MDMS-ANAC

Metodologia de Desenvolvimento e Manutenção de Sistemas da ANAC

Superintendência de Tecnologia e Transformação Digital – STD



Histórico de Alterações

Versão	Data de Publicação	Data de Entrada em Vigor	Responsável	Descrição
1.0	23/08/2010	23/08/2010	Rodrigo Pimenta de Figueiredo	• Versão inicial
1.1	21/10/2010	21/10/2010	Clésio Soares de Moura	• Revisão
1.2	28/10/2010	28/10/2010	Clésio Soares de Moura	• Inclusão de fluxogramas
1.3	05/11/2010	05/11/2010	Clésio Soares de Moura	• Revisão dos procedimentos
1.4	11/11/2010	11/11/2010	Cristiano Machado Cesário	• Revisão Geral
1.5	18/11/2010	18/11/2010	Clésio Soares de Moura	• Descrição de papéis e revisão
1.6	24/11/2010	24/11/2010	Clésio Soares de Moura	• Solicitações de mudanças
1.7	02/12/2010	02/12/2010	Clésio Soares de Moura	• Descrição dos papéis
1.8	17/12/2010	17/12/2010	Clésio Soares de Moura	• Revisão dos apêndices
1.9	30/12/2010	30/12/2010	Clésio Soares de Moura	• Adaptação entre artefatos e templates
1.10	31/01/2011	31/01/2011	Clésio Soares de Moura	• Preparação para publicação
1.11	03/02/2011	03/02/2011	Clésio Soares de Moura	• Revisão dos artefatos da Iniciação
1.12	12/04/2011	12/04/2011	Clésio Soares de Moura	• Revisão dos conceitos de Ciclos e Iterações
1.13	15/10/2013	15/10/2013	Rafael Dias	• Revisão geral dos modelos artefatos das fases de Iniciação, Elaboração, Construção e Transição.
1.14	19/03/2014	19/03/2014	Rafael Dias	• Revisão dos artefatos obrigatórios de cada fase para projeto de desenvolvimento • Alteração dos artefatos de projeto de desenvolvimento de sistema ▪ Alteração do modelo do Documento de Visão ▪ Alteração do modelo da Especificação de Mensagens ▪ Alteração do modelo da Especificação de Regras de Negócio

				▪ Alteração do modelo da Especificação de Caso de Uso ▪ Alteração do modelo da Especificação de Telas ▪ Criação do modelo da Matriz de Rastreabilidade ▪ Criação do modelo do Documento de Especificação Técnica ▪ Criação do modelo do Roteiro de Implantação
1.15	14/03/2014	14/03/2014	Rafael Dias	• Revisão dos artefatos obrigatórios de cada fase para projeto de desenvolvimento. • Alteração dos artefatos de projeto de desenvolvimento de sistema ▪ Alteração do modelo da Especificação de Caso de Uso ▪ Alteração do modelo da Especificação de Telas ▪ Alteração do modelo da Especificação de Regras de Negócio e renomeação para Especificação de Regras do Sistema ▪ Alteração do modelo do Matriz de Rastreabilidade ▪ Alteração do modelo do Documento de Especificação Técnica ▪ Alteração do modelo do Documento de Arquitetura
2.0	13/06/2014	13/06/2014	Rafael Dias	• Revisão dos artefatos obrigatórios de desenvolvimento e manutenção ▪ Alteração do modelo do Documento de Visão ▪ Alteração do modelo do Cronograma ▪ Alteração do modelo do Documento de Arquitetura ▪ Alteração do modelo da Especificação de Mensagens ▪ Alteração do modelo da Especificação de Regras do Sistema

				▪ Alteração do modelo da Especificação Suplementar ▪ Alteração do modelo da Especificação de Caso de Uso ▪ Alteração do modelo do Documento de Especificação Técnica ▪ Alteração do modelo da Especificação de Telas ▪ Alteração do modelo do Glossário do Sistema ▪ Alteração do modelo do Modelo de Caso de Uso ▪ Alteração do modelo da Ordem de Serviço ▪ Alteração do modelo do Plano de Testes ▪ Alteração do modelo do Relatório de Entrega ▪ Alteração do modelo do Roteiro de Testes ▪ Alteração do modelo do Roteiro de Implantação e renomeação para Plano de Implantação ▪ Criação do modelo do Documento de Especificação de Mudança ▪ Alteração do modelo da Análise de Impacto

2.1	22/01/2015	22/01/2015	Rafael Dias	• Alteração do nome de Metodologia de Desenvolvimento de Sistemas (MDS) para Metodologia de Desenvolvimento e Manutenção de Sistemas (MDMS) • Mapeamento dos fluxos da MDMS em BPMN • Revisão da redação da MDMS • Revisão dos artefatos obrigatórios de desenvolvimento e manutenção ▪ Remoção da Especificação Suplementar da MDMS ▪ Remoção do Modelo de Caso de Uso da MDMS ▪ Criação do modelo da Especificação de Serviço ▪ Alteração do modelo do Documento de Especificação de Mudança e renomeação para Especificação de Manutenção ▪ Alteração do modelo do Termo de Homologação e renomeação para Termo de Aceite
2.2	28/01/2015	28/01/2015	Rafael Dias	• Alteração do fluxo da fase de Iniciação • Remoção do Artefato da Ordem de Serviço. Esse artefato será utilizado de acordo com as diretrizes do contrato. Cada contrato definirá quando ele será utilizado. • Alteração na diretriz de duração de iteração; • Alteração nas atividades de testes da fase elaboração e construção; • Correção no fluxo da manutenção. • Atualização do modelo do Documento de Arquitetura

2.3	10/06/2015	10/06/2015	Rafael Dias	• Revisão geral da metodologia para torná-la mais aderente a práticas ágeis. Os marcos de demandas e projetos foram alterados. • Os seguintes artefatos foram retirados da MDMS: ▪ Termo de Aceite de Fase ▪ Relatório de Entrega ▪ Especificação de Telas ▪ Especificação de Mensagens ▪ Especificação de Regras de Sistema • Os seguintes artefatos foram adicionados da MDMS: ▪ Testes Automatizados ▪ Scripts de Banco de Dados ▪ Relatório de Incidentes de Testes • Os seguintes artefatos foram alterados da MDMS: ▪ Cronograma ▪ Documento de Arquitetura ▪ Documento de Especificação Técnica ▪ Documento de Visão ▪ Especificação de Caso de Uso ▪ Especificação de Manutenção ▪ Especificação de Serviço ▪ Glossário do Sistema ▪ Manual do Usuário ▪ Plano de Implantação ▪ Plano de Testes ▪ Roteiro de Testes ▪ Contagem de Pontos de Função • O artefato Documento de Especificação Técnica foi renomeado para Documento de Especificação de Telas
2.4	07/10/2015	07/10/2015	Rafael Dias	• Revisão do Processo de Projeto de Desenvolvimento e Manutenção • Revisão do Processo da Visão do Projeto • Revisão do Processo da Sprint

				• Revisão do Processo de Manutenção • Os seguintes artefatos foram retirados da MDMS: ▪ Análise de Impacto ▪ Documento de Especificação de Telas ▪ Documento de Visão ▪ Especificação de Manutenção ▪ Glossário do Sistema • Os seguintes artefatos foram adicionados da MDMS: ▪ Caso de Desenvolvimento
2.5	22/01/2016	22/01/2016	Rafael Dias/ Eder Souza / Flávio Carvalho	• Correção do fluxo Desenvolvimento e Manutenção; • Correção dos artefatos obrigatórios da seção 3.5.1; • Correção do fluxo Gerenciar Mudança; • Correção do fluxo Validar Contagem. • Exclusão de artefatos (templates Word): ▪ Especificação de Caso de Uso ▪ Plano de Implantação ▪ Plano de Testes ▪ Análise de Impacto ▪ Ordem de Serviço ▪ Roteiro de Testes • Alteração de artefatos (templates Word): ▪ Documento de Arquitetura ▪ Especificação de Serviço
2.6	19/02/2016	19/02/2016	Eder Souza	• Alteração de artefatos (Templates): ▪ Planilha de Contagem de Ponto de Função ▪ Caso de Desenvolvimento

2.7	23/02/2016	23/02/2016	Eder Souza	• Alteração de artefatos (Templates): ▪ Documento de Arquitetura
2.8	30/05/2016	30/05/2016	Eder Souza / Rafael Dias	• Alteração da duração da Visão do Projeto ▪ De até 5 para até 10 dias • Retirado a aprovação do Backlog do fluxo da Visão do Projeto • Correção do nome “Estória” para “História” • Alteração de artefatos (Templates): ▪ Caso de Desenvolvimento
3.0	31/10/2016	31/10/2016	Rafael Dias	• Revisão do Fluxo de Projeto de Desenvolvimento e Manutenção • Revisão do Fluxo de Manutenção • Revisão do Fluxo da Visão do Projeto • Remoção do Subprocesso de Gerenciar Mudanças • Inserção do Subprocesso de Atualizar Planejamento • Revisão do Subprocesso de Validar Contagem
3.1	24/11/2017	24/11/2017	Anne Blagojevic / Rafael Dias	• Detalhamento da Release; • Revisão das cerimônias SCRUM; • Remoção do artefato cronograma; • Inserção de um roteiro para execução das cerimônias.
3.2	02/03/2018	02/03/2018	Eder Souza / Rafael Dias	• Alteração do Item 7 – Artefatos. Retiradas as seções que discorriam sobre as referências aos artefatos na Gerência de Configuração; • Complementação do item 3.5.1 – Artefatos obrigatórios. Complementação das observações sobre alguns artefatos;

				• Inserção de seção para publicação de Documento de Diretrizes e *Checklist*; • Inserção de observação a respeito do momento que a existência da planilha de contagem deve ser verificada. • Definição de um prazo para a homologação.
3.3	23/07/2018	23/07/2018	Bruno Albuquerque	• Remoção da obrigatoriedade de utilização do Wireframe
3.4	21/06/2019	21/06/2019	Arthur Victor Ganzert / Felipe von Sperling / Nathalia Kwirant Tatagiba / Rafael de Souza Berlanda	• Ajustes nas orientações descritas no item 3.2 (Visão do Projeto); • Adequação das atividades do item 3.2.1 (Visão Agrupada da Visão do Projeto); • Alteração do Fluxo da Visão do Projeto (item 3.2.2); • Exclusão do quadro detalhado de atividades no item 3.2.3 (Detalhamento da Visão do Projeto) e inclusão do texto de conformidade com o documento de Diretrizes de Projeto de Desenvolvimento de Sistemas; • Alteração das orientações do item 3.3 (Release); • Inclusão do item 3.3.2 (Releases – revisão do planejamento); • Inclusão do item, 3.3.3 (Entregas); • Alteração das orientações descritas no item 3.4 (Sprint); • Alteração das orientações descritas no item 3.4.1 (Reunião de Planejamento da Sprint); • Alteração das orientações descritas no item 3.4.3 (Revisão da Sprint); • Alteração das orientações descritas no item 3.4.4 (Retrospectiva da Sprint); • Inclusão do item 3.4.5 (Reuniões de Refinamento);

				• Inclusão atividade "Revisar Planejamento" no item 3.4.6 (Visão Agrupada da Sprint); • Inclusão atividade "Revisar Planejamento" e exclusão da atividade "Atualizar planejamento" no quadro do item 3.4.8 (Detalhamento da Sprint); • No item 3.5.1 (Artefatos Obrigatórios), foram alterados os documentos de diretrizes do TAP e PGP, do grupo "Visão do Projeto" • Nos itens 5.2, 5.2.1 e 5.2.2, foi alterado o nome de "Atualizar Planejamento" para "Revisar Planejamento" • Alteração do Fluxo do Subprocesso Revisar Planejamento (item 5.2.1) • Exclusão do quadro detalhado de atividades no item 5.2.2 (Detalhamento do Fluxo do Subprocesso Revisar Planejamento) e inclusão do texto de conformidade com o documento de Diretrizes de Projeto de Desenvolvimento de Sistemas • No item 7 (Artefatos da Metodologia do Escritório de Projetos de TI), foi incluído o texto de conformidade com o documento de Diretrizes de Projeto de Desenvolvimento de Sistemas
3.5	14/08/2019	14/08/2019	Arthur Victor Ganzert	• Ajustes diversos.
3.6	28/10/2019	28/10/2019	Rafael de Souza Berlanda / Arthur Victor Ganzert	• Item 3.3.3: Inclusão de documentação de paralisação de projeto por meio do impedimento de uma entrega • Item 4: inclusão de tópico acerca de projetos expressos. Manutenção passa a ser o item 5

				• Atualização do sumário e dos títulos, após inclusão do tópico sobre projetos expressos
3.7	30/01/2019	30/01/2019	Arthur Victor Ganzert	• Inclusão do item 9., "Requisitos referentes aos papéis envolvidos no MDMS"
3.8	31/01/2019	31/01/2019	Arthur Victor Ganzert	• Modificação do item 9, incluindo requisito de experiência nos processos de negócio no papel de PO.
4.0	07/08/2020	15/12/2020	Arthur Victor Ganzert	• Preparação do documento para atender aos requisitos do novo Termo de Referência de Fábrica
4.1	08/03/2022	08/03/2022	Arthur Victor Ganzert	• Criação da seção "Artefatos Obrigatórios" e remoção da seção "Artefatos opcionais", referentes ao Processo de Manutenção
4.2	28/06/2024	01/07/2024	Marcelo Florencio	• Criação da seção "Boas Práticas de Desenvolvimento de Software e *Security By Design*"

Sumário

1. Introdução 15
1.1. Público alvo 15
2. Finalidade 15
3. Projeto de Desenvolvimento 15
3.1. Fluxo de Projeto de Desenvolvimento 16
3.2. Visão do Projeto 17
3.2.1. Visão Agrupada da Visão do Projeto 18
3.2.2. Fluxo da Visão do Projeto 19
3.2.3. Detalhamento da Visão do Projeto 20
3.3. *Release* 20
3.3.1. Reunião de Planejamento do *Release* 20
3.3.2. *Releases* – revisão do planejamento 21
3.3.3. Entregas 21
3.4. *Sprint* 21
3.4.1. Reunião de Planejamento da *Sprint* 22
3.4.2. Meta da *Sprint* 23
3.4.3. Revisão da *Sprint* 24
3.4.4. Retrospectiva da *Sprint* 24
3.4.5. Reuniões de Refinamento 25
3.4.6. Visão Agrupada da Sprint 26
3.4.7. Fluxo da Sprint 27
3.4.8. Detalhamento da Sprint 28
3.4.9. Roteiro para Execução das Cerimônias 30
3.5. Artefatos 30

3.5.1. Artefatos Obrigatórios 31
3.5.2. Artefatos Opcionais 32
4. Manutenção 33
4.1. Visão Agrupada da Manutenção 34
4.2. Fluxo da Manutenção 35
4.3. Detalhamento do Processo de Manutenção 36
4.4. Artefatos de Manutenção 39
4.4.1. Artefatos Obrigatórios 39
4.4.2. Artefatos Legados 41
5. Subprocessos 43
5.1. Gerenciar Detalhamento do *Backlog* do Produto 43
5.1.1. Fluxo do Subprocesso Gerenciar Detalhamento do *Backlog* do Produto 43
5.1.2. Detalhamento do Fluxo do Subprocesso Gerenciar Detalhamento do *Backlog* do Produto 43
5.2. Revisar Planejamento 43
5.2.1. Fluxo do Subprocesso Revisar Planejamento 44
5.2.2. Detalhamento do Fluxo do Subprocesso Revisar Planejamento 45
5.3. Validar Contagem 45
5.3.1. Fluxo do Subprocesso Validar Contagem 46
5.3.2. Detalhamento do Fluxo do Subprocesso Validar Contagem 47
6. Diretrizes 48
6.1. Publicação de Documento de Diretrizes e *Checklist* 48
7. Artefatos da Metodologia do Escritório de Projetos de TI 48
8. Requisitos referentes aos papéis envolvidos no MDMS 48
9. Boas Práticas de Desenvolvimento de Software e Security By Design 49
9.1. Princípios de Security By Design 50

9.1.1. Incorporação de Segurança no Ciclo de Vida do Desenvolvimento 50
9.1.2. Design Seguro 50
9.1.3. Codificação Segura 50
9.1.4. Testes de Segurança 50
9.1.5. Gestão de Vulnerabilidades 50
9.1.6. Educação e Treinamento 51
9.2. Boas Práticas de Desenvolvimento de Software 51
9.2.1. Revisão de Código 51
9.2.2. Gerenciamento de Dependências 51
9.2.3. Autenticação e Autorização 51
9.2.4. Proteção de Dados 51
9.2.5. Gestão de Configurações 51
9.3. Ferramentas e Recursos 51
9.3.1. Ferramentas de Análise de Segurança 51
9.3.2. Ferramentas de Testes de Segurança 52
9.4. Considerações Finais 52
10. Referências 52

1. Introdução

Esta metodologia visa definir os processos de desenvolvimento e manutenção de software da Superintendência de Tecnologia da Informação (STI) da Agência Nacional de Aviação Civil (ANAC).

Esses processos de desenvolvimento e manutenção detalham as fases do ciclo de vida, as principais atividades, os responsáveis envolvidos em cada etapa e os produtos gerados.

1.1. Público alvo

Este documento destina-se a todos os envolvidos no processo de desenvolvimento e manutenção de softwares da ANAC.

2. Finalidade

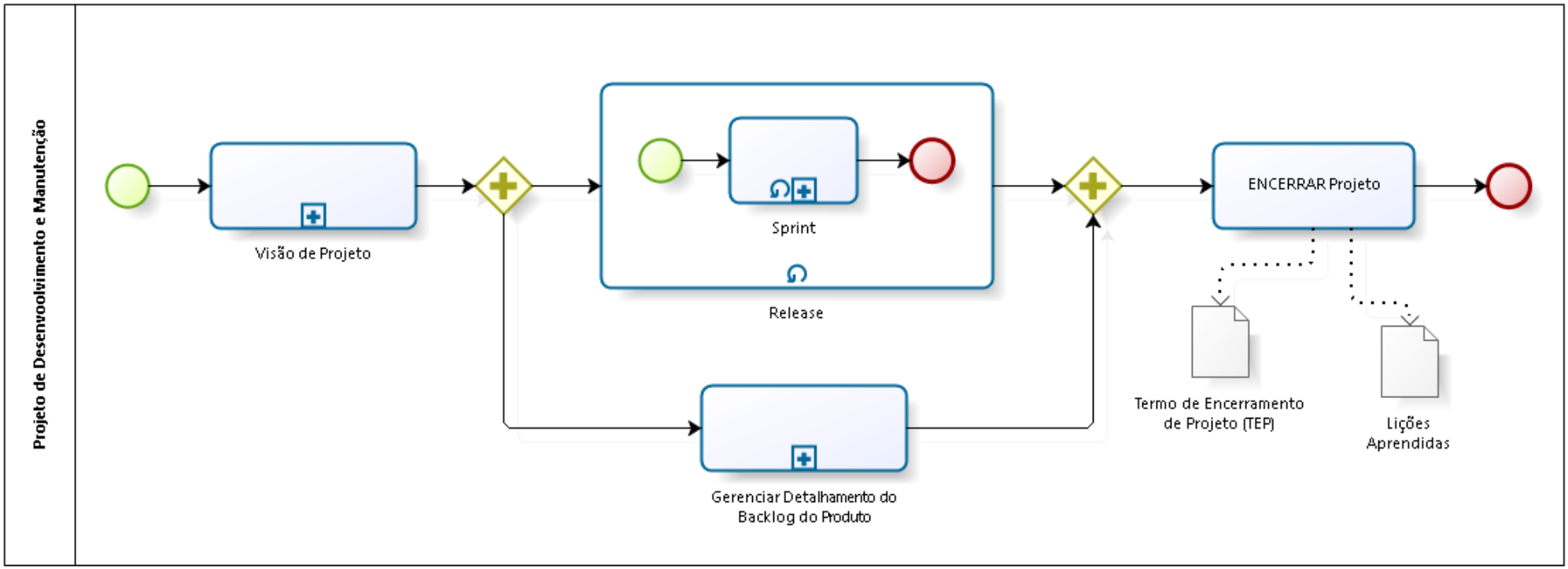
Padronizar os processos de trabalho, as funções desempenhadas e as tarefas a serem realizadas por todos os envolvidos no desenvolvimento e manutenção de software da ANAC.

3. Projeto de Desenvolvimento

Para projetos de desenvolvimento de sistemas, foram utilizadas como referências práticas Scrum e RUP. Nesta metodologia estão presentes as seguintes características:

- Desenvolvimento iterativo e incremental;
- Entrega contínua de funcionalidades;
- Gestão de mudanças;
- Verificação da qualidade;
- Definição de modelos e padrões;
- Visibilidade do planejamento;
- Modelagem visual de software; e
- Foco na arquitetura para minimizar riscos e organizar o desenvolvimento.

3.1. Fluxo de Projeto de Desenvolvimento



Powered by bizagi Modeler

3.2. Visão do Projeto

A Visão do Projeto marca o início do projeto. Os principais objetivos da Visão do Projeto são:

- Garantir etapa de ideação e designer no sentido mais amplo;
- Elaborar uma versão inicial do Backlog do Produto;
- Elaborar o termo de abertura para o projeto (TAP);
- Elaborar o planejamento inicial para o projeto (PGP); e
- Configurar o repositório, de acordo com o documento de Diretrizes de Gerência de Configuração.

É fundamental na fase de planejamento do projeto a realização de oficinas para explorar todas as possibilidades criativas e inovadoras. O design deve abranger não apenas a interface visual, mas também a experiência do usuário (UX), garantido que o produto final seja intuitivo, funcional e alinhado com as necessidades da área.

Na etapa de realização de planejamento do projeto é importante garantir a definição de MVPs (Produto Mínimo Viável) e adotar ciclos de entregas constantes e de curto prazo que ajudam a identificar rapidamente os pontos forte e as áreas que precisam de melhorias.

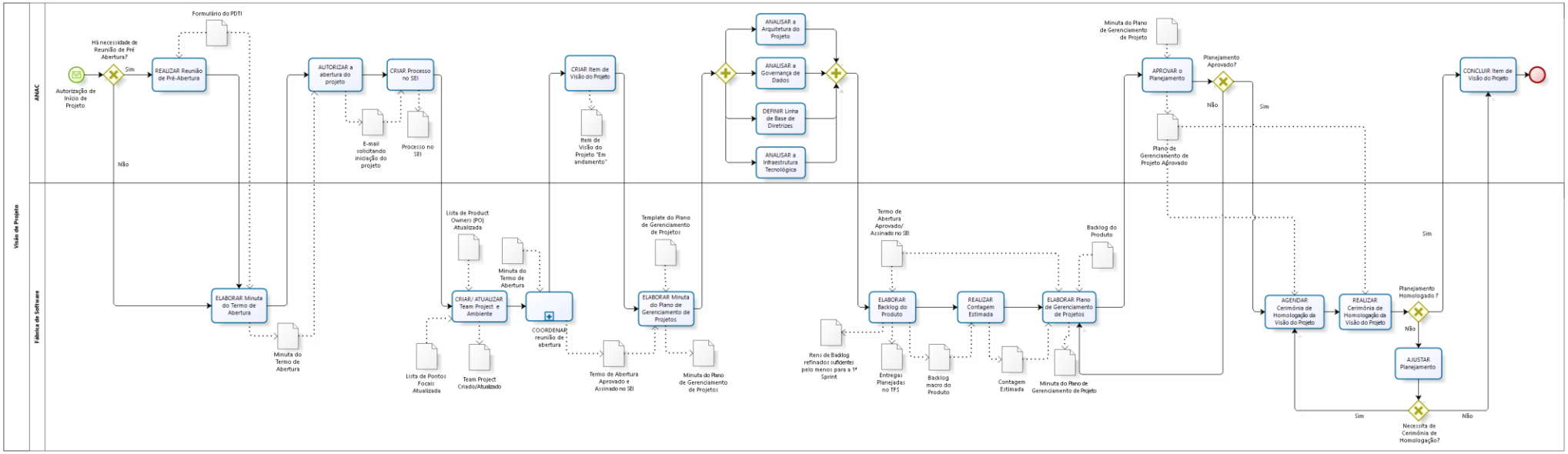
A presença dos usuários é de suma importância na construção de protótipos, assegurando maior aceitação do produto final.

Nessa etapa devem ser seguidas as diretrizes definidas no documento de Diretrizes de Projeto de Desenvolvimento de Sistemas.

3.2.1. Visão Agrupada da Visão do Projeto

	Formalização	Estruturação	Planejamento
Atividades	• Realizar Reunião de Pré-Abertura • Elaborar Minuta do Termo de Abertura • Encaminhar Iniciação do Projeto à Contratada • Criar/Atualizar Team Project • Coordenar Reunião de Abertura • Criar Item de Visão do Projeto	• Analisar a Arquitetura do Projeto • Analisar a Governança de Dados • Analisar a Infraestrutura Tecnológica • Definir a Linha de Base de Diretrizes • Elaborar Backlog do Produto • Realizar Contagem Estimada • Realizar oficina de ideação • Construção do Protótipo quando necessário	• Elaborar Plano de Gerenciamento do Projeto • Aprovar o Planejamento • Agendar Cerimônia de Homologação da Visão do Projeto • Realizar Cerimônia de Homologação da Visão do Projeto • Ajustar Planejamento • Concluir Item de Visão do Projeto

3.2.2. Fluxo da Visão do Projeto



3.2.3. Detalhamento da Visão do Projeto

O detalhamento das atividades executadas durante a Visão do Projeto, incluindo os responsáveis, entradas e saídas, é definido no documento de Diretrizes de Projeto de Desenvolvimento de Sistemas.

3.3. *Release*

As *Releases* do projeto e suas respectivas metas devem ser definidas na etapa de Visão do Projeto. Cada Release deve ser pensada de forma a atender uma necessidade de negócio, sendo capaz de entregar uma versão utilizável do produto para o usuário.

Seguem abaixo as diretrizes que devem ser seguidas para cada Release:

- A primeira *Release* do projeto deve iniciar somente após a conclusão da etapa de Visão do Projeto;
- O fiscal requisitante ou o gestor poderá autorizar início da primeira Release sem a etapa de Visão estar completa
- Cada *Release* deve conter pelo menos uma *Sprint*, que conterá um certo número de itens de *backlog*;
- Para os Níveis de Mínimos de Serviço Exigido (NMSE), serão contabilizados os itens do *Backlog* que estiverem concluídos na *Release*.
- Os itens de *Backlog* da *Release* que retornarem para o *Backlog* do Produto serão excluídos da *Sprint*, mas serão contabilizados para fins de NMSE quando forem executados.
- Itens de *Backlog* de *Releases* futuros iniciados durante o *Release* atual somente serão computados para NMSE após o início do *Release* ao qual eles pertencem.
- Itens de *Backlog* cancelados não serão contabilizados;
- O tamanho do *Release* em Ponto de Função (PF) será a soma da quantidade de PF dos itens concluídos;
- O prazo do *Release* será de acordo com o tamanho do *Release* concluída;
- O Nível Mínimo de Serviço Exigido (NMSE) de prazo do *Release* será o definido no contrato, esse prazo já contempla o tempo de retrabalho do Delta.

3.3.1. Reunião de Planejamento do *Release*

A reunião de Planejamento do *Release* deve ser realizada no início da reunião de Planejamento da primeira *Sprint* do *Release*. Nessa reunião, o planejamento definido na etapa de Visão do Projeto deve ser revisado.

3.3.2. *Releases* – revisão do planejamento

Os *releases* podem ser revisados pelo PO juntamente com a equipe do projeto a qualquer momento desde que não altere o prazo final do projeto. Caso contrário, deverá ser feito no processo de Replanejamento de Projeto.

3.3.3. Entregas

Os *releases* podem ser agrupados em entregas, que se caracterizam por constituírem pacotes que podem ser equalizados, ficando a sua manutenção endereçada à equipe de sustentação.

Apesar de uma Entrega não se caracterizar como um Item de *Backlog*, ela tem um comportamento de estados definido na Diretriz de *Backlog*.

A paralisação de um projeto pode ser sinalizada a partir do impedimento de uma ou mais entregas.

3.4. *Sprint*

A quantidade de *Sprints* será definida na etapa Visão do Projeto e será atualizada ao longo do projeto. Segue abaixo a lista de diretrizes que devem ser seguidas em cada *Sprint*:

- A primeira *Sprint* do projeto pode iniciar mesmo sem a conclusão da etapa de Visão do Projeto.
- A primeira *Sprint* deve iniciar no primeiro dia útil após a Reunião de Planejamento do Release.
- O fiscal requisitante ou o gestor poderá autorizar início da primeira Sprint sem a etapa de Visão estar completa.
- A partir da segunda *Sprint*, uma *Sprint* deve iniciar no primeiro dia útil após a data de conclusão da *Sprint* anterior.
- Uma nova *Sprint* deve iniciar somente após a conclusão da *Sprint* anterior;
- Todos os itens do topo do *Backlog* do Produto devem estar de acordo com os critérios de *Definition of Ready* da Diretriz de *Backlog*, a fim de tornarem-se itens do *Sprint Backlog*;
- Em cada *Release* deve-se desenvolver uma versão incremental do sistema que entregue valor para o usuário;
- Cada *Sprint* deverá ter duração fixa (*timeboxed*) de até 25 (vinte e cinco) dias corridos;
- O valor padrão de duração da *Sprint*, na falta de referência no PGP do projeto, é de 12 (doze) dias corridos, preferencialmente iniciando em uma segunda-feira com a *Sprint Planning*, e concluídos em uma sexta-feira com a *Sprint Review* e *Sprint Retrospective*;
- Transitoriamente, projetos ainda em andamento na ocasião da revisão dessa Metodologia

devem ter seu registro de duração de *Sprint* de referência, se diferente de 12 (doze) dias, expresso como comentário no item de visão do projeto no TFS, realizado ou pelo fiscal requisitante ou pelo gestor do contrato.

- Uma *Sprint* pode ser cancelada antes do *timebox* da *Sprint* terminar, mas seu cancelamento deve ocorrer somente a título de exceção, se solicitado pelo *Product Owner*, ou pelo Fiscal Requisitante ou pelo Gestor do Contrato
- Uma *Sprint* pode ser terminada antes do seu *timebox* terminar, no caso de desenvolvimento e homologação de todos os itens de *backlog* dessa *Sprint*. Se aprovado pelo *Product Owner* e, ou pelo Fiscal Requisitante ou pelo Gestor do Contrato (mínimo de 2 aprovadores), o início da próxima *Sprint* poderá ser antecipado, resultando ou não em um novo *Release*;
- Caso a *Sprint* seja cancelada, os itens do *Backlog* da *Sprint* que não estiverem concluídos devem retornar ao *Backlog* do Produto (não serão considerados cancelados);
- Durante a *Sprint*, os itens do *Sprint Backlog* devem ser desenvolvidos e homologados: a saída da *Sprint* concluída e bem-sucedida são itens de *backlog* no estado "Em Verificação".
- Após o termino da *Sprint*, os itens do *Backlog* da *Sprint* que não tiverem sido desenvolvidos e homologados devem retornar ao *Backlog* do Produto e não serão considerados cancelados.
- Para fins de pagamento do *Release*, todo item de *Backlog* do Produto concluído deve ter atendido no mínimo a *Definition of Done* da Diretriz de *Backlog*;
- Caso todos os itens previstos para a *Sprint* sejam desenvolvidos e postos para homologação antes do término da *Sprint*, outros itens do *Backlog* do Produto podem ser iniciados nesta *Sprint*. A não conclusão destes itens iniciados não implica em falha da *Sprint*, nem participam do cômputo de NMSE até que a *Sprint* à qual pertençam seja iniciada.
- As cerimônias de Revisão, Retrospectiva da *Sprint* atual e de Planejamento da *Sprint* seguinte devem ocorrer nesta sequência, podendo ocorrer no mesmo dia.

3.4.1. Reunião de Planejamento da *Sprint*

O trabalho a ser realizado na *Sprint* é planejado na reunião de planejamento da *Sprint* (*Sprint Planning*). Esse plano é criado com o trabalho colaborativo de todo o Time Scrum, que inclui:

- Time de desenvolvimento – profissionais da contratada que atuam nos aspectos técnicos do objeto do projeto;
- Scrum *Master* – papel exercido por profissional da contratada que inclui, dentre outras atribuições, facilitar a realização da cerimônia e remover os impedimentos encontrados

na execução da *Sprint*.

- *Product Owner* – papel exercido pelo Gestor do sistema novo ou em evolução, que inclui, dentre outras atribuições, fornecer a funcionalidade do sistema traduzido em itens de *backlog* (conforme diretriz de *backlog*), com a devida priorização de implementação;
- Outros participantes da GESI ou de empresa de apoio a gestão, de acordo com a necessidade.

A Reunião de Planejamento da *Sprint* deverá ter uma duração de até 8 (oito) horas. Nela devem ser respondidas as seguintes questões:

- O que pode ser entregue como resultado do incremento da próxima *Sprint*?
  - Durante a reunião, o Time de Desenvolvimento deve revisar as funcionalidades que serão desenvolvidas durante a *Sprint*;
  - O *Product Owner* deve discutir o objetivo da *Sprint* e os itens do *Backlog* do Produto que devem ser implementados para atingir esse objetivo;
  - O Time de Desenvolvimento deve avaliar o número de itens do *Backlog* do Produto que poderá entregar na *Sprint*;
  - O Time Scrum deve determinar a meta da *Sprint* a partir dos itens do *Backlog* do Produto previstos pelo Time de Desenvolvimento para a *Sprint*.
- Como será realizado o trabalho necessário para entregar o incremento?
  - O Time de Desenvolvimento deve decidir como construir as funcionalidades e transformá-las em um incremento de produto "Pronto" durante a *Sprint*;
  - O Time de Desenvolvimento deve planejar o trabalho a ser realizado durante a *Sprint*;
  - O *Product Owner* pode ajudar a esclarecer os itens do *Backlog* do Produto selecionados;
  - Caso o Time de Desenvolvimento determine que possui excesso ou falta de trabalho, deve renegociar os itens do *Backlog* do Produto selecionados com *Product Owner*;
  - Ao final da reunião, o Time de Desenvolvimento deve ter explicado ao *Product Owner* e ao Scrum *Master* como pretende trabalhar para cumprir a meta da *Sprint* e criar o incremento planejado.

3.4.2. Meta da *Sprint*

A Meta da *Sprint* é um objetivo definido para a *Sprint* que pode ser satisfeito através da implementação do *Backlog* do Produto. Este fornece uma direção para o Time de Desenvolvimento sobre o porquê de estar construindo o incremento.

A Meta da *Sprint* é criada durante a reunião de planejamento da *Sprint*. A Meta da *Sprint* dá ao Time de Desenvolvimento alguma flexibilidade a respeito da funcionalidade que será completada dentro dos limites da *Sprint*. Os itens do *Backlog* do Produto selecionados caracterizam a Meta da *Sprint*.

3.4.3. Revisão da *Sprint*

A Revisão da *Sprint* (*Sprint Review*) é executada no final da *Sprint* para inspecionar o incremento e adaptar o *Backlog* do Produto se necessário. Durante a reunião de Revisão da *Sprint* o Time Scrum e as partes interessadas colaboram sobre o que foi feito na *Sprint*. Esta não é uma reunião de status, e a apresentação do incremento destina-se a motivar, obter comentários e a promover a colaboração. Essa reunião deve ter uma duração (*timeboxed*) de até 4 (quatro) horas.

A Reunião de Revisão inclui os seguintes elementos:

- Os participantes incluem o Time Scrum e as partes interessadas;
- O *Product Owner* esclarece quais itens do *Backlog* do Produto estão "Prontos" e quais não estão "Prontos";
- Os itens não prontos são automaticamente transferidos para a próxima *Sprint*. Se a próxima *Sprint* for de outro *Release*, ela é transferida também para o próximo *Release*.
- O Time de Desenvolvimento demonstra em tela e de forma funcional o trabalho que está "Pronto" e responde a questões sobre o incremento;
- Os itens não homologados durante a *Sprint* deverão ser apresentados ao PO nessa cerimônia, para homologação assistida. O resultado da Reunião de Revisão da *Sprint* pode ser composto de: *Backlog* do Produto revisado; itens de *Backlog* homologados; itens de *Backlog* recusados; e itens de *backlog* não desenvolvidos e retornados ao *Backlog* do Produto;
- É também uma das atividades da Reunião de Revisão a realização da homologação de itens de *backlog* de *Sprints* anteriores que estejam pendentes de homologação.

3.4.4. Retrospectiva da *Sprint*

A Retrospectiva da *Sprint* (*Sprint Retrospective*) é uma oportunidade para o Time Scrum inspecionar a si próprio e criar um plano para melhorias a serem aplicadas na próxima *Sprint*.

A Retrospectiva da *Sprint* ocorre depois da Revisão da *Sprint* e antes da Reunião de Planejamento da próxima *Sprint*. Esta é uma reunião *timeboxed* de até 3 (três) horas.

O propósito da Retrospectiva da *Sprint* é:

- O Time de Desenvolvimento discute o que foi bem durante a *Sprint*, quais problemas ocorreram dentro da *Sprint*, e como esses problemas foram resolvidos;

- Inspecionar como a última *Sprint* foi em relação às pessoas, aos relacionamentos, aos processos e às ferramentas;
- Identificar e ordenar os principais itens que foram bem e as potenciais melhorias;
- Definir ações para implementar melhorias na forma de trabalho do Time Scrum;
- Atualizar o artefato de Lições Aprendidas.

3.4.5. Reuniões de Refinamento

Reuniões de refinamento de itens de *backlog* devem acontecer ao longo do projeto para traduzir itens de *backlog* escritos previamente em alto nível num nível de detalhamento que possibilite o desenvolvimento do item. Dela participam:

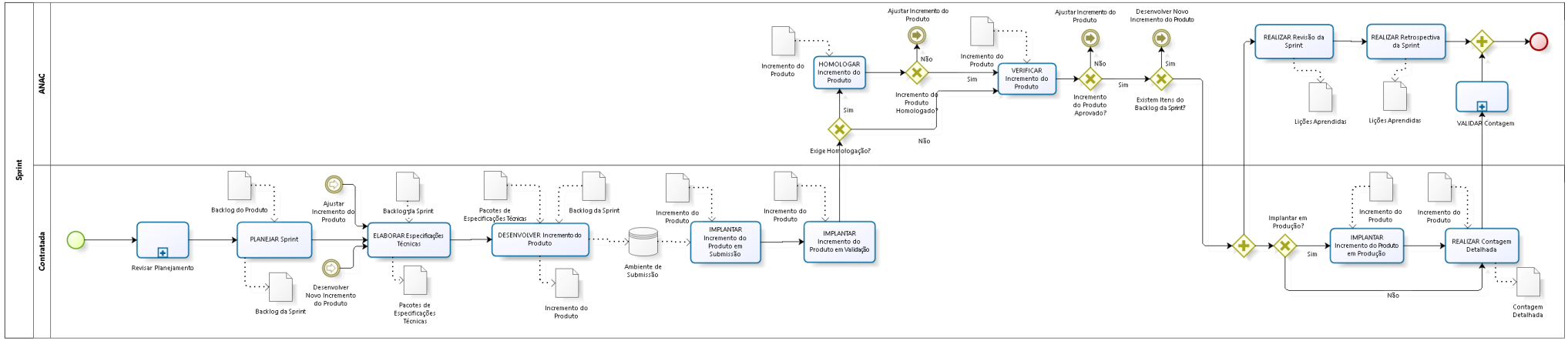
- *Product Owner* – papel exercido pelo Gestor do sistema novo ou em evolução, que inclui, entre outras atribuições, fornecer a funcionalidade do sistema traduzido em itens de *backlog* (conforme diretriz de *backlog*), com a devida priorização de implementação;
- Analista de Requisitos – profissional da empresa contratada que auxilia no detalhamento dos requisitos em itens de *backlog* de acordo com a diretriz de *backlog*
- Outros participantes da GESI ou de empresa de apoio a gestão, de acordo com a necessidade.

O resultado da Reunião de Refinamento é um *Backlog* do Produto revisado, com itens de *backlog* refinados.

3.4.6. Visão Agrupada da Sprint

	Planejamento	Especificação	Desenvolvimento	Fechamento
Atividades	• Revisar Planejamento • Planejar *Sprint*	• Elaborar Especificações Técnicas	• Desenvolver Incremento do Produto • Implantar Incremento do Produto em Submissão • Implantar Incremento do Produto em Validação • Homologar Incremento do Produto • Verificar Incremento do Produto • Implantar Incremento do Produto em Produção • Realizar Revisão da *Sprint* • Realizar Retrospectiva da *Sprint*	• Realizar Contagem Detalhada • Validar Contagem

3.4.7. Fluxo da Sprint



3.4.8. Detalhamento da Sprint

Atividade	Descrição	Responsável	Entradas	Saídas
Revisar Planejamento	Subprocesso que abrange as atividades requeridas para atualizar o planejamento. Ver item: Revisar Planejamento	-	-	-
Planejar Sprint	Atividade para planejar a Sprint. Nessa atividade, ocorre a Reunião de Planejamento da Sprint.		• Backlog do Produto	• Backlog da Sprint
Elaborar Especificações Técnicas	Atividade para elaborar as especificações técnicas referentes a funcionalidade. **Para a elaboração das especificações técnicas, deve ocorrer uma reunião com a equipe técnica definida pela ANAC.**	Contratada	• Backlog da Sprint	Um Pacote de Especificações Técnicas deve conter, no mínimo, os seguintes artefatos: • Modelo de Dados Físico • Documento de Arquitetura
Desenvolver Incremento do Produto	Atividade para desenvolver o Incremento do Produto referente a funcionalidade.	Contratada	• Backlog da Sprint • Pacote de Especificações Técnicas	Um Incremento do Produto deve conter, no mínimo, os seguintes artefatos: • Backlog da Sprint • Pacote de Especificações Técnicas • Testes Automatizados • Roteiros de Testes • Código fonte • Manual do Usuário • Scripts de Banco de Dados

Implantar Incremento do Produto em Submissão	Atividade para implantar a Sprint no ambiente de submissão. Essa atividade contempla todas a ações necessárias para garantir que a versão incremental entregue do sistema esteja pronta para ser utilizada no ambiente. Por esse motivo, nessa atividade todos os testes especificados devem ser reexecutados no ambiente.	Contratada	• Incremento do Produto	-
Implantar Incremento do Produto em Validação	Atividade para implantar Incremento do Produto no ambiente de validação. Essa atividade contempla todas a ações necessárias para garantir que o Incremento do Produto esteja pronta para ser utilizada no ambiente. Por esse motivo, nessa atividade todos os testes especificados devem ser reexecutados no ambiente.	Contratada	• Incremento do Produto	-
Homologar Incremento do Produto	Atividade para homologar o Incremento do Produto. O prazo para homologar o incremento é de 3 dias úteis. Em caso de atraso na homologação maior que 2 dias úteis, a homologação será escalonada para chefe da lotação. Se o chefe não atuar em até 2 dias úteis, o Incremento do Produto será considerado homologado por decurso de prazo e o início da próxima Sprint deverá ser avaliado.	ANAC	-	-

Verificar Incremento do Produto	Atividade para verificar os artefatos do Incremento do Produto.	ANAC	• Incremento do Produto	-
Implantar Incremento do Produto em Produção	Atividade para implantar Sprint em produção. Essa atividade contempla todas a ações necessárias para garantir que Incremento do Produto esteja pronto para ser utilizado no ambiente. Por esse motivo, nessa atividade todos os testes especificados devem ser reexecutados no ambiente.	Contratada	• Incremento do Produto	-
Realizar Contagem Detalhada	Atividade para realizar contagem detalhada dos pontos de função.	Contratada	• Incremento do Produto	• Contagem Detalhada
Validar Contagem	Subprocesso que abrange as atividades requeridas para validar a contagem dos pontos de função. Ver item: Validar Contagem	-	-	-
Realizar Revisão da Sprint	Atividade para realizar revisão da Sprint.	ANAC	-	• Lições Aprendidas
Realizar Retrospectiva da Sprint	Atividade para realizar retrospectiva da Sprint.	ANAC	-	• Lições Aprendidas

3.4.9. Roteiro para Execução das Cerimônias

Durante as cerimônias SCRUM deve ser seguido o roteiro apresentado no documento Roteiro de Execução das Cerimônias.

3.5. Artefatos

As diretrizes para elaboração de alguns artefatos estão detalhadas nos documentos de Diretrizes, conforme detalhado no quadro abaixo.

3.5.1. Artefatos Obrigatórios

Atividade	Artefato	Observação	Documento de Diretrizes
Visão do Projeto	*Backlog* do Produto	Ver o Documento de Diretrizes de *Backlog*.	*Backlog*
	Planilha de Contagem (Contagem Estimada)	A verificação da existência desse artefato deve ser realizada na Etapa de Fechamento.	Métricas
	Termo de Abertura de Projeto (TAP)	Deve ser incluído o número do processo no SEI onde está o documento	Projetos de Desenvolvimento de Sistemas
	Plano de Gerenciamento de Projeto (PGP)	Deve ser incluído o número do processo no SEI onde está o documento	Projetos de Desenvolvimento de Sistemas
Sprint	História de Usuário	Esse artefato será utilizado para detalhar cada item de *Backlog* do Produto. Em regra, o detalhamento do item do *Backlog* do Produto deve ocorrer no Subprocesso Gerenciar Detalhamento do Backlog do Produto. Ver documento de Diretrizes de *Backlog*.	*Backlog*
	Modelo de Dados Físico	Ver Documento de Diretrizes de Banco de Dados.	Banco de Dados
	Documento de Arquitetura		Arquitetura
	Especificação da API	Deve ser entregue caso a História de Usuário seja referente a um serviço.	*Backlog*
	Testes Funcionais Automatizados	Caso a manutenção impacte a cobertura e/ou quebre os testes automatizados existentes, esses testes devem ser atualizados. A Massa de Testes deverá ser entregue juntamente com os Testes Automatizados.	Testes
	Testes Unitários Automatizados	Caso a manutenção impacte a cobertura e/ou quebre os testes automatizados existentes, esses testes devem ser atualizados.	Testes
	Projeto de Testes de *WebServices*	Caso a História de Usuário seja referente a um serviço, deve ser criado um Projeto de	Testes

		Testes de *WebServices* e vinculado à História de Usuário.	
	Código Fonte		
	Manual do Usuário		
	Planilha de Contagem (Contagem Detalhada)	A verificação da existência desse artefato deve ser realizada na Etapa de Fechamento.	Métricas
	Análise de Impacto		Métricas
	Lições Aprendidas		

3.5.2. Artefatos Opcionais

Artefato	Observação	Documento de Diretrizes
Diagrama de Classe	Deve ser entregue somente se for solicitado pela equipe de Arquitetura da ANAC.	Arquitetura
Diagrama de Sequência	Deve ser entregue somente se for solicitado pela equipe de Arquitetura da ANAC.	Arquitetura
Testes de Carga	Caso seja necessária a criação de testes de carga, devem ser criados os testes de carga de acordo com o Documento de Diretrizes de Testes.	Testes
Testes Manuais	Caso seja necessária a criação de testes manuais, devem ser criados os testes manuais no TFS de acordo com o Documento de Diretrizes de Testes.	Testes
Scripts de Banco de Dados	Deve ser entregue caso seja necessária a execução de ações definidas no Documento de Diretrizes de Banco de Dados.	Banco de Dados
WireFrame	Pode ser entregue caso a História de Usuário seja referente a uma tela.	*Backlog*

4. Manutenção

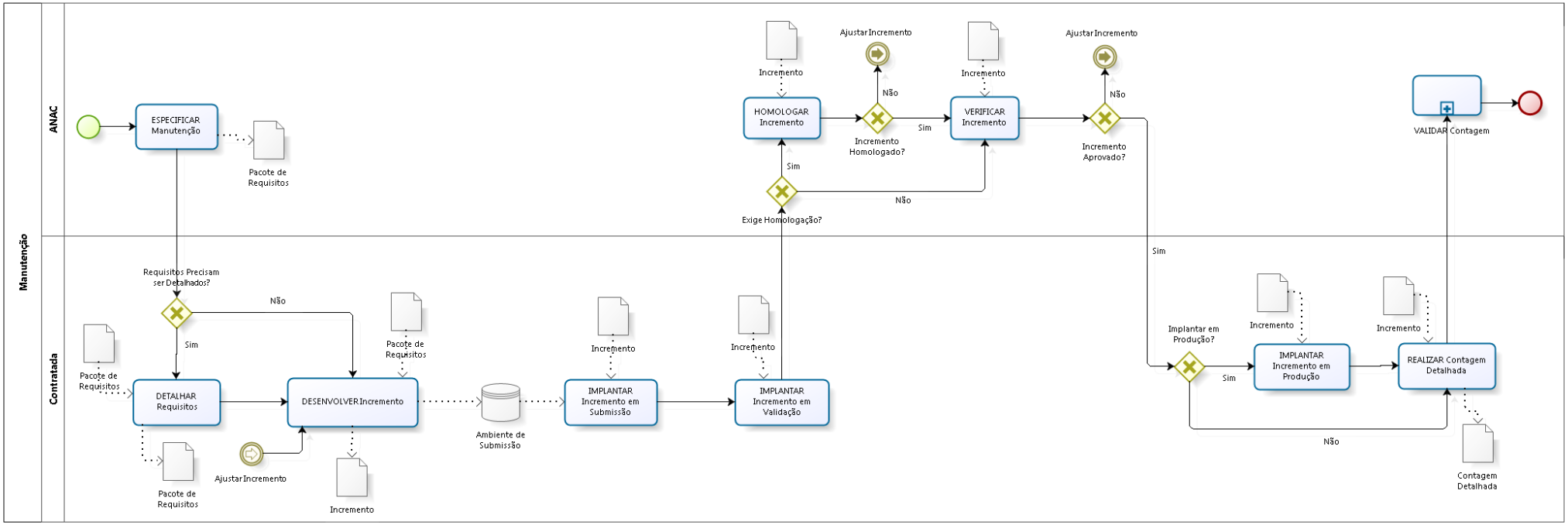
Para o processo de Manutenção, foram utilizados como referência práticas de Scrum, RUP e Kanban. Para o ciclo de vida das demandas de Manutenção, foram adotadas as seguintes características:

- Desenvolvimento iterativo e incremental;
- Entrega contínua de funcionalidades;
- Verificação da qualidade;
- Definição de modelos e padrões;
- Modelagem visual de software; e
- Foco na arquitetura para minimizar riscos e organizar o desenvolvimento.

4.1. Visão Agrupada da Manutenção

	Especificação	Desenvolvimento	Fechamento
Atividades	• Especificar Manutenção • Detalhar Requisitos	• Desenvolver Incremento • Implantar Incremento em Submissão • Implantar Incremento em Validação • Homologar Incremento • Verificar Incremento • Implantar Incremento em Produção	• Realizar Contagem Detalhada • Validar Contagem

4.2. Fluxo da Manutenção



4.3. Detalhamento do Processo de Manutenção

Atividade	Descrição	Responsável	Entradas	Saídas
Especificar Manutenção	Atividade para especificar a manutenção.	ANAC	-	No Pacote de Requisitos deve conter, no mínimo o seguinte artefato: • História de Usuário
Detalhar Requisitos	Atividades para detalhar os requisitos	Contratada	• Pacote de Requisitos	No Pacote de Requisitos deve conter, no mínimo o seguinte artefato: • História de Usuário
Desenvolver Incremento	Atividade para desenvolver o Incremento da manutenção.	Contratada	• Pacote de Requisitos	O Incremento deve conter, no mínimo, os seguintes artefatos: • Pacote de Requisitos • Roteiros de Testes • Testes Automatizados • Código Fonte • Scripts de Banco de Dados • Artefatos Atualizados • Contagem Detalhada
Implantar Incremento em Submissão	Atividade para implantar o Incremento no ambiente de submissão. Essa atividade contempla todas a ações necessárias para garantir que a versão incremental entregue do sistema esteja pronta para ser utilizada no ambiente. Por esse motivo, nessa atividade todos os testes especificados devem ser reexecutados no ambiente.	Contratada	• Incremento	-

Implantar Incremento em Validação	Atividade para implantar Incremento no ambiente de validação. Essa atividade contempla todas a ações necessárias para garantir que a versão incremental entregue do sistema esteja pronta para ser utilizada no ambiente. Por esse motivo, nessa atividade todos os testes especificados devem ser reexecutados no ambiente.	Contratada	• Incremento	-
Homologar Incremento	Atividade para homologar o Incremento. O prazo para homologar o incremento é de 3 dias úteis. Em caso de atraso na homologação maior que 2 dias úteis, a homologação será escalonada para chefe da lotação. Se o chefe não atuar em até 2 dias úteis, a demanda será considerada homologada por decurso de prazo e futuras demandas deverão ser analisadas com maior cautela em relação ao seu atendimento.	ANAC	• Incremento	-
Verificar Incremento	Atividade para verificar os artefatos do incremento.	ANAC	• Incremento	-

Implantar Incremento em Produção	Atividade para implantar Incremento no ambiente de produção. Essa atividade contempla todas a ações necessárias para garantir que a versão incremental entregue do sistema esteja pronta para ser utilizada no ambiente. Por esse motivo, nessa atividade todos os testes especificados devem ser reexecutados no ambiente.	Contratada	• Incremento	-
Realizar Contagem Detalhada	Atividade para realizar contagem detalhada dos pontos de função.	Contratada	• Incremento	• Contagem Detalhada
Validar Contagem	Subprocesso que abrange as atividades requeridas para validar a contagem dos pontos de função. Ver item: Validar Contagem		-	-

4.4. Artefatos de Manutenção

4.4.1. Artefatos Obrigatórios

	Evidências no item de Backlog	Gerência de Configuração	Testes Automatizados	Documentaç
Solução de contorno	- Descrição pelo requisitante no histórico do item de backlog que a Solução de contorno atende a demanda;	**Todos** - Existência de Pull Request; - Código versionado no GIT na branch correta; **Quando não for necessário Pull Request** - Justificativa para a ausência de Pull Request no histórico do item de backlog; **Quando houver testes legados** - Testes feitos com SoapUI estão versionados na pasta "Legado";	**Testes Manuais** - Alteração de UI: Gravação do teste utilizado para evidenciar manutenção realizada, versionada no repositório - Alteração em Microserviço: Gravação da execução do teste no postman, versionada no repositório - Massa de testes entregue e versionada no GIT - Testes manuais inseridos no TFS e vincuados ao Item de backlog **Demais testes** - Observar Diretrizes de Testes	- Motivo c campo Caus
Gerência de Configuração e Ambiente	- Descrição dos itens de configuração alterados ou reexecutados.	Não se aplica	Não se aplica	Evidência verificação. alguns exem restringir - evidência - número do infra (citsm
Apuração Especial		**Todos** - Existência de Pull Request; - Código versionado no GIT na branch correta; **Quando não for necessário Pull Request** - Justificativa para a ausência de Pull Request no histórico do item de backlog; **Quando houver testes legados** - Testes feitos com SoapUI estão versionados na pasta "Legado";		- Script ver GIT e/ou a item de bacl

Manutenção Evolutiva	**Quando houver necessidade de testes** - Apontamento dos testes criados;	**Todos** - Existência de Pull Request; - Código versionado no GIT na branch correta; **Quando não for necessário Pull Request** - Justificativa para a ausência de Pull Request no histórico do item de backlog; **Quando houver testes legados** - Testes feitos com SoapUI estão versionados na pasta "Legado";	**Testes Manuais** - Alteração de UI: Gravação do teste utilizado para evidenciar manutenção realizada, versionada no repositório; - Alteração em Microserviço: Gravação da execução do teste no postman, versionada no repositório; - Massa de testes entregue e versionada no GIT; - Manutenção em Serviço: Projeto de testes de WebServices criado e vinculado a História de usuário; - Testes manuais inseridos no TFS e vincuados ao Item de backlog **Demais testes** - Observar Diretrizes de Testes	- Do versionada associada a backlog; - Evidê atualização arquitetural dados comportam sistema;
Manutenção Adaptativa	**Quando houver necessidade de testes** - Apontamento dos testes criados;	**Todos** - Existência de Pull Request; - Código versionado no GIT na branch correta; **Quando não for necessário Pull Request** - Justificativa para a ausência de Pull Request no histórico do item de backlog; **Quando houver testes legados** - Testes feitos com SoapUI estão versionados na pasta "Legado";	**Testes Manuais** - Alteração de UI: Gravação do teste utilizado para evidenciar manutenção realizada, versionada no repositório - Alteração em Microserviço: Gravação da execução do teste no postman, versionada no repositório - Massa de testes entregue e versionada no GIT - Testes manuais inseridos no TFS e vincuados ao Item de backlog **Demais testes** - Observar Diretrizes de Testes	- Do versionada associada a backlog; - Evidê atualização arquitetural dados comportam sistema;
Manutenção corretiva e Manutenção de garantia	**Quando houver necessidade de testes** - Apontamento dos testes criados;	**Todos** - Existência de Pull Request; - Código versionado no GIT na branch correta; **Quando não for necessário Pull Request** - Justificativa para a ausência de Pull Request no histórico do item de backlog; **Quando houver testes legados** - Testes feitos com SoapUI estão versionados na pasta "Legado";	**Testes Manuais** - Alteração de UI: Gravação do teste utilizado para evidenciar manutenção realizada, versionada no repositório - Alteração em Microserviço: Gravação da execução do teste no postman, versionada no repositório - Massa de testes entregue e versionada no GIT - Testes manuais inseridos no TFS e vincuados ao Item de backlog **Demais testes** - Observar Diretrizes de Testes	- Do versionada associada a backlog; - Evidê atualização arquitetural dados comportam sistema;

Internalização de Software	- Atender diretriz de Equalização	- Atender diretriz de Equalização	- Atender diretriz de Equalização	- Docum implantação no - Atender Equlização
Desenvolvimento de Software Novo	Observar Diretrizes de Backlog	**Todos** - Existência de Pull Request; - Código versionado no GIT na branch correta;	Observar Diretrizes de Testes	- Do versionada associada a backlog; - Evidê atualização arquitetural dados comportam sistema;
Manutenção de Documentação	Não se aplica	Não se aplica	Não se aplica	- Do versionada associada a backlog; - Evidê atualização arquitetural dados comportam sistema;
Verificação de Erros	Não se aplica	Não se aplica	Não se aplica	Evidência verificação. alguns exem restringir - chamado - caus devidament preenchida - indicação de item de b correção identificado
Desenvolvimento de Testes Automatizados	Não se aplica	**Todos** - Existência de Pull Request; - Código versionado no GIT na branch correta;	Observar Diretrizes de Testes	Não se aplic

4.4.2. Artefatos Legados

Caso um sistema possua artefatos que não estejam em conformidade com os modelos vigentes na MDMS, esses artefatos legados deverão ser atualizados para os modelos da MDMS vigente.

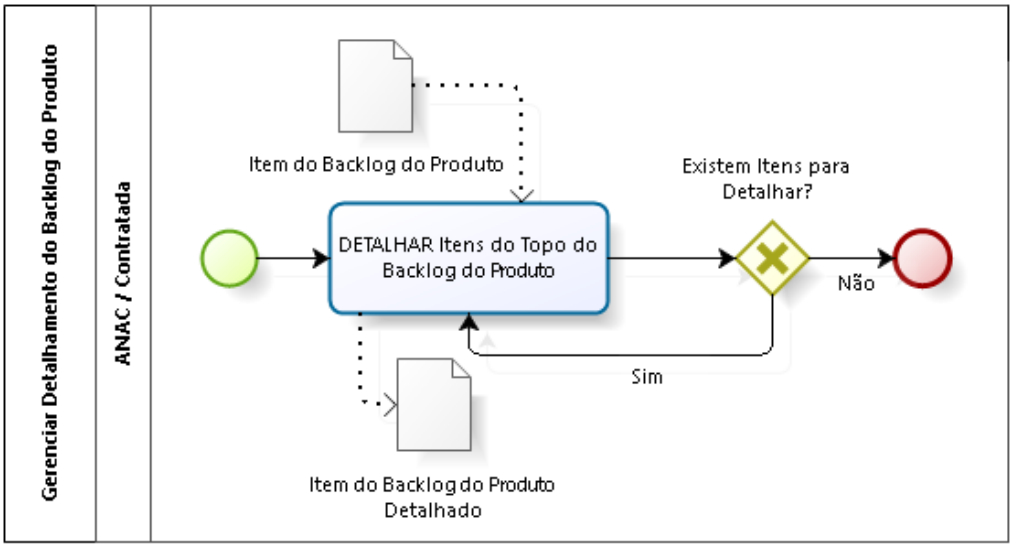
Todas as funcionalidades que forem escopo de uma manutenção deverão ter sua documentação convertida para modelos de artefatos da MDMS vigente.

5. Subprocessos

5.1. Gerenciar Detalhamento do *Backlog* do Produto

O subprocesso Gerenciar Detalhamento do *Backlog* do Produto visa assegurar que o controle das mudanças seja tratado adequadamente. O objetivo principal desse subprocesso é Gerenciar o detalhamento dos itens do *Backlog* do Produto.

5.1.1. Fluxo do Subprocesso Gerenciar Detalhamento do *Backlog* do Produto



5.1.2. Detalhamento do Fluxo do Subprocesso Gerenciar Detalhamento do *Backlog* do Produto

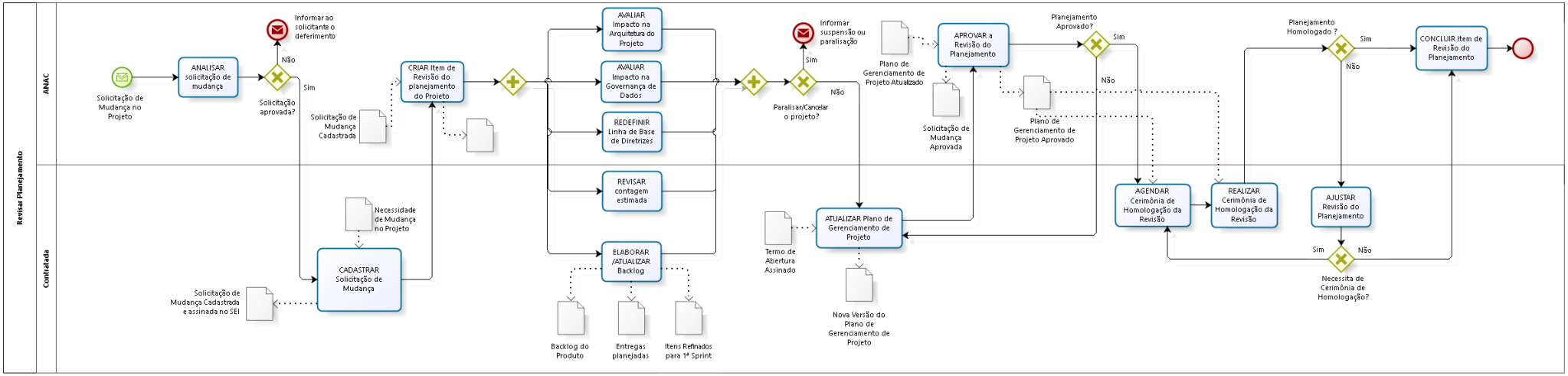
Atividade	Descrição	Responsável	Entradas	Saídas
Detalhar Itens do Topo do *Backlog* do Produto	Atividade para detalhar Itens do topo do *Backlog* do Produto	ANAC e Contratada	Item do *Backlog* do Produto	Item do *Backlog* do Produto Detalhado

5.2. Revisar Planejamento

O subprocesso Revisar Planejamento visa assegurar que a atualização do planejamento seja tratada adequadamente. Os principais objetivos do subprocesso são:

- Monitorar as mudanças do projeto;
- Monitorar as atualizações de planejamento.

5.2.1. Fluxo do Subprocesso Revisar Planejamento



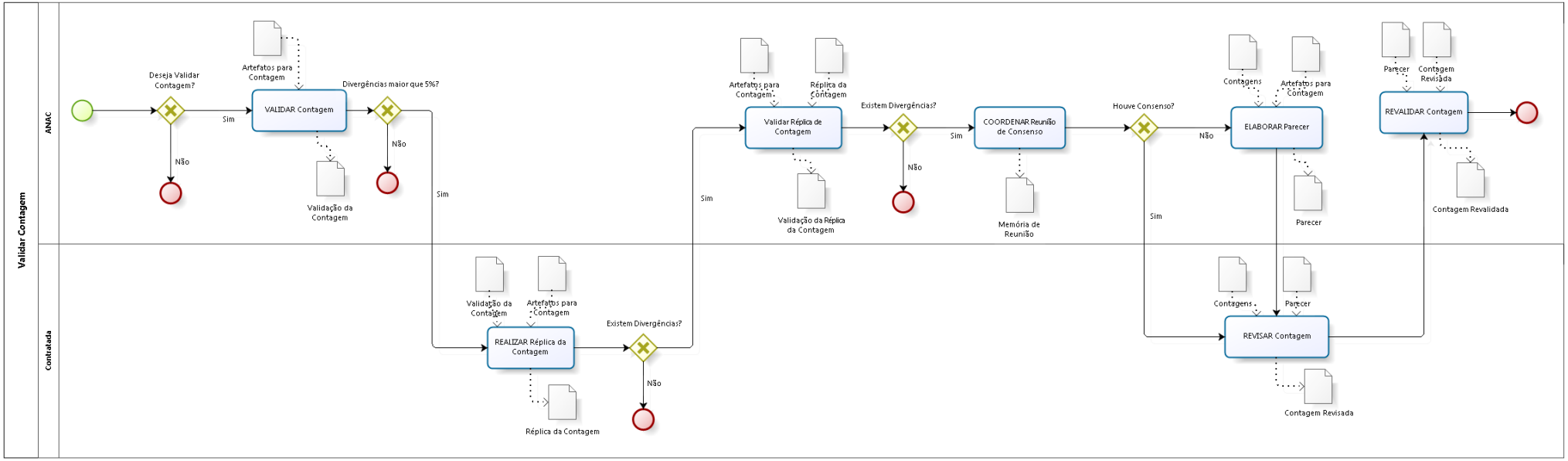
5.2.2. Detalhamento do Fluxo do Subprocesso Revisar Planejamento

O detalhamento das atividades executadas durante o subprocesso Revisar Planejamento, incluindo os responsáveis, entradas e saídas, é definido no documento de Diretrizes de Projeto de Desenvolvimento de Sistemas.

5.3. Validar Contagem

O objetivo do subprocesso é definir a forma de validação da contagem.

5.3.1. Fluxo do Subprocesso Validar Contagem



5.3.2. Detalhamento do Fluxo do Subprocesso Validar Contagem

Atividade	Descrição	Responsável	Entradas	Saídas
Realizar Contagem	Atividade para realizar a contagem.	Contratada	• Artefatos para Contagem	• Contagem
Validar Contagem	Atividade para validar a contagem.	ANAC	• Contagem • Artefatos para Contagem	• Validação da Contagem
Realizar Réplica da Contagem	Atividade para realizar a réplica da contagem validada.	Contratada	• Validação da Contagem • Artefatos para Contagem	• Réplica da Contagem
Validar Réplica de Contagem	Atividade para validar a réplica da contagem	ANAC	• Réplica da Contagem • Artefatos para Contagem	• Validação da Réplica da Contagem
Coordenar Reunião de Consenso	Atividade para coordenar a reunião de consenso entre contador e validador.	ANAC	-	• Memória de Reunião
Elaborar Parecer	Atividade para elaborar o parecer a respeito da divergência de contagem.	ANAC	• Contagens • Artefatos para Contagem	• Parecer
Revisar Contagem	Atividade para revisar a contagem após a emissão do parecer.	Contratada	• Parecer • Contagens	• Contagem Revisada
Revalidar Contagem	Atividade para validar a contagem revisada.	ANAC	• Parecer • Contagem Revisada	• Contagem Revalidada

6. Diretrizes

- As diretrizes a serem utilizadas em conjunto com a MDMS serão as vigentes no momento de execução da Manutenção e do Projeto.

6.1. Publicação de Documento de Diretrizes e *Checklist*

- Após toda criação e elaboração de **Documento de Diretrizes/*Checklist***, as Contratadas que serão impactadas devem ser notificadas por e-mail;
- Após a notificação, as contratadas terão 5 dias úteis para enviar considerações via e-mail;
- Após o recebimento das considerações, a ANAC agendará uma reunião para apresentação da versão final do **Documento de Diretrizes/*Checklist***;
- O **Documento de Diretrizes/*Checklist*** alterado/criado entrará em vigor após 30 dias da realização da reunião;
- Caso não ocorra a reunião, os 30 dias contarão a partir da notificação.

7. Artefatos da Metodologia do Escritório de Projetos de TI

Para projetos de desenvolvimento ou manutenção de sistemas, a MDMS deve utilizar alguns modelos de artefatos da Metodologia do Escritório de Projetos de TI. As versões dos modelos a serem utilizados nos projetos serão as vigentes conforme Documento de Diretrizes de Projeto de Desenvolvimento de Sistemas. Os modelos dos artefatos a serem utilizados são:

Artefato
Termo de Abertura de Projeto (TAP)
Plano de Gerenciamento de Projeto (PGP)
Solicitação de Mudança
Termo de Encerramento de Projeto (TEP)
Lições Aprendidas

8. Requisitos referentes aos papéis envolvidos no MDMS

Os principais papéis envolvidos com o desenvolvimento e sustentação de sistemas e seus requisitos de capacitação estão descritos a seguir:

- Product Owner (PO)– papel normalmente exercido pelo Gestor do Sistema, designado em portaria. O PO deve ter:
    - experiência mínima de seis (6) meses nos processos de negócios relacionados com o sistema em questão, exceto quando o objeto do sistema é um processo novo dentro da agência;

- Conhecimento e experiência em desenvolvimento ágil, que deve incluir framework SCRUM e o papel de PO. O conhecimento pode ser formal ou através de experiência prática no trabalho. No caso de ausência de conhecimento formal prévio do papel do PO no desenvolvimento ágil, a área de facilitação da GESI deverá prover uma capacitação no processo básico e ferramentas através de reunião, e, no prazo de no máximo 6 meses, o PO deverá ser treinado formalmente em curso disponibilizado pela ANAC ou externo que abranja o desenvolvimento ágil e o papel do PO.

- Time de desenvolvimento – papel desempenhado por um ou mais pessoas, com perfis de desenvolvedor, administrador de dados, analista de gestão de configuração, analista de requisitos dentre outras necessárias no ciclo de desenvolvimento de um sistema. A responsabilidade pela capacitação do time é da empresa prestadora de serviços de desenvolvimento e sustentação de software.
- Scrum Master – papel desempenhado por pessoa da empresa prestadora de serviços com experiência em desenvolvimento ágil neste papel. A responsabilidade pela capacitação do Scrum Master é da empresa prestadora de serviços de desenvolvimento e sustentação de software.
- Partes Interessadas (stakeholders) – papel desempenhado por todas as pessoas que são impactadas de alguma forma pelo sistema criado ou desenvolvido. Não requer capacitação específica, cabendo aos participantes do projeto esclarecimentos do processo se e quando necessário.
- Profissional da empresa de apoio a gestão – papel desempenhado por pessoa da empresa de apoio a gestão para atuação no apoio à GESI na remoção de impedimentos e na gestão dos projetos e sustentação de sistemas, com experiência na gestão de projetos convencionais e ágeis. A responsabilidade pela capacitação dos profissionais da empresa de apoio à gestão é da empresa contratada para tal.

9. Boas Práticas de Desenvolvimento de Software e Security By Design

Esta seção aborda as boas práticas de desenvolvimento de software com foco em segurança, incluindo a adoção da metodologia Security By Design. O objetivo é garantir que a segurança seja uma parte

integral do processo de desenvolvimento, desde o planejamento inicial até a manutenção contínua do software.

Devem ser seguidas as diretrizes definidas no documento de Diretrizes de Vulnerabilidade de Software, bem com o aplicar o Processo de Gerenciamento de Vulnerabilidade de Software definidos para a organização.

9.1. Princípios de Security By Design

9.1.1. Incorporação de Segurança no Ciclo de Vida do Desenvolvimento

As equipes de desenvolvimento devem integrar atividades de segurança em todas as fases do ciclo de vida do desenvolvimento de software, desde o planejamento, análise de requisitos, design, implementação, testes, até a manutenção.

Também devem realizar análises de risco e avaliações de segurança durante as etapas de planejamento e design em conjunto com a equipe de segurança da informação.

9.1.2. Design Seguro

As equipes envolvidas no processo de desenvolvimento devem observar e adotar padrões de design seguro, como a separação de privilégios, defesa em profundidade, e princípio do menor privilégio.

Membros da equipe devem realizar revisões de design para identificar e mitigar potenciais ameaças e vulnerabilidades antes da implementação. O uso de checklists de segurança durante as revisões ajuda a garantir que aspectos críticos de segurança sejam abordados.

9.1.3. Codificação Segura

Implementar diretrizes de codificação segura para prevenir vulnerabilidades comuns como injeção de SQL, XSS e buffer overflow.

Utilizar ferramentas de análise estática e dinâmica para identificar problemas de segurança durante a codificação.

Revisar o código regularmente, utilizando checklists de segurança e *pair programming*.

9.1.4. Testes de Segurança

Integrar testes de segurança nos processos de desenvolvimento e integração contínua (CI/CD).

Realizar testes de penetração, testes de *fuzzing* e varreduras de vulnerabilidade periodicamente.

Utilizar técnicas de teste automatizadas para assegurar que as novas mudanças não introduzam vulnerabilidades.

9.1.5. Gestão de Vulnerabilidades

Estabelecer um processo robusto de identificação, avaliação, priorização e remediação de vulnerabilidades.

Monitorar continuamente fontes de informação de segurança para se manter atualizado sobre novas ameaças e vulnerabilidades.

Implementar patches de segurança de forma oportuna e verificar as correções aplicadas.

9.1.6. Educação e Treinamento

Promover a conscientização sobre segurança entre todos os membros da equipe de desenvolvimento.

Realizar treinamentos regulares sobre práticas de desenvolvimento seguro e ameaças emergentes.

9.2. Boas Práticas de Desenvolvimento de Software

9.2.1. Revisão de Código

Estabelecer um processo estruturado de revisão de código que inclua verificações de segurança.

Utilizar checklists específicos de segurança durante as revisões para assegurar que todas as vulnerabilidades potenciais sejam identificadas.

9.2.2. Gerenciamento de Dependências

Utilizar ferramentas para monitorar e gerenciar dependências de software, garantindo que todas estejam atualizadas e livres de vulnerabilidades conhecidas.

Adotar uma política de atualização regular de bibliotecas e frameworks.

9.2.3. Autenticação e Autorização

Implementar mecanismos robustos de autenticação e autorização, utilizando padrões como OAuth, OpenID Connect e JWT.

Garantir que todas as operações críticas sejam autorizadas de acordo com os princípios de segurança.

9.2.4. Proteção de Dados

Implementar criptografia de dados em repouso e em trânsito utilizando algoritmos seguros.

Adotar políticas de gestão de chaves e certificações digitais para proteger dados sensíveis.

9.2.5. Gestão de Configurações

Utilizar ferramentas de gerenciamento de configuração para garantir que os ambientes de desenvolvimento, teste e produção sejam configurados de forma segura e consistente.

Revisar e validar as configurações regularmente para detectar e corrigir possíveis falhas.

9.3. Ferramentas e Recursos

9.3.1. Ferramentas de Análise de Segurança

Análise Estática de Código: SonarQube, OWASP Dependency-Check, Bandit.

Análise Dinâmica de Código: OWASP ZAP, Burp Suite, Nessus.

9.3.2. Ferramentas de Testes de Segurança

Testes de Penetração: Metasploit, Acunetix.

Monitoramento de Vulnerabilidades: OpenVAS, Nexpose.

9.4. Considerações Finais

Adotar boas práticas de desenvolvimento de software e a metodologia *Security By Design* é essencial para garantir a criação de sistemas seguros e confiáveis. A integração de segurança em todas as fases do ciclo de vida do desenvolvimento ajuda a mitigar riscos, proteger dados sensíveis e fortalecer a confiança dos usuários e clientes.

10. Referências

- OpenUP - Processo iterativo para projeto e desenvolvimento de software;
- RUP – disciplinas e fases do RUP;
- PMBOK – Guia de boas práticas de gerenciamento de projetos;
- Metodologia do Escritório de Projetos de TI da ANAC;
- SCRUM – Metodologia ágil de gerenciamento de projetos.
- Guia de Gerenciamento de Vulnerabilidades da SGD/MGI
- Guia de Requisitos Mínimos de Privacidade e Segurança da Informação para Aplicações Web da SGD/MGI