

# Refinamento de Competências do Egresso do Curso de Engenharia de Software

Daltro J. Nunes<sup>1</sup>, Marcelo H. Yamaguti<sup>2</sup>, Ingrid Nunes<sup>1</sup>

<sup>1</sup>Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)  
Porto Alegre – RS – Brasil

<sup>2</sup>Faculdade de Informática – Pontifícia Universidade Católica do Rio Grande do Sul  
(PUCRS)  
Porto Alegre – RS – Brasil

{daltro,ingridnunes}@inf.ufrgs.br, yamaguti@pucrs.br

***Resumo.** O número de cursos de graduação de Engenharia de Software cresce a cada ano no Brasil. A fim de dar orientação a estes novos cursos, ou mesmo homogeneizar os existentes, a comunidade de Engenharia de Software do Brasil trabalhou ao longo de quatro anos para consolidar um Refinamento de Competências do Egresso do Curso de Engenharia de Software. Diversas universidades participaram do trabalho, e participantes do Fórum de Educação em Engenharia de Software (FEES) contribuíram nos anos de 2013 a 2015. Neste artigo, apresentamos um relatório deste trabalho e o seu resultado.*

## 1. Introdução

O número de cursos de graduação de Engenharia de Software cresce a cada ano no Brasil. A fim de dar orientação a estes novos cursos, ou mesmo homogeneizar os existentes, foi iniciado um trabalho para identificar conteúdos que devem fazer parte do currículo do curso. Este trabalho consiste da dedução de conteúdos necessários para que egressos do Curso de Engenharia de Software tenham determinadas competências. As competências foram obtidas das Diretrizes Curriculares Nacionais (DCNs) para os cursos de graduação em Computação (MEC, 2012).

O Conselho Nacional de Educação (CNE) estabeleceu as bases para construção de Diretrizes Curriculares (PARECER N.º: CNE/CES 67/2003), abandonando a abordagem conteudista (adotada nos currículos mínimos) e adotando a abordagem baseada em competências (adotada nas Diretrizes Curriculares Nacionais).

A metodologia, desenvolvida pelo Prof. Daltro Nunes, foi inspirada em uma matriz, utilizada pelo INEP no projeto da prova do ENADE. Ele foi convidado pela Universidade de Goiás para discutir o Projeto Pedagógico do Curso (PPC) de Engenharia de Software na qual foi discutida a adaptação da matriz do ENADE no desenvolvimento de currículos. A metodologia para produzir conteúdos com base em refinamentos de habilidades e competências foi, entretanto, estabelecida mais tarde. Convidado para avaliar o currículo do curso de Engenharia de Software da Universidade de Brasília, Campus Gama, a metodologia foi discutida e aprimorada chegando-se a forma atual. A metodologia foi amplamente discutida e aplicada pela comunidade de Engenharia de Software do Brasil – no contexto do Fórum de Educação em Engenharia de Software (FEES), parte do Simpósio Brasileiro de Engenharia de Software (SBES), nos anos de

2013 a 2015. Alguns seminários foram realizados na PUCRS com a participação de diversos professores de várias universidades gaúchas.

Neste documento, são descritos detalhes relativos ao trabalho realizado, bem como o resultado obtido.

## 2. Metodologia

O trabalho realizado baseia-se na educação por competências. Em vez de se fazer uma listagem *ad hoc* de conteúdos a serem ministrados, há um questionamento inicial a respeito de quais competências o egresso do curso deve possuir, conforme determinação do CNE. Competência é a “capacidade de articular e mobilizar *conhecimentos, habilidades e atitudes*, colocando-os em ação para resolver problemas e enfrentar situações de imprevisibilidade em uma dada situação concreta de trabalho e em um determinado contexto cultural”. A partir destas competências, que são refinadas em níveis de granularidade cada vez mais baixos, listam-se, ao final, conteúdos necessários para adquiri-las. O resultado é uma relação de ordem (grafo dirigido) onde os nodos terminais são conteúdos.

O ponto de partida do trabalho foram as Diretrizes Curriculares Nacionais (DCNs) para os cursos de graduação em Computação (MEC, 2012) – a adequação das diretrizes não foi questionada neste trabalho. Das DCNs, foram extraídas informações relativas ao curso de Bacharelado em Engenharia de Software. As habilidades e competências específicas do egresso de Engenharia de Software, descritas no artigo 5º, parágrafo 3º das DCNs foram consideradas como as competências do nível inicial.

Uma competência de nível N pode, no nível N+1, (i) ser refinada em duas ou mais competências de granularidade mais fina, ou (ii) ter conteúdos apropriados listados para que o estudante adquira aquela competência. Como resultado, as folhas (último nível) do refinamento de cada uma das competências oriundas das DCNs consistem de conteúdos.

Para o detalhamento dos refinamentos de competências, foi utilizada uma terminologia para evitar ambiguidade dos termos. Essa terminologia tem origem na taxonomia utilizada no *Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering* da ACM/IEEE (ACM/IEEE, 2004), que consiste em conhecimento, entendimento e aplicação.

A seguir, cada um dos termos utilizados são especificados.

- **Conhecer:** lembrar do material previamente ensinado. Testa a observação e recuperação da informação, isto é, “trazer à mente a informação apropriada”.
- **Compreender:** entender a informação e o significado do material apresentado. Por exemplo, ser capaz de traduzir o conhecimento a um novo contexto, interpretar fatos, comparar, contrastar, ordenar, agrupar, inferir causas, prever consequências, etc.
- **Aplicar:** usar o material aprendido em situações novas e concretas. Por exemplo, usando informação métodos, conceitos, teorias para resolver problemas que requerem as habilidades e conhecimento apresentados.

No trabalho realizado, “Aplicar” engloba “Compreender” que por sua vez engloba “Conhecer”.

### 3. Ciclos de discussão

O trabalho foi realizado em vários ciclos de reuniões de grupos locais, grupos de trabalho no FEES e discussões remotas em grupos de área. Precisamente, após a reunião na UnB, que resultou em uma primeira versão dos refinamentos das competências, chegando-se a conteúdos, foram realizadas rodadas de trabalho, as quais são detalhadas a seguir.

1. Em setembro de 2013 houve uma reunião, coordenada pelo Prof. Daltro Nunes, com representantes da UFRGS, PUCRS, UNIPAMPA, UNILASALLE, na PUCRS em Porto Alegre, que refinou algumas competências com base no trabalho inicial feito.
2. Em outubro de 2013 ocorreu, no FEES 2013 em Brasília, uma sessão de refinamento com os participantes do evento, coordenada pelo Prof. Daltro Nunes e Prof. Hilmer Neri.
3. Em dezembro de 2013, houve nova reunião, coordenada pelo Prof. Daltro Nunes, com representantes da UFRGS, PUCRS, UNIPAMPA, UNILASALLE, na PUCRS em Porto Alegre, para nova sessão de refinamentos e ajustes.
4. Em janeiro de 2014 foi criado um fórum Web (<https://groups.google.com/forum/?hl=pt-BR#!forum/engswpropostacurriculo>) para discussão com interessados no trabalho em desenvolvimento, gerenciado pelo Prof. Marcelo Yamaguti.
5. Em maio de 2014, a PUC Minas, em Belo Horizonte, colaborou com o refinamento de competências comuns e conteúdos, relatado pela Profa. Maria Augusta Nelson.
6. Em outubro de 2014, no FEES 2014, em Maceió, houve nova sessão de refinamento com os presentes, coordenado pelo Prof. Daltro Nunes e Prof. Marcelo Yamaguti. O intuito original era de se finalizar o refinamento e consolidação de todas as competências. Entretanto, após várias discussões, conseguiu-se efetivamente consolidar apenas 2 de 14 competências. Assim, foram criados grupos de trabalhos para refinamento de cada competência, por sugestão do Prof. José Carlos Maldonado.
7. Em novembro de 2014, cada grupo relatou os seus resultados no fórum Web ou por e-mails.
8. Em abril de 2015, houve uma reunião, coordenada pelo Prof. Daltro Nunes, com representantes da UFRGS, PUCRS, UNIPAMPA, na PUCRS em Porto Alegre, para ajustes de consistência dos refinamentos gerados pelos diversos grupos.
9. Em outubro de 2015, no FEES 2015 em Belo Horizonte, houve a consolidação das competências com a participação dos presentes ao evento, com coordenação do Prof. Marcelo Yamaguti e Profa. Ingrid Nunes. Ainda

assim, houve uma das competências que não foi consolidada no evento e solicitou-se ao grupo de trabalho um parecer.

10. Em janeiro de 2016, houve a consolidação da última competência, realizada por meio de comunicações por e-mail.
11. Em abril de 2016, foi publicado no fórum Web o resultado dos refinamentos de competências do egresso de Engenharia de Software.
12. Finalmente em junho de 2016, foi encaminhado para a Comissão de Educação da SBC o refinamento de competências do egresso de Engenharia de Software.

#### **4. Considerações finais**

A partir da metodologia de trabalho descrita, obteve-se como resultado o modelo que consta no Apêndice 1. No Apêndice 2, listam-se os envolvidos no trabalho realizado durante os ciclos de discussão, inclusive com indicações dos coordenadores e participantes dos grupos de trabalho responsáveis por cada uma das competências não consolidadas no FEES 2014.

Dessa forma, este grande esforço da comunidade da Engenharia de Software do Brasil é documentado e divulgado, a fim de que sirva para ações de melhoria na educação de cursos de bacharelado em Engenharia de Software do Brasil.

A metodologia tem, entretanto, limitações. Considerando que as competências técnicas são profissionais, a aplicação da metodologia gera conteúdos que levam a estas competências. Entretanto, não gera conteúdos básicos (teoria da computação, matemática discreta, lógica, complexidade, etc.) que suportam os conteúdos profissionais.

#### **Referências**

ACM/IEEE. **Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering**. Technical Report. ACM, New York - NY, USA. 2004.

MEC. **Diretrizes Curriculares Nacionais para os cursos de graduação em Computação**. 2012. Disponível em: [http://portal.mec.gov.br/index.php?option=com\\_docman&view=download&alias=11205-pces136-11-pdf&category\\_slug=julho-2012-pdf&Itemid=30192](http://portal.mec.gov.br/index.php?option=com_docman&view=download&alias=11205-pces136-11-pdf&category_slug=julho-2012-pdf&Itemid=30192). Último acesso em: 8/8/2016.

## Apêndice 1: Refinamento de Competências

Competências	Refinamento/Conteúdo	Refinamento/Conteúdo	Refinamento/Conteúdo	Refinamento/Conteúdo	
<b>1. Investigar, compreender e estruturar as características de domínios de aplicação em diversos contextos</b>	Conhecer e analisar as características de domínios de aplicação em diversos contextos	Conteúdo: técnicas de elicitação de requisitos			
	Aplicar técnicas de estruturação das características de domínios de aplicação em diversos contextos	Conteúdos: técnicas de especificação, modelagem, verificação, validação e gerência de requisitos			
<b>2. Compreender e aplicar processos, técnicas e procedimentos de construção de software</b>	Conhecer os fundamentos da teoria de processos	Conteúdos: Teoria Geral de Processos (modelagem, especificação, análise e controle, adaptação)			
	Aplicar processos, técnicas e procedimentos de construção de software	Aplicar processos de construção de software	Conteúdos: Conceito de processo de software; Modelos de processo de software; representação de processo de software		
		Aplicar técnicas e procedimentos de especificação	Conteúdo: técnicas de elicitação de requisitos, técnicas de especificação, modelagem, verificação, validação e gerência de requisitos,		
		Aplicar técnicas e procedimentos de desenvolvimento	Conteúdos: princípios de projeto ( <i>design</i> ), projeto ( <i>design</i> ) de arquitetura de software, padrões, reutilização de software, projeto ( <i>design</i> ) detalhado, projeto ( <i>design</i> ) de dados, projeto ( <i>design</i> ) de interface com usuários e projeto ( <i>design</i> ) de interface com outros sistemas		
	Aplicar técnicas e procedimentos de validação, verificação e teste	Aplicar técnicas e procedimentos de validação e verificação estáticos	Conteúdos: técnicas de revisão e análise estática de artefatos de software		
		Aplicar técnicas e procedimentos de validação e verificação dinâmicos	Conteúdos: técnicas de análise dinâmica de artefatos de software		
	Aplicar técnicas e procedimentos de evolução	Conteúdos: refatoração, engenharia reversa, reengenharia, análise de impacto, manutenção, depuração			
<b>3. Analisar e selecionar tecnologias adequadas para a construção de software</b>	Aplicar tecnologias para a definição do ambiente de construção de software	Conteúdos: ferramentas e <i>frameworks</i> de desenvolvimento e de gerenciamento de configuração de software			

	Aplicar tecnologias a serem utilizadas no produto de software	Conteúdos: técnicas de programação; paradigmas de programação.	
<b>4. Conhecer os direitos e propriedades intelectuais inerentes à produção e utilização de software</b>	Conhecer os direitos, deveres e propriedades intelectuais inerentes à produção de software	Conteúdos: noções básicas de Direito, direito autoral, registro de software, propriedade intelectual, leis, acórdãos e instruções normativas sobre Engenharia de Software	
	Conhecer os direitos, deveres e propriedades intelectuais inerentes à utilização de software	Conteúdos: noções básicas de Direito, direito autoral, registro de software, propriedade intelectual, leis, acórdãos e instruções normativas sobre Engenharia de Software	
<b>5. Avaliar a qualidade de sistemas de software</b>	Entender quais são os atributos de qualidade do produto de software e sua utilidade	Conteúdo: atributos de qualidade de produto de software	
	Aplicar mecanismos de medição da qualidade do produto de software	Conteúdos: métricas de produto de software, técnicas de avaliação de produto	
	Aplicar técnicas e procedimentos de validação, verificação e teste	Aplicar técnicas e procedimentos de validação e verificação estáticos	Conteúdos: técnicas de revisão e análise estática de artefatos de software
		Aplicar técnicas e procedimentos de validação e verificação dinâmicos	Conteúdos: técnicas de análise dinâmica de artefatos de software
<b>6. Integrar sistemas de software</b>	Aplicar técnicas de integração de partes de um sistema	Conteúdos: ambientes de integração, ferramentas de <i>build</i>	
	Aplicar técnicas de integração de sistemas heterogêneos	Conteúdos: interoperabilidade de sistemas, <i>wrappers</i> , software como serviço, sistemas de sistemas, ecossistemas/plataformas (APIs)	
<b>7. Gerenciar projetos de software e processos de desenvolvimento de software</b>	Aplicar técnicas, ferramentas e práticas de gerenciamento considerando as dimensões de gestão de projetos de software	Conteúdo: conceitos básicos de gestão de projetos. Alinhamento da TI com o negócio, formas de gestão, gerenciamento de escopo, tempo, custo, qualidade, comunicação, riscos, pessoas, aquisição, integração, partes interessadas e valor de negócio, métricas de produto e de projeto.	

	Aplicar técnicas, ferramentas e práticas para gerenciamento do processo da produção, aquisição e evolução de um software	Conteúdo: gerenciamento do ciclo de vida de produção; gerenciamento do fluxo de trabalho; engenharia de produto, modelos de ciclo de vida: história e perspectivas, artefatos de software, papéis, métricas de processo de software	
	Conhecer Teorias aplicadas na Gestão da Produção de Software	Conteúdo: Teorias - da Administração Científica, das Restrições, da Complexidade, dos Jogos, do Caos, da Dinâmica de Sistemas, Geral dos Sistemas, Pensamento Sistêmico	
	Entender os elementos que compõe um sistema de produção	Entender as estratégias de Operações	Conteúdo: Cadeia de Valor, Tomada de Decisão, Alinhamento entre a estratégia de TI e estratégia de negócios
		Entender a estrutura dos processos de produção	Conteúdo: competências competitivas, estrutura do processo de bens (manufatura) e serviços (produtos de software)
<b>8. Aplicar adequadamente normas técnicas</b>	Entender as normas de qualidade de produto de software	Conteúdo: modelos e normas de qualidade de produto (nacionais e internacionais)	
	Entender as normas de qualidade do processo de desenvolvimento de software	Conteúdo: modelos e normas de qualidade de processo (nacionais e internacionais)	
	Aplicar conceitos de qualidade de processo para a definição de um processo de software	Conteúdo: modelos e normas de qualidade de processo (nacionais e internacionais), métricas de processo	
<b>9. Qualificar e quantificar seu trabalho baseado em experiências e estudos experimentais</b>	Aplicar métodos de pesquisa experimental	Conteúdo: conhecimento científico; método científico e experimental; métodos quantitativos, qualitativos e mistos de pesquisa; métodos de pesquisa e experimentação em Engenharia de Software; estudos primários e secundários; protocolos de pesquisa	
	Entender procedimentos de análise, interpretação e apresentação de resultados de estudos experimentais em ES	Conteúdo: estatísticas descritivas, teste de hipóteses, análise qualitativa, relato de estudos experimentais de Engenharia de Software	

<b>10. Exercer múltiplas atividades relacionadas a software como: desenvolvimento, evolução, consultoria, negociação, ensino e pesquisa</b>	Aplicar os conhecimentos adquiridos para o desenvolvimento e evolução de software	Conteúdos: práticas de laboratório no desenvolvimento e evolução de software
	Aplicar técnicas de comunicação para apresentar conhecimentos adquiridos	Conteúdos: técnicas de comunicação
	Aplicar métodos de ensino e pesquisa em ES	Conteúdos: metodologias de ensino e pesquisa
	Conhecer metodologias do trabalho de consultoria.	Conteúdos: técnicas de consultorias
	Conhecer os principais modelos e etapas do processo de negociação.	Conteúdos: técnicas de negociação
<b>11. Conceber, aplicar e validar princípios, padrões e boas práticas no desenvolvimento de software</b>	Aplicar os princípios, padrões e boas práticas de desenvolvimento de software	Conteúdos: princípios de ES, aplicação de padrões em ES, melhoria contínua, aplicação de gestão de conhecimento
	Conceber e validar os princípios, padrões e boas práticas de desenvolvimento de software	Conteúdo: conhecimento científico; método científico e experimental; métodos quantitativos, qualitativos e mistos de pesquisa; métodos de pesquisa e experimentação em ES; estudos primários e secundários; protocolos de pesquisa, estatísticas descritivas, teste de hipóteses, análise qualitativa, relato de estudos experimentais de ES, princípios de ES, aplicação de padrões em ES, melhoria contínua, aplicação de gestão de conhecimento.
<b>12. Analisar e criar modelos relacionados ao desenvolvimento de software</b>	Aplicar e selecionar técnicas de modelagem de software	Conteúdos: modelos estáticos, modelos funcionais, modelos dinâmicos, modelos formais.
	Aplicar técnicas de análise de modelos de software	Conteúdos: técnicas de análise de correção, de completitude, de consistência interna e entre modelos, de rastreabilidade entre modelos, de redundância, de ambiguidade.



<b>13. Identificar novas oportunidades de negócios e desenvolver soluções inovadoras.</b>	Entender conceito de empreendedorismo como processo de criar algo novo e com valor de negócio.	Conteúdos: empreendedorismo, análise e modelos de negócio; <i>frameworks</i> para construção de modelos de negócio.
	Entender o conceito de inovação como processo que transforma uma ideia em produto ou serviço.	Conteúdos: modelos de análise do processo de inovação, estratégias de inovação, planejamento e implementação de inovações, indicadores de inovação, políticas públicas e marco regulatório da inovação, inovação e desenvolvimento sustentável. Diferenciação entre inovação e pesquisa. Ecossistemas de inovação.
<b>14. Identificar e analisar problemas avaliando as necessidades dos clientes, especificar os requisitos de software, projetar, desenvolver, implementar, verificar e documentar soluções de software baseadas no conhecimento apropriado de teorias, modelos e técnicas</b>	Aplicar técnicas para identificar e analisar problemas avaliando as necessidades dos clientes	Conteúdo: técnicas de elicitação de requisitos
	Aplicar técnicas de especificação de requisitos de software	Conteúdos: técnicas de especificação, modelagem, verificação, validação e gerência de requisitos

	<p>Aplicar teorias, modelos e técnicas para projetar, desenvolver, implementar e documentar soluções de software</p>	<p>Conteúdos: técnicas de revisão e análise estática de artefatos de software, técnicas de análise dinâmica de artefatos de software, refatoração, engenharia reversa, reengenharia, análise de impacto, manutenção, depuração, conceito de processo de software; modelos de processo de software; representação de processo de software, princípios de projeto (<i>design</i>), projeto (<i>design</i>) de arquitetura de software, padrões, reutilização de software, projeto (<i>design</i>) detalhado, projeto (<i>design</i>) de dados, projeto (<i>design</i>) de interface com usuários e projeto (<i>design</i>) de interface com outros sistemas, princípios de ES, aplicação de padrões em ES, melhoria contínua, aplicação de gestão de conhecimento</p>
	<p>Aplicar teorias, modelos e técnicas para verificar soluções de software</p>	<p>Conteúdos: técnicas de revisão e análise estática de artefatos de software, técnicas de análise dinâmica de artefatos de software</p>

## **Apêndice 2: Colaboradores (em ordem alfabética)**

- Abraham Lincoln Rabelo de Sousa (UNILASALLE)
- Adriana Pereira de Medeiros (UFF)
- Alessandra C. Smolenaars Dutra (PUCRS) \*,\*\*
- Alessandro Garcia (PUCRio) \*
- Alexandre Cidral (UNIVILLE)
- Alfredo Goldman (IME) \*
- Altigran da Silva (UFAM) \*
- Amon José Aidukaitis (PETROBRÁS)
- Ana Paula Chaves Steinmacher (UTFPR) \*\*
- Ana Paula Terra Bacelo (PUCRS) \*
- Ana Regina Rocha (UFRJ) \*
- André Villasboas (CPQD) \*
- Arilo Dias Neto (UFAM) \*
- Augusto Sampaio (UFPE) \*
- Auri Vicenzi (UFG) \*
- Avelino Francisco Zorzo (PUCRS) \*
- Christina Chaves (UFBA) \*
- Cláudia Werner (UFRJ) \*
- Daltro José Nunes (UFRGS) \*
- Edmundo Sérgio Spoto (UFG) \*
- Elaine Venson (UNB)
- Ellen Francine (USP – São Carlos) \*
- Evandro Franzen (UNIVATES)
- Fabio Gomes Rocha (UNIT) \*\*
- Fábio Kon (USP-IME) \*
- Fábio Lucena (UFG) \*
- Fabricio Souza Pinto (UESB)
- Flávio Wagner (UFRGS) \*
- Francisco José Mônaco (ICMC-USP) \*
- Gleison dos Santos Souza (UNIRIO) \*
- Guilherme Travassos (UFRJ) \*

- Heitor Costa (UFLA) \*
- Hilmer Rodrigues Neri (UNB) \*\*
- Igor Fabio Steinmacher (UTFPR) \*
- Igor Wiese (UTFPR) \*
- Ingrid Oliveira de Nunes (UFRGS) \*,\*\*
- Itana Gimenes (UEM) \*
- Jair Leite (UFRN)
- Jean Felipe Cheiran (UNIPAMPA)
- João Pablo Silva da Silva (UNIPAMPA)
- José Carlos Maldonado (USP-ICMC) \*
- José Reginaldo Carvalho (UFAM) \*
- Julio Cesar Sampaio do Prado Leite (PUC-Rio) \*
- Leila Ribeiro (UFRGS)
- Manoel Mendonça (UFBA) \*
- Marcelo Hideki Yamaguti (PUCRS) \*,\*\*
- Marcelo Quinta (UFG) \*
- Marcelo Werneck (PUC Minas) \*
- Marcia Lucena (UFRN)
- Márcio Delamaro (ICMC-USP) \*
- Marcos Kalinowski (UFF) \*
- Maria Augusta Vieira Nelson (PUC Minas) \*
- Maria Claudia Figueiredo Pereira Emer (UTFPR) \*
- Maurício Aniche (USP) \*
- Michael Móra (PUCRS) \*
- Newton Braga Rosa (UFRGS) \*
- Paulo Borba (UFPE) \*
- Paulo Cesar Masiero (USP – ICMC) \*
- Paulo Meirelles (UNB) \*\*
- Rafael Prikladnicki (PUCRS) \*
- Raul Wazlawick (UFSC) \*
- Rejane Figueiredo (UnB)
- Ricardo A. C. de Souza (UFRPE)

- Ricardo Anido (UNICAMP) \*
- Rosana Vacari (ICMC-USP) \*
- Sabrina Marczak (PUCRS) \*
- Sandra Fabbri (UFSCAR) \*
- Seiji Isotani (USP) \*
- Sílvio Meira (UFPE) \*
- Tayana Conte (UFAM) \*,\*\*
- Thais Batista (UFRN) \*
- Thelma Elita Colanzi (UEM) \*\*
- Uirá Kulesza (UFRN) \*
- Vander Alves (UnB) \*

Obs.:

\* Membro de grupo de trabalho (convidado a partir de indicação no FEES 2014, ou por indicação de membros iniciais do grupo)

\*\* Coordenador de grupo de trabalho (definido no FEES 2014)