

**PS- 209**

## **SOFTWARE PRODUCT LINES: AN ANALYSIS OF TOOLS THAT SUPPORT APPLICATION REQUIREMENTS ENGINEERING'S PHASE**

Tarcísio Couto Pereira (Universidade Federal de Pernambuco – UFPE, PE, Brasil) –  
tcp@cin.ufpe.br

José Gilson de Almeida Teixeira Filho (Universidade de Pernambuco – UPE, PE, Brasil) –  
jgatif@upe.poli.br

Many companies are interested in improving the quality of software development, increase productivity, reduce costs, decrease time to market and having more flexibility. These interests can be found in approaches that use the principle of software reuse as in Software Product Line (SPL). In the context of the SPL, the activities of requirements engineering (RE) are more complex because they involve more stakeholders and products in addition to need more attention to the similarities and variability of SPL. Because of the great difficulty and importance of capturing the correct requirements for the development of a software product, this paper aims to analyze the main tools that assist the process of Application Requirements Engineering of SPL. In order to perform this analysis, we selected features that were identified in the tools found during the systematic review of literature, and from the construction of a picture was perceived to analyze the tools according to the features selected.

Keywords: Software Product Line (SPL), Application Requirements Engineering, Tools.

## **LINHAS DE PRODUTOS DE SOFTWARE: UMA ANÁLISE DAS FERRAMENTAS QUE APOIAM A FASE DE ENGENHARIA DE REQUISITOS DA APLICAÇÃO**

Muitas empresas estão interessadas em melhorar a qualidade de desenvolvimento de *software*, aumentar a produtividade, reduzir custos, diminuir o *time-to-market* e possuir mais flexibilidade. Estes interesses podem ser encontrados em abordagens que usam o princípio de reuso de *software* como em Linha de Produtos de *Software* (LPS). No contexto de LPS as atividades de engenharia de requisitos (ER) são mais complexas por envolver mais produtos e *stakeholders* além de necessitar mais atenção as semelhanças e variabilidades da LPS. Por causa da grande dificuldade e importância de capturar os requisitos corretos para o desenvolvimento de um produto de *software*, o presente artigo se propõe a analisar as principais ferramentas que auxiliam o processo de Engenharia de Requisitos da Aplicação em LPS. Com a finalidade de realizar esta análise, foram selecionadas funcionalidades que foram identificadas nas ferramentas encontradas durante a revisão sistemática de literatura, e a partir da construção de um quadro ficou perceptível analisar as ferramentas de acordo com as funcionalidades selecionadas.

Palavras-chave: Linha de Produtos de *Software* (LPS), Engenharia de Requisitos da Aplicação, Ferramentas.

## 1. INTRODUÇÃO

Muitas empresas estão interessadas em melhorar a qualidade de desenvolvimento de *software*, aumentar a produtividade, reduzir custos, diminuir o *time-to-market* e possuir mais flexibilidade, estes interesses podem ser encontrados em abordagens que usam o princípio de reuso de *software* como em Linha de Produtos de *Software* (LPS) (POHL *et. al.* 2005; KIM, *et.al.* 2006; DJEBBI, *et.al.* 2007; LISBOA, 2008; LIMA JÚNIOR, 2008; NEIVA, 2009;).

Linhas de produto de *software* (LPS) é um conjunto de sistemas de *software* que dividem um conjunto de *features* (características) comum e gerenciado, que satisfazem as necessidades de um segmento de mercado particular ou missão e que são desenvolvidos a partir de um conjunto comum de artefatos de uma maneira pré-determinada (CZARNECKI, HELSEN e EISENECKER, 2005).

O desenvolvimento de um único sistema é um problema reconhecido durante as últimas décadas. Suas atividades normalmente envolvem muitos *stakeholders* com necessidades diferentes, o que pode resultar em conflitos. Requisitos incompletos e inconsistentes são alguns fatores para o cancelamento ou deficiência dos projetos de *software* (NEIVA, 2009). No contexto de LPS as atividades de engenharia de requisitos (ER) são mais complexas por envolver mais produtos e *stakeholders* além de necessitar de mais atenção as semelhanças e variabilidades da LPS (BEUCHE *et.al.* 2007).

No processo de desenvolvimento de uma linha de produtos, são capturados as semelhanças e variabilidades de um grupo de *software* em certo domínio e, logo então, é executada uma engenharia de domínio que se refere a produção de artefatos comuns aos produtos da linha, de tal maneira que os membros (produtos de *software*) da linha, através de uma engenharia de aplicação, sejam desenvolvidos a partir dos artefatos criados anteriormente, adicionando-se ou não artefatos que irão compor as partes variáveis dos produtos da linha (QUEIROZ, 2009).

De acordo com Sommerville (2006), um processo de *software* é um conjunto de atividades que lida com a produção de um produto de *software*. No contexto de ER, um processo é um conjunto estruturado de atividades que são seguidos pra obter, validar e manter um documento de requisitos do sistema (KOTONYA e SOMMERVILLE, 1998).

O objetivo da engenharia de requisitos da aplicação (ERA) é de elicitar e documentar os artefatos dos requisitos para uma aplicação em particular e ao mesmo tempo reusar, o máximo possível, os artefatos de requisitos do domínio. O reuso dos artefatos de requisitos do domínio para cada aplicação compreende o objetivo geral para obtenção de um alto grau de reuso dos artefatos do domínio (POHL *et.al.*, 2005). A Figura 1 exibe as inter-relações entre a engenharia de aplicação e seus respectivos sub-processos segundo Pohl *et.al.*, 2005.

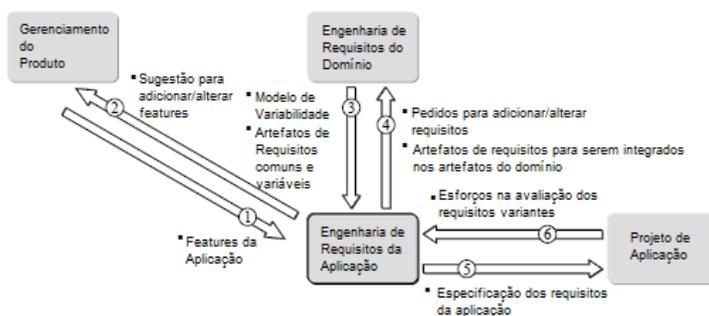


Figura 1 – Fluxo de informação entre a engenharia de requisitos e outros sub-processos

Fonte: Adaptado de Pohl et. al.(2005)

**Inter-relação com o gerenciamento do produto:** Aqui, a ERA usa as partes comuns e escolhe as partes variantes. Como certas *features* são específicas para um único produto, elas precisam ser documentadas e então a ERA faz sugestões para adicionar ou alterar *features* que devem ser incorporadas na LPS.

**Inter-relação com a engenharia de requisitos do domínio:** Aqui, a engenharia de requisitos do domínio oferece o modelo de variabilidade para a ERA definir as variantes e os artefatos do domínio. A ERA pode pedir para adicionar ou alterar requisitos do domínio e se for detectado um requisito específico da aplicação, pode pedir para o artefato de requisito ser integrado nos artefatos de domínio.

**Inter-relação com o projeto de aplicação:** Aqui, a saída é a especificação dos requisitos da aplicação, compostos pelos artefatos do domínio reusáveis e pelo modelo de variabilidade da aplicação derivada do modelo de variabilidade do domínio. A saída também é composta pela avaliação dos requisitos variantes, já que estes possuem um forte impacto na arquitetura da LPS.

O fluxo de informação da engenharia de requisitos da aplicação com relação aos *stakeholders* pode ser percebido na Figura 2, na qual a ERA fornece um modelo de variabilidade e artefatos de requisitos do domínio comuns e variáveis para os *stakeholders*. Estes selecionam os requisitos necessários devolvendo-os, a ERA encaminha o esforço estimado da realização dos requisitos variantes para os *stakeholders* decidirem se estes deverão ser realizados ou não, então os *stakeholders* comunicam a decisão ao engenheiro de aplicação.

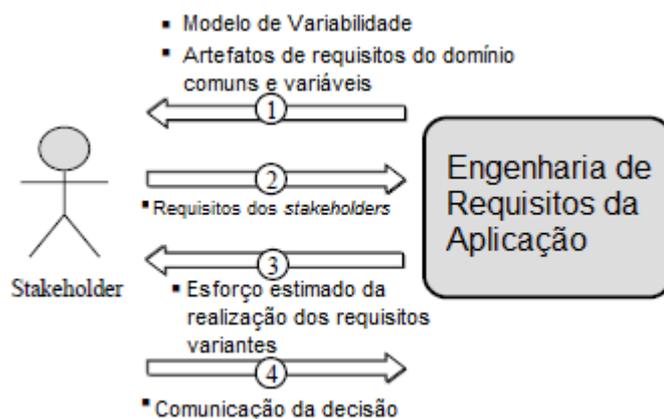


Figura 2 – Fluxo de informação em relação aos stakeholders

Fonte: Adaptado de Pohl et. al. (2005)

A modelagem e o gerenciamento de variabilidade são pré-requisitos muito importantes para a engenharia de aplicação em linha de produtos de *software* no processo de configuração e derivação do produto. Muitos trabalhos que abordam sobre configuração e derivação do produto não focam o suficiente em ferramentas de suporte, havendo necessidade de fornecer um suporte de ferramenta integrada tanto para a configuração do produto quanto para a engenharia de requisitos da aplicação (RABISER, 2007; ARAUJO, 2011).

A engenharia de requisitos da aplicação é responsável por elicitar e documentar os artefatos dos requisitos para uma aplicação em particular e ao mesmo tempo reusar, o máximo possível, os artefatos de requisitos do domínio.

Com o objetivo de tornar o processo de engenharia de requisitos da aplicação o mais eficiente possível Botterweck, *et. al.* (2008), diz que uma ferramenta visual e interativa é necessária para se beneficiar dos ganhos de produtividade.

Por causa da grande dificuldade e importância de capturar os requisitos corretos para o desenvolvimento de um produto de *software*, o presente trabalho se propõe através de uma revisão sistemática de literatura, analisar as principais ferramentas que auxiliam o processo de engenharia de requisitos da aplicação da engenharia de linha de produtos de *software*.

## 2. REVISÃO SISTEMÁTICA DA LITERATURA

A revisão sistemática é uma forma de pesquisa que utiliza como fonte de dados a literatura sobre um determinado tema. Esse tipo de investigação disponibiliza um resumo das evidências relacionadas a uma estratégia de intervenção específica, mediante a aplicação de métodos explícitos e sistematizados de busca, apreciação crítica e síntese da informação selecionada (LITTELL *et al.*, 2008; MAFRA e TRAVASSOS, 2006; BIOLCHINI *et al.* 2005; DYBA *et al.*, 2005; KITCHENHAM *et al.*, 2004).

As fases de condução da revisão sistemática e as atividades utilizadas neste trabalho foram adaptadas de trabalhos de referência citados na literatura (LITTELL *et al.*, 2008; BIOLCHINI *et al.*, 2005; KITCHENHAM, 2004; PAI *et al.*, 2004).

As fases do processo de revisão sistemática apresentadas na Figura 3, não são necessariamente sequenciais, pois possuem iterações, fazendo com que muitas atividades sejam iniciadas na fase de planejamento e refinadas posteriormente.

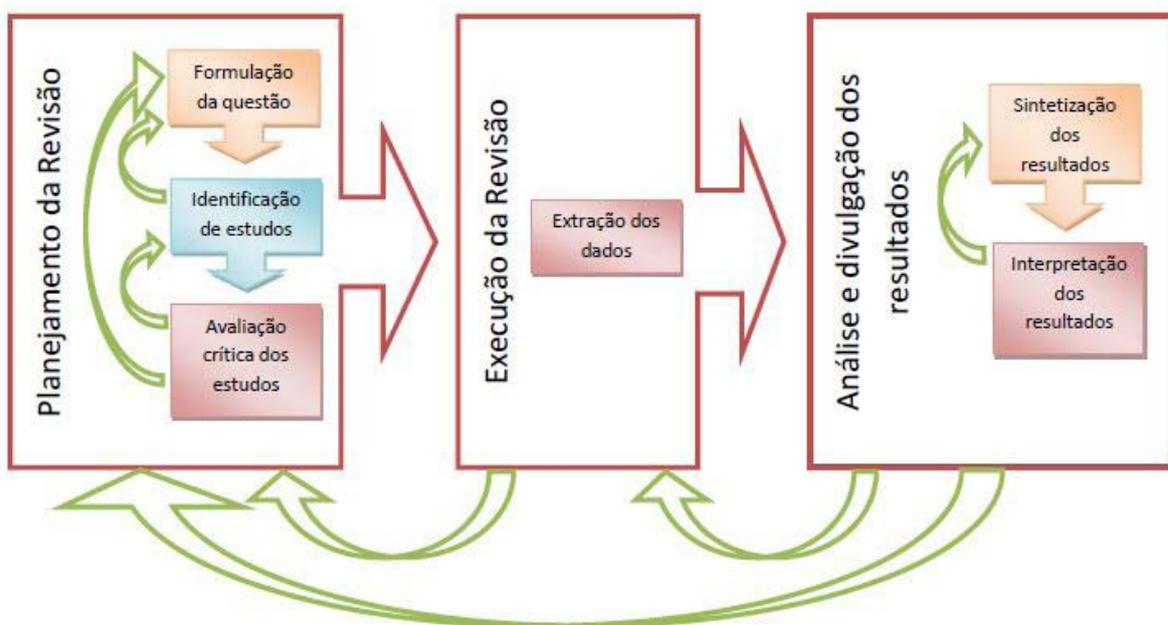


Figura 3 – Processo de Condução da Revisão Sistemática

Fonte: (BIOLCHINI *et al.*, 2005; KITCHENHAM *et al.*, 2004; PAI *et al.*, 2004; LITTEL *et al.*, 2008).

Na fase de planejamento foram estabelecidos os objetivos da pesquisa e foi criado um protocolo de revisão contendo itens como: identificação e seleção das bases de dados, métodos de busca e palavras-chave, estratégia de busca e critérios de inclusão e exclusão.

Ainda na fase de planejamento, foram gerados quatro formulários: o formulário de condução da revisão (FCR) que serviu para a catalogação dos estudos encontrados durante a execução das etapas descritas no processo; o formulário de aprovação dos trabalhos (FAT) que serviu para a catalogação final dos estudos; o formulário de teste piloto (FTP) que serviu para avaliar o protocolo antes mesmo dele ser aplicado na íntegra, ou seja, foi adotado para os testes do protocolo de revisão e o formulário de críticas do avaliador (FCA) que contribuiu para fichar as críticas construtivas/melhorias sugeridas por três avaliadores.

Na fase de execução da revisão foi realizada a condução e análise dos estudos primários. Os estudos foram identificados, coletados e organizados em uma lista estruturada. Após a aplicação, os resultados ainda passaram por uma reavaliação para garantir que não foram eliminados estudos relevantes. Ao final desta fase, as informações foram extraídas somente dos estudos selecionados (KITCHENHAM, 2004; PAI et al., 2004).

Na etapa I de seleção definida no protocolo de pesquisa da revisão sistemática, foram considerados os critérios de qualidade e as strings de busca. Foram obtidos no total 7153 estudos (ver Figura 4).

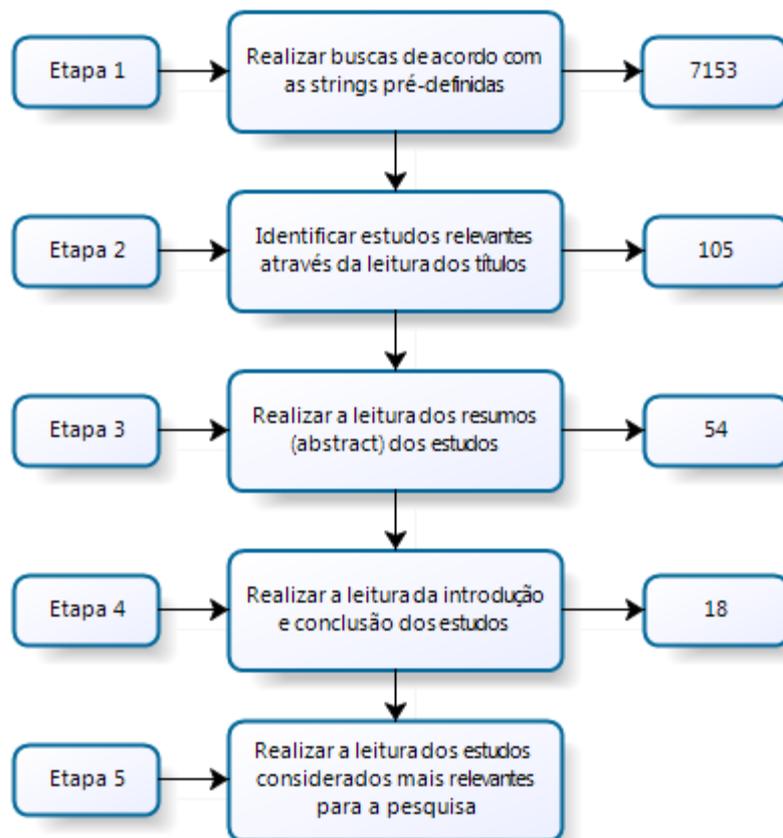


Figura 4 - Etapas para seleção dos trabalhos  
Fonte: Pereira (2011).

A Tabela 1 exibe os resultados da pesquisa em cada base a partir das strings definidas no protocolo. É importante destacar que as strings foram selecionadas de acordo com a disponibilidade dos estudos nos dois idiomas escolhidos, de forma que nas bases em que não havia a possibilidade de realizar a pesquisa em Inglês, não foram utilizadas as strings do respectivo idioma, da mesma forma ocorreu para o idioma Português. A base Google Scholar foi a única que possibilitou o uso de todas as strings, pois a mesma permitia pesquisar nos dois idiomas.

Tabela 1 – Resultado da pesquisa relacionando as bases com as Strings adotadas

Bases/Strings	Português		Inglês		Total
	1	2	1	2	
<b>BASE/STRING</b>	<b>1</b>	<b>2</b>	<b>1</b>	<b>2</b>	<b>Total</b>
<b>ACM</b>	-	-	1243	2440	<b>3683</b>
<b>BDTD</b>	73	93	-	-	<b>166</b>
<b>IEEE Xplore</b>	-	-	1170	800	<b>1970</b>
<b>SciELO</b>	1	2	7	0	<b>10</b>
<b>Science Direct</b>	-	-	114	120	<b>234</b>
<b>Google Scholar</b>	195	95	400	400	<b>1090</b>
<b>Total</b>	<b>269</b>	<b>187</b>	<b>2934</b>	<b>3760</b>	<b>7153</b>

Fonte: Pereira (2011).

Na etapa II foram lidos os títulos dos trabalhos e analisados quais estavam disponíveis por completo. Sendo assim, foram selecionados 105 dos 7153 artigos resultantes da busca inicial. Na etapa III, foram lidos os títulos dos trabalhos, a seleção foi aprimorada com base nos critérios de inclusão e exclusão e a partir da leitura do resumo (*abstract*), foram selecionados 54 dos 105 trabalhos relatados até então. Na etapa IV, os estudos passaram por uma leitura da introdução e conclusão e dos 54 trabalhos, 18 foram selecionados ao final. A etapa V estabelece a leitura completa dos estudos selecionados. Essa leitura final foi realizada de acordo com os propósitos estabelecidos pela pesquisa, ou seja, os estudos foram lidos com o intuito de analisar as ferramentas que apóiam a fase de engenharia de requisitos da aplicação em linhas de produtos de *software*.

No Gráfico 1, observa-se que os trabalhos selecionados foram publicados entre 2005 e 2011, sendo o ano de 2007 o que obteve mais resultados, com um total de 5 resultados, seguido pelo ano de 2008, 2009 e 2010 com 3 trabalhos, 2006 com 2 trabalhos e 2005 e 2011 com 1 trabalho cada.

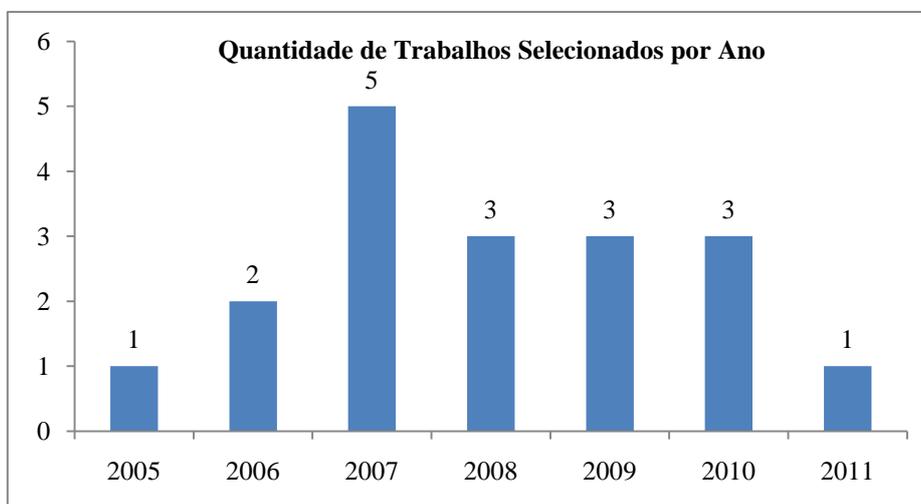


Gráfico 1 – Quantidade de trabalhos relevantes por ano  
Fonte: Pereira (2011).

### 3. METODOLOGIA

Segundo Andrade (2006), pesquisa é um conjunto de procedimentos sistemáticos, baseado no raciocínio lógico, que tem por objetivo encontrar soluções para problemas propostos, mediante a utilização de métodos científicos. Para atingir o objetivo geral desta pesquisa foram utilizados diversos procedimentos e técnicas apresentados no Quadro 1 e detalhados a seguir.

Quadro 1 – Resumo da metodologia escolhida

<b>Natureza da pesquisa</b>	Aplicada
<b>Do ponto de vista de seus objetivos</b>	Descritiva e Exploratória
<b>Procedimentos técnicos da pesquisa</b>	Bibliográfica (Revisão Sistemática)
<b>Formas de abordagem</b>	Qualitativa

Fonte: Pereira (2011).

#### 3.1. Natureza da Pesquisa

Quanto à natureza, de acordo com Silva (2004), a pesquisa objetiva a produção de conhecimentos e aplicações práticas, voltados à solução de problemas reais e específicos. Sendo assim, este trabalho busca, por meio da teoria, aplicar os conhecimentos advindos de diversas fontes a entidades de interesse particular, ou seja, no âmbito da engenharia de requisitos em Linha de Produtos de *Software*.

#### 3.2. Quanto aos Fins

A pesquisa exploratória procura aumentar a familiaridade com um problema, procurando torná-lo mais explícito, envolvendo a procura e levantamento bibliográfico em uma ou mais áreas pertinentes à pesquisa (GIL, 1996; JUNG, 2004). Já a pesquisa descritiva procura observar, registrar, analisar, classificar e interpretar os resultados da pesquisa por meio de técnicas padronizadas de coleta de dados, sem, no entanto, sofrer interferências do pesquisador (SILVA, 2004).

Deste modo, esta pesquisa será do tipo exploratória, pois envolverá análise e levantamento sistemático da literatura a cerca de engenharia de requisitos em Linha de Produtos de *Software*, e descritiva a partir da análise das ferramentas.

#### 3.3. Quanto aos Meios

O procedimento técnico aplicado neste trabalho envolve a pesquisa bibliográfica (Revisão Sistemática da Literatura) que ocorre quando permite ao pesquisador, através de material relevante já publicado, a cobertura de uma gama de fenômenos muito mais abrangente do que se fosse pesquisar diretamente (GIL, 1996). Dessa forma, este trabalho visa por meio da exploração bibliográfica identificar e selecionar ferramentas de engenharia de requisitos da aplicação em Linha de Produtos de *Software* dispostas na literatura, e a partir de uma análise destas ferramentas, permitir a universitários, pesquisadores e empresas de TI obter uma visão melhor sobre funcionalidades empregadas pelas ferramentas com o propósito de escolher a que melhor se encaixar no contexto trabalhado.

#### 3.4. Formas de Abordagem

Quanto à forma de abordagem, uma pesquisa que procura fundamentar verdades de interpretação, onde o pesquisador tem o papel de interpretador e não prioriza a quantificação nos resultados, vem a ser uma pesquisa qualitativa (GIL, 1996). Segundo Gil (1996), a pesquisa com enfoque qualitativo baseia-se na elicitação e avaliação de modelos conceituais, a partir de descrições intuitivas do pesquisador.

Desta maneira, através de uma análise sucinta da literatura, serão levantados insumos teóricos para identificar e analisar as ferramentas como resultado.

## **4. RESULTADOS**

### **4.1. Critérios de Seleção das Ferramentas**

Logo após a seleção dos principais trabalhos fez-se necessário escolher critérios para seleção das ferramentas que serão avaliadas, visando alcançar o objetivo desta pesquisa, que é focado na Engenharia de Requisitos da Aplicação (PEREIRA, 2011).

#### **4.1.1. Critérios de Inclusão**

Os critérios definidos para inclusão das ferramentas encontradas para análise:

- Ferramentas que apoiem a fase de Engenharia de Requisitos da Aplicação com alguma funcionalidade;
- Ferramentas com a documentação de suas funcionalidades ou características.

Estes critérios devem estar presentes em todas as ferramentas encontradas na pesquisa, tendo em vista o grande número de ferramentas retornado.

#### **4.1.2. Critérios de Exclusão**

Os critérios definidos para a exclusão das ferramentas encontradas para análise:

- Ferramentas que apoiam outras fases da LPS que não a fase de Engenharia de Requisitos da Aplicação;
- Ferramentas sem documentação que trate de suas funcionalidades ou características.

Os critérios foram aplicados após a leitura completa da documentação ou referência da ferramenta, quando encontrados.

### **4.2. Ferramentas Selecionadas**

Após a realização da revisão sistemática da literatura, foram encontradas 33 ferramentas que apoiam o processo de Engenharia de Requisitos em Linha de Produtos de *Software*. Nas 33 ferramentas identificadas durante o processo de revisão sistemática da literatura, foram aplicados os critérios de inclusão e exclusão, resultando em 22 ferramentas que apoiam a fase de engenharia de requisitos da aplicação, e serão analisadas, ver Gráfico 2 (PEREIRA, 2011). Uma síntese da documentação, da quantidade das ferramentas e seus respectivos nomes podem ser encontrados no Quadro 2.

## Quantidade de Ferramentas que Apoiam a ERA

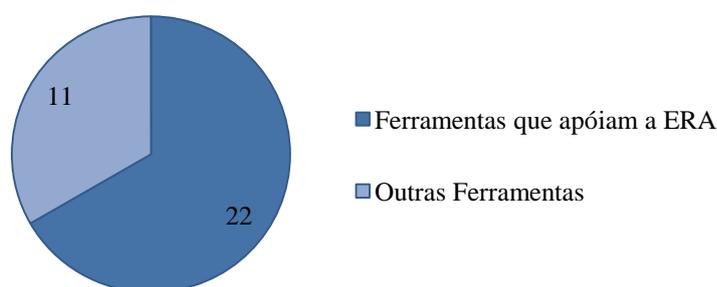


Gráfico 2 – Quantidade de Ferramentas que Apoiam a ERA  
Fonte: Pereira, (2011).

Quadro 2 – Citações de Ferramentas pelos Autores

Ferramentas	Autores / Documentação										
	KIM <i>et.al.</i> (2006)	BEUCHE <i>et.al.</i> (2007)	RABISER e DHUNGANA (2007)	DEHLINGER <i>et.al.</i> (2007)	RABISER <i>et.al.</i> (2007)	DJEBBI <i>et.al.</i> (2007)	BEUCHE (2007)	BOTTERWECK <i>et.al.</i> (2008)	LIMA JÚNIOR (2008)	LISBOA (2008)	ARAUJO (2011)
001											•
Ami Eddi											•
AORA											•
ASADAL	•									•	
Captain Feature										•	
Configuration Wizard			•		•					•	
Decimal		•		•						•	
DOPPLER			•								
FeatureIDE										•	
Feature Plugin										•	
GEARS									•	•	
Holmes										•	
PLFaultCAT				•							
Pluss toolkit										•	
ProjectKing			•		•						
Pure::Variants						•			•	•	
Remap		•									
RequiLine		•				•	•			•	
ToolDay										•	
TREK						•					
VISIT_FC								•			
Xfeature						•			•	•	

Fonte: Adaptado de Pereira, (2011).

### 4.3. Funcionalidades das Ferramentas

As funcionalidades descritas para análise foram selecionadas a partir dos trabalhos identificados na revisão sistemática de literatura. Estas funcionalidades foram escolhidas após a leitura completa dos trabalhos identificados durante a revisão sistemática de literatura. O Quadro 3 mostra qual funcionalidade cada ferramenta oferece suporte. Os

números romanos se referem as funcionalidades descritas a seguir. Este quadro facilita a identificação de lacunas nas ferramentas selecionadas, e além do mais, pode ajudar a descobrir qual ferramenta melhor satisfaz a necessidade de uma empresa.

- (i) **Identificação de *Features* Comuns e Variáveis:** Oferecendo uma maneira automática e sistemática para identificar *features* comuns e variáveis.
- (ii) **Identificação de *Features* Através dos RNF:** Identificação automática de informações existentes do catálogo de requisitos não-funcionais correspondente.
- (iii) **Relação entre Características e Requisitos:** Relata a existência de *features* de um domínio para os requisitos (funcionais ou não-funcionais), e/ou casos de uso definidos. Através desta relação é possível manter a rastreabilidade entre os artefatos produzidos no domínio.
- (iv) **Capacidade de Gerenciar os Requisitos na Derivação:** Usuários necessitam capturar requisitos específicos do produto decorrente da derivação;
- (v) **Suporte para os Requisitos Específicos do Produto Relacionados com a Variabilidade Existente:** Relações entre os requisitos específicos do produto e a disponibilidade das *features* da linha de produto são importantes para apoiar a evolução da linha de produto. Engenheiros deverão estar disponíveis para analisar os novos requisitos e decidir se eles deverão se tornar parte da linha de produto no futuro;
- (vi) **Visões Gráficas dos Requisitos:** Implementação de várias técnicas de visualização e interação que pode ajudar os *stakeholders* no processo de configuração do produto em uma LPS.
- (vii) **Apoiar o Planejamento para a Evolução da LPS:** Os novos requisitos, comentários dos clientes e as experiências dos usuários durante a derivação do produto necessitam serem considerados quanto a manutenção e evolução da linha de produtos de *software*.
- (viii) **Derivação do Produto:** Identifica as *features* que pertencem a um produto de acordo com as *features* definidas no domínio.
- (ix) **Emissão de Relatórios:** Geração de relatórios a partir de informações disponíveis no domínio. Os relatórios podem representar, por exemplo, o número de combinações possíveis; a frequência que uma *feature* aparece em um produto; a documentação dos artefatos.
- (x) **Documentação do Produto:** Apresenta uma documentação para todo produto com informações como: descrição do produto e versão do domínio.
- (xi) **Suporte a tomada de decisões para pessoas não técnicas:** A ferramenta permite anexar material de apoio como imagens, arquivos de áudio e/ou vídeos para ajudar pessoas não técnicas para tomar decisões em diferentes fases da derivação do produto.

Quadro 3 – Ferramentas e suas funcionalidades

Ferramentas	Funcionalidades										
	i	ii	iii	iv	v	vi	vii	viii	ix	x	xi
001	•			•		•		•	•		
Ami Eddi	•			•				•			
AORA	•	•	•	•		•		•			
ASADAL	•			•	•	•		•			
Captain Feature	•			•		•		•			
Configuration Wizard	•			•	•	•		•			•
Decimal	•			•	•		•	•			

DOPPLER	•		•	•		•	•		•
FeatureIDE	•		•		•		•		
Feature Plugin	•		•		•		•		
GEARS	•		•				•	•	
Holmes	•		•		•		•		
PLFaultCAT	•		•	•		•	•		
Pluss toolkit	•	•	•				•	•	
ProjectKing	•		•		•				•
Pure::Variants	•		•				•	•	
Remap	•		•	•					
RequiLine	•	•	•				•	•	•
ToolDay	•	•	•				•	•	•
TREK	•		•						
VISIT_FC	•		•		•				•
Xfeature	•		•		•		•		•

Fonte: Adaptado de Pereira, (2011).

Através do Gráfico 3, pode-se observar que as funcionalidades *Identificação de Features Comuns e Variáveis* e *Gerenciamento de Requisitos na Derivação* estão presentes nas 22 ferramentas e a funcionalidade *Derivação do Produto* está presente em quase todas, totalizando 18 ferramentas.

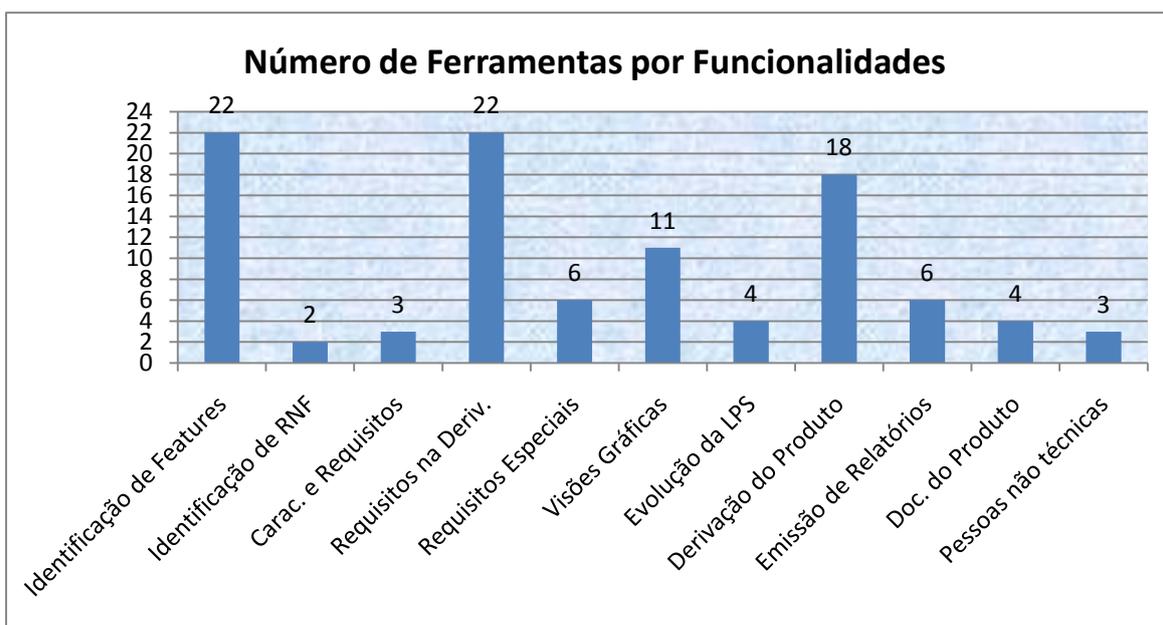


Gráfico 3 – Número de Ferramentas por Funcionalidades

Fonte: Pereira, (2011).

As três funcionalidades são essenciais em ferramentas de engenharia de requisitos da aplicação, pois sem elas não há razão de construir uma ferramenta que apoie a ERA.

Também é possível identificar que as ferramentas não apoiam muito a *Identificação de RNF*, apenas 2 ferramentas oferecem este recurso, além do mais os RNF são de suma importância e deve haver um caminho para analisar e projetar esses requisitos desde o início do ciclo de vida do desenvolvimento de *software* (NGUYEN, 2009).

Em seguida, estão as funcionalidades *Relação entre Características e Requisitos* e *Suporte a Tomada de Decisões para Pessoas Não Técnicas* suportadas por 3 ferramentas cada. Logo após, *Evolução da LPS* e *Documentação do Produto* com 4 ferramentas, *Gerenciamento de Requisitos Especiais* e *Emissão de Relatórios* disponíveis em 6 ferramentas e *Visões Gráficas dos Requisitos* presente em 11 ferramentas.

Assim, toda ferramenta que deseje apoiar efetivamente a fase de engenharia de requisitos da aplicação em uma linha de produtos de *software* deve possuir as três funcionalidades acima, já que estão presentes em quase todas as ferramentas e são essenciais para a construção de uma ferramenta que apoie a ERA.

Baseado no Gráfico 3 é visível que ainda há muitas funcionalidades que são apoiadas por poucas ferramentas. Assim, existe uma lacuna a ser preenchida pelas ferramentas no processo de engenharia de requisitos da aplicação, porque há funcionalidades que estão presentes em poucas ferramentas.

A falta de uma ferramenta com um alto número de funcionalidades implica no grau de adoção de ferramentas por empresas, porque pode forçá-los a usar várias ferramentas ao mesmo tempo, uma para cada funcionalidade. Este cenário deve ser evitado, uma vez que, as chances de ocasionar um erro são muito maiores (LISBOA, 2008).

## 5. CONCLUSÕES

A engenharia de requisitos da aplicação é responsável por elicitar e documentar os artefatos dos requisitos para uma aplicação em particular e ao mesmo tempo reusar, o máximo possível, os artefatos de requisitos do domínio.

Botterweck, *et. al.* (2008) afirma que uma ferramenta visual e interativa é necessária para se beneficiar dos ganhos de produtividade e tornar o processo de engenharia de aplicação o mais eficiente possível.

Por causa da grande dificuldade e importância de capturar os requisitos corretos para o desenvolvimento de um produto de *software*, foi realizada uma revisão sistemática de literatura, onde foi possível identificar e selecionar ferramentas a partir de critérios estabelecidos. Ao todo, foram identificadas 33 ferramentas que apoiam alguma fase da engenharia de requisitos em linha de produtos de *software*. A partir desta identificação, foram selecionadas ferramentas que apoiam a fase de Engenharia de Requisitos da Aplicação, totalizando 22 ferramentas (PEREIRA, 2011).

Depois de selecionar as 22 ferramentas, foi construído um quadro com o mapeamento das funcionalidades por cada ferramenta. Este quadro facilita a identificação de lacunas nas ferramentas selecionadas, e além do mais, pode ajudar a descobrir qual ferramenta melhor satisfaz a necessidade de uma empresa.

## 6. RECOMENDAÇÕES

As empresas de tecnologia e acadêmicos que pretendem usar linha de produtos para o desenvolvimento de *software* devem utilizar o quanto antes à análise das ferramentas realizada nesta pesquisa, pois estarão consequentemente identificando qual ferramenta melhor se encaixa no contexto em que será trabalhado, reduzindo o tempo na busca de uma ferramenta e evitando erros, escolhendo uma ferramenta que não apoie os processos de desenvolvimento desejados. A análise é composta de 11 funcionalidades que foram escolhidas após a leitura completa dos trabalhos identificados durante a revisão sistemática de literatura e de 22 ferramentas que apoiam a fase de engenharia de requisitos da aplicação, também identificadas durante a revisão sistemática.

## REFERÊNCIAS BIBLIOGRÁFICAS

ANDRADE, Maria Margarida de. **Introdução à metodologia do trabalho científico: elaboração de trabalhos na graduação**. 7. ed. São Paulo: Atlas, 2006.

ARAUJO, João; VARELA, Patrícia; BRITO, Isabel; MOREIRA, Ana. **Aspect-Oriented Analysis for Software Product Lines Requirements Engineering**. ACM, 2011.

BEUCHE, Danilo. *et.al.* **Using Requirements Management Tools in Software Product Line Engineering - The State of the Practice**. International Software Product Line Conference, 2007.

BIOLCHINI, J.; MIAN, P. G.; NATALI, A. C.; TRAVASSOS, G. H. **Systematic Review in Software Engineering: relevance and utility**. Relatório Técnico RT-ES-679/05, Programa de Engenharia de Sistemas e Computação (PESC), COPPE/UFRJ, 2005.

BOTTERWECK, G.; Thiel, S.; NESTOR, D.; SAAD, bin A.; CAWLEY C.; **Visual Tool Support for Configuring and Understanding Software Product Lines**, Lero, The Irish Software Engineering Research Centre, 2008.

CZARNECKI, K., HELSEN, S. e EISENECKER, U.. **Formalizing cardinality-based feature models and their specialization**. Software Process Improvement and Practice, 10(1):7–29, jan/mar 2005.

DJEBBI, Olfa. *et.al.* **Industry Survey of Product Lines Management Tools - Requirements, Qualities and Open Issues**. International Requirements Engineering Conference, 2007.

DYBA, T.; KAMPENES, V.; SJOBERG, D. **A Systematic Review of Statistical Power in Software Engineering Experiments**. Journal of Information and Software Technology, v. 1, n. 11, 2005.

GIL, Antônio Carlos. **Como elaborar projetos de pesquisa**. 3. ed. São Paulo: Atlas, 159 p., 1996.

JUNG, Carlos Fernando. **Metodologia para pesquisa e desenvolvimento: aplicada a novas tecnologias, produtos e processos**. Rio de Janeiro: Axcel Books do Brasil, 2004.

KIM, Kyungseok. *et.al.* **ASADAL: A Tool System for Co-Development of Software and Test Environment based on Product Line Engineering**. ACM, 2006.

KITCHENHAM, B. A.; DYBA, T.; JORGENSEN, M. **Evidence-based Software Engineering**. Proceedings of the 26th International Conference on Software Engineering (ICSE'04), p. 273-281, 2004.

KOTONYA, G. and SOMMERVILLE, I. **Requirements Engineering: Processes and Techniques**. John Wiley & Sons, Inc. New York, NY, USA (1998).

LIMA JÚNIOR, Rogério. **Comparação entre Ferramentas para Linha de Produtos de Software**. Recife. Trabalho de Conclusão de Curso (Engenharia da Computação), Universidade de Pernambuco. Recife, PE (2008).

LISBOA, Liana. **ToolDay - A Tool for Domain Analysis**. Recife. Dissertação (Ciência da Computação), Universidade Federal de Pernambuco – Centro de Informática. Recife, PE (2008).

LITTELL J. H.; CORCORAN J.; PILLAI, V. **Systematic Reviews and Meta-analysis**. USA: Oxford University Press, 2008.

MAFRA, S. N.; TRAVASSOS, G. H. **Estudos Primários e Secundários apoiando a busca por Evidência em Engenharia de Software**. Relatório Técnico RT-ES-687/06, Programa de Engenharia de Sistemas e Computação (PESC), COPPE/UFRJ, 2006.

NEIVA, D.F.S., **RiPLE-RE: A Requirements Engineering Process for Software Product Lines**, M.Sc. Dissertação, Universidade Federal de Pernambuco, Brasil, 2009.

NGUYEN, Quyen. **Non-Functional Requirements Analysis Modeling for Software Product Lines**. IEEE Computer Society, 2009.

PAI, M.; MCCULLOCH, M.; GORMAN, J.; PAI, N.; ENANORIA, W.; KENNEDY, G.; THARYAN, P.; COLFORD JR., J. **Systematic reviews and meta-analyses: An illustrated step-by-step guide**. The National Medical Journal of India, v. 17, n. 2, p. 86-95, 2004.

PEREIRA, T. C., **Linhas de Produtos de Software: Uma Análise das Ferramentas para Melhoria da Engenharia de Requisitos**. Caruaru. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) - Universidade de Pernambuco (UPE), Caruaru, PE (2011).

POHL, Klaus.; BÖCKLE, Gunter.; VAN DER LINDEN, Frank.: **Software Product Line Engineering – Foundations, Principles, and Techniques**. Springer, Heidelberg 2005.

QUEIROZ, Paulo Gabriel Gadelha. **Uma abordagem de desenvolvimento de linha de produtos com uma arquitetura orientada a serviços**. Dissertação (Mestrado em Ciências da Computação e Matemática Computacional) – Universidade de São Paulo, São Paulo, 2009.

RABISER, Rick. et.al. **Involving Non-Technicians in Product Derivation and Requirements Engineering - A Tool Suit for Product Line Engineering**. International Requirements Engineering Conference, 2007.

SILVA, C. R. O. **Metodologia e Organização do projeto de pesquisa**. Fortaleza, 34 p., 2004.

SOMMERVILLE, I. **Software Engineering**. Addison-Wesley, 8th edition (2006).