DOCUMENTO DE REQUISITOS PARA O PROCESSO PARA O GERENCIAMENTO DE VERSÕES

v.2

**Histórico de Alterações**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 15/08/2015 | 1.0 | Abertura | Rebeca, Narci, Patrícia e Wagner |
| 15/08/2015 | 2.0 | Melhorias | Rebeca, Narci, Patrícia e Wagner |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sumário

[1. INTRODUÇÃO 4](#_Toc427408919)

[2) TECNOLOGIAS UTILIZADAS 5](#_Toc427408920)

[3) VERSIONAMENTO 6](#_Toc427408921)

[3.1) INGRESSO DE NOVOS DESENVOLVEDORES 6](#_Toc427408922)

[3.2) CRIAÇÃO DE NOVA DISTRIBUIÇÃO 6](#_Toc427408923)

[3.3) CRIAÇÃO DE NOVAS VERSÕES 7](#_Toc427408924)

[3.4) NOVAS ATUALIZAÇÕES 7](#_Toc427408925)

[3.5) NOVAS FUNCIONALIDADES 7](#_Toc427408926)

1. INTRODUÇÃO

Este documento define o processo de gerenciamento de versões de Software da empresa.

Esta documentação responderá as questões:

1. Como serão gerenciadas as:
   1. Novas versões do software
   2. Novas atualizações
   3. Correções de bugs
   4. Novas distribuições
2. Como novos desenvolvedores iniciam seus trabalhos?
3. Como novas funcionalidades poderão ser implementadas?
4. Como novos campos no Banco de Dados poderão ser adicionados?

A documentação deve ser compreensível para:

* O ingresso de novos desenvolvedores
* A criação de novas distribuições
* A criação de novas versões
* A criação de novas atualizações
* A criação de novas funcionalidades

2) TECNOLOGIAS UTILIZADAS

Tecnologias:

* Java
* JSF
* JPA
* PostgreSQL
* NetBeans
* Astah
* Github
* Jenkins

1. VERSIONAMENTO

Todos os arquivos serão versionados no Github e para tanto será definido um gerente como responsável pelo versionamento.

O gerente de versionamento determinará o controle de todas as configurações e controle de usuários que possuem permissão para *versionar* determinados arquivos.

O software terá 3 níveis de versão: trunk, branches e tags.

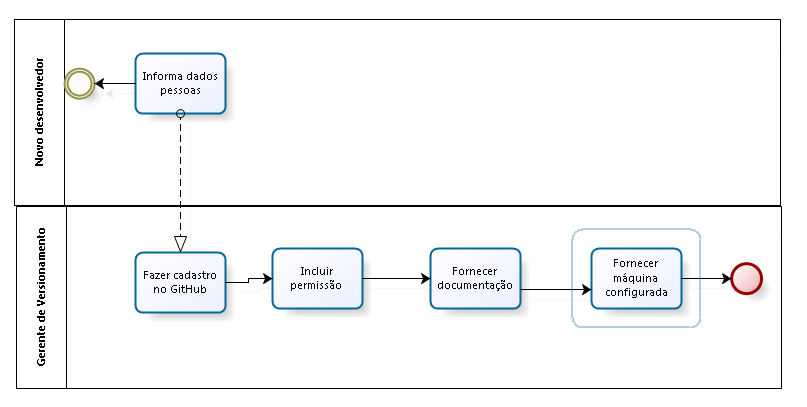
* O nível 1 - trunk pode ser representado como “v1” e é utilizado como um nível principal.
* O nível 2 – branch pode ser representando como “v1.1” e é utilizado quando acontece um acréscimo de funcionalidade.
* O nível 3 – tags pode ser representado como “v.1.1.1” e é utilizado quando é corrigido um bug do sistema.

Inicialmente será escolhido a versão basealine pelo gerente de projeto.

* 1. INGRESSO DE NOVOS DESENVOLVEDORES

Quando novos desenvolvedores forem contratados, o RH deverá informar o gerente de projeto, para que este informe o gerente de versionamento da nova contratação e assim proceder para:

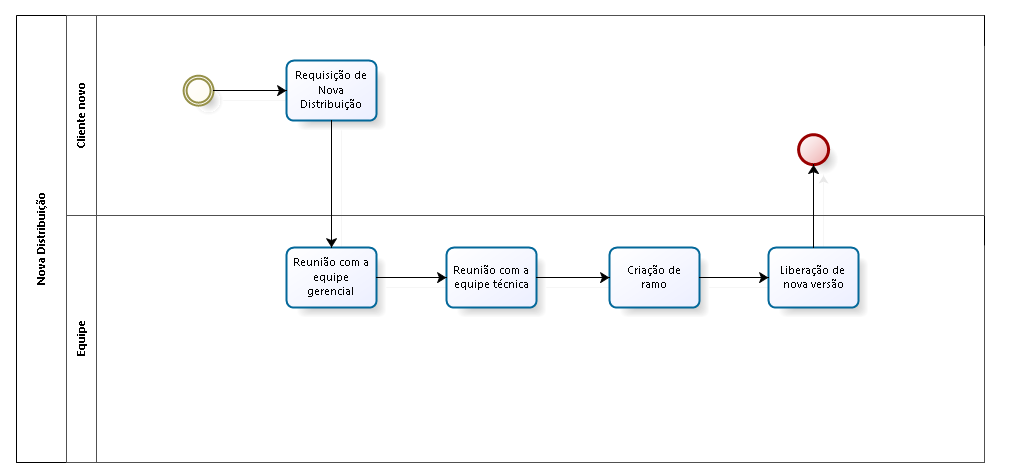
1. Novo desenvolvedor informa dados;
2. Fazer o cadastro do novo colaborador no Github;
3. Incluir as permissões de acesso ao novo colaborador do(s) projeto(s) que irá trabalhar;
4. Fornecer ao novo colaborador a documentação de configuração do projeto;
5. Fornecer a máquina para que o novo colaborador inicie a configuração da sua área de trabalho.



* 1. CRIAÇÃO DE NOVA DISTRIBUIÇÃO

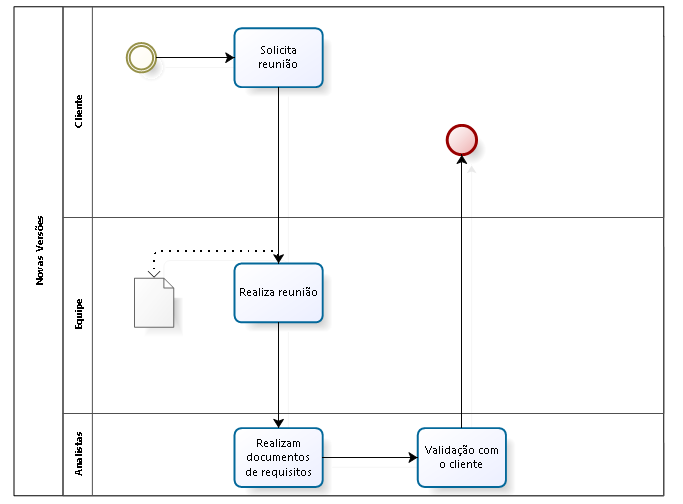
Ao fechar negócio com um novo cliente, será criado um branch da versão master para este cliente e a partir daí será realizado a customização. Assim, serão efetuadas as seguintes atividades:

|  |  |  |
| --- | --- | --- |
|  | Artefato de entrada | Artefato de saída |
| Reunião com a equipe gerencial; |  | Ata de reunião |
| Reunião com a equipe técnica; | Ata de reunião gerencial | Ata de reunião técnica |
| Criação de um branch a partir do ramo máster. | Ata de reunião técnica | Será executado o comando de criação para o novo ramo do branch. Esta versão receberá o nome do cliente. |
| Reunião com a equipe | Liberação da nova versão. | Será informado para a equipe a adição do ramo para o novo cliente e definido qual a equipe que irá trabalhar neste ramo para a customização |



* 1. CRIAÇÃO DE NOVAS VERSÕES

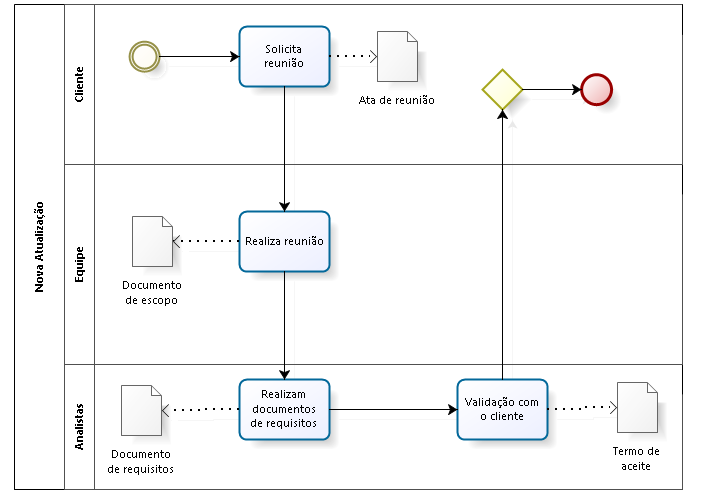
O time de desenvolvedores ao terminar a nova versão irá comunicar ao gerente de projeto, assim será escolhido o nova versão, seguindo os 3 (três) níveis de versão.



* 1. NOVAS ATUALIZAÇÕES

As novas atualizações do sistema seguirão o padrão de versionamento em 3(três) níveis anteriormente definidos. As atualizações deverão ser feitas no terceiro nível de versionamento, exemplo: “v.1.1.1”

|  |  |  |
| --- | --- | --- |
|  | Artefato de entrada | Artefato de Saída |
| Reunião com cliente |  | Ata de reunião |
| Reunião com equipe técnica | Ata de reunião | Atualização de documento de escopo |
| Os analistas deverão criar ou atualizar a documentação necessária de requisitos e de diagramas de UML | Documento de escopo | Documento de requisitos |
| Reunião com cliente | Documento de requisitos | Termo aceite da alteração |
|  |  |  |
|  |  |  |



* 1. NOVAS FUNCIONALIDADES

Ao ser definido uma nova funcionalidade com o cliente, o gerente de projeto fará uma reunião para informar a equipe técnica da nova funcionalidade. As atualizações deverão ser feitas no terceiro nível de versionamento, exemplo: “v.1.1”.

Assim, será realizado as seguintes atividades:

|  |  |  |
| --- | --- | --- |
|  | Artefato de entrada | Artefato de Saída |
| Reunião com cliente |  | Ata de reunião |
| Reunião com equipe técnica | Ata de reunião | Atualização de documento de escopo |
| Os analistas deverão criar ou atualizar a documentação necessária de requisitos e de diagramas de UML | Documento de escopo | Documento de requisitos |
| Reunião com cliente | Documento de requisitos | Termo aceite da alteração |
|  |  |  |
|  |  |  |

