



UNIVERSIDADE ESTADUAL DE PONTA GROSSA
Av. General Carlos Cavalcanti, 4748 - Bairro Uvaranas - CEP 84030-900 - Ponta Grossa - PR - <https://uepg.br>

INSTRUÇÃO NORMATIVA NTI Nº 1/2022

O DIRETOR DO NÚCLEO DE TECNOLOGIA DA INFORMAÇÃO DA UNIVERSIDADE ESTADUAL DE PONTA GROSSA, no uso de suas atribuições legais e estatutárias, *considerando*:

a Resolução UNIV 2021.9 de 05 de Agosto de 2021,

R E S O L V E:

Art. 1º Publicar o documento "Processo de Desenvolvimento de Software - PDS-UEPG", 1ª Edição - 2022, na forma do Anexo que passa a integrar este ato legal.

Art. 2º Esta Instrução Normativa entrará em vigor na data de sua publicação.



Documento assinado eletronicamente por **Luiz Gustavo Barros, Diretor do NTI**, em 28/03/2022, às 15:16, conforme Resolução UEPG CA 114/2018 e art. 1º, III, "b", da Lei 11.419/2006.



A autenticidade do documento pode ser conferida no site <https://sei.uepg.br/autenticidade> informando o código verificador **0924723** e o código CRC **ECDAE2A4**.

Processo de Desenvolvimento de Software PDS-UEPG

Especificações Técnicas para o Desenvolvimento de
Software na UEPG

Processo de Desenvolvimento de Software PDS – UEPG

Reitor da Universidade Estadual de Ponta Grossa
Miguel Sanches Neto

Diretor do Núcleo de Tecnologia da Informação
Luiz Gustavo Barros

Gerente de Sistemas de Informação – NTI
Ademir Mazer Junior

Grupo de Trabalho
Ademir Mazer Junior – redator, elaborador
Luiz Gustavo Barros – revisor
Rafael Afonso Mayer – revisor

Versão	Data	Observações	Responsável
v1.1	31/01/2022	Revisão final para publicação	Luiz Gustavo Barros
v1	17/12/2021	Revisões	Ademir Mazer Junior Luiz Gustavo Barros Rafael Afonso Mayer
beta6	23/06/2021	Revisões da equipe Gestão de Demandas e Projetos + versão enviada para revisão Análise e Projeto de Software	Ademir Mazer Junior
beta5	14/06/2021	Revisões da equipe Gestão de Produto de Software + versão enviada para revisão Gestão de Demandas e Projetos	Ademir Mazer Junior
beta4	02/06/2021	Enviado para revisão Gestão de Produto de Software	Ademir Mazer Junior
beta3	01/06/2021	Revisões do relator + contribuições da equipe no capítulo introdução	Ademir Mazer Junior
beta2	15/05/2021	Primeira revisão	Luiz Gustavo Barros
beta1	13/05/2021	Enviado para primeira revisão	Ademir Mazer Junior
0.0.0	10/05/2021	Primeira versão para revisão	Ademir Mazer Junior

Introdução	5
Objetivo	7
Aplicação	7
Gestão	8
Visão geral do PDS-UEPG	8
Capacitação.....	8
Regras, especificações técnicas e referências	9
Definições dos subprocessos	9
Gestão de demandas e projetos.....	10
Análise e Projeto de Software.....	10
Desenvolvimento de software.....	10
Implantação.....	11
Atendimento ao Usuário/Cliente	11
Manutenção de produto de software.....	12
Dicionário de Conceitos e Siglas	12
Gestão dos Produtos de Software	15
Serviços relacionados	15
Gestão de Projetos	19
Gestão de Demandas.....	20
Papéis e atribuições.....	20
Documentação Técnica e de Usuário.....	24
Documentação técnica ou de projeto	24
Documentação de usuário (Base de conhecimento)	25
Subprocesso de Gestão de Demandas e Projetos.....	26
Registro de Demanda.....	26
Classificação de demanda.....	27
Planejamento de demanda	29
Análise de viabilidade de requisitos	29
Execução e acompanhamento de demanda.....	30
Gestão de projeto	30
Subprocesso de Análise e Projeto de Software	33
Artefatos	33
Artefatos de Análise e Projeto:	33
Gestão e armazenamento das informações	34
Subprocesso de Desenvolvimento de Software	35
Especificações Técnicas e Ferramentas Oficiais da UEPG	36
Softwares de apoio aos processos de TI	36

Gestão de Produto de Software e de Gerenciamento de Projeto de Software (técnico).....	36
Gestão de Atendimento ao Cliente	36
Gestão de Base de Conhecimento.....	36
SEI - Sistema Eletrônico de Informações	37
Arquitetura e tecnologias de referência para o desenvolvimento de sistemas.....	37
Tecnologias padronizadas	37
Ferramentas e ambientes sugeridos para o desenvolvimento	39
Pacotes homologados para Laravel	39
Regras para modificação de arquitetura	40
Tecnologias legado	40
Documentação de sistemas	41
Artefatos previstos para serviços de documentação	41

Introdução

No desenvolvimento de soluções para seus clientes, o NTI – Núcleo de Tecnologia da Informação, e demais unidades que desenvolvem soluções de TI, usam diversas tecnologias e ferramentas de apoio, tendo em vista a aplicação adequada desses recursos para cada processo de negócio a ser atendido.

A complexidade e as diferenças entre as áreas de negócio da instituição, associadas à grande oferta de tecnologia existente gera o risco da adoção, por vezes, de tecnologias e ferramentas divergentes para solucionar um mesmo problema. Conseqüentemente, em casos de adoções não padronizadas, o ambiente computacional torna-se mais heterogêneo, gerando custos adicionais para sua administração, manutenção e integração. Convém ressaltar que estes custos ao longo do tempo podem ser relativamente altos.

Para manter um ambiente computacional simplificado e integrado, o desenvolvimento e manutenção de produtos de software no NTI e unidades que desenvolvem soluções de TI, deve sempre levar em consideração a arquitetura existente, buscando usar ferramentas e infraestrutura já estabelecida e suportada. A adoção de novas tecnologias e ferramentas na arquitetura de software deve seguir um processo de avaliação e homologação prévia, onde são realizadas análises de ordem técnica, operacional e financeira.

Diante do exposto, é essencial que todo profissional de desenvolvimento de software da UEPG tenha o conhecimento e seja orientado em relação as ferramentas disponíveis e em quais situações devem ser utilizadas, em vez de adotar soluções alternativas que não venham a se enquadrar na arquitetura definida.

Desta maneira, este documento apresenta o ambiente computacional definido para o desenvolvimento de aplicações na UEPG, delimitando o escopo tecnológico na construção das aplicações, e as regras para uso e modificação desse ambiente.

Este documento: a norma de **Processo de Desenvolvimento de Software da UEPG – PDS-UEPG**, é resultado da experiência adquirida pelo Núcleo de Tecnologia da Informação – NTI desde sua criação, em 1985, como Centro de Processamento de Dados. Este documento também inclui estudos e assimilação de técnicas de Gestão de Projetos de Desenvolvimento de Software e boas práticas disseminadas no mercado.

O PDS-UEPG é definido visando a utilização de tecnologias, metodologias e padronizações de desenvolvimento e manutenção de software, procurando garantir a escalabilidade, portabilidade, usabilidade, manutenibilidade e acessibilidade dos sistemas computacionais. A obediência às normas e regulamentos é uma necessidade intrínseca à Administração Pública e, como não poderia deixar de ser, todos os sistemas da UEPG devem garantir que todas as ações estejam de acordo com essas normas – incluindo aquelas citadas na seção de Especificações Técnicas e Ferramentas Oficiais.

Neste sentido, o PDS-UEPG constitui-se em uma **norma de utilização obrigatória**, conforme dispõe a Resolução UNIV 2021.9 de 05/08/2021 (Regimento Interno do Núcleo de Tecnologia da Informação), no que se refere ao desenvolvimento e a manutenção de projetos de sistemas no âmbito da UEPG, descrevendo um conjunto de regras, padrões e tarefas imprescindíveis para a execução de projetos com qualidade, produtividade e segurança.

Desta maneira, este documento fornece as especificações conceituais e técnicas mínimas necessárias às quais os produtos ou serviços executados deverão atender, seja pelas áreas de TI da instituição ou por terceiros, deverão obrigatoriamente atender.

Objetivo

O objetivo desta norma é prover uma metodologia simples e eficiente para o desenvolvimento e manutenção de produtos de software na UEPG.

Sua concepção não deve ser definitiva ou estanque, sendo que, ao longo de sua utilização, de acordo com a maturidade do processo e das necessidades vindouras, adequações poderão ocorrer com a publicação de novas versões, visando a melhoria contínua do processo.

A aplicação deste Processo de Desenvolvimento de Sistemas (PDS) tem como principais objetivos:

1. Padronização de tecnologias, métodos e metodologias no processo de desenvolvimento de software no âmbito da UEPG;
2. Primar pelo atendimento dos requisitos de negócio estabelecidos para um projeto com eficácia e qualidade;
3. Possibilitar maior controle quanto a entrega de projetos no prazo definido;
4. Agregar qualidade ao produto, com flexibilidade para receber evoluções exigidas pelo dinamismo dos requisitos de negócio;
5. Possibilitar o acompanhamento e o gerenciamento dos processos de desenvolvimento de softwares na UEPG.

Aplicação

Esta norma aplica-se a todos os servidores, funcionários temporários, residentes técnicos, estagiários, consultores, prestadores de serviços e demais colaboradores da UEPG que necessitem solicitar, especificar e desenvolver sistemas para a instituição.

Sua aplicação estende-se a todas as unidades da UEPG que, por decisão administrativa ou orientação do NTI, possuam autorização para manutenção de equipes de TI setorializadas de desenvolvimento de sistemas.

As unidades que desenvolvem soluções de TI na instituição, bem como recursos externos contratados, para prestar serviços de desenvolvimento e manutenção de sistemas devem:

- Estar de acordo com o PDS–UEPG, com especial atenção as definições de processos, modelos de artefatos, ferramentas de software, tecnologias e critérios de aceitabilidade prescritos;
- Estar de acordo com Modelos de Gestão e Sustentação Ágil de Software que venham a ser definidos pelo NTI, e seus anexos, nos casos em que se aplique este modelo;
- Considerar a arquitetura tecnológica e ambiente de desenvolvimento do NTI, bem como a sua evolução natural;
- Integrar com as bases de dados informatizadas existentes no ambiente do NTI, sem criar modelos de dados que dupliquem dados e/ou regras de negócio já existentes e estabelecidas na instituição, salvo em situações excepcionais e aprovadas pelo NTI;
- Desenvolver sistemas modulares, quando necessário;
- Utilizar, para o desenvolvimento e manutenção dos produtos, as boas práticas definidas no PMBOK, ITIL, CMMI, MPS-BR, COBIT, RUP, XP, SCRUM, e outros modelos de boas práticas pertinentes e

relacionados aos contextos técnicos no desenvolvimento de sistemas ou a gestão de projetos, em especial com relação à engenharia e arquitetura de softwares, como o SWEBOK;

Gestão

Esta norma, no âmbito da UEPG, é elaborada, implementada e mantida pelo NTI, que é o órgão responsável pelas diretrizes dos processos de desenvolvimento de software, conforme disposto na Resolução UNIV 2021.9, Anexo, Artigo 4º, inciso XIX.

Sempre que necessário, o NTI pode estabelecer comissões temporárias com o intuito de estudo, análise e definição de subprocessos e regras de padronização a serem incorporados ou ajustados nas normas do PDS-UEPG.

Visão geral do PDS-UEPG

Um processo, nesta norma, é definido como "um conjunto de atividades que descreve entradas, adiciona valor e fornece saídas, gerando um produto ou serviço valorado para a instituição ou processo de negócio".

O PDS-UEPG é iniciado com a formalização de uma solicitação, sendo organizado e analisado para sua execução unitária ou em agrupamento com outras demandas, priorizado pelas unidades de negócio geradoras das demandas, e finalizado com a execução de um projeto de desenvolvimento ou manutenção de um ou mais softwares.

Um subprocesso é definido como um "conjunto de atividades" que pode ser representado como uma "grande atividade" no escopo geral do PDS-UEPG.

O PDS-UEPG é composto pelos subprocessos:

- Gestão de Demandas e Projetos
- Análise e Projeto de Software
- Desenvolvimento de Software
- Implantação
- Atendimento ao Usuário/Cliente

Cada subprocesso é composto por atividades intrínsecas ao seu contexto, podendo as mesmas estarem ligadas aos demais subprocessos, servindo de entrada/saída dos mesmos. Ainda, cada subprocesso pode ser organizado em grupos de subatividades.

É importante compreender que a listagem acima não esgota novos subprocessos ou suas reorganizações, bem como não define uma sequência de execução, visto ser uma orientação geral que o PDS-UEPG possa ser executado e gerenciado seguindo métodos de gestão ágil e adequado ao contexto de cada projeto e equipe.

Capacitação

Esta norma não se destina a esgotar os conceitos teóricos e de boas práticas de mercado existentes. Desta maneira, é obrigação dos profissionais seguindo este PDS, manterem-se atualizados e capacitados em tais conhecimentos.

O site do NTI (www.uepg.br/nti) pode ser consultado para referências a cursos, livros, artigos e documentação orientativa. Além do site, cada projeto deve organizar documentação de referência quanto às escolhas e padronizações de artefatos e técnicas utilizadas.

Regras, especificações técnicas e referências

Como citado na seção de Capacitação, esta norma não esgota detalhes conceituais dos métodos, técnicas ou ferramentas, porém, define as regras de quais tecnologias podem ou devem ser adotadas.

Define também, para contextos específicos, modelos padrão de artefatos, especificando regras obrigatórias quando necessário, ou ainda, especificando listas de controle (*checklists*) que definem passos ou obrigatoriedades em processos e subprocessos.

O PDS-UEPG, tem a finalidade de manter-se atualizado quanto às determinações das tecnologias e artefatos, porém, não esgota neste documento todas as orientações de definições de regras a serem seguidas nestas questões pontuais. Desta maneira, documentos anexos poderão ser criados e publicados com referência ao PDS-UEPG. Tais anexos serão organizados na mesma base oficial de documentos, sendo publicados e organizados para fácil acesso no site do NTI.

Tais documentos compreendem as definições pontuais e de pequena granularidade quanto a:

- Tecnologias e ferramentas (softwares) homologados e de uso obrigatório para os processos de desenvolvimento de software na UEPG, para os contextos de novos sistemas, módulos ou rotinas. Também definem as tecnologias e ferramentas para os contextos de aplicações legado;
- Referências teóricas, de uso prioritário quando da necessidade de tomadas de decisão conceitual;
- Ferramentas de gestão de processos de TI;
- Modelos de artefatos com suas regras e obrigatoriedades, além das listas de controle para cumprimentos em etapas, subprocessos e processos.

Definições dos subprocessos

Nesta seção são apresentados os conceitos gerais dos subprocessos cobertos pelo PDS-UEPG.

É importante salientar que os subprocessos em um PDS não devem ser vistos ou adotados de maneira a conduzir suas execuções sequencialmente, sendo que os mesmos devem ser, ao implantados, organizados e planejados para que ocorram de maneira paralela quando necessário.

Orienta-se a adotar práticas de metodologias ágeis, com o trabalho planejado em “janelas temporais de iteração”, e dentro destas, a execução dos subprocessos que, quando em intervalos menores de tempo, podem ser organizados de maneira sequencial se necessário, porém, abrangendo nestes contextos, conjuntos menores de requisitos e tarefas a serem cumpridas.

Gestão de demandas e projetos

Subprocessos que determina as fases e procedimentos para a identificação, registro, análise, definição de esforço/custo, priorização, planejamento, acompanhamento da execução e documentação de gestão do projeto como: aceites de requisitos, aceites de planejamento, aceite de entregas, registro de treinamentos.

Esta norma define um capítulo específico para tratar de diretrizes na Gestão de Demandas e Projetos no âmbito do PDS-UEPG.

Análise e Projeto de Software

Compreende os subprocessos e as etapas conhecidas nas disciplinas clássicas e ágeis de desenvolvimento de software, como nos subprocessos descritos a seguir:

- **Análise e Projeto:**
 - *Gestão dos requisitos:* diagramas e especificações de Casos de Uso, Histórias de Usuários e Épicas (SCRUM), mantidos e disponibilizados de acordo com as ferramentas de softwares e infraestrutura padrão definida no PDS-UEPG.
 - *Planejamento lógico e físico:* diagramas UML, Diagramas de Modelos de Dados (MER, IDEF1X), Dicionários de Dados, mantidos e disponibilizados de acordo com as ferramentas de softwares e infraestrutura padrão definida no PDS-UEPG.
- **Documentação técnica:** registro e manutenção necessária para a documentação do projeto de software, como: Regras de negócio, Glossário técnico e de negócio, diagramas necessários para compreensão do projeto de software, Orientações de implantação de ambientes e particularidades de publicação em produção. Mantidos e disponibilizados de acordo com as ferramentas de softwares e infraestrutura padrão definida no PDS-UEPG.
- **Documentação de usuário:** documentos de auxílio ao usuário e às equipes técnicas de suporte ao usuário:
 - Manuais de uso do sistema: FAQ, PDF com manual completo, vídeos, materiais para treinamentos, mantidos e disponibilizados de acordo com as ferramentas de softwares e infraestrutura padrão definida no PDS-UEPG.
 - Sistema oficial para manutenção de base de conhecimento de documentação de usuário, no formato de estrutura conhecida como FAQ, no sistema OTRS, acessível em <https://www.uepg.br/nti> – opção "Base de Conhecimento".

Desenvolvimento de software

Subprocesso que compreende a execução em si do produto, podendo, por vezes, compreender a instalação e configuração de sistemas de terceiros como, por exemplo, o Sistema Eletrônico de Informações (SEI).

As atividades e subprocessos de desenvolvimento envolvem a codificação em si, bem como todas as práticas de manutenção do código e *feedback* da evolução de execução, são elas:

- **Organização do repositório:** seguindo boas práticas de versionamento de código fonte e artefatos do sistema, por meio dos sistemas e ferramentas oficiais definidos pelo NTI:

- Versionamento de código fonte utilizando *GIT*.
- Gestão de repositório de códigos-fonte e gestão de projetos (*issues*, marcos, wiki de documentação técnica, devops, revisão de código): *GITLAB*, acessível em <https://git.uepg.br>.
- Esta norma possui um capítulo específico para definir as regras de gestão do repositório e versionamento de códigos nos projetos UEPG.
- **Organização dos ambientes:** com o auxílio e sob a orientação da padronização de ambientes de servidores da área de Redes e Infraestrutura, os responsáveis pelo desenvolvimento devem solicitar e organizar os ambientes para desenvolvimento, testes e produção para os projetos;
- **Testes:** escrever e organizar testes automatizados unitários, de integração, comportamento, segurança e desempenho. Bem como organizar documentação de planos e execução de testes: *esta documentação é parte do subprocesso de Análise e Projeto de Software*.

Implantação

Organizar a implantação em produção, bem como reprodução de sua publicação em ambientes de teste e desenvolvimento.

Este subprocesso compreende tanto documentos textuais, como *scripts* de implantação por execução manual ou disparo automatizado por meio de ferramentas de *Continuous Integration/Continuous Deployment* – CI/CD. O processo de implantação não será detalhado nesta versão deste documento.

Atendimento ao Usuário/Cliente

O subprocesso de atendimento e suporte ao usuário é parte essencial do PDS-UEPG, sendo auxiliado pela documentação organizada nos demais subprocessos, principalmente pela geração, organização e manutenção da documentação de usuário.

Neste documento o usuário final é definido como **cliente**, ainda que o relacionamento em alguns casos seja sem fins financeiros, o PDS-UEPG entende que a TI presta serviços aos usuários de seus produtos, sendo desta maneira eles considerados clientes que demandam tais serviços baseados em suas necessidades: definindo, especificando, testando, aprovando e adotando os resultados em seu contexto de negócio.

Boas práticas e métodos de gestão dos processos de atendimento ao usuário baseiam-se na metodologia ITIL como: Single Point of Contact – SPOC (ponto único de contato), Service Level Agreement – SLA (acordos de nível de serviços), Bases de conhecimento, Níveis de Suporte e Filas de Atendimento.

Dentre os benefícios com a gestão do processo de atendimento ao cliente, está o resultado de melhoria contínua dos serviços de TI. A metodologia PDCA – Plan, Do, Check, Act (Planejar, Executar, Verificar, Agir) encontra nos processos de atendimento a retroalimentação necessária para a sua verificação e, desta maneira, subsidiar com informações e conhecimento a tomada de decisão (ação) para replanejamentos e novas execuções. O processo de atendimento ao usuário/cliente não será detalhado nesta versão deste documento.

Manutenção de produto de software

Após a implantação de um novo software ou nova funcionalidade (módulo, rotina), estes passam à etapa de ciclo de vida de um software denominada *Manutenção*. A disciplina de metodologias de desenvolvimento de software, denominada *Gestão de Mudanças* engloba essa função, sendo tratado no PDS-UEPG usando todos os subprocessos: Gestão de Demandas, Análise e Projeto de Software, Desenvolvimento de Software, Implantação e Atendimento ao Usuário.

A manutenção de um produto de software deve ser contextualizada e priorizada quanto à correção de erros/bugs; correção de lógica/erros no resultado ao processo que apoia; e melhorias/construção de novas funcionalidades.

No PDS-UEPG a demanda de manutenção deve ser tratada com o mesmo rigor de planejamento e gestão das demandas para novos sistemas, visto que, em geral, acabam por trazer maior complexidade a gestão de produtos de software pelo alto volume de demandas que geram.

Os próximos capítulos tratam de cada subprocesso em específico, determinando diretrizes e regras para cada contexto.

Dicionário de Conceitos e Siglas

Esta seção descreve um conjunto de conceitos e siglas apresentados no documento PDS-UEPG e seus apêndices:

- **Ambiente:** sistema de computador (servidor), ou conjunto de sistemas, em que um software é implantado e executado. Exemplos:
 - Ambiente local: desenvolver e executar na mesma máquina do desenvolvedor;
 - Ambiente de desenvolvimento (“dev”): servidor para implantar e executar enquanto as alterações são feitas;
 - Ambiente de homologação (“stage”): servidor para validação de funcionalidades;
 - Ambiente de produção (“prod”): servidor onde os usuários finais usam o sistema.

Os diversos ambientes são isolados, podendo haver estágios intermediários entre eles como, por exemplo, ambientes de testes. O processo de gerenciamento de versão estruturado permite a implantação em fases (implementação), teste e reversão em caso de problemas.

- **Defeito de software:** Desacordo do sistema em relação a sua especificação, erro, falha, comportamento inadequado e comprometimento da segurança. Ao se julgar se um comportamento do sistema é ou não um defeito, considera-se a perspectiva do usuário, ou seja, é irrelevante se o defeito é ocasionado por falha da codificação, configuração de ambiente ou insuficiência de informação na especificação.
- **Deploy:** consultar *Implantação*.
- **Unidade de desenvolvimento de solução de TI (DESENVOLVEDOR):** equipes do NTI, áreas homologadas com equipes de TI da instituição, terceiros contratados para execução de serviços de TI sob a supervisão do NTI.

- **Homologação:** Fase na qual o sistema será disponibilizado no ambiente de homologação (“stage”), para que o gestor ou usuários finais realizem os testes de aceitação, verifiquem se o que foi encomendado corresponde ao sistema entregue e validem se o sistema contribuirá para a resolução do problema que motivou o seu desenvolvimento. O resultado desta fase pode ser de aceitação integral, aceitação parcial ou rejeição.
- **Implantação:** para o contexto do PDS-UEPG, significa enviar mudanças ou atualizações de um ambiente de execução para outro. Desta maneira, enviar e configurar as atualizações de um ambiente de desenvolvimento para o ambiente de stage e, quando testado e aprovado pelo cliente, enviar e configurar as alterações do ambiente de stage para o ambiente de produção.
- **O.S. - Ordem de Serviço:** documento que descreve um serviço a ser executado por terceiros contratados para desenvolvimento de sistemas. A OS terá de estar adequada à legislação, edital e contrato referente.
- **PMBOK - Project Management Body of Knowledge:** padronização que identifica e conceitua processos, áreas de conhecimento, ferramentas e técnicas da gestão de projetos. Importante verificar que não trata de gestão de projetos de software e sim de projetos em geral. Dentre suas boas práticas pode-se citar as disciplinas que auxiliam projetos de desenvolvimento de sistemas como: gestão de comunicação, gestão de contratos, gestão de custos.
- **Pré-Homologação:** Fase de recebimento de uma versão do sistema entregue pelo DESENVOLVEDOR para realização dos testes de aceitação. Nesta fase, o sistema será primeiramente avaliado quanto a critérios de admissibilidade obrigatórios, e em seguida quanto à presença de defeitos de caráter impeditivo e conformidade com o escopo planejado. O resultado desta fase pode ser de rejeição da entrega ou disponibilização da versão para início dos Testes Formais ou Homologação.
- **Release:** Um release é o termo utilizado para designar uma versão operacional estável e executável do sistema, ou parte dele, incluindo todos os artefatos necessários para o seu uso, manutenção e novos desenvolvimentos.
- **Stage:** no contexto do PDS-UEPG, stage são ambientes de “estágio” de teste ou pré-produção. É um ambiente de teste que se assemelha exatamente a um ambiente de produção real o mais próximo possível e pode se conectar a outros serviços de produção e dados, como bancos de dados. O uso principal de um ambiente de teste é testar todos os scripts e procedimentos de instalação / configuração / migração antes de serem aplicados a um ambiente de produção. Também podem ser testados o desempenho e carga, já que geralmente é sensível ao ambiente. É o ambiente disponibilizado ao cliente principal para que execute seus testes e os valide em relação aos requisitos acordados para uma iteração do software. A partir dos testes no ambiente stage, ocorrendo todos os testes como esperado pede-se a aceitação do cliente e pode-se implantar em produção.
- **Tempo de Atendimento - TA:** Intervalo máximo de tempo que o DESENVOLVEDOR tem, após a abertura formal do chamado, para entrar em contato com o SOLICITANTE e entender o contexto da solicitação.

- **Tempo para Solução do Defeito - TSD:** Intervalo máximo de tempo que o DESENVOLVEDOR tem, após a abertura formal do chamado, para solucionar o defeito. A solução do defeito pode ser uma solução de contorno.
- **Tempo para Solução Definitiva do Defeito - TSDD:** Intervalo máximo de tempo que o DESENVOLVEDOR tem, após a abertura formal do chamado, para solucionar definitivamente o defeito. O conserto aplicado deve ser definitivo e eliminar a causa raiz do defeito.
- **TDD – Test Driven Development:** Desenvolvimento Orientado a Testes, com o objetivo de escrever primeiramente os testes automatizados para uma função ou funcionalidade, e então sua lógica, a qual deverá estar finalizada quando todos os testes escritos estiverem satisfeitos.
- **Testes de Aceitação:** São conjuntos de testes unitários e testes funcionais automatizados (quando solicitados), além de testes funcionais manuais, capazes de verificar o provimento das qualidades mínimas necessárias à aceitação de um software, conforme definido em sua especificação.
- **Testes Formalizados:** Fase na qual o NTI avaliará a qualidade do produto entregue através de várias práticas, como por exemplo: (i) localizar e documentar defeitos no software; (ii) validar as definições feitas nas especificações de requisitos e projeto; (iii) validar as funções do software conforme projetadas e; (iv) verificar se os requisitos foram implementados de forma adequada. O resultado desta fase pode ser de aceitação integral, parcial ou rejeição.

Gestão dos Produtos de Software

A partir do momento em que um software é implantado para uso (em sua completude ou parcialmente), ele se torna um **Produto** que deve ser mantido por uma equipe de TI.

Um **Produto de Software** é um ativo de alto valor para a instituição, servindo como ferramenta de apoio ao dia a dia operacional das áreas de negócio, bem como de fornecedor de análises de dados e informações para a tomada de decisão da gestão estratégica da instituição.

Os produtos de software mantidos pela UEPG devem ser organizados, de acordo com os conceitos e boas práticas definidos pela disciplina de *Gerência de Configuração (GC, ou GCS – Gerência de Configuração de Software)*, usando o sistema de repositório e gestão de projetos de software oficial da instituição, especificado na seção correspondente dos apêndices de especificação técnica.

Com o uso de regras padronizadas para a Gestão de Demandas e Projetos na construção de software no âmbito da UEPG, uma série de benefícios podem ser alcançados ou maximizados:

- Aumento no controle de recursos, com o objetivo de minimizar a aplicação errônea das capacidades disponíveis das equipes de TI, procurando evitar tanto a sobrecarga quanto a subutilização;
- Maior acuracidade no planejamento e cumprimento de prazos;
- Melhor alinhamento estratégico com a instituição na entrega dos serviços e produtos de TI;
- Transparência: nos projetos em andamento, no atendimento das demandas, na capacidade disponível, na priorização e alocação de recursos;
- Aumento constante na agilidade de resposta a mudanças.

Serviços relacionados

Os serviços relacionados ao gerenciamento de produtos de software são classificados como:

- 1 **Desenvolvimento e evolução de sistemas:** Processo incremental e iterativo destinado à construção de um software novo ou um lançamento de versão (*release*) de software existente, a partir de especificações funcionais e não-funcionais definidas.
 - 1.1 O serviço de desenvolvimento e evolução de software tipicamente compreenderá as atividades de atualização das especificações de requisitos, do projeto de sistema e da modelagem de dados, construção (codificação), especificação e execução de testes, elaboração de manuais do usuário e implantação, majoritariamente desenvolvidos de acordo com o PDS-UEPG.
 - 1.2 As atividades de gerenciamento, planejamento, acompanhamento e controle do serviço também fazem parte do escopo desta etapa, de modo que o DESENVOLVEDOR deverá elaborar e manter atualizado o cronograma de atividades periodicamente no sistema de gestão de projetos e demandas NTI, descrito na seção de Especificações Técnicas e Ferramentas Oficiais do PDS-UEPG.
- 2 **Manutenção e sustentação de sistema:** Consiste na manutenção continuada ou sob demanda de um software, estendendo-se até o momento em que for substituído ou descontinuado e abrangendo:
 - 2.1 **Manutenção corretiva:** consistem na correção de incidentes, problemas, erros e defeitos, incluindo sua documentação técnica, para eliminar falhas ocorridas que provoquem a interrupção

inesperada de seu funcionamento ou a eliminação de comportamentos do software que divirjam de suas especificações e requisitos preestabelecidos.

2.2 **Manutenção evolutiva:** Consiste em mudanças de regras de negócio durante o ciclo de vida do negócio e por consequência do software. Pode também estar relacionada às personalizações que influenciam nas funcionalidades ou operações do sistema como: inclusão ou alteração de campos, tabelas, relatórios, leiautes, críticas de entrada de dados e melhorias na navegação.

2.2.1 Atualização de documentação e exclusão de campos ou tela também fazem parte da manutenção evolutiva.

2.3 **Manutenção adaptativa tecnológica:** Consiste na alteração do sistema para adaptá-lo às mudanças do ambiente computacional (compreendendo *hardware* e *software*), onde foi desenvolvido ou onde é executado, inclusive atualização de documentação técnica e melhorias de desempenho que não impliquem em inserção, alteração ou exclusão de funcionalidades e/ou regras de negócio;

2.3.1 São considerados os seguintes componentes tecnológicos passíveis de adaptação: Sistema Gerenciador de Bancos de Dados, Servidor de Aplicações, bibliotecas e/ou *frameworks* utilizados e as evoluções da própria linguagem computacional utilizada.

2.3.2 Englobam, ainda, outras adequações no sistema sem alteração do escopo da funcionalidade ou regra de negócio, mediante intervenção direta no código-fonte;

2.3.3 São considerados, a título de Sustentação de Sistemas, até uma adaptação em cada sistema, nos subitens abaixo a cada período de 1 (um) ano nas seguintes hipóteses:

2.3.3.1 Atualização de versão de navegadores internet;

2.3.3.2 Atualização de servidor de aplicação;

2.3.3.3 Atualização de versão de servidor de banco de dados;

2.3.3.4 Atualização de versão de linguagem de programação;

2.3.3.5 Atualização de versões de frameworks e/ou bibliotecas.

2.4 **Manutenção de interface localizada:** consiste em alteração de interface de usuário que não implique alteração das regras de negócio, de Caso de Uso ou requisitos de qualquer natureza, e que seja realizada de forma localizada, isto é, pela intervenção em um único arquivo ou em um pequeno conjunto de arquivos. Tal manutenção pode ser exemplificada da forma que se segue:

2.4.1 Fontes de letra, cores, logotipos, mudanças de botões, alteração na posição de campos e texto na tela;

2.4.2 Mudanças de texto em mensagens do sistema, título de um relatório ou *labels* de uma tela de consulta;

2.4.3 Mudanças de texto estático em e-mail ou outras notificações enviadas pelo sistema;

3 Atendimento e Suporte:

3.1 Prestação de esclarecimentos, documentação e treinamento, quanto à forma como foram implementados os requisitos do sistema, aos procedimentos requeridos ao seu correto funcionamento ou aos dados mantidos por ele.

3.2 Apoio à identificação e isolamento de falhas e problemas na execução do software;

3.3 Tais atividades devem seguir os procedimentos e fluxo de Atendimento ao Cliente descrito no PDS-UEPG com o objetivo em manter a padronização e qualidade nos atendimentos.

- 4 **Atividades especiais:** Devem ser consideradas como atividades a serem adotadas em situações de exceção nas soluções de software mantidas, sendo necessário sua identificação em manuais técnicos dos produtos de software, indicando: (a) quando são necessárias serem executadas, (b) quem (papéis) tem a autorização de execução, (c) como devem ser executadas, (d) quais os motivos de não serem implementadas como funcionalidades do produto de software e disponibilizadas para execução pelo usuário.

Consiste na preparação de roteiros de execução em linguagem SQL, ou outra adequada ao caso, destinados às extrações de dados não cobertas pelos relatórios do sistema, à correção de inconsistências nos dados mantidos pelo sistema e não realizáveis por meio das interfaces de usuário disponíveis (ou cujo volume inviabilize a sua execução de forma manual), ou à inserção de dados não automatizada no sistema. A criação de scripts para execuções disparadas por comandos manuais em linguagem padronizada pela seção de Especificações Técnicas. Abrange também:

- 4.1 Elaboração de pesquisas às bases de dados para atender a consultas ou inserção/correção de dados diretamente nas bases de dados de produção dos sistemas. Exemplos de atividades:
- 4.1.1 Intervenção em bases de dados para correção de dados ou informações;
 - 4.1.2 Elaboração de relatórios, geração de arquivos e consultas ad-hoc;
 - 4.1.3 Criação e execução de rotinas de limpeza e adequação de dados;
- 4.2 Criação e atualização de manuais e roteiros de orientação aos usuários;
- 4.3 Todas as atividades de intervenção “manual”, necessárias para disponibilizar e manter a solução desenvolvida nos ambientes do NTI.

- 5 **Rotinas operacionais:** Consiste na execução de quaisquer procedimentos operacionais rotineiramente requeridos pelo sistema em função de suas regras de negócio ou forma de construção. Consistem nas rotinas configuradas para execuções temporizadas de maneira automatizada. Sua documentação técnica deve descrever, além do propósito da rotina, a configuração de horários e periodização de execução sugeridos.

5.1 Rotinas operacionais que não estão configuradas e devem ser executadas manualmente são consideradas Atividades especiais, e devem atender aos requisitos definidos de uma atividade especial.

6 **Migração de dados:**

6.1 Elaboração de rotinas visando à transferência de dados de uma base para outra, podendo ser de um sistema gerenciador de banco de dados (SGBD) para outro ou no mesmo SGBD. Essa transferência poderá representar mudanças de estrutura ou não. Tal qual as Atividades Especiais e as Rotinas Operacionais, sua documentação técnica deve descrever, além do propósito da integração, a configuração de horários e periodização de execução sugeridos aos scripts de sincronização. Exemplos de atividades:

- 6.1.1 Criação de script;

6.1.2 Validação e limpeza de massa de dados;

6.1.3 Adequação e dicionarização do modelo dados e dicionário de dados;

7 Testes em sistemas:

7.1 Atividades para realizar testes em sistemas, incluindo os testes de aceitação de usuários visando apoiar a equipe nas mudanças realizadas. Testes devem ser realizados, em qualquer circunstância, mesmo em casos de manutenção emergencial em dia não útil ou fora do horário comercial.

7.2 Implementação de testes automatizados:

Em casos de serviço de Manutenção e Sustentação de Sistemas desenvolvidos por terceiros, englobará ainda as atividades de gerenciamento, planejamento, acompanhamento e controle do projeto, de modo que o RESPONSÁVEL pelo projeto deverá elaborar e manter atualizado o cronograma de atividades periodicamente e/ou sempre que lhe for solicitado. Ademais, a critério do NTI, poderá também contemplar atividades de levantamento de requisitos, análise e projeto de sistema e modelagem de dados, majoritariamente desenvolvidos de acordo com o PDS-UEPG.

Toda documentação e código fonte produzidos durante a execução dos serviços serão armazenados no repositório de código fonte oficial da instituição.

Gestão de Projetos

É comum o *produto de software* ser chamado de *projeto*, levando ao equívoco do conceito descrito pelo PMBOK, que define “um projeto como um esforço temporário, ou seja, finito. Tem, portanto, início e fim bem determinados e empreendidos para se alcançar um objetivo exclusivo, ou seja, um resultado específico que o torna único”.

A partir da publicação do PDS-UEPG, instaura-se a **obrigatoriedade** de trabalhar com manutenção de sistemas (correções, alterações e melhorias) e criação de novos softwares (inexistentes, módulos, novas rotinas) por meio da sua organização em **Projetos**, segundo a definição do PMBOK e diretrizes definidas pelo PDS-UEPG.

O PDS-UEPG define que: toda demanda de nova ferramenta de apoio às áreas de negócio, alteração de produtos de software existentes, sendo elas melhorias ou possíveis correções; passem por um processo de Gestão de Mudanças, o qual resultará em um *projeto* a ser planejado e executado, de acordo com as análises e negociações de complexidade, alinhamento estratégico e priorização.

Um projeto será organizado a partir das demandas identificadas e registradas pelas áreas de negócio, sendo que, dependendo da complexidade, coesão de processo e organização dos recursos de TI, um projeto poderá ser composto por uma única demanda ou um conjunto delas.

Todo projeto de software deverá identificar em sua composição inicial:

- Líder de Projeto – podendo existir mais de um por projeto
- Equipe técnica, de acordo com a natureza e necessidade do projeto: Programadores, Analistas de Sistemas, de Testes, de Métricas dentre outros
- Unidade de Área Coordenadora no NTI responsável pelo projeto
- Responsável pelo Processo (Cliente principal)
- Requisitante(s)/cliente(s), se diferente do cliente principal na definição de regras em subprocessos sendo atendido pelo projeto. Em geral, se o projeto agrupa uma série de demandas a atender, e estas demandas são de diferentes requisitantes, estes são os clientes identificados
- Demandas que compõem o projeto

Durante o desenvolvimento do projeto, mais informações e artefatos são obrigatórios para o bom andamento e correta documentação futura do projeto e/ou produto de software. Em casos de projetos que abrangem demandas de alteração de sistemas, deve atender os itens:

- Análise de esforço
- Planejamento e realizado da execução
- Levantamento de requisitos
- Modelos e diagramas de software utilizados para o atendimento, tais como: de dados, classes, sequência, fluxogramas, de processo
- Manual do sistema
- Aceites do usuário – requisitos, planejamento, testes, treinamento, etc.
- Gestão atualizada de contratos: em caso de projetos que utilizam serviços de terceiros por qualquer meio.

Necessário que o projeto mantenha atualizado o planejamento e uso efetivo da métrica relacionada ao

contrato, como: pontos de função, horas/homem, etc. Além do controle de uso da métrica contratada, os documentos que possam ser responsabilidade de elaboração e entrega do contratado em relação ao projeto.

- Relatório resumido de conclusão descrevendo os sucessos do projeto, principais problemas, eventuais não atendimentos planejados.

Os formatos de arquivo ou ferramentas a serem utilizadas para a gestão e controle das informações dos projetos estão descritas na seção “Especificações Técnicas e Ferramentas Oficiais da UEPG”.

Gestão de Demandas

A Gestão de Demandas de TI tem o objetivo de identificar, organizar e priorizar chamados, solicitações, requisições e iniciativas que necessitam de esforços do NTI e demais áreas na execução de serviços de TI no desenvolvimento de software da UEPG.

É estratégico, pois direciona as ações dos recursos disponíveis para a execução das demandas definidas dentro de planos estratégicos da administração da instituição.

O PDS-UEPG define como demandas apenas as criadas no sistema de gestão de demandas do NTI. Solicitações de *Chamados de suporte* criados no sistema de gestão de chamados do NTI, ou encaminhadas por meio de processos do sistema de tramitação eletrônica de documentos da instituição, deverão ser analisados e classificados como de atendimento cíclico e habitual (serviços), ou como demandas administrativas a serem analisadas e priorizadas.

Papéis e atribuições

Para o bom funcionamento dos processos do PDS-UEPG, esta seção define papéis dos envolvidos, que são classificados como de gestão, execução técnica e negócio, com atribuições e responsabilidades a serem executadas durante todo o ciclo de demanda, desde a sua identificação até a entrega.

Papéis de **Negócio** estão relacionados aos clientes de TI, responsáveis pelos processos a serem auxiliados pelos sistemas; de **Gestão** realizam tarefas de gestão e acompanhamento nos serviços de TI; e **Técnicos** realizam as atividades de operação técnica de todos os níveis para a construção dos sistemas.

Cada papel pode ser executado em um determinado projeto por mais de um profissional, bem como, um mesmo profissional em determinados projetos pode realizar diversos papéis. A organização de papéis por projeto deve ser definida após análise de esforço, situação do comprometimento dos recursos existentes de TI, complexidade e priorização em relação aos interesses da gestão da instituição. A organização das equipes por projeto deve seguir a ordem de decisão proferidas pelas equipes de gestão do NTI: Diretor, Gerente de Sistemas de Informação, Coordenador de Área ou Gestor de TI em Unidade de Processo.

A seguir a tabela de papéis em ordem alfabética:

<i>Papel</i>	<i>Contexto</i>	<i>Definição e atribuições</i>
Analista de Infraestrutura	Técnico	- Profissional responsável em definir e garantir o atendimento das necessidades de infraestrutura de software/hardware para execução do software sendo desenvolvido ou mantido; - Responsável pela definição e preparação, verificação e configuração do ambiente onde o software será executado;

		<ul style="list-style-type: none"> - Responsável pela definição de procedimentos em como solicitar e utilizar os ambientes para desenvolvimento, testes e produção; - Este papel deve ser executado por profissional lotado na Gerência de Infraestrutura de TI.
Analista de Métricas	Técnico	<ul style="list-style-type: none"> - Responsável por realizar estimativas, medições ou validações de tamanho funcional e não funcional de um projeto (desenvolvimento ou manutenção) ou de um software instalado, seguindo os padrões estabelecidos pelos organismos internacionais, guias e diretrizes contratuais da instituição, auxiliando na geração dos indicadores e respectivas análises e sugestões de ações; - Deve apoiar análises de viabilidade e impacto, gestão de requisitos e controle de qualidade de artefatos de acordo com as melhores práticas; - Apoiar a equipe na derivação do tamanho do projeto em esforço, prazo e custo, ponderação das iterações (ou ordens de serviço) e definição da melhor estratégia para o desenvolvimento ou manutenção de um software, conforme as regras estabelecidas na instituição; - Responsável por manter a base histórica de medições dos projetos e sistemas, realizar análises estatísticas, apoiar a calibração contínua dos modelos de estimativas e análises de produtividade, tendências e desvios e evoluir os modelos, normas, padrões e procedimentos de medição e análise na instituição, além de apoiar a melhoria dos demais processos; - Este papel pode ser executado pelo analista de sistemas responsável ou líder do projeto ou coordenador de área.
Analista de Requisitos	Técnico	<ul style="list-style-type: none"> - Responsável pela especificação dos requisitos e/ou do termo de referência da licitação; - Identifica as necessidades do usuário e as formaliza em documentação técnica que orientará o desenvolvimento ou manutenção de um software; - Valida os requisitos com o responsável pelo processo, garantindo a gestão de aceites e documentação organizada dos requisitos; - Este papel pode ser executado pelo analista de sistemas responsável ou líder do projeto ou coordenador de área.
Analista de Segurança da Informação	Técnico	<ul style="list-style-type: none"> - Apoiar na definição das regras de segurança, no planejamento e verificação dos riscos envolvidos no desenvolvimento e implantação do software; - Este papel pode ser executado pelo analista de sistemas responsável, líder do projeto ou coordenador de área.
Analista de Sistemas	Técnico	<ul style="list-style-type: none"> - Responsável pela análise e projeto dos sistemas; - Responsável por produzir e manter a documentação técnica e de usuários dos sistemas e projetos; - Revisar tecnicamente as soluções de código dos programadores; - Mantém o líder do projeto informado do andamento das suas tarefas nas iterações e no projeto como um todo; - Garante que as estruturas arquiteturais dos sistemas se mantêm em conformidade com as definições técnicas do PDS-UEPG; - Mantém-se atualizado nas tecnologias e metodologias de sua área de atuação, compartilhando o conhecimento com a equipe nas ferramentas de gestão de base de conhecimento; - Este papel pode ser executado pelo analista de sistemas responsável, líder do projeto ou coordenador de área.

Analista de Teste	Técnico	<ul style="list-style-type: none"> - Responsável pela iniciação, planejamento, execução e finalização do processo de teste; - Cria os artefatos de testes, utilizando técnicas, critérios e tipos de teste adequados ao projeto de software a ser testado; - Elabora os procedimentos para a execução dos testes; - Este papel pode ser executado pelos desenvolvedores, analista de sistemas ou pelo coordenador de área.
Área Responsável pelo Processo	Negócio	<ul style="list-style-type: none"> - Unidade administrativa executora de um processo de negócio na instituição; - Responsável pela definição de regras a serem implementadas nos sistemas por meio dos Requisitantes e Responsáveis pelo Processo; - Possui equipe de multiplicadores para o atendimento aos clientes em relação aos fluxos do negócio apoiados por produtos de software.
Arquiteto de Software	Técnico	<ul style="list-style-type: none"> - Define tecnologia padrão para o desenvolvimento, plataforma, componentes, <i>frameworks</i> e demais informações necessárias à especificação da arquitetura de softwares na instituição; - Este papel pode ser executado pelo Gerente de Sistemas de Informação ou coordenador de área.
Comitê de Gestão de Mudanças / Demandas	Gestão/Negócio	<ul style="list-style-type: none"> - Grupo composto por representantes das áreas de negócio da instituição e de TI, que avaliarão e apoiarão nas análises de impacto do estudo de viabilidade e no planejamento e priorização da execução das demandas de software; - Formado pelas chefias responsáveis por cada unidade (ou representante por ela definido) demandante de esforço de TI para o desenvolvimento de software – Responsáveis pelo Processo, o Diretor do NTI e Gerente de Sistemas de Informação; - De acordo com a área da demanda, as ações de priorização podem ser formadas também pelo Coordenador de Área de Sistemas, quando necessário.
Coordenador de Área de Sistemas	Gestão / Técnico	<ul style="list-style-type: none"> - Responsável pelos produtos de software e projetos alocados à sua área, quanto à aderência ao PDS e cumprimento de níveis de qualidade e prazos; - Responsável pelo ciclo de vida de projetos e sistemas, além da coordenação das atividades das equipes de análise e programação; - Participa do gerenciamento das demandas, análise de viabilidade, estratégias de definição dos projetos; - Participa da escolha das estratégias de desenvolvimento e definição das respectivas iterações e liberações internas, até a implantação da solução e do acompanhamento do ciclo de manutenção e suporte; - Traduz as necessidades de negócio para uma visão organizada e técnica, própria da área de TI, analisando tendências e propondo novas soluções e melhorias; - Realiza o mapeamento dos processos de negócio, além de realizar ou facilitar os treinamentos e repasse de conhecimento; - Responsável por garantir o cumprimento das normas e padronizações do PDS-UEPG em sua área; - Responsável por garantir a documentação dos projetos e planejamento de treinamentos em uso dos sistemas; - Responsável por acompanhar ou executar (na ausência de profissional específico) o serviço do Analista de Métricas e Gestão das OS de desenvolvimento de serviços externos/terceirizados; - Responsável por todas as atividades técnicas e administrativas de sua área;

		<ul style="list-style-type: none"> - Responsável pela gestão dos produtos de software alocados à sua área; - Responsável por garantir a prestação contínua de informações à sua gerência e clientes, sobre o andamento dos projetos, aplicações de normas, qualidade dos produtos de sua área; - Quando necessário executa os demais papéis técnicos exigidos para um projeto de desenvolvimento de sistemas.
Equipe de Desenvolvimento	Técnico	<ul style="list-style-type: none"> - Desenvolvedores alocados para o projeto; - Podem executar este papel: Analistas de Sistema, Analistas de Informática, Técnicos de Informática, Estagiários, Temporários, Residentes Técnicos ou outro integrante da área de TI com conhecimento possível de realizar o papel de programador na instituição.
Gerente de Sistemas de Informação	Gestão	<ul style="list-style-type: none"> - Responsável pelo PDS; - Facilita, organiza e negocia a alocação de pessoal nos projetos, auxiliando os coordenadores de área; - Pode executar papéis técnicos ou de gestão em projetos de desenvolvimento de sistemas; - Audita os projetos e sua aderência ao PDS e atendimento de qualidade; - Gerencia, coordena e realiza interlocução com os Clientes Principais, donos de processo, negociando prioridades e criticidades.
Líder de Projeto	Gestão/Técnico	<ul style="list-style-type: none"> - Responsável por organizar o ciclo de vida de um projeto de desenvolvimento de software, garantindo o bom andamento dentro dos prazos planejados para execução; - Responsável pelo planejamento de execução do projeto de desenvolvimento e/ou manutenção de software; - Acompanha os programadores e analistas de sistema na entrega iterativa ao cliente final - Realiza a implantação ou garante a implantação em produção das entregas aceitas - Coordena a interação com o cliente: planejamento de prazos, planejamento de testes, planejamento de treinamentos, aceites dos clientes: requisitos, prazos, testes, implantação - Este papel pode ser executado pelo Coordenador de Área de Sistemas, Analistas ou Técnicos efetivos ou temporários.
Multiplicador	Técnico/Negócio	<ul style="list-style-type: none"> - Responsável por participar de treinamentos e capacitações no desenho do processo e uso do software, com o objetivo de disseminar o conhecimento para os usuários dos sistemas; - Este papel pode ser executado pelo Requisitante e/ou Responsável pelo Negócio ou integrantes de sua equipe.
Programador	Técnico	<ul style="list-style-type: none"> - Responsável por realizar todas as atividades relacionadas a codificação e testes automatizados unitários e de comportamento, conforme as normas, padrões e diretrizes do PDS-UEPG; - Implementar os requisitos de acordo com as análises e projetos de sistemas; - Implementar e manter scripts de implantação; - Implementar e manter scripts de transformação de estruturas de bancos de dados (<i>migrations</i>); - Aplicação de boas práticas com o objetivo de garantir a segurança da informação, seguindo as diretrizes e padrões da instituição; - Aplicação de boas práticas de desenvolvimento e adequação à padronização técnica do PDS-UEPG;

		<ul style="list-style-type: none"> - Elaboração e manutenção de documentação técnica e de usuário (Manuais, FAQ, etc.); - Este papel é executado por integrantes de Equipe de Desenvolvimento.
Requisitante / Cliente	Negócio	<ul style="list-style-type: none"> - Responsável por identificar e relatar as demandas, informando necessidades, expectativas e detalhes de regras a serem incorporadas nos sistemas; - Profissional componente das equipes de negócio, atuando nos testes e fornecendo informações para as tomadas de decisão do Responsável pelo Negócio e Líder do Projeto.
Responsável pelo Processo / Cliente Principal	Negócio	<ul style="list-style-type: none"> - Responsável por definir as regras de negócio dos sistemas; - Responsável por gerenciar as demandas de TI levantadas por suas equipes de negócio, na aprovação e definição de prioridades; - Papel executado por cada responsável por unidade de negócio na instituição (chefias) de acordo com a contextualização dada ao processo na instituição. Processos transversais (mais de uma unidade) poderão identificar mais de um responsável pelo processo; - Define o Desenho do Fluxo de Processo; - Decide nas reuniões de: levantamento, validação e aceite de requisitos, desenho de processo, planejamento de projeto; - Gerencia as equipes de negócio no suporte ao usuário direto em relação ao uso de sistemas quando ligado às regras de negócio; - Gerencia as equipes de negócio na realização de testes em alterações, melhorias e novas funcionalidades a serem implantadas nos sistemas sob sua gestão de negócio; - Confirma o aceite de definição de requisitos, prazos negociados, testes realizados; - Define nas suas equipes de negócio, possíveis substitutos para atuarem como Responsável pelo Processo em subprocessos de sua área.

Documentação Técnica e de Usuário

A documentação é um conjunto de artefatos gerados nos vários processos de gestão de produtos de software e podem ser definidos, de maneira ampla e simplificada, como documentos do produto.

No contexto do PDS-UEPG classifica-se em dois grandes segmentos os artefatos gerados: Documentação técnica e Documentação de usuário.

Documentação técnica ou de projeto

A documentação técnica caracteriza-se como todo documento, ou conjunto de informações armazenada em uma ferramenta de gestão de projeto ou produto, que, organiza o conhecimento do produto de software com o objetivo a dar suporte a equipe técnica que o desenvolve ou mantém.

No PDS-UEPG, todo material produzido com foco técnico, deve ser armazenado e gerenciado, na ferramenta oficial de gestão de produto e desenvolvimento de software.

A lista, não exaustiva e não sequencial, de documentos técnicos engloba:

- Requisitos
- Histórias épicas e de usuário

- Descrição e desenho do fluxo de processo do negócio
- Testes: planos, casos de teste, relatórios
- Modelos e dicionários de dados
- Modelos de projeto de software: UML, BPMN, Casos de Uso
- Orientações de implantação

Notar que a documentação com foco no auxílio ao atendimento ao cliente é considerada documentação de usuário. A mesma é especificada na próxima seção.

Documentação de usuário (Base de conhecimento)

A documentação de usuário caracteriza-se como todo documento, ou conjunto de informações armazenada em uma ferramenta ou meio acessível ao usuário, que, organiza o conhecimento do processo de negócio e de uso do produto de software.

Aceita-se uma exceção quanto ao nível de acesso para a documentação, quando o conjunto do conhecimento possui teor técnico, focado na equipe de suporte ao cliente, e este conhecimento não necessita ou não deve ser acessado pelo usuário. Porém, deve-se usar a mesma ferramenta para organizar este conhecimento, visto que o mesmo não tem o objetivo de auxiliar a equipe de desenvolvimento ou manutenção do produto de software.

No PDS-UEPG, todo material produzido com foco no ensino e suporte ao uso do produto por parte do usuário, ou por parte da equipe técnica de suporte ao cliente, deve ser armazenado e gerenciado, na ferramenta oficial de gestão de base de conhecimento de atendimento.

A lista, não exaustiva, de conhecimentos para o usuário e equipe de suporte de atendimento engloba:

- Manuais de uso
- FAQ – Frequent Asked Questions: perguntas frequentes
- Tutoriais
- Procedimentos Operacionais Padrão
- Treinamentos gravados em vídeo
- Orientações de execução de rotinas por parte da equipe de atendimento

Subprocesso de Gestão de Demandas e Projetos

O PDS-UEPG define o processo da gestão de demandas, desde seu registro por meio de solicitante/requerente/cliente externo ou interno de TI, passando por sua classificação em serviço/demanda, classificação de urgência, priorização junto as áreas de negócio e seu planejamento para execução em um projeto.

Todos os sistemas oficiais e ferramentas homologadas que dão suporte aos serviços descritos nesta seção estão detalhados especificado na seção correspondente dos apêndices de especificação técnica.

Os documentos que necessitam de aceite (assinatura) do usuário, devem ser formalizados na ferramenta oficial de tramitação de documentos. Demais documentos podem utilizar a ferramenta de gestão de demandas homologada pelo PDS-UEPG.

Registro de Demanda

As demandas podem ser originalmente registradas por um dos meios oficiais de recebimento de solicitações:

- Sistema de chamados de suporte do NTI
- Sistema Eletrônico de Informações
- Sistema de Demandas do NTI

Além dos sistemas de registro oficiais, a origem de uma demanda pode ser identificada baseando-se em análise do próprio NTI ou áreas executoras de TI e registrada por estes.

Após registrada a solicitação, deve-se realizar sua classificação se é um serviço ou uma demanda de software.

- **Serviço:** qualquer solicitação prevista no catálogo de serviços do NTI, previsto para ser executado em ações operacionais diárias (cíclicas) ou planejadas e que **não resultem em alteração de software**.
- **Demanda de software:** qualquer solicitação que demande esforço técnico para alteração de software (código-fonte) ou sua base (banco de dados) independentemente de sua complexidade, urgência ou prioridade.

Em caso de um serviço, as áreas responsáveis devem realizar seu atendimento. Em caso de demanda relacionada à gestão de produto de software, se o registro foi realizado em um sistema diferente que o de Gestão de Demandas de TI, o Coordenador de Área de Desenvolvimento deve transportar o mesmo para esta ferramenta.

O registro de demanda deve apresentar:

- Área de negócio demandante
- Cliente demandante
- Responsável pelo processo de negócio
- Área de Desenvolvimento
- Usuário que registrou a demanda
- Data e hora de registro
- Sistema de origem quando diferente do sistema de gestão de demandas

- Título e descrição da demanda
- Produto de software alvo, em caso de a demanda ser destinada a um produto de software identificável.

Após o registro da demanda no sistema de gestão, esta será colocada no estado “Aguardando classificação”, a qual deve ser realizada pelo cliente, podendo ser revisada e avaliada pelo Coordenador de Área responsável ou pela gestão do NTI.

Apenas após a classificação a demanda será analisada e pode ser planejada para execução em projeto. Tal planejamento deve ser realizado juntamente com o cliente, unidade hierarquicamente superior à área demandante ou comitê com representantes de unidades que possam estar diretamente envolvidas ou serem impactadas no contexto da demanda.

Classificação de demanda

A classificação de demanda em software quanto sua gravidade é um passo fundamental no monitoramento da qualidade dos produtos entregues, permitindo o estabelecimento de acordos de níveis de serviço, além de servir de insumo básico para a priorização das melhorias e correções.

Desta forma, são apresentadas as classificações de prioridade e gravidade associadas a solicitações de melhorias ou identificação de defeitos de software, bem como os critérios que permitam esta classificação.

Classificação da gravidade

A gravidade pode ser interpretada como a intensidade com a qual a melhoria ou nova funcionalidade influencia no processo ou a sua não existência influencia negativamente no processo. A gravidade de um defeito pode ser interpretada como a intensidade com a qual um defeito influencia negativamente na aceitação ou uso do sistema e como impacta negativamente no negócio.

Serão consideradas 04 classificações para prioridade de solicitações e defeito de software: crítica, alta, média e baixa. As classificações estão descritas a seguir, juntamente com seus respectivos critérios. Uma demanda possuirá apenas uma classificação quanto à gravidade. A tabela de classificação deve ser consultada com frequência devido a possibilidade de atualização.

Estas classificações podem não ser definitivas, vindo a serem alteradas de acordo com decisão da instituição ou do NTI.

- 1 **Crítica:** classifica-se como crítica quando pelo menos uma das situações abaixo ocorrer.
 - 1.1 Comprometimento da segurança do sistema ou do negócio, independentemente da existência de uma regra ou menção no caso de uso que se relacione com o defeito de segurança identificado;
 - 1.2 Interrupção do fluxo de trabalho do negócio devido à inexistência da funcionalidade ou existência de erro no sistema, seja este erro tratado (quando a aplicação leva para uma tela formatada para exibição de mensagem de erro), ou não tratado (quando as informações de erro são as fornecidas diretamente pela plataforma ou servidor de aplicação);
 - 1.3 Prazo externo definido por demanda de órgãos governamentais e de natureza reguladora ou regulamentadora;
 - 1.4 Impactos financeiros negativos no negócio como: pagamento de multas, perda de recursos, custo de tempo excessivo na execução de tarefas;

- 1.5 A impossibilidade de conclusão de um fluxo principal – descrição de uso principal de uso do sistema;
 - 1.6 Inoperância de integrações externas essenciais ao sistema;
 - 1.7 Divergência de regra de negócio em relação à especificação, que impossibilite o uso do sistema;
 - 1.8 Desempenho baixo com tempo de resposta que impossibilite o uso do sistema;
 - 1.9 Corrupção de um ou mais registros de dados.
 - 1.10 Validações inconsistentes;
 - 1.11 Ausência de interfaces de tela para que o usuário realize a manipulação de dados, parâmetros de sistema ou que dificultem o uso do sistema e exijam intervenção manual de nível técnico dos profissionais de TI diretamente em sistemas de bancos de dados;
- 2 **Alta:** classifica-se como alta quando pelo menos uma das situações abaixo ocorrer.
- 2.1 Tratamento incorreto dos dados fornecidos (não persistiu da forma esperada ou tratou uma entrada de forma inadequada);
 - 2.2 Retorno de dados incorretos (os dados trazidos pelo sistema foram dados diferentes do esperado);
 - 2.3 Sistema implementado em desacordo com as regras definidas nos casos de uso;
 - 2.4 Ausência de parâmetros ou tabelas de domínio que dificultem o uso do sistema ou que seja necessária intervenção manual de nível técnico dos profissionais de TI, como execução de scripts, *stored procedures*, etc.
- 3 **Média:** classifica-se como média quando pelo menos uma das situações abaixo ocorrer.
- 3.1 Comprometimento da usabilidade por parte do sistema implementado. Considera-se como usabilidade a facilidade de compreender o funcionamento do sistema bem como de utilizá-lo para realizar alguma tarefa. Alguns exemplos de situações que comprometem a usabilidade são descritos abaixo:
 - 3.1.1 Campos autocomplete que demoram mais de 3 segundos para disponibilizar algum resultado;
 - 3.1.2 Campos autocomplete que, durante o período de pesquisa, não informa ao usuário que a mesma está sendo feita;
 - 3.1.3 Campos autocomplete que não apagam a mensagem de instrução, por exemplo “pesquise aqui”, quando ganham o foco;
 - 3.1.4 Campos que perdem foco, quando não deveria;
 - 3.1.5 Campos que impossibilitam a utilização da funcionalidade “colar”;
 - 3.1.6 Campos do tipo “lista suspensa” que não se retraem quando perdem o foco;
 - 3.1.7 Campos obrigatórios que estão sem a sinalização de obrigatoriedade;
 - 3.1.8 Diferenciação de maiúsculas e minúsculas em campos de pesquisa;
 - 3.1.9 Sensibilidade a espaços em branco inseridos no início ou no fim dos valores informados nos campos de pesquisa;
 - 3.2 Comprometimento da acessibilidade do sistema, quando esta for definida como um requisito do sistema.

- 4 **Baixa:** classifica-se como baixa quando pelo menos uma das situações abaixo ocorrer:
- 4.1 Desacordo com layout definido pelo protótipo;
 - 4.2 Erros de ortografia ou de tradução para outro idioma;
 - 4.3 Formatação equivocada de saída e entrada de dados;
 - 4.4 Inconsistências de massa de dados para homologação que dificultem testes do sistema;
 - 4.5 Demais situações que não se enquadram nos critérios definidos em classificações mais altas.

Planejamento de demanda

O planejamento das demandas define sua prioridade de execução em relação às demandas registradas para uma determinada área de negócio.

A priorização da ordem de execução das demandas deve ser realizada pelo Responsável do Processo, com o apoio do Coordenador de Área de Desenvolvimento para contexto da demanda, ou Gerente de sistemas de informação no NTI.

O planejamento deve levar em consideração a classificação de gravidade definido pelo cliente, confrontando as demais demandas da área solicitante. Em caso de demandas que afetem mais de uma área de negócio, o planejamento deve confrontar as demandas de todas as áreas.

Além do confronto das gravidades das demandas, deve-se analisar as demandas em execução, em relação ao uso de recursos de TI (incluindo alocação de pessoas para a demanda) que podem ser disponibilizados sem comprometer os projetos em andamento. Em caso de necessidade de relocação de recursos, o cliente ou clientes demandantes, devem formalmente solicitar o remanejamento parcial de recursos de projetos em andamento para sua área, ou interrupção temporária de projetos. Estas ações levam obrigatoriamente ao replanejamento dos projetos afetados.

A decisão de execução das demandas deve, obrigatoriamente, resultar no planejamento de projetos para seu controle, desta maneira, garantindo que demandas de menor complexidade, tempo de execução ou prioridade, possam ser atendidas quando agrupadas.

Análise de viabilidade de requisitos

Para o planejamento das demandas é necessária a análise prévia de viabilidade, com o levantamento inicial de requisitos.

O levantamento deve ser realizado pelo Coordenador de Área de Desenvolvimento ou Gerente responsável pelo desenvolvimento, junto aos clientes conhecedores do negócio, com aprovação formal do cliente responsável pela área de negócio requisitante.

A análise de viabilidade deve prever o esforço a ser empregado para atender a demanda, de acordo com a métrica oficial de análise de esforço para o desenvolvimento de software segundo o PDS-UEPG.

O documento de requisitos iniciais deve fazer parte da documentação do projeto.

Execução e acompanhamento de demanda

O planejamento das demandas tem por objetivo priorizar e planejar sua execução, a qual se dá por meio da configuração de projetos de desenvolvimento de software.

Um projeto para execução de demanda poderá atender a uma única demanda, independentemente de sua complexidade ou extensão. Ou ainda, poderá agrupar o atendimento de uma série de demandas, desde que tenham relação contextual que permita a gestão adequada do projeto junto aos clientes responsáveis por aprovações e decisões nos fluxos de negócio.

O arranjo dos projetos deverá organizar cada demanda como uma tarefa no projeto, podendo o projeto ter datas de início e término, planejadas e realizadas, diferentes das datas planejadas e realizadas das demandas.

O acompanhamento da execução das demandas deve ser feito informando o percentual estimado do atendimento da demanda ou pelo fechamento de tarefas planejadas associadas à demanda.

O Líder do projeto é o responsável por reportar o andamento de execução das demandas de seu projeto. A frequência de atualização deve ser, no mínimo, semanal. O Coordenador de Área é responsável por acompanhar a execução e atualização das demandas alocadas à sua área, reportando semanalmente o estado dos projetos e demandas ao Gerente responsável pelo desenvolvimento.

Gestão de projeto

Como descrito nas seções que tratam as demandas, a execução efetiva de atendimento é feito pela organização das demandas formalmente em projetos de desenvolvimento de software.

Após priorizada e planejada uma demanda ou conjunto agrupado de demandas de contexto comum, elas serão alocadas como tarefas ou marcos de projeto.

O projeto deve conter a identificação do líder do projeto, equipe técnica de desenvolvimento, cliente responsável pela área de negócio, data de início e término planejadas. As datas do projeto devem ser compatíveis com as datas planejadas para as demandas que o compõe.

O esforço previsto do projeto é representado pela somatória dos esforços indicado nas demandas que o compõe.

O projeto deve ser registrado e acompanhado nos sistemas oficiais de gestão de produtos de software, demandas e projetos do NTI UEPG, descritos no PDS-UEPG.

O encerramento de um projeto deve ser aceito formalmente pelo Coordenador de Área responsável pelo desenvolvimento, o qual analisará o atendimento aos requisitos de qualidade descritos a seguir:

1. Artefatos relacionados ao Planejamento e Comunicação com o Cliente
 - 1.1. Modelagem do negócio - *opcional*
 - 1.2. Descrição dos requisitos de desenvolvimento – registrados na plataforma gestão de produtos de software descritas no PDS-UEPG - **obrigatório**.
 - a) Aceitação formal do cliente quanto aos requisitos de sistema - **obrigatório**
 - 1.3. Atas de reunião - *opcional*
2. Artefatos relacionados a Análise e Projeto de Software

- 2.1. Documentos atualizados do projeto de software criados durante a análise, projeto e implementação, tais como: modelos de dados, diagramas UML, scripts de teste automatizados, documentação de implantação
 - a) Os documentos de projeto serão analisados de acordo a sua complexidade, sendo que o Coordenador de Área ou Gerente responsável pelo desenvolvimento, poderá solicitar complementação caso necessário;
 - b) Este bloco de artefatos é **obrigatório** para a existência da documentação de projeto de software, sendo que deve ser criada e mantida com o objetivo mínimo de auxiliar a manutenção e implementação de novos módulos dos sistemas, tomando o cuidado de não se tornar burocrática.
3. Artefatos relacionados a Testes
 - 3.1. Documentação de planejamento de testes – *opcional*;
 - 3.2. Aceitação formal do cliente de que a área de negócios testou as implementações em ambiente de homologação (*stage*) e aceitou o resultado da implementação. **O sistema não deve ser publicado em produção antes da etapa de testes de aceitação do cliente, realizada no ambiente de homologação.**
4. Artefatos de código-fonte
 - 4.1. Gerenciado utilizando as ferramentas oficiais de gestão de versionamento de código fonte;
 - 4.2. Código-fonte atualizado na ferramenta de gestão de repositórios oficial da UEPG, com os códigos organizados nos ramos a seguir de forma a serem um espelho dos ambientes de servidor relacionados à direita:
 - a) **master** → servidor de produção – **obrigatório**.
 - b) **stage** → servidor de stage (testes de usuário) – **obrigatório**
 - c) **dev** → servidor de dev – **obrigatório**
 - d) **Ramos feature** → servidores de desenvolvedores – *opcional*.
A definição de versionamento usando o Fluxo “Feature Branch” será descrita nos documentos de apoio e referência ao PDS-UEPG.
5. Artefatos de documentação para o usuário
 - 5.1. Material de auxílio ao usuário para utilização do sistema / rotinas implementadas. Em caso de demanda de manutenção de funcionalidades existentes, atualização do material de auxílio ao usuário - **obrigatório**.
 - a) O material pode ser em um dos formatos a seguir, porém, registrado na ferramenta oficial de Gestão de Base de Conhecimento de TI da UEPG:
 - FAQ – Perguntas e Respostas, por funcionalidade e/ou ações no sistema – formato preferencial. Pode conter texto, imagens e vídeos na estrutura do conteúdo;
 - Manual do sistema (PDF);
 - Treinamento – referenciado em FAQ do sistema oficial de gestão de base de conhecimento.
6. Artefatos de Gestão do Projeto
 - 6.1. Relatório de encerramento do projeto – **de acordo com a complexidade do projeto.**

- a) O relatório, de responsabilidade do líder do projeto, deve especificar de maneira objetiva: sucessos e problemas no andamento do projeto, justificando ou documentando, motivos de atrasos ou replanejamentos.
- 6.2. Aprovação de encerramento pelo Coordenador de Área – **obrigatório**.
- a) Apenas após o aceite de encerramento pelo Coordenador da Área de desenvolvimento, a qual o projeto está associado, os recursos envolvidos, de acordo com sua alocação de esforço, serão liberados para comprometimento em novos projetos.
 - b) O Gerente responsável pelo desenvolvimento pode solicitar revisão dos projetos para validação e acompanhamento quanto ao atendimento da qualidade em relação aos parâmetros especificados no PDS-UEPG. Projetos que, porventura, não atendam aos requisitos de qualidade, deverão ser retomados para realização dos ajustes necessários.

Subprocesso de Análise e Projeto de Software

O PDS-UEPG não especifica detalhes técnicos quanto aos subprocessos - salvo exceções pontuais, e as etapas conhecidas nas disciplinas de análise e projeto (projeto no sentido de projetar o software e não de planejar as atividades de um projeto). Porém, especifica etapas, ferramentas homologadas e artefatos obrigatórios.

Artefatos

Todo projeto por natureza gera são definidos um conjunto de artefatos, resultantes como subprodutos, inclusive os projetos de análise e projeto de software. Tais artefatos, em geral, são compilados no formato de documentos, que nesta norma são classificados por meio dos conceitos a seguir:

- **Documentação técnica do produto de software - DTSP:** como análises, modelagens, requisitos;
- **Documentações da gestão de projeto - DTPRJ:** como aceites de clientes, formalizações de abertura e encerramento de projetos;
- **Documentações do usuário - DU:** como manuais de uso do sistema, ajuda, bases de conhecimento.

A classificação supra definida abrange, além dos artefatos em formato clássico conhecido como “documento”, informações em formatos diversos de acordo com a ferramenta onde está sendo armazenada e gerenciada, como FAQ, *issue*, base de conhecimento, dentre outras.

Artefatos de Análise e Projeto:

- *Visão Geral do Projeto:*
 - Desejáveis:
 - Documento com a Visão geral do negócio ou Regras de negócio (DTSP)
 - Glossário técnico e de negócio (DTSP)
- *Gestão dos requisitos:*
 - Obrigatórios:
 - Requisitos (DTSP): documento ou organização dos requisitos trabalhados no projeto.
 - Para a gestão pelas equipes de TI, de requisitos a serem implementados, deve ser utilizado o formato de *issues* no sistema de gestão de projetos de software.
 - Aceite de requisitos (DTPRJ): documento que especifique os requisitos a serem desenvolvidos em um projeto e contenha o aceite do(s) cliente(s) principal(is).
 - Deve ser uma versão que reflita os requisitos inseridos como *issues* no sistema de gestão de projetos de software.
 - Assinado no SEI.
 - Desejados:
 - (DTSP) diagramas e especificações de Casos de Uso ou Histórias de Usuários e Épicas – SCRUM.
- *Planejamento lógico e físico:*
 - Obrigatórios:

- Descrição dos dados (DTPS): ou em um modelo de dados atualizado, ou em migrações do framework, ou em dicionário de dados.
- Especificação do banco de dados (DTPS): documento que descreva qual o SGBD em uso e os bancos de dados. Os bancos de dados podem ser especificados na descrição de dados.
- Desejáveis:
 - diagramas UML - DTPRJ ou DTPS, dependendo do objetivo do diagrama, se para análise de entendimento, ou se para documentação do produto; Dicionários de Dados (DTPS).
- *Planejamento técnico*:
 - Obrigatórios:
 - Orientações de implantação de ambientes e particularidades de publicação (*deploy*) em produção e demais ambientes (DTPS ou DTPRJ).

Gestão e armazenamento das informações

Os documentos técnicos de produto de software (**DTPS**) – ou a organização das informações referentes a estes documentos, devem ser armazenados e gerenciados na ferramenta de gestão de projetos de software especificada neste PDS. Esta organização pode seguir os formatos da ferramenta, como: *issues*, marcos de projetos (*milestones*), tarefas, histórias épicas, ou outro conceito que a mesma utilize.

Os documentos de usuário (**DU**) – ou a organização das informações referentes a estes documentos, devem ser armazenados e gerenciados na ferramenta de gestão de base de conhecimento especificada neste PDS. Esta organização pode seguir os formatos da ferramenta, como: FAQ, artigo, vídeo, ou outro conceito que a mesma utilize.

Os conjuntos de informações, que devem ser assinados por clientes e equipes de TI, referentes a gestão dos projetos (**DTPRJ**), como documentos de requisitos com aceite, documentos de aceite de testes executados, documentos de aceite de entregas em produção, documentos de formalização de etapas do projeto, devem ser gerenciados como processos administrativos, armazenados e assinados no SEI, ferramenta oficial de tramitação de documentos eletrônicos da instituição, ou outro que venha a substituí-lo.

O gerenciamento dos documentos de DTPS, DTPRJ e DU podem ser orquestrados por ferramentas de suporte ao PDS-UEPG identificadas nas especificações técnicas desta norma.

Subprocesso de Desenvolvimento de Software

As atividades inerentes ao subprocesso de desenvolvimento de software, como a codificação do software em si, scripts e rotinas auxiliares como testes automatizados, migrações de bancos de dados, scripts de manutenção; devem ser organizados e gerenciados pelos coordenadores de área dos produtos de software.

O PDS-UEPG especifica detalhes técnicos quanto a este subprocesso, principalmente na seção de Especificações Técnicas, que pode definir padronizações para linguagens de programação, frameworks, bibliotecas de código, pacotes/*addons*/plugins; etc.

Obrigações no processo de desenvolvimento envolvem o versionamento de código fonte, utilizando o sistema Git e o sistema de gerenciamento de projetos de software especificado na seção de Especificações Técnicas.

É necessário utilizar um documento de padronização de código, podendo ser definido por projeto, sendo que o mesmo deve ser especificado na documentação do mesmo.

O item desejável a ser evoluído e amadurecido é o da cultura / política de testes automatizados. Nesta versão do PDS a escrita de testes é uma orientação como “desejável”.

Especificações Técnicas e Ferramentas Oficiais da UEPG

Esta seção define o detalhamento técnico e tecnológico a ser utilizado no desenvolvimento de sistemas no âmbito da UEPG, sob a égide do PDS-UEPG. Fornece as especificações conceituais e técnicas mínimas necessárias às quais os produtos ou serviços executados pelas áreas de TI da instituição, ou ofertados por terceiros, deverão obrigatoriamente atender.

Também define o conjunto de ferramentas oficiais e aquelas homologadas para a gestão de informações relacionadas a processos de negócio da instituição e aos processos de gestão do NTI e demais unidades a executar serviços de TI da instituição.

Softwares de apoio aos processos de TI

Definição dos sistemas de terceiros, oficiais, homologados e mantidos pelo NTI; no uso de processos da instituição e internos das áreas de TI.

Atenção: para as integrações com estes softwares e ferramentas, quando necessário, devem seguir as orientações oficiais de interfaceamento descritos neste documento, ou, quando não definido aqui, utilizar pacotes de programação já estipulados pela equipe e orientados pelos Coordenadores de Área ou Gerente responsável pelo desenvolvimento.

Gestão de Produto de Software e de Gerenciamento de Projeto de Software (técnico)

Gitlab para o gerenciamento de código-fonte, versionamento, CI/CD, gestão de projetos de desenvolvimento e produtos de software. Repositório oficial de códigos-fonte dos sistemas desenvolvidos no âmbito da UEPG.

- Acesso: <https://git.uepg.br>

Gestão de Atendimento ao Cliente

Sistema OTRS para gestão de tickets de suporte para solicitação de serviços e demandas de TI.

- Acesso público: <https://suporte.uepg.br>
- Acesso agentes de suporte: <https://suporte.uepg.br/otrs>

Gestão de Base de Conhecimento

Sistema OTRS, módulo de FAQ – Frequent Asked Questions (Base de conhecimento no formato Perguntas Frequentes)

- Acesso público: <https://suporte.uepg.br/otrs/public.pl>
- Acesso agentes de suporte: <https://suporte.uepg.br/otrs/index.pl?Action=AgentFAQExplorer>

SEI - Sistema Eletrônico de Informações

Sistema Eletrônico de Informações (SEI), desenvolvido pelo Tribunal Regional Federal da 4ª Região (TRF4), é uma plataforma que engloba um conjunto de módulos e funcionalidades que promovem a eficiência administrativa. Possui regulamentação própria que define o processo de tramitação eletrônica de documentos e da ferramenta no âmbito da UEPG.

No portal SEI UEPG, além do acesso à ferramenta, o regimento, manuais e cursos são encontrados.

- Portal SEI UEPG: <https://sei.uepg.br>
- Sistema SEI: <https://sei.uepg.br/sei>

Arquitetura e tecnologias de referência para o desenvolvimento de sistemas

Esta seção apresenta o conjunto de tecnologias e linguagens de programação utilizados nas aplicações desenvolvidas e mantidas no NTI e unidades executoras de TI da UEPG.

Tecnologias padronizadas

Tecnologias, frameworks, ferramentas e linguagens de programação padronizadas e homologadas para a construção de novos sistemas, bem como, novas funcionalidades e módulos de sistemas existentes:

- 1 A arquitetura **web** deve ser considerada como a principal opção de escolha na construção de softwares na instituição.
 - 1.1 Os sistemas devem ser testados e garantir a execução nos navegadores Firefox, Chrome e Chromium, considerando os sistemas operacionais Windows, Linux e Mac OS.
 - 1.2 Quando o sistema não for, explicitamente, compatível com algum dos navegadores em seu respectivo contexto de publicação (intranet ou internet), a tentativa de acesso feita por um navegador incompatível deve gerar uma mensagem para usuário informando quais são os navegadores recomendados.
- 2 A linguagem **PHP** versão 7.3, ou superior, determina a base para o desenvolvimento de sistemas na instituição, porém não deve ser usada com arquitetura definida isoladamente por aplicação, isto significa, não usar PHP “puro” – *plain PHP*, sem framework.
 - 2.1 <https://www.php.net>
 - 2.1.1 Manual em Português: https://www.php.net/manual/pt_BR/
- 3 O framework **Laravel**, versão 8.0 ou superior, é o padrão para construção de sistemas na linguagem PHP para web.
 - 3.1 <https://laravel.com>
 - 3.2 Para o desenvolvimento em ambiente web deve ser considerado a “renderização” de telas clássica no lado servidor – *server rendering*, como arquitetura padrão para construção da interface gráfica do usuário.

- 4 Autenticação e gestão de usuários e níveis de permissão nos sistemas Laravel devem utilizar o pacote **SGI Authorizer**, o qual se conecta de maneira transparente com a base de gestão de usuários oficial da instituição **SGI**. A documentação do pacote está disponível na seção Wiki do repositório Gitlab oficial.
 - 4.1 <https://sgi.apps.uepg.br>
 - 4.2 <https://git.uepg.br/sgi/sgi-authorizer>
 - 4.3 A gestão de usuários, banco de dados, ldap, sistema de gestão, deve obrigatoriamente seguir as definições do SGI, não sendo permitida a criação de módulos isolados criando ilhas de dados de usuários em sistemas.
- 5 Considerar a utilização de **Javascript**, linguagem interpretada no navegador com o intuito de melhorar a experiência de utilização do usuário, como componentes autocompletar, carregamento cruzado de dados baseado em seleção.
 - 5.1 <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>
 - 5.2 Deve ser evitada a construção de aplicações com a arquitetura SPA – Single Page Applications, que demandam a implementação de clientes “gordos” com alta complexidade de implementação e uso de frameworks Javascript dedicados a este fim. Casos devidamente justificados deverão ser aprovados em reunião de Arquitetura e/ou análise de Especificação de Dados juntamente à Gerência responsável pelo desenvolvimento e gestão de dados.
- 6 Para facilitar o uso de funções padrões em Javascript nos navegadores, o uso da biblioteca **JQuery** é aprovado, desde que seguindo a regra de construção de componentes e não utilizada para uma arquitetura SPA.
 - 6.1 <https://jquery.com>
- 7 O layout das aplicações web devem utilizar como base o framework CSS **Twitter Bootstrap** para padronização de telas HTML. Pode-se considerar a utilização de AdminLTE – template do tipo “admin” construído em Twitter Bootstrap
 - 7.1 <https://getbootstrap.com>
 - 7.2 <https://adminlte.io>
- 8 Para a construção de testes unitários utilizar a biblioteca **PHPUnit**, já devidamente integrada ao framework Laravel, ou **PestPHP**.
 - 8.1 PHPUnit: <https://phpunit.de>
 - 8.2 PestPHP: <https://pestphp.com>
- 9 A utilização do SGBD **PostgreSQL** é obrigatória para novos aplicativos, exceto em casos justificados em reunião de Arquitetura e/ou análise de Especificação de Dados juntamente à Gerência responsável pelo desenvolvimento e gestão de dados.
 - 9.1 <https://www.postgresql.org>
- 10 **Jasper Reports Studio e Servidor** na construção de relatórios para os sistemas – o uso do servidor Jasper Server para publicação de relatórios, deve ser solicitado à Gerência de Infraestrutura de TI.
 - 10.1 <https://community.jaspersoft.com/project/jaspersoft-studio>
 - 10.2 <https://community.jaspersoft.com/project/jasperreports-server>

11 Apache eCharts – biblioteca Javascript para a construção de gráficos para visualização de dados

11.1 <https://github.com/apache/echarts>.

12 Para a construção de “sites” institucionais, utilizar o CMS **WordPress**, utilizando o ambiente Multisite, com tema e plugins padronizados e atualizações mantidas pelo NTI. Na UEPG o ambiente multisite padrão e oficial para novos sites é chamado de plataforma W2. A modalidade de “hosting” de sites não é contemplada, sendo que, qualquer site institucional deve ser criado ou migrado para esta plataforma. O tema padrão utilizado na plataforma W2, e no projeto do Portal UEPG, chama-se Materialis, sendo que para este a instituição possui licença da versão “PRO”.

12.1 <https://br.wordpress.org>

12.2 <https://www2.uepg.br>

12.3 <https://extendthemes.com/materialis>

13 Para contextos de desenvolvimento de **aplicativos móveis**, o padrão a utilizar é linguagem Dart com o framework **Flutter**. O desenvolvimento de aplicativos móveis somente será autorizado **após análise e decisão estratégica de gestão do NTI**.

13.1 <https://dart.dev>

13.2 <https://flutter.dev>

Ferramentas e ambientes sugeridos para o desenvolvimento

Um conjunto de ferramentas e ambientes são sugeridos no auxílio para a produtividade, por terem sido testados e amplamente utilizados nos ambientes de desenvolvimento de sistemas na UEPG em conjunto com a arquitetura padrão definida no PDS-UEPG.

1 O IDE **PHPStorm**, que proporciona alta produtividade para o desenvolvimento em linguagem PHP, bem como com o framework Laravel.

1.1 <https://www.jetbrains.com/phpstorm>

2 **Dbeaver** é um cliente para gerenciamento de bancos de dados com muitos recursos para acesso e manipulação de diversos SGBDs, sendo multi-plataforma.

2.1 <https://dbeaver.io>

3 Para testes, modelagem e documentação de APIs, utilizar o aplicativo **Postman**, o qual oferece uma versão gratuita.

3.1 <https://www.postman.com>

Pacotes homologados para Laravel

Lista de pacotes homologados para uso nos contextos identificados.

Em caso de necessidade de utilização de um novo pacote não pertencente a esta lista, solicitar análise e homologação para a Gerência de Sistemas de Informação.

Visitar a Wiki do projeto NTI → <https://git.uepg.br/nti/nti/-/wikis/laravel-pacotes>

Regras para modificação de arquitetura

Apresentar um documento detalhado de requisitos, prospecção de tecnologias e conjunto de justificativas com avaliação custo x benefício para a construção de uma solução em arquitetura ou tecnologia diferente dos padrões UEPG.

Necessário incluir custo x benefício para a manutenção de médio/longo prazo caso a modificação seja aceita.

Tecnologias legado

Tecnologias legado representam aquelas nas quais sistemas ativos em ambiente de produção, porém escritos em tecnologias e linguagens de programação em uma das situações de obsolescência em relação às decisões de padronização de tecnologia do NTI:

- Descontinuados pelas empresas ou instituições mantenedoras/fornecedoras;
- Em desuso e sem manutenção continuada pelas comunidades de desenvolvimento Open Source;
- Ferramentas, tecnologias, frameworks ou linguagens de programação definidos como em desacordo com as padronizações oficiais definidas pelo PDS-UEPG;
- Arquiteturas de sistemas ou frameworks definidos isoladamente para um ou um conjunto de sistemas em desacordo com as padronizações oficiais definidas pelo PDS-UEPG;
- Tecnologias que, apesar de atuais no mercado de trabalho, possuem “ilhas de conhecimento” na instituição, com equipes pequenas, gerando riscos e altos custos para a continuidade e expansão de aplicações implementados as utilizando.

Apesar de se enquadrarem em uma das classificações citadas, os sistemas existentes devem ser mantidos até que sua reescrita na arquitetura padronizada seja possível.

Porém, **novos sistemas** e, quando possível, **melhorias** ou novas funcionalidades, devem ser implementadas já nas **definições de padronização do PDS-UEPG**, com o objetivo de minimizar o passivo tecnológico que tecnologias legadas geram.

São tecnologias legado atualmente suportadas:

- 1 **Centura/SQL Windows:** linguagem e ambiente de programação para sistemas em plataforma *desktop*.
- 2 **PHP “puro”:** ou *Plain PHP*, apesar de a linguagem PHP ser apresentada a neste documento como a base da arquitetura atual para o desenvolvimento de sistemas, diversas aplicações foram implementadas com arquitetura particular, criando estruturas de sistemas totalmente individualizadas, dificultando a manutenção e expansão destes sistemas. Desta maneira, o desenvolvimento de softwares usando esta linguagem, deve ser realizado por meio de arquitetura de software ou *frameworks* especificados nesta norma.
- 3 **Grails:** *framework* de desenvolvimento construído na linguagem Groovy que, apesar de atual em relação a conceitos e comunidade de desenvolvimento, possui na instituição equipe pequena com o conhecimento para sua manutenção.
- 4 **Delphi:** ambiente e *framework* de desenvolvimento na linguagem Object Pascal que, apesar de sua ampla utilização no mercado de trabalho, possui uma equipe pequena com o conhecimento para sua manutenção.

- 5 **Sybase:** sistema gerenciador de banco de dados, o qual atualmente suporta boa parte das aplicações da instituição. Embora possua suporte e atualizações periódicas, é considerado como tecnologia legado por apresentar:
- 5.1 Alto custo financeiro em contratos de manutenção – existem alternativas *Open Source* com custo inferior e igual ou maior capacidade de recursos. Ainda, o custo de licenciamento, que é contabilizado por processador, limita a expansão/escalabilidade dos sistemas;
 - 5.2 Indisponibilidades de estruturas técnicas atualmente comuns no mercado de bancos de dados, tais como: suporte a consultas particionadas, estruturas de árvores hierárquicas, dados espaciais sem custo adicional, dentre outros;
 - 5.3 Dificuldade de configuração e manutenção de drivers de conectividade com as linguagens de programação utilizadas e padronizadas no PDS-UEPG.

Documentação de sistemas

Para o desenvolvimento de um novo sistema, ou a manutenção sob demanda de sistemas existentes mantidos pelo NTI, ou para fins de documentação de sistemas desenvolvidos fora do âmbito do NTI, este poderá requerer a execução do serviço de documentação, com a produção dos artefatos previstos nos documentos que regem o PDS-UEPG, referentes às fases de requisito, projeto de sistemas, modelagem de dados, além da realização ou atualização da contagem detalhada do sistema em pontos de função.

Todos os artefatos a serem produzidos durante a execução do serviço de documentação deverão estar de acordo com o PDS-UEPG vigente ou deverão ser previamente submetidos para sua aprovação.

Caso a equipe gestora e executora de projeto de desenvolvimento de software, sendo ela CONTRATADA ou INTERNA, que identifique que não será necessária a atuação em algumas das fases de documentação do projeto (requisito, projeto de sistemas, modelagem de dados), em virtude do escopo e natureza da demanda, o esforço de análise da solicitação não será remunerado – em caso de contratação, ou não será contabilizado como esforço em caso de equipe interna. Desta forma, apenas as fases efetivamente realizadas e com aceite definitivo serão objeto de faturamento ou contabilização de esforço em projeto.

Artefatos previstos para serviços de documentação

- 1 Fase de Requisitos:
 - 1.1 Diagrama de Caso de Uso;
 - 1.2 Especificação de Caso de Uso ou História do Usuário, a critério;
 - 1.3 Especificação Suplementar;
 - 1.4 Documento de Regras de Negócio;
 - 1.5 Glossário de Negócio;
 - 1.6 Protótipo de Interface do Usuário;
 - 1.7 Atas de reunião;
 - 1.8 Documento de Mensagens do Sistema;
 - 1.9 Planilha de Contagem Detalhada;

- 2 Fase de Modelagem de Dados
 - 2.1 Dicionário de Dados;
 - 2.2 Modelo de Dados;
 - 2.3 Scripts DDL e DML;
 - 2.4 Código fonte de *Migrations*
- 3 Fase de Projeto de Sistema:
 - 3.1 Documento de Arquitetura de Projeto;
 - 3.2 Diagrama de Classe – a critério do NTI poderá ser dispensado;
 - 3.3 Diagrama de Sequência – a critério do NTI poderá ser dispensado;
 - 3.4 Realização de Caso de Uso – a critério do NTI poderá ser dispensado;
- 4 Medição em ponto de função;
 - 4.1 Planilha de Contagem Detalhada;
- 5 Excepcionalmente, a depender do projeto relacionado, outros artefatos poderão ser previstos pelo NTI na definição de exigências de Documentação do projeto.
- 6 Em caso de prestação dos serviços por terceiros, será executada mediante emissão de Ordens de Serviço de Documentação (OS de Documentação) pelo NTI;
 - 6.1 A critério do NTI, o serviço de Documentação poderá ser solicitado como escopo de uma OS de Desenvolvimento de um novo projeto ou release de software existente.
- 7 Métricas de serviço:
 - 7.1 O dimensionamento da OS de Documentação será feito por meio da técnica de pontos de função;
 - 7.2 Para fins de definição inicial da quantidade de pontos de função relativa a cada OS, o NTI se baseará nas seguintes premissas:
 - 7.2.1 Na hipótese de já haver a contagem estimada ou detalhada em Pontos de Função para o escopo do sistema/projeto sobre o qual o serviço de documentação será executado, a OS será aberta de acordo com a contagem apurada;
 - 7.2.2 Na hipótese de não haver contagem estimada ou detalhada em Pontos de Função para o escopo do sistema/projeto sobre o qual o serviço de documentação será executado, a OS de Documentação será aberta com valor correspondente a 10 (dez) pontos de função para fazer jus ao levantamento inicial;
 - 7.2.3 Ao final da produção de todos os artefatos requeridos, o valor da OS será ajustado levando-se em consideração a Planilha de Contagem Detalhada produzida como artefato e/ou o esforço relativo às atividades não mensuráveis em ponto de função.
- 8 Metodologia de execução dos serviços:
 - 8.1 O NTI abrirá a OS de Documentação com informações sobre a demanda a ser realizada e o escopo do serviço;
 - 8.2 A seu critério, o NTI encaminhará, junto a OS de Documentação, documentos que julgue relevantes para a execução dos serviços;

- 8.3 O NTI definirá na OS de Documentação o prazo máximo admitido para a total execução dos serviços;
- 8.3.1 O DESENVOLVEDOR (interno ou contratado) terá o prazo máximo de 02 (dois) dias úteis, contados do recebimento da OS, para apresentar, caso julgue necessário, pedido formal de revisão do prazo de execução indicado na OS, com exposição de fundamentos técnicos. Em tal hipótese, incumbirá ao NTI a decisão final sobre a ampliação do prazo ou não;
- 8.3.2 O prazo de execução poderá ser alterado por acordo entre as partes, observando-se a complexidade e urgência dos serviços demandados e a disponibilidade de tempo da área requisitante do NTI (dono do produto);
- 8.3.3 O cronograma para execução de cada OS deverá ser enviado pelo DESENVOLVEDOR, conforme prazos e critérios definidos;
- 8.4 Ao término do atendimento da OS de Documentação pelo DESENVOLVEDOR, os artefatos produzidos deverão ser entregues ao NTI, para fins de recebimento provisório, definitivo e/ou demais providências;
- 8.5 Os artefatos produzidos serão objeto de avaliação e aprovação pela equipe do NTI, acompanhada ou não de terceiros;
- 8.5.1 Os produtos entregues serão validados conforme os Níveis Mínimos de Serviço acordados;
- 8.5.2 A critério do NTI, a validação dos produtos entregues poderá ser realizada em conjunto com empresas especialistas por ele designadas;
- 8.6 O DESENVOLVEDOR deverá efetuar as alterações que se fizerem necessárias nos artefatos produzidos a partir dos apontamentos da equipe do NTI, em prazo por este definido, de acordo com a complexidade da demanda.
- 8.6.1 Todas as correções deverão ser feitas sem ônus para o NTI.
- 8.7 Após correções de eventuais não conformidades pelo DESENVOLVEDOR, os artefatos deverão ser submetidos à nova avaliação do NTI até que seja dado o aceite definitivo da OS de Documentação.
- 8.8 A eventual suspensão da OS de Documentação, solicitada pelo NTI, implicará a remuneração proporcional aos produtos entregues e aceitos até o momento da solicitação, desde que permaneça suspensa por período superior a 60 (sessenta) dias, ocasionando, neste caso, o fechamento da OS.