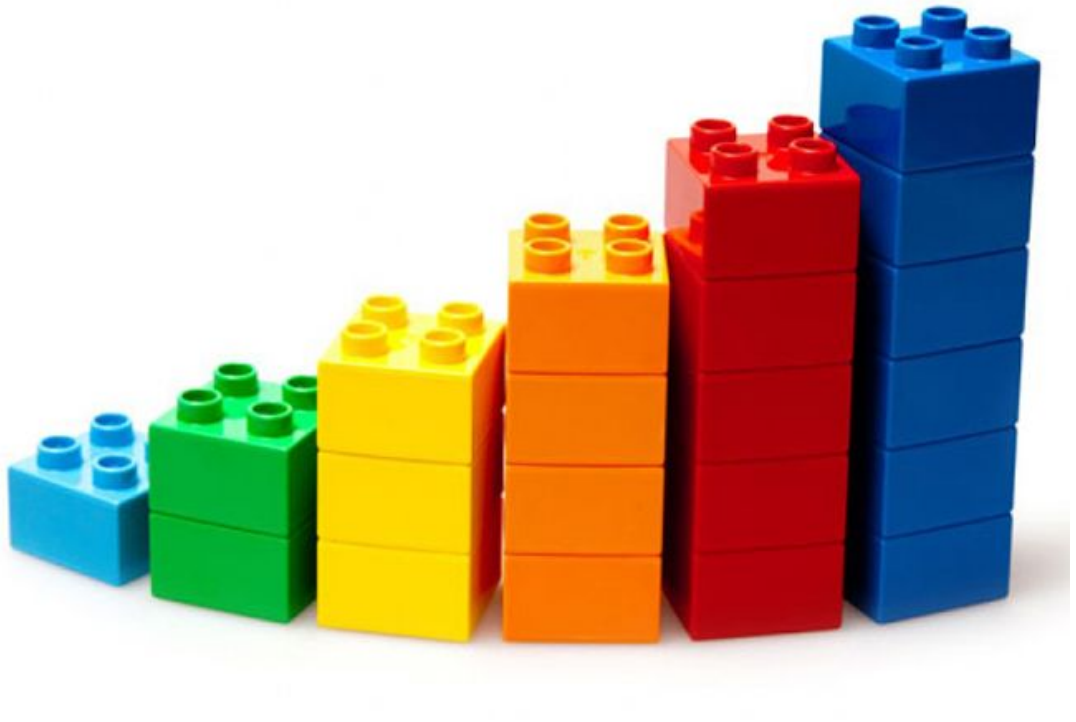
- Se refiere a la capacidad de un sistema para proteger los datos, recursos y funcionalidades contra amenazas y riesgos de seguridad. Un software seguro es aquel que ha sido diseñado e implementado de manera que minimiza las vulnerabilidades y proporciona mecanismos efectivos para detectar, prevenir y responder a posibles ataques o eventos de seguridad no deseados.
- Aspectos claves :
  - Confidencialidad
  - Integridad
  - Autenticación
  - Auditoría

Eficiencia

- Se refiere a la capacidad de un sistema para utilizar sus recursos de manera óptima, logrando un rendimiento máximo y respondiendo a las solicitudes de los usuarios de manera rápida y eficaz.
- Aspectos claves :
  - Optimización de recursos
  - Optimización de código
  - Algoritmos eficientes
  - Carga y tiempo de inicio
  - Optimización de consultas

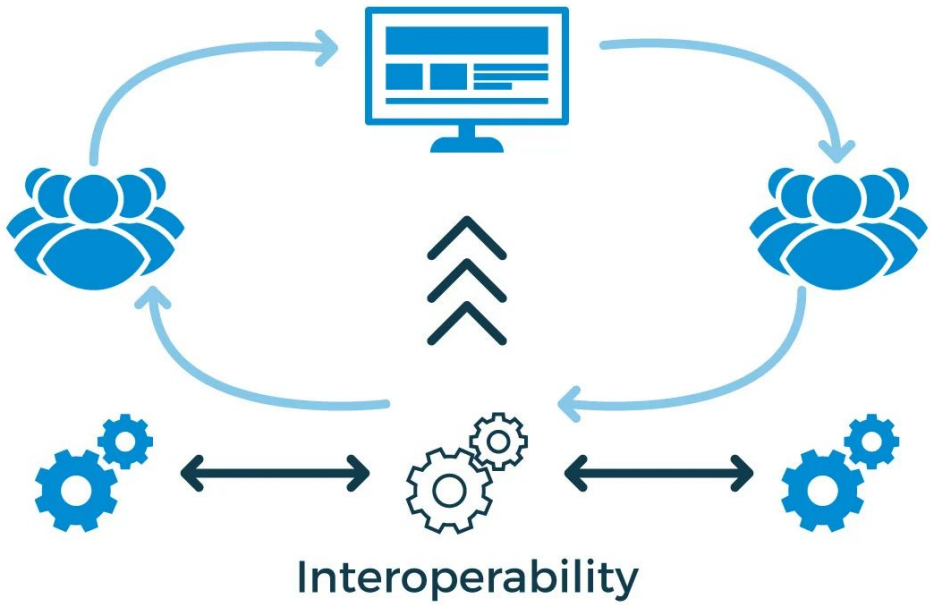
Escalabilidad

- Se refiere a la capacidad de un sistema para manejar un aumento en la carga de trabajo, el volumen de datos o el número de usuarios, sin degradación significativa del rendimiento y sin requerir cambios sustanciales en la arquitectura o infraestructura.
- Aspectos claves :
  - Escalabilidad horizontal o vertical
  - Desempeño consistente
  - Particionamiento
  - Base de datos escalable
  - Gestión de sesiones
  - Gestión de transacciones distribuidas



Interoperabilidad

- La capacidad de interoperar con sistemas de gestión de identidad y autenticación para permitir el acceso seguro a recursos compartidos.
- Aspectos claves :
  - Compatibilidad en formatos
  - API abiertas
  - Estandares de la industria
  - Integración de sistemas externos
  - Mensajería y comunicación
  - Compatibilidad de plataformas



Bibliografía

- "Enterprise Architecture as Strategy: Creating a Foundation for Business Execution" (Arquitectura empresarial como estrategia: creando una base para la ejecución empresarial) de Jeanne W. Ross, Peter Weill y David C. Robertson.
- "Enterprise Architecture at Work: Modelling, Communication and Analysis" (Arquitectura empresarial en acción: modelado, comunicación y análisis) de Marc Lankhorst.
- "The Open Group Architecture Framework (TOGAF®) Version 9.2" (El marco de trabajo de arquitectura de The Open Group (TOGAF®) Versión 9.2).
- "Enterprise Architecture Planning: Developing a Blueprint for Data, Applications, and Technology" (Planificación de arquitectura empresarial: desarrollo de un plan para datos, aplicaciones y tecnología) de Steven H. Spewak y Michael J. Tiemann.
- "An Introduction to Enterprise Architecture: Third Edition" (Una introducción a la arquitectura empresarial: tercera edición) de Scott A. Bernard.
- "The Zachman Framework for Enterprise Architecture: A Primer on Enterprise Engineering and Manufacturing" (El marco de trabajo de Zachman para arquitectura empresarial: una introducción a la ingeniería y fabricación empresarial) de John A. Zachman.
- "Building Enterprise Architecture: How to Define, Design, and Deliver a Successful EA" (Construyendo arquitectura empresarial: cómo definir, diseñar y entregar una EA exitosa) de Melissa Cook, S. Jane Fritz y Peter Matthijssen.
- "Enterprise Architecture Body of Knowledge (EABOK)" (Cuerpo de conocimiento de arquitectura empresarial) del Architecture and Governance Magazine.
- "No Silver Bullet: Essence and Accidents of Software Engineering" fue escrito por Frederick P. Brooks Jr. y fue publicado por primera vez en 1986.