



# METODOLOGIA ÁGIL DE DESENVOLVIMENTO E SUSTENTAÇÃO DE SOFTWARE MDS-ÁGIL

# CONTROLE DE REVISÕES

Data	Versão	Descrição	Autor
25/11/2021	0.1	Revisão para modelo ágil dos processos de trabalho e texto em geral	STIC/MEC

# SUMÁRIO



INTRODUÇÃO .....	4
CLASSIFICAÇÃO DAS DEMANDAS .....	11
PROCESSO DE DESENVOLVIMENTO E MANUTENÇÃO.....	14
PAPÉIS E RESPONSABILIDADES.....	27
ARTEFATOS .....	39
PADRÕES DE DESENVOLVIMENTO .....	54
FERRAMENTAS DE APOIO.....	56
REFERÊNCIAS BIBLIOGRÁFICAS.....	60



# INTRODUÇÃO

A Metodologia Ágil de Desenvolvimento de Software do Ministério da Educação define os processos e atividades dos serviços de desenvolvimento e manutenção de aplicações de software no âmbito da Subsecretaria de Tecnologia da Informação e Comunicação – trazendo o detalhamento das fases do ciclo de vida, descrição de atividades, papéis e responsabilidades envolvidos em cada etapa e os produtos gerados.





## DOCUMENTOS COMPLEMENTARES E INTERFACEADOS

Documentos complementares e interfaceados (MGP, políticas ITIL, boletins de serviços, arquitetura de referência, metodologia de administração de dados, procedimentos administrativos como fiscalização e pagamento e contratos de prestação de serviço).



## DIRETRIZES

As diretrizes a serem utilizadas em conjunto com a Metodologia Ágil de Desenvolvimento de Software serão as vigentes no momento da execução do projeto e/ou da manutenção, conforme estabelecido por portarias, resoluções, decretos, dentre outros.



## JUSTIFICATIVA

Visando padronizar e institucionalizar os processos de desenvolvimento de aplicações de software da Subsecretaria de Tecnologia da Informação e Comunicação, se fez necessária a definição de regras e padrões que orquestrem as ações estruturantes que viabilizem à STIC cumprir sua missão institucional de forma alinhada com as diretrizes da Administração Pública e com as melhores práticas de mercado no desenvolvimento e sustentação de sistemas.

A **MDS Ágil** pretende contemplar todas as atividades necessárias ao processo de desenvolvimento de software no Ministério da Educação, para aprimorar a qualidade desejável dos produtos entregues.

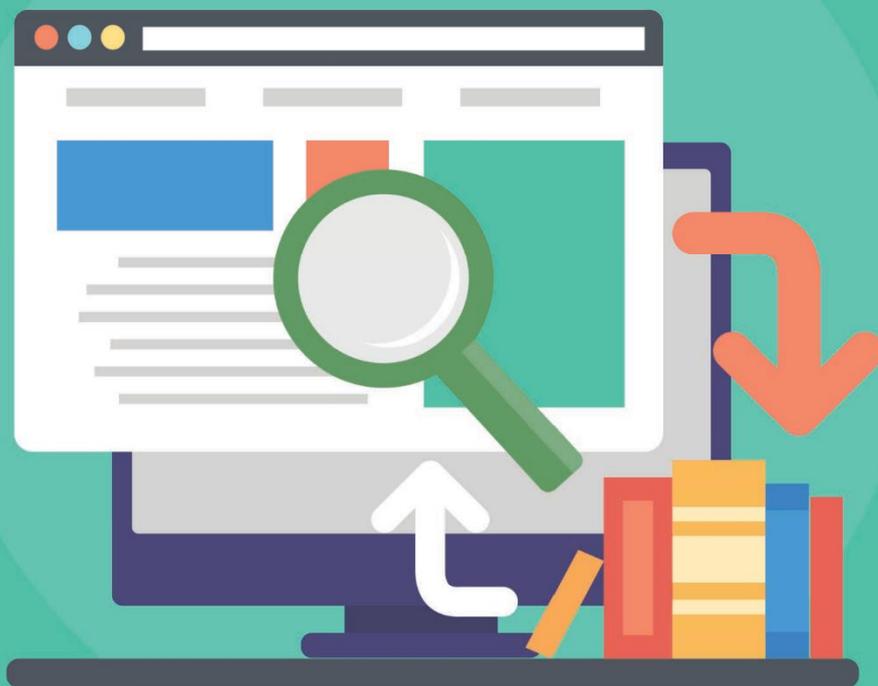
Os sistemas desenvolvidos e/ou mantidos deverão seguir os fluxos de atividades estabelecidos na metodologia, promovendo a padronização das soluções, eficiência, eficácia e efetividade do processo de desenvolvimento de sistemas.





## OBJETIVO

Padronizar os processos de trabalho, as funções desempenhadas e as tarefas a serem realizadas por todos os envolvidos no processo de desenvolvimento e manutenção de software da Subsecretaria de Tecnologia da Informação e Comunicação.



## POLÍTICA DE REVISÃO

Este documento poderá ser atualizado pela Subsecretaria de Tecnologia da Informação e Comunicação sempre que houverem novas diretrizes ou necessidade de alteração de diretrizes existentes. Também poderá sofrer modificações a fim de atender a normas, situações não previstas, recomendações de órgãos de controle, bem como adequar texto para eliminar eventuais ambiguidades, omissões ou contradições.



# CLASSIFICAÇÃO DAS DEMANDAS

Ministério da  
**Educação**

As demandas relacionadas a sistemas de informação podem receber a seguinte classificação:

- Desenvolvimento, manutenção adaptativa e evolutiva de soluções de software;
- Sustentação de soluções de software;

## **Desenvolvimento, manutenção adaptativa e evolutiva de soluções de software**

### **Desenvolvimento**

Compreende a criação de um novo software ou sistema de informação que pode ser desenvolvido integralmente a partir de uma nova necessidade do negócio, reconstruído a partir de um legado;

### **Manutenção evolutiva**

Compreende as demandas de incremento de funcionalidades a um sistema. Reflete mudanças em requisitos funcionais da aplicação, ou seja, inclusão, alteração ou exclusão de regras ou funcionalidades em aplicações já existentes.

### **Manutenção adaptativa**

Compreende as demandas de adaptação de um sistema a novas regras de negócio, melhorias de segurança e/ou a novas tecnologias, visando manter os sistemas alinhados aos processos de negócio e ao ambiente tecnológico seguro e disponível.

Demais manutenções serão classificadas como sustentação.

## **Sustentação de soluções de software**

É considerada sustentação de soluções de software as atividades técnicas de operação e manutenção continuada de soluções de software implantadas nos ambientes de produção do MEC, cujo principal resultado seja a correção de defeitos, manutenção da disponibilidade, estabilidade e desempenho de soluções. Além desses, estão incluídas no escopo, intervenções tempestivas ou pontuais de caráter perfectivo, corretivo, preventivo, atendimento ao usuário ou atividade operacional, **não se limitando a essas**, conforme segue:

### **Manutenção de disponibilidade, estabilidade e desempenho**

- Intervenção Corretiva;
- Análise e solução de incidentes;
- Monitoramento contínuo;
- Apoio à produção;
- Operação de soluções de software;
- Integração e entrega contínua.

### **Intervenções tempestivas ou pontuais**

- Intervenção perfectiva;
- Intervenção cosmética;
- Intervenção preventiva;
- Apuração especial;
- Migração de dados;
- Internalização de sistemas de informação;
- Monitoramento de programas críticos;
- Criação e atualização dos manuais e roteiros de orientação aos usuários;
- Elaboração de documentação técnica;
- Criação e/ou alteração de componentes arquiteturais;
- Suporte ao usuário;
- Emissão de parecer, nota ou relatório técnico;
- Análise e adequação de sistema;
- Apoio à implantação de sistema;
- Otimização e melhorias em *scripts* de banco de dados;
- Geração de massa de dados;
- Geração de casos de testes para realização de testes de carga.



## PROCESSO DE DESENVOLVIMENTO E MANUTENÇÃO

Para projetos de desenvolvimento e manutenção de sistemas, foram utilizadas como referência práticas ágeis baseadas no Scrum. Nesta metodologia estão presentes as seguintes características:

- Desenvolvimento iterativo e incremental;
- Disponibilização e manutenção de registros visíveis sobre o andamento das etapas dos atendimentos;
- Entrega contínua de funcionalidades;
- Revisões constantes do backlog do produto;
- Colaboração entre as equipes internas da STIC e áreas de negócio;
- Inovação no uso de modelos, padrões e ferramentas;
- Modelagem visual de software; e
- Arquitetura em camadas utilizando orientação a objetos, microserviços e integração contínua;
- Aplicação em projetos complexos em que não se conhece desde o início todas as necessidades ou etapas do processo de negócio.

## Release

Um **release (lançamento)**, em português) ocorre quando o produto atinge um nível de maturidade potencialmente "entregável" para o negócio, ou seja, reúne um conjunto mínimo utilizável para o negócio, resultante de uma ou mais sprints de trabalho.

Segue abaixo algumas diretrizes para o Release:

- Para os níveis de mínimos de serviço (NMSE), serão contabilizados os itens do backlog que estiverem concluídos na entrega da release. Os itens que retornarem para o backlog do produto ou forem cancelados não serão contabilizados;
- O tamanho da release em ponto de função (PF) será a soma da quantidade de PF dos itens concluídos;
- O prazo de entrega da release será combinado entre o time e o requisitante sendo a soma das sprints necessárias para que haja uma entrega de valor para o requisitante e que poderá ou não ser disponibilizada em produção, conforme a necessidade do negócio.

## Sprint

Um sprint é um período curto e fixo de tempo, em que uma equipe scrum trabalha, para entregar uma quantidade definida de itens de backlog do produto, combinadas com o requisitante, no início da sprint.

A quantidade de sprints é variável conforme o tamanho do projeto e é contabilizada sequencialmente ao longo do projeto. Segue abaixo a lista de diretrizes que devem ser seguidas em cada sprint:

- Todos os itens do topo do backlog devem estar com o estado de pronto "ready", ou seja, totalmente aptos e claros o suficiente para que o desenvolvedor possa implementá-lo no início da sprint;

- Primar pelo desenvolvimento de uma versão incremental utilizável do sistema sob o ponto de vista do usuário;  
Uma nova sprint terá início somente após a conclusão da Sprint anterior;  
Cada sprint deverá ter duração fixa (time-boxed) de até 15 dias corridos;  
Uma sprint pode ser cancelada antes do time-boxed da sprint terminar, mas o cancelamento de sprint deve ocorrer somente como exceção;
- Somente o product owner - PO tem a autoridade para cancelar a sprint, embora ele possa fazer isso sob influência das partes interessadas, do time de desenvolvimento ou do scrum master;
- Caso a sprint seja cancelada, os itens do backlog da sprint que não estiverem concluídos devem retornar backlog do produto, esses itens não serão considerados cancelados;
- Após o termino da sprint, os itens do backlog da sprint que não estiverem concluídos devem retornar backlog do produto, esses itens não serão considerados cancelados.

## **Reunião de Planejamento da *Sprint***

Por meio do trabalho colaborativo de todo o Time Scrum, é criado um plano com as atividades a serem desenvolvidas na sprint. Essa reunião deverá ter uma duração de até oito horas, para responder às seguintes questões: O que pode ser entregue como resultado do incremento da Sprint? Como o trabalho necessário para entregar o incremento será realizado?

## **Meta da *Sprint***

A meta da Sprint é definida a partir da implementação do backlog do produto, fornecendo uma direção para o time de desenvolvimento sobre as razões da construção do incremento, além de alguma flexibilidade a respeito da funcionalidade que será completada, mas sempre nos limites da Sprint.

## **Revisão da *Sprint***

A revisão da sprint 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 e obter comentários e promover a colaboração. Esta reunião deve ter uma duração (time-box) de até 4 horas. A reunião de revisão inclui os seguintes elementos:

- Os participantes incluem o time scrum e os convidados pelo product owner;
- O product owner esclarece quais itens do backlog do produto estão “prontos” e quais não estão “prontos”;
- O time de desenvolvimento discute o que foi bem durante a sprint, quais problemas ocorreram dentro da sprint, e como estes problemas foram resolvidos;
- O time de desenvolvimento demonstra o trabalho que está “pronto” e responde as questões sobre o incremento;
- Análise da próxima versão esperada do produto.

O resultado da reunião de revisão da sprint é um backlog do produto revisado.

## **Retrospectiva da *Sprint***

A retrospectiva da sprint 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 time-boxed de até três horas. O propósito da retrospectiva da sprint é:

- Inspecionar como a última sprint foi em relação às pessoas, aos relacionamentos, aos processos e às ferramentas;

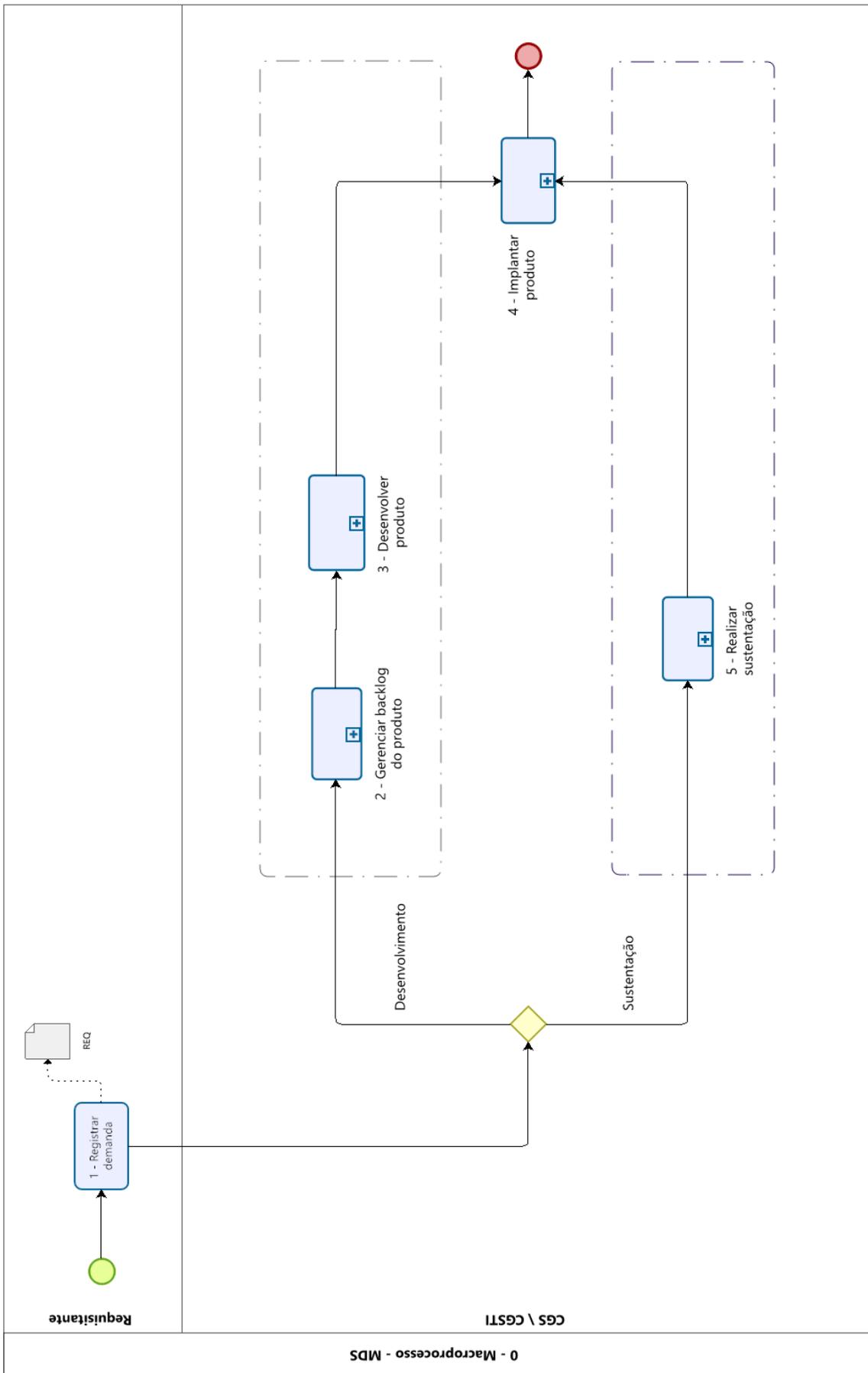
- Identificar e ordenar os principais itens que foram bem e as potenciais melhorias;
- Atualizar o artefato de lições aprendidas.

### **Manutenção**

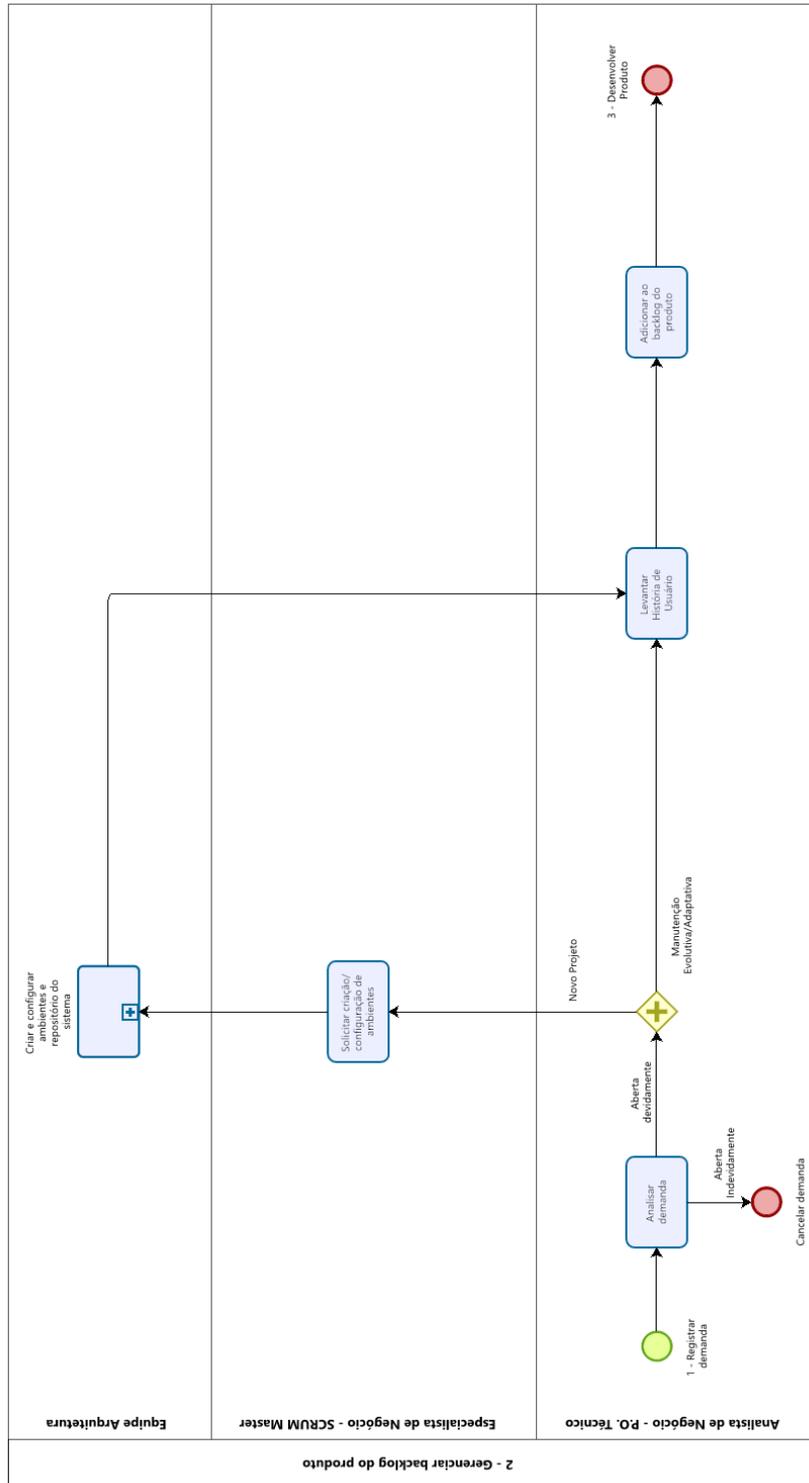
Para o processo de manutenção, foram utilizados como práticas ágeis baseadas no Scrum. Para o ciclo de vida das demandas de manutenção, foram adotadas as seguintes características:

- Desenvolvimento iterativo e incremental;
- Disponibilização e manutenção de registros visíveis sobre o andamento das etapas dos atendimentos;
- Entrega contínua de funcionalidades;
- Revisões constantes do backlog do produto;
- Colaboração entre as equipes internas da STIC e áreas de negócio;
- Inovação no uso de modelos, padrões e ferramentas;
- Modelagem visual de software; e
- Arquitetura em camadas utilizando orientação a objetos, microserviços e integração contínua;
- Aplicação em projetos complexos em que não se conhece desde o início todas as necessidades ou etapas do processo de negócio.

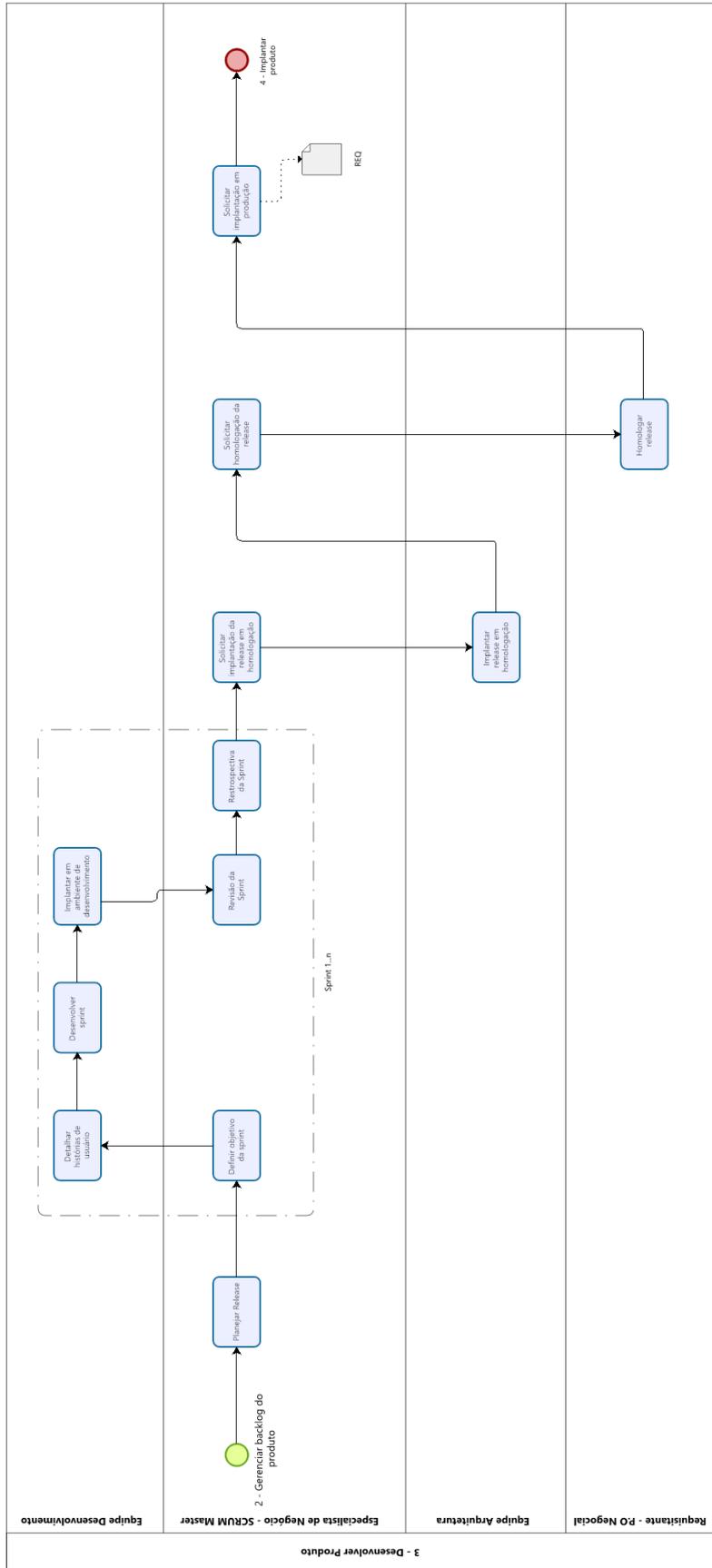
### Fluxo 1 - MacroProcesso MDS



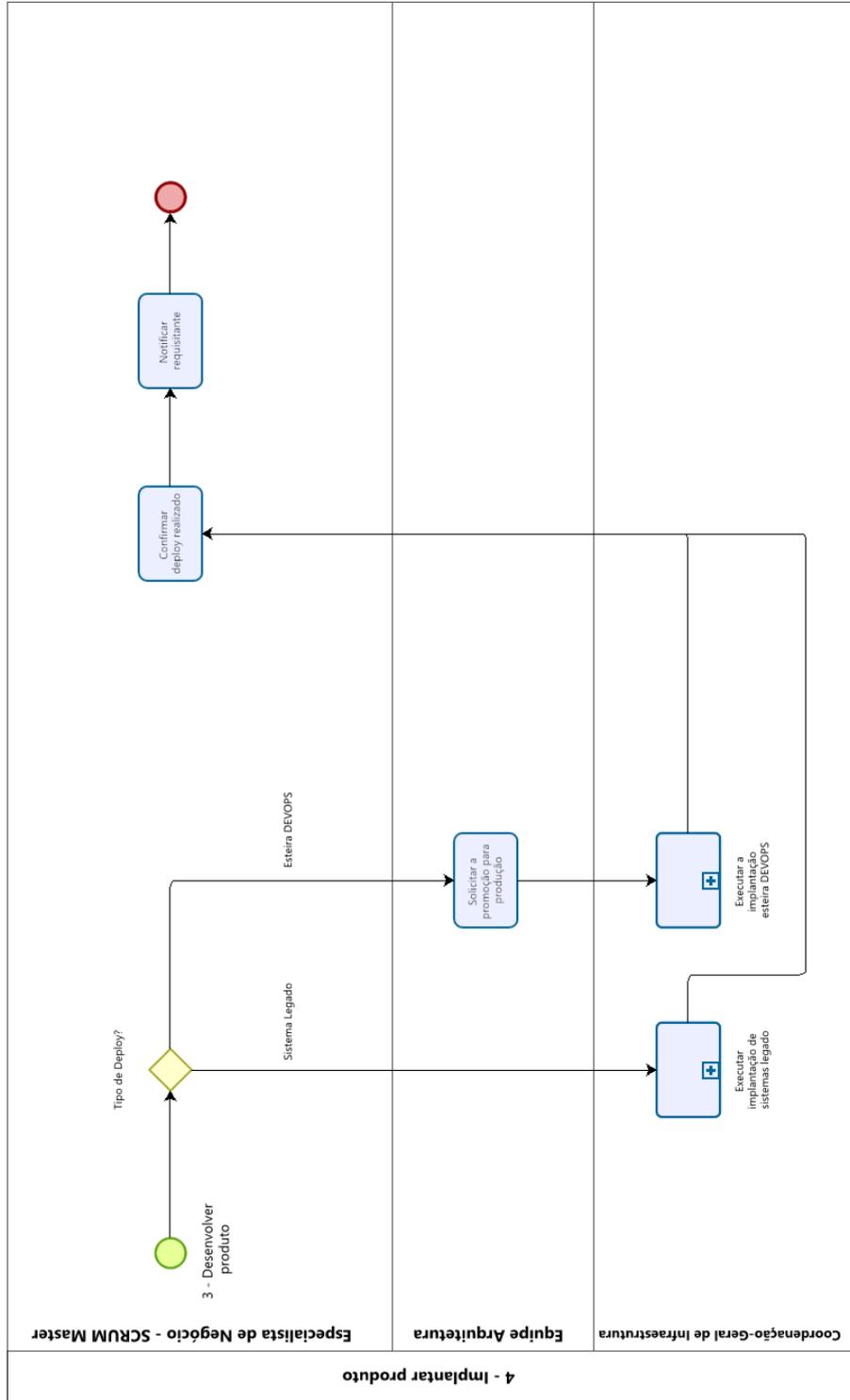
## Fluxo 2 – Gerenciar Backlog do Produto



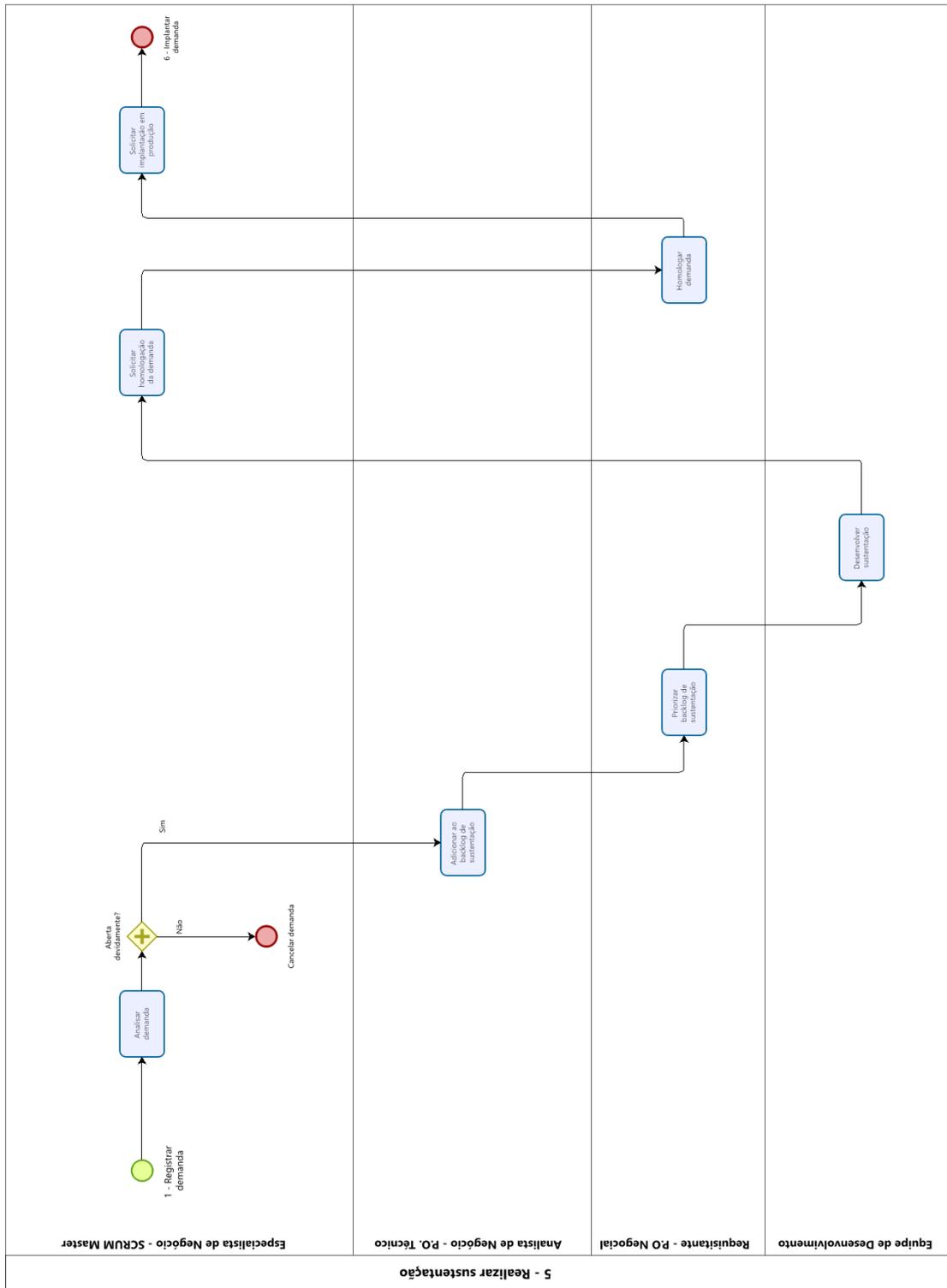
### Fluxo 3 – Desenvolver Produto



## Fluxo 4 – Implantar Produto



## Fluxo 5 – Realizar Sustentação



## Orientações para navegação web da metodologia da desenvolvimento de sistemas

A última versão web válida da metodologia de desenvolvimento de sistema estará sempre disponível no portal da STIC na Intranet do MEC. Estão também disponíveis no mesmo local, as versões navegáveis dos processos de trabalho. Para navegação siga os seguintes passos:

### Passo 1)

Escolher o processo a ser analisado, podendo:

#### Passo 1.1)

Para seleccionar um processo há duas formas: seleccionar na lista (retângulo verde) ou por meio da seleção do diagrama (retângulo vermelho), ambas têm a mesma funcionalidade.



Figura 1

Passo 1.2) Caso selecione o fluxo “0 – Macroprocesso MDS”, poderá também abrir os sub-processos contidos nele, vide figura 2.

Passo 2) Para selecionar um processo por meio do fluxo “0 - MacroProcesso MDS” há duas formas: ao clicar diretamente na atividade (retângulo vermelho) ou ao clicar no ícone (retângulo verde). Resultados:

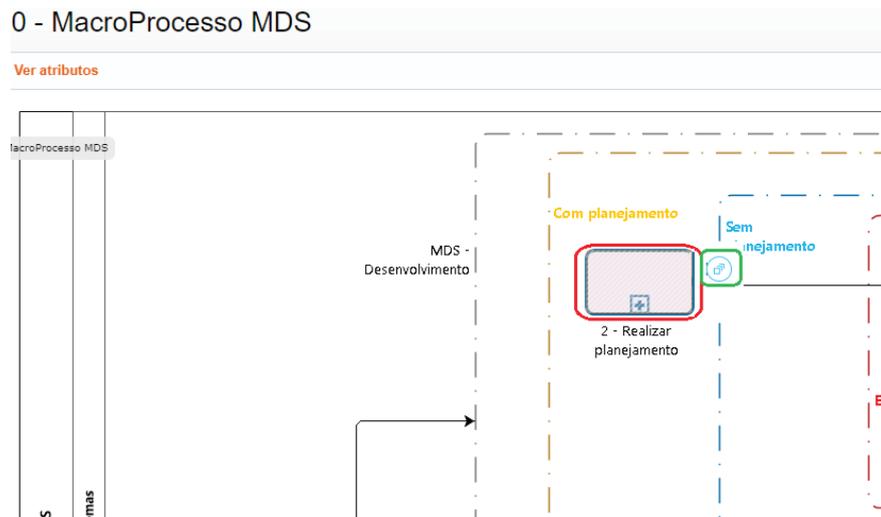


Figura 2

Passo 2.1) Caso opte por clicar no ícone (retângulo verde), o subprocesso será exibido. Caso opte por clicar na atividade (retângulo vermelho) então um novo campo será exibido com o nome do subprocesso “2 – Realizar planejamento”, para exibi-lo basta clicar no mesmo conforme mostrado na Figura 3, ambas têm a mesma funcionalidade.

3) Se clicar no ícone (quadrado vermelho) ou no nome do próprio subprocesso (retângulo verde) o referido subprocesso será exibido.



Figura 3

Passo 4) Ao abrir o processo, apenas suas atividades/tarefas estarão visíveis (retângulos vermelhos) Figura 4, isto é, os detalhes destas ficarão ocultos. Caso precise ter acesso aos detalhes proceder com o passo 5 deste tutorial.

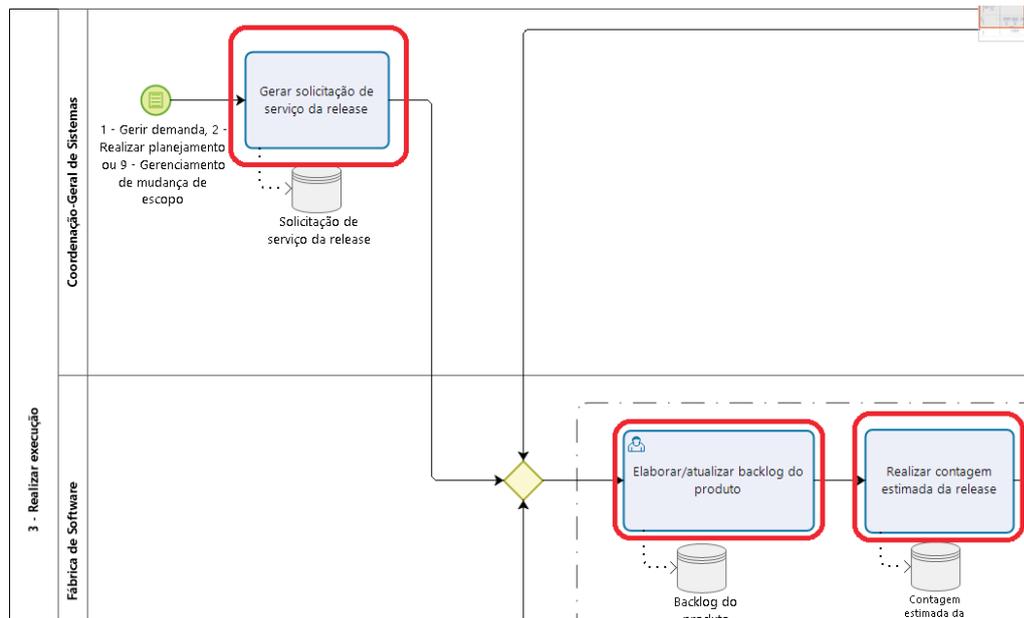


Figura 4

Passo 5) Se necessário ter conhecimento ao detalhamento das atividades do processo, basta clicar na atividade para exibição conforme apresentado na Figura 5.

3 - Realizar execução

**Gerar solicitação de serviço da release**

Descrição Objetivo Gerar a solicitação de serviço da release para elaboração, pela fábrica de software, da documentação de planejamento da release. Descrição A Coordenação-Geral de Sistemas registra a solicitação de serviço da release para elaboração da documentação de planejamento da release para planejamento do atendimento da demanda de desenvolvimento registrada pela área requisitante. A Coordenação-Geral de Sistemas atribui à fábrica de software a solicitação de serviço para atendimento dentro do prazo estabelecido em contrato. NOTA: no subprocesso "9 - Gerenciamento de Mudança de Escopo" caso a quantidade de ponto de função ultrapasse o limite máximo dos 30% previsto em contratos, o referido subprocesso voltará para este ponto, isto é, o fluxo da IMDS é reiniciado. Entrada(s) Demanda de

[Expandir](#)

Figura 5

Passo 5.1) Caso o detalhamento ultrapasse o tamanho do campo aberto (retângulo vermelho) basta clicar em "expandir" (retângulo verde) para exibição integral do detalhamento.



## PAPÉIS E RESPONSABILIDADES

Ministério da  
**Educação**

A Metodologia de Desenvolvimento de Sistemas da Coordenação-Geral de Sistemas (CGS/STIC/SE/MEC) contempla diversos papéis que executam, cada um, atividades e responsabilidades específicas dentro de cada processo dessa metodologia.

Para facilitar a compreensão da metodologia, será apresentado a seguir uma relação das responsabilidades atribuídas a cada um dos papéis envolvidos. Os papéis apresentados representam uma adaptação da metodologia SCRUM à realidade operacional do Ministério da Educação.

Abaixo, estão relacionados os papéis previstos na Metodologia de Desenvolvimento de Sistemas agrupados em 5 áreas de atuação:

- Área Requisitante (ou área de negócio): representa o demandante e/ou usuário do sistema;
- Coordenação-Geral de Sistemas: representa a área de gerenciamento, suporte e apoio ao desenvolvimento do sistema;
- Time de desenvolvimento: representa a equipe responsável por desenvolver o sistema;
- Equipe de Métrica: representa a equipe responsável pela aferição medição de sistemas em Pontos de Função (PF).
- Equipe de Arquitetura;
- Coordenação-Geral de Infraestrutura e Segurança da Informação: representa a área de gerenciamento, suporte e apoio à Infraestrutura de Tecnologia da Informação;

A tabela abaixo apresenta os papéis e responsabilidades necessários para a execução das atividades:

ATOR	PAPEL	RESPONSABILIDADES
Área Requisitante	Product Owner (PO)	<p>Segundo o <i>Scrum Guide</i>, Product Owner (PO), ou dono do produto, é o responsável por maximizar o valor do produto resultado do trabalho do Time de Desenvolvimento. Como isso é feito pode variar amplamente através das organizações, Times Scrum e indivíduos.</p> <p>O Product Owner é a única pessoa responsável por gerenciar o Backlog do Produto. O gerenciamento do Backlog do Produto inclui:</p> <ul style="list-style-type: none"> <li>• Expressar claramente os itens do Backlog do Produto;</li> <li>• Ordenar os itens do Backlog do Produto para alcançar melhor as metas e missões;</li> <li>• Otimizar o valor do trabalho que o Time de Desenvolvimento realiza;</li> <li>• Garantir que o Backlog do Produto seja visível, transparente, claro para todos, e mostrar o que o Time Scrum vai trabalhar a seguir; e,</li> <li>• Garantir que o Time de Desenvolvimento entenda os itens do Backlog do Produto no nível necessário.</li> </ul> <p>O Product Owner pode fazer o trabalho acima, ou delegar para o Time de Desenvolvimento fazê-lo. No entanto, o Product Owner continua sendo o responsável pelos trabalhos.</p> <p>O Product Owner é uma pessoa e não um comitê. O Product Owner pode representar o desejo de um comitê no Backlog do Produto, mas aqueles que quiserem uma alteração nas prioridades dos itens de Backlog devem endereçar ao Product Owner.</p> <p>Para que o Product Owner tenha sucesso, toda a organização deve respeitar as decisões dele(a). As decisões do Product Owner são visíveis no conteúdo e na priorização do Backlog do Produto. Ninguém pode forçar o Time de Desenvolvimento a trabalhar em um diferente conjunto de requerimentos.</p>

ATOR	PAPEL	RESPONSABILIDADES
Área Gestora de Desenvolvimento de Software (CGS/CGSTI)	Coordenador (a)- Geral (CGS ou CGSTI)	Segundo o Modelo Operacional, coordenar o desenvolvimento de soluções de tecnologia, a sustentação de sistemas, a evolução e os projetos de melhoria, em conformidade com as necessidades das áreas de negócio do Ministério da Educação, com aderência à estratégia de TIC do MEC, normativos internos e externos e padrões de conformidade, eficiência e economicidade, observando a evolução das Tecnologias de Informação e buscando aderência aos padrões tecnológicos do mercado nacional e internacional.
	Equipe de Arquitetura de Sistemas	Segundo o Modelo Operacional, realizar a gestão operacional de padrões de engenharia, arquitetura, segurança e configuração de sistemas/software e demais soluções de TIC, onde devem ser estabelecidos de modo consistente os seguintes processos: Gerenciamento de Padrões de Engenharia e Arquitetura, Gerenciamento de Segurança e Configuração, Operações de Teste e Medição de Qualidade de Softwares.
	Coordenador (a) de Sistemas	Segundo o Modelo Operacional, realizar a <b>gestão operacional</b> de sistemas e demais soluções de TIC, onde devem ser estabelecidos de modo consistente os seguintes processos: Gerenciamento de Manutenção de Sistemas (Sustentação e Evolução), Desenvolvimento de Sistemas, Integração de Dados e Tratamento de Dados Corporativos e Compartilhados, Monitoramento de Serviços e de Contratos para Prestação de Serviço, Gerenciamento de Sítios, Portais e Mobile.
	Fiscais técnicos de Contrato	Segundo o Modelo Operacional, realizar a execução de atividades privativas aos servidores da área de Desenvolvimento, onde devem ser estabelecidos de modo consistente os seguintes processos: Planejamento Técnico e Operacional; Execução da Supervisão de Sistemas e Serviços; Gestão de Desempenho de Sistemas, Aplicações, Sítios e Portais; Controle de Integração de Dados; Fiscalização Técnica de Contratos. Nota: Papel vinculado às Equipes da Coordenação de Desenvolvimento de Sistemas - CDS e Coordenação de Engenharia de Software - CES.
	Relacionamento com o Negócio	Segundo o Modelo Operacional, realizar a execução de atividades que podem ser terceirizadas desde que sob supervisão de servidores, onde devem ser estabelecidos de modo consistente os seguintes processos: Controle Operacional de Sistemas de Atendimento das Áreas de Negócio; Apoio ao controle de Níveis de Serviço (Fábricas e Infraestrutura); Controle de Desempenho Técnico de Sistemas, Aplicações, Sítios e Portais; Apoio ao Gerenciamento e à Execução de Projetos; Integração de Dados; Desenvolvimento de padrões de arquitetura de software; Desenvolvimento de padrões de configuração de software; Desenvolvimento de padrões de segurança de software; Controle de Conformidade de Mensurações das Aplicações de TIC; Controle de Protocolos de Testes de Sistemas e Aplicações; Aferição de Controles Quantitativos das Execuções de Contratos de Fábricas e outros Indicadores de Medição (REQs, WOs, HST, UST, PF, outros); Aferição e Análise de Indicadores de Projetos, Processos e Ações; Reporte de Métricas e Aferições às Coordenações da CGS e à STIC.

ATOR	PAPEL	RESPONSABILIDADES
Time de Desenvolvimento	Time de Desenvolvimento	<p>Segundo o Scrum Guide, Time de Desenvolvimento consiste de profissionais que realizam o trabalho de entregar um incremento potencialmente liberável do produto “Pronto” ao final de cada Sprint. Um incremento “Pronto” é requerido na Revisão da Sprint. Somente integrantes do Time de Desenvolvimento criam incrementos.</p> <p>Os Times de Desenvolvimento são estruturados e autorizados pela organização para organizar e gerenciar seu próprio trabalho. A sinergia resultante aperfeiçoa a eficiência e a eficácia do Time de Desenvolvimento como um todo.</p> <p>Os Times de Desenvolvimento tem as seguintes características:</p> <ul style="list-style-type: none"> <li>• Eles são auto-organizados. Ninguém (nem mesmo o Scrum Master) diz ao Time de Desenvolvimento como transformar o Backlog do Produto em incrementos de funcionalidades potencialmente liberável;</li> <li>• Times de Desenvolvimento são multifuncionais, possuindo todas as habilidades necessárias, enquanto equipe, para criar o incremento do Produto.</li> <li>• O Scrum não reconhece títulos para os integrantes do Time de Desenvolvimento, independentemente do trabalho que está sendo realizado pela pessoa;</li> <li>• O Scrum não reconhece sub-times no Time de Desenvolvimento, independente dos domínios de conhecimento que precisam ser abordados, tais como teste, arquitetura, operação ou análise de negócios; e,</li> <li>• Individualmente os integrantes do Time de Desenvolvimento podem ter habilidades especializadas e área de especialização, mas a responsabilidade pertence ao Time de Desenvolvimento como um todo;</li> </ul> <p>Tamanho do Time de Desenvolvimento</p> <p>O tamanho ideal do Time de Desenvolvimento é pequeno o suficiente para se manter ágil e grande o suficiente para completar um trabalho significativo dentro da Sprint. Menos de três integrantes no Time de Desenvolvimento diminuem a interação e resultam em um menor ganho de produtividade. Times de desenvolvimento menores podem encontrar restrições de habilidades durante a Sprint, gerando um Time de Desenvolvimento incapaz de entregar um incremento potencialmente liberável. Havendo mais de nove integrantes é exigida muita coordenação. Times de Desenvolvimento grandes geram muita complexidade para que um processo empírico seja útil. Os papéis de Product Owner e de Scrum Master não são incluídos nesta contagem, a menos que eles também executem o trabalho do Backlog da Sprint.</p> <p><b>FAZEM PARTE DO TIME DE DESENVOLVIMENTO:</b></p> <p>1) Cada EQUIPE DE DESENVOLVIMENTO deverá ser composta, no mínimo, pelos seguintes PERFIS PROFISSIONAIS:</p> <ul style="list-style-type: none"> <li>• Scrum Master;</li> <li>• Analista de Requisitos;</li> <li>• Engenheiro de Software.</li> </ul> <p>Para os demais perfis profissionais diretamente envolvidos na execução dos serviços ao CONTRATANTE:</p> <ul style="list-style-type: none"> <li>• Administrador de Dados;</li> <li>• Gerente de Configuração e Mudanças;</li> <li>• Analista de Segurança;</li> <li>• Analista de Testes e Qualidade.</li> </ul> <p>2) A execução dos serviços de SUSTENTAÇÃO de soluções de software deverá ser composta de acordo com os requisitos definidos em Contrato, para o respectivo serviço.</p>

ATOR	PAPEL	RESPONSABILIDADES
Equipe de Métricas	-	Equipe terceirizada coordenada por um servido, cuja competencia é realizar estimativas e medições, com base na Análise de Pontos de Função, para mensuração das demandas de desenvolvimento e manutenção de software executadas pela fábrica de software, ou ainda, a contagem funcional de sistemas legados do Ministério da Educação.
Equipe de Arquitetura	Arquitetos e Engenheiros de Software	Equipe terceirizada coordenada pelo Coordenador (a) de Arquitetura de Sistemas, cuja competencia é definir, manter, implementar e configurar em parceria com a equipe de infraestrutura, os padrões, o modelo de arquitetura, as rotinas e esteiras de deploy utilizando integração contínua usando o modelo DEVOPS bem como manter e migrar os modelos legados de sistemas sendo o apoio arquitetural do MEC para os times de desenvolvimento.

<p>Coordenador (a)- Geral de Infraestrutura e Segurança da Informação</p>		<p>Segundo a Portaria nº 787, de 14 de agosto de 2009 que aprovar o Regimento Interno da Diretoria de Tecnologia da Informação da Secretaria Executiva do Ministério da Educação: [...] Seção III Da Coordenação-Geral de Infraestrutura - CGI Art. 14 À Coordenação Geral de Infraestrutura compete: I - orientar, acompanhar e avaliar a execução das atividades de atendimento de suporte técnico, manutenção de equipamentos de informática e administração da rede corporativa do Ministério; II - coordenar, gerir e supervisionar os projetos de comunicação de voz e dados, rede elétrica estabilizada, rede local com e sem fio, infraestrutura computacional, serviços de atendimento de informática do Ministério; III - coordenar a execução da política de segurança de Tecnologia da Informação, no âmbito do Ministério; IV - coordenar, propor e manter políticas, diretrizes e critérios referentes a recursos de rede de computadores, banco de dados e suporte técnico; V - propor ações e sugerir prioridades nas atividades de rede de computadores, banco de dados e suporte técnico; VI - elaborar e rever, periodicamente, os documentos normativos necessários ao bom andamento das atividades da rede de computadores, banco de dados e suporte técnico, segundo padrões e orientações emanados do Governo Federal; e VII - avaliar e propor a aquisição de novas tecnologias, novos produtos e serviços que garantam funcionamento adequado e compatível com as necessidades de funcionamento da rede e banco de dados. [...]</p>
---	--	--

# ARTEFATOS



Abaixo são relacionados os artefatos obrigatórios e/ou opcionais a serem elaborados durante a execução do serviço de desenvolvimento e manutenção de soluções ou serviço de sustentação de soluções de software e entregue ao final da execução dos serviços em menção.

### Artefatos do serviço de desenvolvimento e manutenção de soluções de software

Artefato	Novo Projeto			Manutenção Adaptativa			Manutenção Evolutiva		
	Obrigatório	Opcional	Não se aplica	Obrigatório	Opcional	Não se aplica	Obrigatório	Opcional	Não se aplica
Backlog do produto	X	-	-	X	-	-	X	-	-
Contagem pontos de histórias	X	-	-	X	-	-	X	-	-
Documento de arquitetura de software – DAS	X	-	-	-	X	-	-	X	-
Requisitos de implantação	X	-	-	-	X	-	-	X	-
Histórias de usuário - HU	X	-	-	-	X	-	X	-	-
Critérios de aceitação (dentro da HU)	X	-	-	X	-	-	X	-	-
Código-fonte e/ou scripts	X	-	-	X	-	-	X	-	-
Modelo de dados com dicionário de dados	X	-	-	-	X	-	-	X	-
Regras de negócio e mensagens	X	-	-	-	X	-	-	X	-
Manual do sistema	X	-	-	-	X	-	-	X	-
Documento de entrega da release	X	-	-	X	-	-	X	-	-
Contagem detalhada da release	X	-	-	X	-	-	X	-	-

## Artefatos do Serviço de Sustentação - Atividades de Intervenções Tempestivas ou Pontuais

Tabela 5 - Serviço de sustentação de soluções de software - Atividades de intervenções tempestivas ou pontuais

Artefatos	Incidente			Serviço		
	Obrigatório	Opcional	Não se aplica	Obrigatório	Opcional	Não se aplica
Código fonte/script	x	-	-	x	-	-
Evidência de execução	x	-	-	x	-	-
Modelo de banco de dados	-	x	-	-	-	x
Nota técnica	-	x	-	-	x	-
Relatório/Extração de Dados	-	x	-	x	-	-

## Modelos dos artefatos

### Serviço de desenvolvimento e manutenção de soluções de software

Artefato	Forma de entrega
Backlog do produto	Ferramenta
Contagem pontos de história	Ferramenta
Documento de arquitetura de software – DAS	Template
Requisitos de implantação	Ferramenta
Histórias de usuário - HU	Ferramenta
Critérios de aceitação (dentro da HU)	Ferramenta
Código-fonte e/ou scripts	Ferramenta
Modelo de dados com dicionário de dados	Ferramenta
Regras de negócio e mensagens	Ferramenta
Manual do sistema	Site
Documento de entrega da release	Template
Contagem detalhada da release	Ferramenta

### Serviço de sustentação de soluções de software

Artefato	Forma de entrega
Código fonte/script	Ferramenta
Evidência de execução	Print de tela
Modelo de banco de dados	Ferramenta
Nota técnica	Template
Relatório/Extração de Dados	Arquivo

## Nomenclatura dos artefatos

### Serviço de desenvolvimento e manutenção de soluções de software

Artefato	Nomenclatura
Documento de arquitetura de software – DAS	MDS_Documento_Arquitetura - [Sigla do sistema]
Requisitos de implantação	Readme.md
Histórias de usuário - HU	HU[numero sequencial] – [nome da HU] Ex: HU001 – Manter usuário
Regras de negócio e mensagens	RN[numero sequencial] – [nome da RN] Ex: RN001 – Não permitir valor menor que zero  MSG[numero sequencial] – [nome da MSG] Ex: MSG001 – CPF inválido
Documento de entrega da release	MDS_Documento_Entrega_Release - [Sigla do sistema]



Metodologia de Desenvolvimento de Sistemas  
MdsCGS | MEC

# PADRÕES DE DESENVOLVIMENTO

Padrões de Arquitetura de Sistemas

Ministério da  
**Educação**

O Ministério da Educação dispõe de uma arquitetura padronizada para a construção de sistemas, que deve ser observada e seguida por todos os projetos de sistema. Seu detalhamento está publicado na página da STIC na intranet do MEC.

### **Padrões de Interface**

Para o frontend será utilizado o novo padrão de design do Governo Federal (Design System - <https://gov.br/design>). Seu detalhamento está apresentado em <https://gov.br/design>

### **Padrões de Administração de Dados**

O Ministério da Educação dispõe de padrões de Administração de Dados denominada MAD-Tec - Metodologia de Administração Técnica de Dados para a construção de sistemas, que deve ser observada e seguida por todos os projetos de sistema. Seu detalhamento está publicado na página da STIC na intranet do MEC.

### **Padrões de Gerenciamento de Configuração**

O Ministério da Educação dispõe de padrões de gerenciamento de configuração para a construção de sistemas, que deve ser observada e seguida por todos os projetos de sistema. Seu detalhamento está publicado na página da STIC na intranet do MEC.



# FERRAMENTAS DE APOIO AO PROCESSO DE DESENVOLVIMENTO

Ministério da  
**Educação**



## **Sistema de gestão de demandas**

As demandas de desenvolvimento e sustentação deverão ser registradas e acompanhadas pela área requisitante através das ferramentas disponíveis no portal de serviços de T.I. do Ministério da Educação (<http://servicosdeti.mec.gov.br>)

O acompanhamento e encaminhamento das demandas de desenvolvimento e sustentação registradas pela área requisitante será realizado pela equipe técnica através das ferramentas BMC (<http://atendimentodeti.mec.gov.br>) e SIMEC (<http://simec.mec.gov.br/login.php>).

## **Sistema de controle de versão**

O sistema GIT será utilizada como sistema de controle de versões de código para versionamento de novos projetos.

O sistema SVN será utilizada como sistema de controle de versões de código para versionamento de sistemas legados ainda não migrados para o GIT.

## **Sistema automatizado de testes e avaliação da qualidade do código**

O sistema automatizado de teste e avaliação da qualidade do código tem como finalidade a detecção de erros, vulnerabilidades e indícios de problemas mais profundo no código através da inspeção contínua de código no repositório do projeto.

Atualmente, em processo de análise para definição da ferramenta a ser utilizada.

## **Sistema automatizado de testes e avaliação da segurança do código**

O sistema automatizado de teste e avaliação da segurança do código tem como finalidade a verificação de possíveis falhas na segurança do código através da análise de vulnerabilidade de aplicação.

Atualmente, em processo de análise para definição da ferramenta a ser utilizada.





# REFERÊNCIAS BIBLIOGRÁFICAS

Ministério da  
**Educação**

BIZAGI PROCESS MODELER – Software utilizado no desenvolvimento dos fluxos. Disponível em: <<https://www.bizagi.com/>> Acesso em: 05/07/2019.

MINISTÉRIO DA EDUCAÇÃO (DF). Ministério da Educação. Pregão Eletrônico nº 11/2019. Contratação de serviços técnicos especializados de desenvolvimento, manutenção e sustentação de soluções de software – de acordo com as especificações, as métricas e os padrões de desempenho e de qualidade estabelecidos pelo Ministério da Educação no Termo de Referência e em seus anexos – limitado ao quantitativo máximo estimado, sem garantia de consumo mínimo e com pagamento em função de resultados. Brasília, ano 2019, 19 ago. 2019. Acesso em: 03/09/2019

MINISTÉRIO DA EDUCAÇÃO (DF). Ministério da Educação. MGP: Metodologia de Gerenciamento de Projetos. Disponível em: <<http://portaldti.mec.gov.br/2016-09-12-12-34-22/escritorio-de-programas-e-projetos-de-tic>> Acesso em: 05/07/2019.

SCHWABER, K. SUTHERLAND, J. Guia do ScrumMR - Um guia definitivo para o Scrum: As regras do Jogo. 2017. Disponível em: <<https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Portuguese-Brazilian.pdf>> Acesso em: 02/12/2019



Ministério da  
**Educação**