**Plano de Gerenciamento de Configuração – Byte INC.**

1. **Introdução**
   1. **Finalidade**

A finalidade deste documento é apresentar o Plano de Gerenciamento de Configuração a ser seguido no projeto EveRemind para controlar sistematicamente alterações na configuração e manutenção da integridade e rastreabilidade de todo o ciclo de vida do sistema.

* 1. **Escopo**

O escopo deste plano afeta todos os artefatos a serem controlados, como documentação, código fonte e todos os demais artefatos gerados durante a execução deste projeto, inclusive este documento. O modelo deste documento segue com uma introdução, apresentação da gerencia de configuração, apresentação do programa de gerenciamento, e por fim as definições de marcos.

* 1. **Definições**

EVE – Abreviação de EveRemind

* 1. **Visão Geral**

A seguir, este documento irá dar informações precisas de quais ferramentas serão usadas para o versionamento, a infraestrutura do repositório, o ambiente e a responsabilidade da equipe.

1. **Gerenciamento de Configuração de Software**
   1. **Responsabilidades**

|  |  |  |
| --- | --- | --- |
| Papel | Responsabilidade | RH |
| Gestor de Configuração | Define e redefine a estrutura do repositório de versionamento, faz auditorias e delega responsabilidades para a Equipe de Configuração. | Igor Moura Brandão  Leonardo Freitas dos Santos |
| Equipe de Configuração | Ajuda na realização das auditorias e realiza atividades delegadas pelo Gestor de Configuração. | Breno Fernandes Daniel de Carvalho Johnnathan Gomes Lima Matheus Henrique Lima Moisés Hilário Rodrigues Vinicius Carvalho Machado |
| Desenvolvedor | Produz os itens de configuração que serão gerenciados. | Igor Moura Brandão  Leonardo Freitas dos Santos  Moisés Hilário Rodrigues |

* 1. **Ferramentas, Ambiente e Infraestrutura**

A ferramenta de versionamento usada para este projeto será o GitHub. O repositório (o qual este documento se situa) criado para este projeto pode ser encontrado neste [link](https://github.com/leonardo-freitas-1995/P.I.-ES-UFG-2015-BIJLMMV) do site do GitHub. A infraestrutura do repositório foi criada de modo a separar documentos de processo geral dos documentos de processo aplicados neste projeto, e em cada caso, existe uma separação entre disciplinas para a fácil localização de um artefato na hora da avaliação do trabalho. A estrutura detalhada do repositório pode ser encontrada na [Definição Estrutural do Repositório](../../../../Definição%20Estrutural%20do%20Repositório.docx).

1. **Programa de Gerenciamento de Configuração**
   1. **Identificação de Configuração**
      1. **Métodos**

Todos os artefatos no repositório devem seguir as seguintes convenções:

* Todo documento que ainda se encontra em construção deve ser identificado com o prefixo [Parcial].
* Caso um mesmo documento seja usado por diversas disciplinas, este documento deve constar na pasta da disciplina que faz a sua criação. Exemplo: As disciplinas de Qualidade de Software, Arquitetura de Software, Gerência de Projeto e Gerência de Requisitos fazem referência ao Documento de Especificação de Objetivos e Requisitos (EOR), porem sua criação é feita na disciplina de Gerência de Requisitos, então é nela que o documento deve ser armazenado.
* Um documento só pode ser referenciado em outro quando este for considerado finalizado ou satisfatoriamente preenchido, a modo de não ser mais considerado parcial.
* O código fonte deve ser armazenado exclusivamente dentro de ‘Artefatos de Código Fonte/EveRemind’. Todo código fonte armazenado fora desta pasta não deve ser considerado e deve ser excluído.
  + 1. **Baselines**

As baselines deverão ser criadas no fim de cada iteração especificada no plano de projeto e no início da execução do plano de projeto. Após o lançamento da baseline, sua edição só poderá ser efetuada com a aprovação do Gestor de Configuração e do Gerente de Projeto.

* 1. **Estimativa do Status de Configuração**
     1. **Processo de Armazenamento de Mídia e Liberação do Projeto**

O armazenamento da configuração é feito pela própria ferramenta de versionamento GitHub. Os dados ficam armazenados na nuvem e podem ser visualizados por qualquer um com acesso ao link do projeto no GitHub (A alteração por outro lado fica restrita às contas adicionadas como colaboradoras no projeto). A release será efetuada quando for lançada uma linha de base que possua todas as funcionalidades especificadas no EOR.

* + 1. **Auditorias**

As auditorias realizadas pela Gerência de Configuração serão as auditorias físicas. Elas podem ser feitas em qualquer momento durante o desenvolvimento do software, e avaliam se o repositório e os itens de configuração foram devidamente criados e mantidos, seguindo os padrões especificados neste documento. Após uma auditoria realizada, deve ser criado e devidamente armazenado um Relatório de Auditoria Física.

1. **Marcos**

Os marcos de projeto são gerados sempre no início e fim de uma etapa importante, bem como nos momentos de release. Alguns marcos importantes do projeto são:

* Início da execução do plano de projeto
* Final de uma iteração de desenvolvimento
* Lançamento de uma versão
* Pós-implementação de uma modificação