

# LEVANTAMENTO DE REQUISITOS, ANÁLISE E DOCUMENTAÇÃO DE SISTEMAS

APÓS TODO EMBASAMENTO FUNDAMENTAL E TEÓRICO SOBRE A CONCEPÇÃO DE UMA EMPRESA E DE UM NEGÓCIO, INICIA-SE A PARTE DA ANÁLISE, LEVANTAMENTO DE REQUISITOS E DOCUMENTAÇÃO DOS SISTEMAS NECESSÁRIOS, PARA DAR SUBSÍDIOS AO CONTROLE DOS PRODUTOS E SERVIÇOS A SEREM GERIDOS PELA EMPRESA.

AUTOR(A): PROF. ANTONIO ANDRADE DOS SANTOS

## Desenvolvimento de Requisitos

Identificada a necessidade do desenvolvimento de um sistema, primeiramente, precisam ser realizadas algumas reuniões entre os profissionais de TI e o “cliente” (pessoa interessada no sistema) para o entendimento do que será desenvolvido.

Precisamos de atenção nas expectativas de todos os envolvidos em um projeto. O cliente, o analista, o programador, o usuário... Enfim. Pois nem tudo que se vislumbra pode ser viável financeiramente, ou o desenvolvimento pode não atender as necessidades. Vamos ver o descontraído vídeo a seguir para entender um pouco mais destes diferentes mundos:

Legenda: O DILEMA DE UM PROJETO DE SOFTWARE

O vídeo acima trata o assunto de forma engraçada, mas retrata bem as expectativas de cada participante de um processo de desenvolvimento e a realidade que muitas vezes não atendem as expectativas. Pior que isso, não atende as necessidades imprescindíveis para se gerenciar.

É importante ter sempre em mente, quais são as atividades fundamentais no desenvolvimento de um software, para tentar sempre desenvolver o melhor software, dentro do conceito de melhor custo/benefício. Ou seja, o menor custo com maior benefício.

Vamos abrir um parênteses para dar uma pequena olhada para um conceito muito interessante em relação ao que se fala hoje em dia de melhor custo/benefício. O MVP:

# MVP

O MVP (*Minimum Viable Product*) é uma metodologia que trata a definição de um Produto Mínimo Viável, que serve para definir o ciclo de vida de um empreendimento.

É importante sempre valorizar os recursos existentes e buscar a identificação de determinada tendência de forma rápida, para direcionar bem os recursos empreendidos.

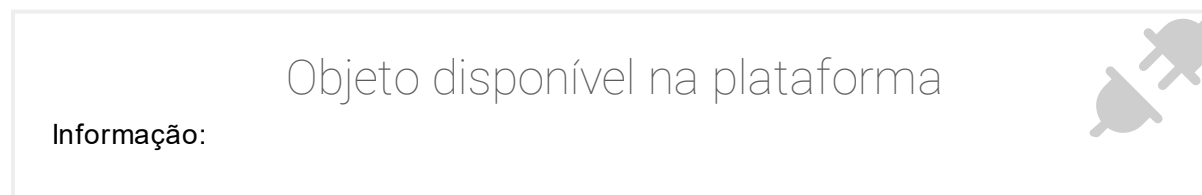
Segundo SEBRAE (2017), o americano Eric Ries “passou os últimos anos combinando ideias de marketing, tecnologia e gestão e criou o termo ‘*Lean Startup*’”. Este conceito se concentra em desenvolver um determinado produto, de forma ágil, sem desperdícios de tempo, custo ou recursos, chegando-se ao ponto de comercialização o mais rápido possível. (SEBRAE, 2017).

Essa metodologia foi idealizada com base na criação de Startups, mas vem se tornando também aceitável e utilizada para o start em qualquer outro tipo de empreendimento.

Segundo definição documentada no SEBRAE (2017), os conceitos referentes às três letras do MVP são:

- *Minimum*: o menor tamanho possível a ser entregue no menor tempo possível.
- *Viable*: um produto que possa ser adotado pelo cliente e que se possível gere lucro.
- *Product*: um produto coeso e útil com as funcionalidades essenciais de utilização.

Veja o gráfico abaixo, e veja quantos produtos podem ser criados, sobre uma mesma ideia de negócio:



## EXEMPLO DE PLANO DE EVOLUÇÃO DO PRODUTO BASEADO EM MVP

Sobre a ideia de um simples cortador de gramas, veja quantos produtos foram criados. Oito para ser exato! E poderia ter outras com certeza!!! . Há casos em que uma versão do produto substitui a versão anterior. No entanto, há produtos que as várias versões são viáveis a diferentes tipos de consumidores.

Legal ter uma ideia fechada sobre um produto! Certo?

No entanto, mesmo enquanto se desenvolve a primeira versão, sempre vêm novas ideias sobre o produto que está sendo desenvolvido.

No início do desenvolvimento, é importante se chegar à definição clara do que é a versão inicial do produto (MVP1). As novas ideias precisam ser documentadas para uma versão 2 ou superior (MVP2, MVP3, Etc.). Isso fará com que a primeira versão do produto seja colocada em uso mais rapidamente. Com a versão MVP1 do produto sendo consumida, consegue-se mensurar seus pontos fracos e fortes e até agregar novas funcionalidades para uma nova versão do produto.

## QUE TAL VER MAIS SOBRE MVP?

Veja o artigo (com vídeo) - “Novos Empreendedores #10 – MVP”:  
<http://academiasebrae.com.br/atitude/novos-empreendedores-10-mvp/>  
(<http://academiasebrae.com.br/atitude/novos-empreendedores-10-mvp/>)  
Estude “O Guia Prático para o seu MVP – Minimum Viable Product”: <https://goo.gl/8cTT3a>  
(<https://goo.gl/8cTT3a>)  
Na revista PEGN, veja o que expõe Ribeiro Jr (2016) sobre “5 Passos Para Fazer Um Produto Mínimo Viável (MVP)”: <https://goo.gl/wBmitB> (<https://goo.gl/wBmitB>)  
Um clube de empreendedores que o SEBRAE criou que pode ajudar muito quem está tentando entender melhor sobre o assunto e obter ajuda para começar:  
<https://clubesebrae.com.br/startup-conecta/seja-bem-vindo-ao-startup-conecta>  
(<https://clubesebrae.com.br/startup-conecta/seja-bem-vindo-ao-startup-conecta>)

(ht  
tp:  
//s  
av  
efr  
o  
m.  
ne  
t/?  
url  
=h  
ttp  
s  
%  
3A  
%  
2F  
%  
2F  
yo  
ut  
u.  
be  
%  
2F  
b0  
M

a  
m  
Th  
bh  
cU  
&  
ut  
m  
\_s  
ou  
rc  
e=  
ch  
a  
m  
ele  
on  
&  
ut  
m  
-  
m  
ed  
iu  
m  
=e  
xt  
en  
si  
on  
s  
&  
ut  
m  
\_c  
a  
m  
..

pa  
ig  
n=  
lin  
k\_  
m  
od  
ifi

Assista também ao vídeo: <https://youtu.be/b0MamThbhcU> (<https://youtu.be/b0MamThbhcU>)er)

Agora que entendemos e estamos motivados a desenvolver, voltamos ao entendimento de como conduzir a criação do projeto do produto a ser criado.

## Especificação, análise, implementação, validação e entrega

Vejamos um pouco mais sobre os conceitos fundamentais para o desenvolvimento de um bom software:

Informação:

Objeto disponível na plataforma



### ATIVIDADES FUNDAMENTAIS NO DESENVOLVIMENTO DE SOFTWARE

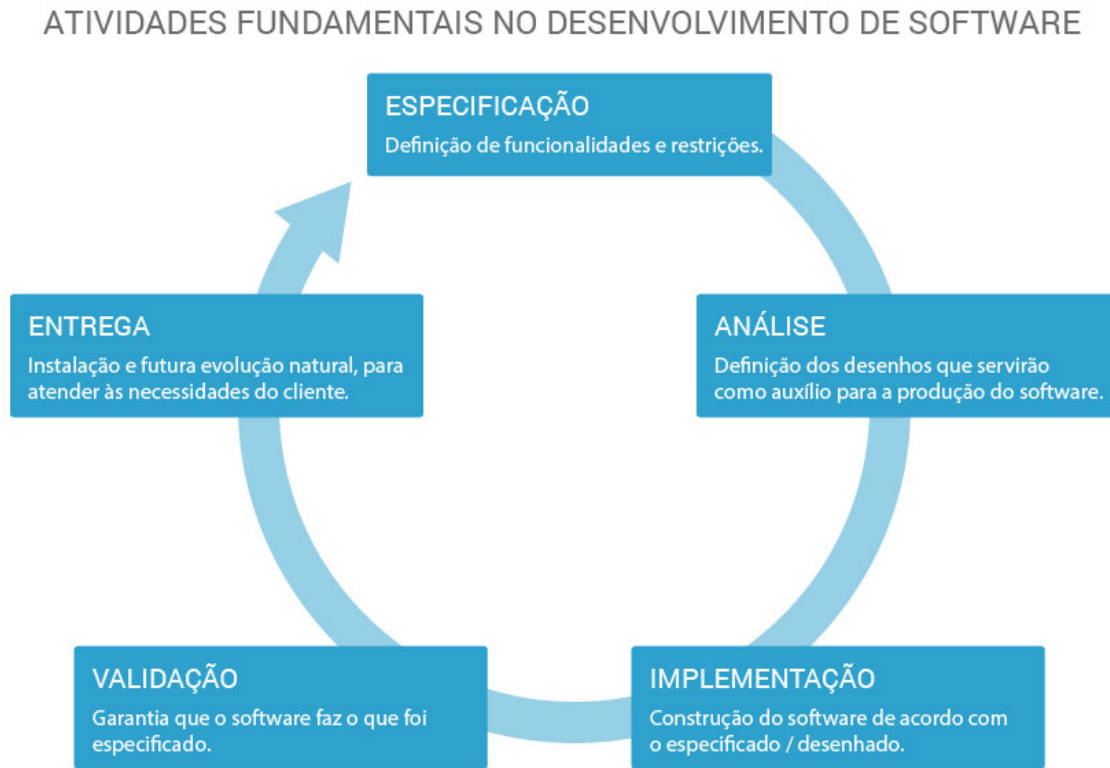
Entendeu?

Todo processo se dá pelas seguintes fases:

- **Especificação:** que é a definição das funcionalidades e restrições que podem ser encontradas em um projeto.
- **Análise:** onde serão desenhados os processos e requisitos para produção do software.
- **Implementação:** Criação do software de acordo com tudo que foi especificado ou desenhado nas fases anteriores.
- **Validação:** Onde se busca a garantia de que o software deve fazer o que foi especificado, com a realização de testes inclusive.
- **Entrega:** É a fase em que faz a instalação do software no cliente e acompanha a evolução natural do mesmo. Sempre objetivando atender as necessidades sempre devidamente especificadas, analisadas, implementadas e validadas.

Independentemente de qual seja o modelo de desenvolvimento adotado, a sequência de desenvolvimento do trabalho e fases, não mudam muito em relação ao que acaba de ser apresentado. Você verá apenas, que em alguns modelos, destaca-se mais uma ou outra fase, ou se dá mais ênfases a outras.

Resumindo o que foi dito, analise a seguinte imagem:



Legenda: ATIVIDADES FUNDAMENTAIS NO DESENVOLVIMENTO DE SOFTWARE

Dando sequência à análise, é importante destacarmos aqui alguns modelos consagrados que podem ser adotados para acompanhar todo ciclo de desenvolvimento de um software.

## Modelo espiral

Neste modelo o desenvolvimento de um software se divide em iterações com as atividades de: planejamento; análise de riscos; modelagem do software; construção do código e testes; implantação, entrega e *feedback*; por fim, especificação.

Veja melhor na ilustração seguinte:



Legenda: MODELO ESPIRAL

## VEJA MAIS:

Dissertação de Mestrado de Leila Ribeiro de Oliveira, que dispõe deste e diversos outros modelos, fazendo uma análise sobre técnicas de estimativas de projeto de software:

Link: <http://www.fumec.br/revistas/sigc/article/viewFile/1480/961>  
(<http://www.fumec.br/revistas/sigc/article/viewFile/1480/961>)

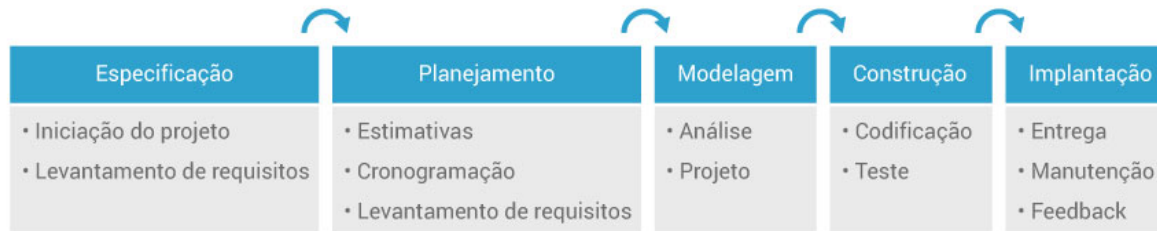
## Modelo cascata

Este modelo, um pouco mais simplificado aparentemente, prevê as seguintes fases:

- Especificação: Onde se inicia o projeto, partindo de uma ideia formada, para que se consiga levantar e documentar todos os requisitos necessários.
- Planejamento: Fase em que se realizam as estimativas se elabora os cronogramas e finalizam os levantamentos de requisitos.
- Modelagem: Fase em que se documenta a análise e fecha o projeto.
- Construção: Realiza-se a codificação ou criação do produto e também os testes.

- Implantação: Entrega do produto, treinamento, manutenção de possíveis inconformidades e acompanhamento e documentação do *feedback*.

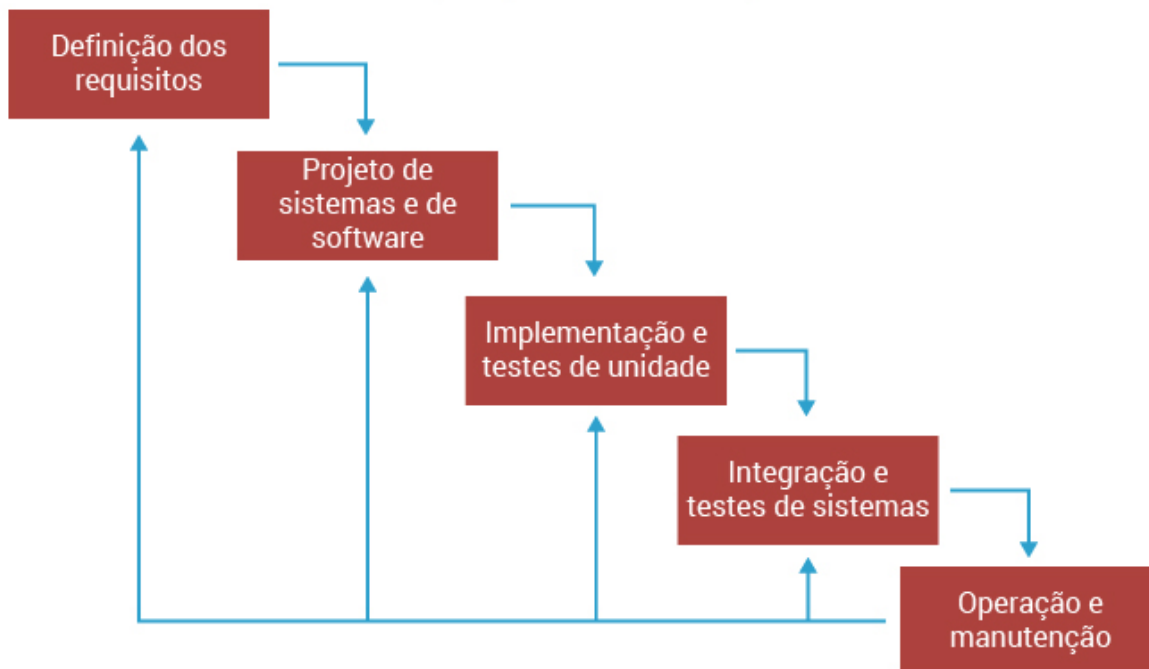
DIAGRAMA: MODELO CASCATA



Legenda: DIAGRAMA: MODELO CASCATA

Segundo Oliveira (2016), este modelo teve início com Royce (1970), sendo evoluído mais tarde dividido em 5 fases. (SOMMERVILLE, 2011). Veja o modelo representado graficamente na seguinte figura:

(Royce, 1970)



Legenda: MODELO CASCATA

VEJA MAIS:

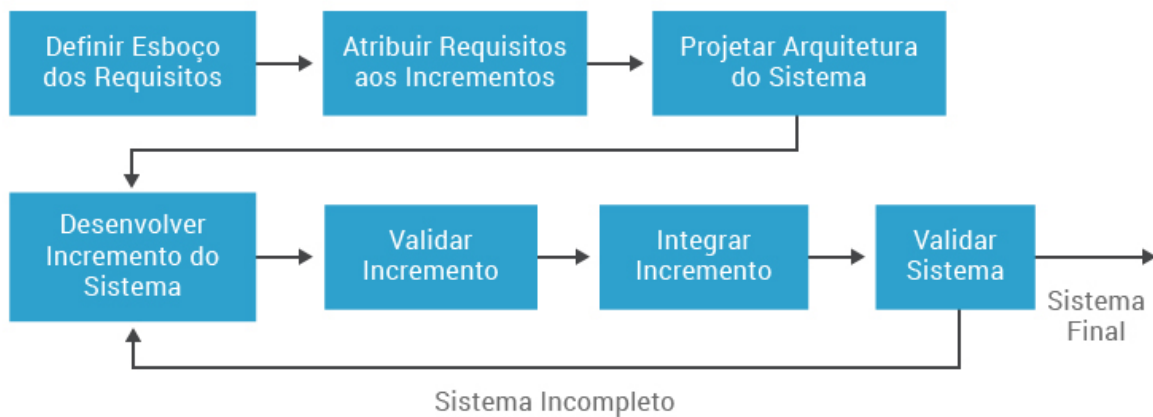


Acesse o trabalho de mestrado de Paulo H. R. de Oliveira. Sobre Engenharia de Requisitos (<https://bibliotecatede.uninove.br/bitstream/tede/1479/2/Paulo%20Henrique%20Ribeiro%20De%20Oliveira.pdf>): <https://goo.gl/CjDxqF> (<https://goo.gl/CjDxqF>)

Veja diversos modelos sobre O processo de desenvolvimento de Software: [http://www.macoratti.net/proc\\_sw1.htm](http://www.macoratti.net/proc_sw1.htm) ([http://www.macoratti.net/proc\\_sw1.htm](http://www.macoratti.net/proc_sw1.htm))

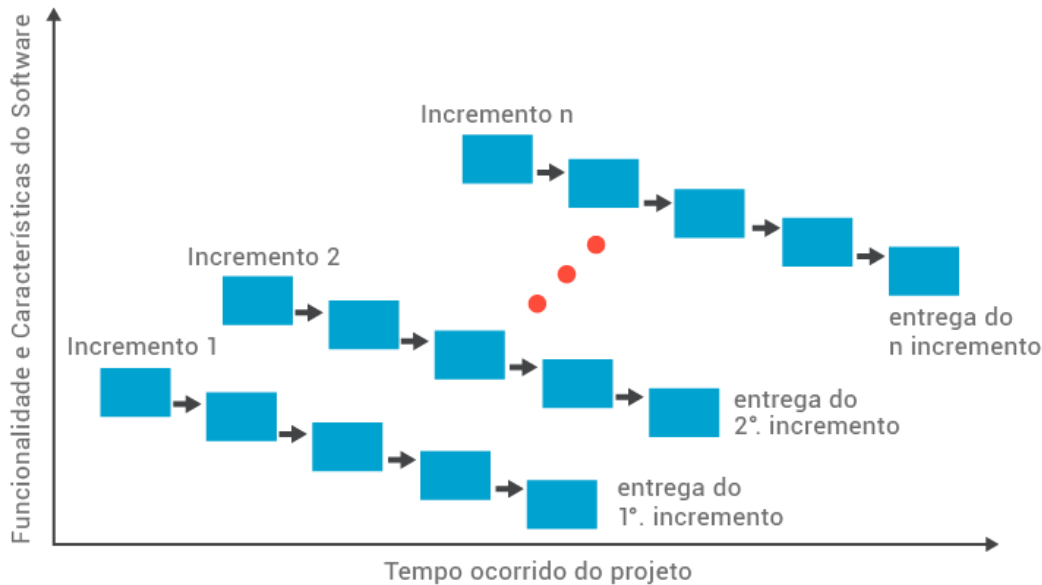
## Modelo Incremental

O modelo incremental permite a criação de um produto básico, com a possibilidade de fazer incrementos e melhorar o produto, a partir de novas necessidades.



Legenda: MODELO INCREMENTAL

Analise a figura abaixo para entender melhor as funcionalidades e características na elaboração de um software:



Legenda: MODELO INCREMENTAL

A cada novo incremento, o produto é melhorado. Cada incremento, passa novamente por todas as fases do modelo.

Vejamos um exemplo de solução incremental sobre um produto existente, porém com uma nova necessidade:

## VEJA MAIS:

Leia mais sobre "O Modelo Incremental": <http://jkolb.com.br/o-modelo-incremental/>  
(<http://jkolb.com.br/o-modelo-incremental/>)

Independentemente do modelo utilizado para elaboração do projeto, é fácil observar que é fundamental o levantamento de requisitos, realizado de forma minuciosa e cuidadosa para que o projeto seja bem elaborado e desenvolvido.

## Requisitos

Vamos nos aprofundar um pouco mais sobre requisitos, assistindo ao pequeno vídeo a seguir:



Legenda: DESENVOLVIMENTO DE REQUISITOS

Como se pode observar, os requisitos são elaborados por fases, sendo:

- Elicitação: Momento onde o analista interage com os usuários (ou futuro usuários do sistema).
- Análise: Fase de registro, organização das ideias e análise da melhor solução a ser desenvolvida.
- Especificação: Escrita de um documento que contempla a solução a ser desenvolvida.
- Validação: Momento em que o analista volta ao cliente (usuário) para validar tudo que foi levantado, organizado e documentado. Esta fase pode se repetir várias vezes, para permitir adaptações se necessário.

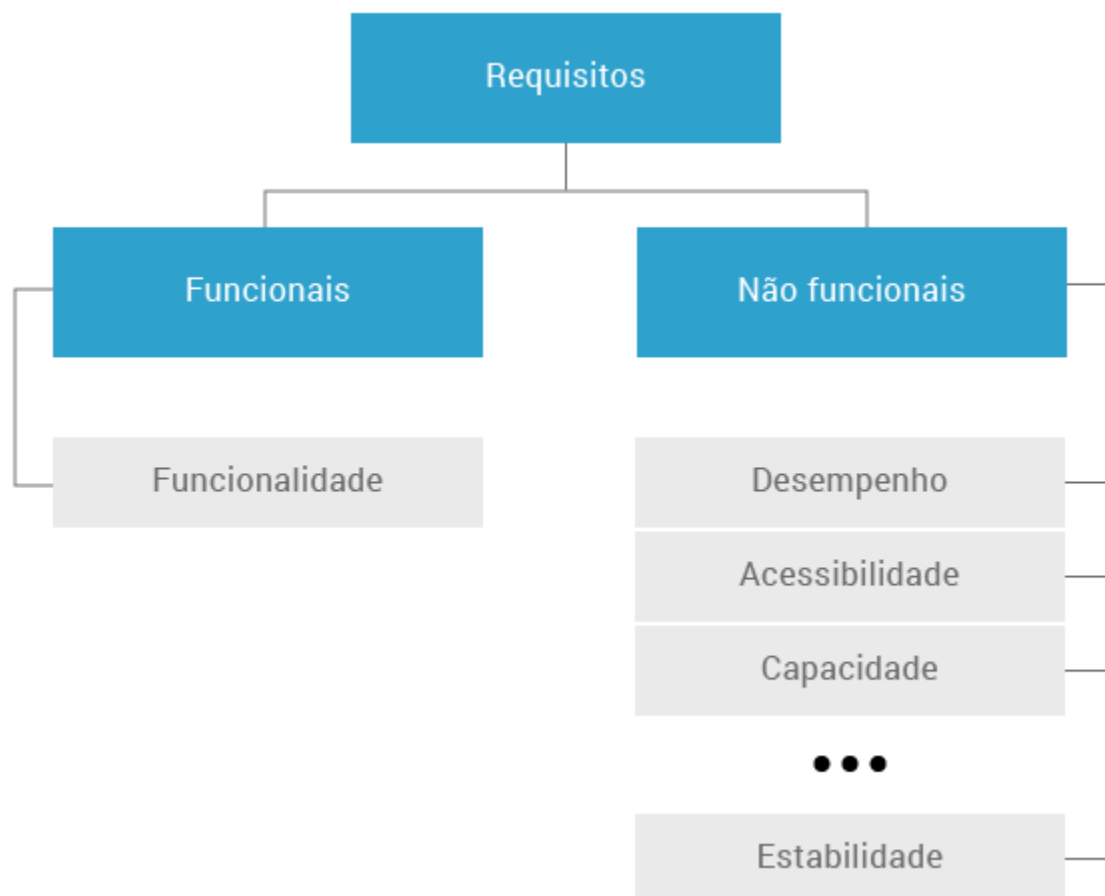
Após analista e cliente estarem satisfeitos, o levantamento de requisitos está terminado, podendo se encaminhar para a próxima fase do processo de desenvolvimento do software / produto.

No entanto, é necessário entender dois tipos diferentes de requisitos: funcionais e não funcionais.

## Requisitos funcionais e não funcionais

Os requisitos são as necessidades ou desejos dos clientes, que precisam ser entendidos e documentados adequadamente. Pois esses requisitos são a base do desenvolvimento.

Analise a figura abaixo para melhor entendimento da diferença entre os requisitos funcionais e não funcionais:



### Legenda: CATEGORIZAÇÃO CLÁSSICA DO LEVANTAMENTO DE REQUISITOS

Os requisitos funcionais são requisitos que estão ligados diretamente às funcionalidades do que esta sendo desenvolvido. Em um software, por exemplo:

- Solicitação de identificação para entrada em sistema.
- Incluir, alterar, pesquisar, excluir, consultar informações referente a determinada tela em um sistema.
- Um relatório ou uma grade de consulta específica.
- Um pagamento de determinada compra.
- Consultas de saldos em conta ou de produtos.

Entre muitas outras, claro!

Os requisitos funcionais regem o que o sistema deve fazer, enquanto que os requisitos não funcionais regem como isso será feito. (VENTURA, 2016).

Segundo ainda Ventura (2016), os Requisitos Não Funcionais (RNF) são difíceis de estimar. Pois trata de mensurar o tempo, esforço e o custo de determinadas atividades.

Estes requisitos estão ligados diretamente a: desempenho, disponibilidade, segurança, interoperabilidade, usabilidade, compatibilidade, confiabilidade, padrões e inclusive à legalidade do que está se produzindo. (VENTURA, 2016).

Estes documentos devem ter um padrão. Para isso é importante seguir um modelo.

## Modelo de levantamento de requisitos

Faça download do modelo de requisitos e faça as adaptações necessárias para o projeto que estiver elaborando, aproveitando os modelos de Requisitos Funcionais (RFxxx) e Requisitos Não Funcionais (RNFxxx ou NFxxx). Onde xxx é um número sequencial do requisito.

Abaixo, download do modelo de levantamento de requisitos para trabalhar nele.

### VEJA MAIS:

O que é um Requisito Não-Funcional: <http://www.ateomomento.com.br/o-que-e-um-requisito-nao-funcional/> (<http://www.ateomomento.com.br/o-que-e-um-requisito-nao-funcional/>)

O que são requisitos funcionais e requisitos não funcionais?:  
<http://codificar.com.br/aplicativos/requisitos-funcionais-nao-funcionais/>  
(<http://codificar.com.br/aplicativos/requisitos-funcionais-nao-funcionais/>)

Depois de tanto conceito bacana, é hora de revisar os conteúdos dessa aula e colocar alguma coisa em prática.

Sei que você não vê a hora desse momento!

## Proposta de atividade em sala

- Conhecimento

Abordar sobre requisitos, entendendo o que é um MVP para elaboração de um projeto com especificação, análise, implementação, validação e entrega.

Abordar os conceitos dos modelos: espiral; cascata e incremental, e a diferença entre requisitos funcionais e não funcionais.

- Habilidade

Buscar como desenvolver um MVP e os modelos de requisitos funcionais e não funcionais.

- Atitude

Criar um MVP1 e criar a documentação do levantamento de requisitos funcionais e não funcionais.

## O que foi visto neste tópico

Nesta aula tivemos uma introdução ao desenvolvimento de requisitos, sobre a importância do MVP nos negócios dos dias atuais e a importância de uma análise bem realizada.

Vimos ainda sobre alguns modelos de desenvolvimento: espiral, cascata e incremental. Além de requisitos funcionais e não funcionais.

E o mais importante, vimos modelos que podem e devem ser utilizados no dia-a-dia do desenvolvimento dos projetos.

## Teste seu aprendizado

### ATIVIDADE FINAL

Quando falamos em MVP, estamos mais especificamente falando do que?

- A. Estamos falando de um produto mínimo viável.
- B. É um produto menor viável.
- C. É a definição de um produto minimamente variável.
- D. Produto mais variável.

Em relação aos requisitos funcionais e não funcionais, estamos nos referindo respectivamente à:

- A. Os requisitos funcionais são o que será feito, e os requisitos não funcionais, como será feito.
- B. Requisitos funcionais são os que estão diretamente ligados às funcionalidades do software; e não funcionais àqueles que não funcionarão.
- C. Os requisitos não funcionais são aqueles que definem o que deve ser feito e os funcionais, como será feito.
- D. Como será feito e o que deve ser feito.

As fases da elaboração de requisitos envolvem:

- A. Elicitação, Análise, Especificação e Validação.
- B. Especificação, Planejamento, Modelagem, Construção e Implantação.
- C. Elicitação, Planejamento, Construção e Implantação.
- D. Elicitação Especificação, Validação e Implantação.

## REFERÊNCIA

OLIVEIRA. Paulo Henrique Ribeiro de. Engenharia de Requisitos Aplicada em Sistema Legado de Gestão e Custeio de Propostas Comerciais: pesquisa-ação em empresa do setor de estampa. Disponível em: <<https://goo.gl/CjDxqF> (<https://goo.gl/CjDxqF>)>. Acesso em 26 dez 2017.

RIBEIRO JR, Valdir. 5 Passos Para Fazer Um Produto Mínimo Viável (MVP). Revista PEGN, 2016. Disponível em: <<https://goo.gl/wBmitB> (<https://goo.gl/wBmitB>)>. Acesso em 26 dez 2017.

ROYCE. W. W., Managing the development of large software systems ([http://leadinganswers.typepad.com/leading\\_answers/files/original\\_waterfall\\_paper\\_winston\\_royce.pdf](http://leadinganswers.typepad.com/leading_answers/files/original_waterfall_paper_winston_royce.pdf)). In: