

## **Ferramenta para Automatizar o uso do Personal Software Process**

**Área de Pesquisa: Ciências Exatas e da Terra**

**Assis – Dezembro/2013**

Projeto submetido ao Programa  
Institucional de Bolsas de Iniciação Científica

PIC/FEMA/ 2013

Autores:

Rodolfo Camarotto Santos – Aluno

Dr. Luiz Ricardo Begosso - Orientador

## Sumário

1	Introdução	4
2	O Modelo PSP	5
3	O Software Process Dashboard	9
3.1	Importância da ferramenta para uso do PSP	9
3.2	Funcionalidade	9
3.3	Facilidade de uso	10
3.4	Flexibilidade	10
3.5	Independência de Plataforma	10
3.6	Custo	10
3.7	Requisitos para uso da ferramenta	11
3.8	Versões	11
3.9	Utilização	11
3.10	Registro de tempo	13
3.11	Ferramenta PROBE – Estimativa de tamanho	14
3.12	Registro de defeito	15
3.13	Menu de Scripts	15
3.14	Hierarquia	16
3.15	Indicador de nota	17
4	Conclusões	17
5	Referências	18

## 1. Introdução

O processo de engenharia de software de uma organização é altamente influenciado pelas ferramentas adotadas, regras de negócio, estrutura organizacional, métodos de gerenciamento, políticas estruturais, ambientes de trabalho, formas de comunicação, interações sociais, e pessoas.

As organizações de desenvolvimento de software possuem uma série de mecanismos para auxiliá-las no gerenciamento e melhoria da qualidade dos seus produtos. Os aspectos relacionados à garantia da qualidade de software têm sido muito ressaltados nos últimos anos, especialmente pelo fato das empresas estarem preocupadas com a qualidade em todos os seus segmentos.

A falta de adoção de práticas de qualidade pelas pequenas organizações de desenvolvimento de software contribui para a alta taxa de mortalidade destas empresas, já que parte de seus escassos recursos são desperdiçados por falta de planejamento adequado. Além disso, estima-se que uma quantidade elevada dos serviços realizados pode ser definida como retrabalho, comprometendo os custos e os prazos estabelecidos [1].

Uma das soluções propostas para a melhoria da qualidade do software produzido pelas pequenas empresas ou organizações individuais relaciona-se com a adoção formal de um processo de desenvolvimento de software. O PSP – Personal Software Process [2] é uma alternativa que tem se mostrado bastante eficiente neste sentido, sendo que sua adoção incorpora benefícios aos desenvolvedores individuais ou às pequenas equipes de desenvolvimento de software. Os aspectos sobre qualidade de software têm sido altamente valorizados pela indústria de desenvolvimento de software. O número de pequenas empresas de desenvolvimento de software é muito grande e justamente por isso torna-se fundamental o estudo de práticas de qualidade voltadas para este público alvo. A informatização do processo PSP contribui para as equipes de desenvolvimento de software, facilitando a sua adoção.

Este projeto tem o objetivo de realizar um estudo sobre o Processo Pessoal de Desenvolvimento de Software PSP e, em seguida, analisar uma ferramenta que automatize o processo do PSP, que é o Software Process Dashboard, que é uma iniciativa open-source para criar uma ferramenta de suporte ao PSP.

## 2. O Modelo PSP

O modelo PSP foi criado por HUMPHREY [1] como um processo de qualidade para auxiliar no controle, gerenciamento e melhoria da forma pessoal de desenvolvimento de software, possibilitando a apropriação de dados históricos para o atendimento de requisitos e tornando os elementos de rotina do trabalho mais previsíveis e eficientes. Assim, o PSP pode ser usado nos processos de planejamento, previsão (especialmente de prazos, custos e recursos), e medição da qualidade dos produtos de software.

Para obter, garantir e melhorar a qualidade de um software, pesquisadores ao longo do tempo desenvolveram diversos processos que auxiliam e facilitam nessa tarefa, focando em desenvolver um software com qualidade, são processos que já estão estruturalmente bem definidos, testados e com eficiência aprovada.

Dentre os modelos conhecidos temos os mais famosos como o CMM, SIGMA S, e o processo adotado na pesquisa, o PSP, que foi criado por HUMPHREY em 1995, trata-se de um processo para engenheiro individual, que resulta em auto-melhoria possibilitando ao programador informações precisas para que o mesmo possa se organizar, planejar e tomar medidas que melhorem a forma como trabalha.

O processo é bem estruturado por meio de formulários, e procedimentos a serem seguidos no desenvolvimento do software, sendo que cada nível do PSP trás consigo seus próprios scripts, de forma incremental, pois o nível superior adiciona scripts ao nível anterior. O processo possibilita ainda um histórico dos trabalhos já realizados assim o programador saberá como melhor distribuir seu tempo e prever datas para término de cada etapa do desenvolvimento do sistema.

O PSP é um processo bem definido que é dividido em 4 níveis e 3 sub-níveis, conforme ilustrado na Figura 1.

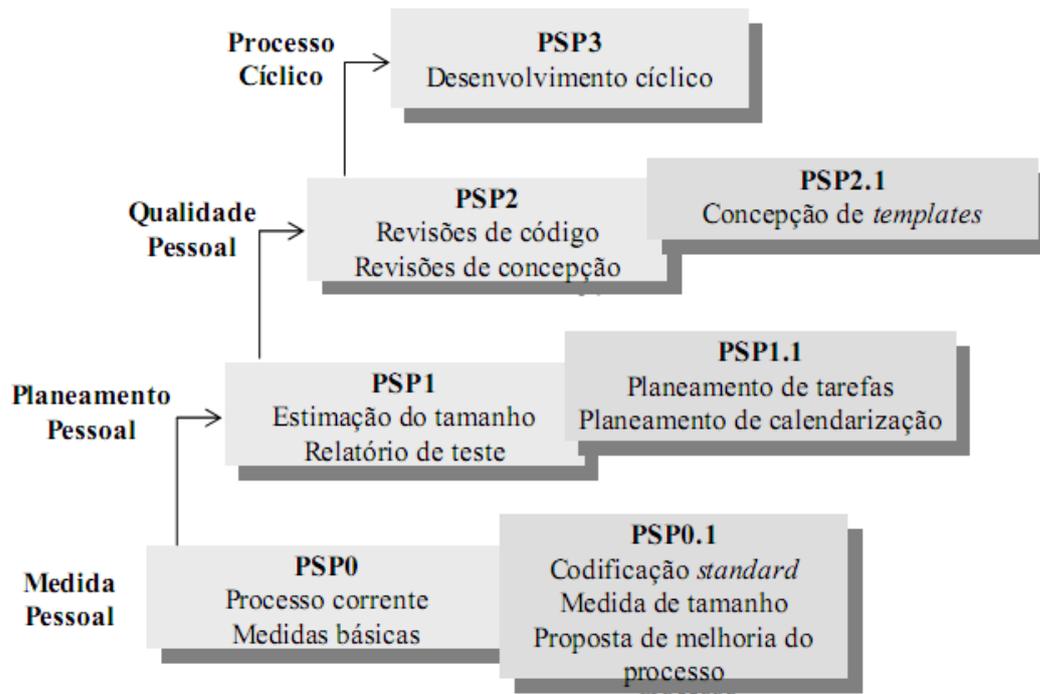


Figura 1 – Estrutura do PSP

- PSP 0 – A base - o primeiro passo do PSP é estabelecer uma base (baseline) que inclui algumas medidas básicas e um relatório. Esta *base* providencia uma base concreta para medir o progresso e uma fundação definida sobre à qual melhorar. O PSP 0 deve ser o processo normal que você usa para escrever software.
- PSP 0.1 – Medindo o tamanho do software - o PSP 0 é reforçada para o PSP 0.1, adicionando um padrão de código, medidas de tamanho e a proposta de melhoria de processo (PIP – Process Improvement Proposal). O PIP é a forma que providencia uma maneira estruturada para gravar os problemas do processo, experiências e sugestões de melhorias.

- PSP 1 – Estimando tamanho e tempo - O processo de planejamento pessoal: o PSP 1 adiciona o planejamento de passo ao PSP 0. O Incremento inicial adiciona um relatório de testes e estimativas de recursos e tamanhos.
- PSP 1.1 – Planejamento de tarefas e tempo
- PSP 2 – Melhorando a qualidade - O processo de administração da qualidade pessoal: para administrar os seus defeitos, o desenvolvedor tem que saber quantos ele faz. O PSP 2 acrescenta técnicas de revisão ao PSP 1, para ajudar a achá-los no início, quando forem menos caros para resolver. Faz-se isto juntando e analisando os defeitos achados na compilação e nos testes dos primeiros programas. Com esses dados, o desenvolvedor pode estabelecer listas de conferência de revisão e fazer suas próprias avaliações de qualidade e processo.
- PSP 2.1 – Modelos de projeto - estabelece critérios de perfeição de projeto e examina varias técnicas de verificação e consistência de projeto.
- PSP 3 – Desenvolvimento Cíclico - A estratégia do PSP 3 é subdividir um programa maior em pedaços do tamanho requerido pelo PSP 2. A primeira construção é um modulo básico ou núcleo (KERNEL), que aumenta em ciclos de iteração. Em cada repetição, faz-se um PSP 2 completo, incluindo projeto, compilação e testes. Assim, o PSP 3 é satisfatório para programas de até varias mil linhas de código. O processo PSP 3 cíclico, efetivamente escala programas grandes contanto que cada incremento sucessivo seja de alta qualidade.

A lógica para o PSP é a seguinte:

- Os Profissionais de software entenderão melhor o que eles fazem se eles definirem, medirem e monitorarem o seu trabalho.
- Eles terão uma estrutura de processo definida e critérios mensuráveis para avaliar e aprender através de suas próprias experiências e das dos outros.

- Com este conhecimento e experiência, eles podem selecionar quais métodos e práticas que melhor se adaptem as suas tarefas e habilidades particulares.
- Usando um conjunto padronizado ordenadamente, constantemente praticado, e práticas pessoais de alta qualidade, eles serão os membros mais efetivos dos seus grupos de desenvolvimento e projeto.

Essa lógica é baseada em cinco princípios. Esses princípios são resumidos abaixo:

1. Um processo definido e estruturado pode melhorar a eficiência do trabalho.
2. Processos pessoais definidos devem se ajustar às habilidades individuais e preferências de cada engenheiro de software.
3. Para os profissionais estarem confortáveis com o processo definido, eles devem estar envolvidos em sua definição.
4. Como as perícias e as habilidades dos profissionais evoluem, esses processos deveriam fazer o mesmo.
5. A melhoria continua de processo é alcançada através de rápido e explícito feedback.

Para aprendizado e utilização do processo, deve-se iniciar do PSP 0, e ir progredindo de nível com o uso, pois o processo é incremental, e cada nível que avança é

Para utilização do processo faremos um projeto piloto, trata-se de uma agenda telefônica disponibilizada via WEB, utilizaremos da tecnologia Java Server Faces, em conjunto com Hibernate e Prime Faces, também utilizaremos o banco de dados MySQL para o projeto.

Durante o desenvolvimento da aplicação será aplicado o PSP 0, para termos uma noção e visualização do uso do processo, assim na próxima etapa será feito uma aplicação que automatize o uso do processo.

### **3. O Software Process Dashboard**

O Software Process Dashboard [3] é uma ferramenta de suporte ao PSP, que começou a ser desenvolvida em 1998 pela Força Aérea dos Estados Unidos e continuou a ser evoluída sob o modelo de código aberto. A ferramenta dá suporte às fases de levantamento de informações, planejamento, acompanhamento do projeto, análise de dados e exportação de dados para outros ambientes.

A ferramenta esta disponível sob as condições da Licença Pública GNU.

#### **3.1. Importância da ferramenta para uso do PSP**

O modelo PSP requer um refinamento muito detalhado na aquisição dos dados e principalmente na análise dos dados que resultarão em uma métrica, sendo assim o uso do método é algo complexo. Essa complexidade acaba impossibilitando aos engenheiros utilizarem o método no dia a dia devido à necessidade de maior agilidade.

Pensando em minimizar este problema da complexidade e grande demanda de tempo na utilização do método PSP é que a ferramenta foi introduzida. Garantindo agilidade no registro de dados, análise rápida dos dados, facilidade na exibição das métricas e integração com ambientes de desenvolvimento.

#### **3.2. Funcionalidade**

- Coleta de dados: tempo, defeitos, tamanho.
- Planejamento: integrado scripts, modelos, formulários e resumos.
- Rastreamento: poderoso suporte
- Análise de dados: gráficos e relatórios que ajudam a análise e tendência de dados históricos

- Exportação de dados: permite exportar dados para o excel ou para formato de texto para uso em ferramentas externas

### **3.3. Facilidade de uso**

- Coleta das métricas mais comumente realizadas (tempo e defeitos)
- Scripts, formulário, modelos e resumos, com rápido acesso
- Tela pequena permitindo o uso integrado com ambientes de desenvolvimento
- Tarefas organizadas hierarquicamente

### **3.4. Flexibilidade**

- Novos processos e tipos de dados podem ser adicionados sem programação
- Roteiros de processo, modelos, formar e resumos em HTML

### **3.5. Independência de Plataforma**

- 100% implementação Java – Pode ser utilizado em qualquer Sistema Operacional (Windows, Unix, Linux, Macintosh, etc).

### **3.6. Custo**

- Por se tratar de uma ferramenta open-source, é distribuída sem nenhum custo.

### 3.7. Requisitos para uso da ferramenta

Para utilização da ferramenta é necessário que se tenha instalado no mínimo a versão 1.5 da máquina virtual Java instalada na máquina, Java Runtime Environment. Ativar o JavaScript no browser. Em caso de uso de proxy, configurar o navegador para ignorar o proxy no endereço de localhost.

### 3.8. Versões

A ferramenta esta disponível no site [www.processdash.com](http://www.processdash.com) em mais de uma versão, sendo elas:

- Process Dashboard: versão que trás o painel e o template de projeto do PSP nível 2.1;
- Process Dashboard with PSP and TSP Process Extensions: versão que trás o painel com extensões para o TSP e mais templates de níveis de projeto para o PSP;
- Process Dashboard with PSP Course Support: versão que contempla dois módulos, que pode ser de Estudante e Instrutor do modelo PSP, trás o painel com template de projeto de vários níveis do PSP e materiais de guia de uso, em inglês os materiais;
- Process Dashboard Source Code: a ferramenta é open source e sobre a licença GNU, então, por fim é disponibilizado o código fonte da ferramenta.

É interessante utilizar a versão que trás suporte aos vários níveis do PSP, assim o processo de aprendizado do modelo PSP se torna mais fácil, e de forma incremental conforme os níveis.

### 3.9. Utilização

Por ser uma ferramenta feita para trabalhar em integração com as IDEs de desenvolvimento, o painel dela é pequeno e prático.

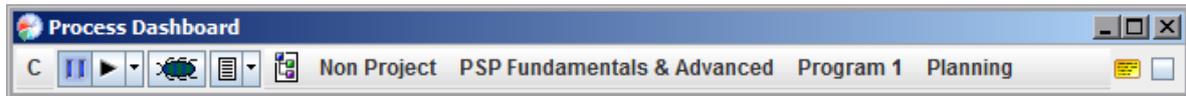
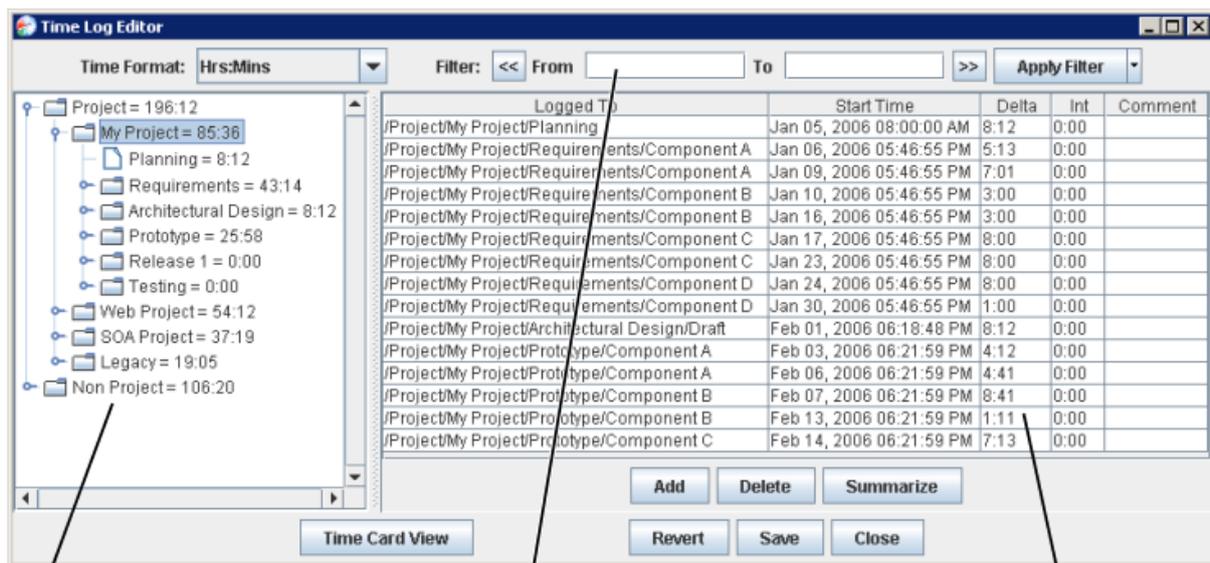


Figura 2 – Menu Principal da Ferramenta

- Menu de configuração: dá acesso a todas ferramentas desde hierarquia de divisão de tarefas, registro de tempo e defeitos, ferramenta PROBE (estimativa tamanho do software).
- Botão Play e Pause: adiciona automaticamente as entradas no log de tempo.
- Botão de Defeito: traz a o menu de diálogo para adicionar um defeito no registro de defeito.
- Botão de Script: dá acesso aos scripts e formulários de processos.
- Série de menus: permite navegar na hierarquia do projeto.
- Indicador de nota: adicionar comentários para aquela fase do projeto.
- Caixa de conclusão: marcar como finalizada aquela fase do projeto e avançar para próxima.

### 3.10. Registro de tempo

Com o uso do botão de registro de tempo, após a entrada dos registros nas diversas fases do projeto, o log de registro de tempo é exibida assim:



Hierarchical project view

Date filter

Time entries

Figura 3 – Log de Registro de tempo

No canto esquerdo é mostrada a hierarquia do projeto e suas respectivas fases, com o tempo total de registro em cada uma delas, há o campo de filtrar pela data, e as entradas são listadas na tabela, com opção de adicionar novas ou apagar alguma.

### 3.11. Ferramenta PROBE – Estimativa de tamanho

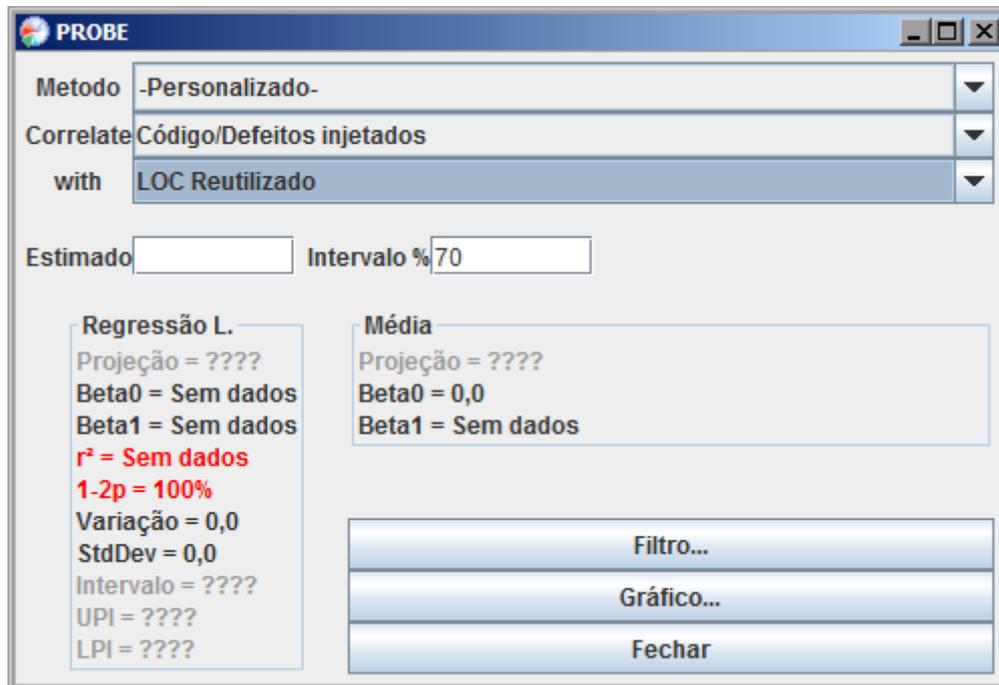


Figura 4 – Estimativa de tamanho – Ferramenta PROBE

Com base em uma serie de filtros e metodos é possivel gerar gráficos comparando os dados atuais do projeto com sua estimativa inicial.

Os métodos podem ser por tamanho ou tempo.

E as correlações de comparação podem ser por defeitos, tamanho, tempo, número de linhas, e pela fase do projeto.

### 3.12. Registro de defeito

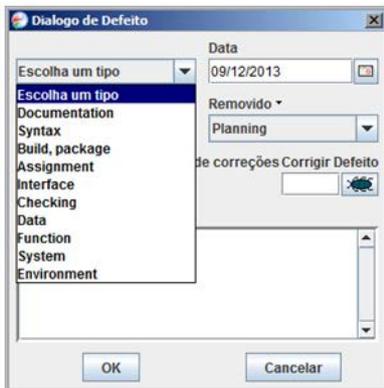


Figura 5 – Tipo de Defeito

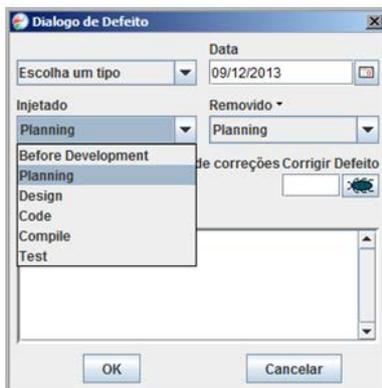


Figura 6 – Fase Encontrado

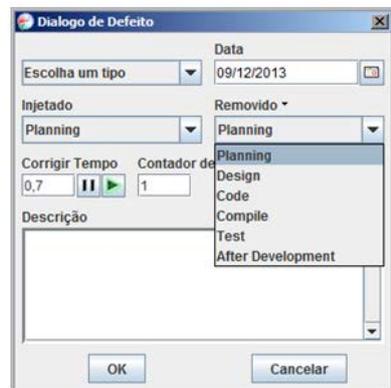


Figura 7 – Fase Remoção

O menu de contexto de registro de defeito aborda opções como: Tipo de defeito, fase em que foi encontrado, e fase em que foi removido. Contêm o log de tempo que levou no reparo do defeito, e um botão de defeito para inserir um defeito novo encontrado durante a remoção deste. Ainda é possível inserir a descrição do defeito no campo de texto abaixo da tela.

### 3.13. Menu de Scripts

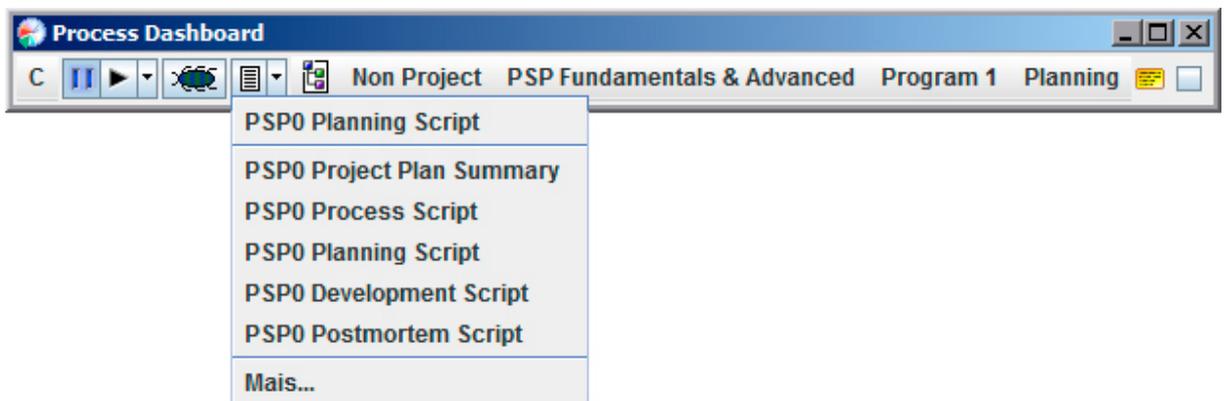


Figura 8 – Menu de Scripts

O menu de script exibe todos os scripts contidos em determinada fase do projeto, os scripts são carregados via HTML e as entradas vêm dos registros do sistema, ou conforme o script você quem insere o conteúdo necessário para determinado campo.

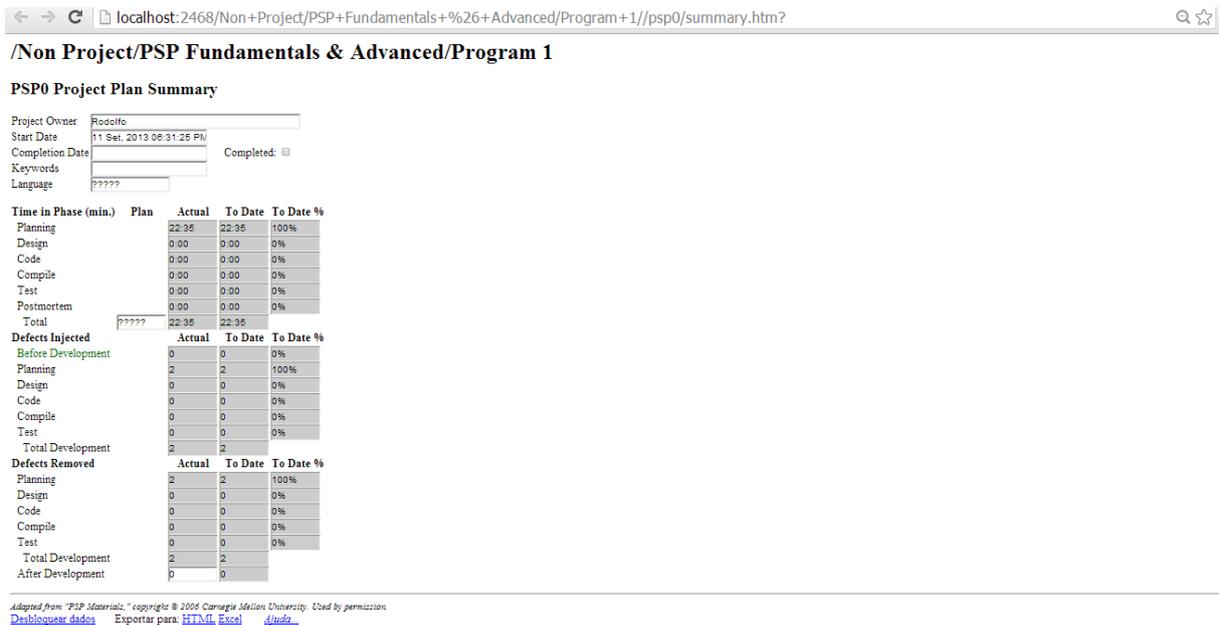


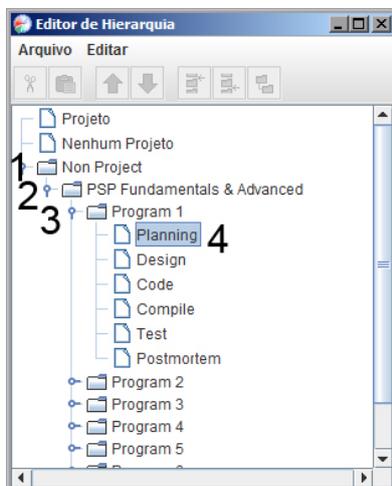
Figura 9 – Script Aberto em HTML

### 3.14. Hierarquia

A hierarquia do projeto é exibida no menu principal.



Figura 10 – Menu Principal - Hierarquia



Hierarquia do projeto acessada pelo menu de configuração e como é mostrado a hierarquia no menu principal:

1. Nome do projeto inicial da hierarquia
2. Nome do Segundo na hierarquia de projeto
3. Nome do programa
4. Nome da fase

Figura 11 – Menu de Hierarquia

### 3.15. Indicador de Nota

É possível inserir notas conforme a fase do projeto, basta clicar no menu de indicador de nota, escrever e salvar, há como editar no futuro essa nota, acrescentando ou excluindo texto.

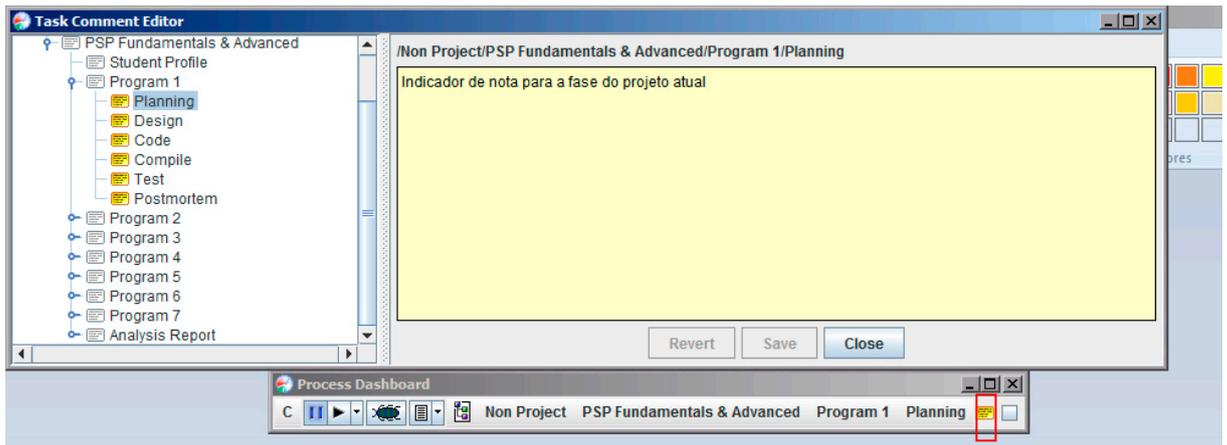


Figura 12 – Indicador de Nota

## 4. Conclusões

A ferramenta Software Process Dashboard engloba uma série de métricas importantes para o processo de desenvolvimento de software. Além disso, um conjunto de características gráficas relacionadas ao cronograma e defeitos encontrados permite ao desenvolvedor de software um melhor entendimento do andamento do projeto.

O ganho de tempo garantido no uso do PSP com auxílio da ferramenta Software Process Dashboard é o principal atrativo para que estudantes, professores, engenheiros e equipes utilizem a ferramenta, pois, o PSP acaba não sendo adotado nas práticas diárias devido sua grande demanda de tempo, e com essa ferramenta esse problema é minimizado, além de suas métricas e dados serem de fácil acesso e visualização, gerando gráficos e exportando os dados para outras ferramentas.

## Referências Bibliográficas

[1] PRESSMAN, Roger S. **Engenharia de Software: Uma Abordagem Profissional**. São Paulo: Editora McGraw-Hill, 2011.

[2] HUMPHREY, Watts. **PSP: A Self-Improvement Process for Software Engineers**. Editora Addison-Wesley, 2005.

[3] <http://www.processdash.com/>, acessado em 12/08/2013.