**Sistema de Distribuição de Disciplinas - UFG**

**Plano de Gerência de Configuração**

**versão** 1.1.01.2.0

**Histórico da Revisão**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 21/10/2015 | 1.0.0 | Criação do Plano de Gerência de Configuração | Weslley Martins Araujo |
| 15/11/2015 | 1.1.0 | Novas pastas e novos arquivos | Pedro Teles |
| 16/11/2015 | 1.2.0 | Gerenciamento de novos artefatos e definição de branchs | Weslley Martins Araujo |

1. **INTRODUÇÃO**

**1.1.Finalidade**

Este documento possui a finalidade de gerenciar e controlar os artefatos de software gerados no projeto Sistema de Distribuição de Disciplinas UFG definindo os meios utilizados para manter a integridade, histórico de alterações e controlar as mudanças. Assim, busca oferecer uma estruturação base para o repositório padrão do projeto de modo que quaisquer documentos e artefatos gerados durante as variadas fases do ciclo de vida possam ser controlados a mantidos bem como possíveis alterações e solicitações de mudança em quaisquer dos documentos controlados.

**1.2.Escopo**

Este Plano de Gerência de Configuração é destinado para todos os integrantes da equipe responsáveis pelo desenvolvimento do projeto Sistema de Distribuição de Disciplinas UFG, e abrange todo o controle e gerenciamento da configuração do projeto.

**1.3.Definições, Acrônimos e Abreviações**

SDD-UFG - Sistema de Distribuição de Disciplinas UFG

UFG – Universidade Federal de Goiás

GC - Gerência de Configuração

IC - Item de Configuração

**1.4.Visão Geral**

O contéudo deste documento está dividido em três seções principais: Identificação dos itens de configuração; Controle de todas as mudanças ocorridas no projeto; Estimativa dos itens adicionais à gerência de configuração.

1. **ORGANIZAÇÃO E FERRAMENTAS**

**2.1.Organização, Responsabilidades e Interfaces**

Os responsáveis pela execução das diversas atividades de gerenciamento de configuração do projeto serão os alunos Weslley Araujo e Pedro Teles, que tem por atribuições principais o estabelecimento de políticas e diretrizes de GC, a escrita e acompanhamento do Plano de Gerência de Configuração, a configuração do ambiente de GC, a criação de espaços de trabalho e a criação de baselines.

Não obstante, todos os envolvidos no desenvolvimento do projeto SDD-UFG devem seguir os padrões estabelecidos e dispostos neste documento, de modo que o projeto e todos os seus processos estejam, de fato, sob o controle da GC.

**2.2.Ferramentas, Ambiente e Infra-estrutura**

Para que a Gerência de Configuração, como um todo, possua resultados satisfatórios, a equipe utilizará a ferramenta GitHub.

1. **GERENCIAMENTO DA CONFIGURAÇÃO**

**3.1. Identificação do repositório da configuração**

Todos os artefatos resultantes durante a execução do projeto de construção e desenvolvimento do SDD-UFG serão armazenados em nuvem (github.com) no Github, no endereço: <https://github.com/weslleyma/sdd-ufg> sendo utilizado por toda a equipe do projeto e gerenciado pelo gerente de configuração.

**3.2. Critérios para escolha dos itens de configuração**

A Tabela 1 mostra os produtos de trabalho que são considerados itens de configuração (IC) no SDD-UFG. Os critérios utilizados para seleção desses IC para compor a configuração do projeto são:

1) importância para o planejamento do projeto;

2) importância para o controle de mudanças no projeto;

3) necessidade de retorno a estados anteriores do produto de trabalho;

4) necessidade de entrega do produto de trabalho como resultado do projeto.

**3.3. Itens de configuração**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sigla** | **IC** | **Tipo** | **Propósito** | **Controle** |
| ppr | Plano de Projeto | Doc. Texto | Apresentar o planejamento do projeto | Alto |
| gcs | Plano de Configuração | Doc. Texto | Definição da gerência de configuração do projeto | Alto |
| eor | Especificação de Objetivos e Requisitos | Doc. Texto | Definição das informações necessárias sobre as requisições do cliente e que o software deve atender. | Alto |
| ris | Plano de Riscos | Doc. Texto | Definição de riscos e sua probabilidade de ocorrência, impacto e ação. | Alto |
| req | Requisitos | Doc. Texto, PDF ou JPG | Registro dos Requisitos do Projeto, incluindo: Critérios de aceitação, Dados de casos de uso, Especificação de casos de uso, Histórias de usuário, Protótipos e Documentação de Requisitos | Alto |
| pte | Plano de Teste | Doc. Texto | Plano de Teste do Projeto | Alto |
| cat | Cronograma de Atividades | Doc. Texto | Cronograma de Atividades do Projeto | Alto |
| rhm | Recursos Humanos e Materiais | Doc. Texto | Documento de Recursos Humanos e Materiais | Alto |
| eap | EAP | Doc. Texto, PDF ou JPG | Documento de EAP do Projeto | Alto |
| ppj | Políticas do Projeto | Doc. Texto | Documento de Políticas do Projeto | Alto |
| arq | Arquitetura | Doc. Texto, PDF ou JPG | Documentação da arquitetura do Projeto | Alto |
| meb | Método de Estimativas Back-end | Planilha | Método de estimavas Back-end | Médio |
| mec | Método de Estimativas  Gerência de configuração (GCO) | Planilha | Método de estimavas Gerência de Configuração | Médio |
| mep | Método de Estimativas  Gerência de Projetos | Planilha | Método de estimavas Gerência de Projetos | Médio |
| met | Método de Estimativas  Teste | Planilha | Método de estimavas Teste | Médio |
| mon | Monitoramento | Planilha | Registrar o monitoramento do repositório | Médio |
| ret | Relatório de Execução de Testes | Doc. Texto | Relatórios e Casos de Testes utilizados no software, registros de não conformidades. | Médio |
| rap | Relatório de Acompanhamento de Projeto | Doc. Texto | Registrar o monitoramento periódico da execução do projeto. | Médio |
| ckb | Checklists de Auditoria em Baseline | Planilha ou Doc. Texto | Registrar a avaliação física e funcional das baselines do projeto, registrando não conformidades. | Médio |
| ckq | Checklists de Qualidade de Processo e Produto | Planilha ou Doc. Texto | Registrar as revisões de qualidade da execução de processos e produtos, registrando não conformidades. | Médio |
| med | Medição do Projeto | Planilha | Registrar a coleta de dados de medição e análise dos indicadores do projeto | Médio |
| fob | Código Fonte do Back-end do Projeto | Texto | Códigos Fonte do Back-end do Projeto em Grails | Médio |
| fof | Código Fonte do Front-end do Projeto | Texto | Códigos Fonte do Front-end do Projeto em HTML/CSS e AngularJs | Médio |
| man | Manuais | Doc. Texto | Manuais de instalação, configuração e utilização das funcionalidades básicas do software | Baixo |

Tabela 1

**3.4. Controle de Mudança em Baseline**

Uma vez aprovados e colocados em baseline, os IC ficam sujeitos ao controle de mudança da GC, de acordo com seu respectivo nível de controle (vide campo “Controle” na Tabela 1), que pode ser:

● **Alto**: O IC só é alterado após aprovação de solicitação de mudança pelo Gerente de Projetos e pelo Gerente de Configuração. O gerente de configuração deve analisar a solicitação de mudança e retornar seu impacto na configuração.

● **Médio**: O IC só é alterado após aprovação de solicitação de mudança pelo Gerente de Configuração.

● **Baixo**: O IC só é alterado após comunicação de solicitação de mudança ao Gerente de Configuração. A comunicação deve registrar claramente a modificação que será efetuada e o respectivo impacto na configuração.

Mudanças em IC que envolvam exclusivamente correções de ortografia e gramática no texto do documento devem ser tratadas com nível de controle baixo, independentemente do nível especificado para o documento no campo “Controle” da Tabela 1. A auditoria de baseline deve verificar que a alteração realizada envolveu, de fato, apenas correções ortográficas e gramaticais.

**3.5. Nomenclatura IC**

Cada documento correspondente a um IC e deve possuir um nome único. A formação do nome do documento deverá seguir o padrão abaixo:

**<item\_de\_configuração>\_<versao>\_<projeto>**

Onde:

**<item\_de\_configuração>**: Nome do item de configuração descrito na coluna “Item de Configuração” das tabelas 1 e 2 deste documento.

**<versao>**: Numero da versão atual do artefato.

**<projeto>**: Sigla do nome do projeto.

Exemplo: plano\_de\_projeto\_1.0.0\_SDD-UFG.

Todo IC terá como primeira versão 1.0.0. Quando o IC for alterado para correções, o digito mais a direita deverá ser incrementado.

Exemplo:

* Antes da alteração: plano\_de\_projeto\_1.0.0\_SDD-UFG
* Depois da alteração: plano\_de\_projeto\_1.0.1\_SDD-UFG

Quando o IC for alterado de modo que seu conteúdo reflita algo novo no projeto, o segundo dígito deverá ser incrementado.

Exemplo:

* Antes da alteração: plano\_de\_projeto\_1.0.0\_SDD-UFG
* Depois da alteração: plano\_de\_projeto\_1.1.0\_SDD-UFG

**3.5.1. Nomenclatura de IC de codificação**

Os ICs de codificação deverão seguir os padrões determinados pela equipe de desenvolvimento.

**3.5.2. Nomenclatura de outros artefatos**

Itens de configuração criados rotineiramente, como resultados de execução de casos de testes, planilhas de checklist da qualidade e monitoramento do respositório deverão ser nomeado seguindo o padrão mencionado acima, acrescentado um "date-time" (ano, mês, dia, hora e minutos) no final, como mostrado no exemplo abaixo.

**<item\_de\_configuração>\_<versao>\_<projeto>\_<datetime>**

Exemplo: caso\_teste\_1.0.0\_SDD-UFG\_201510232010

Lembrando que a versão somente será incrementada se o template do documento for alterado. Caso o template seja o mesmo, a versão será sempre 1.0.0, nesse caso a diferenciação ocorrerá pela data/hora no nome do documento.

**3.5.3. Criação de Baselines**

As baselines serão criadas de acordo com o cronograma do Plano de Projeto do Sistema de Distribuição de Disciplinas da UFG.

**3.6. Estrutura de Armazenamento os ICs do Projeto**

Dentro do repositório do github citado no item 3.1 deste documento, há duas pastas principais, são elas: Code-DD-UFG e Doc-DD-UFG.

A primeira pasta será utilizada para os códigos fonte do proejto, a segunda para os demais artefatos, como gerenciamento, requisitos, testes, etc.

Segue abaixo a disposição das pastas no repositório e a respectiva localização dos itens:

|  |  |
| --- | --- |
| **Pasta (Diretório)** | **IC** |
| Code-DD-UFG/Back-End | fob |
| Code-DD-UFG/Front-End | fof |
| Doc-DD-UFG/Arquitetura | arq |
| Doc-DD-UFG/Arquitetura/Definições | arq |
| Doc-DD-UFG/Arquitetura/Protótipos | arq |
| Doc-DD-UFG/Arquitetura/arquitetura | arq |
| Doc-DD-UFG/Configuração | gcs, ckb |
| Doc-DD-UFG/Configuração/Monitoramento | mon |
| Doc-DD-UFG/Gerência | ppr, ris, eor, rhm, eap, ppj |
| Doc-DD-UFG/Gerência/Relatórios | rap |
| Doc-DD-UFG/Manual | man |
| Doc-DD-UFG/Medição | med |
| Doc-DD-UFG/Método de Estimativas | meb, mec, mep, met |
| Doc-DD-UFG/Qualidade | ckq |
| Doc-DD-UFG/Requisitos | eor, req |
| Doc-DD-UFG/Templates | ckq |
| Doc-DD-UFG/Teste | ret, pte |

Tabela 2

**3.7. Criação de Branchs**

Dentro do repositório do github citado no item 3.1 deste documento, há duas pastas principais, são elas: Code-DD-UFG e Doc-DD-UFG.

Os artefatos que envolver a parte de Code-DD-UFG deverá ser criado um “branch” obrigatoriamente. Ou seja, deve ser criado um “branch” para Arquitetura, outro para Front-End e outro para Back-End.

Os artefatos que envolver a parte de documentação que está em Doc-DD-UFG, pode ser feito “commit” diretamente no “master”, visto que estes artefatos devem ser propagados para todos imediatamente.

Assim, é da responsabilidade de cada integrante do projeto fazer o “commit” dos artefatos referentes às suas tarefas, independente de ser em “branchs” ou no “master”. Lembrando que, todo “commit” deve ter o id da tarefa programada no “Tuleap” e sua descrição.

**3.8. Armazenamento de Mídia e Liberação do Projeto**

Todos os itens de gerência de configuração deverão ser mantidos no repositório do projeto no GitHub, já que a ferramenta fornece a opção de recuperar todas as alterações e/ou remoções de arquivos.

**3.9. Permissões de Acesso**

Por se tratar de um projeto acadêmico, toda a equipe do projeto terá acesso as pastas e arquivos, porém apenas os GC poderá criar e modificar baselines.

**3.10. Treinamentos**

Os membros da equipe deverão, caso não possuam familiaridade, estudar tutoriais de utilização da ferramenta GitHub para que todos possuam a habilidade necessária para manter e lidar com as diversas atividades de GC providas pela ferramenta.