**PLANO DE GERENCIAMENTO DE CONFIGURAÇÃO**

**SISTEMA DE PATRIMÔNIO**

SUMÁRIO

[1. INTRODUÇÃO 3](#_Toc504519626)

[1.1. Objetivos 3](#_Toc504519627)

[1.2. Evolução 3](#_Toc504519628)

[2. Gerência de Configuração de Software 4](#_Toc504519629)

[2.1. Ferramentas 4](#_Toc504519630)

[2.2. Ambientes 4](#_Toc504519631)

[3. o programa de gerenciamento de configuração 5](#_Toc504519632)

[3.1. Estrutura do Repositório 5](#_Toc504519633)

[4. Implantação 5](#_Toc504519634)

[4.1. Implantação 5](#_Toc504519635)

[4.2. Dicas Uteis 5](#_Toc504519636)

# INTRODUÇÃO

O Plano de Gerenciamento de Configuração descreve todas as atividades do Gerenciamento de Controle de Configuração e Mudança que serão executadas durante o ciclo de vida do produto. Suas atividades envolvem identificar a configuração do software, manter sua integridade durante o projeto e controlar sistematicamente as mudanças.

## Objetivos

O objetivo deste documento é criar um padrão a ser seguido por todos os membros da equipe com o intuito de garantir o maior controle do produto no decorrer do projeto. Para que isso aconteça serão detalhados os recursos necessários (equipes, ferramentas e ambiente), as responsabilidades atribuídas e o cronograma de atividades.

## Evolução

O Plano de Gerenciamento de Configuração deve ser mantido atualizado para refletir o planejamento corrente. Dessa forma, as seguintes situações representam gatilhos para atualização do plano e nova aprovação deste documento:

* Mudança nos itens de configuração;
* Mudança na identificação dos arquivos;
* Mudança na identificação das *Tags*/*Branches*;
* Mudança no padrão de versionamento;

# Gerência de Configuração de Software

### Ferramentas

|  |  |  |
| --- | --- | --- |
| Termo | Versão | Descrição |
| *GitHub* | 0.0.0 | Ferramenta de administração dos repositórios, serviço disponibilizado no endereço: |
| Laravel Framework | 5.4.32 | Framework de Desenvolvimento, serviço disponibilizado no endereço: <https://laravel.com> ou composer create-project --prefer-dist laravel/laravel nome-do-projeto |
| Composer | 1.5.1 | Gerenciador de Dependencias, serviço disponibilizado no endereço: <https://getcomposer.org/download/> |
| *PHP* | 7.0 | Linguagem de Programação Utilizada, serviço disponibilizado no endereço: [http://](http://sirius.saude.gov.br/)php.net |
| Netbeans | 8.2.1 | IDE de desenvolvimento, download disponibilizado no endereço: <http://download.netbeans.org/netbeans/8.2/final/bundles/netbeans-8.2-windows.exe> ou $ Sudo su -  #wget <http://download.netbeans.org/netbeans/8.2/final/bundles/netbeans-8.2-linux.sh>  # Chmod + x netbeans-8.0.2-php-linux.sh  # sh netbeans-8.0.2-php-linux.sh |
| *Git* | 2.14.1 | Ferramenta de controle de versão, download disponibilizado no endereço: <https://git-scm.com/download/win> ou $ apt-get install git |
| *MySqlWorkbench* | 6.3.1 | Ferramenta de Modelagem do Banco, download disponibilizado no endereço: <https://dev.mysql.com/downloads/workbench/> |
| *XAMMP* | 5.6.30 | Ferramenta Pacote Com Servidor Apache, PHP e SGBD Mysql para Desenvolvimento no Windows, download disponibilizado no endereço: <https://www.apachefriends.org> |

### Ambientes

O ambiente que será entregue a equipe de desenvolvimento, deverá ser mantido pela equipe de arquitetura, através de *Virtual* *Machines* que seguiram os padrões dos ambientes mantidos pela equipe de infraestrutura. As ferramentas de desenvolvimento “*IDEs*” serão de livre escolha do desenvolvedor, desde que a mesma seja uma ferramenta de Software Livre, tais como *Atom*, *Eclipse*, mas de preferência Netbeans...

# o programa de gerenciamento de configuração

### Estrutura do Repositório

O repositório do projeto está armazenado no GitHub, podendo ser acessado pelo link: https://github.com/fgpacheco/Projeto\_APS.

# Implantação

## Implantação

* 1. **Clone o projeto**

git clone https://github.com/FuturoDaUAG/ProjetoPW.git

* 1. **Acesse o projeto**

cd seuprojeto

* 1. **Instale as dependências e o framework**

composer update

composer install --no-scripts

* 1. **Copie o arquivo .env.example**

cp .env.example .env

* 1. **Crie uma nova chave para a aplicação**

php artisan key:generate

* 1. **Em seguida você deve configurar o arquivo .env e rodar as migrations com:**

php artisan migrate –seed

## Dicas Uteis

Ao apagar Migrations e Models de modo manual e tentar cria-los novamente:

* 1. Delete manualmente migration e model
  2. Resetar arquivos autoload do composer com o comando: composer dump-autoload