UNIVERSIDADE FEDERAL DE SERGIPE

DEPARTAMENTO DE SISTEMAS DE INFORMAÇÃO

BACHARELADO EM SISTEMAS DE INFORMAÇÃO

RELATÓRIO DO ESTÁGIO

SUPERVISIONADO

ISABELLA MENEZES SANTOS

ITABAIANA – SE

JUNHO - 2011

RELATÓRIO DO ESTÁGIO

SUPERVISIONADO

ISABELLA MENEZES SANTOS

Relatório de estágio apresentado ao Curso de Sistemas de Informação como parte das exigências da disciplina Estágio Supervisionado, sob a orientação pedagógica do Prof. Marcos Barbosa Dósea e orientação técnica da Gerente de Processos e Qualidade da INFOX Tecnologia da Informação Elenildes de Almeida Santos Costa.

ITABAIANA – SE

JUNHO – 2011

UNIVERSIDADE FEDERAL DE SERGIPE

DEPARTAMENTO DE SISTEMAS DE INFORMAÇÃO

BACHARELADO EM SISTEMAS DE INFORMAÇÃO

RELATÓRIO DO ESTÁGIO

SUPERVISIONADO

ISABELLA MENEZES SANTOS

Aprovado em: \_\_/\_\_/\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Me. Prof. Marcos Barbosa Dósea

Orientador Pedagógico

RESUMO

Com o aumento da concorrência de empresas que trabalham desenvolvendo software, a busca pela qualidade dos produtos e serviços vem se tornando cada vez mais um diferencial para a organização que pretende oferecer as melhores soluções, visando à satisfação de seus clientes e o aumento da sua participação no mercado, para competir com empresas já consolidadas no setor da tecnologia da informação.

Usualmente os problemas mais freqüentes nessas empresas são: fechamento do escopo do software, a falta de validação e interação do cliente nas definições do software, prazos e custos irreais e falta de qualidade do produto desenvolvido.

Diante desse contexto, a definição de um processo de desenvolvimento torna-se cada vez mais importante para a obtenção dos resultados esperados para o projeto e a qualidade do produto final, uma vez que, observa-se que a qualidade do produto está fortemente ligada ao seu processo de desenvolvimento. Em virtude das mudanças que podem ocorrer nas empresas com relação às pessoas, tecnologias e processo, é essencial a definição de um processo padrão, que seja gerenciado e melhorado, sendo esse, de propriedade da organização.

A mesma situação acontece com INFOX Tecnologia da Informação Ltda, é uma empresa especializada em desenvolvimento e integração de sistemas de informação em ambiente de fábrica de software e na prestação de serviços de suporte, fundada em 1986, com sede na cidade de Aracaju, é considerada uma empresa de médio porte, com setores e papéis definidos, e recursos disponíveis. A estrutura da empresa atualmente é composta dos setores administrativo, de negócios, suporte, gerência de projetos e fábrica de software.

Buscando a melhoria contínua nos seus processos, a empresa é certificada nível G do MPSbr, e ISO 9001, portanto, o atual processo requer uma revisão para identificar possíveis problemas .

Esse relatório tem como objetivo relatar o processo padrão da empresa, como também melhorias no mesmo, através do uso adequado de ferramentas, e análise da situação atual da mesma.

**Palavras-Chave:** Qualidade; Processos, MPS.br, PPDSI.

SUMÁRIO

[1. INTRODUÇÃO 6](#_Toc295161219)

[1.1. Local do estágio 6](#_Toc295161220)

[1.2. Objetivos do estágio 6](#_Toc295161221)

[1.3. Estrutura do relatório 6](#_Toc295161222)

[2. REFERENCIAL TEÓRICO 7](#_Toc295161223)

[2.1. Qualidade de Software 9](#_Toc295161226)

[2.2. Modelo Cascata. 11](#_Toc295161229)

2.3. MPSbr ......................................................................................................................14

[3. FERRAMENTAS UTILIZADAS 15](#_Toc295161231)

[3.1. EPFComposer 15](#_Toc295161232)

[3.2. SGP 15](#_Toc295161233)

[3.3. Redmine 15](#_Toc295161234)

[4. ATIVIDADES DO ESTÁGIO 16](#_Toc295161238)

[4.1. Modelando PPDSI no EPFComposer 16](#_Toc295161239)

[4.2. Avaliando Melhorias no PPDSI 16](#_Toc295161240)

[4.3. Realizando a Rastreabilidade Bidirecional 16](#_Toc295161241)

[5. CONCLUSÃO 17](#_Toc295161242)

[6. REFERÊNCIAS BIBLIOGRÁFICAS 18](#_Toc295161243)

1. **INTRODUÇÃO**

Este relatório apresenta a vivência do estagiário em um ambiente de produção real onde será conciliado, de forma prática, o conhecimento aprendido durante o curso.

* 1. **Local do estágio**

A realização do estágio foi concebida na INFOX Tecnologia da Informação, no setor de qualidade de software, desempenhando a função de Analista de Qualidade. Utilizando a disponibilidade de 20 horas semanais durante o período de 11 de março de 2011 a 11 de Agosto de 2011.

* 1. **Objetivos do estágio**

O objetivo geral do estágio é a inspeção e melhoria no Processo Padrão de Desenvolvimento de Software Infox, acrônimo PPDSI, seguindo as práticas, processos e procedimentos da empresa.

Objetivos específicos do estágio:

* Auxiliar na implantação de processos de qualidade e desenvolvimento;
* Alimentar as informações nas ferramentas de apoio à gestão da qualidade;
* Auxiliar a gerência da qualidade nas realizações das reuniões e atividades da área;
* Conscientizar os colaboradores da utilização dos procedimentos e registros da qualidade;
* Relatar à gerência sobre as informações ligadas a qualidade e qualquer necessidade de melhoria;
* Inspecionar, analisar e dar conhecimento da qualidade interna de artefatos;
* Realizar pesquisa sobre novas tendências de modelos de qualidade e ferramentas;
* Ajudar no mapeamento dos processos organizacionais e software;
* Auxiliar a gestão de qualidade na realização de treinamentos e disseminação das informações pertinentes ao setor;
* Realizar revisões de todas as atividades de software nos projetos.
  1. **Estrutura do relatório**

Esse relatório está dividido da seguinte forma: na Seção 2 é abordado o conhecimento teórico necessário para a atuação do estagiário, na Seção 3 são apresentadas as ferramentas necessárias para a execução do trabalho a ser realizado, a seguir, na Seção 4, são listadas as atividades realizadas, e por fim, na Seção 5, uma breve conclusão sobre o estágio realizado.

1. **REFERENCIAL TEÓRICO**

Para a realização do trabalho foi necessário o estudo de processos de desenvolvimento de software, bem como o guia de melhoria de processo para o software brasileiro, denominado MPS.br. Para o desenvolvimento de software, a INFOX possui de um processo padrão denominado PPDSI, definido baseado no modelo Cascata com iterações, e alinhado ao resultados esperados do MPS.br.

* 1. **Qualidade de Software**

Nos dias atuais a realidade das empresas que trabalham desenvolvendo software é a busca pela qualidade dos seus produtos e serviços, visando à satisfação de seus clientes em um mercado cada vez mais competitivo e seletivo. Essa exigência de mercado tem se tornado uma condição básica para a continuidade da empresa. Diante desse cenário, a melhoria de processos de software tem mostrado, na prática, ser uma abordagem viável, eficaz e eficiente para a necessária melhoria das organizações [1].

A definição de um processo de desenvolvimento de software se torna cada vez mais importante para a obtenção dos resultados esperados para o projeto e a qualidade do produto final, foi observado que a qualidade do produto está fortemente ligada ao seu processo de desenvolvimento. Com as possíveis mudanças que podem ocorrer nas empresas relacionadas às pessoas, tecnologias e processo, foi verificada a importância da definição de um processo que seja gerenciado e melhorado, sendo esse de propriedade da organização.

A prática de definição e melhoria de processos de software está sendo adotada por diversas empresas que buscam organizar e controlar os processos dentro do ciclo de vida do software. A melhoria de software consiste na realização de ações que alterem os processos, para que estes satisfaçam, de forma mais eficiente, os objetivos e necessidades das organizações, com a finalidade de reduzir custos, prazos e problemas relacionados a escopo, buscando aumentar sua produtividade, melhorar a satisfação do cliente, estabelecer uma linguagem comum entre toda a equipe, construindo uma visão compartilhada e ao mesmo tempo uma estrutura para priorizar ações e projetos.

De acordo com pesquisas do Standish Group são apontadas como as principais causas de problemas em projetos a: falta de especificação do usuário (12,8%), requisitos incompletos (12,3%), mudança de requisitos (11, 8%), falta de apoio executivo (7,5%), tecnologia imatura (7%), falta de recursos (6,4%), expectativas irreais (5,9%), objetivos obscuros (5,3%), tempo irreais (4,3%), novas tecnologias (3,7%) e outros (23%) [2].

Desta forma, entende-se por qualidade de software, a área de conhecimento da engenharia de software que objetiva garantir a qualidade do produto desenvolvido através da definição e normatização de processos de desenvolvimento. Apesar dos modelos aplicados na garantia da qualidade de software atuarem principalmente no processo, o principal objetivo é garantir um produto final que satisfaça às expectativas do cliente, dentro daquilo que foi acordado inicialmente.

Segundo a atual norma brasileira (NBR ISO 8402), qualidade é: A totalidade das características de uma entidade que lhe confere a capacidade de satisfazer às necessidades explícitas e implícitas.[3]

**2.2 Modelo Cascata**

O modelo Cascata é um modelo de engenharia projetado para ser aplicado no desenvolvimento do software. A idéia principal que o dirige é que as diferentes etapas de desenvolvimento seguem uma seqüência: a saída da primeira etapa “fluí” para a segunda etapa e a saída da segunda etapa “fluí” para a terceira e assim por diante. As atividades a executar são agrupadas em tarefas, executadas seqüencialmente, de forma que uma tarefa só poderá ter início quando a anterior tiver terminado.

O modelo em cascata tem a vantagem que só avança para a tarefa seguinte quando o cliente valida e aceita os produtos finais da tarefa atual. O modelo pressupõe que o cliente participa ativamente no projeto e que sabe muito bem o que quer. Este modelo minimiza o impacto da compreensão adquirida no decurso de um projeto, uma vez que se um processo não pode voltar atrás de modo a alterar os modelos e as conclusões das tarefas anteriores, é normal que as novas idéias sobre o sistema não sejam aproveitadas. Numa tentativa de resolver este tipo de problema foi definido um novo tipo de processo baseado no clássico em cascata, designado por modelo em cascata revisto, cuja principal diferença consiste em prever a possibilidade de a partir de qualquer tarefa do ciclo se poder regressar a uma tarefa anterior de forma a contemplar alterações funcionais e/ou técnicas que entretanto tenham surgido, em virtude de um maior conhecimento que entretanto se tenha obtido. O risco desta abordagem é que, na ausência de um processo de gestão do projeto e de controlo das alterações bem definido, podemos passar o tempo num ciclo infinito, sem nunca se atingir o objetivo final, ou seja disponibilizar o sistema a funcionar.



**Descrição do modelo**

**- Definição dos requisitos**

Nesta etapa, estabelecem-se os requisitos do produto que se deseja desenvolver, o que consiste usualmente nos serviços que se devem fornecer limitações e objetivos do software. Sendo isso estabelecido, os requisitos devem ser definidos de uma maneira apropriada para que sejam úteis na etapa seguinte. Esta etapa inclui também a documentação e o estudo da facilidade e da viabilidade do projeto com o fim de determinar o processo de início de desenvolvimento do projeto do sistema; pode ser vista como uma concepção de um produto de software e também como o início do seu ciclo de vida.

**- Projeto do sistema**

O projeto do sistema é um processo de vários passos que se centraliza em quatro atributos diferentes do sistema: estrutura de dados, arquitetura do software, detalhes procedais e caracterização das interfaces. O processo de projeto representa os requisitos de uma forma que permita a codificação do produto (é uma prévia etapa de codificação). Da mesma maneira que a análise dos requisitos, o projeto é documentado e transforma-se em uma parte do software.

**- Implementação**

Esta é a etapa em que são criados os programas. Se o projeto possui um nível de detalhe elevado, a etapa de codificação pode implementar-se automaticamente. A princípio, sugere-se incluir um teste unitário dos módulos nesta etapa; nesse caso, as unidades de código produzidas são testadas individualmente antes de passar a etapa de integração e teste global.

**- Teste do sistema**

Concluída a codificação, começa a fase de teste do sistema. O processo de teste centraliza-se em dois pontos principais: as lógicas internas do software e as funcionalidades externas. Esta fase decide se foram solucionados erros de “comportamento” do software e assegura que as entradas definidas produzam resultados reais que coincidam com os requisitos especificados.

**- Manutenção**

Essa etapa consiste na correção de erros que não foram previamente detectados, em melhorias funcionais e de preferência e outros tipos de suporte. A etapa de manutenção à parte do ciclo de vida do produto de software e não pertence estritamente ao seu desenvolvimento.

Melhorias e correções podem ser consideradas como parte do desenvolvimento.

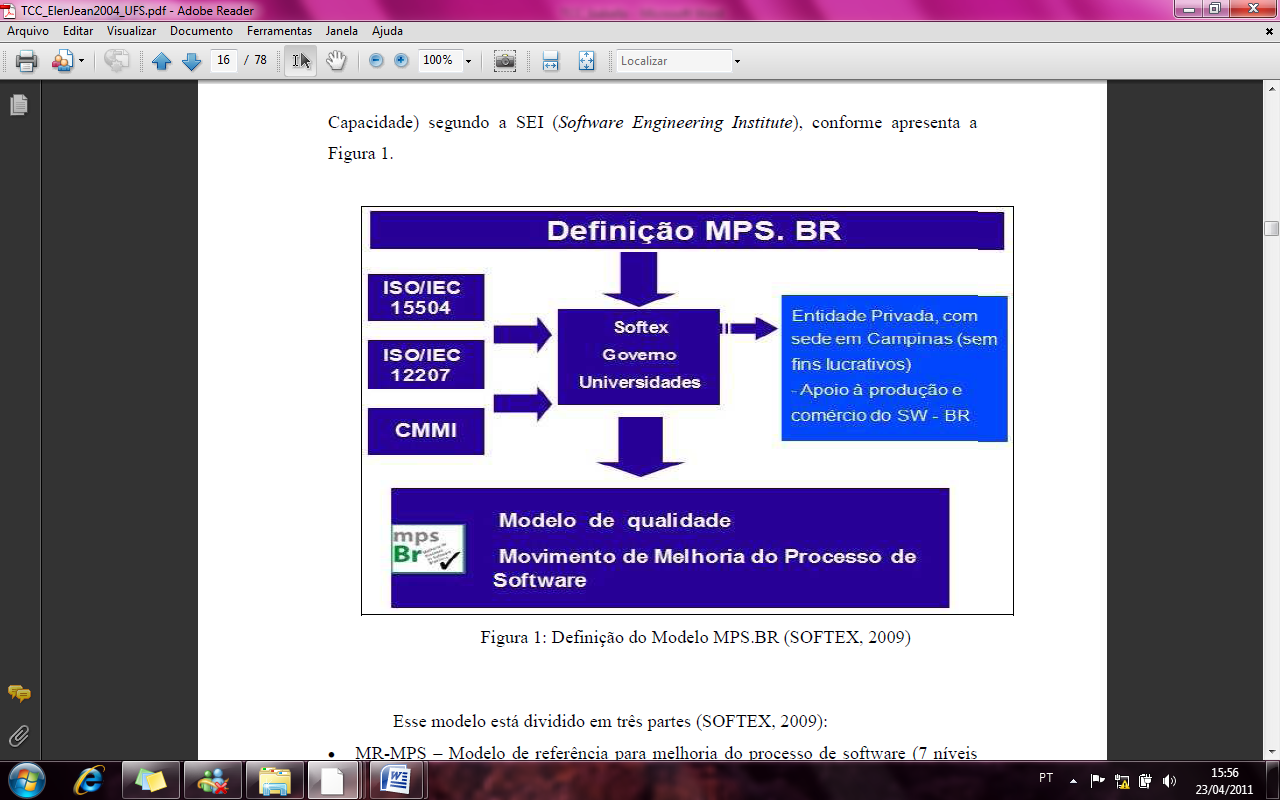
As etapas descritas são as principais, porém existem sub-etapas dentro de cada etapa, as quais diferem muito de um projeto para outro. Também é possível que certos projeto de software exijam a incorporação de uma etapa extra ou a separação de uma etapa em outras etapas.

Com certeza, todas essas variações do modelo Cascata possuem o mesmo conceito básico: a idéia de que uma etapa fornece saída que serão usadas como entradas para a etapa seguinte. Portanto, o processo de desenvolvimento de um produto de software de acordo com o modelo Cascata é simples de conhecer e controlar.

Outras atividades que também são levadas em consideração em cada uma das etapas de desenvolvimento do software: a documentação, a verificação e a administração das etapas serem documentos. A verificação, por sua vez, é necessária para que uma etapa forneça os dados corretos para a etapa seguinte. Já a administração, efetua a gestão e o controle da etapa.

**2.3.** **MPS.BR**

O MPS.br é um programa mobilizador, de longo prazo, criado em dezembro de 2003, coordenado pela SOFTEX, que conta com apoio do Ministério da Ciência e Tecnologia (MCT), Financiadora de Estudos e Projetos (FINEP), Serviço de Apoio às Micro e Pequenas Empresas (SEBRAE) e Banco Interamericano de Desenvolvimento (BID), e tem como objetivo a Melhoria de Processo do Software Brasileiro (MPS.br) (SOFTEX, 2009a, p. 4).

****

**Figura 1: Definição do Modelo MPS.BR (SOFTEX, 2009)**

Esse modelo está dividido em três partes (SOFTEX, 2009):

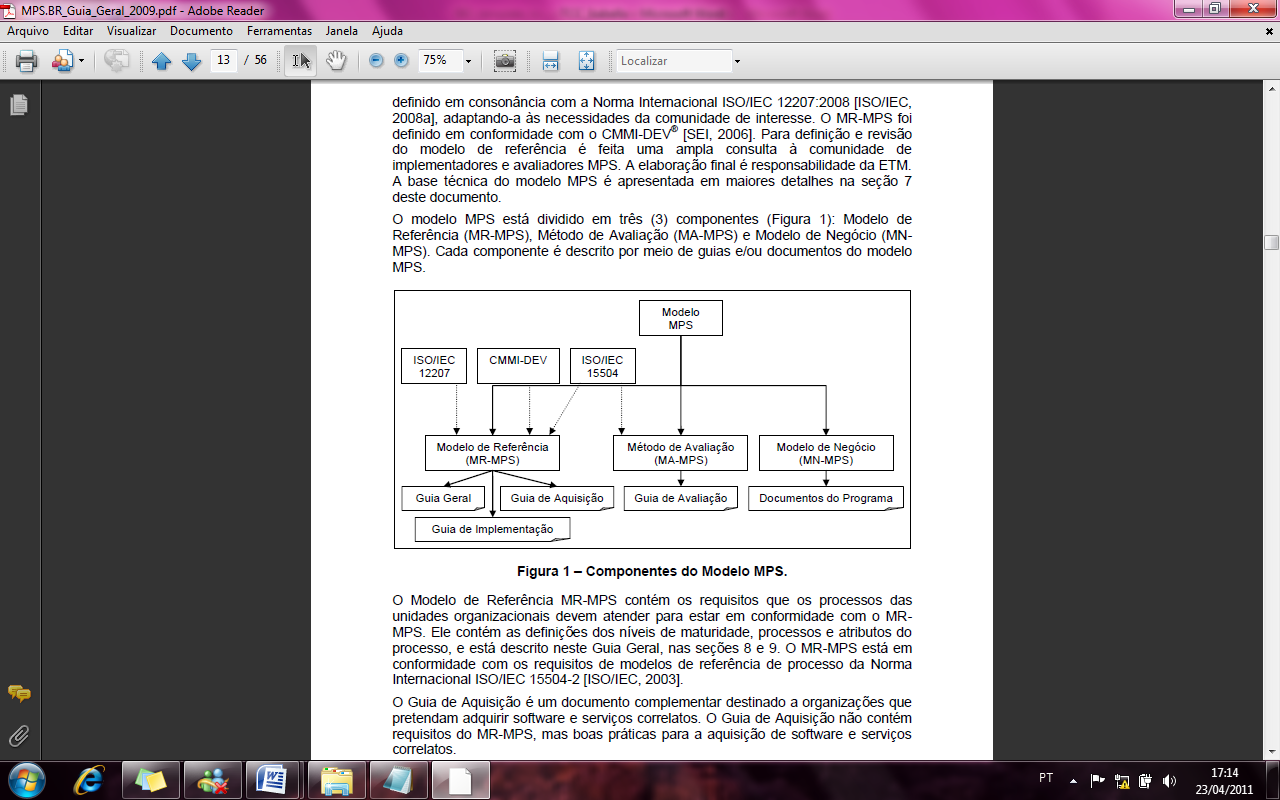
· MR-MPS – Modelo de referência para melhoria do processo de software (7 níveis de maturidade e suas áreas de processo);

· MA-MPS – Método de avaliação para melhoria do processo de software (Guia de Avaliação 2009), descreve os requisitos para os avaliadores líderes, avaliadores adjuntos e Instituições Avaliadoras (IA);

· MN-MPS – Modelo de negócio para melhoria do processo de software (discorre sobre o protocolo para a implementação do MR-MPS pelas Instituições Implementadoras (II), a avaliação seguindo o MA-MPS pelas instituições avaliadoras (IA), organização de grupos de empresas para implementação do MRMPS e avaliação MA-MPS pelas Instituições Organizadoras de Grupos de 17 Empresas (IOGE), certificação de consultores de aquisição e programas anuais de treinamento por meio de cursos e *workshops* MPS.BR).

**2.1.1- Estrutura do MPS.BR**

Os três componentes pertencente ao modelo MPS e seus desdobramentos estão ilustrados na Figura 2.



**Figura 2 – Componentes do Modelo MPS** **(Guia Geral MPS.BR, 2009)**

O Modelo de Referência MR-MPS contém os requisitos que os processos das unidades organizacionais devem atender para estar em conformidade com o modelo, nele se encontra o Guia Geral, que contém as definições dos níveis de maturidade, processos e atributos dos processos, o Guia de Aquisição, que é um documento complementar destinado a organizações que desejam adquirir software e serviços correlatos, procurando dar uma orientação quando da aquisição desses produtos e/ou serviços e o Guia de Implementação que é composto de sete partes que descrevem como implementar cada um dos níveis do MR-MPS, detalhando os processos contemplados nos respectivos níveis de maturidade e os resultados esperados com a implementação dos processos.

O método de avaliação encontra-se o Guia de Avaliação, que contém o processo e o método de avaliação MA-MPS, os requisitos para os avaliadores líderes, avaliadores adjuntos e Instituições Avaliadoras (IA), métodos e formulários.

O modelo de negócio descreve regras de negócio para implementação do MR-MPS pelas Instituições Implementadoras (II), e avaliação seguindo o MA-MPS pelas Instituições Avaliadoras (IA).

**2.1.2- Níveis de Maturidade**

O MR-MPS define sete níveis de maturidade que são: A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Parcialmente Gerenciado), como demonstrado na figura 3.



**Figura 3: Níveis de Maturidade do MPS.BR (SOFTEX, 2009c)**

O nível de maturidade é a combinação entre processos e sua capacidade, este corresponde ao grau de melhoria de processo para um pré-determinado conjunto de processos no qual todos os objetivos dentro do conjunto devem ser atendidos. Para cada um destes sete níveis de maturidade é atribuído um perfil de processos que indicam onde a organização deve colocar o grau de melhoria. O progresso e o alcance de um determinado nível de maturidade do MR-MPS se obtêm quando são atendidos os propósitos e todos os resultados esperados dos respectivos processos e os resultados esperados dos atributos de processo estabelecidos para aquele nível.

No entanto nota-se que existe uma correlação entre os níveis de maturidade do MPS.BR com o CMMI (Modelo Integrado de Maturidade e Capacidade) que é o modelo de qualidade internacional, onde os níveis G e F correspondem ao nível 2 do CMMI, os níveis E, D e C correspondem ao nível 3, o B ao nível 4 e o A ao nível 5.

A divisão em sete estágios tem o objetivo de se adequar à realidade das empresas brasileiras, mais especi*alm*ente as micros, pequena e médias empresas. Segundo a SOFTEX (2009a, p. 16), isto possibilita se realizar avaliações com resultados de melhoria de processos em prazos mais curtos.

**2.1.3- Processos**

Para cada nível de maturidade existe um perfil de processos com requisitos para a organização concentrar esforços para melhorias (SOFTEX, 2009a). Estes processos estão descritos na seguinte forma:

* Propósito – Descreve o objetivo geral a ser atingido na execução do processo.
* Resultados esperados – Estabelece os resultados a serem obtidos com a efetiva implementação do processo.

Segundo a SOFTEX (2009a, p. 23), alguns processos presentes no MR-MPS podem ser excluídos total ou parcialmente, do escopo de uma avaliação por não serem pertinentes ao negócio da organização que está sendo avaliada. Por exemplo, em organizações do tipo Fábrica de Software é permitida na avaliação a exclusão dos processos referentes à Aquisição.

À medida que a organização avança na escala de níveis do MPS.br, ela tem um maior grau de refinamento na execução de seus processos. Este critério é denominado Capacidade do Processo, e é representado por um conjunto de atributos de processos (AP) e seus espectivos resultados esperados (RAP). Os atributos de processo são requeridos de acordo com o nível de maturidade correspondente em conformidade com seus resultados esperados, e são acumulativos, ou seja, para obter a certificação do nível “E” por exemplo, a organização deve possuir todos os atributos para os níveis “G” e “F”. No nível G do MPS.br são considerados os atributos de processo AP 1.1 – O Processo é executado e AP 2.1 – O processo é gerenciado. Segundo a SOFTEX (2009b, p. 32), em seu MA-MPS, para se considerar um processo satisfeito no nível G o atributo AP 1.1 é necessário estar caracterizado como T (Totalmente implementado) e o atributo AP 2.1 T (Totalmente implementado) ou L (Largamente implementado). Estes níveis de Capacidade do Processo são descritos a seguir (SOFTEX, 2009a, p. 17-18):

 **AP 1.1 O processo é executado**

o Este atributo é uma medida do quanto o processo atinge o seu propósito.

o Resultado esperado:

 **RAP 1.** O processo atinge seus resultados definidos.

 **AP 2.1 O processo é gerenciado**

o Este atributo é uma medida do quanto à execução do processo é gerenciada.

o Resultados esperados:

**RAP 2.** Existe uma política organizacional estabelecida e mantida para o processo;

 **RAP 3.** A execução do processo é planejada;

 **RAP 4.** (Para o nível G). A execução do processo é monitorada e ajustes são realizados;

 **RAP 4.** (A partir do nível F). Medidas são planejadas e coletadas para monitoração da execução do processo e ajustes são realizados;

 **RAP 5.** (Até o nível F) As informações e os recursos necessários para a execução do processo são identificados e disponibilizados;

 **RAP 5.** (A partir do nível E) Os recursos e informações necessários para executar o processo definido são disponibilizados, alocados e utilizados;

 **RAP 6.** (Até o nível F) As responsabilidades e a autoridade para executar o processo são definidas, atribuídas e comunicadas;

 **RAP 6.** (A partir do nível E) Os papéis requeridos, responsabilidades e autoridade para execução do processo definido são atribuídos e comunicados;

 **RAP 7.** (Até o nível F) As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência;

 **RAP 7.** (A partir do nível E) As pessoas que executam o processo definido são competentes em termos de formação, treinamento e experiência;

 **RAP 8.** A comunicação entre as partes interessadas no processo é gerenciada de forma a garantir o seu envolvimento;

 **RAP 9**. (Até o nível F) Os resultados do processo são revistos com a gerência de alto nível para fornecer visibilidade sobre a sua situação na organização;

 **RAP 9.** (A partir do nível E) Métodos adequados para monitorar a eficácia e adequação do processo são determinados e os resultados do processo são revistos com a gerência de alto nível para fornecer visibilidade sobre a sua situação na organização;

 **RAP 10.** (Para o nível G) O processo planejado para o projeto é executado.

 **RAP 10.** (A partir do nível F) A aderência dos processos executados às descrições de processo, padrões e procedimentos é avaliada objetivamente e são tratadas as não conformidades.

1. **FERRAMENTAS UTILIZADAS**

Para a realização do estágio foi necessária à utilização de ferramentas de terceiros como Eclipse Process Framework Composer, acrônimo EPFComposer, e Redmine, e de uma ferramenta proprietária da INFOX que é o SGP – Sistema Gerenciador de Processos. A seguir serão mostradas algumas características de cada uma delas.

* 1. **Eclipse Process Framework Composer**

O Eclipse Process Framework (EPF), é um Framework para criação e descrição de processos de desenvolvimento de software customizados. Esta ferramenta que visa produzir um processo de software personalizável, com conteúdo de processo exemplar e funcionalidades que permitem uma ampla variedade de tipos de projetos e estilos de desenvolvimento.

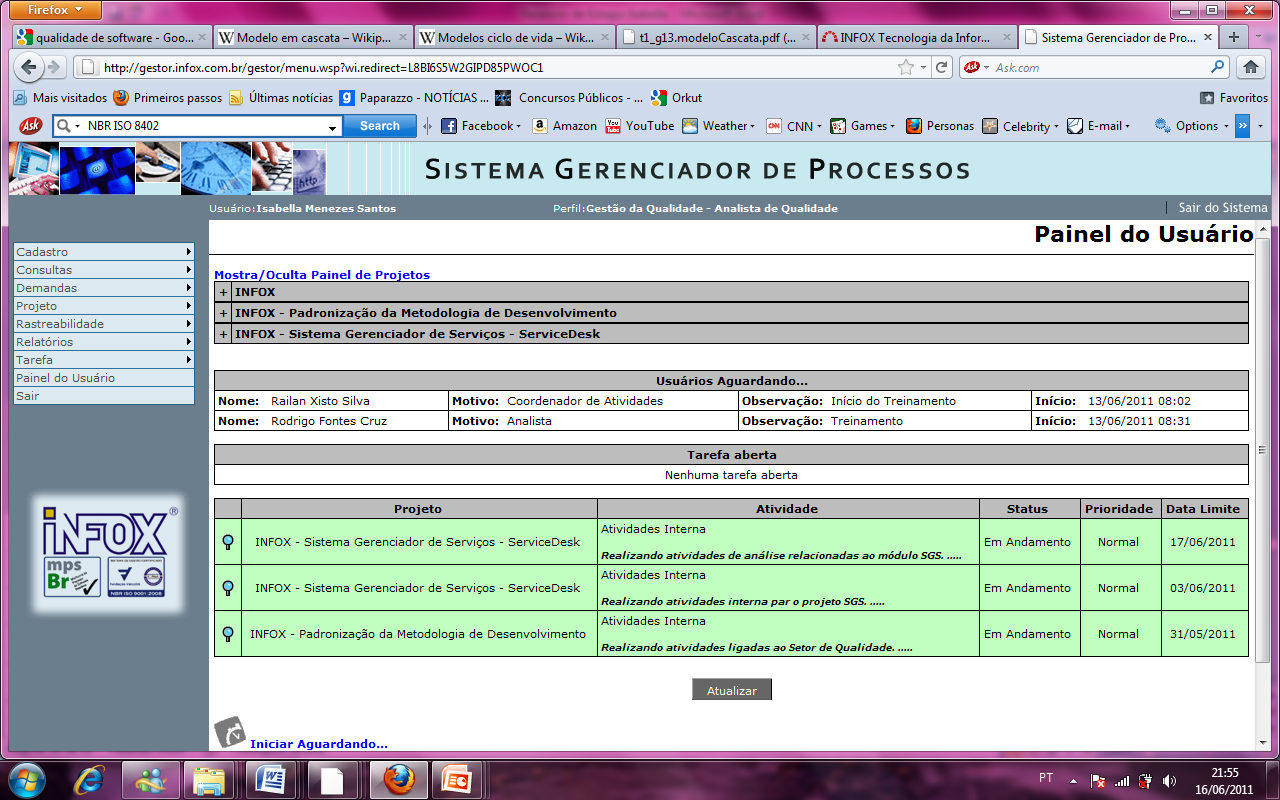
 A ferramenta conta com exemplos de processos prontos para serem adaptados e ferramentas que são suportadas por vários tipos de projetos e formas de desenvolvimento. A suíte é composta por uma ferramenta baseado no Eclipse que descreve o processo (EPF Composer) e alguns processos já prontos para serem customizados como: BasicUP que é uma versão básica do RUP (Rational Unified Process) e XP (eXtreme Programming) que descreve um processo de desenvolvimento Ágil.

Para descrever o processo a ferramenta mistura conceitos de disciplina de Engenharia de Software, ciclo de desenvolvimento, iteração, papeis, tarefas e descrição de como tudo deve ser feito. O cruzamento destes dados é feito de forma lógica gerando uma arvore de conhecimento que pode ser visualizada pelo Browser, bem parecido com o RUP da IBM.

**3.2- Sistema Gerenciador de Processos (SGP)**

O Sistema Gerenciador de Processos é uma ferramenta de apontamento e acompanhamento de tarefas, de requisitos, casos de uso, rastreabilidade vertical e horizontal dos casos de uso e requisitos (mecanismo que permite rastrear a dependência entre os requisitos e os produtos de trabalho), homologação das iterações dos projetos, demandas, cadastro do Método de Estimativa de Tempo do Caso de Uso dentre outros.

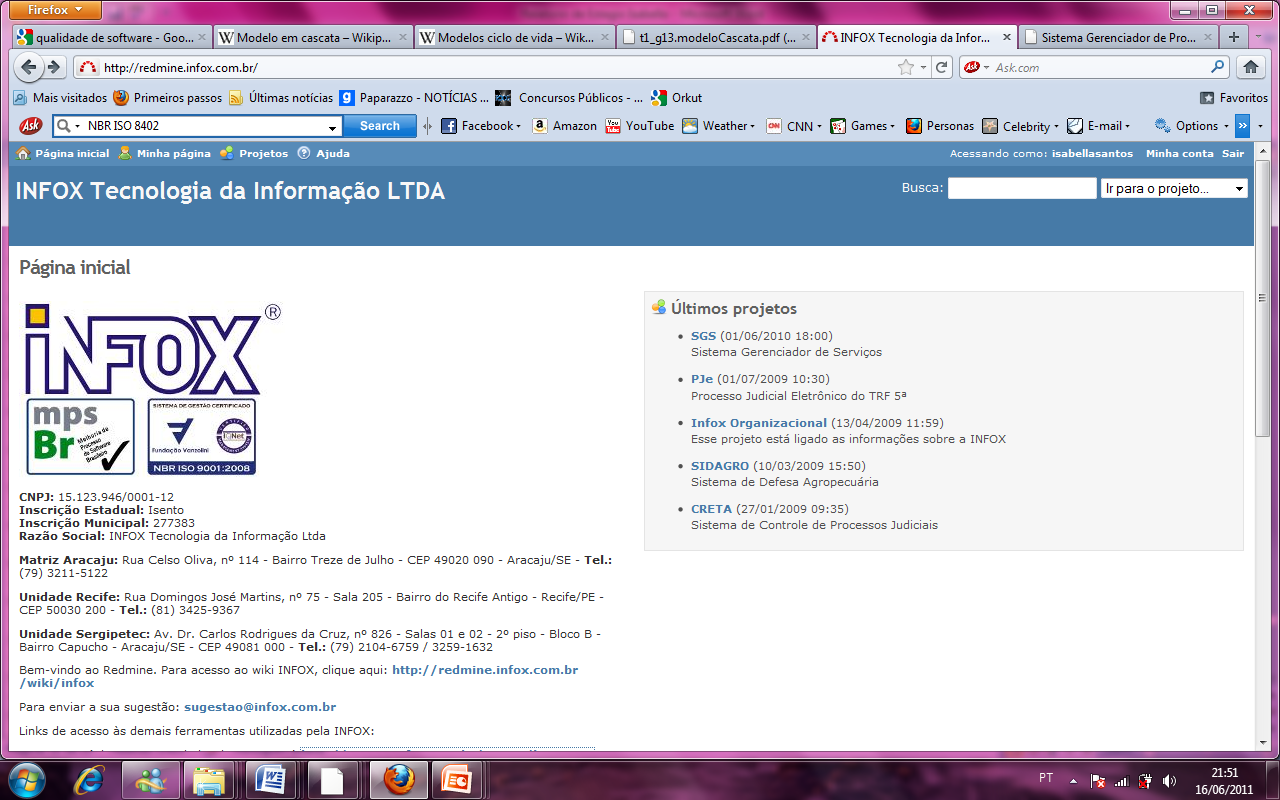
Cada usuário cadastra suas tarefas, definindo o projeto em que está envolvido, a atividade a ser realizada, e a data limite para término, como mostra a figura 4.

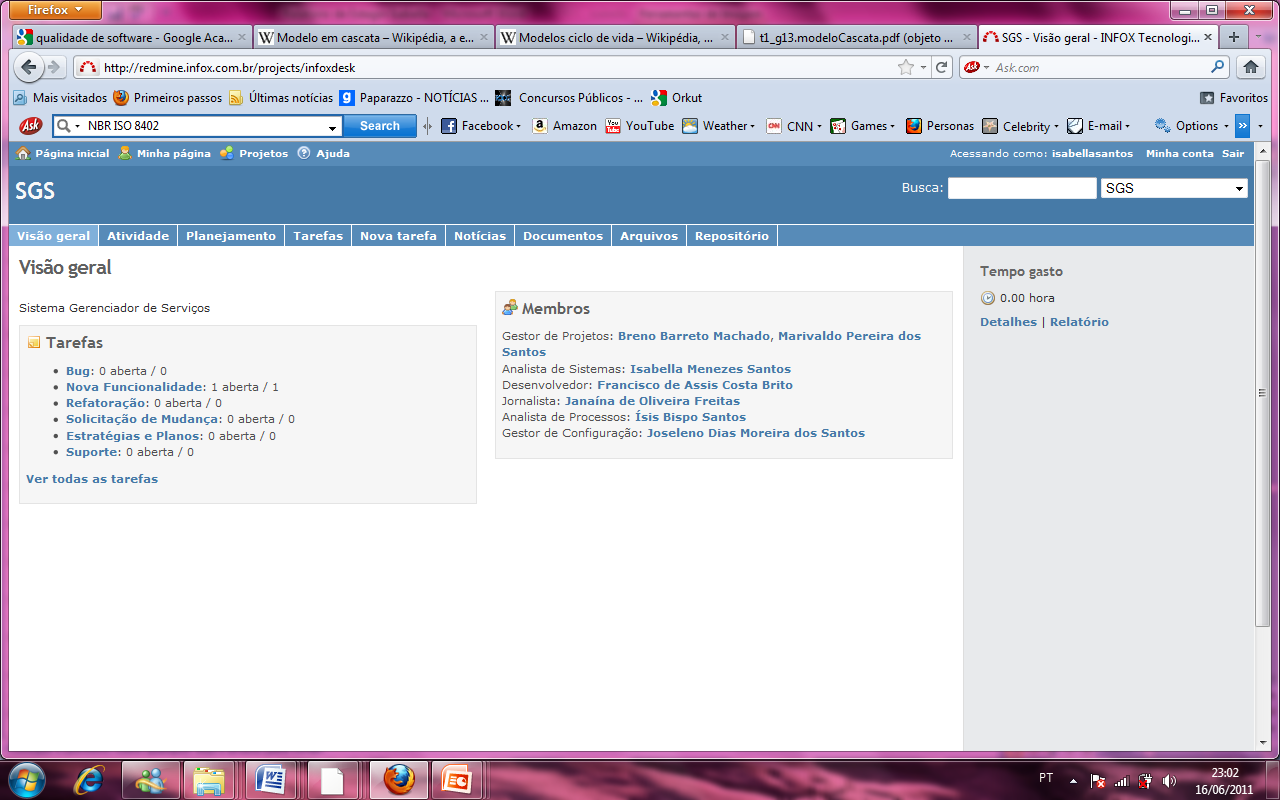


**Figura 4: Tela inicial do SGP**

**3.3- Redmine**

O *Redmine* é um software livre*,* gerenciador de projetos baseado na *web* e ferramenta de gerenciamento de *bugs*. Ele contém calendário e gráficos de *Gantt* para ajudar na representação visual dos projetos e seus *deadlines* (prazos de entrega). Essa ferramenta é utilizada na organização para gerenciar *bugs* e disponibilizar informações dos projetos através dos módulos *wiki*, fórum e notícias, como mostra a figura 6. Ele pode também trabalhar com múltiplos projetos, como mostra a figura 5.

**Figura 5: Tela inicial do Redmine**

****

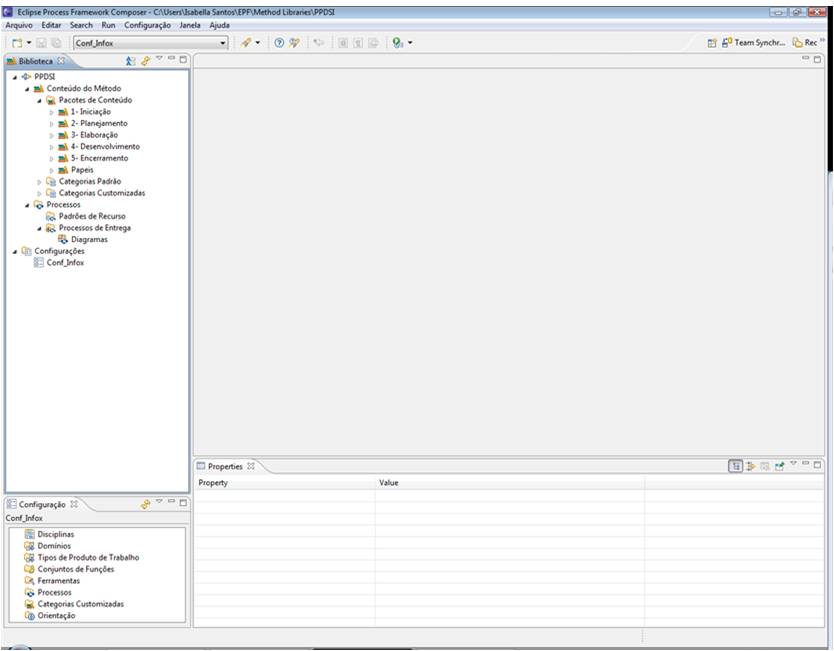
**Figura 6: Funcionalidades do Redmine**

1. **ATIVIDADES DO ESTÁGIO**
   1. **Modelando PPDSI no EPFComposer**

Como primeira atividade do estágio, foi desenvolvida a modelagem do Processo Padrão de Desenvolvimento de Software Infox – PPDSI, na ferramenta EPFComposer, primeiro feito uma pesquisa para definir uma ferramenta, visto que a maior necessidade era de uma ferramenta que fosse visualizada por todos, ou seja, disponibilizada na web, foi definido o EPFComposer, por ser uma ferramenta poderosa na criação de processos e por se tornar navegável, onde torna o entendimento do processo mais iterativo.

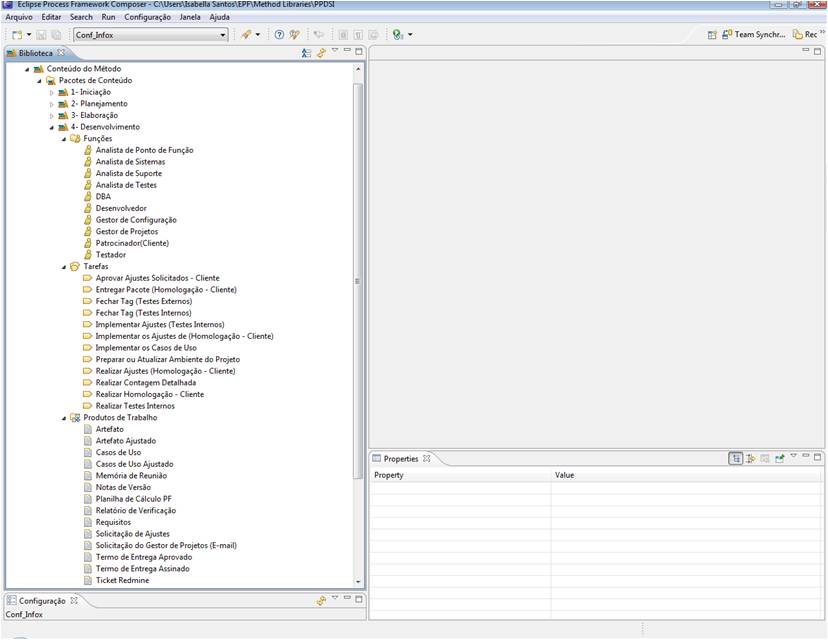
Para um melhor detalhamento desta atividade, abaixo segue as figuras com os passos necessários para criar o processo e fazer a publicação na internet.

Para a definição das fases do ciclo de vida do software, é preciso criar os Pacotes de Conteúdo, cada pacote representa uma fase.



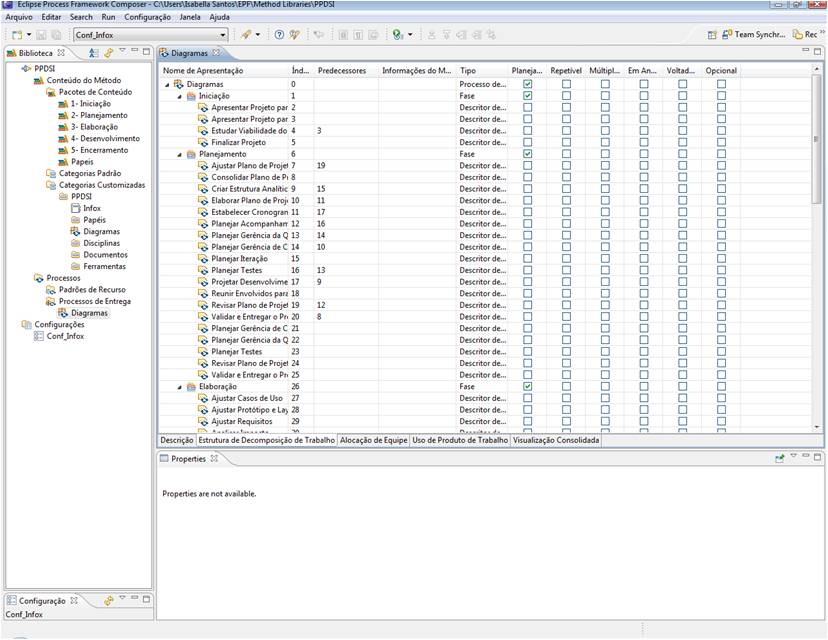
**Figura 7: Criação dos Pacotes de Conteúdo(Fases PPDSI)**

Para cada fase, são definidas as funções, as tarefas, e os produtos de trabalho.

****

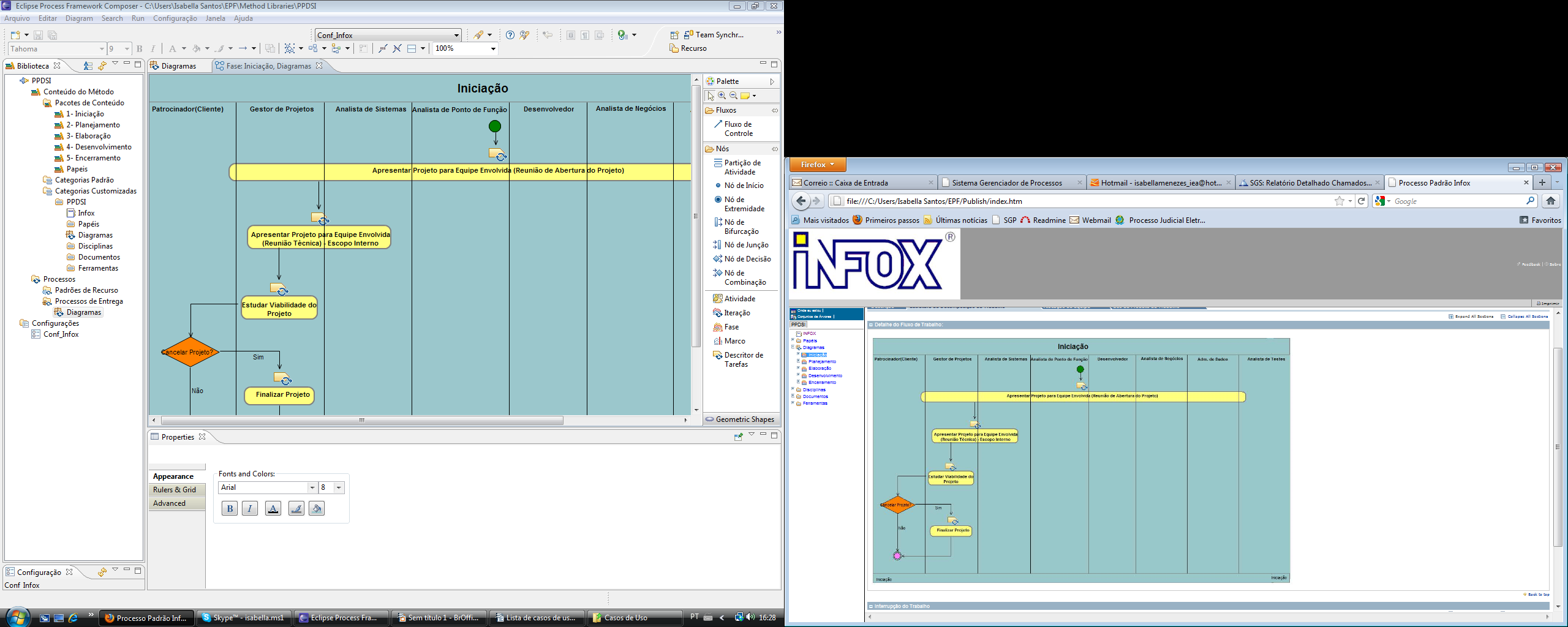
**Figura 8: Criando as funções, tarefas e produtos de trabalho**

É necessário criar os diagramas do processo, cada fase (agrupamento de tarefas) contém um diagrama de atividades, como mostra a figura 9.

****

**Figura 9: Agrupamento de tarefas**

Componentes do diagrama de atividades são utilizados para descrever o fluxo, como mostra na figura 10.

****

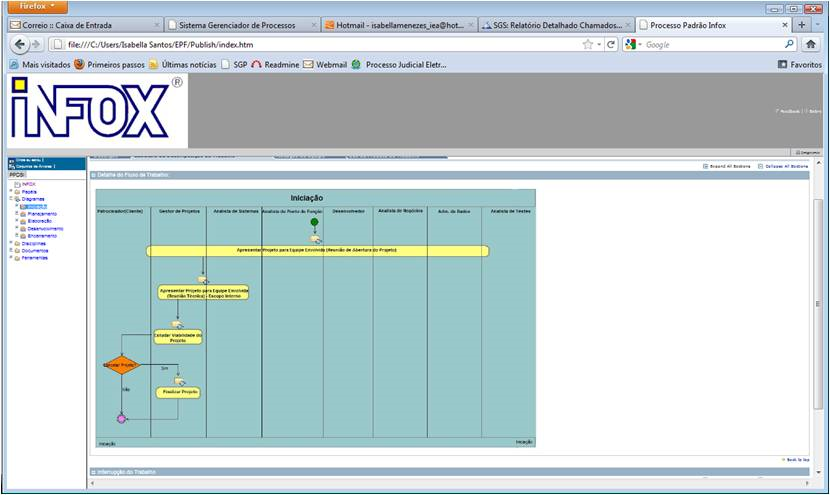
**Figura 10: Fluxo de atividades**

Publicando o processo, tornando-o navegável, onde pode ir navegando e visualizando os detalhes do diagrama por fase, como mostra na figura 11.

****

**Figura 11: Processo publicado**

Com o site gerado podemos visualizar todos os detalhes do processo e exibindo seu conteúdo, modelos padrões de documentos, informações sobre ferramentas utilizadas, entre outros, como mostra na figura 12.

****

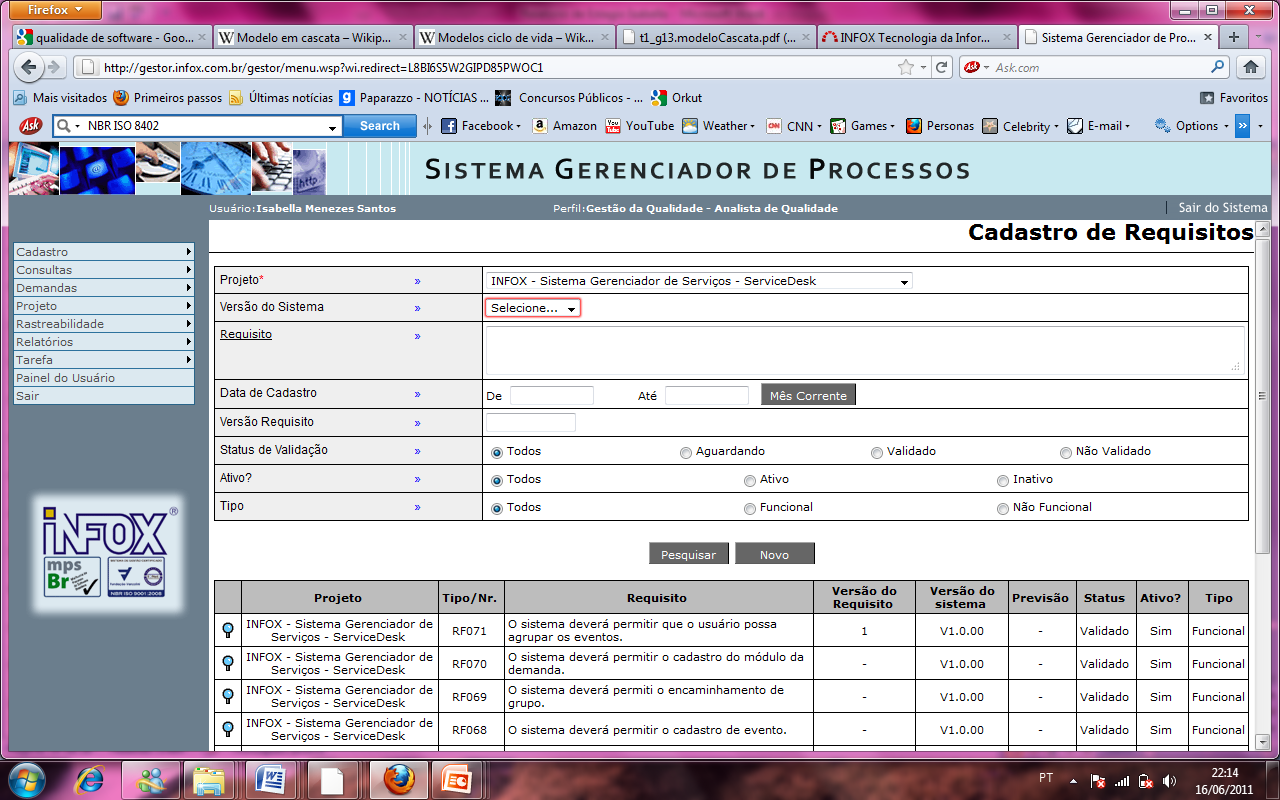
**Figura 12: Visualizando os diagramas**

* 1. **Avaliando melhorias no PPDSI**

Para se obter um bom processo de software torna-se necessário propor várias melhorias no processo padrão, buscando a melhoria contínua. Desta forma, participei de reuniões, propondo possíveis melhorias para o processo, dentre as quais, destaquei a possível adoção do Scrum, um framework ágil para a realização de projetos complexos, sendo bastante difundido por diversas empresas, por priorizar o trabalho em equipe e a entrega do produto em um curto espaço de tempo agregando assim um maior valor ao negócio do cliente o mais breve possível.

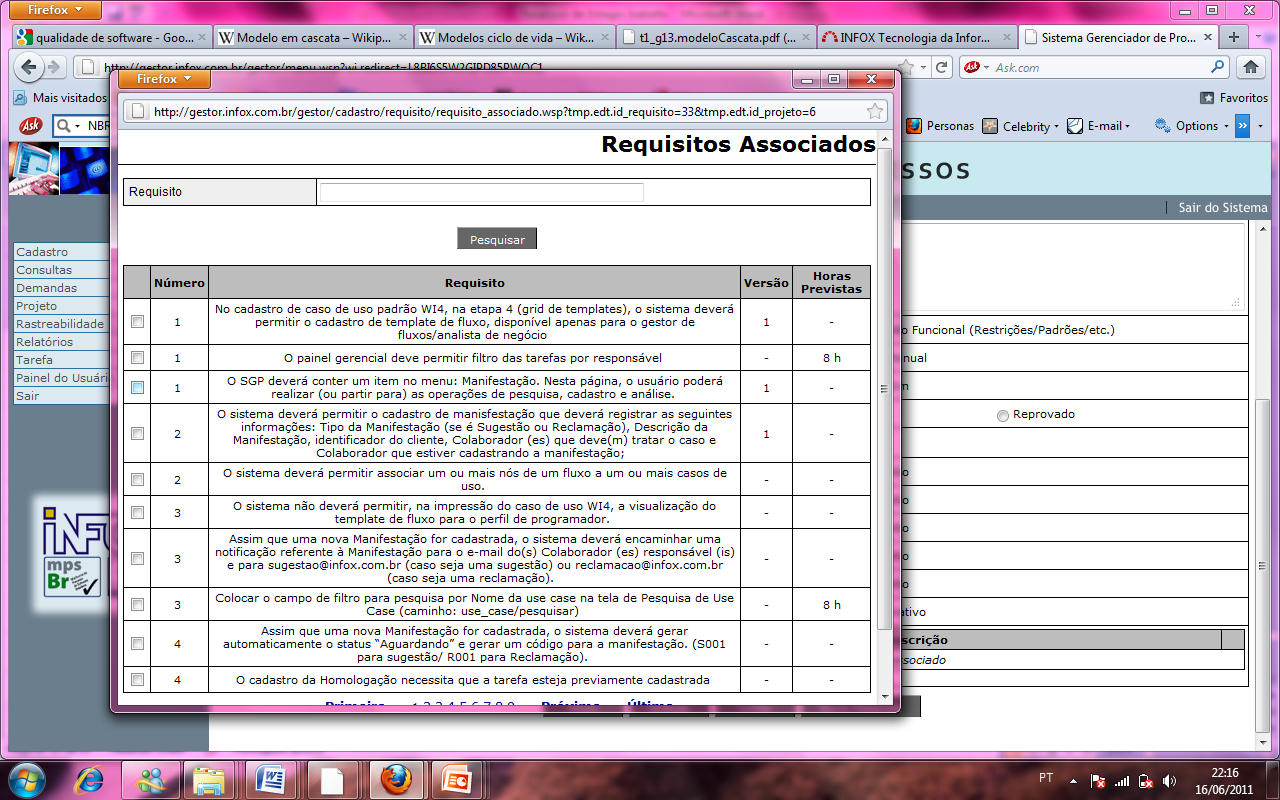
* 1. **Realizando a rastreabilidade bidirecional**

Outra atividade realizada no estágio foi a rastreabilidade bidirecional dos casos de uso na ferramenta SGP, utilizando um módulo do projeto interno da empresa denominado SGP, que atende ao setor de qualidade, obedecendo aos resultados esperados do processo de gerência de requisitos, do nível G MPS.br, que é gerenciar todos os requisitos recebidos pelo projeto, acompanhar e monitorar a rastreabilidade vertical e horizontal dos requisitos do projeto. Inicialmente foram cadastrados todos os requisitos, como mostra a figura 13.



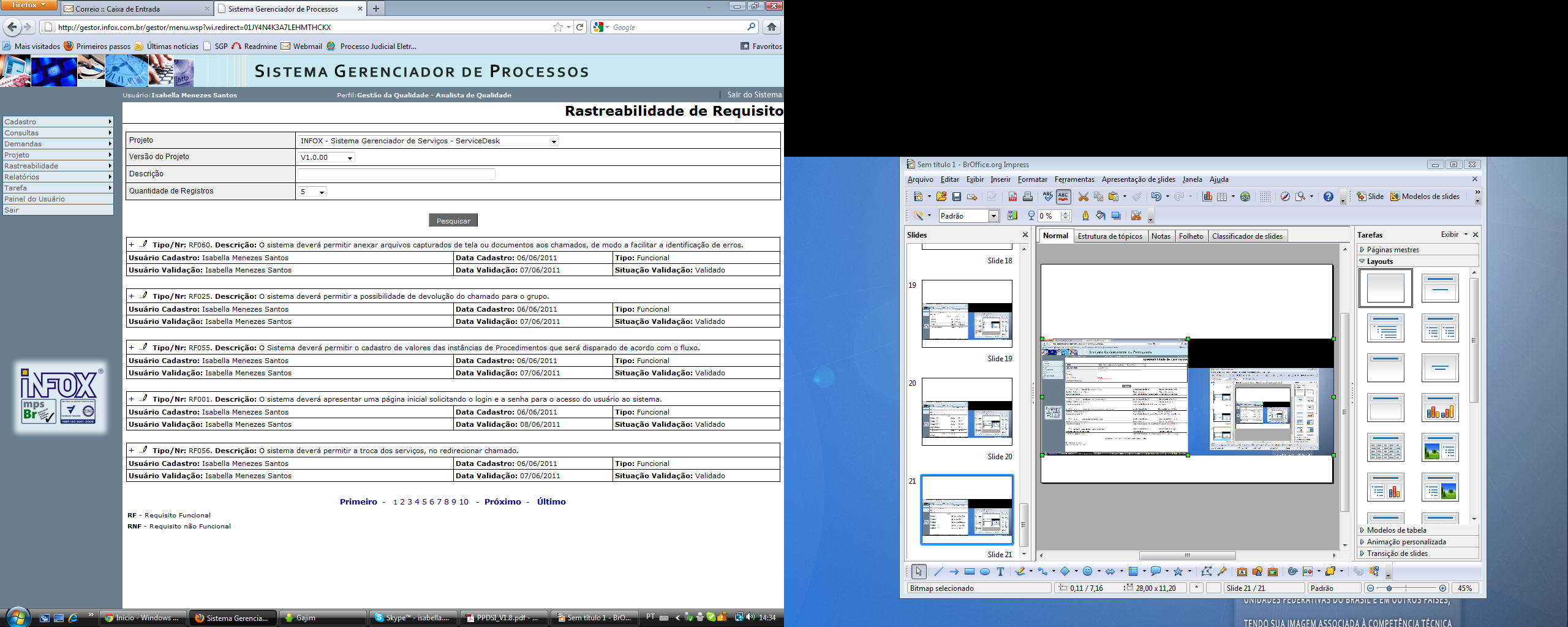
**Figura 13: Cadastro de Requisitos**

Logo após foi realizado a associação dos requisitos, ou seja, requisitos que associam com outros requisitos, como exibido na figura 14.



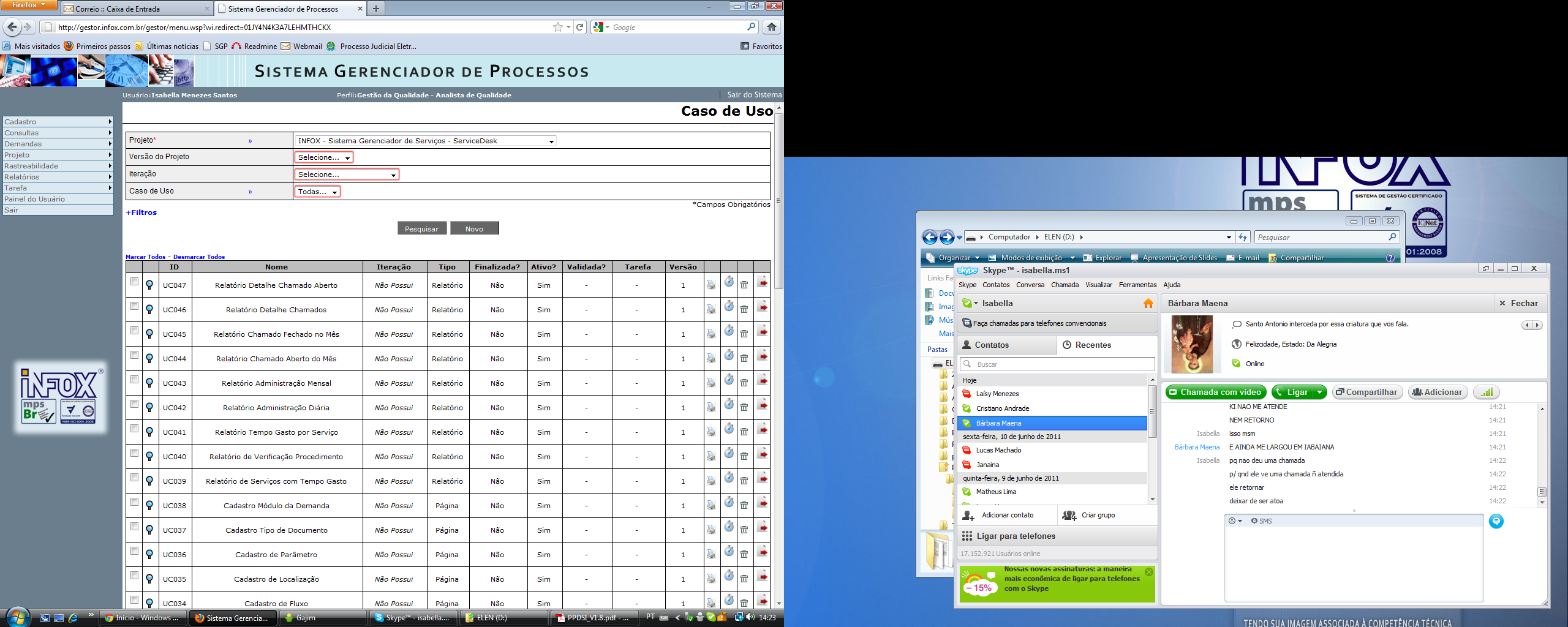
**Figura 14: Associação de Requisitos**

Após feita a associação dos requisitos, é permitido visualizar a rastreabilidade por requisitos, como mostra a figura 15.



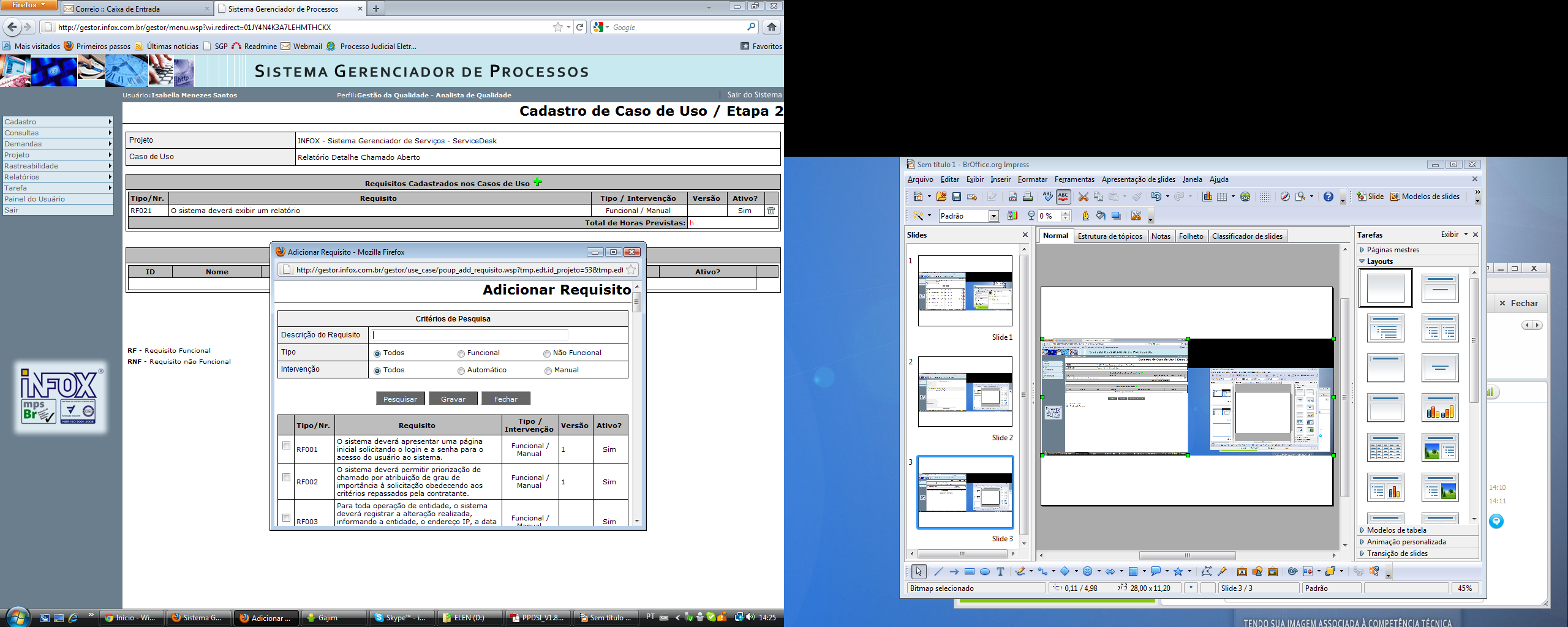
**Figura 15: Rastreabilidade por Requisitos**

Com os requisitos cadastrados, é realizado o cadastro de casos de uso, com todos os detalhes e informações de cada caso de uso, como mostra na figura 16.



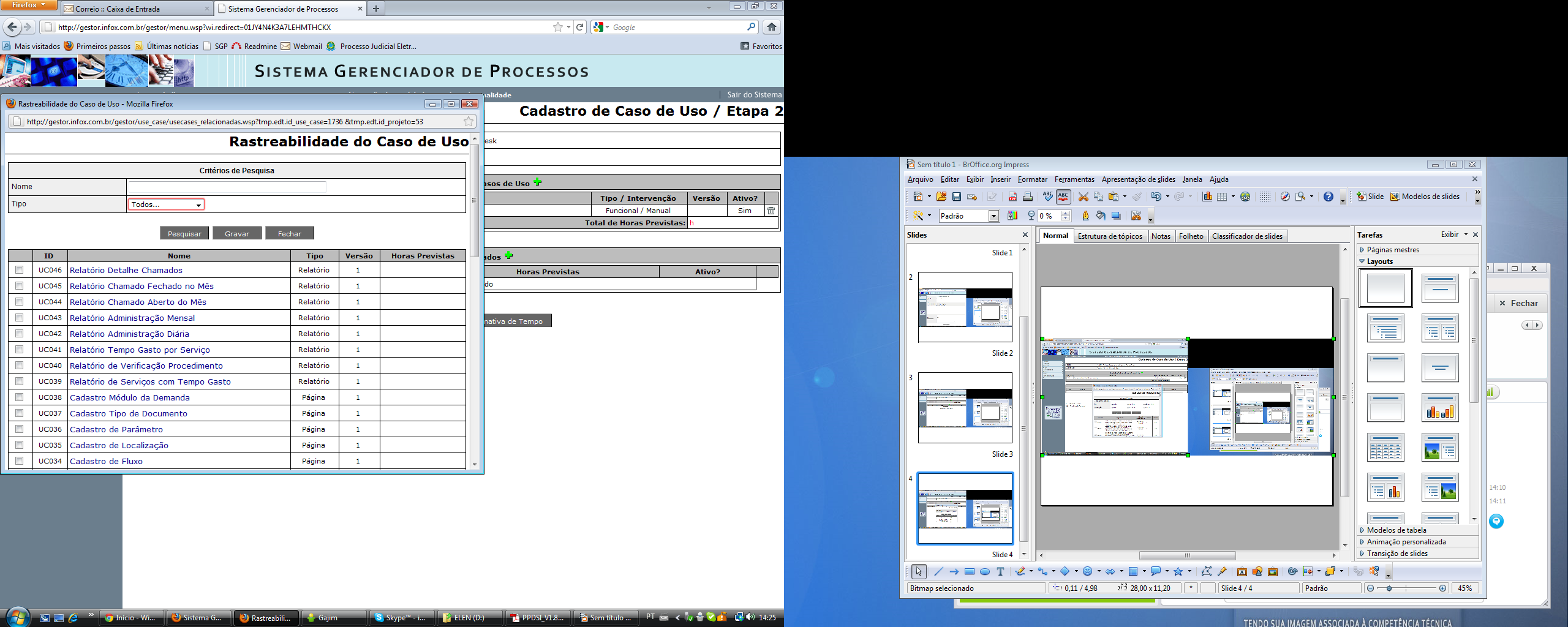
**Figura 16: Casos de Uso Cadastrados**

Também é feito a associação dos requisitos pertinentes ao caso de uso, como mostra na figura 17.



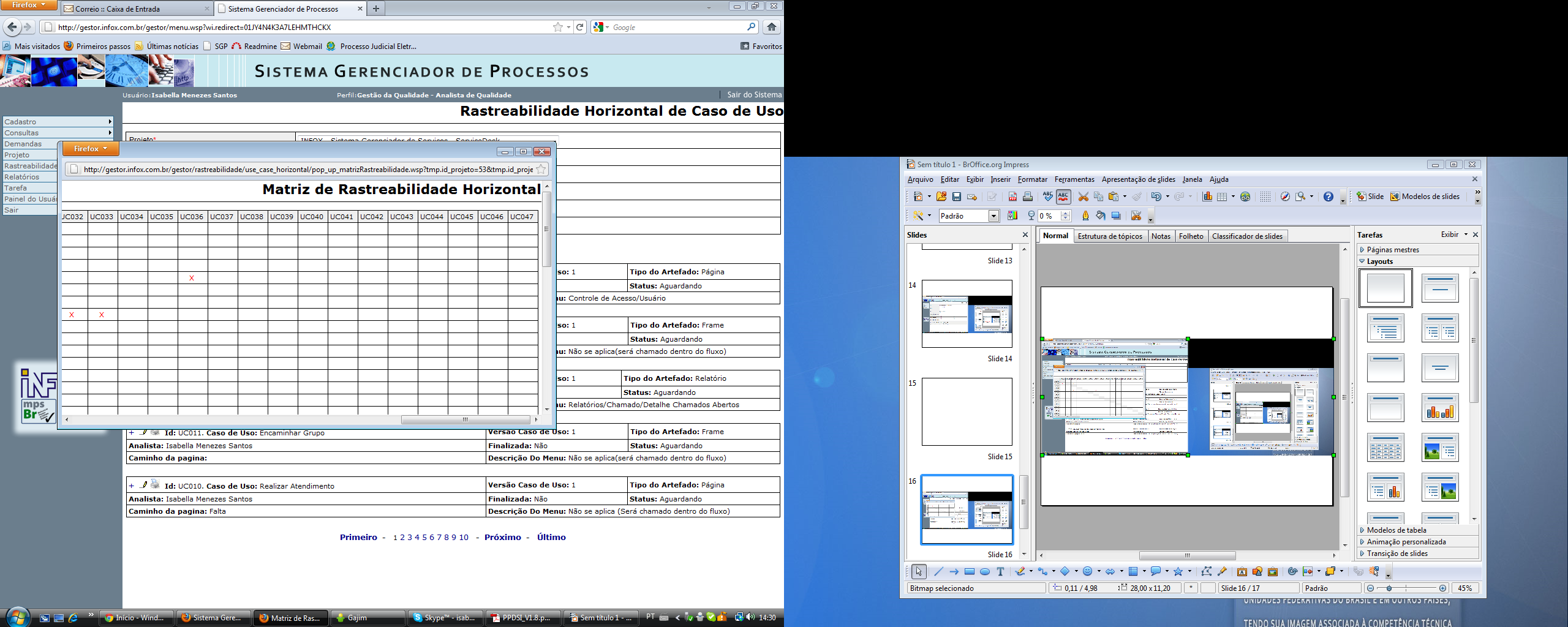
**Figura 17: Associando Requisitos com Caso de Uso**

Para manter a rastreabilidade bidirecional é feita a associação de caso de uso com caso de uso, como mostra a figura 18.



**Figura 18: Associação entre Casos de Uso**

A associação entre os casos de uso gera a matriz de rastreabilidade horizontal, como mostrado na figura 19.



**Figura 19: Matriz de Rastreabilidade Horizontal**

1. **CONCLUSÃO**

Com a institucionalização de processos, que seja adotado e usado por todos os envolvidos no projeto, com certeza irá proporcionar um aumento significativo na qualidade do produto, com redução do retrabalho, escopo atendido dentro do prazo e com funcionalidades previamente estabelecidas, entre outros problemas. Desta forma, realizar avaliações propondo melhorias no processo é de suma importância, para garantia da melhoria contínua.

A prática de definição e melhoria de processos de software está sendo adotada por diversas empresas que buscam organizar e controlar os processos dentro do ciclo de vida do software. A melhoria de software consiste na realização de ações que alterem os processos, para que estes satisfaçam, de forma mais eficiente, os objetivos e necessidades das organizações, com a finalidade de reduzir custos, prazos e problemas relacionados a escopo, buscando aumentar sua produtividade, satisfação do cliente, estabelecendo uma linguagem comum entre toda a equipe, construindo uma visão compartilhada e ao mesmo tempo, uma estrutura para priorizar ações e projetos.

Contudo, vários aprendizados foram adquiridos a partir do dia-a-dia com reuniões e discussões com profissionais de diversas áreas, já que para garantir a qualidade tem que entender como as partes funcionam, ou seja, entender o todo, como as pessoas estão trabalhando, as ferramentas que estão sendo utilizadas, e se estão utilizando o processo da maneira correta. Esse conhecimento adquirido a partir da execução de tarefas em projetos reais é muito importante, visto a importância da disciplina Estágio Supervisionado para a melhor formação do discente.

1. **REFERÊNCIAS BIBLIOGRÁFICASx**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**x**

1 STANDISH GROUP. **Chaos**: pesquisa sobre o desenvolvimento de software e o panorama caótico da indústria de software dos dias de hoje. Disponível na Internet. <www.standishgroup.com/chaos.html>. Acesso em 08 de junho de 2011.

2 [Salviano and Jino 2006a] Salviano, C. F., Jino, M.: **Uma Proposta Orientada a**

**Perfis de Capacidade de Processo para Evolução da Melhoria de Processo de**

**Software. In: Anais do Quinto Simpósio Brasileiro de Qualidade de Software**

(SBQS), Vitória, ES, Brasil, 2006, p. 393-408.

3 ABNT, NBR ISO 8402. Disponível no site: <http://www.abntcatalogo.com.br/norma.aspx?ID=58542>. Acessado em 09 de junho de 2011.

4 SOFTEX. **Sociedade Brasileira para Promoção da Exportação de Software - Histórico**. 2009. <Disponível em http://www.softex.br/\_asoftex/historico.asp#>. Acessado em abril de 2011.

5 \_\_\_\_\_\_\_. **MPS.BR Melhoria de Processo de Software Brasileiro - Guia Geral**. 2009a. Disponível em <http://www.softex.br/mpsbr/\_guias/guias/MPS.BR\_Guia\_Geral\_2009.pdf>. Acessado em abril de 2011.

6 ECLIPSE - IDE. Disponivel em: <http://www.eclipse.org/org/>. Acesso em: 27 de maio de 2011.

7 CASCATA. Modelo Clássico. Disponível em <http://www.api.adm.br/GRS/referencias/t1\_g13.modeloCascata.pdf> Acessado em 24 de maio de 2011.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |