



Módulo 2

Diseño y Arquitectura

Semana 1

Arquitectura de Software

- La arquitectura de un sistema de software es la estructura o estructuras del sistema, que comprenden los elementos de software, sus propiedades visibles externamente y las relaciones entre ellos



Arquitectura de software

Definiciones

- Elemento de arquitectura de software (element): Un elemento de la arquitectura de software es una pieza fundamental a partir de la que se puede construir un sistema
- Involucrado (stakeholder): En la arquitectura de software es una persona, grupo o entidad con un interés o preocupación sobre la realización de la arquitectura
- Preocupación (concern): Una preocupación sobre una arquitectura es un requisito, un objetivo, una intención o una aspiración que un involucrado tiene para esa arquitectura

Stakeholders

Involucrados

- Individuo, equipo u organización que tiene un interés en la realización del sistema

Acquirers	Supervisan la obtención del sistema o producto
Assessors	Supervisan que el sistema cumpla con estándares y regulaciones legales
Communicators	Explican el sistema a otros involucrados a través de documentación y manuales
Developers	Construyen e implementan el sistema a partir de las especificaciones
Maintainers	Administran la evolución del sistema ya que está en producción
Production Engineers	Diseñan, implementan y administran los ambientes de hw y sw sobre los que se va a construir, probar y ejecutar el sistema
Suppliers	Construyen o consiguen el hw, sw o infraestructura sobre la que se va a ejecutar el sistema
Support Staff	Proveen soporte a los usuarios cuando el sistema está en producción
System Administrators	Ejecutan el sistema una vez que se ha implementado
Testers	Prueban el sistema para asegurar que es apropiado para su uso
Users	Definen la funcionalidad del sistema y finalmente hacen uso del mismo

Problemas para describir una arquitectura

Preguntas

- ¿Cuáles son los principales elementos funcionales de la arquitectura?
- ¿Cómo van a interactuar estos elementos entre ellos y con el mundo exterior?
- ¿Qué información se va a administrar, almacenar y presentar?
- ¿Qué elementos físicos de hardware y qué elementos de software se requieren para soportar los elementos funcionales y de información?
- ¿Qué características y capacidades operativas se van a proveer?
- ¿Qué ambientes de desarrollo, pruebas, soporte y entrenamiento se requieren?

Software Design Description

SDD

- Conjunto de artefactos que documenta una arquitectura de forma que los involucrados puedan entenderla y que demuestra que la arquitectura resuelve sus preocupaciones
- No es posible capturar la arquitectura en un solo sistema complejo que sea entendible y valorado por todos los involucrados
- La estrategia es usar un conjunto de vistas interrelacionadas, que colectivamente ilustran las características funcionales y propiedades cualitativas y que demuestran que la arquitectura cumple con sus objetivos

Arquitectural Views

Views

- Una vista es una representaciones de uno o más aspectos estructurales de una arquitectura que ilustra cómo es que la arquitectura resuelve una o más preocupaciones de uno o más de los involucrados
- Tip: incluye en una vista solo los detalles que van encaminados a cumplir con los objetivos de tu SDD, es decir, los detalles que ayudan a explicar la arquitectura a los involucrados o a demostrar que se están tomando en cuenta sus preocupaciones

Architectural Viewpoints

Viewpoints

- Un punto de vista es una colección de patrones, machotes y convenciones para construir un tipo de vista. Define a los involucrados cuyas preocupaciones se reflejan en la perspectiva y los lineamientos, principios y modelos para construir sus vistas
- Tip: cuando desarrolles una vista, debes ser muy claro en las preocupaciones que la vista aborda, qué tipo de elementos de arquitectura presenta, y a quién van dirigidos
- Asegúrate de que los involucrados también lo comprendan

Viewpoints

Catálogo

Context	Describe las relaciones, dependencias e interacciones entre el sistema y su ambiente (personas, sistemas y entidades con las que interactúa)
Functional	Describe los elementos funcionales, sus responsabilidades, interfaces e interacciones principales
Information	Describe la forma en que la arquitectura almacena, manipula, administra y distribuye información
Concurrency	Mapea elementos funcionales a unidades de concurrencia para identificar claramente las partes del sistema que podrían ejecutarse de forma concurrente y cómo se van a coordinar y controlar.
Development	Describe la arquitectura que soporta el proceso de desarrollo de software.
Deployment	Describe el ambiente en el que sistema se va a implementar, incluyendo las dependencias que tiene el sistema en su ambiente de ejecución.
Operational	Describe como se va a operar, administrar y dar soporte al sistema una vez que se esté ejecutando en su ambiente de producción.

Architectural Perspectives

Perspectives

- Una perspectiva de arquitectura es una colección de actividades, tácticas y lineamientos que se usan para asegurar que un sistema exhibe un conjunto de propiedades de calidad que requieren considerarse de forma transversal a través de varias vistas del sistema
- Tip: normalmente los problemas que abordan las perspectivas están relacionados a requerimientos no funcionales de la arquitectura
- Nunca vas a trabajar con las perspectivas por si solas, sino que las vas a usar para validar la calidad de tu arquitectura
- Esto se describe como aplicar la perspectiva a la vista

Perspectives

Catálogo

Accessibility	Capacidad de ser utilizado por personas con discapacidades
Availability and Resilience	Capacidad de ser completa o parcialmente operacional y manejar de forma efectiva fallos que pudieran afectar la disponibilidad
Development Resource	Capacidad de ser diseñado, construido, implementado y operado dentro de las restricciones conocidas de personal, presupuesto, tiempo y material.
Evolution	Capacidad de ser flexible ante los cambios inevitables
Internationalization	Capacidad de ser independiente de cualquier lenguaje, país o grupo cultural
Location	Capacidad de resolver problemas relacionados a la localización de sus elementos y la distancia entre ellos
Performance and Scalability	Capacidad del sistema de ejecutarse de forma predecible y manejar los incrementos de volumen de procesamiento
Regulation	Capacidad de apegarse a las leyes locales e internacionales, regulaciones legales, políticas de la compañía y otras regulaciones y estándares
Security	Habilidad de controlar, monitorear y auditar quién puede ejecutar qué acciones sobre qué recursos y detectar y recuperarse de fallos en los mecanismos de seguridad
Usability	Facilidad con que los usuarios que interactúan con el sistema pueden trabajar eficientemente

Referencias

- Rozanski, N. & Woods, E. (n.d.) Software Systems Architecture. Disponible en: <https://www.viewpoints-and-perspectives.info>