

Uma Ferramenta Web Colaborativa para Apoiar a Engenharia de Requisitos em Software Livre

Marco Aurélio Graciotto Silva

Orientadora: Profa. Dra. Renata Pontin de Mattos Fortes

Departamento de Computação
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo

Defesa de Mestrado

Roteiro

- 1 **Introdução**
 - Motivação
 - Objetivos
- 2 **Engenharia de Requisitos**
 - Definições
 - Problemas
 - Soluções
- 3 **Hiperdocumentos de Requisitos**
 - Definições
 - Sistemas Hipermídia para Engenharia de Requisitos
- 4 **Wikis**
 - Definição
 - Implementações
- 5 **Wiki/RE**
 - Definição
 - Desenvolvimento
- 6 **Conclusões**

Introdução

O que se deseja de um software, desde sua concepção até a utilização?

- Custos baixos (produção e venda).
- Prazos razoáveis e respeitados (tempo necessário para entrega e longevidade).
- Produtos de qualidade (fácil de manter, com poucos defeitos, confiável, etc).
- Interessados satisfeitos.

Como satisfazer os interessados?

- Atendendo seus requisitos.

Introdução

Engenharia de Requisitos

A Engenharia de Requisitos é responsável pela criação e aperfeiçoamento de técnicas que permitam o correto desenvolvimento dos requisitos de um software.

Engenharia de requisitos é diferente do restante da engenharia de software:

- Envolvimento direto com os clientes.
- Início do processo.
- Multifacetada.

E no desenvolvimento de software livre?

- Clientes e desenvolvedores (!) desconhecidos.
- Início “nebuloso”.
- Desenvolvimento distribuído.
- Processo pouco estudado.

Introdução

O que é engenharia de requisitos em software livre

Seria estranho, para não dizer ilógico, um software de qualidade sem requisitos bem definidos.

Facilitadores:

- Desenvolvedor é um dos principais usuários.
- Trabalho facilitado por outros projetos:
 - Especificações.
 - Cópias de outros softwares.
 - *Fork* de outro software livre.

Complicadores:

- Comunicação limitada.
- Prazos? Que prazos?
- Ausência de documentação.

Introdução

Como melhorar a engenharia de requisitos em software livre

Comunicação e prazos, principalmente em empresas, são problemas menores (ao menos passíveis de solução). Já a documentação. . .

- Arquivos textos, páginas Web, listas de email.
- Sem uma organização bem definida.
- Atualmente utilizam-se ferramentas de uso geral.
- Manutenção da documentação é cara.
- Existe uma carência por ferramentas que apoiem a engenharia de requisitos em software livre.

É preciso:

- Instrumento mais adequado para documentar requisitos.
- Permitir o uso de técnicas mais sofisticadas.
- Preservar a liberdade do desenvolvedor.

Objetivos

- Propor uma solução para a documentação dos requisitos em projetos de software livre:
 - Sistemas hipermídias.
 - Wikis.
- Desenvolver uma ferramenta para a criação de documentos de requisitos de qualidade:
 - Colaborativa.
 - Web.
 - Livre.
 - Flexível.

Engenharia de Requisitos

Requisito:

- É uma propriedade que o software possui para resolver um problema do mundo real.
- Traduz um desejo ou expectativa do usuário quanto ao software.

Classificados em:

- Funcionais: ação do software quando confrontado ao meio.
- Não-funcionais: restrições ou condições.

Processo de Engenharia de Requisitos

A geração dos requisitos requer o uso de técnicas, métodos e ferramentas, coordenados por um processo de engenharia de requisitos.

- (...)
- Elicitação.
- Análise.
- Especificação.
- Validação.

Interessados

Interessados (ou *stakeholders*):

- Indivíduos ou entidades envolvidos no processo.
- Quem tem influência na definição dos requisitos.

Por exemplo:

- Usuários finais.
- Desenvolvedores.
- Cliente.

Problemas na interface

Usuários querem linguagem acessível, alto nível de abstração, descrições práticas, assumem conceitos do domínio como conhecidos, mudam de idéia constantemente, etc.

Desenvolvedores preferem linguagem precisas, vários níveis de abstração, detalhamento, explicações precisas e completas dos usuários.

Resumindo, surgem problemas de:

- Ambigüidade.
- Domínio de aplicação.
- Inconsistências.
- Volatilidade.
- Comunicação.

Outros problemas

Também existem problemas do lado mais “engenharia”:

- Rastreabilidade.
- Gerenciamento de requisitos.
- Elaboração e manutenção da documentação de requisitos.

As soluções

Existem várias técnicas:

- Facilitadores de comunicação (LAL).
- Casos de uso.
- Casos de mau uso.
- Pontos de vista.
- Técnicas de inspeção.
- ...

A questão é: como unir os resultados das diversas técnicas?

Documentação de requisitos

Os resultados são disponibilizados por documentos:

- Conceitos operacionais.
- Especificação de Requisitos.

Existem normas e recomendações para tais documentos. Ditam características desejadas e estrutura do documento:

- MIL-STD 498 (DI-IPSC-41433) (e J-STD-016 F.2.4)
- IEEE 830-1998
- IEEE P2063 (C/S2ESC) (Standard for System and Software Life Cycle Processes-Requirements Engineering)

Características esperadas da especificação de requisitos:

- Corretitude, ausência de ambigüidades, completitude, consistência.
- Classificação, verificabilidade, modificabilidade, rastreabilidade.

Como criar documentos de qualidade

As técnicas garantem as características em seus artefatos, mas e quando eles são unidos aos artefatos de outras técnicas?

Hiperdocumentos apresentam-se como um instrumento adequado para a documentação de requisitos:

- Armazenam dados heterogêneos.
- Ligações entre os artefatos = rastreabilidade.
- Facilidade de navegação.

Definições

Hiperdocumentos armazenam dados em estruturas denominadas nós e os relacionamentos entre os dados são armazenados na forma de ligações (links) ancorados nos nós.

- Artefatos de requisitos = dados armazenados em nós.
- Rastreabilidade = ligações entre os nós.

Sistemas hipermídia são caracterizados por:

- Hiperdocumentos com nós, ligações, âncoras.
- Nós simples e compostos.
- Distinção entre estrutura e conteúdo.
- Armazenamento de conteúdo em diferentes formatos.
- Diferentes tipos de ligações.
- Mecanismos de apresentação e controle de acesso ao hiperdocumento.

Sistemas Hipermídia para Engenharia de Requisitos

Aspectos explorados

Principais aspectos explorados:

- Armazenamento de diferentes tipos de dados.
- Rastreabilidade.
 - Geração automática ou semi-automática de ligações.
 - Verificação de atributos do documento de requisitos.
- Modelos de documento.

Sistemas Hiperímia para Engenharia de Requisitos

Aspectos a serem explorados

Aspectos importantes a serem explorados:

- Web.
- Desenvolvimento colaborativo.
- Facilitação da criação manual de ligações.
- Geração automática ou semi-automática em documentos hiperímia.
- Ferramenta livre.

Wikis

Definições

Wikis são sistemas hipermídia apropriados para a rápida criação de hiperdocumentos.

- Fáceis de usar.
- Ferramenta colaborativa.
- Criação da ligação (relacionamento) antecede a criação do nó (artefato).
- A “organização” do documento emerge naturalmente.

Lógica:

- 1 Mecanismo simples encoraja as pessoas a testarem.
- 2 Pessoas sentem-se confortáveis e criam novos nós.
- 3 Outras pessoas alteram os nós e acrescentam mais informações.
- 4 O ciclo se repete até que as pessoas fiquem satisfeitos com o conteúdo.

Por que funcionam

As wikis funcionam porque existe confiança mútua entre os colaboradores e nos dados.

Um conjunto de princípios, aos quais as wikis devem seguir, é estabelecido:

- Aberta: qualquer um participa.
- Incremental: o documento cresce ordenadamente (graças a precedência de ligações aos nós).
- Orgânica: estrutura aberta para alteração.
- Mundana/Banal: número reduzido de convenções para as atividades mais úteis/freqüentes.
- Universal: mecanismo de edição e organização é o mesmo.
- Evidente: conteúdo apresentado sugere a formatação dos dados de entrada.

Princípios

- Unificada: um único espaço de nomes para os nós.
- Precisa: nomes dos nós é definido precisamente.
- Tolerante: comportamento interpretável ao invés de mensagens de erro.
- Observável: toda atividade é observável por qualquer pessoa.
- Convergente: duplicação de conteúdos é evitada.

Funcionamento na prática

Na prática, funciona da seguinte forma:

- Toda página é identificada por um nome (substantivos).
- A página “inicial” é uma convenção.
- Ao acessar uma página que não existe, é oferecida a oportunidade de editá-la.
- A edição busca ser o mais simples possível.
- A criação de referências a outras páginas é feita através de uma convenção textual.

Funcionamento para engenharia de requisitos

Para a documentação de requisitos, funcionaria da seguinte forma:

- A página inicial é criada com a descrição do projeto e o problema a ser resolvido.
- O projeto é discutido e novos termos são elaborados (formam o universo de discurso).
- O problema é estudado e derivam-se requisitos (cenários, por exemplo).
- Derivam-se casos de teste para o requisito (Fitness).
- Um cenário é refinado, um novo cenário é incluído.
- Altera-se o problema (uma nova lei é criada). Observam-se nas páginas relacionadas quais os artefatos afetados e liga-se o texto da lei ao artefato em questão.
- ...

Funcionamento para engenharia de requisitos

A rastreabilidade, geralmente difícil de criar e manter, é obtida “de graça”.

- Todo artefato possui uma origem.
- Chance de duplicação de documentação diminui.
- Existe liberdade para ligar artefatos que, normalmente, não teriam relação.
- Se necessário, você pode burlar o mecanismo (exceção, não regra!)

Implementações estudadas

Existiam duas alternativas quanto à criação da Wiki/RE:

- Começar do zero.
- Começar a partir de outro projeto.

Foram realizados dois estudos:

- Avaliação de wikis Java.
- Manutenção da CoTeia.

Implementações avaliadas

Avaliaram-se as seguintes implementações de wikis:

- JSPWiki.
- Snipsnap.
- VeryQuickWiki.

Critérios:

- Software livre (preferencialmente compatível com a GPL).
- Qualidade do código: LOC, documentação da API, suite de teste, clareza do código.
- Uso: Dependências, instalação, popularidade.
- Processo de desenvolvimento: quantos lançamentos e com que frequência, uso de controle de versão e de alteração.
- Características wiki: atendimento aos princípios, expansibilidade, controle de versão.

Resultados da avaliação

- JSPWiki é a melhor.
- Nenhuma atende todos os requisitos.
- O modo como implementam a expansibilidade não é evidente/trivial.

Manutenção da CoTeia

Atividades executadas:

- Disponibilização com software livre (GPL)
- Evolução da ferramenta (segurança).
- Manutenção da compatibilidade com CoTeias antigas.

Resultados:

- Bom conhecimento sobre wikis.
- Documentação de projeto da Wiki/RE mantida na CoTeia.
- Descoberta de deficiências da CoTeia (e outras wikis) para a documentação de requisitos.

Análise dos estudos

Estratégia atual

Para trabalhar com artefatos de tipos diferentes, é necessário uma estratégia diferente.

Estratégia corrente: descrever artefatos complexos utilizando convenções textuais complexas

- Difícil, não intuitivo.
- Viola os princípios.
- Dificuldade em rastrear esse “sub-artefato”.

Análise dos estudos

Estratégia nova

Nova estratégia: se o elemento é complexo, melhor utilizar uma linguagem diferente. E cada recurso diferente que utilize uma linguagem diferente.

- As referências podem especificar, além do nome, o tipo do artefato.
- Artefatos diferentes = páginas e convenções diferentes.
- Eventualmente, pode-se fazer a transclusão (e conseguir, na apresentação, o mesmo efeito da convenção anterior).

Proposta

A nova estratégia não é compatível com as wikis atuais.

Solução:

- Construir uma nova wiki, utilizando o conhecimento adquirido nos estudos.

Wiki/RE

Definições

A Wiki/RE é uma wiki para engenharia de requisitos.

- Flexível.
- Permite o uso de artefatos diferentes (sem ferir os princípios).
- Facilita o desenvolvimento de tipos de artefatos diferentes.
- Busca manter a rastreabilidade além das outras wikis.

Desenvolvimento

- Software livre.
- Para software livre.
- Utiliza componentes livres.
- Todas as ferramentas de desenvolvimento utilizadas são livres.

Processo

- Baseado no Unified Process (conceitos).
- Inspirado em algumas idéias do OOHDM.
- Resultados documentados (na medida do possível).

Requisitos

Interessados:

- Marco, Renata, Kiko.
- Visões diferentes, algumas conflitantes.

Metas (documentação de requisitos, autoria colaborativa, extensibilidade, etc).

Casos de uso (processo):

- Aplicação de técnicas.
- Consulta à wiki durante a aplicação de técnicas.
- Documentação dos resultados na wiki.

Casos de uso

Casos de uso (wiki):

- Criação de wikipage.
- Alteração de wikipage.
- Visualização de wikipage.
- Visualização de versão antiga de wikipage.
- Comparação de wikipages.
- Remoção de wikipage.
- Alteração de identificador (nome) de wikipage.
- Refatoração de wikipage.

Alteração de nome e refatoração são importantes para manter a rastreabilidade.

Casos de mau uso

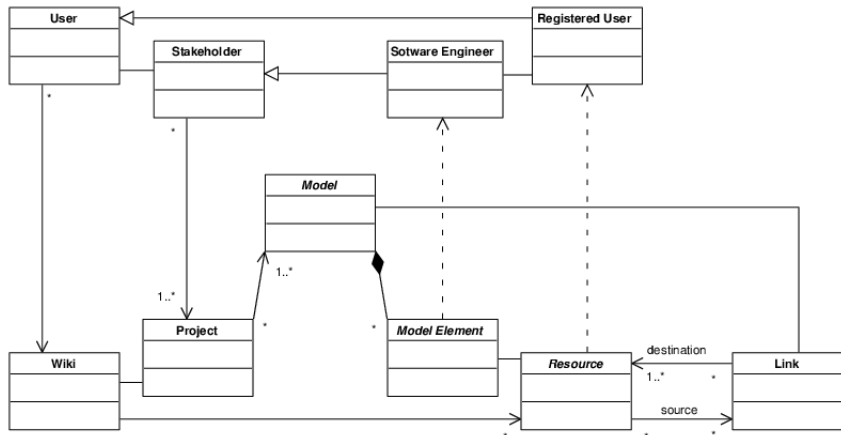
Previsão para tratamento de anti-requisitos:

- Vandalismo.
- Adulteração.
- Negação de serviço.

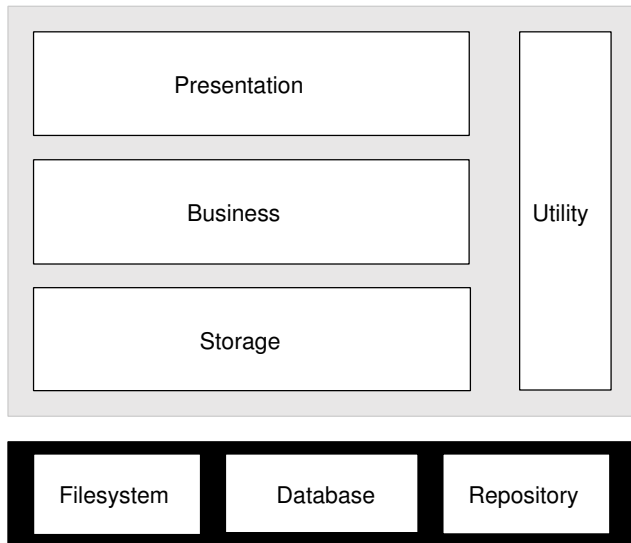
Soluções (previstas ou implementadas):

- Implementar a aplicação de modo a preservar escalabilidade.
- Autenticação (utilizando o SAFE).
- Utilizar servidores que suportem conexões seguras.

Projeto conceitual



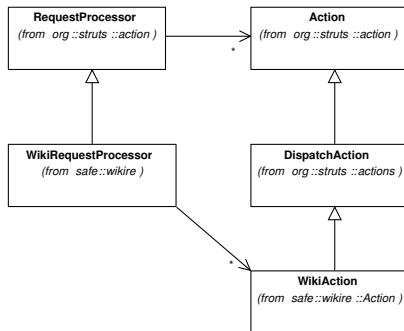
Arquitetura



Persistência

- Hibernate: objetos de negócio
- XStream: documentos XML a serem transformados.
- Subversion: controle de versão (dos documentos XML).

Apresentação e Negócio



- Struts: controle.
- JSP, JSTL: visões.

Transações

Todas as alterações são realizadas em transações.

- Encapsulam outras transações (Hibernate e Repository).
- Podem durar mais de uma requisição HTTP.
 - Ações básicas (Resource) são curtas (uma requisição, uma transação).
 - Transações longas são voltadas para ações complexas.

Roteiro para criação de um novo recurso

O roteiro básico para criação de um novo recurso é:

- 1 Criar a classe do recurso.
- 2 Criar o mapeamento (Hibernate).
- 3 Criar uma XSLT apropriada para o XML resultante do Xstream.
- 4 Criar uma ação a partir do ResourceAction.

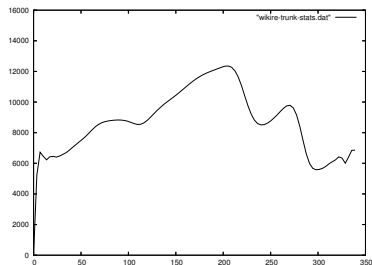
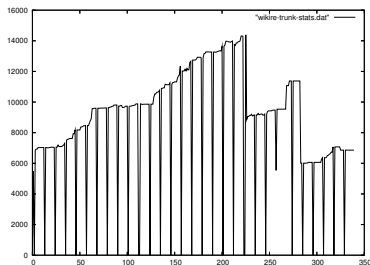
Roteiro para criar uma ação complexa

O roteiro para criar uma nova ação é:

- 1 Criar a ação.
- 2 Criar um descritor (documento XML) para a ação.
 - Recursos necessários.
 - Descrição da ação.
 - URL com documentação (uma wikipage provavelmente).
 - Ícone.

Evolução do código

Evolução do código (LOC x Revisão)



Atualmente com 11000 LOC (e mais um degrau).

Estágio atual

- A procura de um lar (sf.net, berlios?).
- Ainda não é uma ferramenta de gerenciamento de requisitos (falta homens/hora de investimento).
- Apesar a extensibilidade estar adequada, a apresentação e características de preservação de rastreabilidade ainda precisam ser melhoradas.

Conclusões

- Hiperdocumentos são boas soluções para a documentação de requisitos.
 - Muita informação.
 - Informação composta de pedaços menores de informação.
 - Relacionamentos entre os pedaços podem ser definidos e representados.
- Wikis demonstram-se adequadas para a edição desses hiperdocumentos.
- E podem ser boas ferramentas de engenharia de requisitos.

Conclusões

- A experiência com a CoTeia demonstrou o quão viável é a documentação de requisitos com wikis.
- No entanto, é preciso disciplina para tanto.
- E perde-se rastreabilidade (e modificabilidade com preservação de qualidade).
- Mudar a cultura dos projetos de software livre seria muito difícil.

Conclusões

- A Wiki/RE é um primeiro passo de uma ferramenta de gerenciamento de requisitos.
- Vale-se da vantagem de não precisar ser compatível com tecnologias anteriores:
 - Banco de dados, adaptações para a Web, dependência para outras aplicações.
- Utiliza uma estratégia diferente das demais wikis.
- É software livre (são poucas as ferramentas de ER que são livres).
- Aliás, projetos de software livre já utilizam wikis (mas não intensamente ou a sub-utilizando).

Trabalhos futuros

Trabalhos imediatos na ferramenta:

- Concluir a implementação dos casos de uso de rastreabilidade.
- Criar novos tipos de recurso.
- Criar ações que avaliem a qualidade do hiperdocumento (MOV, GLC, etc).
- Implementar novo mecanismo de controle de versão (algo simples ou 100% Java).
- Criação semi-automática de ligações.
- ... (ver TODO).

Trabalhos quanto a engenharia de requisitos em software livre (longo prazo):

- Utilizar a ferramenta em um software livre.
- Avaliar o impacto da documentação de requisitos em software livres:
 - Federação de wikis (push/pull)
 - Configuração de requisitos.
 - Quanta sofisticação é possível/viável?

Publicações

- Silva, M.A.G, Fortes, R.P.M. Processo de Engenharia de Requisitos com Apoio de Hiperdocumentos. VII Workshop de Teses em Engenharia de Software, XVII Simpósio Brasileiro de Engenharia de Software, Manaus - AM, 2003, p. 65-70.
- FORTES, R. P. M.; SILVA, M. A. G.; REIS, C. R. Levantamento sobre Processo de Software Livre. Anais V Workshop sobre Software Livre. Porto Alegre-RS, Junho 2004, pag 207-210.
- FORTES, R. P. M.; SILVA, M. A. G.; FREIRE, A. P.; JUNQUEIRA, D. C. SAFE - Software Engineering Available For Everyone. Anais V Workshop sobre Software Livre. Porto Alegre-RS, Junho 2004, pag 203-206.
- PAIVA D.M.B., MANDUCA A.M., SILVA, M.A.G., QUEMELLO L.J., BRAGA R.T.V, FORTES R.P.M. Reforçando a Comunicação com Uso de uma Ferramenta de Software Livre em Ensino de Engenharia de Software. XIII Workshop sobre Educação em Computação (WEI 2005), XXV Congresso da Sociedade Brasileira de Computação (SBC 2005). p. 2444-2455.