|  |
| --- |
| INSTITUTO DE INFORMÁTICA – UFG |
| Plano de Projeto  YGG |
| Versão 1.5 |
|  |
| **Goiânia, maio de 2016.** |
|  |

|  |
| --- |
|  |

**Colaboradores:**

|  |  |
| --- | --- |
| **Aluno** | **Matrícula** |
| Gabriel Barbosa | 201401977 |
| João Henrique | 201401985 |
| Lucas Vinicios | 201401987 |
| Pedro Victor | 201401989 |
| Raphael Ferreira | 201400171 |
| Rony Nogueira | 201406318 |

Controle de Alterações

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Autor** | **Versão** | **Descrição da Mudança** |
| 09/05/16 | Pedro Victor | 1.0 | Criação do Documento de PGPS, template definido para as informações que deverão constar no Documento posteriormente; |
| 09/05/16 | Pedro Victor | 1.1 | Finalização do Plano de Gerência e Configuração e Controle de Mudança; |
| 09/05/16 | Rony Nogueira | 1.2 | Preenchimento das tabelas relacionadas aos Planos de Ambiente, Qualidade e Comunicação; |
| 10/05/16 | Pedro Victor | 1.3 | Finalização do tópico referente a organização da equipe (Tópico 2); |
| 11/05/16 | Pedro Victor | 1.4 | Complementação do Tópico relacionado à Gerência de Configuração. |
| 11/05/16 | Pedro Victor | 1.5 | Finalização do Plano até o tópico 8. Revisão e adição de legendas nas tabelas. |

Sumário

[1. Introdução 3](#_Toc450725629)

[2. Organização e Responsabilidades Gerais 3](#_Toc450725630)

[3. Plano de Gerência de Configuração 4](#_Toc450725631)

[3.1 Política de Configuração 4](#_Toc450725632)

[3.1.1 Itens de configuração 4](#_Toc450725633)

[3.1.2 Método de Identificação e Numeração das Versões 4](#_Toc450725634)

[3.1.3 Procedimentos para Integração Contínua 6](#_Toc450725635)

[3.2 Organização e Responsabilidades 7](#_Toc450725636)

[3.3 Atividades e Cronograma Macro 7](#_Toc450725637)

[4. Plano de Ferramentas e Ambiente 7](#_Toc450725638)

[4.1 Ambientes 8](#_Toc450725639)

[4.1.1 Ambiente de Laboratório 8](#_Toc450725640)

[4.1.2 Ambiente de Testes 10](#_Toc450725641)

[4.2 Organização e Responsabilidades 10](#_Toc450725642)

[4.3 Atividades e Cronograma Macro 10](#_Toc450725643)

[5. Plano de Gerência de Riscos 11](#_Toc450725644)

[6. Plano de Qualidade 11](#_Toc450725645)

[6.1 Métricas de Qualidade 12](#_Toc450725646)

[6.2 Organização e Responsabilidades 12](#_Toc450725647)

[6.3 Atividades e Cronograma Macro 13](#_Toc450725648)

[7. Controle de Mudanças 13](#_Toc450725649)

[8. Plano de Comunicação 13](#_Toc450725650)

[8.1 Organização e Responsabilidades 14](#_Toc450725651)

[8.2 Atividades e Cronograma Macro 14](#_Toc450725652)

[9. Plano de Implementação 14](#_Toc450725653)

[9.1 Organização e Responsabilidades 14](#_Toc450725654)

[9.2 Estratégia e Cronograma Macro 15](#_Toc450725655)

[9.2.1 Detalhamento do Release interno 1 15](#_Toc450725656)

[9.2.2 Detalhamento do Release interno 2 16](#_Toc450725657)

[9.2.3 Detalhamento do Release interno 3 16](#_Toc450725658)

[9.2.4 Detalhamento do Release interno 4 16](#_Toc450725659)

[10. Plano de Testes 17](#_Toc450725660)

[10.1 Organização e Responsabilidade 17](#_Toc450725661)

[10.2 Requisitos e Cronograma Macro 17](#_Toc450725662)

[10.2.1 Etapa 1 17](#_Toc450725663)

[10.3 Registro dos Resultados 19](#_Toc450725664)

[11. Plano de Aceitação 19](#_Toc450725665)

[11.1 Organização e Responsabilidade pela Aceitação 19](#_Toc450725666)

[11.2 Cronograma Macro 20](#_Toc450725667)

[12. Plano de Implantação 20](#_Toc450725668)

[13. Plano de Equipe 20](#_Toc450725669)

[14. Cronograma Geral 20](#_Toc450725670)

[15. Referências 21](#_Toc450725671)

[16. Assinaturas 21](#_Toc450725672)

# Introdução

Este documento define o Plano de Projeto para o projeto **YGG** com o objetivo de integrar e apresentar conjuntamente os demais planos, facilitando o entendimento geral do mesmo e definindo o planejamento para a realização do trabalho.

As demais seções apresentam o planejamento e definições para o gerenciamento das diversas áreas do projeto: Equipe, Comunicação, Desenvolvimento, Testes, Distribuição, Gerência de Configuração, Ambiente, Riscos, Aceitação, Qualidade, Controle de mudanças, Cronograma e Custo.

# Organização e Responsabilidades Gerais

O nosso projeto será dividido em três grandes áreas, duas de desenvolvimento e uma de documentação. Abaixo podemos ter uma noção dessas áreas, e o respectivo colaborador responsável por cada uma:

|  |  |
| --- | --- |
| **Área** | **Responsável** |
| Front-end | Gabriel Barbosa |
| Back-end | Rony Nogueira |
| Documentação | Pedro Victor |

Tabela 1 – Divisão do Projeto.

A seguir visualizamos mais detalhadamente qual atividade cada colaborador irá desempenhar ao longo do desenvolvimento YGG.

|  |  |  |
| --- | --- | --- |
| **Responsável** | **Tarefas** | **Função** |
| Gabriel Barbosa | Desenvolver Front-end | Coordenador de Desenvolvimento (Front-end) |
| Guilherme Caixeta | Apoio na Gerência de Configuração | Gerente Auxiliar |
| João Henrique | Desenvolver Front-end | Desenvolvedor |
| Lucas Vinicios | Desenvolver Back-end | Desenvolvedor |
| Pedro Victor | Gerenciar a Configuração/Desenvolver Back-end | Gerente/Desenvolvedor |
| Raphael Ferreira | Desenvolver Front-end | Desenvolvedor |
| Rony Nogueira | Gerenciar o Projeto/Desenvolver Back-end | Gerente/Coordenador de Desenvolvimento (Back-end) |

Tabela 2 – Atividades distribuídas.

# Plano de Gerência de Configuração

Essa seção incorpora ao Plano de Projeto o Plano de Gerência de Configuração, que documenta as atividades e padrões de gerência de configuração a serem adotados durante o projeto.

## Política de Configuração

Nessa seção, são definidos os itens de configuração sob a responsabilidade da gerência de configuração e a nomenclatura para os identificadores dos itens. São apresentados, também a política para numeração de versões e para os *baselines* do projeto.

### Itens de configuração

A tabela abaixo identifica os itens que estarão sob a gerência de configuração, apresentando uma breve descrição de cada um.

|  |  |
| --- | --- |
| **Item de Configuração** | **Descrição** |
| Programas | Representam o código fonte responsável pela aplicação. |
| Configuração | Representam todos os arquivos de configuração para que a aplicação seja executada. |
| Artefato metodologia | Representam todos os documentos e arquivos gerados pelo projeto. |

Tabela 3 – Itens de Configuração.

### Método de Identificação e Numeração das Versões

Todos os itens de configuração (por exemplo, documentos, relatórios e tutoriais), exceto código fonte, devem ser identificados por quem o produziu, com a seguinte nomenclatura:

<SIGLA\_PROJETO>\_<ID\_ARTEFATO>\_<VERSÃO>\_<DATA>

Sendo que:

* <SIGLA\_PROJETO> é a sigla do projeto, no caso estamos desenvolvendo o sistema web YGG (nome originado da árvore da mitologia Nórdica, Yggdrasil, a árvore dos mundos), então a sigla definida previamente é “YGG”;
* <ID\_ARTEFATO> é a identificação do assunto no artefato (ver Tabela 2);
* <VERSÃO> é o número da versão do documento, apenas se for a primeira versão não é necessário escrever no documento.
  + É definido que o número da versão tem o seguinte padrão: X.Y
    - X sendo o número inteiro, representando a versão final do artefato, variando entre 1 e infinito;
    - Y sendo o número decimal que representa uma simples modificação da versão do artefato, variando entre 0 e 9;
  + A primeira versão do artefato será 1.0 e pode ser ocultada da nomenclatura;
  + A partir da 1.1, a versão deve ser informada no nome do artefato.
* <DATA> é a data de criação do artefato, da primeira versão. O formato que a descreve é AAAAMMDD, onde A (ano), M (mês) e D (dia). Um exemplo seria 20160509, correspondente ao dia de hoje (09 de maio de 2016).

**Observação:** Todas as letras dos nomes dos artefatos devem estar escritas em caixa alta.

A Tabela 4 apresenta os artefatos que podem ser gerados ao longo do processo de desenvolvimento, e cada um contém sua respectiva identificação.

|  |  |
| --- | --- |
| **Artefato** | **Identificação** |
| Ata de Reunião | ATA |
| Backlog | BL |
| Banco de Dados | BD |
| Caso de Teste | TST |
| Caso de Uso | UC |
| Definições | DEFN |
| Documento de Arquitetura | ARQ |
| Documento de Requisitos | REQ |
| Pedido Formal de Mudança | PFM |
| Plano de Gerência de Projeto | PGPR |
| Plano de Gerência de Requisitos | PGRE |
| Plano de Gerência de Configuração | PGCS |
| Plano da Garantia de Qualidade | PGQA |
| Relatório de Avaliação do Cliente | RAC |
| Representational State Transfer | REST |
| Reunião | REU |
| User Stories | US |

Tabela 4 – Identificação de Artefatos.

❖ Depois de nomear os artefatos, temos que armazená-los. Basicamente, temos no GitHub as pastas, que se relacionam com .

A Tabela 5 é referente aos Macros Áreas que compõem o nosso repositório (YGG) no GitHub:

|  |  |
| --- | --- |
| **Itens** | **Identificação de Pasta** |
| Todos os Códigos produzidos no Projeto | Código\_Fonte |
| Todos os Documentos produzidos no Projeto | Documentos |

Tabela 5 – Macro Pastas.

❖ Dentro de cada Pasta, relacionada acima, os artefatos ficam organizados, além de serem, além de termos novas pastas ligadas às Áreas de Processo que tenham mais de um artefato relacionado. Ao longo do desenvolvimento, diretórios e pastas serão criados, mas sempre havendo uma correlação entre o artefato, pasta, diretório e Macro Área ( Documentação e Codificação).

* A nomenclatura das pastas terá que seguir o seguinte padrão:

<NOME>\_<SOBRENOME>

Sendo que:

* <NOME> é o nome que identifica os artefatos que vão compor a pasta;
* <SOBRENOME> é o segundo nome a fim de identificar a macro área, mostrando o caminho que terá que se seguir para encontrar determinado artefato.

**Observação:** Apenas a primeira letra dos nomes das pastas deve estar escrita em caixa alta.

* O comentário para subir arquivos ao repositório terá que dar uma visão geral, resumidamente em apenas uma frase, a respeito do conteúdo que foi inserido pelo(s) artefato(s) colocado(s) no repositório, transformado em um item de Configuração.

### Procedimentos para Integração Contínua

Essa seção define, essencialmente, como se dará o processo de construção de builds no projeto. Um build é uma versão operacional da aplicação que demonstra um subconjunto dos requisitos da aplicação final a ser desenvolvida, não constituindo necessariamente um release interno. Costuma-se dizer que a geração de builds atua como um medidor do batimento cardíaco do projeto.

Será utilizado no projeto o conceito de builds periódicos, onde a demanda de builds será definida, pela necessidade da realização de testes de unidade durante o desenvolvimento.

Após a liberção do build, deve-se automaticamente verificar os erros e se todos os itens especificados no documento de build foram incluídos corretamente.

A princípio todos participantes do projeto podem realizar as operações (check-in, check-out, update, merge, lock, unlock e tag) sobre os itens de configuração.

## Organização e Responsabilidades

|  |  |  |
| --- | --- | --- |
| **Papéis** | **Equipe** | **Responsabilidade** |
| Gestor de Configuração | Pedro Victor | Estabelecer Políticas de GCS;  Escrever PGCS;  Configurar Ambiente de GCS;  Criar Espaços de Trabalho de Integração;  Criar Baselines;  Promover Baselines; |
| CCM | Guilherme | Estabelecer Processo de Controle de Mudança;  Revisar Solicitação de Mudança; |
| Desenvolvedor | Gabriel Barbosa  Lucas Vinicios  Raphael Ferreira  Rony Nogueira  João Henrique | Seguir os padrões e procedimentos definidos no PGCS; |

Tabela 6 – Organização da Equipe.

## Atividades e Cronograma Macro

|  |  |  |
| --- | --- | --- |
| **Atividade** | **Responsabilidade** | **Data Término** |
| Liberação do Build 1.0 | Gabriel Barbosa | 13/05/2016 |
|  |  |  |
|  |  |  |

Tabela 7 – Cronograma Geral.

# Plano de Ferramentas e Ambiente

Essa seção define o Plano de Ambiente para o projeto, com o objetivo de descrever os recursos de hardware software disponível e detalhar os diferentes ambientes do projeto.

A Tabela 3 abaixo descreve as ferramentas que serão utilizadas no processo de desenvolvimento do Sistema YGG.

|  |  |
| --- | --- |
| **Ferramenta** | **Descrição** |
| Eclipse | IDE a ser utilizada para o desenvolvimento do Sistema. |
|  |  |
| Git v2.x.x | Ferramenta de Versionamento. |
| GitHub | Servidor de Versionamento. |
| GitKraken v1.2.0 | Ferramenta de suporte ao GitHub. |
| Hangout (Google) | Ferramenta de comunicação (Reuniões). |
| Word, Excel e Power Point | Ferramentas para edição de documentos, planilhas e apresentações. |
| Drive (Google) | Ferramenta para edição de documentos entre mais de uma pessoa ao mesmo tempo. |
| Apache Tomcat v9.0 | Servidor da aplicação. |
|  |  |
|  |  |
| Windows 10 | Sistema Operacional (Desenvolvimento). |
|  |  |
|  |  |

Tabela 8 – Ferramentas e Ambiente

## Ambientes

Um ambiente é uma configuração específica de hardware e software, estabelecida com um propósito pré-definido. Esta seção descreve os ambientes identificados para o projeto.

### Ambiente de Laboratório

Esse será o ambiente utilizado pela equipe de projeto (mais especificamente pelos desenvolvedores), com o objetivo de implementar a solução. Seguem na tabela abaixo as necessidades e configuração atual para o ambiente.

|  |  |  |  |
| --- | --- | --- | --- |
| **Hardware** | **Software** | **Finalidade** | **Usuário** |
| Nome Máquina: Lenovo G460  Configuração: Core i3 370M 4GB RAM  320GB Hard Disk | Windows 10  SublimeText Editor 2  Git | Desenv | Rony |
| Nome Máquina:...  Configuração: Core i7  8GB RAM  920GB Hard Disk | Windows 10  Atom/Brackets  Git | Desenv | Pedro |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Tabela 9 – Ambiente de Trabalho de Cada Membro.

De modo a poder *debugar* a aplicação, é necessário realizar a instalação do interpretador PHP, é recomendada o uso do XAMPP, por possuir todos os componentes necessários para a execução e depuração da aplicação. Para realizar a instalação os seguintes passos deverão ser executados:

**Windows**

1. Realize o download do instalador através do site <http://www.apachefriends.org/pt_br/xampp.html>
2. Siga os passos da instalação;
3. Vá ao menu iniciar procure por XAMPP Control Panel;
4. Realize a inicialização dos componentes Apache e MySQL;
5. Acesse o endereço <http://localhost/> ou <http://127.0.0.1/>

**Linux**

1. Realize o download do instalador através do site   
   <http://www.apachefriends.org/pt_br/xampp.html>
2. Abra terminal e vá até a pasta que foi salva o arquivo baixado;
3. Digite o seguinte comando **chmod 755 xampp-linux-version-installer.run**
4. Após digite o seguinte comando: .**/xampp-linux-version-installer.run**
5. Aparecerá o instalador, siga os passos e conclua a instalação;
6. Para iniciar o serviço, digite o seguinte comando no terminal **sudo /opt/lampp/lampp start**
7. Acesse o endereço <http://localhost/> ou <http://127.0.0.1/>

### Ambiente de Testes

Esse ambiente serve para testar a solução, de acordo com suas especificações.

|  |  |  |  |
| --- | --- | --- | --- |
| **Hardware** | **Software** | **Finalidade** | **Usuário** |
| Nome Máquina:  IP: | Linux Server | Execução dos  Testes | Equipe |

Tabela 10 – Máquina designada para Testes.

Os procedimentos para configuração das ferramentas do ambiente de testes, antes da execução dos mesmos, estarão no Release Notes de cada Release.

## Organização e Responsabilidades

|  |  |
| --- | --- |
| **Nome** | **Responsabilidade** |
| Pedro Victor | Coordenação do Ambiente |
| Rony Nogueira | Instalação de Softwares e disponibilização de treinamentos |

Tabela 11 – Organização do Ambiente.

## Atividades e Cronograma Macro

|  |  |  |
| --- | --- | --- |
| **Atividade** | **Responsabilidade** | **Data** |
| Instalação Ambiente de Laboratório | Rony Nogueira | 28/04/2016 |
| Instalação Ambiente de Testes | Rony Nogueira | 28/04/2016 |

Tabela 12 – Atividades sobre o Ambiente.

# Plano de Gerência de Riscos

Essa seção tem como objetivo definir como serão executadas algumas etapas do processo de gerenciamento dos riscos.

O processo de identificação dos riscos será feito continuamente durante toda a execução do projeto por toda a equipe. Tais riscos devem ser reportados diretamente para o gerente do projeto que deverá discuti-los e analisá-los nas reuniões de status. A freqüência das reuniões deverá ser determinada de acordo com a necessidade do projeto.

Para uma correta priorização dos riscos deve ser utilizada a seguinte escala a definição da probabilidade e Impacto

Probabilidade:

**[3]** Alta - [100% - 66,67%]

**[2]** Média - [66,66% - 33,33%]

**[1]** Baixa - [33,32% - 0%]

Impacto**:**

**[3]** Alto

**[2]** Médio

**[1]** Baixo

De uma maneira padrão, a priorização dos riscos deve ser feita de acordo com o cálculo da exposição do risco: exposição = probabilidade X impacto.

Para cada risco identificado, deve-se analisar a sua exposição, definindo o quão grave é o risco para o projeto.

A exposição, mais outras informações sobre os riscos devem estar documentados na planilha de riscos. A planilha de riscos deve conter no máximo 10 riscos os quais devem ser reportados no relatório de status.

Poderão ser adotadas as seguintes estratégias para o tratamento do risco identificado:

**Aceitar**: a equipe aceita conviver com o risco.

**Eliminar:** ocorre uma mudança no escopo do projeto de modo que o risco seja eliminado.

**Transferir:** transfere-se a responsabilidade do risco para alguém fora da equipe.

**Evitar:** elaboram-se estratégias de modo a tentar diminuir a probabilidade que o risco ocorra.

# Plano de Qualidade

Essa sessão define o Plano de Qualidade para o projeto, identificando como a qualidade da aplicação, dos artefatos e dos processos envolvidos no progresso da solução será garantida.

|  |  |  |
| --- | --- | --- |
| **Artefato** | **Fase em que será desenvolvido** | **Responsável** |
| Visão e Escopo | Visão | Raphael Ferreira |
| Plano de Projeto | Planejamento | Pedro Victor |
| Cronograma | Planejamento | Rony Nogueira |
| Lista de Riscos | Todas | Pedro Victor e Rony Nogueira |
| Especificação Funcional | Planejamento | Gabriel Barbosa |
| Glossário | Todas | Pedro Victor |
| Documento de Build | Desenvolvimento e Estabilização | Gabriel Barbosa |
| Relatório de Fechamento do Projeto | Implantação | Rony Nogueira |
| Status do Projeto | Planejamento, Desenvolvimento e Implantação | Rony Nogueira |

Tabela 13 – Responsáveis pela Qualidade.

## Métricas de Qualidade

As métricas servem para ajudar na medição da qualidade. A tabela abaixo apresenta as métricas que serão utilizadas no projeto.

|  |  |  |  |
| --- | --- | --- | --- |
| **Métrica** | **Possíveis valores** | **Interpretações** | **Dicas para medição e análise** |
| Não-conformidades para documentos | Qualquer número inteiro maior ou igual a zero | Qualquer valor **diferente de zero** indica a necessidade de reajustar o item inspecionado de modo a garantir sua qualidade | Avaliar o produto seguindo os itens obrigatórios dos templates contidos no YGG; |
| Não-conformidade para código fonte | Qualquer número inteiro maior ou igual a zero | Qualquer valor **maior ou igual a quatro**, indica a necessidade de reajustar o item inspecionado de modo a garantir sua qualidade. | Avaliar o produto seguindo o padrão de codificação contido no YGG |

Tabela 14 – Métricas de Qualidade.

## Organização e Responsabilidades

|  |  |
| --- | --- |
| **Nome** | **Responsabilidade** |
| João Henrique | Responsável final pela Qualidade |
| Rony Nogueira | Inspeções de Código |

Tabela 15 – Organização da Qualidade.

## Atividades e Cronograma Macro

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atividade** | **Tipo** | **Objetivo e Procedimentos** | **Responsável** | **Data** |
| Revisão da especificação funcional | Inspeção | Verificar se o código fonte atende a métrica “Não-conformidade para código fonte”. Deve ser escolhido pelo menos um código para cada programador. | Rony Nogueira | 12/05/2016 |
| Revisão de Código | Inspeção | Verificar se o código fonte atende a métrica “Não-conformidade para código fonte”. Deve ser escolhido pelo menos um código para cada programador. | Gabriel Barbosa | 12/05/2016 |

Tabela 16 – Atividade de Qualidade.

# Controle de Mudanças

Essa seção especifica todas as etapas do processo de mudanças no projeto (solicitação, aprovação, homologação, etc.). As mudanças devem seguir o seguinte ciclo:

Para a solicitação de uma mudança é necessário o registro do pedido na seção Issues do repositório do projeto, o pedido será analisado pelo responsável e mediante a aprovação e homologação do pedido, a mudança solicitada é enviada à equipe de desenvolvimento.

# Plano de Comunicação

Por principio todas as informações geradas pelo projeto devem ser limitadas apenas a equipe do projeto sem nenhuma restrição. Semanalmente, toda equipe deve se reportar ao gerente de projetos informando todo o trabalho realizado. Isso deve ser realizado no final de cada sexta-feira. De posse dessas informações, o gerente de projetos deve elaborar um Relatório de Status que deve ser comunicado a toda a equipe do projeto e mais alguns usuários listados na tabela seguinte:

|  |  |  |  |
| --- | --- | --- | --- |
| **Documento** | **Destino** | **Responsável** | **Freqüência** |
| Relatório de Status | Github  Realizar a adição do relatório semanal no repositório do projeto. | Rony Nogueira | Toda sexta-feira até as 22:00 |

Tabela 17 – Informação sobre a Comunicação.

## Organização e Responsabilidades

|  |  |
| --- | --- |
| **Nome** | **Responsabilidade** |
| Rony Nogueira | Comunicação Geral |

Tabela 18 – Organização da Comunicação.

## Atividades e Cronograma Macro

|  |  |  |
| --- | --- | --- |
| **Atividade** | **Responsabilidade** | **Data** |
| Elaboração Relatório de Status 1.0 | Rony Nogueira | 13/05/2016 |
| Elaboração Relatório de Status 2.0 | Rony Nogueira | 20/05/2016 |
| Elaboração Relatório de Status 3.0 | Rony Nogueira | 27/05/2016 |
| Elaboração Relatório de Status 4.0 | Rony Nogueira | 03/06/2016 |

Tabela 19 – Atividades da Comunicação.

# Plano de Implementação

Essa seção define o Plano de Desenvolvimento para o projeto, com o objetivo de registrar a estratégia de desenvolvimento a ser seguida em cada Release Interno. As atividades de desenvolvimento englobam três diferentes disciplinas: requisitos, análise & projeto e implementação. Essas atividades têm como objetivo, basicamente, o detalhamento, a análise, o projeto e a implementação dos casos de uso.

## Organização e Responsabilidades

|  |  |
| --- | --- |
| **Nome** | **Responsabilidades** |
| Gabriel Barbosa | Desenvolver interface de comunicação com o Servidor |
| Raphael Ferreira | Auxiliar no desenvolvimento da interface gráfica da Web site e comunicação com o Servidor |
| João Henrique | Auxiliar no Desenvolvimento de interface interação com o usuário |
| Rony Nogueira | Desenvolver interface de comunicação com o cliente |
| Lucas Vinicios | Auxiliar no desenvolvimento da aplicação servidora |
| Pedro Victor | Auxiliar no desenvolvimento da aplicação servidora |

Tabela 20 – Organização da Implementação.

## Estratégia e Cronograma Macro

A tabela abaixo apresenta a estratégia de desenvolvimento.

|  |  |  |  |
| --- | --- | --- | --- |
| **Release Interno** | **Data Término** | **Objetivo** | **Funcionalidade** |
| 0.1 | 12 /05/2016 | Implementar a tela de cadastro e login para que o usuário se cadastre e entre no sistema. | 1.1 |
| 0.2 | 13/05/2016 | Disponibilizar a opção de cadastro rápido, fornecida pelo Gmail ou Facebook. | 1.2 |
| 0.3 | 15/05/2016 | Usuário deverá ser capaz de criar um mapa mental e o sistema terá que fornecer a data e hora de criação, automaticamente. | 2 |
| 0.4 | 20/05/2016 |  | 1  2 3 |

### Detalhamento do Release interno 1

|  |  |  |
| --- | --- | --- |
| **Funcionalidade** | **Objetivos** | **Pendências** |
|  | Toda a interface gráfica para entrada de dados do usuário (métricas e aplicar e escopo) deve estar pronta. A interface deve passar para as demais camadas da aplicação os objetos que representam a seleção realizada pelo usuário. |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | Requisitos a serem observados durante todo o desenvolvimento. |  |

### Detalhamento do Release interno 2

|  |  |  |
| --- | --- | --- |
| **Funcionalidades** | **Objetivos** | **Pendências** |
|  | Implementar a geração de um arquivo |  |
|  | Implementar métricas |  |
|  | Implementar métricas e parser |  |
|  | Requisitos a serem observados durante todo o desenvolvimento. |  |

### Detalhamento do Release interno 3

|  |  |  |
| --- | --- | --- |
| **Funcionalidades** | **Objetivos** | **Pendências** |
|  | Implementar as métricas de complexidade essenciais. | * A métrica [RF011] Determinar a posição do maior aninhamento de loops não foi terminada, por o parser não ter terminado a tempo. |
|  | Requisitos a serem observados durante todo o desenvolvimento. |  |

### Detalhamento do Release interno 4

|  |  |  |
| --- | --- | --- |
| **Funcionalidades** | **Objetivos** | **Pendências** |
|  | Implementar |  |
|  | Implementar métricas). |  |
|  | Implementar |  |
|  | Implementar. |  |
|  | Implementação do Help da aplicação. |  |
|  | Requisitos a serem observados durante todo o desenvolvimento. |  |

# Plano de Testes

Esta seção define o planejamento de Testes para o projeto, com o objetivo de registrar o que será testado em cada iteração e também documentar aspectos globais relacionados a testes. Isto possibilitará um bem-sucedido gerenciamento e condução de testes no projeto.

## Organização e Responsabilidade

|  |  |
| --- | --- |
| **Nome** | **Responsabilidades** |
| Gabriel Barbosa e Rony Nogueira | Planejamento, Coordenação e Execução dos testes |
| Lucas Vinicios, João Henrique, Raphael Ferreira | Projeto de Testes |

## Requisitos e Cronograma Macro

Nessa seção serão definidas as necessidades para a realização dos testes dos requisitos estabelecidos para o projeto. A aplicação será testada por etapa, as quais serão detalhadas nas seções seguintes. Para cada etapa serão estabelecidos objetivos, datas e responsabilidades.

### Etapa 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Início:** | **19/08/2003** | **Data Fim:** | **19/08/2003** |
| **Estágio(s) Teste:** | **[X] Integração [X] Sistema [ ] Aceitação** | | |
| **Objetivo:** | Testar os requisitos para as métricas básicas | | |
| **Responsável:** |  | | |
| **Indicadores de Sucesso:** | TODOS os casos de teste devem ser concluídos com sucesso | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requisitos Funcionais** | | | |
| **Prioridade** | **Caso de Uso** | **Observações** | |
| 1 |  | **€** Automático  **X** Manual | **X** Caixa Preta  **€** Caixa Branca |
| *Várias variações de escopo serão testadas. O outuput da aplicação será armazenado no arquivo “output.txt”.* | |
| 2 |  | **€** Automático  **X** Manual | **X** Caixa Preta  **€** Caixa Branca |
| *Variações simples de escopo serão testadas. O outuput da aplicação será armazenado no arquivo “output.txt”.* | |
| 2 |  | **€** Automático  **X** Manual | **X** Caixa Preta  **€** Caixa Branca |
| *Variações simples de escopo serão testadas. O outuput da aplicação será armazenado no arquivo “output.txt”.* | |

|  |  |  |
| --- | --- | --- |
| **Requisitos Não-Funcionais** | | |
| **Prioridade** | **Requisito** | **Casos de uso relacionados** |
| 1 | NF (Interface gráfica) |  |
| 1 |  |  |
| 2 | NF (Tempo de resposta) |  |
| 1 | NF (Plataforma de execução) |  |
| 1 |  |  |
|  |  |  |

## Registro dos Resultados

Os resultados dos testes deverão ser registrados na Macro Área “Documentos”, pasta “Status” de cada caso de teste e nas planilhas de teste, caso existam.

# Plano de Aceitação

Essa seção define o Plano de Aceitação para o projeto, com o objetivo de registrar como será realizado o processo de aceitação da solução.

O projeto será concluído com sucesso caso os seguintes critérios estejam satisfeitos:

* Os deliverables descritos no documento de visão sejam entregues:

1. Software contemplando os requisitos exigidos
2. Manual do Usuário

* O software desenvolvido deve satisfazer os critérios estabelecidos no plano de qualidade
* A aceitação da solução será realizada de forma gradativa, à medida que as funcionalidades forem sendo desenvolvidas. Existirão duas etapas no processo:
  + Aprovação das definições funcionais: As funcionalidades a serem desenvolvidas devem ser descritas, com protótipo de GUI, para que o cliente possa aprova o seu desenvolvimento.
  + Aprovação do software desenvolvido no ambiente de testes pelo cliente.
* A aceitação do trabalho realizado deve ser comprovada através da aceitação dos relatórios de status, que descreverão o trabalho realizado e já homologado pelo cliente.

## Organização e Responsabilidade pela Aceitação

|  |  |
| --- | --- |
| Componente |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## Cronograma Macro

|  |  |  |
| --- | --- | --- |
| **Atividade** | **Responsabilidade** | **Data** |
| Homologar Métricas Básicas |  |  |
|  |  |  |
|  |  |  |

# Plano de Implantação

Não haverá visto que não é compromisso do projeto a instalação do software no cliente.

# Plano de Equipe

Esta seção apresenta um resumo da equipe responsável pelo projeto, registrando seus integrantes, suas responsabilidades e a sua estrutura de equipes multidisciplinares. As responsabilidades de cada componente podem ser consultadas no tópico 2 deste documento.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Papel** | **Carga horária diária (h)** | **Contato** |
| Gabriel |  | 6 |  |
| Lucas |  | 4 |  |
| João |  | 6 |  |
| Pedro |  | -- |  |
| Raphael |  | 6 |  |
| Rony | Scrum, Analista de Usuário e Desenvolvedor | 8 |  |

# Cronograma Geral

O cronograma do projeto está disponível no Trello, a definição de tarefas e prazos de entrega. O link para acesso é: <https://trello.com/b/2DNd4uSo/projeto-ygg>

# Referências

[1] Tópicos 2 e 6 foram retirados do documento {GEDOCARG\_PGCS\_2.3\_20160419} [Plano de Gerência de Configuração de Software – Disciplina do Rubens (Grupo: Pedro Victor, Gabriel, Lucas e João)] entregue dia 05/05 para o Professor, com adaptações;

[2] <http://www.cin.ufpe.br>

# Assinaturas

Eu concordo e aprovo esse documento de Plano de Projeto do software YGG, comprometendo-me a executar todas as tarefas atribuídas a mim, descritas no Plano acima.

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Pedro Victor de Almeida Lopes Rony Nogueira  
Gestor de Documentos Gestor do Back-End

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Gabriel Barbosa João Henrique  
Gestor do Front-End Desenvolvedor

# \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Gabriel Barbosa Lucas Vinicios

Desenvolvedor Desenvolvedor