

Engenharia de Sistemas Pontos-chaves em sua implementação eficaz

1. Ciclo de vida do Sistema

Definir o conjunto de resultados necessários e desejados por toda a vida do sistema pelos stakeholders deste mesmo sistema.

Distinguir entre os stakeholders primários (aqueles que nós servimos) e os stakeholders secundários (cujos interesses podem servir os interesses daqueles a quem servimos).

Distinguir entre os requisitos (devem ser atendidos), medida da eficácia (MOEs – características que agregam valor além dos requisitos) e metas explícitas no que diz respeito às medidas de eficácia.

Problemas com requisitos historicamente entram para a lista do porquê projetos dão errado. Concentre inicialmente no entendimento e definição do problema a ser resolvido.

Definir, inicialmente, a solução conceitualmente, fornecendo arquitetura. Depois detalhar a arquitetura selecionada, fornecendo um projeto detalhado ao nível da definição de solução física em que estamos trabalhando.

Captar os requisitos e os objetivos, e projetar a solução, para progressivamente elementos menores do sistemas, dividindo cada grande problema em um conjunto de problemas individualmente menores e bem definidos.

Verificar os requisitos, MOEs e objetivos do sistema de interesse, utilizando um Sistema de Análise de Requisitos. Verificar o projeto de arquitetura utilizando pelo menos uma revisão do projeto arquitetônico. Verificar seu projeto detalhado utilizando pelo menos uma revisão do projeto detalhado. Aplicar esta estratégia em elementos progressivamente menores da solução como o projeto evolui.

Usar “big bang” (cachoeira, grande projeto, seqüencial) quando tiver requisitos bons e estáveis, e baixo risco devido à complexidade e tecnologia.

Usar desenvolvimento incremental quando tiver requisitos bons e estáveis porém com um nível médio ou alto risco devido à complexidade e tecnologia.

Usar desenvolvimento evolutivo quando tiver requisitos ruins que não podem melhorar, ou alterar os requisitos.

Não confunda desenvolvimento evolutivo com um fracasso à tentativa de capturar o que é conhecível sobre requisitos (sempre o mais recente).

Usar uma fase baseada em uma fase de passagem “espiral” (risk driven) desenvolvimento de uma estratégia global.

Para modificações, fazer uma comparação antes/depois dos requisitos, e uma construção estado/construção de comparação padrão, como uma entrada para o projeto.

Todos os seguidores deste curso estão neste contexto de diferentes estilos de desenvolvimento.

2. O contexto da Engenharia de Sistemas (padrões, etc)

Acordo geral sobre a definição e aplicação de Engenharia de Sistemas que não foi alcançada.

Não importa!

Todos os padrões da Engenharia de Sistemas contemporânea compartilham temas em comum.

Padrões na Engenharia de Sistemas podem ser úteis mas nunca espere muito deles.

EIA/IS-632 (substituída), contém pérolas de sabedoria, mas é mal escrito. Este padrão pode ser a melhor escolha para uma aplicação contratual.

EIA 632 (atual) é mais uma orientação do que uma norma. A (geralmente) orientação de alta qualidade faz o EIA 632 mais adequado para aplicação voluntária.

IEEE 1220 possui uma gramática inglesa boa, mas é fraca em seus conceitos. A norma também direciona os métodos detalhados que estão longe da melhor prática.

ISO/IEC 15288 é um padrão do ciclo de vida do sistema que é conceitualmente fraco. Além disso, o padrão redefine a língua inglesa. Usar o ISO/IEC 15288 poderia estabelecer uma organização madura de volta.

IEA 731 é o melhor dos puros modelos de maturidade de capacidade da engenharia de sistemas. (CMM's), mas também considerado o CMMI.

A maioria dos US DoD cancelados ou substituídos, da engenharia MIL-STDs, ficam no grupo dos pobres e positivamente perigosos (MIL-STD-499B, MIL-STD-498 são as exceções). Cuidado!

3. Princípios, Conceitos e Processos.

Catorze princípios podem ser destilados para governar a engenharia de sistemas.

Pelo menos treze destes princípios são aplicáveis a engenharia de não-sistemas. Embora o equilíbrio de importância difere entre os sistemas e os não-sistemas.

Todos os catorze princípios se aplicam igualmente à engenharia de software.

Engenharia de sistemas e software estão se unindo inexoravelmente, uma tendência que se acelerou com a liberação do Unified Modelling Language (UML) Versão 2, e o lançamento iminente de SysML.

O processo da engenharia de sistemas é apoiada por vários conceitos, dois conceitos dominantes relacionados, sendo sistema de contexto e limite do sistema.

O processo da engenharia de sistemas pode ser descrito em termos de seis processos centrais:

- Captura dos requisitos e validação (análise de requisitos)
- Desenho da solução física
- Desenho da solução lógica
- Avaliação da eficácia e decisão
- Especificação dos elementos do sistema
- Gestão da engenharia, que também pode ser adicionado;
- Integração da especialidade da engenharia
- Verificação
- Validação

para cada resultado procurado. Esses resultados podem, ou não, envolver seus próprios processos.

Rastreabilidade dos requisitos existem em duas formas – uma forma em análise de requisitos e a outra em design.

Implementar a rastreabilidade de requisitos adicionará mais valor do que o custo.

Sempre implemente a rastreabilidade de verificação (teste)

4. Análise de Requisitos (Captura e Validação dos Requisitos)

O processo da análise de requisitos procura capturar e validar os requisitos (características que cada solução deve possuir), medindo sua eficácia (os resultados mensuráveis que individualmente representam graus de bondade, além da simples satisfação dos requisitos), e objetivos (metas específicas com relação ao indivíduo MOEs).

O processo da análise de requisitos também procura desenvolver, para cada requisito ou objetivo, o requisito correspondente como prova de que o requisitos foi satisfeito (requisito de verificação).

O processo de análise de requisitos também procura desenvolver uma descrição de sistema centrado de quem os usuários de um sistema serão, para que eles irão utilizar o sistema, como eles o usarão, e as condições em que eles o usarão (Descrição do Conceito Operacional – OCD).

Uma boa compreensão dos tipos de requisitos (estado/modo, funcional, desempenho, interface externa, recursos, física, outra qualidade e design), permite uma análise de requisitos eficaz, constituir a base da estrutura de especificação do som, e influencia muitos passos em design.

Dez medidas individuais que fornecem os critérios pelos quais a qualidade geral de um conjunto de requisitos podem ser avaliados:

- viabilidade
- completude
- exatidão
- não-ambigüidade
- clareza
- verificabilidade
- orientação funcional
- rastreabilidade
- singularidade
- liberdade do processo

Métricas de requisitos de qualidade pode ser um instrumento fundamental para evitar perdas devido a problemas de requisitos.

As normas técnicas necessárias para um conjunto de requisitos varia enormemente com a finalidade e as circunstâncias de sua utilização. Um intervalo de 0.00 à 0.99 para a métrica de qualidade de requisitos preferidas

Uma abordagem analítica é quase sempre necessária para transformar um conjunto de requisitos inadequados em adequados (baixo risco).

Uma abordagem analítica, para ser eficaz, normalmente precisará de pelo menos análises de contexto, análise de requisitos de concepção, estados & modos de análise, análise funcional, análise ERA, análise da análise (quando os requisitos são expressados em linguagem natural), análise fora-de-gama, pesquisa de outras restrições e uma atividade final de limpeza.

Em casos de uso das UML serem úteis (sem trocadilhos), mas raramente são formas suficientes de uma análise funcional.

Utilização de um processo estruturado para captar e gerenciar a resolução de cada problema com requisitos.

Na realização da análise de requisitos, não faça nenhum trabalho que não agregue valor em termos de resultados!

5. Concepção do processo de solução física

O primeiro ato do projeto é identificar os conceitos no mundo físico, que poderá fornecer uma solução.

O segundo ato do projeto é eliminar qualquer conceito que, após análise, revela-se inviável.

O terceiro ato do projeto é eliminar qualquer conceito que, embora viável, não pode ser o mais eficaz das alternativas.

O quarto ato do projeto é re-expressar cada conceito sobrevivente em termos de candidato e um conjunto de produto do sistema e elementos do processo.

O quinto ato do projeto (físico) é a maioria do trabalho! Este trabalho geralmente envolve o mapeamento do projeto lógico (produzido pelo processo de “solução do projeto lógico”) à sua implementação física, seu desempenho do projeto para satisfazer os requisitos não representados no projeto lógico, e definição das características das interfaces internas (concepção de interface interna).

Há muitos passos a serem dados para a realização do projeto físico.

Object Oriented Design e Design of Physical Solution Process, em Engenharia de Sistemas, diferem apenas no nome – eles são essencialmente idênticos.

Na concepção da solução física, não faça nada que não agregue valor em termos de resultados!

6. O processo do projeto da solução lógica

Cada projeto lógico corresponde à um conceito físico.

O projeto lógico é uma abstração do que é planejado ou possível no mundo físico.

O projeto lógico NÃO é independente da implementação física, exceto onde as tecnologias de execução física não podem fazer nada por isso! O que é, nada! O mais próximo vem com a independência, em seu projeto lógico de uma entidade a ser implementada em software.

A forma mais comum do projeto lógico de uma entidade física, é um projeto funcional, expressado em termos de um conjunto de nível de solução de funções alocáveis, seu desempenho associado, suas entradas e saídas, suas condições de início

e de suas condições de realização, o tempo sequencial e relações concorrentes (fluxo controlado) e os fluxos de pertences entre eles (fluxo único).

Cada projeto lógico está errado – É somente uma aproximação à realidade física.

Projeto funcional com base na decomposição das funções de nível dos requisitos para as funções dos níveis de solução que estão baseadas em um pressuposto inválido – refinar o projeto funcional para refletir a realidade física através da análise do desempenho do segmento.

Realizar modos de falha, efeitos e análise de criticidade (FMECA) após o mapeamento do modelo lógico para a sua execução física, e iterativamente redesenhar as soluções funcionais (e físicas).

Ao desempenhar o projeto lógico, não faça nada que não agregue valor em termos de resultados.

7. Avaliação da eficácia e processo da decisão

Usar o modelo de julgamento ou eficácia formal (possibilidade de análise de requisitos)

Construir um modelo de eficácia geral do sistema (possibilidade de análise de requisitos)

Para aplicações importantes, use técnicas de atributos de múltiplas utilidades:

- o peso de cada stakeholder assim como sua influência na tomada de decisão.
- para o desenvolvimento do produto, tratar diferentes mercados como stakeholders diferentes.
- pesar cada uso diferente do sistema (ou apoiados pela missão)
- agregar contribuições individuais para a eficácia (usando técnicas numéricas para acomodar a incerteza, exemplo: Monte Carlo ou Latin Hypercube Simulation)
- em um modelo de eficácia do sistema, NUNCA pese os critérios abstratos.
- em um modelo de eficácia do sistema, NUNCA pese o risco. Reflita a incerteza através da utilização das funções de densidade de probabilidade (PDFs) para caracterizar as alternativas de solução.

Caracterizar cada alternativa de projeto com relação a cada MOE. Onde há a incerteza no resultado para um MOE, expressando o valor do resultado como uma função de densidade de probabilidade.

SOMENTE dividir o custo em eficácia ou eficácia em custo se tiver validado a linearidade do valor atribuído ao dinheiro. Cada linearidade é uma exceção. Caso contrário, tratar o custo como qualquer outro MOE.

Usar decisões e eventos de árvores para auxiliar a tomada de decisão na presença de incertezas nos eventos de base.

Para aplicações importantes, usar o Analytic Hierarchy Process (AHP) para desenvolvimento de pesos, e Multiple Attribute Utility Technique (MAUT) para um modelo geral de eficácia.

Nas tomadas de decisões da engenharia, não faça nenhum trabalho que não adicione valores em termos de resultados!

8. Especificação do Processo de Elementos do Sistema

Especificar os requisitos de cada elemento da solução (produto ou elemento do processo) em um nível de detalhes que correspondem à baixo risco de requisitos relacionados.

Estabelecer e usar uma empresa padrão para cada tipo de especificação produzida por sua empresa.

Adote um bom padrão!

Alguns tipos básicos da especificação de requisitos são:

- especificação de requisitos de sistema
- especificação de requisitos de software
- especificação de requisitos de interface
- especificação material
- especificação do processo
- especificação do serviço (declaração de trabalho, acordo de nível de serviço, etc)
- lista de contrato de dados de requisitos
- especificação de dados de requisitos
- definição de dados do elemento
- especificação da verificação de requisitos

Usar tipos de requisitos como uma estratégia básica para estrutura de especificação de requisitos.

Ter uma um guia de estilos organizacionais para especificação de requisitos.

Ao especificar os elementos do sistema, não faça nenhum trabalho que não adicione valores em termos de resultados!

9. Integração da Especialidade da Engenharia

Integração efetiva das disciplinas não-tecnológicas (exemplo: confiabilidade, produção, segurança, uso fácil, etc) com os requisitos e o projeto da solução do

sistema simplesmente “não somente” acontecem, ambos exigem esforço e concentração para acontecer.

Um programa eficaz de especialidade, geralmente inclui um número de tarefas, cada individualmente pequenas, embutidas dentro dos processos gerais da engenharia do sistema.

As tarefas de especialidade podem variar enormemente entre suas diferentes especialidades.

Usar o Integrated Product Teams e outras estruturas de equipe de base para fomentar a integração de especialidade.

Ao prosseguir a integração da especialidade da engenharia, não faça nenhum trabalho que não adicione valores de resultados!

10. A Transição para o Produto

As instruções de integração (construídas) são os processos de solução do produto do projeto físico e projeto lógico. Essas instruções são executadas na construção de cada elemento do sistema, e construção do sistema completo.

Ao ter um elemento de sistema construído, é necessário a resposta de duas questões fundamentais:

- o elemento do sistema corresponde seus requisitos?
- se o elemento de sistema corresponder seus requisitos, o elemento de sistema atual corresponde à descrição do projeto para aquele elemento?

Perform Requirements Satisfaction Audits (caso contrário, também conhecido como Functional Configuration Audits) e Build State/Build Standard Correspondence Audits (também conhecido como Physical Configuration Audits) em cada elemento do sistema até o nível de interesse do sistema.

Não faça nada que não agregue valores em termos de resultados!

11. Verificação & Validação

Verificação de conduta: o produto do trabalho corresponde aos requisitos para aquele mesmo produto do trabalho?

Considerar a verificação de conduta: o produto do trabalho corresponde a tudo que é necessário?

Use a análise do projeto como uma estratégia de verificação principal.

Considerar o IEEE 1012 como uma fonte de conceitos e métodos para o IV&V, mas não dê demasiada importância às tarefas individuais.

12. Gerenciamento da Engenharia

Todos os princípios de gestão, em geral, são aplicáveis à gestão de engenharia

Todos os princípios de gestão do projeto são aplicáveis à gestão de um projeto de engenharia.

O plano do gerenciamento da engenharia de sistemas é (ou deve ser) somente um plano, como qualquer outro plano, mas tendo como seu escopo o desempenho da engenharia.

Um projeto que é organizado utilizando o Integrated Product Terms (IPT) precisará de planos de engenharia múltiplos.

O gerenciamento da engenharia requer atenção especial na estrutura (PBS/WBS) do projeto (trabalho), gerenciamento da configuração, gerenciamento de interface, medição de desempenho e gestão de risco.

PBS deve ser orientado para o produto, descrevendo como os resultados do projeto devem ser atingidos.

PBS deve estar sempre preparado para um projeto utilizando os primeiros princípios, nunca utilizando um modelo.

MIL-STD-881 (cancelado) no Work Breakdown Structure pode ser também um dos padrões mais prejudiciais já emitidos.

Ao gerenciar as configurações, identifique as configurações utilizando linha de base, mudança de controle, manter os registros, configuração de áudio. Em relação ao processo, distinguir entre as mudanças que afetam uns e outros não.

Ao gerenciar as interfaces, utilize as especificações dos requisitos de interface, definir os processos para alcançar um acordo sobre as características da interface.

Aplique, primeiramente, as melhores práticas; se o risco não for baixo, em seguida aplicar à gestão de risco focalizada.

Ao medir o desempenho da engenharia, definir o produto significativo e métricas do processo, planejar um perfil de realização ao longo do tempo, medir contra o plano, usar variâncias e tendências para o diagnóstico precoce e o desencadeamento de ações corretivas.

Entender a natureza do risco. O risco tem os ingredientes dos critérios do sucesso (que resultados são importantes), ameaças para a consecução desses resultados (o que pode dar errado?), as vulnerabilidades (o quanto vulnerável estamos se algo der errado?). Risco é “uma perda esperada” – a perda cronometra a probabilidade da perda que está ocorrendo.

Se o risco não é baixo, gerencie o risco de uma forma orientada, utilizando um processo que inclui a avaliação inicial dos riscos, identificação e avaliação dos critérios de sucesso, a análise inicial da quantidade de risco em relação a cada critério de sucesso, replanejamento iterativo para reduzir o risco ou para alterar o equilíbrio entre risco e oportunidade, e o acompanhamento contínuo do montante e fontes de risco residual.

Na tentativa de melhorar o desempenho da engenharia, Marque como referência as práticas atuais. Dê prioridade as áreas para melhoria. Desenvolva caminhos melhores para resolver tudo que for preciso; nunca imponha os processos externalmente desenvolvidos. Planeje para evoluir, não revolucionar.

