

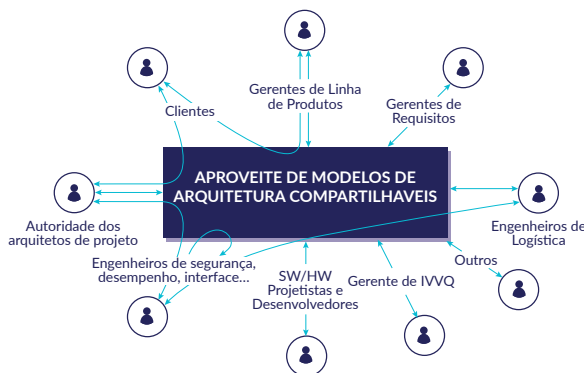


eclipse.org/capella

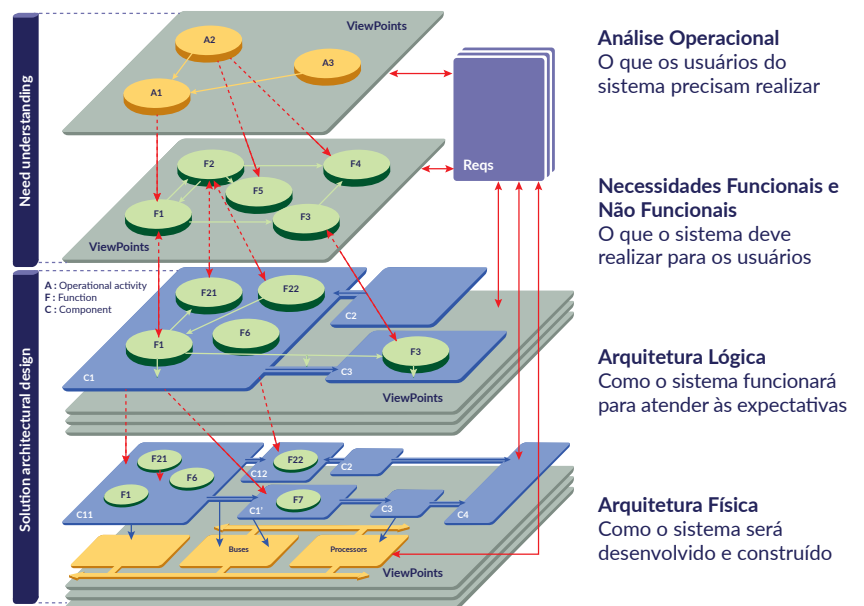
UM MÉTODO E UMA FERRAMENTA PARA

Definir, Analisar, Projetar e Validar arquiteturas de sistema de hardware e de software

Apoiando a
Colaboração Eficiente
em Engenharia



Validando/Justificando
soluções em relação às **Necessidades dos Stakeholder**
facilitando as **Análises de Impacto**



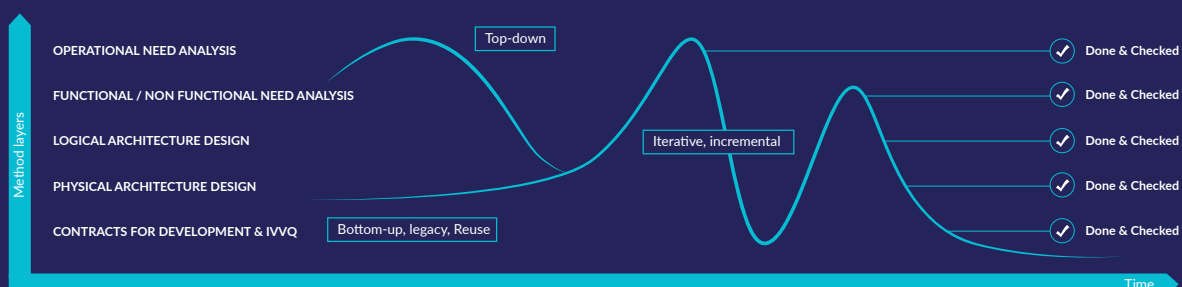
Análise Operacional
O que os usuários do sistema precisam realizar

Necessidades Funcionais e Não Funcionais
O que o sistema deve realizar para os usuários

Arquitetura Lógica
Como o sistema funcionará para atender às expectativas

Arquitetura Física
Como o sistema será desenvolvido e construído

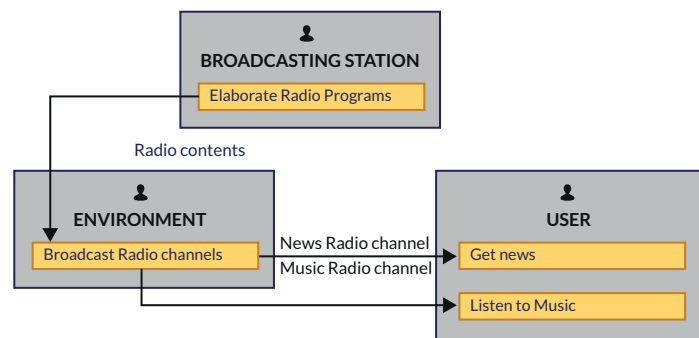
Compatível com a maioria dos processos
top-down, bottom-up, iterativo, legacy-based, misto...



Análise das Necessidades dos Stakeholders

O que os usuários do sistema precisam realizar

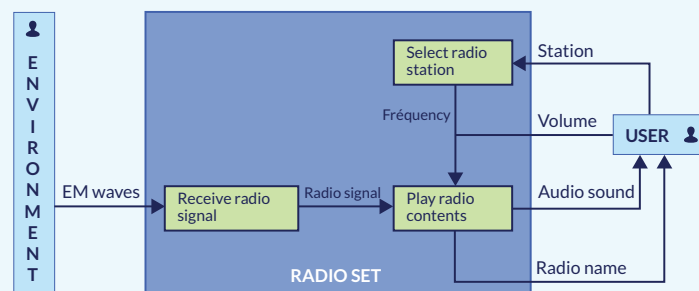
- ✓ Definir capacidades operacionais
- ✓ Realizar a análise de necessidades



Análise das Necessidades para o Sistema (HW/SW)

O que o sistema precisa realizar para os usuários

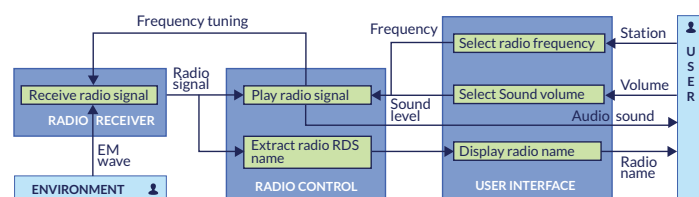
- ✓ Realizar a análise das capacidades
- ✓ Realizar análise funcional e não funcional
- ✓ Formalizar e consolidar os requisitos



Arquitetura Conceitual

Como o sistema funcionará para atender às expectativas

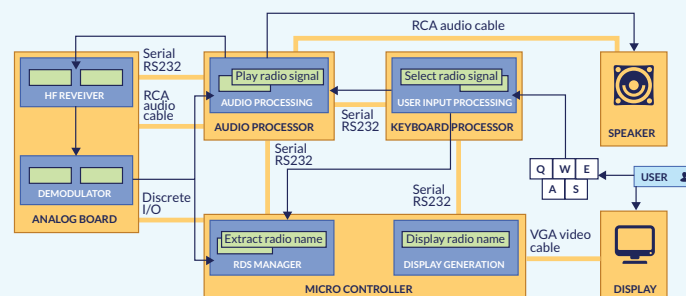
- ✓ Definir drivers de arquitetura e viewpoints
- ✓ Desdobrar arquitetura em componentes
- ✓ Selecionar a melhor arquitetura



Arquitetura Realizada / Arquitetura Concreta

Como o sistema será desenvolvido e construído

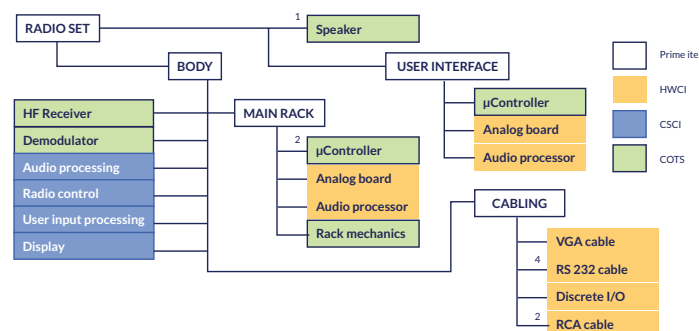
- ✓ Definir padrões de arquitetura
- ✓ Considerar a reutilização de elementos existentes
- ✓ Projetar uma arquitetura física de referência
- ✓ Verificar e Validar o sistema



Contratos de Desenvolvimento

O que se espera de cada projetista/subcontratado

- ✓ Definir uma estratégia de IVVQ (Integrar, Verificar, Validar e Qualificar)
- ✓ Estratégia de EAP e definição de cadeia produtiva



- Capacidades Operacionais
- Atores, entidades operacionais
- Atividades de atores
- Interações entre atividades e atores
- Informações usadas em atividades e interações
- Atividades de encadeamento de processos operacionais
- Cenários para dinâmica do comportamento

- Atores e sistema, capacidades
- Funções do sistema e dos atores
- Fluxo de dados de trocas entre funções
- Fluxo de dados de cadeias funcionais
- Informações usadas em funções e trocas, modelo de dados
- Cenários para dinâmica do comportamento
- Modos e Estados

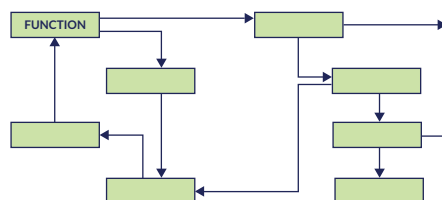
MESMOS CONCEITOS DAS OUTRAS ETAPAS, ALÉM DE:

- Componentes
- Portas de componentes e interfaces
- Relações entre componentes
- Alocação de funções em componentes
- Justificativa da interface de componente através da alocação das relações entre funções

MESMOS CONCEITOS DAS OUTRAS ETAPAS, ALÉM DE:

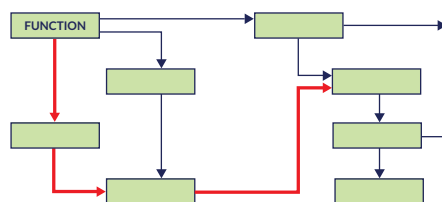
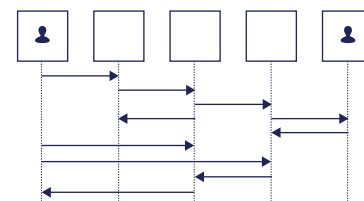
- Componentes de comportamento para refinar a implementação funcional nos componentes físicos
- Componentes para alocação de comportamentos
- Relações entre os componentes a serem desenvolvidos

- Árvore de configuração de itens
- Código das peças e quantidades
- Contrato de desenvolvimento (comportamento esperado, interfaces, cenários, consumo de recursos, propriedades...)

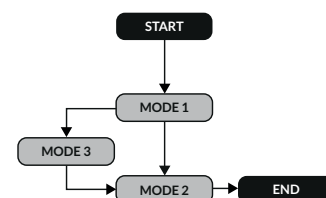
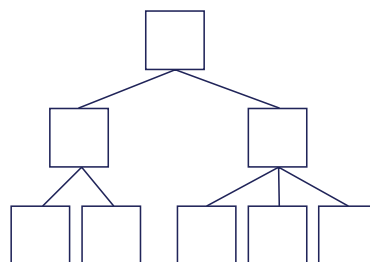


Fluxo de dados:
relações entre funções e atividades operacionais

Cenários:
atores, sistema, interações e relações entre componentes



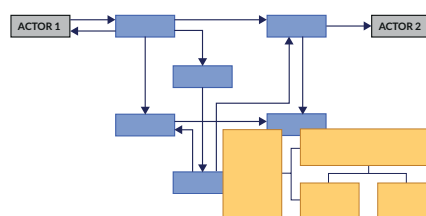
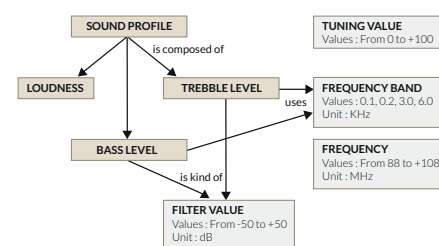
Cadeia funcional e processos por mapeando funções e atividades



Modos e Estados
de atores, sistema, componentes

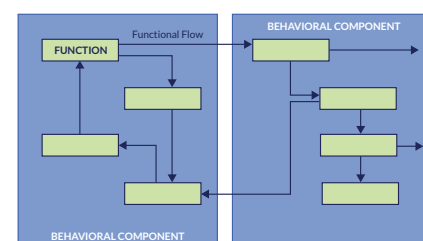
Diagrama de detalhamento
de funções e componentes

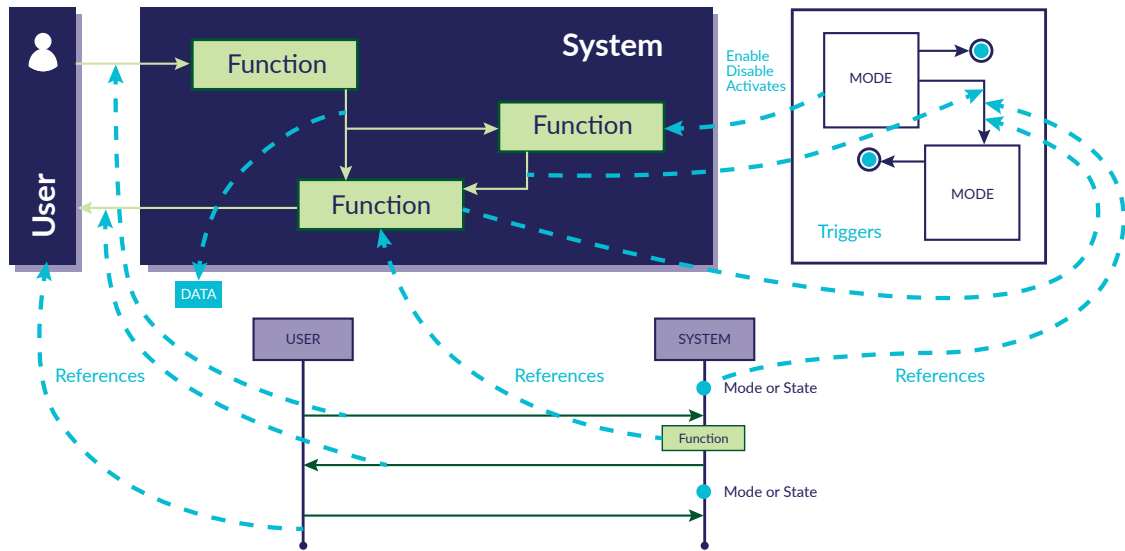
Modelo de dados:
conteúdo das relações, definição e justificativa das interfaces



Interligação de componentes:
todos os tipos de componentes

Alocação
de atividades para atores, de funções para componentes, de componentes comportamentais para implementação de componentes, de fluxos de dados para interfaces, de elementos para itens de configuração





Verificação e Validação da solução em relação aos requisitos e premissas para industrialização

Etapas do método	Exemplos de desempenho	Exemplos de segurança
ANÁLISE DE NECESSIDADES DOS STAKEHOLDERS	Tempo máximo de reação à ameaça	Eventos indesejados
ANÁLISE DE NECESSIDADES FUNCIONAIS E NÃO FUNCIONAIS	Cadeia funcional (CF) para reagir à ameaça. Latência máxima permitida na CF	Cadeias funcionais críticas associadas aos eventos
PROJETO DE ARQUITETURA CONCEITUAL	Complexidade de processamento e suas relações	Caminhos de redundância protegendo cadeias funcionais
PROJETO DE ARQUITETURA CONCRETA	Consumo de recursos na CF. Latência computacional resultante	Modos de falha comuns. Propagação de falhas em CF
CONTRATOS DE DESENVOLVIMENTO E IVVQ	Recursos alocados para satisfazer a latência	Nível de confiabilidade necessário

- ✓ Custos e Cronograma
- ✓ Interfaces
- ✓ Desempenho

- ✓ Manutenibilidade
- ✓ Segurança/Proteção
- ✓ ...

- ✓ IVVQ
- ✓ Política do produto

