

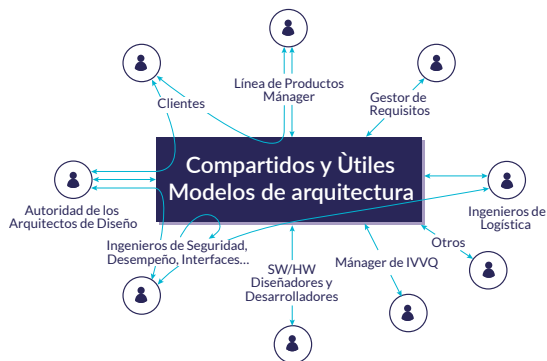


eclipse.org/capella[★]

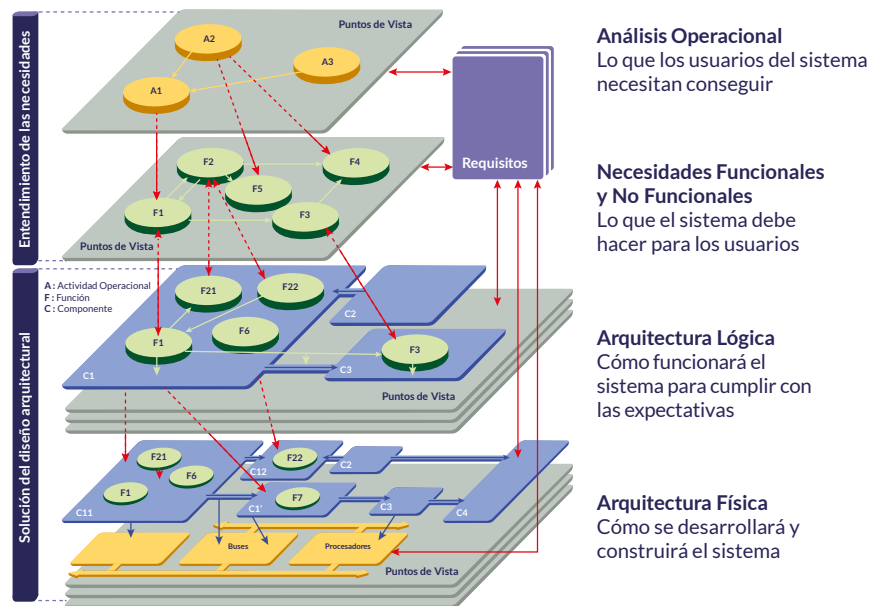
UNA METODOLOGÍA CON EL SOPORTE DE UNA HERRAMIENTA PARA

Definir, Analizar, Diseñar y Validar Arquitecturas de Sistema, Software y Hardware

Permitiendo la **Colaboración Eficiente** en los Procesos de Ingeniería



Validando/Justificando la solución con respecto a las **Necesidades Operacionales**
Facilitando el **Análisis de Impacto**



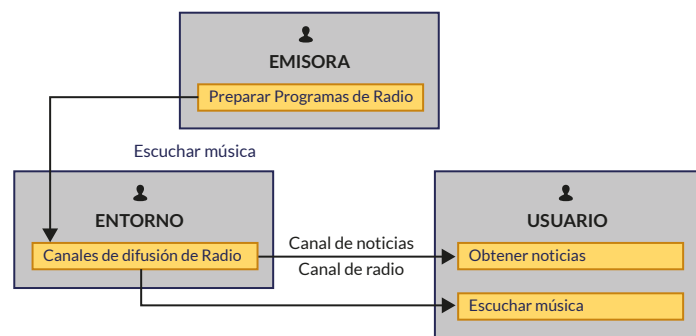
Facilitando el Análisis de Impacto Arriba-Abajo, Abajo-Arriba, iterativo, basado en herencia, mixto,...



Análisis de las Necesidades Operacionales del Cliente

Lo que los usuarios del sistema necesitan conseguir

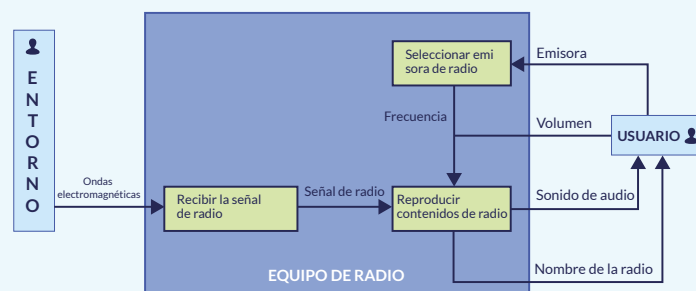
- ✓ Definir capacidades operacionales
- ✓ Análisis de las necesidades del Sistema/SW/HW



Análisis de las necesidades del Sistema/SW/HW

Lo que el sistema debe hacer para los usuarios

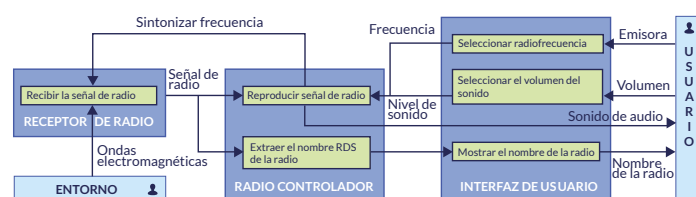
- ✓ Realizar un análisis de capacidades
- ✓ Realizar un análisis funcional y no funcional
- ✓ Formalizar y consolidar requisitos



Diseño de la Arquitectura Lógica

Cómo funcionará el sistema para cumplir con las expectativas

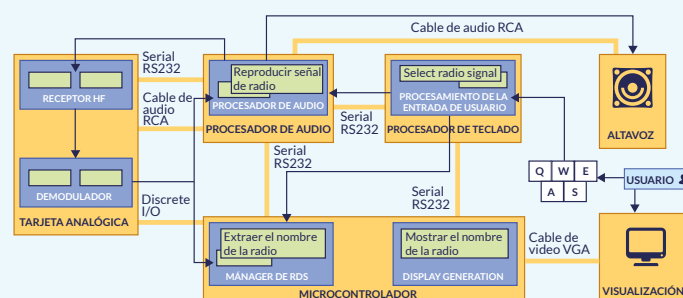
- ✓ Tomar decisiones arquitecturales y definir puntos de vista
- ✓ Construir la descomposición arquitectural en componentes
- ✓ Seleccionar la mejor arquitectura



Diseño de la arquitectura Física

Cómo se desarrollará y construirá el sistema

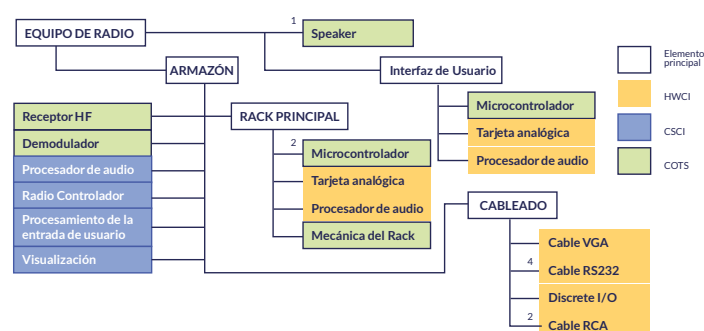
- ✓ Definir patrones arquitecturales
- ✓ Considerar la reutilización de elementos existentes
- ✓ Diseñar una arquitectura física de referencia
- ✓ Diseñar una arquitectura física de referencia



Contratos de desarrollo

Lo que se espera de cada diseñador / subcontratista

- ✓ Definir una estrategia IVVQ de componentes
- ✓ Definir y hacer cumplir un PBS y un contrato de integración de componentes



- Capacidades operacionales
- Actores, entidades operacionales
- Actividades de los actores
- Interacciones entre actividades y actores
- Información utilizada en actividades e interacciones
- Actividades de la cadena de procesos operacionales
- Escenarios para el comportamiento dinámico

- Actores y sistema, capacidades
- Funciones del sistema y de los actores
- Flujo de datos entre funciones
- Flujo de datos de las cadenas funcionales
- Información utilizada en funciones e intercambios, modelo de datos
- Escenarios para el comportamiento dinámico
- Modos y estados

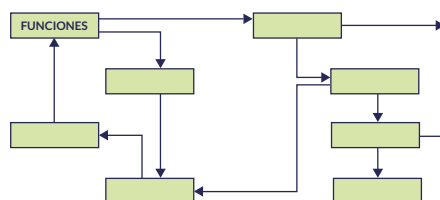
MISMOS CONCEPTOS QUE EN LOS OTROS PASOS Y ADEMÁS:

- Componentes
- Puertos de componentes e interfaces
- Intercambios entre componentes
- Asignación de funciones a componentes
- Justificación de la interfaz de componente mediante la asignación de los intercambios entre funciones

MISMOS CONCEPTOS QUE EN LOS OTROS PASOS Y ADEMÁS:

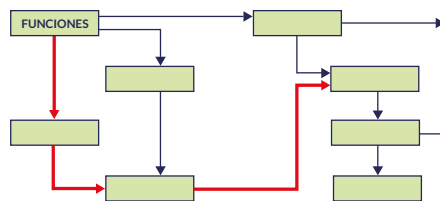
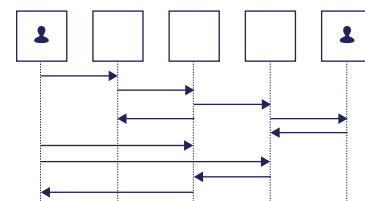
- Componentes de comportamiento que refinan los componentes lógicos e implementan el comportamiento funcional
- Componentes de implementación que proporcionan recursos a los componentes de comportamiento
- Enlaces físicos entre componentes de implementación e implementación

- Árbol de elementos de configuración
- Código de piezas, cantidades
- Contrato de desarrollo (comportamiento esperado, interfaces, escenarios, utilización de recursos, propiedades no funcionales...)

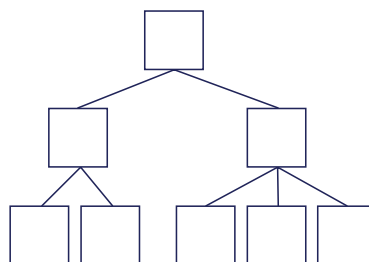


Flujo de datos:
funciones, relaciones e intercambios entre actividades operacionales

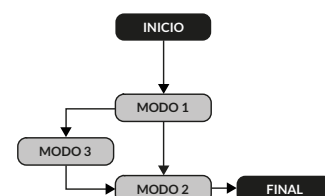
Escenarios:
Actores, sistema, interacciones e intercambios entre componentes



Cadenas funcionales, procesos operacionales
a través de funciones y actividades operacionales

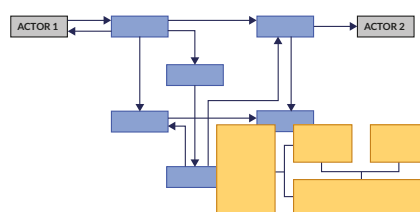
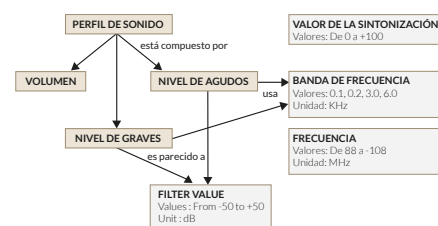


Descomposición de funciones y componentes



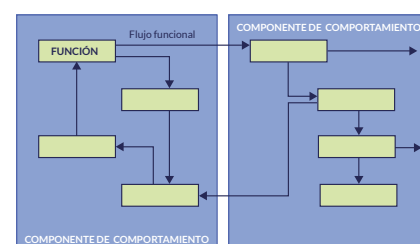
Modos y estados
de actores, sistema, componentes

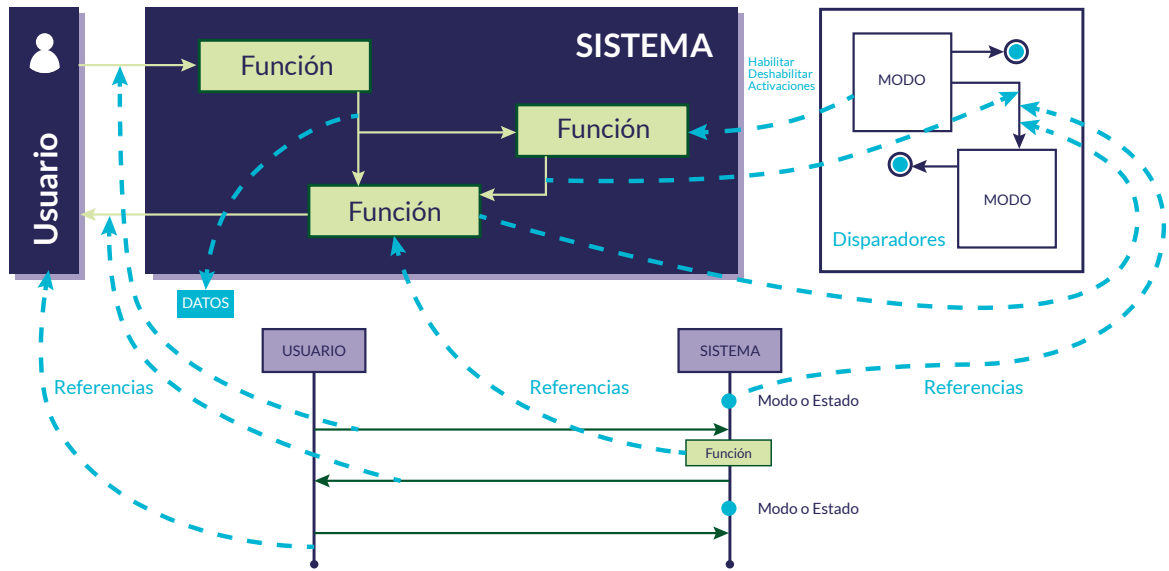
Modelo de datos:
Contenido del flujo de datos y de los escenarios, definición y justificación de las interfaces



Conexión de componentes: Todo tipo de componentes

Asignación:
de actividades operacionales a actores, de funciones a componentes, de componentes de comportamiento a componentes de implementación, de flujo de datos a interfaces, de elementos a elementos de configuración





Verificando y comprobando la solución con respecto a intereses no funcionales e industriales

Niveles del Método	Ejemplos de comportamiento	Ejemplos de seguridad
ANÁLISIS DE NECESIDADES OPERACIONALES	Máximo tiempo de reacción a la amenaza	Eventos no deseados
ANÁLISIS DE NECESIDADES FUNCIONALES Y NO FUNCIONALES	Cadena funcional (CF) para reaccionar ante una amenaza Latencia máxima permitida en una CF	Cadenas funcionales críticas asociadas a eventos
DISEÑO DE LA ARQUITECTURA LÓGICA	Complejidad del procesamiento y de los intercambios Asignación de cadenas funcionales	Caminos redundantes que aseguran cadenas funcionales
DISEÑO DE LA ARQUITECTURA FÍSICA	Utilización de recursos de una CF Cálculo de latencias	Modos de fallo comunes Propagación de fallos en CF
CONTRATOS DE DESARROLLO & IVVQ	Recursos asignados para cumplir con la latencia	Nivel de fiabilidad necesario

- ✓ Coste y calendario
- ✓ Interfaces
- ✓ Comportamiento

- ✓ Mantenibilidad
- ✓ Seguridad
- ✓ ... (etc)

- ✓ IVVQ
- ✓ Política de productos

