

Processo de Software - Dev7

Índice

- [Índice](#)
- [Páginas Filhas](#)
- [Histórico de Mudanças](#)
- [Apresentação](#)
- [Objetivos](#)
- [Definições e Abreviações](#)
- [Papéis e Responsabilidades](#)
- [Processos](#)
- [Normativos / Conformidade](#)
- [Revisão e Auditoria Periódicas do Processo](#)

Páginas Filhas

- [Processo de Ciclo de Vida de Software](#)
- [Processo de Desenvolvimento de Software](#)
- [Processo de Escopo e Requisitos de Software](#)
 - [Roteiro de Levantamento de Requisitos Funcionais](#)
 - [Roteiro de Levantamento de Requisitos Não Funcionais](#)
- [Processo de Arquitetura de Software](#)
 - [Arquitetura de Referência, Padrões e Boas Práticas](#)
 - [Plano de Gerência de Configuração](#)
- [Processo de Implantação de Software](#)
- [Processo de Sustentação de Software](#)
- [Processo de Descontinuidade de Software](#)

Histórico de Mudanças

Versão	Data	Descrição	Responsável
1.0	25/02/2016	Versão inicial	Wellington Luiz Gaboardi
1.1	31/10/2017	Alteração do Backlog Sprint (planejamento da sprint)	Wellington Luiz Gaboardi
2.0	13/09/2021	Revisão do Processo de Software e seus Sub Processos	Igor Bessa / Wiler Coelho / Jonathan Maia / José Mário
2.1	02/08/2022	Revisão do Processo de Software e seus Sub Processos	Thiarley Fontenele / Igor Bessa / José Mário

Apresentação

O Processo de *Software* do Tribunal Regional do Trabalho da 7ª Região (Dev7) tem como principal objetivo abranger todo o ciclo de vida dos *softwares* deste Regional. Ele envolve desde o recebimento da demanda de um novo produto, até o gerenciamento da arquitetura, o desenvolvimento, a sustentação e a sua desativação na Secretaria de Tecnologia da Informação e Comunicação (SETIC) do TRT7.

Este documento é parte de um conjunto de manuais que descrevem os processos de Gerenciamento de Escopo e Requisitos, Gerenciamento de Arquitetura, Processo de Desenvolvimento, Processo de Sustentação, Gerenciamento do Ciclo de Vida, Processo de Implantação de Software e Processo de Descontinuidade de Software. Tais processos usam como base as melhores práticas de mercado, adaptadas às experiências, necessidades e à cultura organizacional deste Regional.

Objetivos

- Definir procedimentos padronizados às atividades do ciclo de vida dos *softwares* deste Regional;
- Obter *softwares* de qualidade e que atendam às necessidades dos usuários de TIC;
- Alinhar-se à metodologia de gestão de portfólio e projetos (PROJ7) do TRT7, evitando a duplicidade de definições de processos de trabalho;
- Atender às recomendações dos órgãos de controle e legislações cabíveis, como a Lei Geral de Proteção de Dados Pessoais (LGPD) e a Resolução CNJ nº 370/2021;
- Observar o COBIT 2019, no que for pertinente e relevante à realidade do TRT7;
- Ser o mais flexível e enxuto possível para adotar as metodologias e práticas consagradas no mercado.

Aplicabilidade

Definir procedimentos padronizados às atividades relacionadas ao ciclo de vida do *software*. Implantações de sistemas não desenvolvidos pelo TRT7, manutenções corretivas (defeitos), evolutivas (aprimoramentos) ou adaptativas (adequação) também fazem parte do escopo desse processo.

Os procedimentos aqui definidos são de observação obrigatória por todos os servidores e estagiários responsáveis pelo desenvolvimento e sustentação de *softwares* no âmbito do TRT7.

Definições e Abreviações

- **Backlog:** Fila de demandas / tarefas / atividades em espera; Pode ser subdivido em backlog de produto referentes a todas as atividades em espera de um determinado produto ou backlog do ciclo, quando se trata das atividades em espera dentro de um mesmo Ciclo de trabalho, mas não necessariamente de um mesmo produto;
- **Cartão virtual:** Cartão virtual é a forma como as demandas ou entregáveis são formatados para possibilitar a visualização no Quadro Kanban;
- **Catálogo de Serviços de TIC:** Representa o conjunto de todos os serviços ativos e aprovados que são oferecidos aos usuários de TIC do TRT7;
- **Ciclo de trabalho:** Período de tempo com no máximo 4 semanas, onde os trabalhos são planejados (ou refinados/reabastecidos), executados, inspecionados e disponibilizados. São ciclos de feedback, que promovem a melhoria contínua dos fluxos de trabalho. No Scrum, temos um conceito similar, chamado de Sprint. Entretanto, o Ciclo de trabalho aqui definido não necessariamente implica em um período com escopo completamente planejado, e permite a utilização da abordagem de fluxo contínuo, como ocorre no Kanban;
- **Ciclo PDCA:** Um ciclo de quatro estágios (*Plan-Do-Check-Act*) para o gerenciamento e melhoria de processos. O ciclo Planejar-Executar-Verificar-Agir é também conhecido por "Ciclo de Deming";
- **COBIT:** Objetivos de Controle para Informação e Tecnologia relacionada (*Control Objectives for Information and related Technology*); provê recomendações e Melhores Práticas para o gerenciamento de Processos de TI. COBIT é publicado pelo IT Governance Institute;
- **Comitê de Gestão de TIC :** Órgão colegiado de natureza executiva e de caráter permanente, com representantes das unidades gestoras de TIC e responsabilidades de cunho gerencial, instituído por ato próprio;
- **Confluence:** Ferramenta de Gestão de conhecimento utilizada pelo TRT7;
- **CSJT:** Conselho Superior da Justiça do Trabalho
- **Demanda:** Uma demanda representa as necessidades que precisam ser avaliadas e, possivelmente, atendidas pelas equipes de *software*. Pode ser um incidente, uma requisição de serviço, um novo *software* ou uma alteração de um *software* já existente. O Demandante, conforme o caso, deverá informar necessidades, expectativas e requisitos, e aprovar as entregas. Uma demanda pode gerar um conjunto de entregáveis. Caso a demanda necessite de autorização prévia, neste processo ela já deve estar aprovada;
- **Demandante:** É um magistrado, servidor, gestor, unidade, comitê ou grupo de trabalho que encaminha demandas. Preferencialmente, fará parte do Time do projeto (caso a demanda faça parte do escopo de um) e terá sua necessidade atendida por meio do desenvolvimento de funcionalidades ou serviços de TIC. É a fonte primária dos requisitos, dores e necessidades motivadoras dos softwares;
- **DITIC:** Divisão de Infraestrutura de TIC, unidade do TRT7 responsável pelas atividades de infraestrutura (rede, servidores, banco de dados, etc) nos processos de *softwares* de TIC no Regional;
- **DSTIC:** Divisão de Sistemas de TIC, unidade do TRT7 responsável pela condução da sustentação, desenvolvimento, implementação ou implantação de *softwares* de TIC no Regional;
- **DSSUTIC:** Divisão de Serviços e Suporte aos Usuários de TIC, unidade do TRT7 responsável pela suporte de microinformática e sustentação de 1º nível nos processos de *softwares* de TIC no Regional;
- **eMAG:** Modelo de Acessibilidade em Governo Eletrônico do Governo Federal;
- **Épicos:** são grandes partes de trabalho que podem ser divididas em várias tarefas menores ou estórias de usuário;
- **Entregável:** Referência ao trabalho planejado que pode ser agendado, ter seu custo estimado, monitorado e controlado;
- **Estória de Usuário:** Representação do entregável, na forma de texto, que pode ser compreendida tanto pelo desenvolvedor quanto pelo demandante;
- **Facilitador:** É um dos membros do Time, e possui algumas atribuições similares ao do **Scrum Master (SM)** no Scrum ou do **Service Delivery Manager (SDM)** no Kanban, adicionando algumas atribuições de liderança técnica do Time. Ver responsabilidades na seção logo abaixo;
- **Gestor de Solução de TIC:** Servidores designados por ato da Presidência para responderem pela definição de processos de trabalho, requisitos, regras de negócio e níveis de serviços aplicáveis a uma solução de TIC;
- **ICP Brasil:** Infraestrutura de Chaves Públicas Brasileira;
- **Jira:** Ferramenta de gestão de demandas utilizada pelo TRT7;
- **Kanban:** É um método para gerenciar todos os tipos de serviços profissionais, também conhecidos como trabalho do conhecimento. Com o Método Kanban você visualiza o trabalho do conhecimento que está invisível e como ele se move através de um fluxo de trabalho;
- **Parte consultada (RACI) (Consulted):** Referese àquelas pessoas cujas opiniões são solicitadas em uma atividade (comunicação bidirecional). Em uma tabela RACI, responde à pergunta: Quem é responsável pelas entradas?
- **Parte informada (RACI) (Informed):** Referese às pessoas mantidas informadas e atualizadas sobre o andamento de uma atividade (comunicação unidirecional). Em uma tabela RACI, responde à pergunta: Quem recebe a informação?
- **Parte Aprovadora (RACI) (Accountable):** Pessoa, grupo ou entidade responsável por um assunto, processo ou escopo. Em uma tabela RACI, responde à pergunta: Quem responde pelo sucesso da tarefa?
- **Parte Responsável (RACI) (Responsible):** Referese à(s) pessoa(s) que deve(m) garantir que as atividades sejam concluídas com sucesso. Em uma tabela RACI, responde à pergunta: Quem está realizando a tarefa? As funções que tiverem o principal interesse operacional na realização da atividade relacionada e criarem o resultado esperado;
- **Priorizador:** É um dos membros do Time, preferencialmente um representante da área de negócio, entretanto pode ser assumido por um membro ou comitê da SETIC. Ele desempenha algumas atribuições similares às do **Product Owner (PO)** no SCRUM ou do **Service Request Manager (SRM)** no Kanban. Ver responsabilidades na seção logo abaixo;
- **Scrum:** Abordagem de gestão ágil para o desenvolvimento de produtos complexos. Nela, o trabalho é dividido em ciclos (sprints) e é executado iterativamente / incrementalmente;
- **SETIC:** Secretaria de Tecnologia da Informação e Comunicação;
- **Tabela RACI:** Ilustra quem é a pessoa responsável, aprovadora, consultada ou informada por cada tarefa dentro da estrutura organizacional;
- **TIC:** Tecnologia da Informação e Comunicação;
- **TRT7:** Tribunal Regional do Trabalho da 7ª Região.

Papéis e Responsabilidades

Os papéis do processo são os seguintes:

- O **Priorizador** possui as seguintes responsabilidades:
 - Gerenciar o *backlog* do produto (*software*);
 - Priorizar os entregáveis representando os interesses do demandante, oferecendo uma solução que realmente agregue valor ao negócio;
 - Transmitir ao Time as demandas preparadas (*ready*) para que comecem a planejar/desenvolver/executar;
- O **Facilitador** possui as seguintes responsabilidades:

- Facilitar a adoção das práticas estabelecidas;
- Remover impedimentos, atuando como um facilitador, permitindo que o Time fique focado na solução;
- Promover o bom andamento do fluxo de trabalho nos quadros Kanban, monitorando e atuando para eliminar os possíveis gargalos;
- Facilitar as reuniões, quando necessário;
- Promover a formação de Times auto organizados e multi funcionais;
- Promover a transparência e as boas práticas no Time;
- Atuar como um líder técnico junto ao Time;
- O **Time** possui as seguintes responsabilidades:
 - Representa, majoritariamente, a atuação e atribuições dos servidores da SETIC no ciclo de vida do *software* (de desenvolvimento, infraestrutura até suporte)
 - O **Desenvolvimento de TIC** tem as seguintes responsabilidades:
 - Refinar as estórias de usuário junto ao Priorizador e, quando necessário, junto ao Demandante;
 - Desenvolver os entregáveis disponíveis para serem codificados, deixando-os prontos para homologação;
 - Registrar todo o histórico da solução adotada;
 - Documentar as funcionalidades do *software* na ferramenta de gestão de conhecimento institucional;
 - Resolver falhas encontradas na homologação;
 - Atividades majoritariamente (mas não obrigatoriamente) executadas pela Divisão de Sistemas de TIC (DSTIC);
 - A **Infraestrutura de TIC** tem as seguintes responsabilidades:
 - Disponibilizar e manter os ambientes necessário para as atividades de desenvolvimento, implantação, sustentação e desativação de *software*;
 - Implantar o *software* nos ambientes requeridos pelo Demandante/Facilitador (desenvolvimento, bugfix, homologação, produção, etc);
 - Atividades majoritariamente (mas não obrigatoriamente) executadas pela Divisão de Infraestrutura de TIC (DITIC);
 - O **Suporte de TIC** tem as seguintes responsabilidades:
 - Manter o Catálogo de Serviços atualizado;
 - Prestar suporte aos *softwares* em produção;
 - Atividades majoritariamente (mas não obrigatoriamente) executadas pela Divisão de Serviços e Suporte aos Usuários de TIC (DSSUTIC);
- O **Demandante** possui as seguintes responsabilidades:
 - Papel definido e utilizado na gestão de projetos (PROJ7);
 - Definir as regras de negócio do *software* a ser desenvolvido;
 - Elaborar proposta de Projeto;
 - Homologar e receber entregáveis.
- O **Gestor de Solução de TIC** possui as seguintes responsabilidades:
 - Administrar e gerenciar, do ponto de vista negocial, *softwares* em produção;
 - Definir as regras de negócio do *software* em produção;
 - Esclarecer dúvidas de negócio do *software* em produção;
 - Tomar decisões sobre o ciclo de vida do *software* em produção.

O Dev7 não define de forma imutável qual dos papéis acima exercerá o papel de gerente de projeto previsto na metodologia de gestão de projetos (PROJ7) do TRT7, podendo variar de projeto para projeto.

Processos

Realize a leitura dos processos na seguinte ordem:

- [Processo de Ciclo de Vida de Software](#)
- [Processo de Desenvolvimento de Software](#)
- [Processo de Escopo e Requisitos de Software](#)
- [Processo de Arquitetura de Software](#)
- [Processo de Implantação de Software](#)
- [Processo de Sustentação de Software](#)
- [Processo de Descontinuidade de Software](#)

Normativos / Conformidade

Normativos, demandas de conformidade e índices que subsidiam a manutenção do Dev7.

- [Resolução CNJ nº 370/2021 - Estabelece a Estratégia Nacional de Tecnologia da Informação e Comunicação do Poder Judiciário \(ENTIC-JUD\);](#)
- [Portaria CNJ nº 211/2021 - iGovTIC-JUD - Dispõe sobre o Índice de Governança, Gestão e Infraestrutura de Tecnologia da Informação e Comunicação do Poder Judiciário \(iGovTIC-JUD\);](#)
- [IGG-TCU - Índice de Governança e Gestão do Tribunal de Contas da União;](#)
- [Lei nº 13.709/2018 - Lei Geral de Proteção de Dados Pessoais \(LGPD\);](#)
- [Resolução CNJ nº 324/2020 - Institui diretrizes e normas de Gestão de Memória e de Gestão Documental e dispõe sobre o Programa Nacional de Gestão Documental e Memória do Poder Judiciário – Proname;](#)
- [Resolução CNJ nº 408/2021 - Dispõe sobre o recebimento, o armazenamento e o acesso a documentos digitais relativos a autos de processos administrativos e judiciais;](#)
- [Resolução nº 91/2009 - MoReq-Jus - Institui o Modelo de Requisitos para Sistemas Informatizados de Gestão de Processos e Documentos do Poder Judiciário e disciplina a obrigatoriedade da sua utilização no desenvolvimento e manutenção de sistemas informatizados para as atividades judiciárias e administrativas no âmbito do Poder Judiciário;](#)
- [Modelo Nacional de Interoperabilidade do Poder Judiciário \(MNI\);](#)

Revisão e Auditoria Periódicas do Processo

Todos os processos são revisados anualmente e aperfeiçoados quando necessário.

O desempenho e a conformidade dos processos também serão avaliados anualmente pela SETIC e eventuais ajustes são realizados quando necessários. Um relatório da avaliação será submetido ao Comitê de Governança de TIC para validação.

Essas atividades foram incluídas na lista de [procedimentos necessários realizados pela DSTIC no início de cada ano](#).

Processo de Ciclo de Vida de Software

Índice

- [Índice](#)
 - [Objetivo](#)
 - [Definições e Abreviações](#)
 - [Papéis e Responsabilidades](#)
 - [Fluxo de Trabalho](#)
 - [Descritivo](#)
 - [Mapeado](#)
 - [Matriz de Responsabilidades](#)
 - [Indicadores de desempenho](#)
-

Objetivo

O objetivo deste macroprocesso é possibilitar uma melhor gestão do *software* como um todo, abrangendo todas as fases do seu ciclo de vida. Ele relaciona os processos de desenvolvimento, de implantação, de sustentação e de descontinuidade. Tem como finalidade também desenvolver os seguintes valores:

- Previsibilidade das entregas das *demandas* de softwares, que serão alcançadas por meio da utilização de indicadores e gráficos na ferramenta *Jira*;
 - Melhoria contínua do processo, realizada periodicamente com execução de ciclos *PDCA*s adaptando-se às necessidades do TRT7;
 - Visibilidade e transparência, que serão propiciadas com a utilização de Quadros *Kanban*, disponibilizados na ferramenta *Jira*. Todo trabalho em progresso deve estar visível, bem como gargalos e impactos gerados por falhas na execução das tarefas;
 - Melhoria dos índices de satisfação das outras unidades do TRT7, que poderão acompanhar o atendimento de suas demandas de *softwares*, além de priorizar o que é mais importante periodicamente;
 - Gestão compartilhada, que será propiciada por reuniões constantes com as equipes envolvidas nas atividades do *ciclo de vida* do *software* e participação efetiva das unidades demandantes na priorização de suas *estórias de usuário*;
 - Aumento da qualidade dos *entregáveis* de *softwares*, conforme o controle do trabalho em progresso, regulado de acordo com a capacidade disponível, diminuindo o tempo de espera por *demandas* prioritizadas e aumentando a qualidade dos serviços prestados, tornando a natureza do trabalho mais proativa e preventiva e menos reativa e emergencial.
-

Definições e Abreviações

Ver definições e abreviações na página do [Processos de Software](#).

Papéis e Responsabilidades

Os papéis do processo estão definidos na página do [Processos de Software](#).

Fluxo de Trabalho

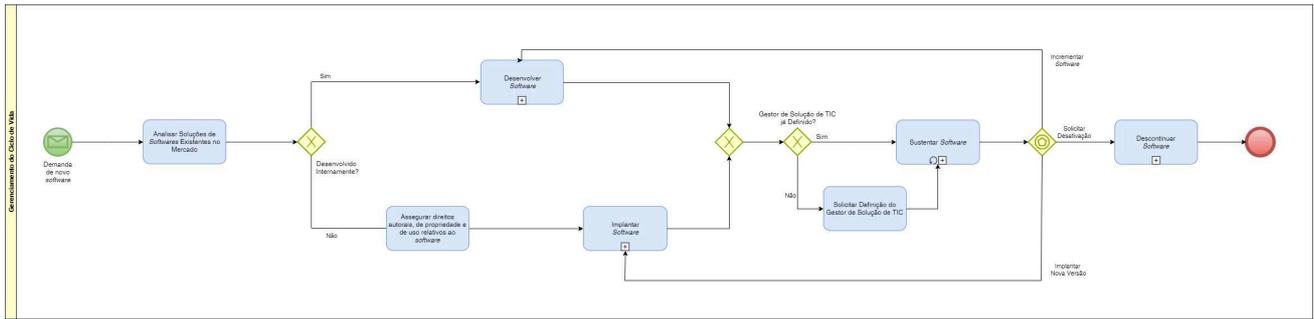
Descritivo

O fluxo de trabalho inicia com o recebimento da *demanda* de novo *software*. Logo após, serão analisadas as soluções de *softwares* existentes no mercado. Caso o novo *software* precise ser desenvolvido pelo TRT7, a *demanda* será direcionada para o [Processo de Desenvolvimento de Software](#). Caso contrário, serão adotadas todas as medidas necessárias para assegurar os direitos autorais, de propriedade e de uso relativos ao *software* e a *demanda* será encaminhada para o [Processo de Implantação de Software](#).

Quando o *software* for implantado em produção, será solicitado à Administração Superior a definição do Gestor de Solução de TIC para o *software* e o fluxo seguirá para o [Processo de Sustentação de Software](#), que é contínuo e permanece ativo até que haja uma solicitação de desativação do *software*. Neste caso, o fluxo caminhará para o [Processo de Descontinuidade de Software](#).

É possível também que novas versões do *software* já em produção sejam demandadas, estas serão direcionadas para o [Processo de Desenvolvimento de Software](#), caso o desenvolvimento seja realizado internamente, ou para o [Processo de Implantação de Software](#), para o caso de *softwares* desenvolvidos externamente.

Mapeado



Matriz de Responsabilidades

Atividades	Observações	Priorizador	Facilitador	Time	Demandante	Gestor de Solução de TIC
Analisar Soluções de Softwares Existentes no Mercado		A	I	R	C	
Desenvolver Software	Ver Processo de Desenvolvimento de Software	A/C*	R	R	C	A/C*
Assegurar direitos autorais, de propriedade e de uso relativos ao software		A	I	R	R	
Implantar Software	Ver Processo de Implantação de Software .	A/C*	R	R	I	A/C*
Solicitar Definição do Gestor de Solução de TIC		I	I	R	I	
Sustentar Software	Ver Processo de Sustentação de Software	A/C	R	R		A/C
Descontinuar Software	Ver Processo de Descontinuidade de Software	A/C	R	R		A/C

* Enquanto o software ainda não estiver implantado em produção, o aprovador das atividades de Desenvolvimento de Software e Implantação de Software será apenas o Priorizador, que representa os interesses do Demandante. Quando o software já estiver em produção, essas atividades passam a ter como responsável exclusivo o Gestor de Solução de TIC.

Legenda:

- R - *Responsible* (Responsável)
- A - *Accountable* (Aprovador)
- C - *Consulted* (Consultado)
- I - *Informed* (Informado)

Indicadores de desempenho

Processo de Escopo e Requisitos de Software

Índice

- [Índice](#)
- [Objetivo](#)
- [Definições e Abreviações](#)
- [Papéis e Responsabilidades](#)
- [Fluxo de Trabalho](#)
 - [Descritivo](#)
 - [Mapeado](#)
 - [Matriz de Responsabilidades](#)
- [Indicadores de desempenho](#)
 - [Índice de Entregáveis Aprovados](#)

- [Roteiro de Levantamento de Requisitos Funcionais](#)
- [Roteiro de Levantamento de Requisitos Não Funcionais](#)

Objetivo

O objetivo deste processo é possibilitar um melhor entendimento das necessidades do(s) Demandante(s), mapeando-as, preferencialmente, em *cartões virtuais* no formato de *épicos*, *estórias de usuário*, podendo ser também em formato de texto livre, alcançando os valores já definidos no [Processo de Ciclo de Vida de Software](#).

Definições e Abreviações

Ver definições e abreviações na página do [Processos de Software](#).

Papéis e Responsabilidades

Os papéis do processo estão definidos na página do [Processos de Software](#).

Fluxo de Trabalho

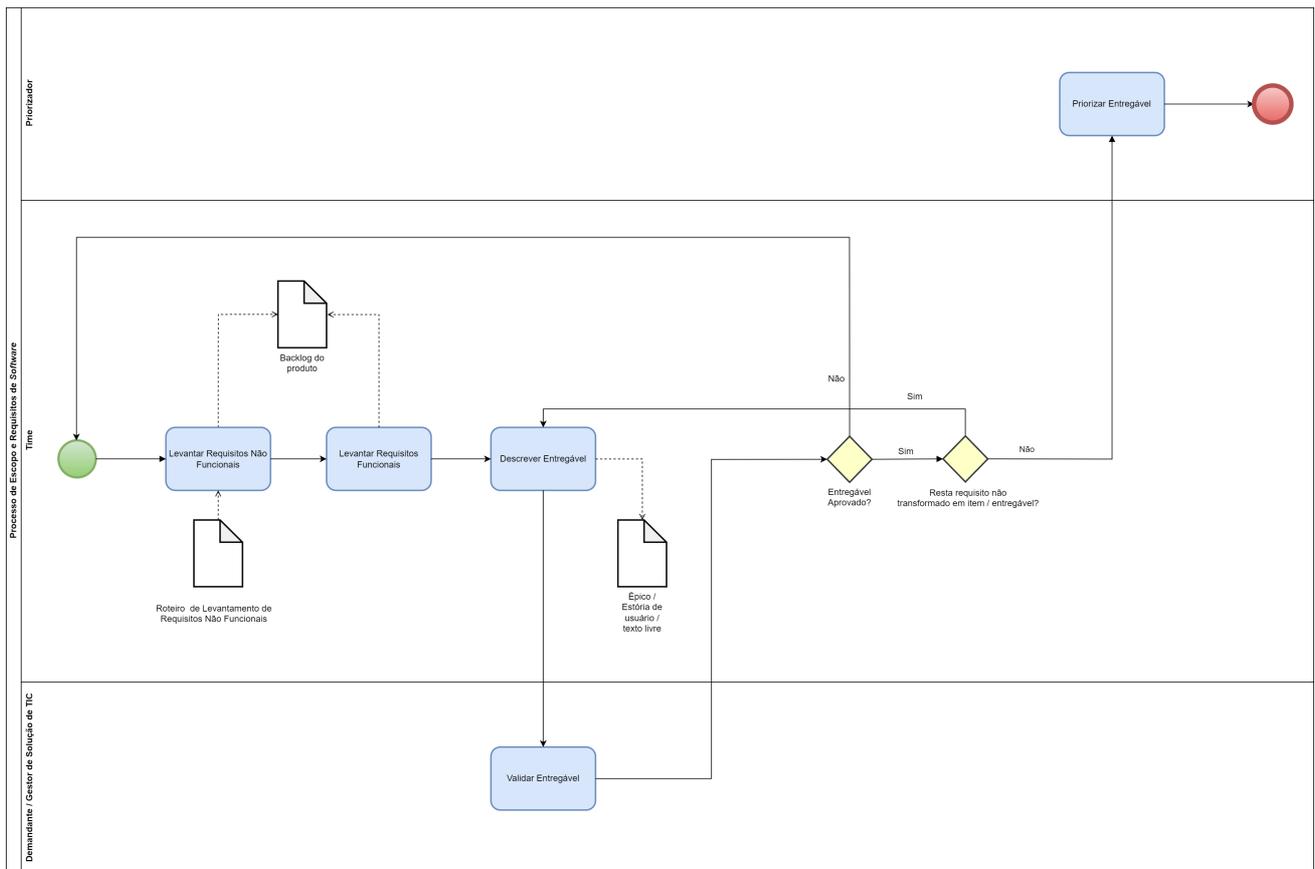
Descritivo

O fluxo de trabalho inicia com o levantamento junto ao Demandante do *software* (ou Gestor da Solução de TIC caso já esteja em produção) dos requisitos funcionais e não funcionais (ver [Roteiro de Levantamento de Requisitos Não Funcionais](#) e [Roteiro de Levantamento de Requisitos Funcionais](#)), com o objetivo de elencar os itens (*entregáveis*) que farão parte do *backlog* do produto.

Os itens levantados deverão ser registrados conforme o documento [Procedimentos para criação de cartões virtuais \(issues\) no JIRA](#), desde que correspondam às necessidades do Demandante (ou Gestor da Solução de TIC).

Por fim, estes itens devem ser validados pelo Demandante do *software* (ou Gestor de Solução de TIC), e depois priorizados pelo Priorizador com base na sua relevância (importância/urgência).

Mapeado



Matriz de Responsabilidades

Atividades	Observações	Priorizador	Facilitador	Time	Demandante / Gestor de Solução de TIC *
Levantar Requisitos Não Funcionais	Ver Roteiro de Levantamento de Requisitos Não Funcionais	C	I	R	A/C
Levantar Requisitos Funcionais	Ver Roteiro de Levantamento de Requisitos Funcionais	C	I	R	A/C
Descrever <i>Entregável</i> (épico /estória de usuário/texto livre)	Ver Procedimentos para criação de cartões virtuais (issues) no JIRA	C	I	R	A/C
Validar <i>Entregável</i> (épico /estória de usuário/texto livre)	Pode ser formalmente (e-mail, reunião, apresentação, etc), ou informalmente como uma simples conversa junto ao Demandante /Gestor de Solução de TIC.	C	I	I	A/R
Priorizar <i>Entregável</i>	O Priorizador juntamente com o demandante elege a ordem em que os requisitos levantados devem ser executados, diretamente no JIRA.	A/R	I	I	C

* Quando o *software* ainda não tiver implantado em produção, as atividades pertencerão ao Demandante. Já nas atualizações de versões, quem atuará será o Gestor de Solução de TIC.

Legenda:

- R - *Responsible* (Responsável)
- A - *Accountable* (Aprovador)
- C - *Consulted* (Consultado)
- I - *Informed* (Informado)

Indicadores de desempenho

Índice de *Entregáveis* Aprovados

- Descrição: Percentual de *entregáveis* que são aprovados pelo Priorizador
 - Periodicidade: Anual
 - Forma de cálculo: $(TEA / TES) \times 100$
 - TEA: Total de *Entregáveis* Aprovados
 - TES: Total de *Entregáveis* Escritos
 - Meta: 90%
-

Roteiro de Levantamento de Requisitos Funcionais

- [Requisitos funcionais](#)
- [Prioridades](#)
- [Diretrizes](#)
- [Normativos / Conformidade](#)

Objetivo	Trabalhar com o demandante para aprender sobre o domínio da aplicação e traduzir as regras de negócio em descrições compreensíveis aos atores envolvidos, de modo a se atingir pactuação das funcionalidades do software a ser desenvolvido, antes de iniciada sua implementação.
Entradas	Documento de Proposta de projeto
Atividades	<ul style="list-style-type: none">• Realizar entrevista com o demandante para identificar os requisitos funcionais;• Descrever e detalhar os requisitos;• Classificar os requisitos;• Elaborar Documento de Especificação de Requisitos;• Se necessário, elaborar os Diagramas de Casos de Uso, Protótipos de Tela ou outro artefato que facilite o entendimento do usuário;• Realizar verificações de validade, consistência, completeza, realismo e facilidade dos requisitos preliminares;• Solicitar aceite preliminar dos requisitos identificados.
Saídas	<ul style="list-style-type: none">• Backlog do produto no JIRA.

Requisitos funcionais

Requisitos funcionais são aspectos técnicos do software que retratam funcionalidades que o sistema proposto deve atender. A definição dos requisitos deve ser realizada por meio de descrição das funcionalidades que o sistema deve incorporar e prover ao usuário.

A etapa de levantamento de requisitos funcionais visa conciliar as necessidades dos diversos usuários do sistema, com ampla participação do demandante e do time de desenvolvimento, de modo a traduzir as regras de negócio do *software* em descrições compreensíveis à equipe técnica antes do início do desenvolvimento.

Para identificar as funcionalidades, o ideal é se orientar sempre pelas necessidades e dores do demandante, de modo a manter a concordância dos interesses do demandante com a funcionalidade. A regra básica é: para cada necessidade/dor, identificar as funcionalidades do sistema que a atendem.

A saída desta etapa é o documento de levantamento dos requisitos funcionais, buscando identificar as funcionalidades macro do sistema em três grupos de prioridades: obrigatórios, importantes e desejáveis. Com isso, é possível documentar as necessidades esperadas pelo demandante e subsidiar o desenvolvimento e a priorização da solução pela SETIC

Um ou mais requisitos será a entrada da etapa **Descrever Entregável**, a qual transformará esses requisitos em um Épico, Estória de Usuário ou outro texto livre. O item/entregável deve ser, em seguida, validado com o demandante.

Prioridades

A prioridade dos requisitos é informação balizadora no gerenciamento do escopo e na definição da priorização do entregável.

Para estabelecer a prioridade dos requisitos foram adotadas as denominações abaixo.

Obrigatório: o requisito é imprescindível. A sua ausência impede que o sistema entre em funcionamento.

Importante: a falta do requisito implica que o sistema pode entrar em funcionamento de forma não satisfatória, Tais requisitos devem ser implantados tão logo seja possível.

Desejável: o requisito não compromete as funcionalidades básicas do sistema. O sistema pode funcionar de forma satisfatória sem ele. Podem existir razões válidas em circunstâncias particulares para se ignorar um determinado item, mas a totalidade das implicações deve ser cuidadosamente examinada antes da escolha de uma proposta diferente

fonte: [MoReq-Jus](#).

Diretrizes

- Para o levantamento de requisitos funcionais, é essencial: entender o processo de trabalho do usuário e identificar as necessidades que devem ser atendidas com o sistema; levantar as soluções possíveis para atender às necessidades dos usuários; propor alternativas para desenvolvimento, apresentando as soluções já existentes no Tribunal; propor reuniões de brainstorming; criar protótipos iterativos etc;
- É necessário atenção para que não sejam identificadas funcionalidades duplicadas ou conflitantes. Também deve-se atentar para que não sejam classificadas como funcionalidades os [requisitos não funcionais](#);
- Os requisitos funcionais podem ter um nível de detalhamento mais genérico, a fim de facilitar a compreensão dos atores. No entanto, devem estar disponíveis detalhes suficientes para fornecer à equipe de desenvolvimento e ao priorizador as informações necessárias para criação das estórias e épicos e posteriormente a priorização dos entregáveis;
- É essencial que conste em cada requisito a sua prioridade, para fins de priorização e planejamento do backlog;
- O Demandante, conforme o caso, deverá informar necessidades, expectativas sobre a funcionalidade debatida e aprovar ou rejeitar as entregas, retroalimentando o processo de definição dos requisitos;
- Uma demanda pode gerar um conjunto de entregáveis. Caso a demanda necessite de autorização prévia, neste processo ela já deve estar aprovada;
- Se o software já está em produção, deve-se analisar a necessidade de adequações técnicas para permitir o fornecimento de tais informações;
- Os requisitos funcionais devem antecipar informações que a organização deseja utilizar e quais unidades poderiam provê-las;
- Os requisitos devem oferecer aos desenvolvedores do sistema uma compreensão melhor dos requisitos do sistema: definir as fronteiras do sistema, fornecer base para planejar o conteúdo técnico das iterações, fornecer base para estimar o custo e o tempo de desenvolvimento da demanda ou projeto, compreender a definição e o escopo do problema que se deseja resolver com o sistema ou funcionalidade;

Normativos / Conformidade

Ver [Normativos / Conformidade](#)

Roteiro de Levantamento de Requisitos Não Funcionais

- [Requisitos não funcionais](#)
- [Diretrizes](#)
- [Normativos / Conformidade](#)

Objetivo	Trabalhar com o demandante para aprender sobre o domínio da aplicação e traduzir os requisitos não funcionais em descrições compreensíveis aos atores envolvidos, de modo a se atingir pactuação das restrições ou atributos não relacionados às funcionalidades do software a ser desenvolvido, antes de iniciada sua implementação.
Entradas	Documento de Proposta de projeto
Atividades	<ul style="list-style-type: none">• Realizar entrevista com o demandante para identificar os requisitos não funcionais;• Descrever e detalhar os requisitos;• Elaborar Documento de Especificação de Requisitos;• Realizar verificações de validade, consistência, completude, realismo e facilidade dos requisitos preliminares;• Solicitar aceite preliminar dos requisitos identificados.
Saídas	<ul style="list-style-type: none">• Backlog do produto no JIRA.

Requisitos não funcionais

Requisitos não funcionais são aspectos técnicos do software que retratam restrições, qualidades ou atributos do sistema que não estejam diretamente relacionados às funcionalidades que ele deve prover ao usuário. A definição dos requisitos deve ser realizada por meio de descrição das características que o sistema deve atender e/ou prover ao usuário, em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenibilidade, tecnologias envolvidas etc.

A etapa de levantamento de requisitos não funcionais visa conciliar as necessidades dos diversos usuários do sistema com as restrições externas e características que o sistema deve atender (de natureza legal, de segurança da informação, de tempo de processamento, de arquitetura etc), com ampla participação do demandante e do time de desenvolvimento, de modo a traduzir tais requisitos em descrições compreensíveis à equipe técnica antes do início do desenvolvimento.

Para identificar os requisitos não funcionais, o ideal é se orientar em primeiro lugar pelas necessidades e dores do demandante, que podem implicar alguma restrição (por exemplo, orçamentária) ou exigir alguma qualidade específica do software (por exemplo, elevado poder de processamento ou rápido tempo de resposta). Em segundo lugar, deve-se observar as restrições legais que permeiam a atividade jurisdicional. Além disso, deve-se atentar para os requisitos de usabilidade, que implicam melhor interoperabilidade, responsividade e acessibilidade, facilitando a difusão do software para todos os públicos.

A saída desta etapa é o documento de levantamento dos requisitos, buscando identificar todos os requisitos não funcionais. Com isso, é possível documentar as necessidades esperadas pelo demandante e subsidiar o desenvolvimento e a priorização da solução pela SETIC

Um ou mais requisitos será a entrada da etapa **Descrever Entregável**, a qual transformará esses requisitos em um Épico, Estória de Usuário ou outro texto livre. O item/entregável deve ser, em seguida, validado com o demandante.

No mínimo, deve-se levantar os requisitos não-funcionais de:

- **Segurança da informação** - em especial **proteção de dados pessoais** (para o atendimento do Art. 46 § 2º da LGPD);
- **Portabilidade**;
- **Interoperabilidade** - incluindo atendimento ao MNI (Modelo Nacional de Interoperabilidade) do Poder Judiciário;
- **Responsividade** (adequação para dispositivos **móveis**);
- **Acessibilidade** - incluindo o atendimento ao **eMAG**;
- **Usabilidade**;
- **Assinatura** - Suporte para assinatura baseado em certificado emitido por Autoridade Certificadora credenciada na forma da Infraestrutura de Chaves Públicas Brasileira (ICP Brasil);
- **MoReq-Jus** - Atendimento ao Modelo de Requisitos para Sistemas Informatizados de Gestão de Processos e Documentos do Judiciário brasileiro;
- **RDC-Arq** - Atendimento à normatização pelo CNJ do Repositório Arquivístico Digital Confiável (RDC-Arq), quando houver acesso a documentos digitais relativos a autos de processos judiciais e administrativos.

Diretrizes

- Deve-se realizar a identificação precoce de requisitos de acessibilidade e de usabilidade, bem como a gestão permanente desses requisitos durante todo o ciclo de vida do *software* (Fonte: IGG TCU):
 - Os sistemas novos que forem desenvolvidos no TRT7 devem obrigatoriamente passar por avaliação do e-MAG , por meio da ferramenta ases.gov.br (Art. 33 da Resolução CNJ nº 370/2021), com um percentual mínimo de aprovação sugerido de 85%, sendo desejável atingir o percentual de 95%;
- Deve-se realizar a identificação precoce de requisitos de segurança da informação e a gestão permanente desses requisitos durante todo o ciclo de vida do *software* (Fonte: IGG TCU);
- Deve-se realizar a identificação precoce de requisitos de interoperabilidade e a gestão permanente desses requisitos durante todo o ciclo de vida do *software* (Fonte: IGG TCU):
 - Os sistemas de informação deverão atender a padrões de interoperabilidade e outros que venham a ser recomendados pelo Comitê Nacional de Gestão de Tecnologia da Informação e Comunicação do Poder Judiciário. (Art. 33 Resolução CNJ nº 370 /2021);
- Os requisitos contidos no Moreq-Jus do CNJ, que estabelece processos e requisitos mínimos para um Sistema Informatizado de Gestão de Processos e Documentos (GestãoDoc), foram considerados na elaboração desta lista, porém não há imposição normativa que o exija (Resolução CNJ nº 91/2009). Observa-se que, na prática, o Moreq-Jus está cada vez mais em desuso;
- O Repositório Arquivístico Digital Confiável (RDC-Arq) foi normatizado pelas Resoluções CNJ nº 324/2020 e 408/2021 para fins de preservação digital de documentos digitais relativos a autos de processos judiciais e administrativos. Mais detalhes no item "6.2.10. RDC-Arq: Repositório Arquivístico Digital Confiável" do Manual de Gestão de Memória do CNJ e o "11.5.4. Adoção de Repositório Arquivístico Digital Confiável – RDC-Arq" do Manual de Gestão Documental.
- Deve-se observar as recomendações do guia Padrões Web em Governo Eletrônico: Cartilha de Usabilidade ou as melhores práticas equivalentes

Normativos / Conformidade

Ver [Normativos / Conformidade](#)

Processo de Arquitetura de Software

Índice

- [Índice](#)
- [Páginas Filhas](#)
- [Histórico de Mudanças](#)
- [Objetivo](#)
- [Definições e Abreviações](#)
- [Papéis e Responsabilidades](#)
- [Fluxo de Trabalho](#)
 - [Descritivo](#)
 - [Mapeado](#)
 - [Matriz de Responsabilidades](#)
- [Indicadores de desempenho](#)
 - [Índice de Protótipos Validados](#)

Páginas Filhas

- [Arquitetura de Referência, Padrões e Boas Práticas](#)
- [Plano de Gerência de Configuração](#)

Histórico de Mudanças

Versão	Data	Descrição	Responsável
1.0	02/08/2022	Versão inicial (revisão do processo de software)	Igor Bessa / Wiler Coelho / José Mário / Thiarley / Rafael Ximenes

Objetivo

O objetivo deste processo é possibilitar uma melhor definição da arquitetura do *software*, seus componentes, suas propriedades externas, e seus relacionamentos com outros *softwares*, alcançando os seguintes valores:

- Validar as necessidades do Demandante, através de protótipo, quando necessário, para entregar valor ao negócio;
- Fornecer ao Time a estrutura básica para o desenvolvimento do *software*;
- Garantir a padronização dos *softwares*;
- Garantir o controle de versão dos *softwares*.

Definições e Abreviações

Ver definições e abreviações na página do [Processos de Software](#).

Papéis e Responsabilidades

Os papéis do processo estão definidos na página do [Processos de Software](#).

Fluxo de Trabalho

Descritivo

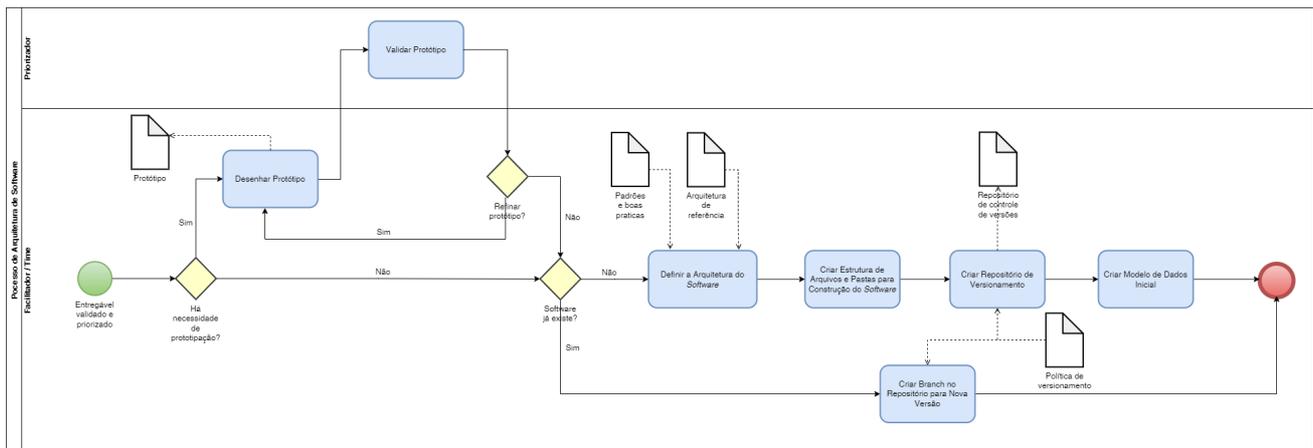
O fluxo de trabalho inicia-se a partir dos *entregáveis* já descritos e priorizados no *backlog* do produto no [Processo de Escopo e Requisitos de Software](#) (já tendo como entrada possíveis requisitos *funcionais* ou *não funcionais*).

Deve-se inicialmente verificar a necessidade de desenvolver um protótipo do *entregável* a ser desenvolvido. Caso positivo, deve ser construído, validado e, se necessário, refinado até ser aceito pelo Priorizador.

Após o aceite do protótipo ou caso não seja considerado necessário, deve-se avaliar se esse é o primeiro *entregável* a ser construído para o *software*. Caso positivo, sua arquitetura deve inicialmente ser definida tendo como base, preferencialmente a arquitetura de referência, e seguir os padrões e as boas práticas (ver [Arquitetura de Referência, Padrões e Boas Práticas](#)).

Em seguida, deve-se criar a estrutura de arquivos e pastas para a construção do *software*, bem como o repositório de versionamento, seguindo a política de versionamentos (ver [Plano de Gerência de Configuração](#)). Deve-se também criar o modelo de dados inicial a ser utilizado pelo *software*. Caso o *entregável* seja de um *software* já existente, deve-se criar uma nova *branch* no repositório de dados para a nova versão.

Mapeado



Matriz de Responsabilidades

Atividades	Observações	Priorizador	Facilitador	Time	Demandante / Gestor de Solução de TIC *
Desenhar Protótipo		A/C	R/A	R	C
Validar Protótipo		R	I	I	C/I
Definir a Arquitetura do Software	ver Arquitetura de Referência, Padrões e Boas Práticas	C/I	R/A	R	C/I
Criar Estrutura de Arquivos e Pasta para Construção do Software			R/A	R	
Criar Repositório de Versionamento	ver Plano de Gerência de Configuração		R/A	R	
Criar Modelo de Dados Inicial			R/A	R	
Criar <u>Branch</u> no Repositório para Nova Versão	ver Plano de Gerência de Configuração		R/A	R	

* Quando o *software* ainda não tiver implantado em produção, as atividades pertencerão ao Demandante. Já nas atualizações de versões, quem atuará será o Gestor de Solução de TIC.

Legenda:

- R - *Responsible* (Responsável)
- A - *Accountable* (Aprovador)
- C - *Consulted* (Consultado)
- I - *Informed* (Informado)

Indicadores de desempenho

Índice de Protótipos Validados

- Descrição: Percentual de protótipos validados pelo Priorizador
- Periodicidade: Anual
- Forma de cálculo: $(TPV / TPD) \times 100$
 - TPV: Total de Protótipos Validados
 - TPD: Total de Protótipos Desenhados
- Meta: 100%

Arquitetura de Referência, Padrões e Boas Práticas

Índice

- [Índice](#)
 - [Apresentação](#)
 - [Arquitetura de Referência](#)
 - [Padrões e Boas Práticas](#)
-

Apresentação

Este documento tem como proposta apresentar a arquitetura de referência, padrões, boas práticas e demais guias utilizados pela SETIC, contemplando as definições das tecnologias envolvidas para implementação de um nova versão ou de um novo *software*, atendendo aos requisitos funcionais e aos não funcionais, promovendo melhor disponibilidade, confiabilidade, escalabilidade e interoperabilidade dos *softwares* desenvolvidos.

Arquitetura de Referência

Considerando que o PJe é o *software* de referência de toda a Justiça do Trabalho (JT) e que já dispõe de uma equipe no CSJT dedicada à constante revisão dos seus padrões de arquitetura, a SETIC adotará na definição da arquitetura de novos softwares, preferencialmente, a mesma arquitetura do PJe (https://pje.csjt.jus.br/documentacao/index.php/Arquitetura_do_PJe_2). Deve-se considerar também o documento de arquitetura geral do CSJT: <https://pje.csjt.jus.br/documentacao/index.php/Arquitetura>. Desta forma, os *softwares* a serem desenvolvidos localmente estarão alinhadas com as tecnologias mais recentes empregadas pela JT.

Embora a arquitetura do PJe seja de uso recomendado, ela não é de uso mandatório, podendo a arquitetura definida variar de acordo com as necessidades de negócio (funcionais / não funcionais) ou técnicas. Também há que se pesar que, no caso de *softwares* já existentes e com arquiteturas já definidas, e diferentes da do PJe, a mudança arquitetural pode ser extremamente onerosa e sem ganhos práticos, devendo ser avaliado o custo / benefício.

É importante ressaltar que para a integração aos serviços de autenticação e autorização utilizados localmente no TRT7 (ex: CAS, KeyCloak, Google, etc), de acordo com as necessidades do *software* a ser desenvolvido, algumas adaptações pontuais podem ser necessárias à arquitetura padrão do PJe.

Padrões e Boas Práticas

Devemos observar, também, a vasta e completa documentação sobre padrões e boas práticas elaboradas pelo setor de arquitetura do CSJT.

- [Guia de Desenvolvimento Recomendado do PJe 2](#)
 - Dentre outras, trata sobre diretrizes básicas, backend (camada de API, serviço, etc.), dicas arquiteturais, batch jobs, frontend, angular (diretivas, componentes, etc.), typescript, acessibilidade, serviços REST
 - [Guia de Desenvolvimento Recomendado para o PJe 2.0](#)
 - Apresenta os temas Revisão Automática, PJe Arquitetura Base (PJe Segurança, PJe Auditoria, etc.), aplicando segurança às APIs, Exceções, Utilizando Parâmetros do *Software*, Implementando APIs/Serviços/Persistência, Mapeando Entidades JPA, Frontend - Visão Geral, Criação de Classes de Modelo (DTOs), Implementação de acesso ao end-point, Aplicando Segurança no Frontend, Boas práticas de acessibilidade, Banco de Dados, Gestão de configuração, etc.
 - [Processo para revisão de código consolidado](#)
 - Mais um documento bastante detalhado e valioso que versa sobre os temas procedimentos para criação de scripts SQL, orientação para desenvolvimento DML/HQL/EJBQL, boas práticas de testes unitários, de performance, de acessibilidade, de segurança, implementação de código, e padronização de nomenclatura de código fonte
 - [Arquitetura do PJe 2 - 10 Módulos - Boas práticas](#)
 - Apresenta algumas boas práticas, como Consumo de serviços Rest, Dependência de entidades de outros módulo, Acessar parâmetros do *software*, DTO e Token de segurança
 - [Padrao visual do PJe 2](#)
 - Aborda com detalhes assuntos sobre o padrão visual, como diretivas de usabilidade, layout, definições de cores, fontes e tipografia, navegação, paginação, elementos das páginas HTML, elementos Angular Material Design, etc.
 - [Tabela de ícones do PJe 2](#)
 - [Testlink - caso de teste](#)
 - Orienta os responsáveis pela especificação dos casos de teste (CT) para facilitar a execução dos testes e garantir a cobertura das funcionalidades do PJE. Está organizado em uma série de passos que devem ser seguidos para a especificação dos casos de teste no Testlink, melhores práticas e um exemplo (template).
 - [Verificação automática código](#)
 - Trata sobre ferramentas de regras automatizadas, como o gitlab runner, Utilitário Validação MR, Sonar, Lint, e o Jira
 - [Relatório Técnico - Comparativo entre o uso de Triggers e Observers para movimentações do JBPM](#)
-

Plano de Gerência de Configuração

Índice

- [Índice](#)
 - [Apresentação](#)
 - [Atribuições](#)
 - [Versionamento](#)
 - [Número das versões do software](#)
 - [Gerenciamento e Controle](#)
 - [Gerenciamento e acompanhamento](#)
 - [Controle dos artefatos](#)
-

Apresentação

O plano de gerência de configuração compreende as informações pertinentes à padronização de repositório para armazenamento de código fonte, assim como nomenclatura dos itens de configuração, informações relativas ao versionamento de itens e dos *software* desenvolvidos, descrição dos procedimentos a serem adotados no versionamento e *deploy* de *softwares* entregues pela SETIC.

O plano de gerência de configuração do *software* PJe definido pelo CSJT (https://pje.csjt.jus.br/documentacao/index.php/Plano_de_gerencia_de_configuracao) é a principal referência utilizada pela SETIC.

Atribuições

- De Desenvolvimento de TIC:
 - Gerenciar e controlar as versões dos *softwares* na ferramenta *Jira*;
 - Controlar os artefatos que são liberados através de baselines usando as ferramentas Git e GitLab;
 - Comunicar à Central de Serviços de TIC da liberação de cada nova versão;
 - Comunicar aos usuários, quando necessário, as alterações que serão liberadas e que terão impacto na forma de trabalho dos mesmos;
 - Disponibilizar para a infraestrutura o pacote de liberação de *entregáveis* devidamente documentado;
 - Comunicar à infraestrutura sobre a necessidade de uma nova liberação.
 - De Infraestrutura de TIC:
 - Realizar a implantação das versões dos *softwares* de acordo com a documentação;
 - Versionar a infraestrutura dos *softwares*, sempre que possível;
 - Automatizar a implantação das versões dos *softwares*, sempre que possível;
 - Considerando a cultura DevOps, é interessante que o Time possua as competências necessárias para executar as atribuições de desenvolvimento e infraestrutura, sempre que possível, fomentando o intercâmbio de conhecimentos entre servidores que trabalham com desenvolvimento ou com infraestrutura e automatizando as etapas "automatizáveis".
-

Versionamento

As regras estabelecidas no presente tópico tratam das versões do *software* a serem liberadas, quer a liberação seja para uso em homologação, quer a liberação seja para uso em produção. Também são definidas as regras para a nomeação das versões.

Número das versões do software

O *software* terá versões numeradas seguindo o seguinte padrão: X.Y.Z, onde:

X	Número principal da versão, somente alterado quando: <ul style="list-style-type: none">a) há modificação da arquitetura do <i>software</i>, ainda que não tenha havido modificação da estrutura de dados;b) há modificação da estrutura de dados que demande uma migração significativa de uma base para outra base de dados, não sendo suficiente para a mudança mera migração de dados entre tabelas de um mesmo banco de dados;c) há modificação substancial nas funcionalidades do <i>software</i>; Esse número deve ser 0 para versões piloto (ou MVPs - <i>Minimum Viable Products</i> - Produtos Mínimos Viáveis).
Y	Número intermediário de versão, modificado sempre que houver inclusão de um ou mais conjuntos de novas funcionalidades. Esse número deve iniciar em 0 e reiniciado quando da troca do número principal. São frequentemente denominadas versões MAJOR.

Z	Número menor de versão, modificado sempre que liberada versão de correção de erros ou de comportamento inesperado na versão do <i>software</i> . Esse número deve iniciar em 0 e reiniciado quando da troca do número intermediário ou do número principal. São frequentemente denominadas versões MINOR.
---	---

Gerenciamento e Controle

Gerenciamento e acompanhamento

As versões dos *softwares* serão gerenciadas e acompanhadas através da ferramenta *Jira*:

- Cada *software* terá um projeto correspondente e todas as suas versões serão registradas, gerenciadas e acompanhadas na própria ferramenta.

Controle dos artefatos

Usaremos as ferramentas Git e GitLab para controlar os artefatos de cada versão liberada:

- A cada versão liberada uma baseline (tag) será criada, desta forma, sempre teremos o estado original dos artefatos naquela versão;
- Via de regra, sempre existirão 2 linhas de desenvolvimento:
 - A linha principal, master, será usada para o desenvolvimento da próxima versão programada;
 - A segunda linha será sempre em uma branch, criada a partir da tag da última liberação em produção, ou seja, resolução de bugs e urgências serão realizadas nesta linha:
 - Após cada entrega em produção realizada a partir da branch de correção, deve ser criada uma nova tag e realizado o merge para o master, de forma a garantir que as alterações na branch sejam consolidadas no master.

Processo de Desenvolvimento de Software

Índice

- [Índice](#)
 - [Objetivo](#)
 - [Definições e Abreviações](#)
 - [Papéis e Responsabilidades](#)
 - [Fluxo de Trabalho](#)
 - [Descritivo](#)
 - [Mapeado](#)
 - [Matriz de Responsabilidades](#)
 - [Indicadores de desempenho](#)
 - [Índice de Atividades Resolvidas no Ciclo](#)
 - [Índice de Tempo Médio em Dias de uma Atividade desde sua Criação até sua Resolução](#)
 - [Índice de Tempo Médio em Dias de uma Atividade desde do Início da sua Execução até sua Resolução](#)
-

Objetivo

O objetivo deste processo é mapear as etapas necessárias para o desenvolvimento ou codificação de artefatos de *software*.

Definições e Abreviações

Ver definições e abreviações na página do [Processos de Software](#).

Papéis e Responsabilidades

Os papéis do processo estão definidos na página do [Processos de Software](#).

Fluxo de Trabalho

Descritivo

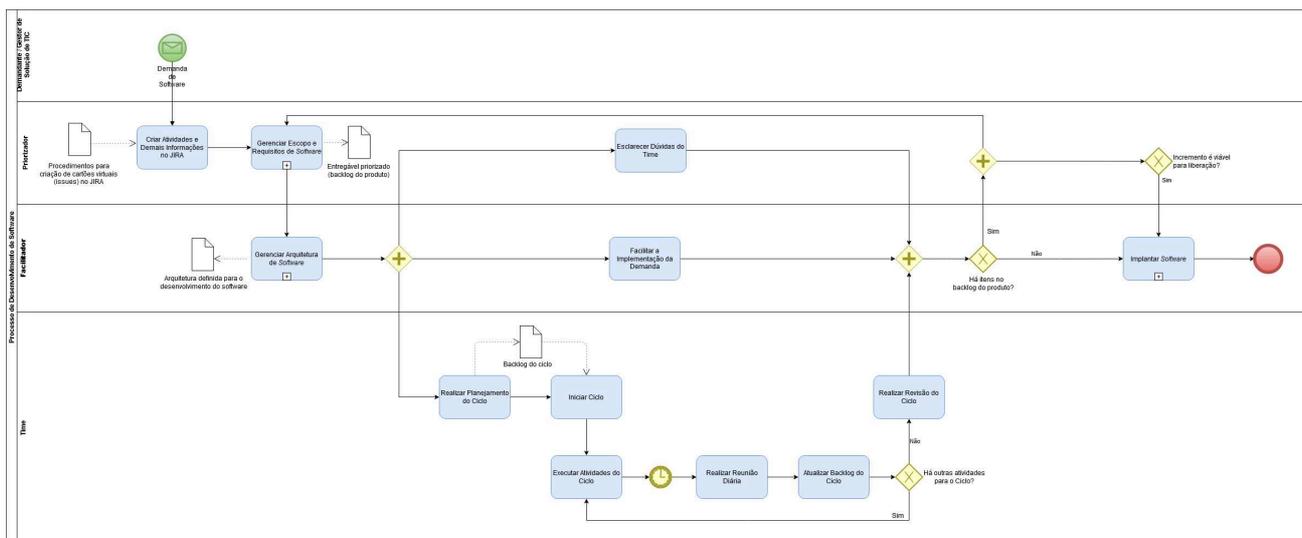
O fluxo de trabalho inicia-se com a demanda de um novo *software* ou incremento em um *software* já existente. O Priorizador deve criar uma *issue* no *JIRA* seguindo os [Procedimentos para criação de cartões virtuais \(issues\) no JIRA](#). Logo em seguida, o Priorizador deverá conduzir a execução do [Processo de Escopo e Requisitos de Software](#), que definirá os entregáveis que serão desenvolvidos ou requisitos a serem atendidos pelo novo *software* ou incremento.

A seguir, o Facilitador e o Time serão responsáveis por definir a arquitetura do *software*, realizando a execução do [Processo de Arquitetura de Software](#).

Uma vez superadas estas etapas, o Time responsável pela construção do *software* realizará o planejamento (que pode incluir refinamento dos itens ou reabastecimento do *backlog* do ciclo) para selecionar quais atividades do *backlog* do produto (*software*) serão priorizados para execução no próximo *ciclo de trabalho*, permitindo a inclusão de atividades durante a execução do ciclo. Em cada ciclo, um conjunto de etapas devem ser observadas, desde o planejamento do ciclo, passando pelo acompanhamento nas reuniões diárias e atualização do quadro *Kanban*, até a realização da revisão, onde os *entregáveis* são demonstrados ao Priorizador e/ou Demandante e posterior encerramento do ciclo. Durante a execução do ciclo, o Facilitador se encarregará de remover os impedimentos para a execução das atividades pelo Time e o Priorizador poderá ser consultado, caso surjam dúvidas negociais durante a implementação de uma atividade.

Ao final de cada ciclo, se o *entregável* do ciclo já for viável para utilização, executa-se o [Processo de Implantação de Software](#), e o fluxo de trabalho é reiniciado. Quando não existirem mais itens a serem executados no *backlog* do produto, ou quando o projeto for encerrado, o [Processo de Implantação de Software](#) pode executado e o desenvolvimento do *software* é finalizado.

Mapeado



Matriz de Responsabilidades

Atividades	Observações	Priorizador	Facilitador	Time	Demandante / Gestor de Solução de TIC*
Criar Atividades e Demais Informações no JIRA	Ver Procedimentos para criação de cartões virtuais (issues) no JIRA	R/A	I/R	I/R	I
Gerenciar Escopo e Requisitos de Software	Ver Processo de Escopo e Requisitos de Software	A/R	I	I/R	C
Gerenciar Arquitetura de Software	Ver Processo de Arquitetura de Software		R/A	R	
Esclarecer Dúvidas do Time		A/R	I	I	C
Facilitar a Implementação da Demanda			A/R	R	
Realizar Planejamento do Ciclo		A	R	R	C/I
Iniciar Ciclo		I	A	R	I
Executar Atividades do Ciclo		A	R	R	C/I
Realizar Reunião Diária		C	A	R	
Atualizar Backlog do Ciclo		A/C	I/R	R	
Realizar Revisão do Ciclo		A	R	R	C/I
Implantar Software	Ver Processo de Implantação de Software	A	R	R	C/I

* Quando o *software* ainda não tiver implantado em produção, as atividades pertencerão ao Demandante. Já nas atualizações de versões, quem atuará será o Gestor de Solução de TIC.

Legenda:

- R - *Responsible* (Responsável)
- A - *Accountable* (Aprovador)
- C - *Consulted* (Consultado)
- I - *Informed* (Informado)

Indicadores de desempenho

Índice de Atividades Resolvidas no Ciclo

- Descrição: Percentual de atividades resolvidas na ciclo

- Periodicidade: Quinzenal
- Forma de cálculo: $(TAR / TAS) \times 100$
 - TAR: Total de Atividades Resolvidas no ciclo
 - TAS: Total de Atividades Seleccionadas no ciclo
- Meta: 90%

Índice de Tempo Médio em Dias de uma Atividade desde sua Criação até sua Resolução

- Descrição: Tempo médio em dias desde a criação de uma atividade no backlog do produto até a sua resolução
- Periodicidade: Quinzenal
- Forma de cálculo: TCR / TAR
 - TCR: Tempo total em dias, desde a Criação de cada atividade até a sua Resolução.
 - TAR: Total de Atividades Resolvidas
- Meta: Resolver 90% do total de atividades desde sua criação em até 45 dias

Índice de Tempo Médio em Dias de uma Atividade desde do Início da sua Execução até sua Resolução

- Descrição: Tempo médio em dias desde do início da execução de uma atividade até a sua resolução
 - Periodicidade: Quinzenal
 - Forma de cálculo: TER / TAR
 - TER: Tempo total em dias, desde o início da Execução de cada atividade até a sua Resolução.
 - TAR: Total de Atividades Resolvidas
 - Meta: Resolver 90% do total de atividades iniciadas em até 10 dias
-

Processo de Implantação de Software

Índice

- [Índice](#)
 - [Objetivo](#)
 - [Definições e Abreviações](#)
 - [Papéis e Responsabilidades](#)
 - [Fluxo de Trabalho](#)
 - [Descritivo](#)
 - [Mapeado](#)
 - [Matriz de Responsabilidades](#)
 - [Indicadores de desempenho](#)
-

Objetivo

O objetivo deste processo é instalar um novo *software* ou atualizar um *software* já existente em um ambiente (ex: produção, homologação, testes, etc). Contempla *softwares* desenvolvidos internamente ou externamente.

Definições e Abreviações

Ver definições e abreviações na página do [Processos de Software](#).

Papéis e Responsabilidades

Os papéis do processo estão definidos na página do [Processos de Software](#).

Fluxo de Trabalho

Descritivo

O fluxo de trabalho inicia com o recebimento da demanda de implantação ou atualização de software. O Priorizador deve criar as issues iniciais da demanda no JIRA, conforme [Procedimentos para criação de cartões virtuais \(issues\) no JIRA](#) e priorizar a atividade junto ao Desenvolvimento de TIC.

O Desenvolvimento de TIC deve inicialmente realizar uma Análise Técnica dos Requisitos para Implantação do software e, se necessário, complementar as issues do JIRA com as informações técnicas necessárias para liberar o software em ambiente de produção. Em seguida, deve encaminhar a demanda para a Infraestrutura de TIC providenciar a montagem do ambiente de homologação.

Após montagem do ambiente de homologação, o Desenvolvimento de TIC ficará responsável pela adequação, configuração e validação técnica da versão do software já implantado neste ambiente mantendo atualizados os [Procedimentos para Validação Técnica de Sistemas](#). Em seguida, deve liberar o ambiente de homologação para o Demandante (quando o software não estiver em produção ainda) ou para o Gestor de Solução de TIC (quando o software já estiver em produção e a implantação for a atualização de uma nova versão) realizar a validação negocial da versão do software.

Concluída a validação negocial, o Desenvolvimento de TIC deve atualizar a documentação para prover a base de conhecimento, mantida preferencialmente na ferramenta Confluence, do Suporte de TIC, assim como tomar as medidas necessárias para prover o treinamento das equipes de TIC e dos usuários finais, quando necessário.

O Priorizador, auxiliado pelo Comitê Gestor de TIC, deve planejar juntamente com o Demandante ou Gestor de Solução de TIC a liberação da versão do software em ambiente de produção e então encaminhar a demanda para a Infraestrutura de TIC.

Antes da data de liberação do ambiente em produção a Infraestrutura de TIC, auxiliada pelo Comitê Gestor de TIC, pode providenciar a divulgação de notícia ou consultar as áreas afetadas sobre a viabilidade da indisponibilidade de sistema a fim de adequar a data;

Durante a liberação do ambiente em produção podem ocorrer as seguintes atividades, quando necessárias:

- O Desenvolvimento de TIC, auxiliado pelo Comitê Gestor de TIC quando necessário, pode providenciar juntos aos canais apropriados a publicação da certidão de indisponibilidade do sistema e divulgação de novidades da versão do Software implantada;
- O Priorizador pode encaminhar demanda ao Comitê Gestor de TIC para atualizar os dados do software no Catálogo de Serviços de TIC de acordo com a regulamentação do TRT7;
- O Priorizador pode encaminhar demanda ao Comitê Gestor de TIC para atualizar o Gestor da Solução de TIC de acordo com a regulamentação do TRT7;

Planejar Indisponibilidade do <i>Software</i>		C/I	C/I	R	C/I	C
Encaminhar atualização do Gestor do Sistema		R	I			C/I
Encaminhar atualização do Catálogo de Serviços		A	C/I	C/I	R	C/I
Encaminhar divulgações da Implantação de <i>Software</i>		C/I	R	C/I	C/I	A
Liberar para Produção		A	C/I	R	C/I	C/I

* Quando o *software* ainda não tiver implantado em produção, as atividades pertencerão ao Demandante. Já nas atualizações de versões, quem atuará será o Gestor de Solução de TIC.

Legenda:

- R - *Responsible* (Responsável)
- A - *Accountable* (Aprovador)
- C - *Consulted* (Consultado)
- I - *Informed* (Informado)

Indicadores de desempenho

Processo de Sustentação de Software

Índice

- [Índice](#)
 - [Objetivo](#)
 - [Definições e Abreviações](#)
 - [Papéis e Responsabilidades](#)
 - [Fluxo de Trabalho](#)
 - [Descritivo](#)
 - [Mapeado](#)
 - [Matriz de Responsabilidades](#)
 - [Indicadores de desempenho](#)
 - [Índice de Resolução de Chamados Dentro do Prazo](#)
 - [Índice de Resolução de Requisições de Usuários Não VIPs Dentro do Prazo](#)
-

Objetivo

O objetivo deste processo é assegurar a correção, adaptação e evolução de *softwares* mantidos pelo TRT7 no Catálogo de Serviços de TIC.

Definições e Abreviações

Ver definições e abreviações na página do [Processos de Software](#).

Papéis e Responsabilidades

Os papéis do processo estão definidos na página do [Processos de Software](#).

Fluxo de Trabalho

Descritivo

Para dar suporte a este processo existem diversas [orientações e roteiros](#) que mantidos pelo Times de Desenvolvimento, Infraestrutura e Suporte de TIC responsáveis pelas demandas de sustentação de *softwares*.

O Time de Suporte de TIC deve realizar o atendimento inicial da demanda de sistemas de forma a garantir o registro eletrônico no sistema *Assyst* :

- Ainda que a solicitação seja recebida por outros meios de comunicação disponibilizados aos usuários, p.ex., suporte telefônico, e-mail ou mensageiros eletrônicos corporativos utilizados pelas unidades da SETIC, deve-se garantir o registro da demanda no sistema *Assyst*, com o apoio da Central de Serviços de TIC.
- Somente em caso de incidentes considerados urgente é autorizado realizar o atendimento da demanda sem necessidade prévia de registro formal. Nestas situações, o próprio Time que solucionou o problema deve providenciar, *a posteriori*, a abertura do chamado, informando o conteúdo da solicitação e a solução aplicada, de forma que as estatísticas de resolução de chamados das unidades da SETIC não sejam prejudicadas.

A seguir o Time de Suporte de TIC deve realizar análise inicial observando especialmente os pontos abaixo :

- verificar se a demanda já foi resolvida pelo atendimento telefônico ou outro meio disponível e, em caso de solução já aplicada, o chamado deve ser encerrado.
- verificar se as informações constantes no chamado são suficientes para a resolução do chamado. Caso seja necessário, o demandante deve ser contactado pela Central de Serviços de TIC para complementar essas informações.
- verificar se já existe conhecimento sobre a demanda cadastrado para o 1º nível de atendimento. Caso positivo, o chamado deve ser resolvido pela Central de Serviços de TIC. Nos demais casos, o chamado deve ser encaminhado para o Time especializado para resolução da demanda.

Após encaminhamento da Central de Serviços de TIC, o chamado deve ser priorizado pelo Priorizador, ou pelo próprio Time, e aguardará na fila de atendimento priorizado. Seguindo as regras previamente definidas no [Roteiro de Atendimento de Chamados no Assyst](#), o Time realizará a análise especialidade do chamado observando as seguintes condições :

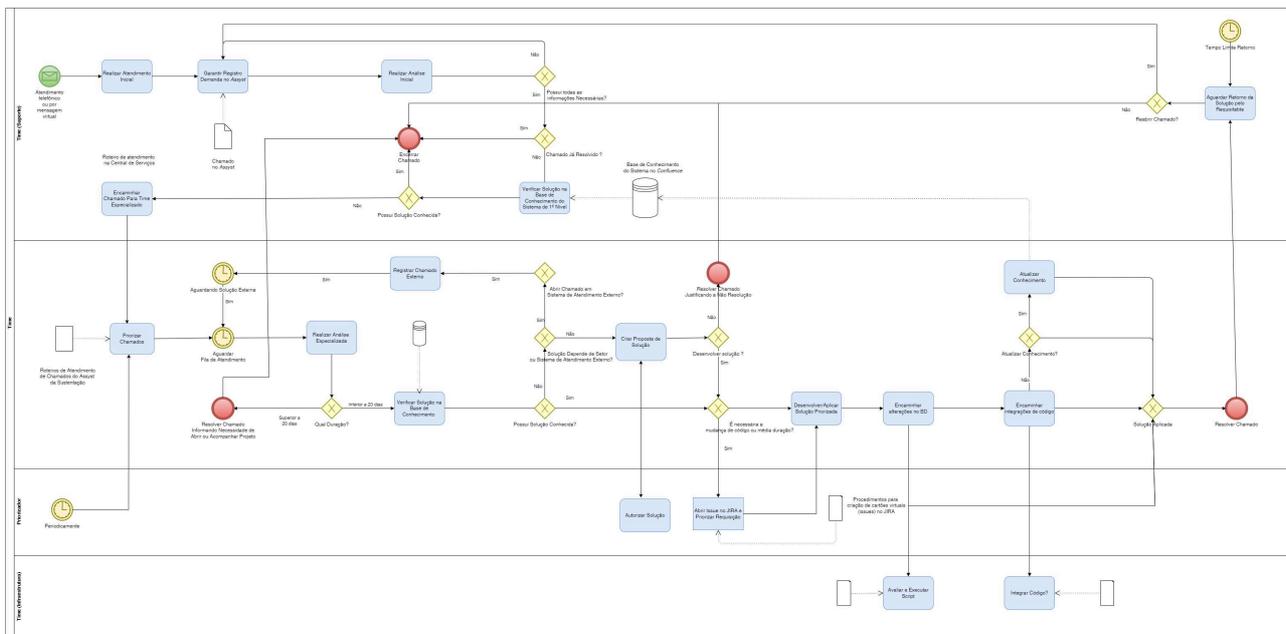
- Em se tratando de uma demanda cuja solução demore mais de 20 dias, o chamado deve ser resolvido informando que deve ser criada uma proposta de projeto ou acompanhar uma já existente, conforme [Metodologia de Projetos do TR7 \(PROJ7\)](#).
- Se não existir solução para a demanda existente na Base de Conhecimentos, deve-se verificar, primeiramente, se há dependência externo para sua solução definitiva.
 - Caso positivo, deve-se registrar a demanda no sistemas de chamados ou meio equivalente disponibilizado pelo setor/organização externa , pausar o chamado no Sistema Assyst e aguardar o retorno da demanda externa pra retomar a análise especializada do chamado, mesmo que tenha sido aplicada solução de contorno temporária;
 - Caso negativo, deve analisar a viabilidade do desenvolvimento da solução local, com auxílio do Priorizador, para prosseguir com sua execução ou encerrar o chamado justificando o não atendimento ao demandante;
- Se já existir solução existente na Base de Conhecimentos, prosseguir com a aplicação da solução;

Durante os procedimentos de desenvolvimento ou aplicação da solução, o Time deve observar as seguintes recomendações :

- Se for necessário alterações de código no desenvolvimento da solução ou tratar-se de requisição de média/alta complexidade, é necessário abrir um ou mais cartões virtuais no sistema *Jira* , com auxílio do Priorizador, a fim de registrar a demanda corretamente, de acordo com o [P rocedimentos para criação de cartões virtuais \(issues\) no JIRA](#);
- Se for necessária a execução de scripts em banco de dados, uma demanda será encaminhada para a Infraestrutura de TIC para validação e execução.;
- Se for necessária a execução de integração de código, uma demanda pode ser encaminhada para a Infraestrutura de TIC para validação e execução.

Por fim, se a requisição ou incidente tiver gerado conhecimento para qualquer unidade da SETIC, ele deverá ser registrado na ferramenta de gestão de conhecimento (*Confluence*), e logo depois deve ser resolvido e informada a solução para o requisitante. O chamado aguardará retorno do usuário para validação da demanda resolvida ou aguardar um tempo pré-definido para ser encerrado pela *Central de Serviços de TIC*. Em caso de não validação do usuário, o chamado deverá ser reaberto e o fluxo reiniciado.

Mapeado



Matriz de Responsabilidades

Atividades	Observações	Priorizador	Time (Suporte)	Time	Time (Infraestrutura)
Realizar Atendimento Inicial			R		
Garantir Registro Demanda no Assyst	Abrir ou Complementar informações da demanda em um chamado no Assyst		R/A		
Realizar Análise Inicial			R		
Verificar Solução na Base de Conhecimento do Sistema de 1º Nível		C	R		
Encaminhar Chamado para Time Especializado	Atribuição do Chamado no Assyst	C	R	I	

Priorizar Chamados	Ver Roteiro de Atendimento de Chamados no Assyst	R/C		R/C	
Realizar Análise Especializada	Ver Roteiro de Atendimento de Chamados no Assyst			R	
Verificar Solução na Base de Conhecimento				R	
Registrar Chamado Externo		C		R	
Criar Proposta de Solução		I/C		R	
Autorizar Solução		R		I	
Abrir Issue no Jira e Priorizar Requisição	Ver Procedimentos para criação de cartões virtuais (issues) no JIRA	R/A		R/I	
Desenvolver/Aplicar Solução Priorizada				R	
Encaminhar Alterações no BD				R	
Avaliar e Executar Script				C/I	R
Encaminhar alterações de código				R	
Integrar Código				C/I	R
Atualizar Conhecimento			R/I	R/A/I	R/A/I

Legenda:

- R - *Responsible* (Responsável)
- A - *Accountable* (Aprovador)
- C - *Consulted* (Consultado)
- I - *Informed* (Informado)

Indicadores de desempenho

Índice de Resolução de Chamados Dentro do Prazo

- Descrição: Percentual de chamados resolvidos cuja data de abertura tem menos de 15 dias
- Periodicidade: Quinzenal
- Forma de cálculo: $(TCFI / TCF) \times 100$
 - TCFA: Total de Chamados Fechados nos últimos 15 dias com duração Inferior a 15 dias
 - TCF: Total de Chamados Fechados nos últimos 15 dias
- Meta: 90%

Índice de Resolução de Requisições de Usuários VIPs Dentro do Prazo

- Descrição: Percentual de requisições de usuários VIPs resolvidos cuja data de abertura tem menos de 15 dias
- Periodicidade: Quinzenal
- Forma de cálculo: $(TRFI / TRF) \times 100$
 - TRFI: Total de Requisições Fechadas nos últimos 15 dias com duração Inferior a 15 dias
 - TRF: Total de Requisições Fechadas nos últimos 15 Dias
- Meta: 90%

Índice de Resolução de Requisições de Usuários Não VIPs Dentro do Prazo

- Descrição: Percentual de requisições de usuários Não VIPs resolvidos cuja data de abertura tem menos de 45 dias
- Periodicidade: Quinzenal
- Forma de cálculo: $(TRFI / TRF) \times 100$
 - TRFI: Total de Requisições Fechadas nos últimos 15 dias com duração Inferior a 45 dias
 - TRF: Total de Requisições Fechadas nos últimos 15 Dias
- Meta: 90%

Processo de Descontinuidade de Software

Índice

- [Índice](#)
 - [Objetivo](#)
 - [Definições e Abreviações](#)
 - [Papéis e Responsabilidades](#)
 - [Fluxo de Trabalho](#)
 - [Descritivo](#)
 - [Mapeado](#)
 - [Matriz de Responsabilidades](#)
 - [Indicadores de desempenho](#)
-

Objetivo

O objetivo deste processo é definir os passos da descontinuidade de *software* (desligamento, inativação) e possui os seguintes valores:

- Aprovação do Gestor de Solução de TIC quanto à descontinuidade;
 - Garantir o armazenamento das informações do *software* a ser descontinuado, caso necessário;
 - Padronização do processo;
 - Garantir a qualidade na descontinuidade do *software*.
-

Definições e Abreviações

Ver definições e abreviações na página do [Processos de Software](#).

Papéis e Responsabilidades

Os papéis do processo estão definidos na página do [Processos de Software](#).

Fluxo de Trabalho

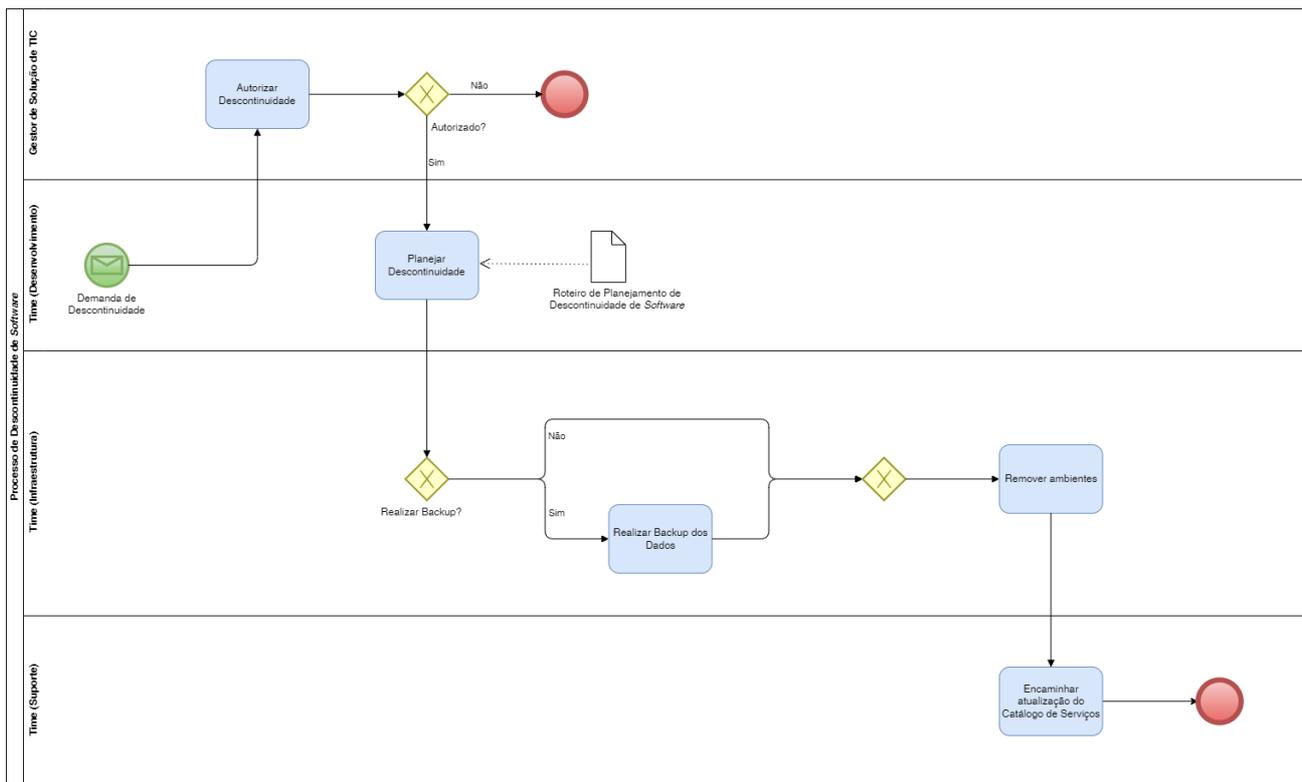
Descritivo

O fluxo de trabalho inicia com o recebimento da *demanda* de descontinuidade de *software* pelo Desenvolvimento de TIC (pode vir de normativos, do Gestor da Solução TIC, da administração do Tribunal, etc). A solicitação deve ser encaminhada ao Gestor de Solução de TIC, que irá autorizar (ou não) a descontinuidade.

Caso aprovada, é realizado o planejamento da descontinuidade, com a subsequente realização de backup, se necessário, e a remoção dos ambientes do *software* (produção, homologação, bugfix, testes, etc) pela Infraestrutura de TIC.

Por fim, o Suporte de TIC encaminha demanda ao Comitê de Gestão de TIC para remover o *software* do Catálogo de Serviços de TIC.

Mapeado



Matriz de Responsabilidades

Atividades	Observações	Time (Desenvolvimento)	Time (Infraestrutura)	Time (Suporte)	Gestor de Solução de TIC
Autorizar Descontinuidade		I	I	I	A/R
Planejar Descontinuidade		A/R	R/I/C	R/I/C	I/C
Realizar Backup dos Dados		I	A/R	I	I
Remover ambientes		I	A/R	I	I
Encaminhar atualização do Catálogo de Serviços		I	I	A/R	I

Legenda:

- R - *Responsible* (Responsável)
- A - *Accountable* (Aprovador)
- C - *Consulted* (Consultado)
- I - *Informed* (Informado)

Indicadores de desempenho