

Unidade 04

Conhecendo outros processos e metodologias/*frameworks*.

Introdução da Unidade

Olá, querido aluno! Como vai? Nessa unidade vamos expandir nossos conhecimentos sobre o processo de desenvolvimento de *software*, vamos buscar conhecer outros modelos de processo, guias de práticas e conhecer como tudo isso pode agregar no processo de desenvolvimento.

Na aula 01, iremos detalhar e introduzir o MPS.BR e seus conceitos em conjunto com o CMMI (*Capability Maturity Model Integration*), para que dessa forma seja possível observar como ocorre esses processos e até instigar uma possível comparação entre eles.

Na aula 02, vamos compreender outras guias e práticas que podem auxiliar no processo de desenvolvimento como o ITIL e o *Design Thinking*, além de visualizar o quanto isso pode elevar o nível do *software* durante seu desenvolvimento.

Objetivos

- Compreender o funcionamento do MPS.BR;
- Compreender o funcionamento do ITIL e do *Design Thinking*.

Conteúdo programático

Aula 01 – Conhecendo o MPS.BR e o CMMI.

Aula 02 – Aprendendo sobre o ITIL e Design Thinking.



Você poderá, também, **assistir as videoaulas** em seu celular, basta apontar a câmera para os **QR Codes** distribuídos neste conteúdo.

Referências

ISD BRASIL. **O que é CMMI?**. Disponível em: <<http://www.isdbrasil.com.br/o-que-e-cmmi.php>>. Acesso em: 5 mar. 2021.

NASCIMENTO, Lorryne. **Design thinking**: veja como ele pode ajudar na melhoria da experiência do usuário. Access. Run, 2020. Disponível em: <<https://www.access.run/2020/07/design-thinking-2/>>. Acesso em: 5 mar. 2021.

SOFTEX. **MPS.BR - Melhoria de Processo do Software Brasileiro**. 2012. Disponível em: <https://www.softex.br/wp-content/uploads/2013/07/MPS.BR_Guia_Geral_Software_2012-c-ISBN-1.pdf>. Acesso em: 5 mar. 2021.

SOFTEX. **MPS.BR - Melhoria de Processo do Software Brasileiro**. 2006. v.1.1. Disponível em: <http://www.softex.br/wp-content/uploads/2013/10/apresenta_o_sbqs_2006_mps11.pdf>. Acesso em: 5 mar. 2021.

Aula 01 - Conhecendo o MPS.BR e o CMMI

Olá aluno! Hoje vamos iniciar uma nova aula, iremos conversar e conhecer melhor o MPS.BR (Melhoria de processo de *software* Brasileiro) e sua aplicação, também vamos conversar sobre o CMMI (*Capability Maturity Model Integration*).

O MPS.BR é um programa criado em dezembro de 2003 e seu projeto é coordenado pela SOFTEX (Associação para Promoção da Excelência do *Software* Brasileiro), o seu foco é gerar competitividade entre as organizações por intermédio da melhora dos processos, tendo sua atenção maior em micro, pequenas e médias empresas.

Com isso, vale ressaltar que o MPS.BR é um programa e não um projeto, este programa tem seus objetivos pautados nos modelos MPS (*Master Production Schedule*) em português “Plano Mestre de Produção”, que são compatíveis com os padrões internacionais de qualidade. O MPS.BR tem duas metas que podem ser obtidas no médio e no longo prazo, são elas:

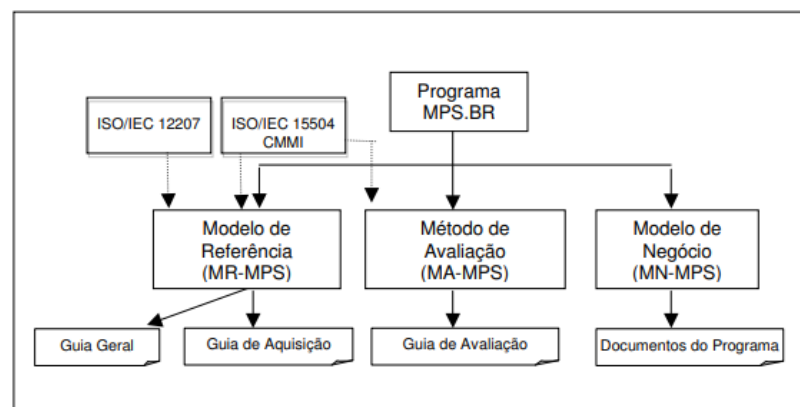
→ Técnica

Proporciona o aprimoramento do modelo MPS com seus resultados esperados, formação de consultores, instrutores de cursos, avaliadores, implementadores e avaliadores nos modelos de maturidade do MPS.

→ Negócio

Busca disseminar e viabilizar a adoção de modelos de maturidade MPS para a melhoria da competitividade entre as empresas. Com isso, o modelo MPS descreve como é separado e onde estão localizados os modelos de referências, modelos de negócio e os métodos de avaliação.

Figura 01 - Componentes do MPS



Fonte: SOFTEX (2006)

Conforme descreve a Figura 01, o modelo geral MPS se divide em três pilares sendo eles: modelos de referência, modelo de negócio e método de avaliação. Na parte superior encontramos os documentos internacionais que são utilizados para alcançar a qualidade em cada um dos pilares.

Ou seja, possuímos três grandes componentes:

- Modelo de Referência (MR-MPS);

- b) Método de Avaliação (MA-MPS);
- c) Modelo de Negócio (MN-MPS).



VIDEOAULA 01

Utilize o QRcode para assistir!

Assista o vídeo para entender melhor o assunto.



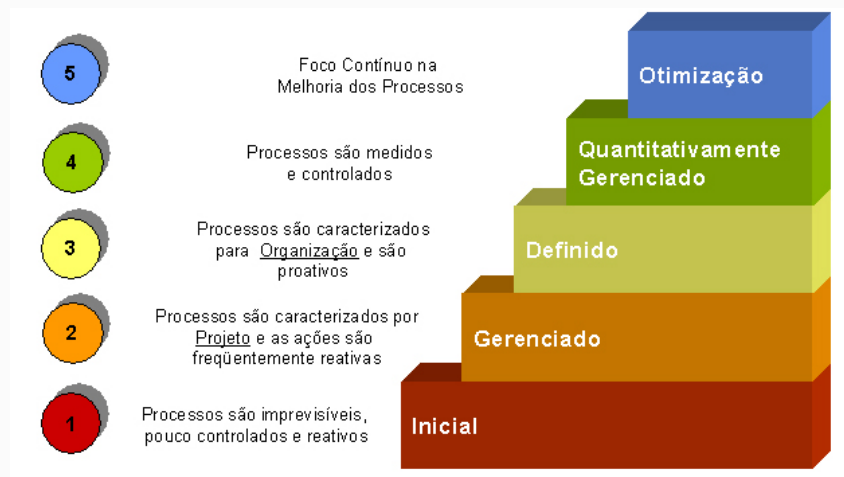
Os modelos de referência estão divididos em três subgrupos os relacionados a *software* (MR-MPS-SW), serviços (MR-MPS-SV) e recursos humanos (MR-MPS-RH). Os MR-MPS definem diversos níveis de maturidade que são uma combinação entre processos e sua capacidade, com isso é possível avaliar efetivamente a execução dos processos.

Os níveis de maturidade definem pontos para a evolução de processos, assim com estágios de melhoria da implementação de processos dentro das organizações. Este nível de maturidade está ligado diretamente ao funcionamento dos processos dentro das organizações.

Como podemos observar na Figura 01, no MR-MPS o CMMI está ligado a ele, para isso vamos falar sobre os níveis de maturidade no CMMI-DEV que é o parâmetro utilizado para a produção de produtos, são cinco níveis, sendo eles:

- 1) Inicial: os processos estão envoltos em um caos decorrente e não existe obediência a regras de processos e nem existem padrões para a produção de um produto no caso de um *software*;
- 2) Gerenciado: os projetos têm os requisitos gerenciados nesse ponto, existe nesse nível o planejamento e controle de diferentes processos para o desenvolvimento do produto;
- 3) Definido: os processos de desenvolvimento estão bem definidos e claros entre todos os participantes, existe um padrão e já é possível aplicar esse modelo em diversos ambientes;
- 4) Gerenciado quantitativamente: quando se torna previsível o desempenho dos processos, uma vez que são controlados quantitativamente;
- 5) Otimizado: quando existe uma melhoria contínua dos processos para entregar um produto.

Figura 02 - Níveis do CMMI



Fonte: ISD Brasil (2021)

Na Figura 02, podemos observar os níveis do CMMI com uma estrutura de escalada, em que o objetivo é atingir o topo. Com isso o CMMI é recomendado para ser utilizado em grandes empresas de desenvolvimento de *software*, para assim, conseguir efetivar o desenvolvimento do *software*.



VIDEOAULA 02

Utilize o QRcode para assistir!

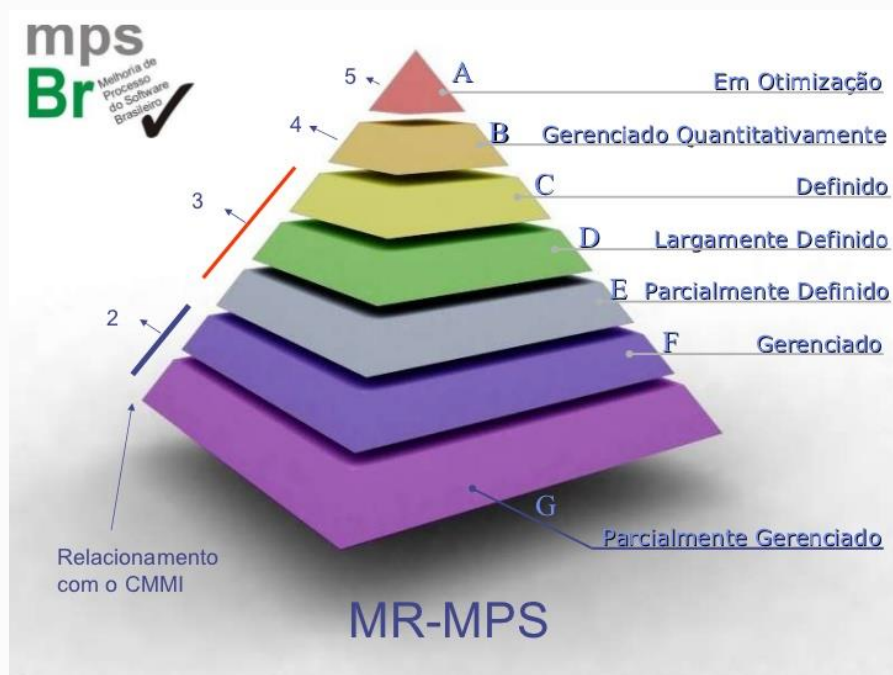
Assista à videoaula para ampliar seus conhecimentos sobre o assunto.



Mas, como o foco do MPS.BR são micro, pequenas e médias empresas os níveis de maturidade dentro do MPS são diferentes, são sete níveis, sendo eles:

- A) Em Otimização;
- B) Gerenciado Quantitativamente;
- C) Definido;
- D) Largamente Definido;
- E) Parcialmente Definido;
- F) Gerenciado;
- G) Parcialmente Gerenciado.

Figura 03 - Níveis do MR-MPS



Fonte: disponível em:

<http://3.bp.blogspot.com/_AiYEhm6AHGI/TJk3nF77Hal/AAAAAAAAAADY/fs4AK5PbAWg/s1600/mps-br.jpg>. Acesso em: 10 maio 2021.

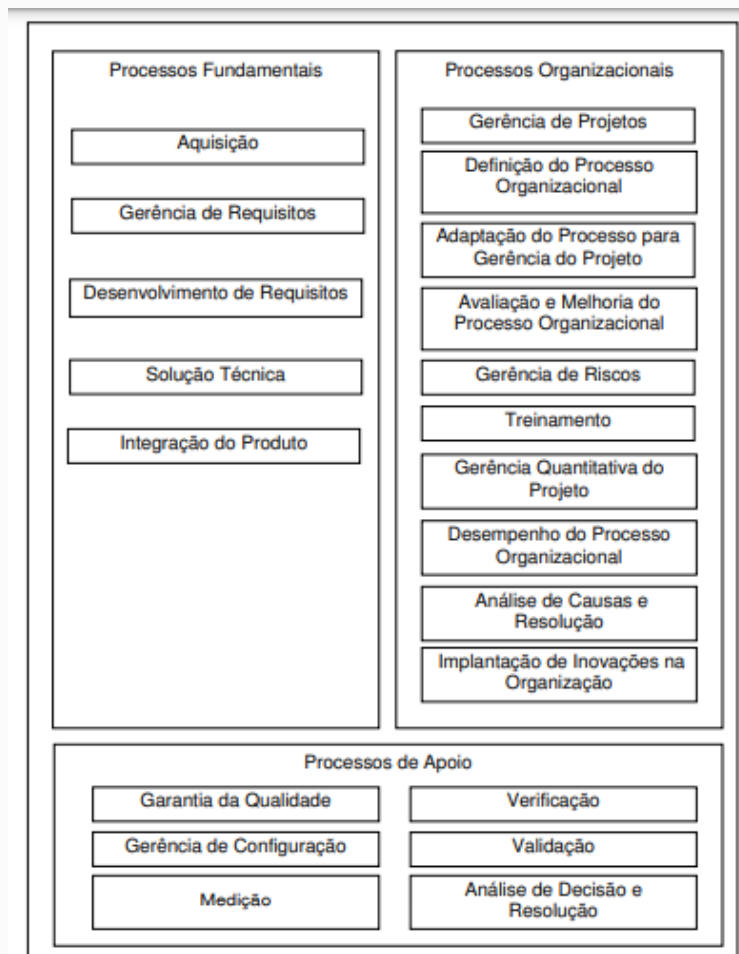
A Figura 03, nos mostra os níveis de maturidade do MR-MPS, sendo que no CMMI o objetivo é alcançar o nível A que apresenta a otimização, pois é nesse nível que o processo passa por uma melhoria contínua.

Para cada um desses níveis é agregado um perfil de processos que auxilia a organização a fim de saber onde dedicar esforço para melhorar. Mesmo sendo baseado no CMMI, o MR-MPS tem uma granulação maior para possibilitar a implementação em organizações menores, tornando visível mais rapidamente as escalas dos processos.

Os processos no MR-MPS são vistos como propósitos, resultados e informações adicionais. Esses processos são agrupados por três grupos de acordo com a sua natureza, sendo eles:

- Processos fundamentais: atendem o início do desenvolvimento, realizam a manutenção e suporte para os serviços durante o ciclo de vida do *software*;
- Processos de apoio: são aqueles que auxiliam um, e outros processos, além de contribuem para a qualidade do *software*;
- Processos organizacionais: são aqueles que existem dentro de um nível corporativo para estabelecer, implementar e melhorar o processo do ciclo de vida de um *software*.

Figura 04 - Grupos de processos do MR-MPS



Fonte: SOFTEX (2006)

A Figura 04 demonstra os grupos com seus processos que precisam ser realizados dentro de cada um, lembrando que cada processo pode estar em diferentes níveis de maturidade para alcançar o próximo nível.

LEITURA OPCIONAL

Para ampliar seus conhecimentos sobre assunto leia o **Capítulo 8.2 do “Guia Geral do MPS-BR”**: Disponível em:

<https://homepages.dcc.ufmg.br/~rodolfo/GPS1-Turma11/MPS.BR_Guia%5b1%5d.pdf>.

Acesso em: 10 maio 2021.

O método de avaliação é definido em conformidade com os modelos de referência, de avaliação e de processos estabelecidos na norma ISO/IEC 15504-2, o foco dessa norma é estabelecer a qualidade de um *software*, com isso avaliando seus processos, etapas e modo de desenvolvimento.



VIDEOAULA 03

Utilize o QRcode para assistir!

Assista à videoaula que abordará mais sobre o tema proposto.



Dessa forma, o processo de avaliação é composto por quatro “subprocessos”, sendo eles:

- a) Contratar a avaliação;
- b) Preparar a realização da avaliação;
- c) Realizar a avaliação;
- d) Documentar os resultados da avaliação.

O método de avaliação tem como propostas tornar-se de modo contínuo o processo avaliativo, assim, tornando cada vez mais assertivo o desempenho no desenvolvimento do *software*.

Já o modelo de negócio ou MN-MPS é um conjunto de regras de negócio para implementação do MR-MPS pelas organizações responsáveis por implantar e avaliar seguindo o MA-MPS, organizando as instituições para conseguirem desenvolver as práticas propostas no MPS.BR.

Caro aluno, todos os guias e práticas podem ser encontradas no site oficial da SOFTEX: <https://softex.br/mpsbr/guias/>.

Aula 02 - Aprendendo sobre o ITIL e Design Thinking

Olá aluno, tudo bem? Estamos aqui para o nosso último encontro nesta unidade, hoje vamos conversar um pouco sobre o ITIL e o *Design Thinking*, e como eles podem auxiliar no processo de desenvolvimento de *software*.

O objetivo deste conteúdo é compreender que no processo de desenvolvimento de *software* existem diversas possibilidades que o mercado oferece e como elas podem afetar diretamente na qualidade de um bom processo de desenvolvimento.

ITIL

Quando falamos de ITIL (*Information Technology Infrastructure Library*) é necessário abstrair o conceito de processo de desenvolvimento, deste modo, desenvolver um *software* não pode

simplesmente ser uma execução de entregas que resultam em um produto, mas sim em toda a sua cadeia, ou seja, em todo o seu ciclo de vida.

ITIL é um *framework*, no qual podemos encontrar boas práticas e etapas que devem ser aplicadas seguindo os objetivos das organizações a fim de alcançar um resultado de sucesso em seus negócios, as práticas propostas pelo ITIL estão descritas em um conjunto de 5 livros, são eles:

- a) Estratégia de Serviço;
- b) Desenho de Serviço;
- c) Transição de Serviço;
- d) Operação de Serviço;
- e) Melhoria Contínua de Serviço.

Os livros descrevem práticas e processos que são adaptáveis, desse modo, o ITIL pode ser aplicado em diversas organizações, ajudando a solucionar problemas e alavancando os negócios.



VIDEOAULA 01

Utilize o QRcode para assistir!

Assista à videoaula que abordará conceitos importantes sobre o tema.



Dentro de cada livro podemos encontrar processos que indicam a aplicação do ITIL, sendo que todos esses consideram os objetivos de negócio para realizar qualquer prática. Desse modo, a seguir vamos conhecer resumidamente cada livro e os processos descritos dentro de cada um deles:

→ **Service Strategy (Estratégia de Serviço)**

Neste livro, são trabalhados os processos para elaborar um plano que direcione as ações da TI, com o foco no objetivo que deseja ser alcançado pela organização e suas necessidades.

A estratégia de serviço tem como foco modelar as ideias de *softwares* para solucionar problemas e assim ajudar as empresas a crescerem, facilitando alterações no negócio e considerando os objetivos da organização, alguns processos descritos na estratégia de serviço são:

- a) Gerenciamento Estratégico para TI;
- b) Gerenciamento de Portfólio;
- c) Gerenciamento de Demanda;
- d) Gerenciamento Financeiro para TI;
- e) Gerenciamento de Relacionamento de Negócio.

→ **Service Design (Desenho de Serviço)**

O desenho de serviço modela e mapeia os requisitos para compor a solução que a tecnologia da informação irá fornecer e seu ciclo de vida, sempre seguindo os objetivos propostos pelas organizações a fim de cumprir com seu crescimento. Sendo assim, é nessa fase que as necessidades do *software* devem ser ouvidas e esboçadas para que seja possível desenvolver um *software* com qualidade. Os processos que devem ser realizados nessa fase são:

- a) Coordenação de Desenho;
- b) Gerenciamento do Catálogo de Serviço;
- c) Gerenciamento de Nível de Serviço;
- d) Gerenciamento de Disponibilidade;
- e) Gerenciamento da Capacidade;
- f) Gerenciamento de Continuidade de Serviço;
- g) Gerenciamento de Segurança da Informação;
- h) Gerenciamento de Fornecedores.

→ **Service Transition (Transição de Serviço)**

Na transição de serviço são implementados, por intermédio de processos, o que foi realizado durante o desenho de serviço, com o foco de liberar para a organização o serviço ou *software*. Sendo assim, com a transição de serviço se desenvolve e realiza a implantação do *software*.

Desse modo, para que os processos descritos se tornem um sucesso, é preciso de apoio da organização para disponibilizar ao usuário o serviço desenvolvido, com isso é necessário ter atenção, pois, no momento de transição, os riscos e falhas podem ser maiores do que em outras etapas.

Assim como as demais etapas na transição de serviço, também se faz necessário alguns processos, vamos descrever somente alguns:

- a) Gerenciamento do Conhecimento;
- b) Avaliar Mudança;
- c) Gerenciamento de Liberação e Implantação;
- d) Validação e Teste;
- e) Planejamento e Suporte da Transição.

INDICAÇÃO DE VÍDEO

Agora assista ao vídeo, no qual será apresentado de forma breve e clara um pouco do ITIL. Disponível em: <<https://www.youtube.com/watch?v=5zm-ruBIfYg>>. Acesso em: 5 mar. 2021.

→ **Operation Service (Operação de Serviço)**

Após a transição de serviço, o usuário começa a utilizar da solução e assim surge a necessidade de suporte ao usuário. Desse modo, a operação de serviço acontece quando a empresa de TI ou o departamento de TI realiza o suporte do *software* entregue, ou seja, garante que o *software* se mantenha em funcionamento atendendo as necessidades da organização e assim impactando na mesma.

Na operação de serviço, o valor do serviço/*software*, os seus benefícios e como ele impactou no valor de negócio da organização precisam ser claros para os clientes, a fim de conseguir tirar conclusões objetivas sobre seu uso. Alguns processos a serem executados são:

- a) Gerenciamento de Incidente e Problemas;
- b) Gerenciamento de Acessos e Eventos;
- c) Cumprimento da Requisição.

→ **Continual Service Improvement (Melhoria Contínua de Serviço)**

Quando o *software* já está em funcionamento, basta para a equipe somente acompanhar e implementar melhorias para otimizar o sistema, a fim de atender às mudanças necessárias para a organização.

Importante que essas melhorias sejam continuadas, para o ITIL um projeto/serviço/*software* possua um ciclo de vida contínuo, sendo melhorado sempre que necessário para atender o mercado ao qual a organização está inserida.

Conforme observaremos na Figura 1 o ITIL é um *framework* com uma arquitetura interligada, ou seja, seus conceitos e boas práticas se ligam e todos sempre estão envoltos na melhoria contínua, dessa forma até mesmo o *framework* pode ser melhorado e adaptado para ser melhor aplicado conforme a situação e os pontos de objetivo para as organizações.

Figura 01 - Arquitetura ITIL



Fonte: disponível em: <<https://www.diariodeti.com.br/wp-content/uploads/2015/12/ciclo-de-vida-til-1.png>>. Acesso em: 5 mar. 2021.

Muitas empresas de *software* já utilizam do ITIL para seus processos de suporte ao usuário, porém é um *framework* muito adaptável e com seus conceitos e boas práticas será possível aplicar o ITIL no processo de desenvolvimento de *software*, claro que podendo ser um norteador para a inclusão de outros *frameworks* em conjunto.

Com o auxílio desse *framework* é possível estruturar um *software* que ao ser desenvolvido agregue valor junto à organização, de uma forma na qual essa instituição possa impactar a vida do usuário final, e com a constante melhoria o ITIL favorece a construção de soluções sólidas.

Design Thinking

Bom aluno, agora vamos conhecer uma pouco sobre o *Design Thinking* e compreender como essa metodologia pode ser um processo de desenvolvimento de *software*, que eleva a possibilidade de construir um *software* bem-sucedido.



VIDEOAULA 02

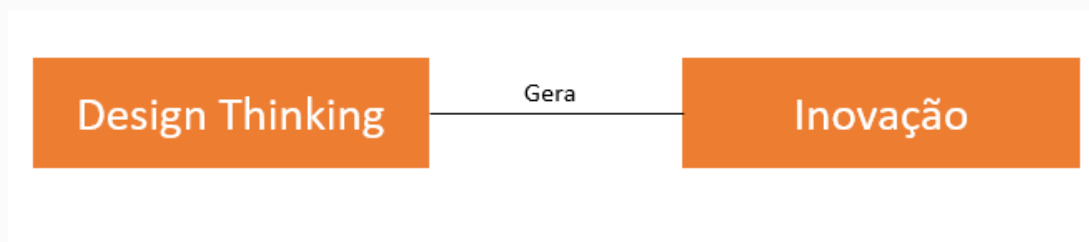
Utilize o QRcode para assistir!

Assista à videoaula para entender melhor o assunto.



O termo *Design Thinking*, em português “Pensamento de Design”, é uma metodologia colaborativa e rápida que propõem o auxílio na resolução de problemas. A proposta inicial do Pensamento de Design é gerar um olhar crítico, oportunista que produza valores para o negócio. Transformando assim, o modo de visualizar problemas e suas soluções, gerando inovação.

Figura 02 - Design Thinking x Inovação



Fonte: o autor (2021)

Desse modo, conforme demonstra a Figura 02, o *Design Thinking* apresenta também a geração de inovação, para solucionar problemas. Para isso, o Pensamento de Design possui etapas que quando seguidas resultam em soluções inovadoras.

INDICAÇÃO DE VÍDEO

Agora assista ao vídeo, no qual apresentado de forma breve o *Design Thinking*. Disponível em: <<https://www.youtube.com/watch?v=4N5arlodPGw>>. Acesso em: 5 mar. 2021.

O *Design Thinking* (DT), tem o foco total no ser humano e suas necessidades, ele é utilizado em muitas empresas de *softwares* para resolver problemas complexos, pois também é uma metodologia facilmente adaptável.

Para conseguir gerar inovação e solucionar o problema, o DT defende que é necessário visualizar a situação com outro olhar, transformando necessidades em oportunidades, agregando os valores humanos com negócios e tecnologia. Dessa maneira, muitos processos do DT resultam em *softwares* aplicáveis para a área empresarial.

Na literatura existem várias abordagens para o *Design Thinking*, a abordagem que iremos conhecer neste estudo sobre o *Design Thinking* é dividida em cinco etapas, sendo elas:

- a) Imersão;
- b) Análise e Síntese;
- c) Ideação;
- d) Prototipação;
- e) Validação/Implantação.

Figura 03 - Estrutura *Design Thinking*



Fonte: Nascimento (2020)

Como podemos observar na Figura 03, é possível notar as cinco fases do *Design Thinking*, de modo circular. No DT todas as fases possuem o mesmo valor e se tornam necessárias para solucionar um problema, estas fases bem aplicadas resultam em soluções/*softwares* de sucesso.

Para melhor compreensão, iremos detalhar cada fase do pensamento de design e como elas funcionam:

→ **Imersão**

Nesta fase, o desenvolvedor é chamado a se colocar no lugar do usuário que está com problemas e assim levantar opiniões, necessidades, sentir o problema e saber com o que realmente o usuário está sofrendo. Desse modo, é uma premissa nesta fase sentir empatia para com as dificuldades do usuário.

Nesse ponto, é preciso observar como o usuário tenta resolver esse problema já existente, e assim compreender como funciona o processo que resulta nessa necessidade, por intermédio de debates e conversas é possível realizar esse processo de compreensão.

→ **Análise e Síntese**

A análise e síntese é a etapa que as informações coletadas na fase anterior se organizam de forma física. Sendo assim, em desenhos e rabiscos da necessidade e de possíveis soluções.

Podendo usar recursos visuais de quadros, folhas, *post it* e outros com a proposta para sempre estar visível para todos os participantes, assim tornando claro todas as necessidades e propostas de soluções. É importante deixar as informações geradas nesta etapa com fácil acesso, para futura visualização caso necessário.

→ **Ideação**

Nesta fase a proposta é idealizar soluções, pode ser realizado um *Brainstorm* (chuva de ideias) com os participantes, dessa maneira exibindo ideias e pensamentos convergentes entre os integrantes do time.

A proposta é explanar sobre os pensamentos adversos e inovadores, para que a solução seja algo novo dentro do contexto já conhecido pelo usuário final. Assim, será possível analisar e validar qual ideia é a melhor a seguir, implementar e solucionar o problema.

→ **Prototipação**

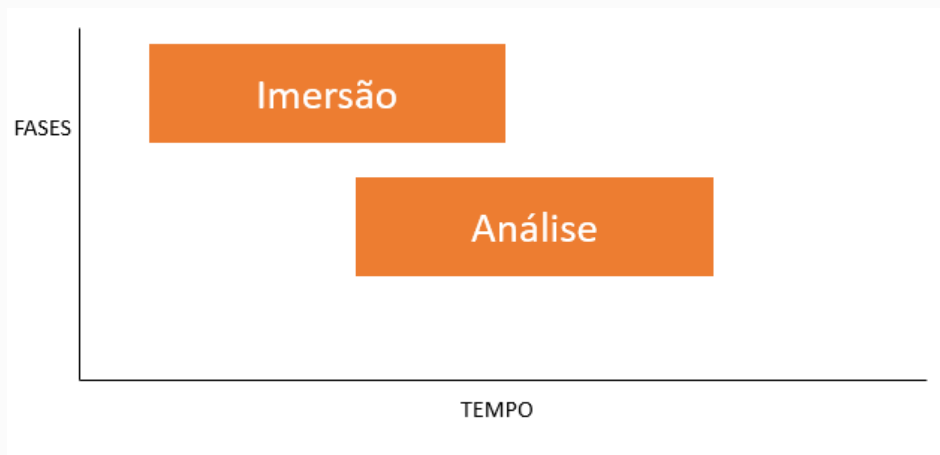
Na prototipação é quando se escolhe uma das ideias explanadas pelos participantes após o processo de ideação. Nessa fase se desenvolve um modelo de protótipo com a ideia escolhida, podendo ser ele um MVP (Mínimo Produto Viável) e assim, reduz o risco de desenvolver uma solução que seja ineficaz, além de não gerar grandes custos.

→ **Validação/Implantação**

Na etapa de validação/implantação a solução que passou por prototipação é implantada para o usuário para assim ser validada e receber um respaldo dele. Com isso é possível saber se o problema foi solucionado, e se a solução entregue foi um sucesso ou não.

Podendo por intermédio da validação do usuário, conhecer e implementar futuras melhorias no *software* desenvolvido, de modo que este se torne uma ferramenta que agregue valor no cotidiano do usuário.

Figura 04 - Fases x Tempo



Fonte: o autor (2021)

A Figura 04 demonstra que o DT não precisa respeitar um fluxo sequencial das cinco fases apresentadas. Pois em alguns casos uma fase se inicia no meio da fase anterior, dessa maneira, reduzindo custo e tempo.



VIDEOAULA 03

Utilize o QRcode para assistir!

Assista à videoaula a fim de ampliar seus conhecimentos sobre o assunto.



O Pensamento de Design também pode ser aplicado de forma circular para o desenvolvimento de *software*, a cada final de ciclo, que é o percorrer das cinco fases, se entrega uma iteração do *software*.

Dessa forma, o *Design Thinking* é uma proposta nova para desenvolver *softwares*, com foco nas pessoas, que busca por intermédio da solução de problemas, entregar um *software* inovador e que agreguem valor para o usuário e seu negócio, impactando assim o seu dia a dia.

Caro aluno, durante esse tempo conhecemos diversos recursos, como: UML, BPMN, RUP, PMBOK, ITIL e *Design Thinking*. A partir disso você já compreende que o conjunto de recursos apresentados são formas de buscar um resultado de sucesso no processo de desenvolvimento de *software*.

Infelizmente, não existe uma receita mágica que demonstre como desenvolver um *software* bem-sucedido, porém com o conhecimento adquirido nesta disciplina é possível analisar, e escolher qual é melhor modelo de processo e boas práticas a seguir para desenvolver um *software*.

Encerramento

Chegamos ao fim de mais uma unidade, prezado aluno. Espero que tenha gostado dos conhecimentos que foram apresentados nesta unidade e que esses sejam de grande contribuição para sua vida profissional.

Na aula 01, você conheceu o ITIL e seus conceitos básicos, a visão sobre serviços, suas etapas e os processos presentes em cada uma delas, além de como isso pode ser aplicado durante o processo de desenvolvimento de *softwares*.

Em nossa aula 02, aprofundamos os nossos conhecimentos dentro do *Design Thinking* e sua proposta de agregar valores humanos e de negócio para solucionar problemas e como ele pode ser aplicado no cotidiano de desenvolvimento de *softwares*.

Com isso encerramos a unidade 04, com um olhar de que o processo de desenvolvimento de *software* deve seguir o contexto de negócio do cliente. Assim encerramos também a nossa disciplina, despertando em você mais interesse e desejo de seguir em frente. Até a próxima e bons estudos!

Bom trabalho!



ENCERRAMENTO

Utilize o QRcode para assistir!



Esperamos que este guia o tenha ajudado compreender a organização e o funcionamento de seu curso. Outras questões importantes relacionadas ao curso serão disponibilizadas pela coordenação.
Grande abraço e sucesso!

