



Tribunal Regional Eleitoral da Paraíba
Coordenadoria de Sistemas
Seção de Análise e Desenvolvimento de Sistemas

Modus 3.1 - Modelo de desenvolvimento de software

Introdução

O Modus é o processo de desenvolvimento de software adotado pela SEDES. Atualmente, o processo está na sua terceira versão. Optou-se pela elaboração de um processo mais enxuto, baseado em práticas que vem sendo adotadas ordinariamente pela unidade.

Não é objetivo deste documento explicar em detalhes as práticas adotadas pela SEDES oriundas de processos de desenvolvimento ágil, abordagem bastante utilizada atualmente em diversas instituições e empresas, com bastante eficiência. Apenas a título de informação, são utilizadas práticas e técnicas de algumas metodologias ágeis, tais como: Scrum, XP (extreme programming) e FDD (feature driving development).

O modelo de processo utilizado é iterativo e incremental: as versões do produto são homologadas pelo cliente e melhorias e novas funcionalidades podem ser adicionadas durante o projeto, conforme feedback dos usuários.

Dentre as práticas e técnicas, destacam-se: reunião diária (stand-up meeting ou daily meeting), uso de taskboard, entregas frequentes (disponibilização de versões com funcionalidades já implementadas em curto espaço de tempo - tipicamente um mês), elicitação de requisitos através de histórias de usuários, planning poker (técnica para estimar tamanho - tempo - a ser dedicado a uma história de usuário) e melhoria contínua (lições aprendidas do projeto servem para a melhoria do processo e do modelo de desenvolvimento empregado).

Papéis

Papel	Descrição
Gerente do projeto	É a pessoa responsável pela condução do projeto, planejando e coordenando o desenvolvimento, mantendo o time motivado e resolvendo impedimentos e conflitos de interesses que possam prejudicar o andamento do projeto. Dentre suas atribuições está o papel de <i>Scrum Master</i> do método ágil <i>Scrum</i> .
Time	É uma equipe formada por desenvolvedores que executarão as atividades de análise, construção e de manutenção dos produtos, sob a coordenação do gerente de projeto
Cliente	São pessoas interessadas no projeto que fornecem os requisitos para o time e homologam as entregas
Gestor do sistema	É uma pessoa destacada do grupo cliente com bastante interesse pelo projeto, disponibilidade e influência suficiente para advogar em favor dos propósitos do projeto. Junto ao time ele deve decidir sobre requisitos conflitantes e estabelecer prioridades. Deve ser designado por portaria.
Principal fornecedor de requisitos	É a pessoa do grupo cliente que detém um bom domínio do negócio e com maior disponibilidade para fornecer os requisitos ao time e esclarecer dúvidas. Em geral esse papel é assumido pelo gestor do sistema.

Modus Projetos

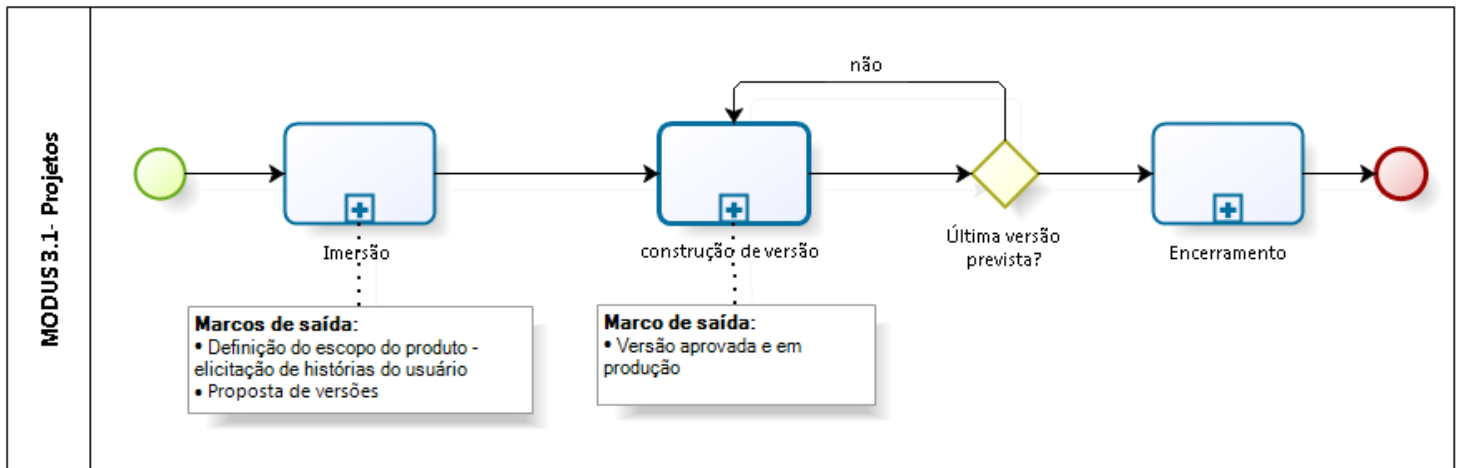


Ilustração 1: Fases do modelo de desenvolvimento para projetos

Imersão

O propósito desta etapa é realizar uma imersão no domínio do cliente, identificar claramente a demanda e os benefícios que ela deverá agregar. Com base nessas informações, o time deve elicitar as principais histórias do usuário e elaborar um plano macro e uma proposta de prazo para finalização do projeto.

Dentro da proposta de desenvolvimento ágil, obviamente, novas histórias e alterações no plano de versões podem ocorrer naturalmente no decorrer do projeto.

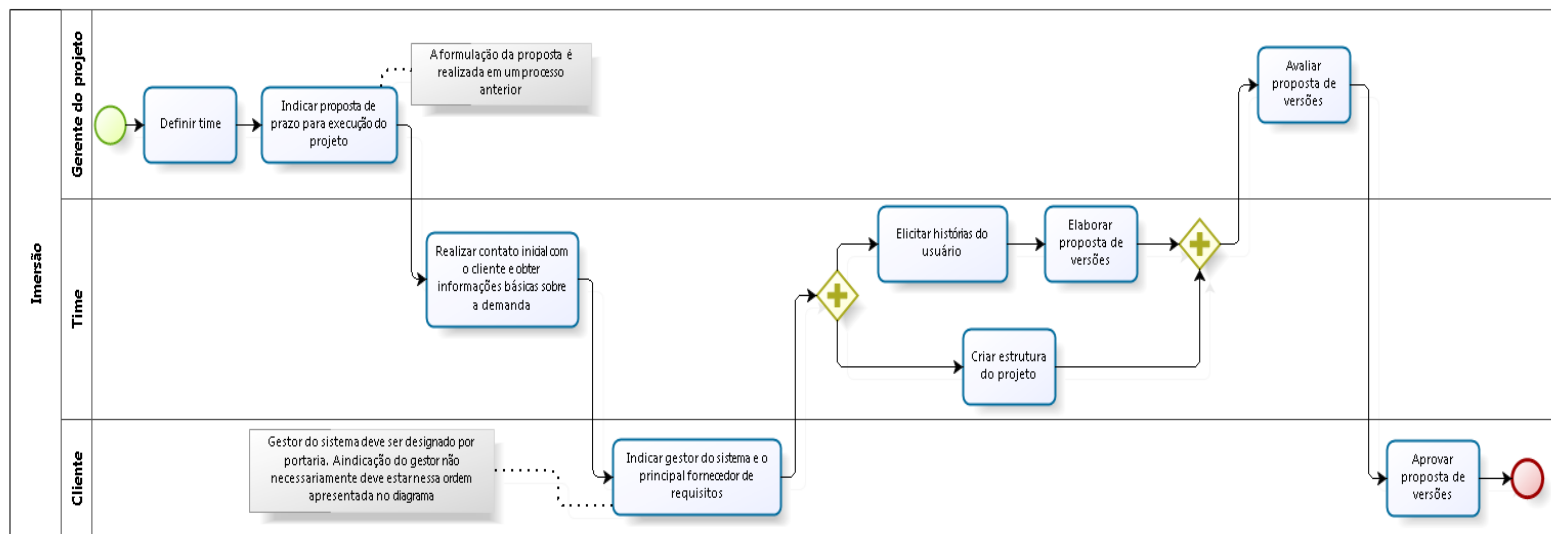


Ilustração 2: Fluxo da fase de imersão

Definir time

Ator: Gerente do projeto

Após a aprovação do desenvolvimento do projeto (realizada em um processo anterior), o gerente define o time do projeto, ou seja, quais pessoas atuarão na análise e construção do produto.

Entradas	Saídas
<ul style="list-style-type: none"> Projeto aprovado para execução 	<ul style="list-style-type: none"> Determinação do time (pessoas que executarão o projeto)

Indicar proposta de prazo para execução do projeto

Ator: Gerente do projeto

O gerente deve informar ao time qual o prazo proposto pela STI e COSIS para realização do projeto. A definição dessa proposta é realizada em um processo anterior ao de desenvolvimento. Essa proposta de prazo poderá ser ajustada posteriormente com o time e também com o cliente, mas serve de base para a definição do escopo do projeto.

Entradas	Saídas
<ul style="list-style-type: none"> Calendário de projetos da COSIS 	<ul style="list-style-type: none"> Indicação para o time do prazo sugerido

Realizar contato inicial com o cliente e obter informações básicas sobre a demanda

Ator: Time

Durante a execução do projeto são realizadas várias reuniões com o cliente, mas esse contato inicial destaca-se por ser um marco para o projeto. Nele, além de conhecer o cliente, o time e o gerente poderão extrair a ideia principal do produto e que benefícios ele visa alcançar.

Entradas	Saídas
Projeto aprovado para execução	<ul style="list-style-type: none"> Ata de reunião inicial

Indicar gestor do sistema e o principal fornecedor de requisitos

Ator: Cliente

Nas reuniões iniciais ou até antes delas, a pessoa que desempenhará o papel de gestor do sistema deve ser definida dentre os clientes. Portaria específica deve ser elaborada para designação do gestor do sistema e seu suplente.

Junto ao time, o gestor será responsável por determinar prioridades e atuar como referência no fornecimento de requisitos e no esclarecimento de dúvidas sobre o negócio.

Entradas	Saídas
Projeto aprovado para execução	<ul style="list-style-type: none"> Definição do gestor do sistema e do principal fornecedor de requisitos para o time

Criar estrutura do projeto

Ator: Time

Esta tarefa consiste em criar os artefatos estruturais do projeto:

- Projeto na ferramenta Redmine (<http://redmine.tre-pb.gov.br/>)
- Inclusão do produto no catálogo da COSIS (<http://redmine.tre-pb.gov.br/projects/cat-tec-sist>)
- Página do produto
- Arcabouço do sistema armazenado no repositório SVN (<http://svn.tre-pb.gov.br/svn/cosis>) e gerado a partir de arquétipo de arquitetura padrão das aplicações.

Entradas	Saídas
Projeto aprovado para execução	<ul style="list-style-type: none"> Projeto no Redmine Página do projeto Inclusão do produto no catálogo Arcabouço do sistema no SVN

Elicitar histórias do usuário

Ator: Time

Histórias de Usuário devem ser identificadas a partir dos registros em atas de reunião. São relatos de funcionalidades que agregam valor para o cliente. Assim, o time deve traduzir as expectativas e demandas do cliente em Histórias do Usuário.

Nesta fase, é necessária apenas uma descrição sucinta de cada história de usuário identificada durante os contatos com o cliente. Convém destacar que a ideia geral já deva estar descrita em ata de reunião – ou outro artefato – já realizada com o cliente.

As histórias são cadastradas como *tickets* dentro do projeto na ferramenta Redmine, ainda sem o detalhamento necessário a ser elaborado em outra etapa do processo. O conjunto dessas histórias determina o escopo base do projeto.

Entradas	Saídas
<ul style="list-style-type: none"> Projeto aprovado para execução Ata de reunião inicial 	<ul style="list-style-type: none"> Histórias cadastradas no Redmine

Elaborar proposta de versões

Ator: Time

Com base nas histórias criadas, na priorização do cliente e ainda no prazo sugerido pelo gerente do projeto, o time elabora uma proposta de plano de versões do produto.

Cada versão é uma unidade funcional do sistema em produção contemplando um conjunto de histórias.

Entradas	Saídas
<ul style="list-style-type: none"> Histórias elicítadas no Redmine Priorização do cliente Sugestão de prazo de conclusão do projeto 	<ul style="list-style-type: none"> Proposta de versões do projeto no Redmine, indicando o escopo para cada uma delas.

Avaliar proposta de versões

Ator: Gerente do projeto

Após a elaboração da proposta de versões, que corresponde ao plano de entregas do produto para o cliente, o time deve apresentar o resultado para o gerente do projeto.

A proposta pode apresentar um prazo final diferente da sugestão inicial do gerente e o time deve apontar as justificativas. O gerente então avalia o planejamento, aponta sugestões e todos definem a proposta final a ser entregue ao cliente.

Entradas	Saídas
<ul style="list-style-type: none"> Histórias cadastradas no Redmine Proposta de versões do projeto no Redmine 	<ul style="list-style-type: none"> Proposta de versões do projeto no Redmine acordada entre o time e o gerente

Aprovar proposta de versões

Ator: Cliente

A proposta de versões é apresentada e discutida com o cliente. Todos devem chegar a um consenso que é referendado com a aprovação da proposta pelo cliente.

Entradas	Saídas
<ul style="list-style-type: none"> Proposta de versões do projeto no Redmine acordada entre o time e o gerente 	<ul style="list-style-type: none"> Proposta de versões do projeto no Redmine acordada entre o time, o gerente e o cliente

Construção

Execução de atividades usando princípios de metodologias ágeis para construção e entrega de uma versão funcional para o cliente. O acompanhamento diário das atividades pode ser feito através do *taskboard*.

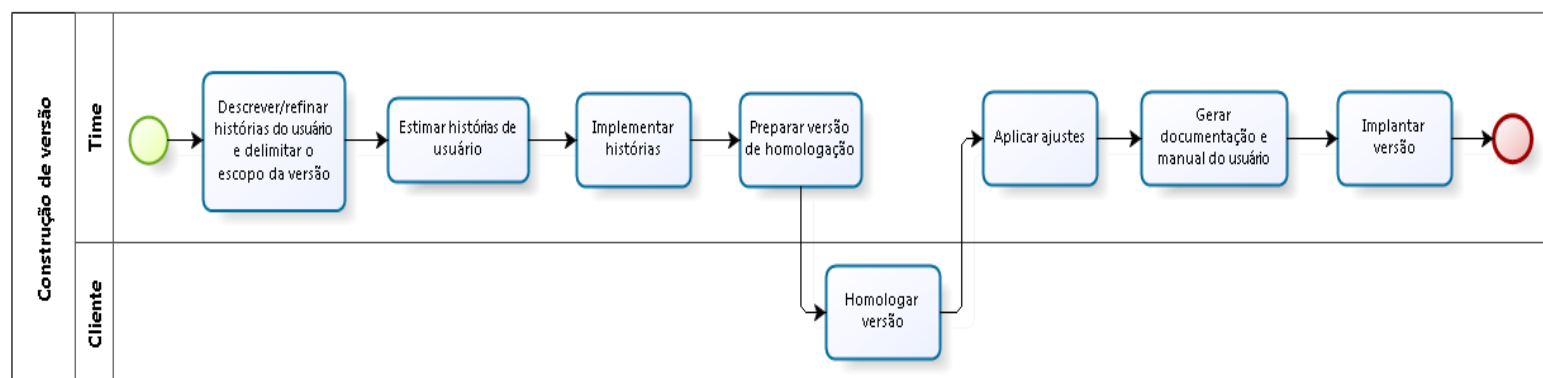


Ilustração 3: Fluxo de atividades da fase de construção de versão

Descrever/refinar histórias do usuário e delimitar o escopo da versão

Ator: Time

As histórias de usuário já elicitadas deverão ser atualizadas de forma detalhada na ferramenta Redmine. Numa história deve ser indicado o propósito

que ela visa atender, que papel dentro do sistema executa a história e o que ela realiza.

São indicadas também as condições esperadas e as ações que deverão ser executadas. Com essas informações, são estabelecidos critérios de aceitação, sob os quais pode-se avaliar se o propósito foi atendido.

Com a evolução do time no domínio do negócio e com a experiência do cliente após as entregas das primeiras versões, pode ser necessário o refinamento das histórias para alinhar a versão a ser construída com as expectativas do cliente, bem como delimitar o escopo da versão em decorrência de mudanças no planejamento inicial. As histórias do usuário poderão ser ajustadas conforme entendimento entre o time e o cliente.

Entradas	Saídas
<ul style="list-style-type: none"> Histórias do usuário propostas para a versão Feedback do cliente 	<ul style="list-style-type: none"> Histórias do usuário refinadas no Redmine Escopo da versão ajustado

Estimar histórias

Ator: Time

O time deve discutir as histórias e estimar o esforço de realização de cada uma delas individualmente. Todo o esforço despendido para analisar (se ainda existirem detalhes não especificados), implementar, revisar e documentar a história deve ser avaliado.

Esta tarefa é realizada com base na técnica conhecida como *Planning Poker* e a unidade de esforço é um dia de trabalho de um desenvolvedor.

Entradas	Saídas
<ul style="list-style-type: none"> Histórias escritas no Redmine 	<ul style="list-style-type: none"> Estimativas indicadas em cada história no Redmine

Implementar histórias

Ator: Time

Implementação das histórias do usuário tendo como base os padrões de arquitetura, de banco de dados e de design estabelecidos, fazendo uso de técnicas de desenvolvimento ágil do Scrum, como por exemplo o uso de *taskboard* e a realização de reuniões diárias de acompanhamento. As histórias do usuário são divididas em atividades diárias que serão executadas e em seguida revisadas por outro membro do time. A revisão das atividades objetiva minimizar os defeitos e as não conformidades com a especificação da história.

Os artefatos produzidos (código fonte, modelos de dados, scripts, etc.) são armazenados no repositório SVN e associados ao número da história no Redmine para possibilitar a rastreabilidade.

Entradas	Saídas
----------	--------

<ul style="list-style-type: none"> • Histórias do usuário detalhadas e refinadas selecionadas para a versão • Diagrama de modelagem de processos de negócio, se houver • Protótipos e outros artefatos produzidos na etapa de imersão 	<ul style="list-style-type: none"> • Código-fonte da versão • Scripts de criação/atualização do banco de dados • Modelo entidade-relacionamento
--	--

Preparar versão de homologação

Ator: Time

Construção da versão e implantação em ambiente de homologação para que o cliente valide as histórias implementadas. O banco de dados de homologação deve ser criado, ou atualizado, e as histórias implementadas devem ser testadas minimamente pelo time em ambiente de homologação para evitar falhas de configuração.

O time também deve preparar os dados básicos para que o cliente possa realizar os testes de aceitação. Dependendo da complexidade do sistema, diversidade de usuários e funcionalidades envolvidas, o time poderá elaborar também oficinas de treinamento no sistema. Nesse caso, roteiros das oficinas devem ser elaborados.

Entradas	Saídas
<ul style="list-style-type: none"> • Código-fonte da versão • Scripts de banco de dados 	<ul style="list-style-type: none"> • Versão implantada no ambiente de homologação • Base de dados para testes preparada • Roteiros para oficinas, se houver

Homologar versão

Ator: Cliente

A equipe apresenta a versão ao cliente para que realize os testes de aceitação. O cliente avaliará principalmente a conformidade com o que foi solicitado. O cliente analisará também aspectos não-funcionais como a usabilidade do sistema, o tempo de resposta e a segurança.

Além da reunião com o time, a versão de homologação é disponibilizada para exploração dos usuários por alguns dias. Também podem ser realizadas oficinas de treinamento para clientes e usuários do sistema.

Ao final desse processo, o cliente deve dar sua aprovação à entrega ou indicar melhorias e correções necessárias.

Entradas	Saídas
<ul style="list-style-type: none"> • Versão implantada no ambiente de homologação 	<ul style="list-style-type: none"> • Feedback do cliente para o time sobre aprovação da versão e

<ul style="list-style-type: none"> • Base de dados para testes preparada • Roteiro de oficinas, se houver 	<ul style="list-style-type: none"> • ajustes necessários • Ata da reunião de homologação
---	--

Aplicar ajustes

Ator: Time

Essa tarefa consiste em implementar as melhorias e correções apontadas pelo cliente durante a homologação.

A estratégia de desenvolvimento de forma evolutiva e incremental associada a entregas frequentes, permite que não conformidades sejam identificadas e solucionadas de forma mais rápida, de modo que os ajustes apontados na homologação sejam geralmente pontuais e o time consiga fazer as adaptações sem comprometer a entrega planejada.

Entradas	Saídas
<ul style="list-style-type: none"> • Feedback do cliente para o time • Ata da reunião de homologação 	<ul style="list-style-type: none"> • Código-fonte ajustado • Histórias do usuário ajustadas, se for o caso

Gerar documentação e manual do usuário

Ator: Time

O objetivo dessa tarefa é fazer a inclusão ou atualização do produto no catálogo de produtos, e elaboração ou atualização do manual do usuário contemplando as histórias implementadas na versão construída. Outros artefatos de documentação podem ser elaborados conforme a necessidade do sistema.

Entradas	Saídas
<ul style="list-style-type: none"> • Versão implementada • Histórias do usuário detalhadas e refinadas selecionadas para a versão 	<ul style="list-style-type: none"> • Catálogo de produtos atualizado • Manual do usuário elaborado ou atualizado

Implantar a versão

Ator: Time

O objetivo desse passo é a implantação em ambiente de produção para uso do cliente. O banco de dados de produção deve ser criado ou atualizado e um pacote com a versão deve ser implantado no ambiente de produção.

A versão é numerada com dígitos na forma **X.Y.Z**. Os dois primeiros dígitos mais significativos (**X** e **Y**) são utilizados para incrementar o número de versão e o último (**Z**) para *patches* com correções de *bugs*.

Uma *tag* deve ser criada no sistema de controle de versões SVN para

permitir recuperar o código-fonte conforme a versão foi construída. O cliente deve ser informado que a versão está disponível para uso em produção.

Entradas	Saídas
<ul style="list-style-type: none"> Código-fonte da versão Scripts de banco de dados 	<ul style="list-style-type: none"> Tag no sistema de controle de versões SVN Versão implantada no ambiente de produção Comunicação via e-mail da entrada em produção para o cliente e demais interessados

Encerramento

Execução das atividades de finalização do projeto, como a formalização do responsável pelo suporte de negócio e a discussão sobre melhorias do projeto, de seu modelo/arquitetura, dos procedimentos, das técnicas e do método de desenvolvimento.

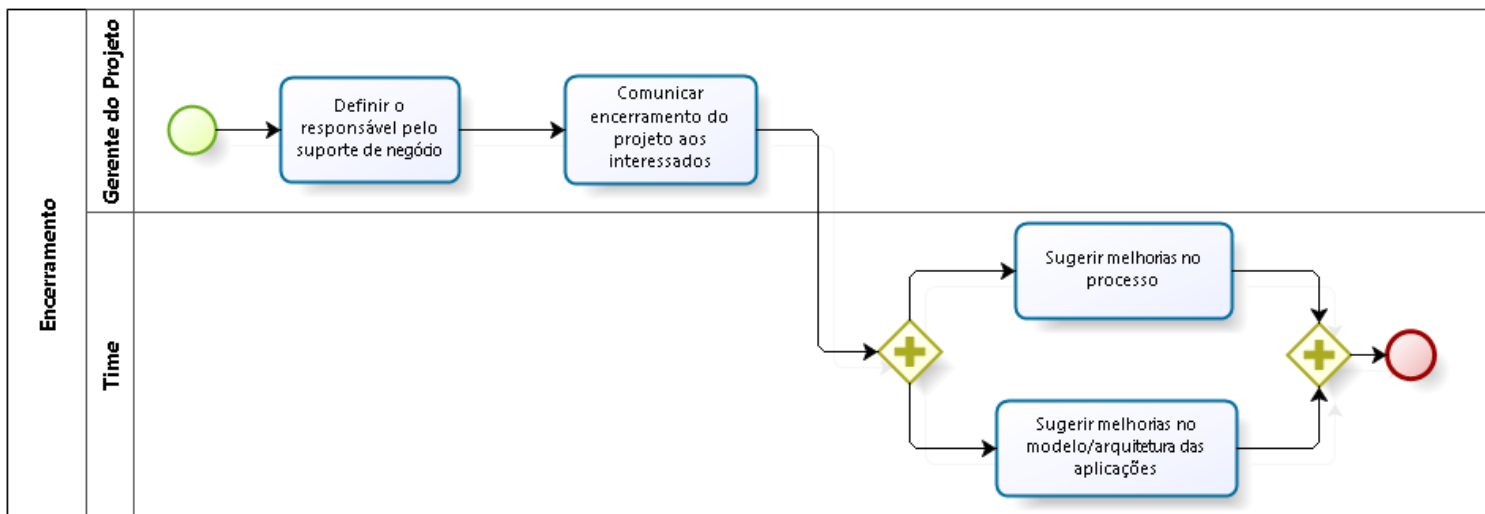


Ilustração 4: Fluxo de atividades da fase de encerramento do projeto

Definir o responsável pelo suporte de negócio

Ator: Gerente do projeto

Realização de uma reunião final de entrega do produto com o time, o gerente e o cliente, na qual deverá ser definido o responsável pelo suporte de negócio do produto. Preferencialmente, o gestor do sistema, ou alguém por ele delegado, deve assumir este papel, que consiste em esclarecer as dúvidas

negociais reportadas pelos usuários do produto. À SEDES caberá o suporte técnico para correções e melhorias no sistema.

Entradas	Saídas
<ul style="list-style-type: none"> Versões finalizadas 	<ul style="list-style-type: none"> Ata da reunião

Comunicar encerramento do projeto aos interessados

Ator: Gerente do projeto

Comunicação aos interessados, através de e-mail ou despacho/documento em processo administrativo, informando o encerramento do projeto.

Entradas	Saídas
<ul style="list-style-type: none"> Versões finalizadas 	<ul style="list-style-type: none"> E-mail de encerramento do projeto ou documento/despacho de encerramento em processo administrativo dando publicidade aos interessados.

Sugerir melhorias no processo

Ator: Time

Realização de reunião entre o time e o gerente para discutir os problemas relativos ao processo de trabalho enfrentados e as soluções adotadas durante o projeto. As lições aprendidas deverão gerar oportunidades de melhorias que deverão ser incluídas no *backlog* de demandas da SEDES para implementação em momento oportuno e adoção em futuros projetos.

Entradas	Saídas
<ul style="list-style-type: none"> Problemas enfrentados e soluções adotadas 	<ul style="list-style-type: none"> Melhorias cadastradas no Redmine

Sugerir melhorias no modelo/arquitetura das aplicações

Ator: Time

Realização de reunião entre o time e o gerente para avaliação de sugestões de melhorias no modelo/arquitetura das aplicações. Serão discutidos os problemas enfrentados e as soluções adotadas durante o projeto. As melhorias serão incluídas no *backlog* de demandas da SEDES para implementação em momento oportuno.

Entradas	Saídas
<ul style="list-style-type: none"> Problemas enfrentados e soluções adotadas 	<ul style="list-style-type: none"> Melhorias cadastradas no Redmine

Modus Manutenções

Processo de manutenção de produtos para implementação de melhorias ou correções provenientes do *backlog* de demandas mantido no projeto SEDES Demandas do Redmine. Compreende as atividades realizadas após a seleção da demanda para ser executada, até a liberação do produto para produção. As atividades de cadastro de demandas, priorização e seleção para execução são realizadas em um processo anterior.

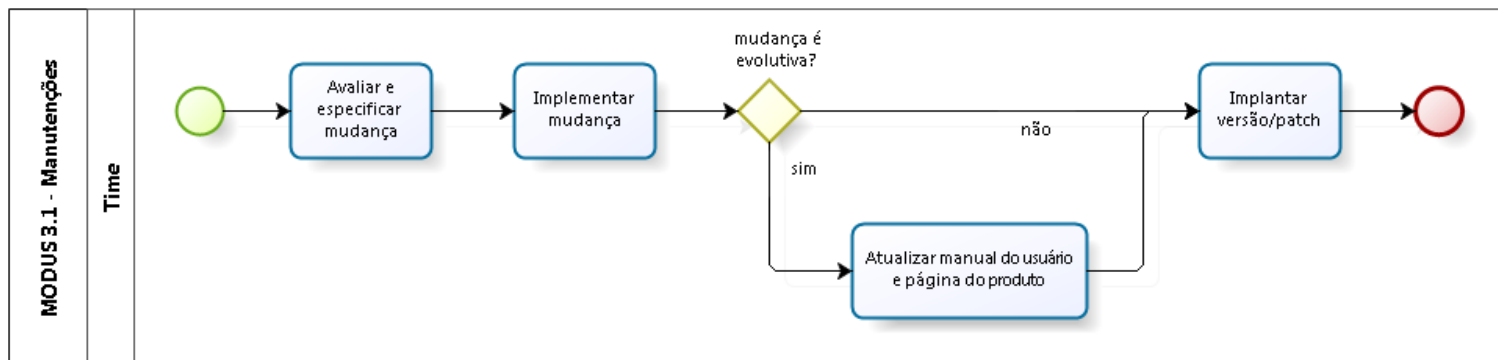


Ilustração 5: Fluxo de atividades do processo de manutenção de produtos

Avaliar e especificar mudança

Ator: Time

Uma demanda acarretará em mudanças no produto com a incorporação de novas histórias do usuário, melhorias ou correções em histórias existentes. O time deve analisar mais detalhadamente o que foi solicitado na demanda e especificar a mudança, registrando novas histórias, melhorias ou correções no projeto correspondente ao produto no Redmine.

A tarefa de melhoria ou correção de bug deve obrigatoriamente ser associada a história do usuário correspondente no Redmine.

A mudança deverá gerar uma nova versão ou *patch* do produto. Se a mudança for evolutiva, uma nova versão deve ser criada incrementando o segundo dígito identificador em relação a versão anterior. Ou seja, se a versão anterior era a 1.2.4, agora a nova versão será 1.3.0. Se a mudança for apenas corretiva, deve ser gerado um novo *patch*, incrementando o terceiro dígito

identificador. Tomando o exemplo anterior, o *patch* seria nomeado como 1.2.5.

Por fim, com base na avaliação da mudança, o time poderá informar ao chefe da unidade e ao cliente qual o prazo de entrega da mudança.

Entradas	Saídas
<ul style="list-style-type: none"> • Demanda selecionada 	<ul style="list-style-type: none"> • Mudança especificada no Redmine como novas histórias do usuário, melhorias ou bugs • Prazo para entrega • Identificação do <i>patch</i> ou versão do produto a ser gerado

Implementar mudança

Ator: Time

Essa atividade é semelhante a atividade *Implementar histórias* descrita na fase *Construção da versão* do processo *Modus - Projetos*.

Entradas	Saídas
<ul style="list-style-type: none"> • Mudança especificada no Redmine como novas histórias do usuário, melhorias ou bugs 	<ul style="list-style-type: none"> • Código-fonte da versão • Scripts de criação/atualização do banco de dados • Modelo entidade-relacionamento

Atualizar manual do usuário e página do produto**Ator: Time**

Se a mudança é evolutiva, ou seja, novas características foram incorporadas, o manual do usuário e a página do produto precisam ser atualizados.

Entradas	Saídas
<ul style="list-style-type: none"> Mudança especificada no Redmine como novas histórias do usuário, melhorias ou bugs Mudança implementada 	<ul style="list-style-type: none"> Catálogo de produtos atualizado Manual do usuário atualizado

Implantar versão/patch**Ator: Time**

O objetivo dessa atividade é a implantação do *patch* ou versão em ambiente de produção para uso do cliente. Nela são realizadas as mesmas tarefas e procedimentos descritos na atividade *Implantar versão* da fase *Construção da versão* do processo *Modus - Projetos*.

Entradas	Saídas
<ul style="list-style-type: none"> Código-fonte da versão Scripts de banco de dados 	<ul style="list-style-type: none"> Tag no sistema de controle de versões SVN Versão implantada no ambiente de produção Comunicação via e-mail da entrada em produção para o cliente e demais interessados