

# FIAP GRADUAÇÃO

**ASSUNTO:**

**ENGENHARIA E DESIGN DE SOFTWARE**

**AULA:**

**3-LEVANTAMENTO DE REQUISITO DE SOFTWARE**

**PROFESSOR:**

**RENATO JARDIM PARDUCCI**

**PROFRENATO.PARDUCCI@FIAP.COM.BR**

**AGENDA DA AULA****LEVANTAMENTO DE REQUISITOS**

- ✓ Técnicas

**ENGENHARIA DE  
REQUISITOS – VISÃO DO  
PROCESSO**

## APRENDENDO NA PRÁTICA



O dono do estacionamento fechou o contrato com a sua empresa, finalmente!

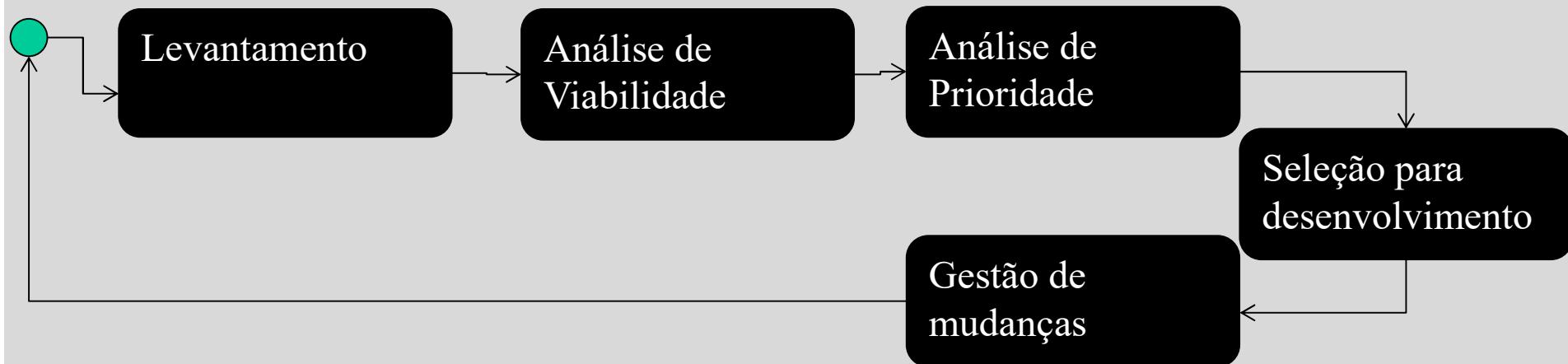
Agora, você precisa iniciar os trabalhos de identificação de requerimentos a serem atendidos no projeto, sendo que esses requerimentos incluem todos os aspectos relevantes de funcionamento esperado do software e requisitos técnicos de implementação (reaproveitamento de componentes de software já existentes, arquitetura de segregação de componentes, infraestrutura).

Você e a sua equipe, vão a partir de agora, aplicar técnicas para identificar e registrar esses requerimentos.

## ENGENHARIA DE REQUISITOS

A **engenharia de requisitos** é um processo que engloba todas as atividades que contribuem para a produção de um documento de requisitos e sua manutenção ao longo do tempo.

O processo de engenharia de requisitos é composto pelas atividades:



*O gerenciamento de requisitos é um modelo para encontrar, documentar, organizar e rastrear os requisitos variáveis de um sistema, controlando assim as mudanças.*

## ENGENHARIA DE REQUISITOS

### **IDENTIFICAÇÃO / LEVANTAMENTO DE REQUISITOS**

Consiste em descobrir quais questões de negócio ou do dia a dia das pessoas precisam ser resolvidas.

Essas necessidades podem envolver ou não, a revisão ou aposentadoria de soluções existentes (sistemas legados).

## ENGENHARIA DE REQUISITOS

**O trabalho de levantamento de requisitos é o primeiro passo de um projeto e envolve as etapas:**

- 1º) Registrar interesses das partes interessadas, criando documentos em formato livre, chamados **histórias de usuários**;
- 2º) Coletar **documentação de sistemas legados** que expliquem os repositórios de dados e a lógica de funcionamento dos softwares que serão substituídos pela nova solução;
- 3º) Usar o aprendizado desse trabalho para construir um **painel resumido de requisitos e regras de negócio associadas**, criando uma lista de requisitos.

## ENGENHARIA DE REQUISITOS

Quando vamos investigar os requisitos de um software, com quem devemos falar?

- Usuários (clientes das soluções de TI)
- Patrocinador (quem está “financiando” o projeto)
- Equipe de TI ligada a Operação das Soluções Atuais
- Equipe de TI ligada ao Suporte das Soluções Atuais
- Equipe de TI ligada a Inovação (Projetos)

**TÉCNICAS DE  
LEVANTAMENTO DE  
REQUISITOS**

## ENGENHARIA DE REQUISITOS

## Preparação para iniciar os trabalhos

A partir de agora, você terá que realizar diversos registros sobre os levantamentos de requisitos e futuramente sobre o desenho da solução de software.

Para começar de forma organizada, vamos usar duas ferramentas:

-  com a qual você vai planejar e controlar o projeto
-  que você usará para guardar documentos do seu projeto

## ENGENHARIA DE REQUISITOS



## Preparação para iniciar os trabalhos

No GIT, você vai criar o repositório “PROJETO ESTACIONAMENTO”.

Dentro desse repositório, crie a pasta ”REQUISITOS”.

Você vai trabalhar em dupla com um colega, portanto, compartilhe essas pastas com ele, dando permissão para edição.

Siga as instruções do professor para realizar essas atividades.

# GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Faça o login para ver se está tudo Ok.

The screenshot shows a GitHub profile page with the following details:

- Profile Picture:** A black cat icon.
- Name:** GitHub
- Slogan:** How people build software.
- Location:** San Francisco, CA
- Website:** <https://github.com/about>
- Email:** support@github.com
- Verified:** Verified badge.
- Repositories:** 289 (highlighted)
- People:** 206
- Projects:** 0
- Pinned repositories:**
  - fetch**: A window.fetch JavaScript polyfill. (JavaScript, 20.8k stars, 1.9k forks)
  - hub**: hub helps you win at git. (Go, 14.1k stars, 1.4k forks)
  - training-kit**: Open source on demand courses and cheat sheets for Git and GitHub. (HTML, 1.9k stars, 1.9k forks)
  - choosealicense.com**: A site to provide non-judgmental guidance on choosing a license for your open source project. (Ruby, 1.4k stars, 457 forks)
  - scientist**: A Ruby library for carefully refactoring critical paths. (Ruby, 4.7k stars, 200 forks)
  - gh-ost**: GitHub's Online Schema Migrations for MySQL. (Go, 5.9k stars, 453 forks)

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Logado no GITHUB, acesse a área de repositórios.

The screenshot shows the GitHub user interface. At the top, there's a dark header bar with the GitHub logo, a search bar, and navigation links for Pull requests, Issues, Marketplace, and Explore. On the right side of the header, there's a user dropdown menu. A red arrow points from the text below to the 'Your repositories' option in this menu, which is highlighted with a red box. The main content area shows the GitHub logo, the tagline 'How people build software.', and basic account information like location (San Francisco, CA), website (https://github.com/about), and email (support@github.com). Below this, there are three pinned repository cards: 'fetch', 'hub', and 'training-kit'. The bottom of the page features a pink footer bar with the URL 'www.fiap.com.br' on the left and the professor's contact information 'Prof. Renato Jardim Parducci – profrenato.parducci@fiap.com.br' on the right.

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Seus repositórios de projetos de software aparecerão, caso você já tenha catalogado algum.

The screenshot shows a GitHub profile interface. At the top, there is a blue banner with the text "ProTip! Updating your profile with your name, location, and a profile picture helps other GitHub users get to know you." and a green "Edit profile" button. Below the banner, the profile navigation bar includes "Overview", "Repositories 12", "Stars 0", "Followers 0", and "Following 0". There are also search and filter options: "Find a repository...", "Type: All ▾", "Language: All ▾", and a green "New" button. The main content area displays three repository cards: "1TDSJ" (updated on 26 Apr 2017), "RepExemplo" (updated on 17 Jan 2017), and "ExemploTDSS". Each card has a small green horizontal bar to its right.

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Seus repositórios de projetos de software aparecerão, caso você já tenha catalogado algum.

The screenshot shows a GitHub profile interface. At the top, there is a blue banner with the text "ProTip! Updating your profile with your name, location, and a profile picture helps other GitHub users get to know you." and a green "Edit profile" button. Below the banner, the profile navigation bar includes "Overview", "Repositories 12", "Stars 0", "Followers 0", and "Following 0". There is also a search bar with placeholder text "Find a repository...", and dropdown menus for "Type: All" and "Language: All", along with a green "New" button. The main content area displays three repository cards: "1TDSJ" (updated on 26 Apr 2017), "RepExemplo" (updated on 17 Jan 2017), and "ExemploTDSS". Each card has a small green horizontal bar to its right.

# GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Crie um repositório.

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner  / Repository name  

Great repository names are short and memorable. Need inspiration? How about sturdy-spoon.

Description (optional)

 Public  
Anyone can see this repository. You choose who can commit.

 Private  
You choose who can see and commit to this repository.

Initialize this repository with a README  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None | 

**Create repository**

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Entenda a tela do GitHub.

The screenshot shows a GitHub repository page for 'RenatoJardimParducci / TesteGITHUB'. The page includes navigation links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. A message states 'No description, website, or topics provided.' with an 'Edit' button. Below this, there's a 'Manage topics' section and summary statistics: 1 commit, 1 branch, 0 releases, and 1 contributor. A red box highlights the 'Branch: master ▾' dropdown and the 'New pull request' button. Another red box highlights the 'README.md' file entry in the commit list. A third red box highlights the 'Upload files' button. Red numbers 1, 2, and 3 with arrows point to these three features respectively. The commit list shows an 'Initial commit' by 'RenatoJardimParducci' at 'a minute ago'.

1 Seleciona a área/cópia de fontes para trabalho

2 Nomes dos arquivos que constam na área

3 Usado para carregar arquivos fonte no GIT

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

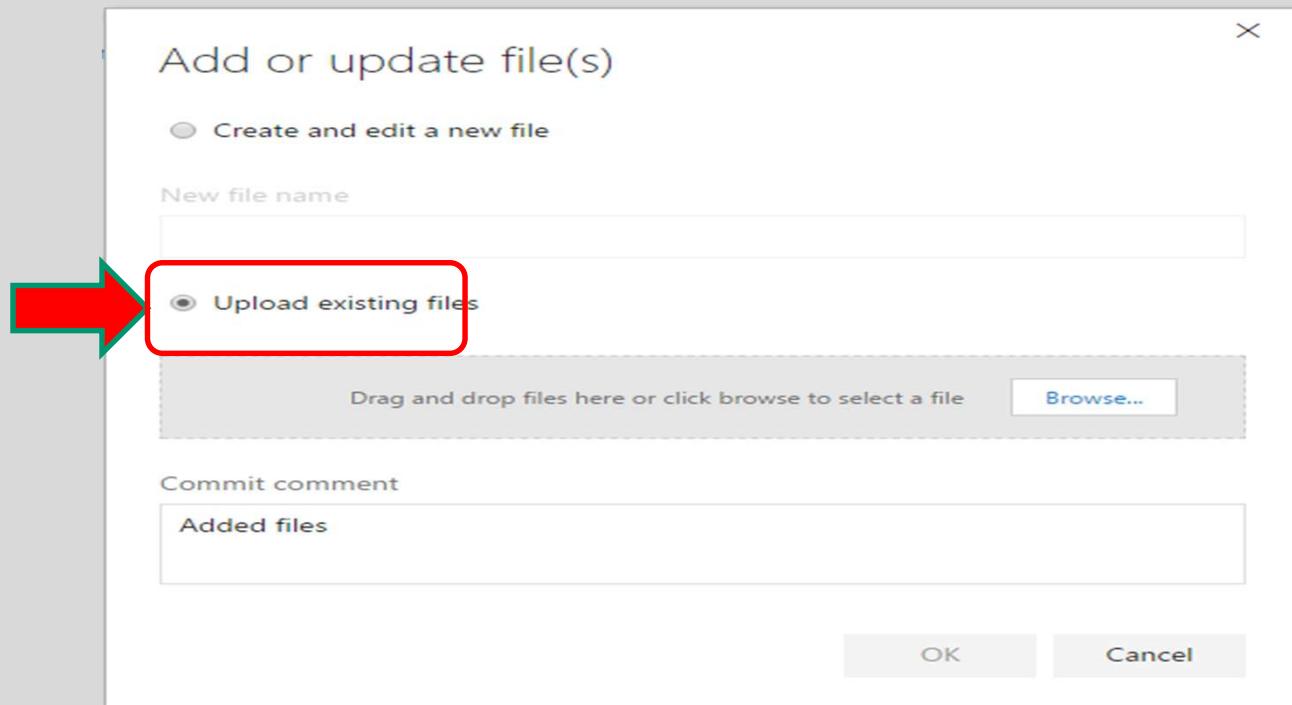
Você pode criar pastas para separar tipos de arquivos

The screenshot shows a GitHub repository page for 'RenatoJardimParducci / TesteGITHUB'. The page includes navigation links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. A red callout box with a green border and white text contains the instruction: 'Crie um nome para a pasta/nome de um arquivo Readme que será criado'. A red arrow points from this box to the 'Create new file' button in the top right corner of the repository stats section. The stats section also shows 1 commit, 1 branch, 0 releases, and 1 contributor. Below the stats, there are buttons for Branch: master, New pull request, Create new file (which is highlighted with a red box), Upload files, Find file, and Clone or download. The repository history shows an initial commit by RenatoJardimParducci. The bottom of the page features a large title 'TesteGITHUB'.

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Acesse o link de Upload para subir para o GitHub um arquivo do seu computador.

Suba um arquivo .JAVA ou .SQL para experimentar!



Como alternativa,  
você pode abrir a  
pasta com o seu  
arquivo no  
Windows  
Explorer e  
arrastá-lo para a  
página do  
GITHUB.

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Na página principal do projeto, você consegue pesquisar pelo e-mail ou ID de login GIT dos seus colegas de equipe e compartilhar o projeto com eles! O GIT vai criar uma nova versão do seu arquivo, toda vez que você atualizá-lo!

Para editar e alterar um arquivo, faça o Download, edite do seu PC e depois, faça o Upload na área do GIT onde o arquivo antigo está! O versionamento é automaticamente feito pelo GIT!

The screenshot shows a GitHub repository settings page. At the top, there is a navigation bar with links: Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security, Insights, and Settings. A red box highlights the 'Settings' link, which is also pointed to by a green arrow. On the left, a sidebar lists options: Manage access (highlighted with a red box and green arrow), Branches, Webhooks, Notifications, Integrations & services, Deploy keys, and Secrets. The main content area is titled 'Who has access'. It shows two sections: 'PUBLIC REPOSITORY' (This repository is public and visible to anyone) and 'DIRECT ACCESS' (0 collaborators have access to this repository. Only you can contribute to this repository). Both sections have a 'Manage' button. A red box highlights the 'Manage access' button in the sidebar, and a green arrow points to it. Another red box highlights the 'Manage access' button in the main content area, and a green arrow points to it.

ASSISTA OS VÍDEOS NO CANAL DO PROFESSOR



<https://youtu.be/MYhIM0bk9aQ>

A screenshot of a video player interface. The video content shows a man speaking about GitHub. The video player has a progress bar at 0:30 / 13:49. Below the video, there is a callout box with the text "Discover interesting projects and people to populate your personal news feed." On the right side of the video player, there is a message about a new graph feature in GitHub. The video player is set against a background of a GitHub landing page.



Conteúdo didático complementar - GitHub

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Existem algumas outras formas de você acessar e operar o conteúdo do Git Hub a partir do seu PC:

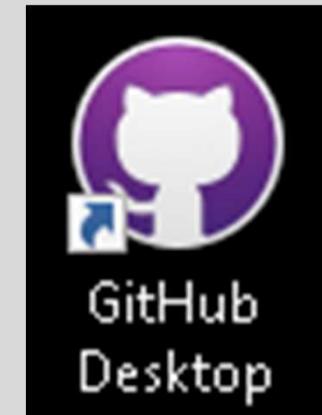
- GIT Gui (por menus em um software cliente);
- GIT CMD (por linha de comando).

Não exploraremos essas modalidades nas aulas de Data Governance.  
Vamos focar em compreender e usar bem o GIT HUB.

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Outro aplicativo muito interessante é o GitHub Desktop que pode ser obtido no próprio portal do GITHUB.

Esse recurso é usado quando você quer editar arquivos no seu computador e não no HUB, porém, o GIT HUB não mostra a opção de Download!



Daqui por diante, vamos usar o GIT DESKTOP, sempre criando uma Branch para cada manutenção que fizermos no repositório e refletindo essa Branch na Master, só depois que os fontes estiverem validados (pós edição e testes).

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Acesso ao GITDesktop para editar arquivos no PC com editores originais de código.

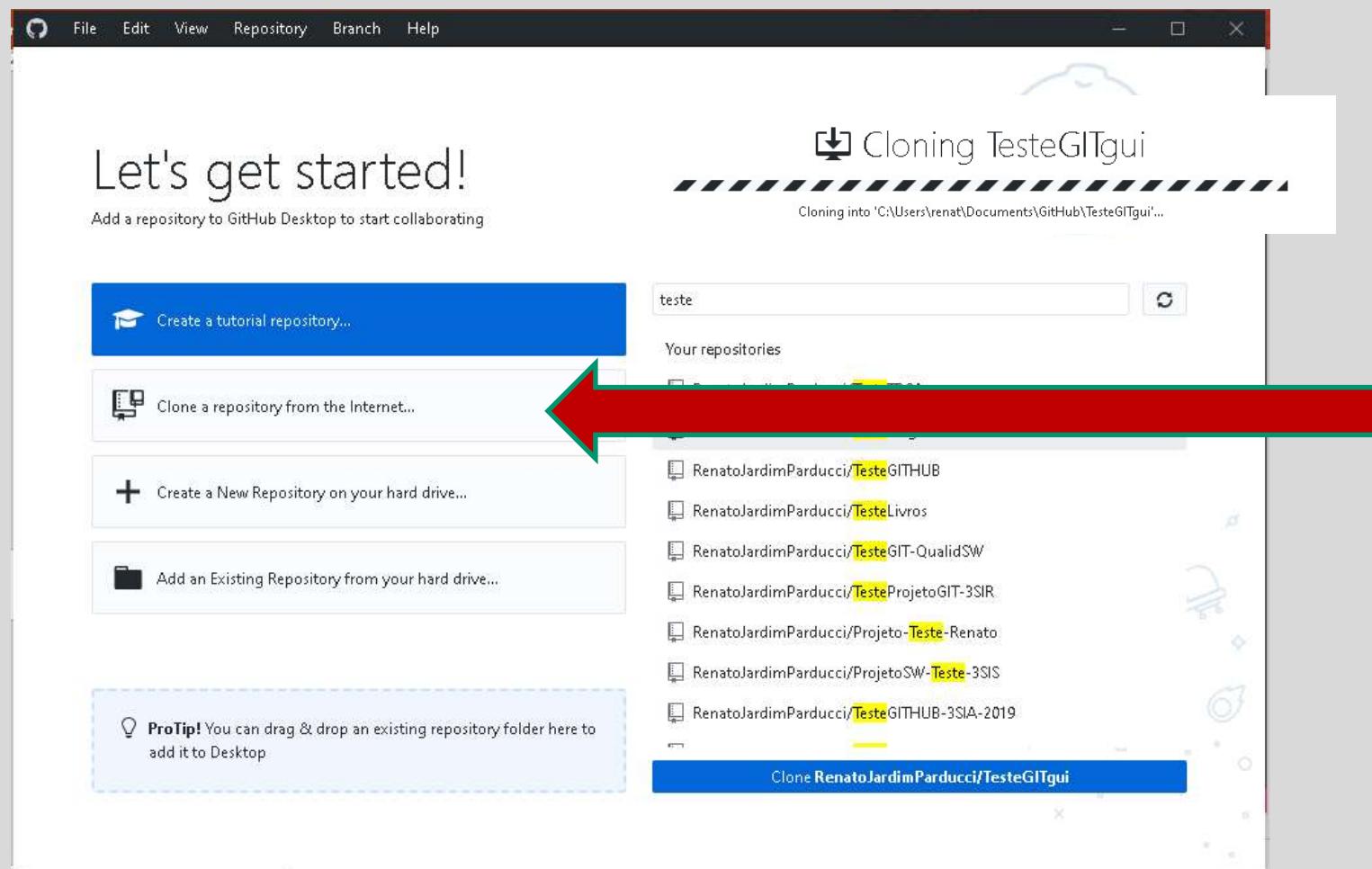
The screenshot shows a GitHub repository page for 'TesteGITHUB / Aluno.java'. The top navigation bar includes links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. Below the navigation, it shows the Branch: master dropdown, the file path TesteGITHUB / Aluno.java, and buttons for Find file and Copy path. The commit history for the file shows a single commit by RenatoJardimParducci with the message 'Update Aluno.java', timestamped e92ad94 9 minutes ago. The commit details show 1 contributor. The code editor displays the following Java code:

```
1  public class Aluno1 extends Pessoa {  
2  
3      private Matricula matricula;  
4  
5      public void SolicitarMatricula() {  
6          }  
7      }  
8  
9  }
```

The file toolbar at the bottom right of the code editor includes Raw, Blame, History, a monitor icon, and a trash bin icon. A red box highlights the History button, and a red arrow points from this box to a red callout box containing the text 'Acesso ao GIT Desktop'.

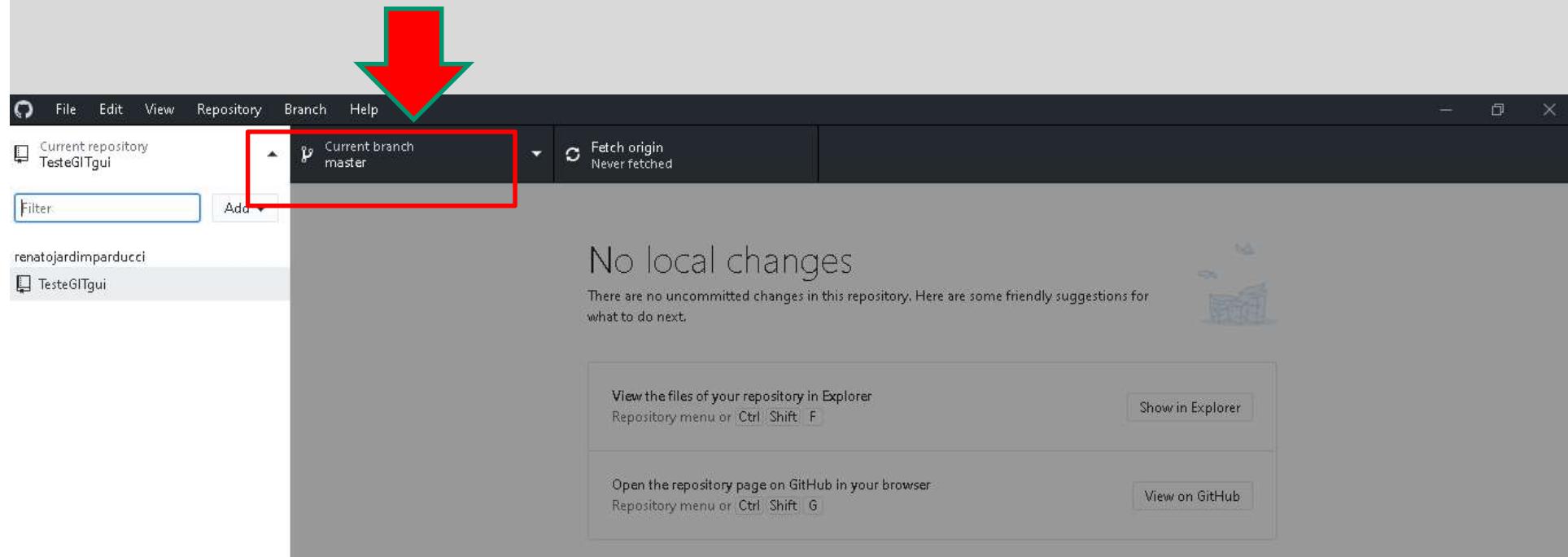
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode sincronizar os repositórios em nuvem com o computador pessoal.



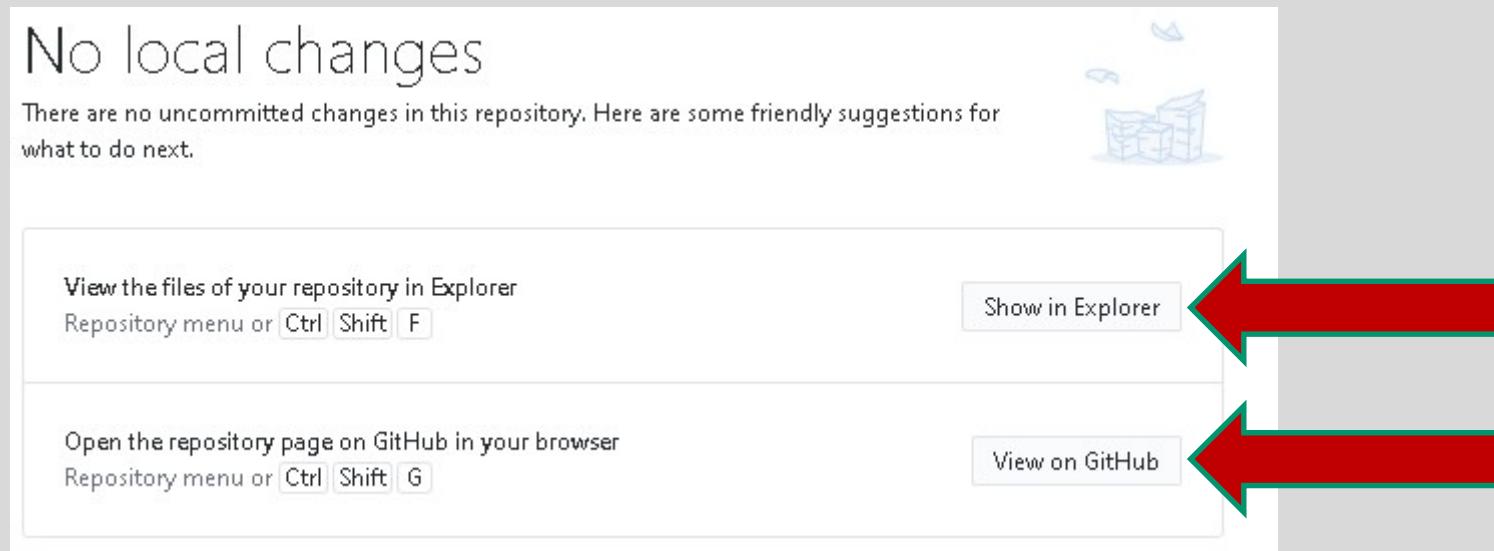
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Com o GitGUI, por exemplo, você pode acessar as áreas Master, Branches e pastas pelos menus o seu PC..



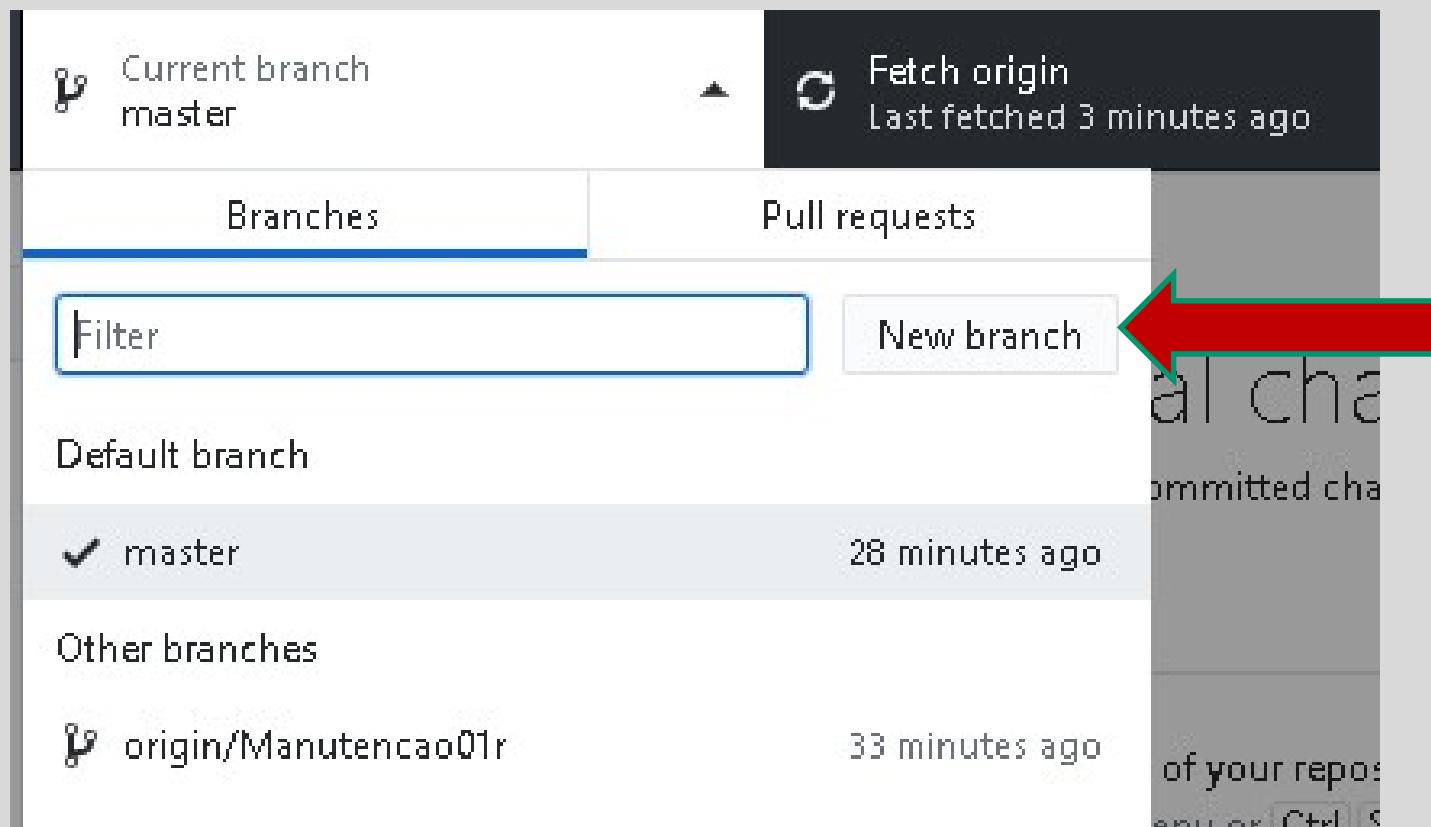
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Você pode acessar o clone local ou a pasta remota do GITHUB por esse software cliente.



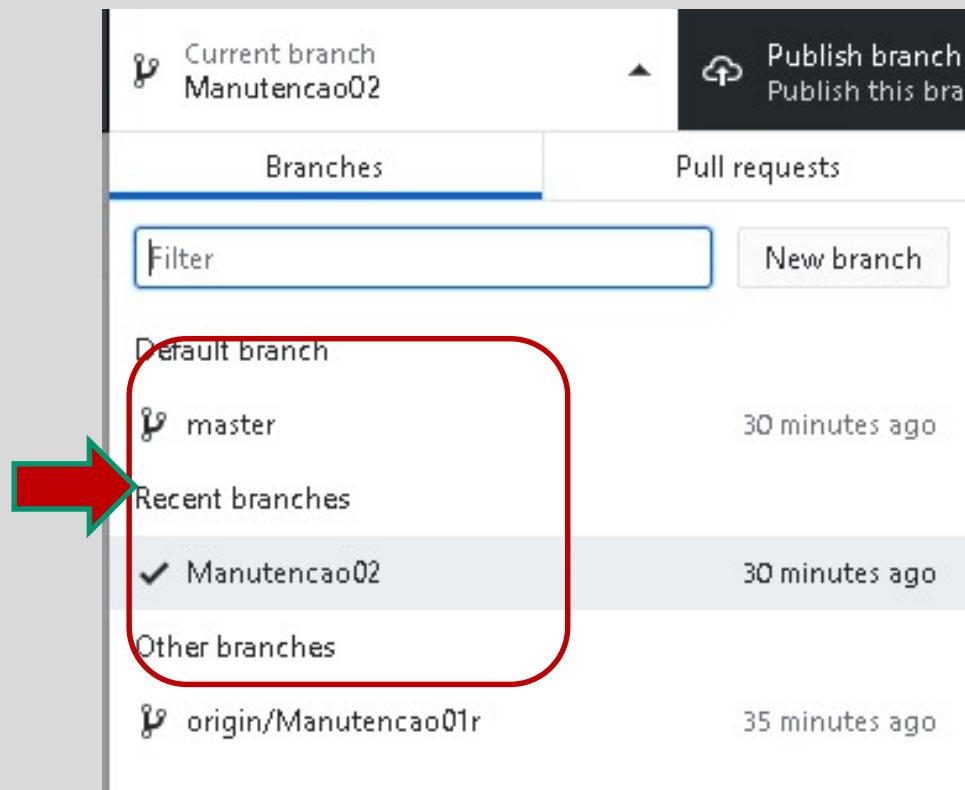
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Pode ainda criar branch...



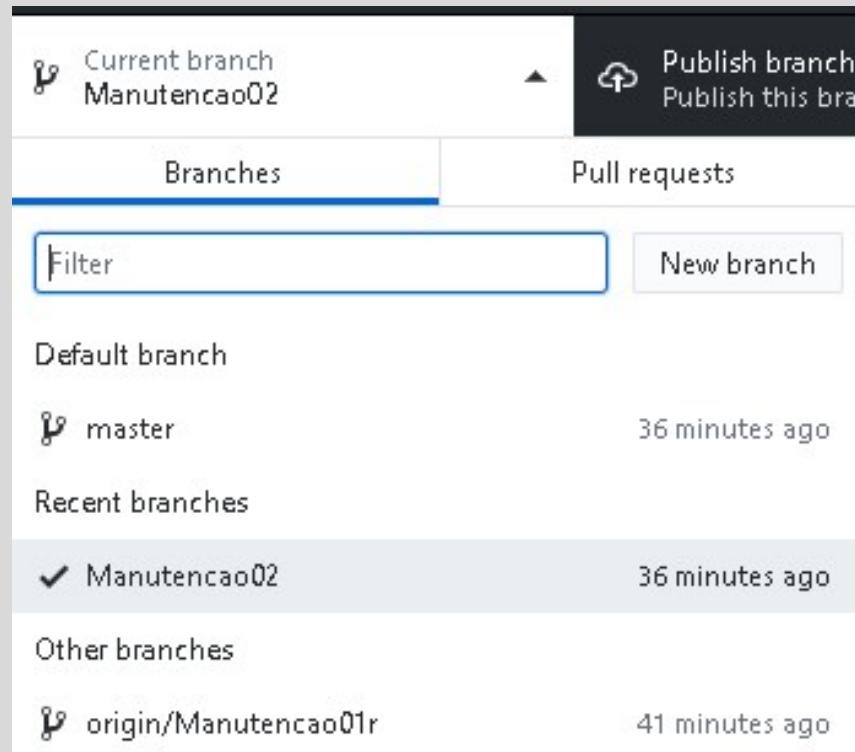
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

E essa Branch quando criada localmente, pode ser visualizada no GitHub client.



## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

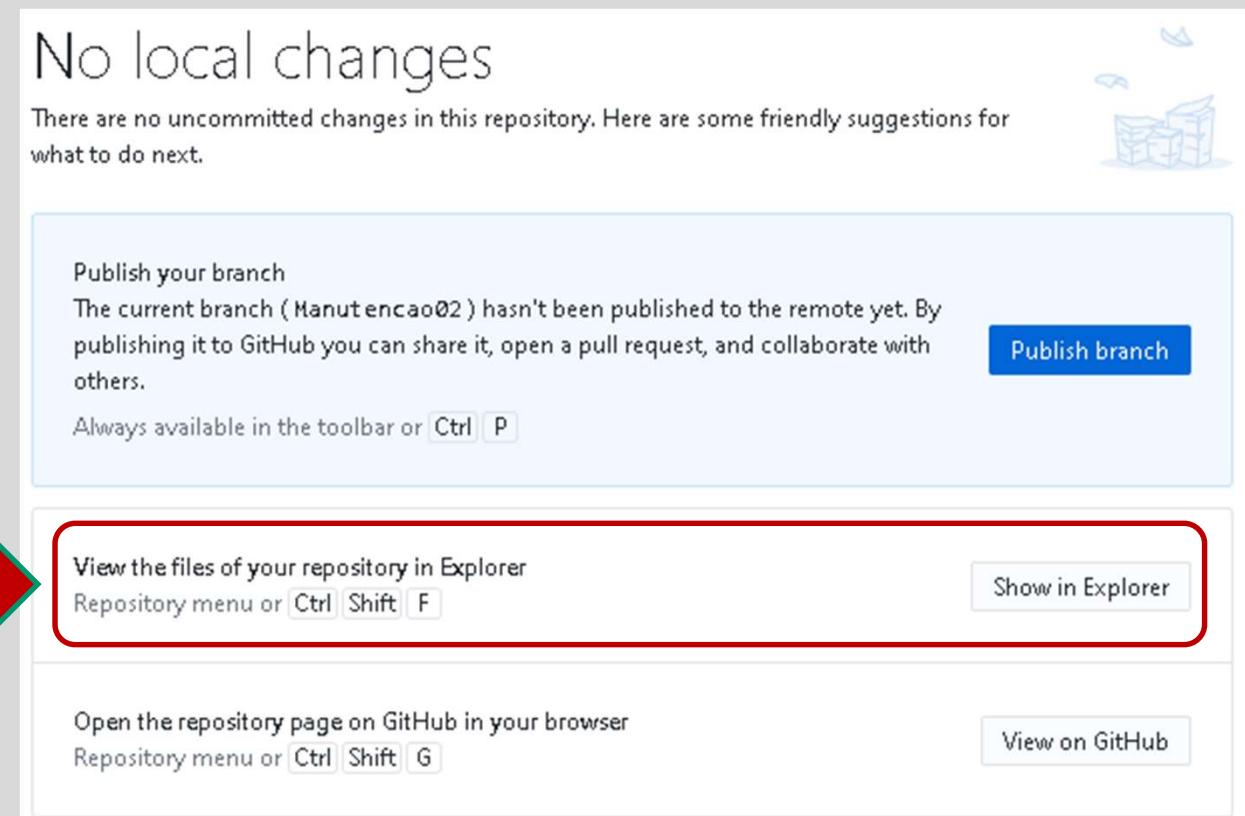
Para editar arquivos na Branch, faça um Checkout!



Selecione a branch no menu de navegação da ferramenta!

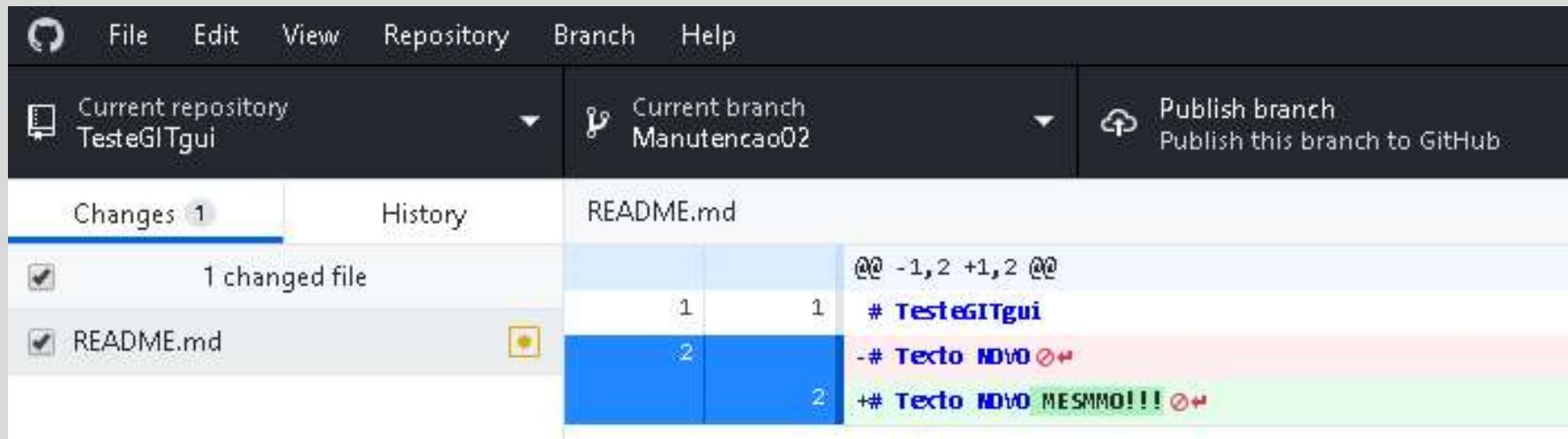
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Após criada a Branch, a navegação de acesso aos fontes é via menu da aplicação GUI.



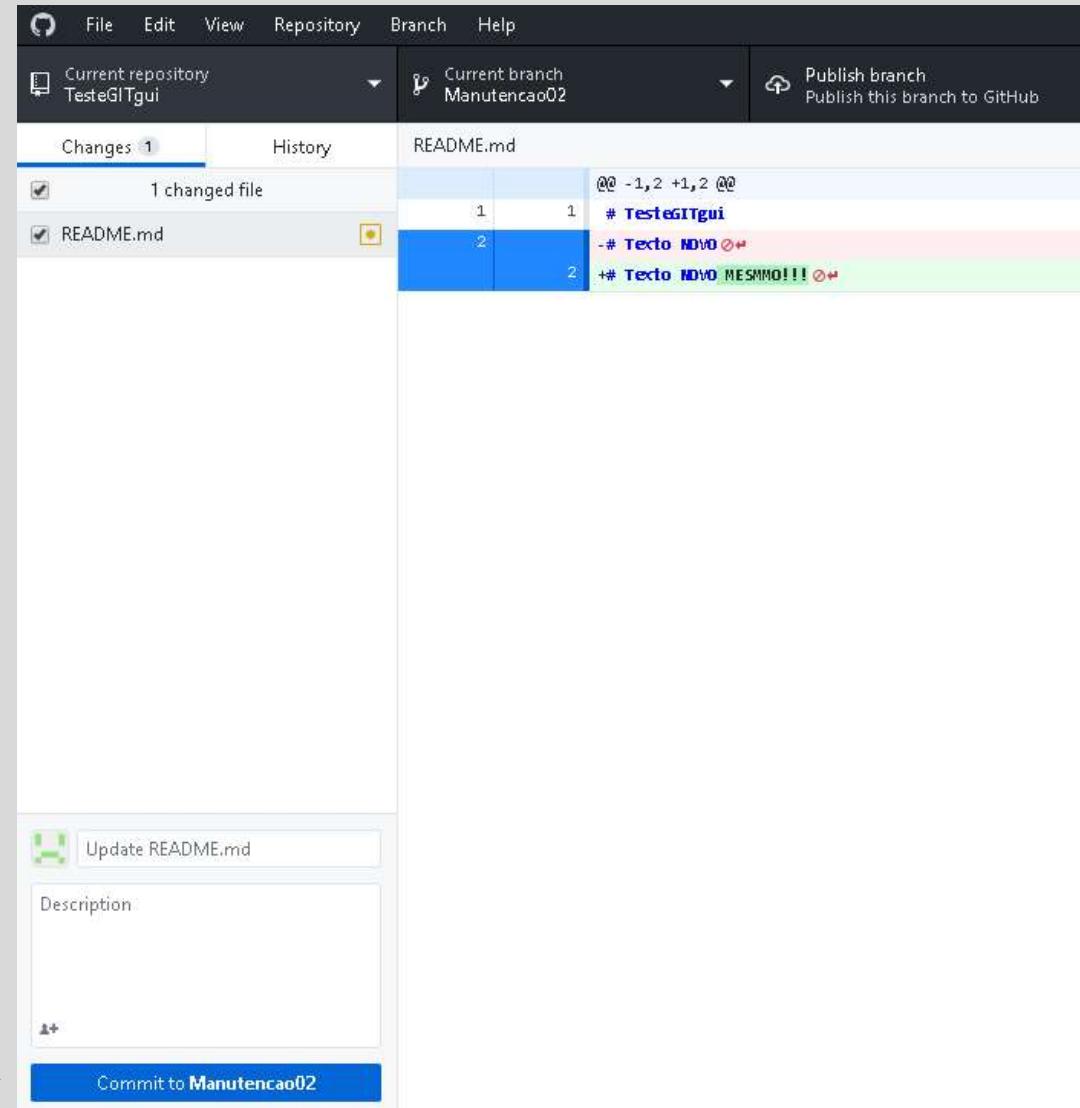
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Assim que alterado o arquivo ou criado um arquivo novo na pasta do GIT local, o GitHub Client vai mostrar as diferenças entre a Master e a Branch.



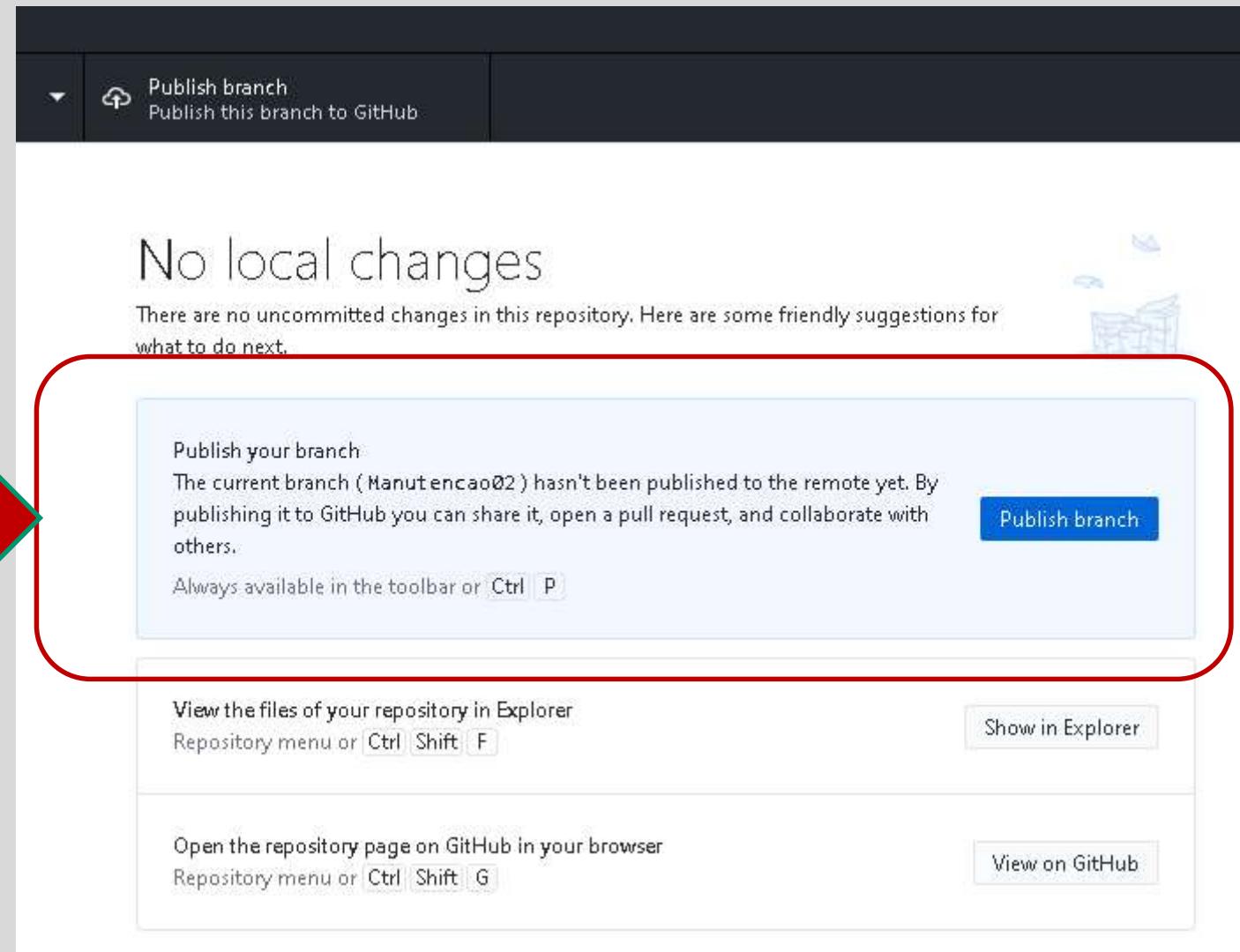
## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Se você desejar publicar a Branch criada no seu PC no GitHub.com, primeiro, faça um Commit no Git local



## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

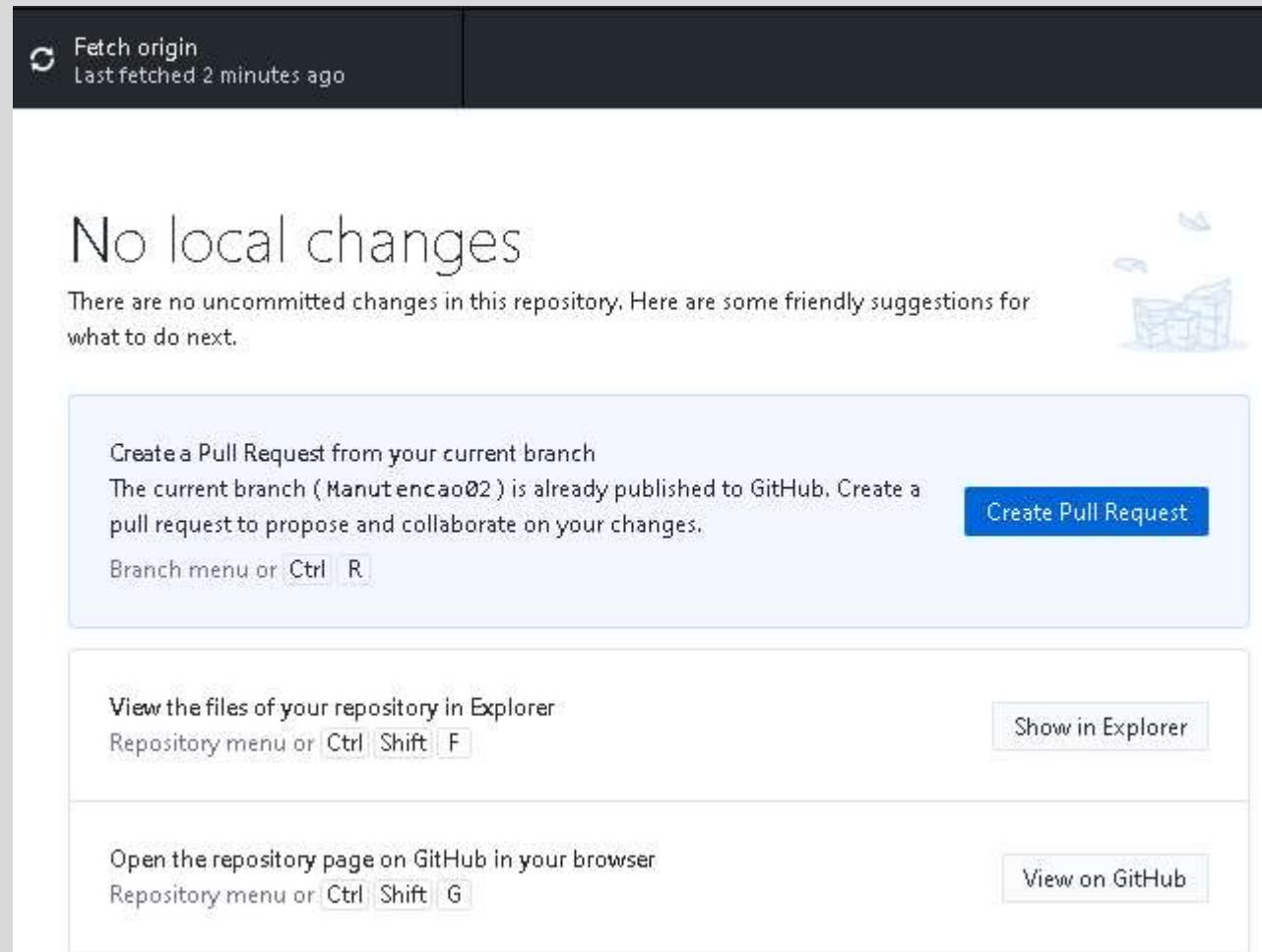
Faça o PUSH no GITHUB remoto (WEB).



## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

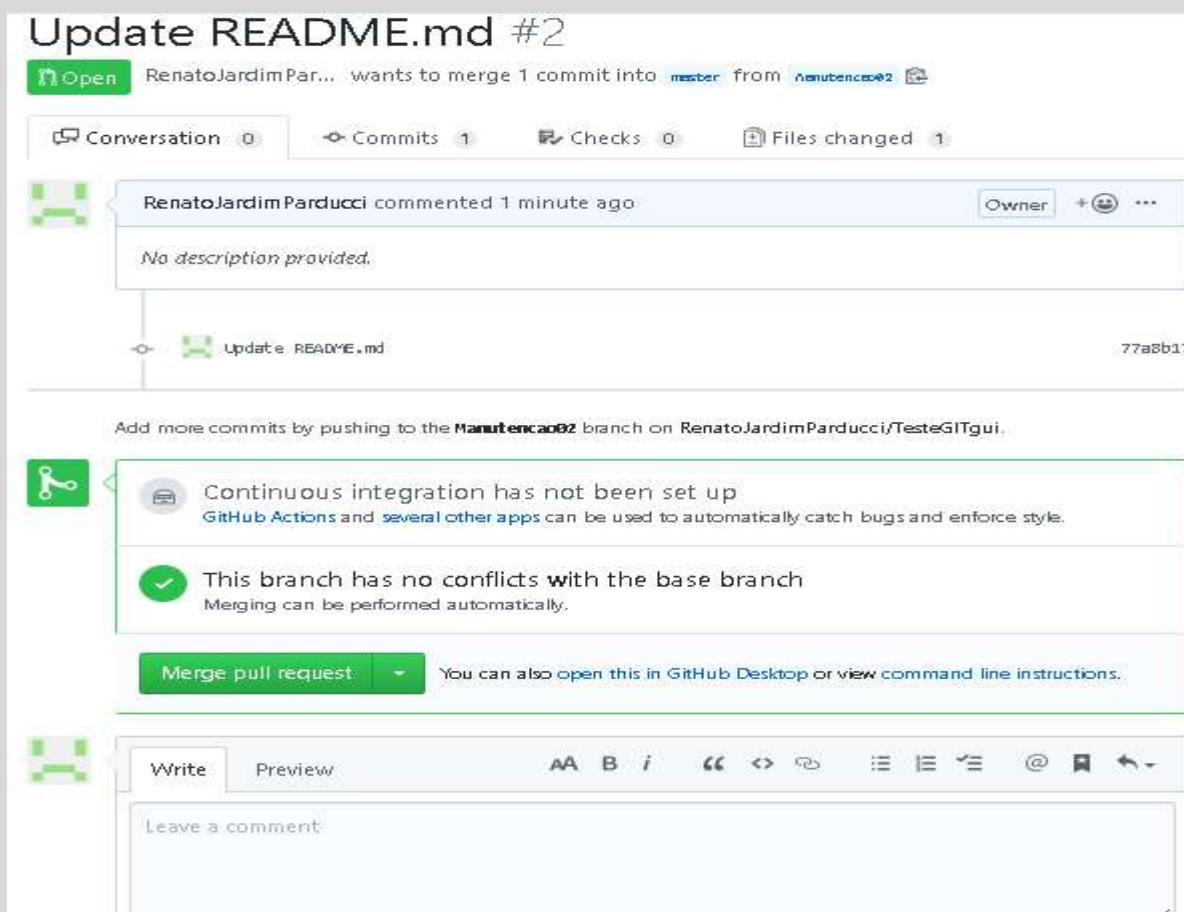
As edições são refletidas do GitGui para o GitHub!

Crie a Pull Request remota via GITHUB Client para atualizar a Master remota.



## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

Ao solicitar a Pull Request, o Git Hub é acionado. Você fará Login e confirmará o Merge/Pull Request!



## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

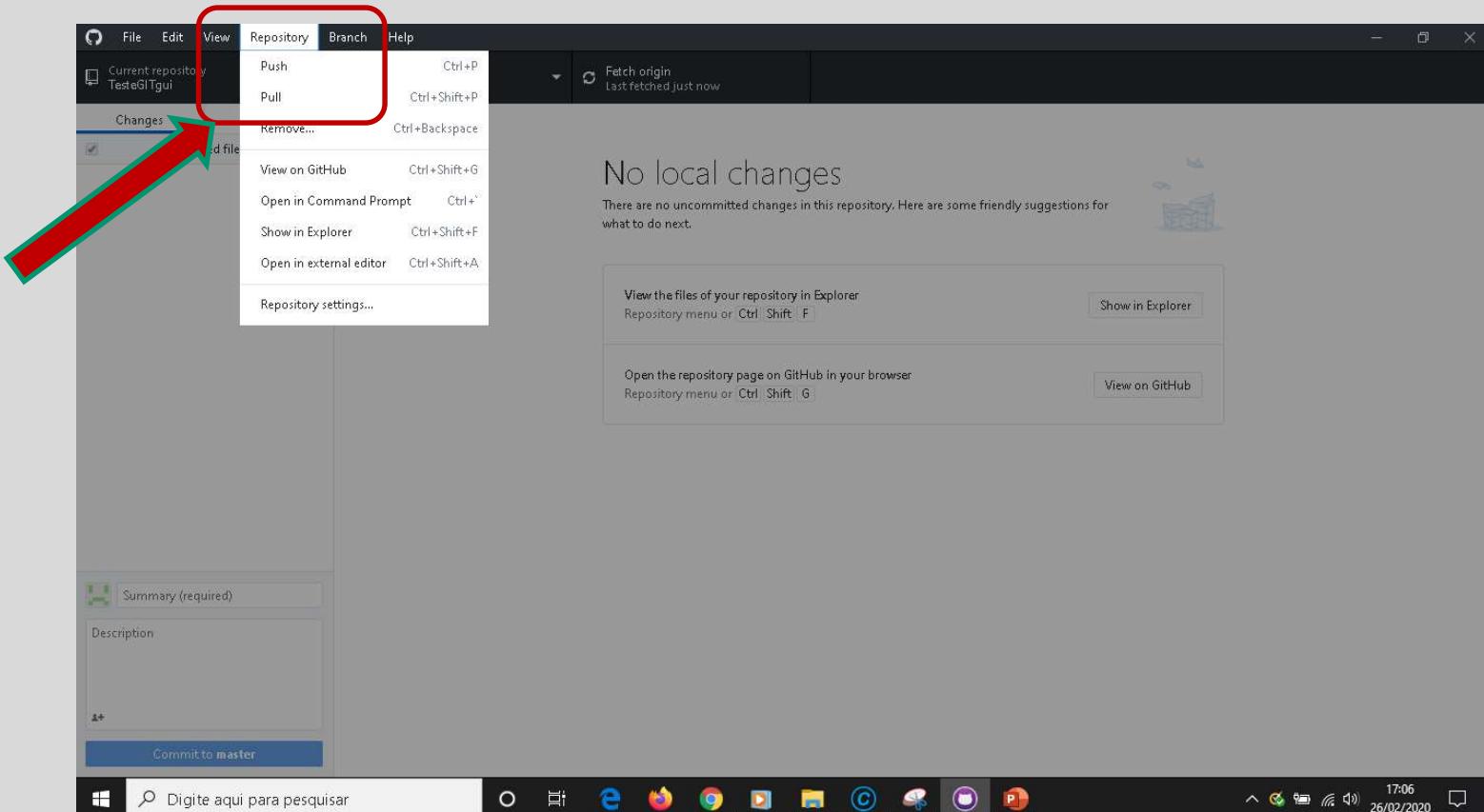
Com esses procedimentos, você consegue fazer Check out e Check in (Push) , mantendo sincronia entre o ambiente remoto dos fontes e o seu ambiente local de trabalho.

Após os Pushes, confira sempre se o GitHub.com foi atualizado corretamente.

The screenshot shows a GitHub repository interface. At the top, there are statistics: 5 commits, 3 branches, 0 packages, 0 releases, and 1 contributor. Below this, a pull request from RenatoJardimParducci is shown, merged into the master branch. The latest commit was made 1 minute ago. A file named README.md has been updated 9 minutes ago. The content of README.md is displayed below, showing the text "TesteGITgui" and "Texto NOVO MESMMO!!!".

## GERENCIAMENTO DA PRODUÇÃO E ATUALIZAÇÃO DE SOFTWARE

**IMPORTANTE:** o fato de você ter atualizado a Branch e Master no GITHUB não implica que a Master Local tenha sido atualizada.  
Para atualizar a Master Local, faça um Pull do Repositório.



ASSISTA OS VÍDEOS NO CANAL DO PROFESSOR



[https://youtu.be/ps\\_ZTFrasAg](https://youtu.be/ps_ZTFrasAg)

A screenshot showing two side-by-side windows. On the left is a GitHub repository page for 'RenatoJardimParducci / ExemploGitDesktop'. It shows 1 commit, 1 branch, 0 packages, and 0 releases. The README.md file contains the text 'Initial commit'. On the right is a video player window showing a close-up of a man's face, identified as Prof. Renato Jardim Parducci. Below the video player is a large YouTube logo.

## ENGENHARIA DE REQUISITOS

### As principais técnicas de levantamento de dados



-Pesquisa de referências;



-Brainstorm e mapa mental;



-Entrevista individual;



-Reunião coletiva, Workshop de trabalho e JAD;



-Observação de campo;



-Questionário.

## ENGENHARIA DE REQUISITOS

Brainstorm



Vamos iniciar por analisar um desafio e trabalhar com tempestade de ideias e mapas mentais para avaliar possíveis alternativas de solução!

## ENGENHARIA DE REQUISITOS

## Brainstorm e mapa mental



## Regras

- \* Todos são livres para propor ideias;
- \* Quanto mais ideias, melhor;
- \* Não bloqueie ideias dos outros;
- \* Procure construir novas ideias sobre as que foram colocadas;
- \* Associe as ideias entre si, compondo grupos de propostas.

## ENGENHARIA DE REQUISITOS

Brainstorm e mapa mental



### Mapa mental

\*Associa as ideias, formando uma cadeia de ações que implementam uma solução completa para um problema

## ENGENHARIA DE REQUISITOS

Brainstorm:



Prós:

- Grande geração de propostas
- Criatividade livre



Contras:

- Muitas ideias podem ser pouco pragmáticas
- A falta de conflito e discussão bloqueia a maturação

## APRENDENDO NA PRÁTICA



Um estacionamento do centro da cidade de São Paulo está com problemas para controlar a entrada de saída de veículos.

Clientes reclamam de portas amassadas e riscos na pintura, ocasionados pelo desrespeito às vagas de marcadas, por cauda de superlotação.

Os manobristas que são em 10, não sabem ao certo se cabe mais algum carro em algum canto do estacionamento ou não.

O proprietário desconfia que veículos estão ocupando vagas e saindo sem pagar ou pagando de forma diferente da permanência realizada.

Hoje, o controle do estacionamento é todo manual e os recebimentos são em espécie (dinheiro vivo).

Agrupados em duplas ou trios, apliquem Brainstorm e listem ideias para resolver esses problemas.

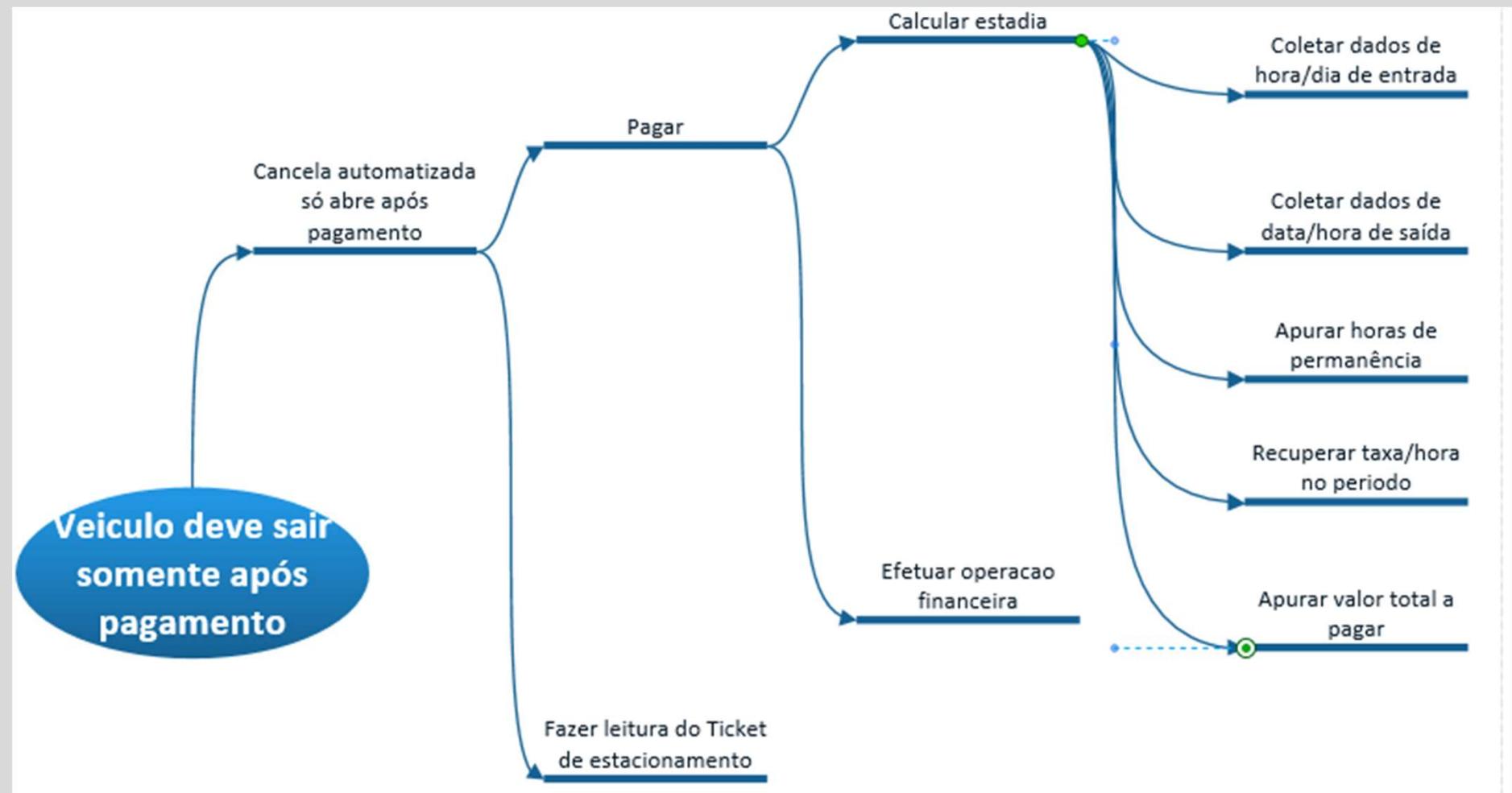
Vocês têm 15 minutos.

Anote tudo em um edito de textos e guarde o documento no GITHUB, com o nome Brainstorm.



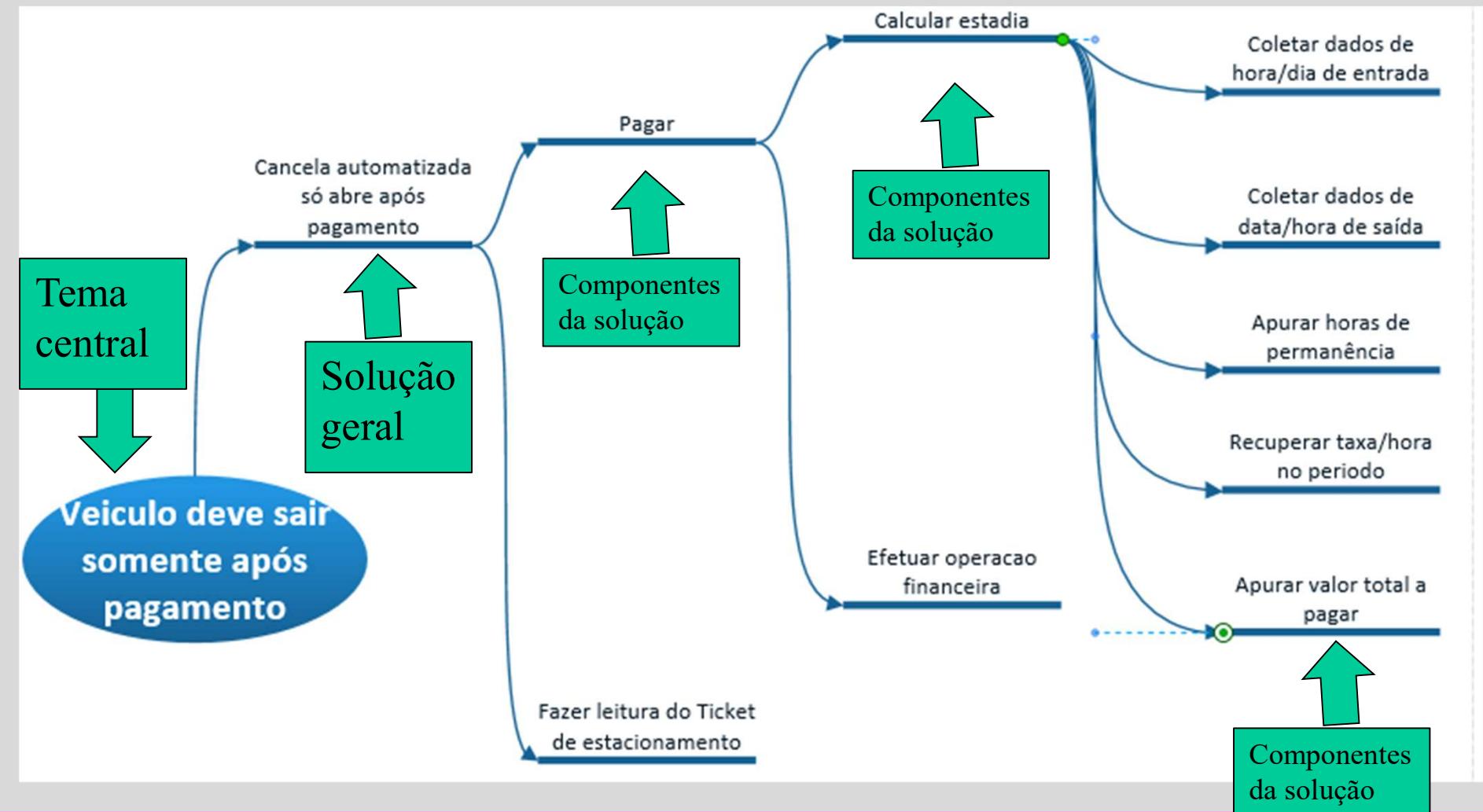
## ENGENHARIA DE REQUISITOS

## Exemplo de Mapa mental



## ENGENHARIA DE REQUISITOS

## Exemplo de Mapa mental



## APRENDENDO NA PRÁTICA



Um estacionamento do centro da cidade de São Paulo está com problemas para controlar a entrada de saída de veículos.

Clientes reclamam de portas amassadas e riscos na pintura, ocasionados pelo desrespeito às vagas de marcadas, por cauda de superlotação.

Os manobristas que são em 10, não sabem ao certo se cabe mais algum carro em algum canto do estacionamento ou não.

O proprietário desconfia que veículos estão ocupando vagas e saindo sem pagar ou pagando de forma diferente da permanência realizada.

Hoje, o controle do estacionamento é todo manual e os recebimentos são em espécie (dinheiro vivo).

**FAÇA AGORA, O MAPA MENTAL!**

Vocês têm 15 minutos.

**USE O VISIO PARA DESENHAR!**

Guarde o documento no GITHUB com o nome MapaMental



## ENGENHARIA DE REQUISITOS

## Pesquisa de referências



Busca de informações em bibliografia, através de estudos de caso publicados em qualquer mídia, ou conversas com especialistas que permitam o entendimento mais amplo de termos técnicos e do funcionamento da área de negócio que é alvo do levantamento.

Gera conhecimento que subsidia as atividades do analista de negócio e de sistemas e o prepara para aplicar outras técnicas complementares de identificação de requisitos do software.



## ENGENHARIA DE REQUISITOS

Na busca de referência:



### Prós:

- Permite assimilação rápida de um contexto grande de informações.
- Recheada de opiniões, pareceres e dados estatísticos que sustentam defesas de pontos de vista sobre o assunto



### Contras:

- É fonte secundária de informações (não permite acesso a quem realiza o trabalho que é foco da investigação de requisitos).
- Podem ser encontradas informações conflitantes, falsas ou imprecisas, demandando cruzamento de pesquisas para confirmar a qualidade da informação.

## APRENDENDO NA PRÁTICA



Mantendo o seu grupo de trabalho anterior...

Pesquise sobre produto para automação e controle de estacionamentos.

Use a Internet/WEB para pesquisar e analise os recursos disponíveis em softwares de informatização de estacionamentos.

Liste os recursos desses softwares.

Vocês têm 15 minutos.

Em seguida, vamos construir uma lista geral dos recursos possíveis de serem incorporados em uma solução de gestão de estacionamento.

Anote tudo livremente em um editor de textos.

Guarde o documento no GITHUB com o nome Pesquisa



## DINÂMICAS DE LEVANTAMENTO DE REQUISITOS DE SOFTWARE

### Entrevista Individual



Ocorre por agendamento individual com um entrevistado.

O entrevistado deve ser comunicado do objetivo da entrevista e se possível, receber informações sobre o conteúdo que será explorado para que ele se prepare da melhor maneira para receber o entrevistador e contribuir com suas opiniões.

O entrevistador deve estar preparado para direcionar perguntas exploratórias sobre o tema investigado.



A conversa se dá de forma livre.

Na entrevista individual:



Prós:

- Exploração calma do tema com o entrevistado (maior aproveitamento dos conhecimentos);
- Captura daquilo que é realmente importante para o entrevistado (suas prioridades, preocupações e elogios);
- Uso e observação da linguagem corporal.



Contras:

- Gera como resultado uma visão parcial sobre o tema;
- Registro do conteúdo da entrevista é difícil (ideal que ocorram anotações em um documento de entendimentos);
- A tabulação de resultados é difícil;
- Não esgota o entendimento de um tema de pesquisa;
- Exige muita preparação do entrevistador que não contará com o apoio de outros colegas.

## ATIVIDADE PRÁTICA



Mantendo o seu grupo de trabalho anterior...

Uma pessoa do grupo será entrevistada para explicar quais as suas expectativas como consumidor de serviços de um estacionamento.

Elabore uma lista de perguntas de forma a dirigir a conversa de forma objetiva, esclarecendo o que você pretende confirmar como requisito do software.

Vocês têm 15 minutos para tentar complementar a lista que montamos com novas ideias.

Anote tudo livremente em um editor de textos (perguntas e respostas).  
Suba o documento no GITHUB com o nome Entrevista.



## DINÂMICAS DE LEVANTAMENTO DE REQUISITOS DE SOFTWARE

### Observação da rotina



Baseado na visitação de espaços de trabalho para observação e coleta de documentos que evidenciem resultados dos trabalhos realizados.

Em geral, é complementado pela elaboração por parte dos analistas de fluxogramas e documentos que explicam o entendimento das operações da empresa.



Devem ser feitos registros sobre o volume de trabalho, momentos de pico e vale de demanda e opiniões de colaboradores da rotina observada.

## ATIVIDADE PRÁTICA



O seguinte vídeo foi gravado em visitas que você fez a estacionamentos para obter mais ideias para o seu projeto:

<https://youtu.be/QJjsvOF6enQ>

Com base nessas observações, quais outros requisitos o seu grupo incluiria na lista?  
Vocês têm 10 minutos para discutir e listar.

Em seguida, vamos criar um documento com o resultado das observações, chamado Observação e subir esse documento no GITHUB.

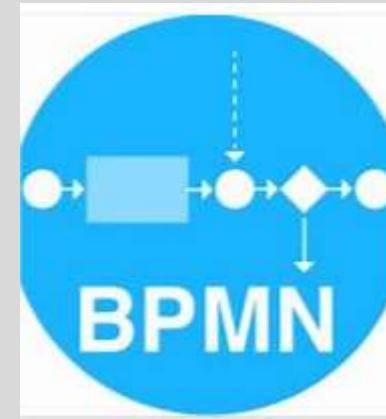


## DINÂMICAS DE LEVANTAMENTO DE REQUISITOS DE SOFTWARE

### Observação da rotina



Vamos estudar o padrão BPMN para usarmos na nossa documentação de observação de campo!



Usaremos o software



nessa tarefa!

## BPMN – PRINCIPAIS SÍMBOLOS



Atividade



Evento Inicial



Evento Final



Fluxo



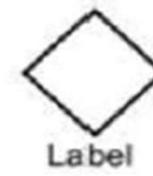
Mensagem



Lane / Pool



Evento Intermediário



Gateway Exclusivo  
*(pode receber X dentro do losango)*



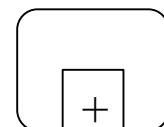
Gateway Inclusivo



Gateway Paralelo



Conector *(pode ser numerado)*



Subprocesso

## BPMN – LANES E POOL

- Lanes (ou Raias) servem para representar áreas/departamentos ou perfis de usuário;
- Lanes podem ser agrupadas em Pools para melhor entendimento (quando estiverem associadas).



■ POOL – Pizzaria Aleluia  
■ LANES – Atendimento e Cozinha

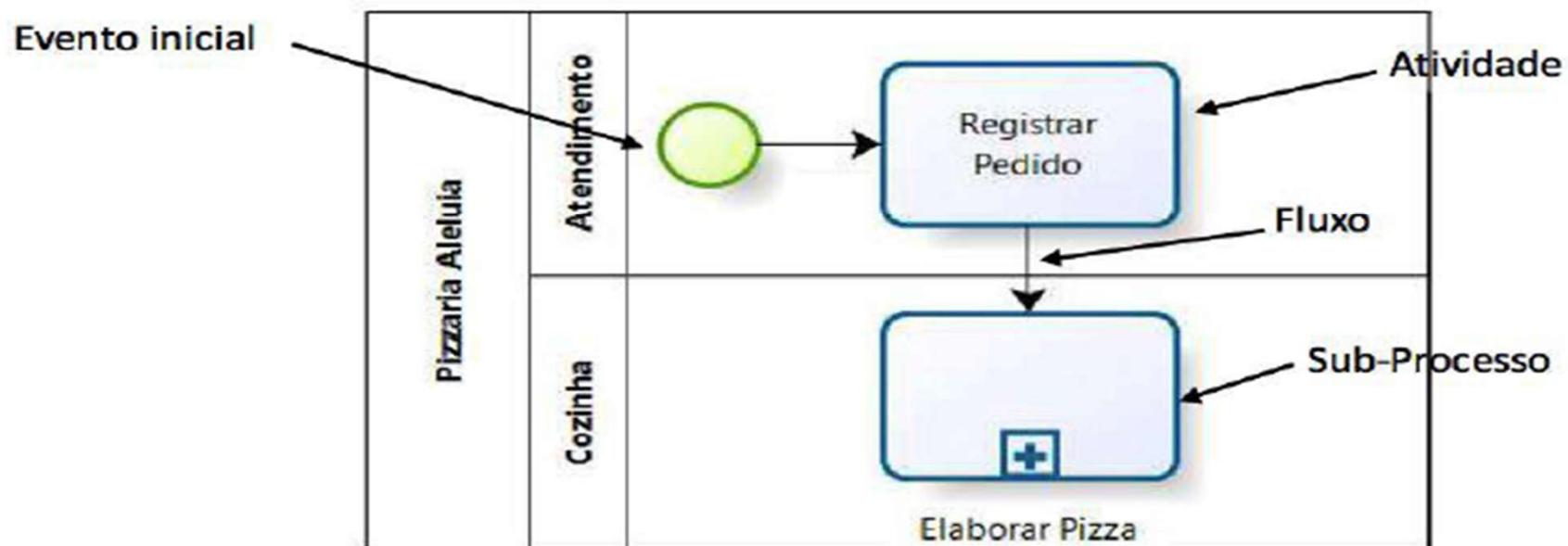
## BPMN – EVENTOS, ATIVIDADES, SUB-PROCESSOS E FLUXOS

**Eventos:** Gatilhos que representam o inicio e fim de um processo;

**Atividades:** Ações realizadas dentro do processo

**Fluxos:** Ligam eventos, atividades e gateways representando a sequencia;

**Sub-Processos:** Agrupam duas ou mais atividades, com seus fluxos, gateways e o que for necessário;

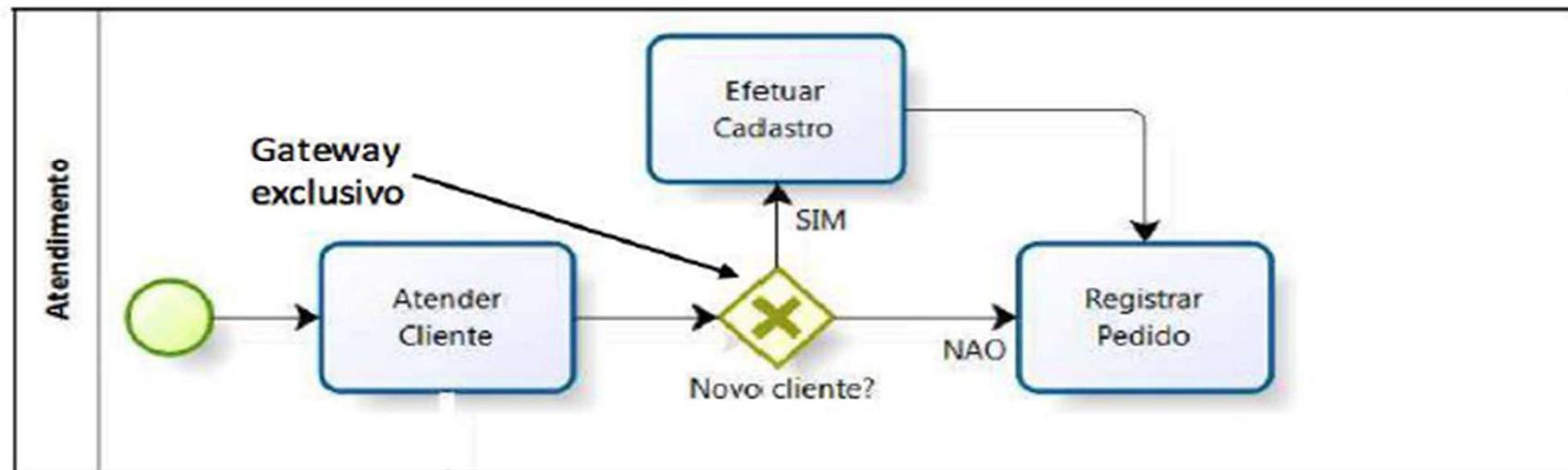


## BPMN – GATEWAYS

**Gateways:** Representam uma bifurcação → mais de um caminho possível para um fluxo.

**Tipos:**

- ✓ **Exclusivos:** Apenas um dos caminhos é possível, existe uma condição representada;
- ✓ **Inclusivos:** Representam atividades paralelas ou simultâneas.

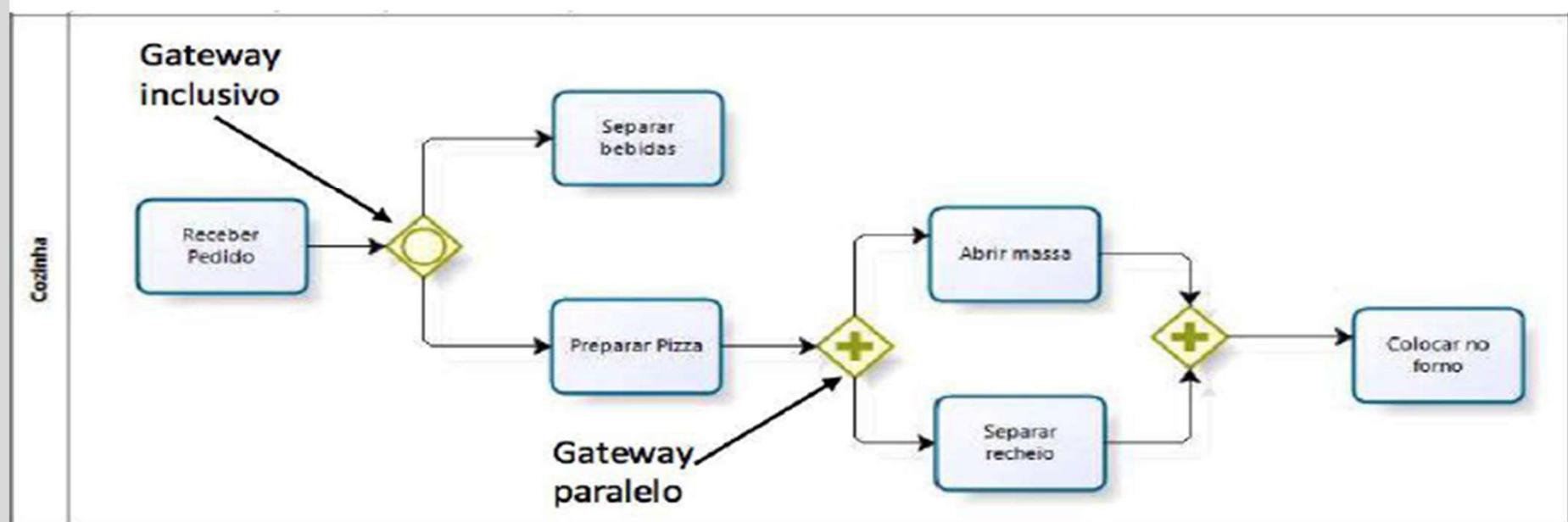


## BPMN – GATEWAYS

**Inclusivo:** Representa atividades feitas simultaneamente;

**Paralelo:** Representam atividades feitas em paralelo, em geral com responsáveis;

**Detalhe:** No gateway paralelo, todas as atividades paralelas devem terminar para que o fluxo continue: no exemplo, só com a massa aberta e o recheio separado e que a pizza vai para o forno.



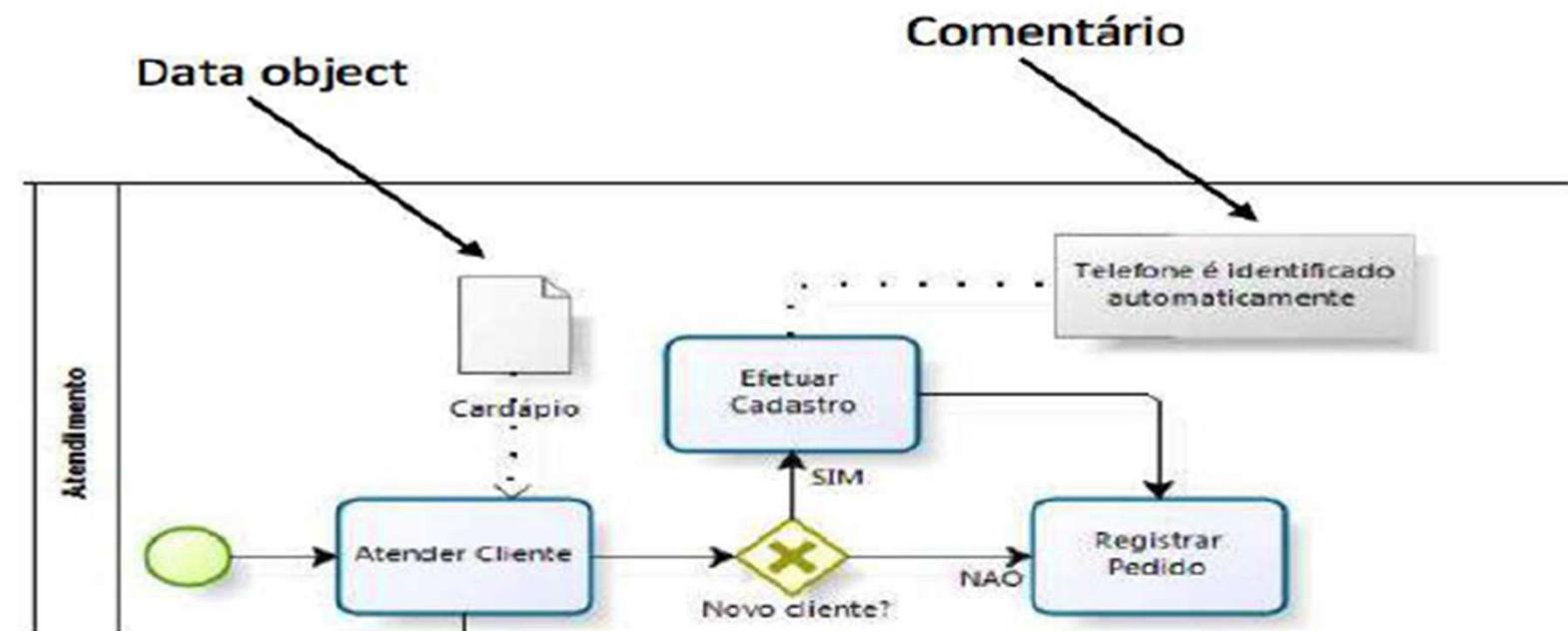
## BPMN – Data Objects e Comentários

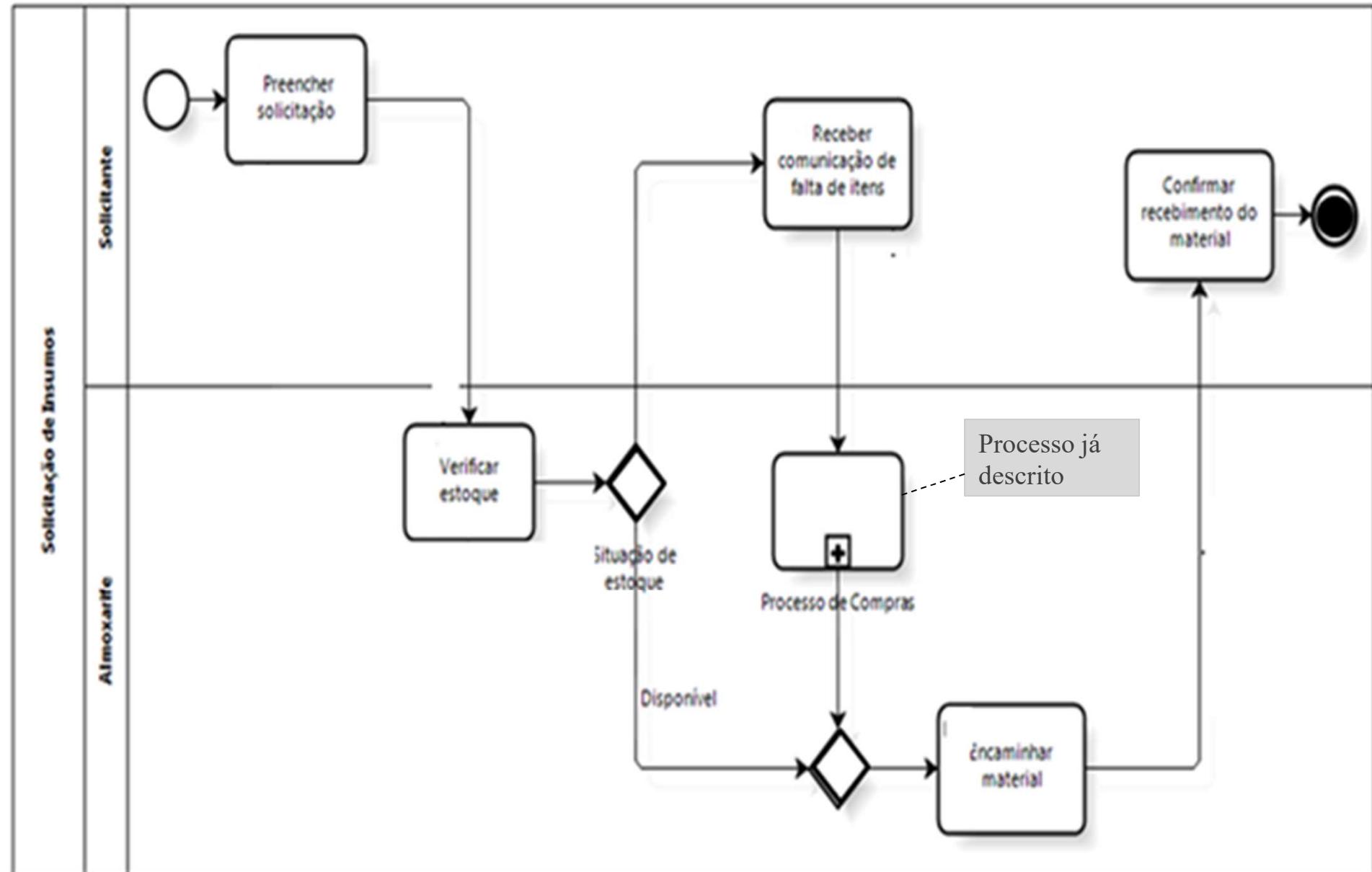
**Data Objects:** Representam entradas associadas a uma atividade ou fluxo.

- ✓ Documentos, normas, manuais, contratos, relatórios e outros.

**Comentários:** visam detalhamento de atividade ou fluxo.

- ✓ Ideal para regras de negócio a serem destacadas.





A BPMN (Business Process Modeling Notation) possui ainda um conjunto de marcadores que definem características operacionais das atividades de processo...

	<b>Marcador de Repetição</b>	→ A tarefa fica em execução até uma condição de saída ser alcançada
	<b>Marcador de Instâncias Múltiplas em Paralelo</b>	→ Atende várias demandas em paralelo
	<b>Marcador de Instâncias Múltiplas em Sequência</b>	→ Atende várias demandas, uma após a outra, criando uma fila
	<b>Tarefa de Usuário</b>	→ Tarefa realizada com o auxílio de um computador/sistema/máquina
	<b>Tarefa Manual</b>	→ Tarefa realizada sem o auxílio de um computador/sistema/máquina
	<b>Tarefa de Invocação de Serviço</b>	→ Tarefa automatizada

...mais um conjunto de marcadores que indicam a natureza dos eventos que disparam as atividades (podem ser usados em conectores iniciais ou intermediários:



- Condiciona um horário/dia/periódico (um texto pode acompanhar o desenho, explicando o evento)

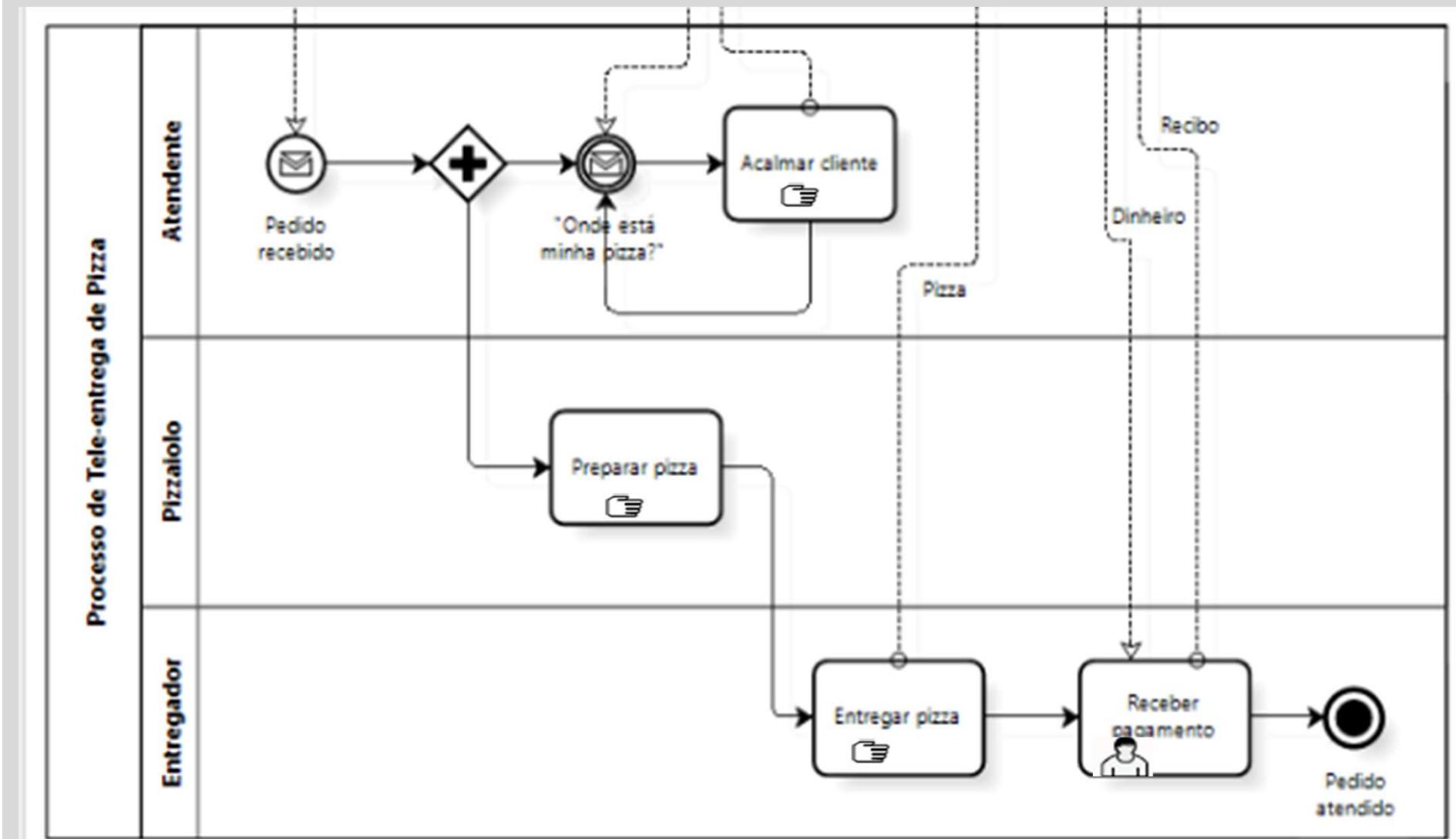


- Condiciona as atividades seguintes a chegada de uma mensagem (um texto pode acompanhar o desenho, explicando o evento)



- Condiciona as atividades seguintes a chegada de um sinal digital (um texto pode acompanhar o desenho, explicando o evento)

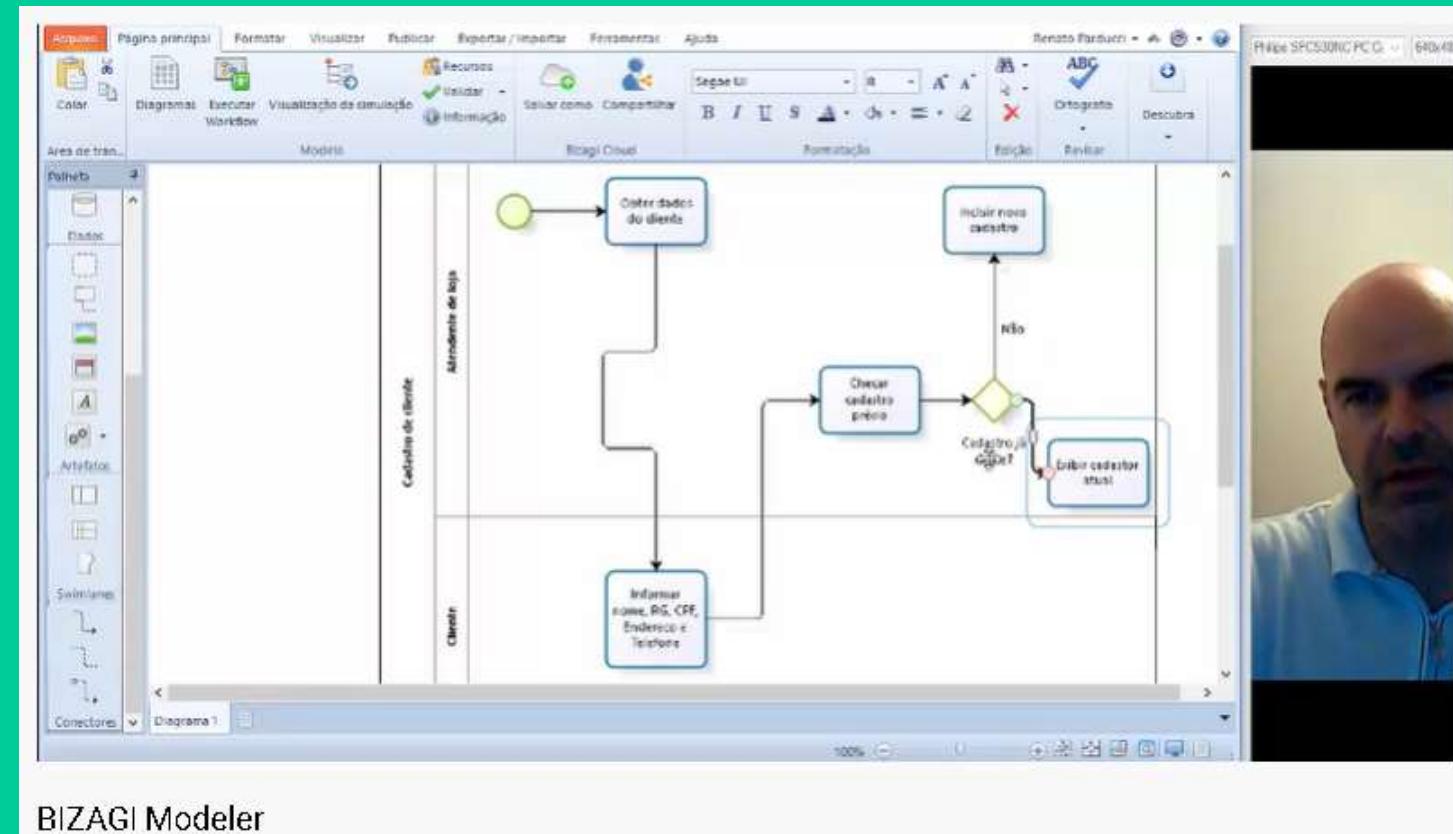
Exemplo da representação de marcadores em um fluxograma:



ASSISTA OS VÍDEOS NO CANAL DO PROFESSOR



<https://youtu.be/imLoIwfuo2g>



Nas observações de rotinas de trabalho:



### Prós:

- Exploração do tema sob várias perspectivas simultaneamente.
- Retratação fiel da realidade, sem distorções de opinião ou humor momentâneo.
- Coleta de documentos base para definir as estruturas de dados da solução
- Identificação de pontos positivos e negativos das soluções atuais



### Contras:

- Necessidade de realizar repetidas visitas ao mesmo local e processo de trabalho para identificar variações de produtividade e qualidade influenciadas pelo perfil de profissionais e competências que mudam em função do turno de trabalho, excesso ou falta de pressão no ritmo de trabalho e gerenciamento sobre resultados.
- Tabulação de resultado bastante complexa.

## ATIVIDADE PRÁTICA



Observando a operação do estacionamento, você identificou que existem duas portarias de entrada e saída de veículos.

O carro para na entrada. O recepcionista confirma com os manobristas se ainda há vagas disponíveis no pátio. Em seguida, o recepcionista captura a placa, o modelo e marca do veículo e o motorista libera o carro para o manobrista que o levará até uma vaga livre identificada.

O recepcionista entrega o recibo de entrada do carrão para o motorista, o qual inclui a taxa por hora de estacionamento no dia e a data e hora de entrada.

Você observou que os manobristas se atrapalham para encontrar as vagas dentro do estacionamento que tem espaço para 5.000 veículos com vagas demarcadas.

Desenhe o processo com BPMN.



Com base nessas observações, quais outros requisitos o seu grupo incluiria na lista?

Vocês têm 10 minutos para discutir e listar.

Edite o documento de Observação e gere uma nova versão no GITHUB.



## DINÂMICAS DE LEVANTAMENTO DE REQUISITOS DE SOFTWARE

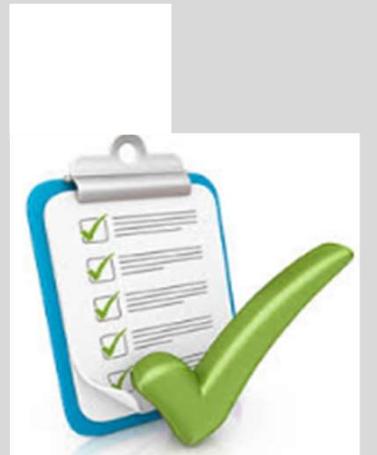
### Questionário de pesquisa



Aplica um formulário que será preenchido por um público alvo.

De preferência, as perguntas e respostas devem ser dirigidas, com alternativas de respostas a serem escolhidas pelo entrevistado, de forma a facilitar a tabulação de dados.

Pode existir espaço para o entrevistado expressar opiniões livremente mas isso pode gerar complicações na compilação de resultados da pesquisa (geração de análises estatísticas).



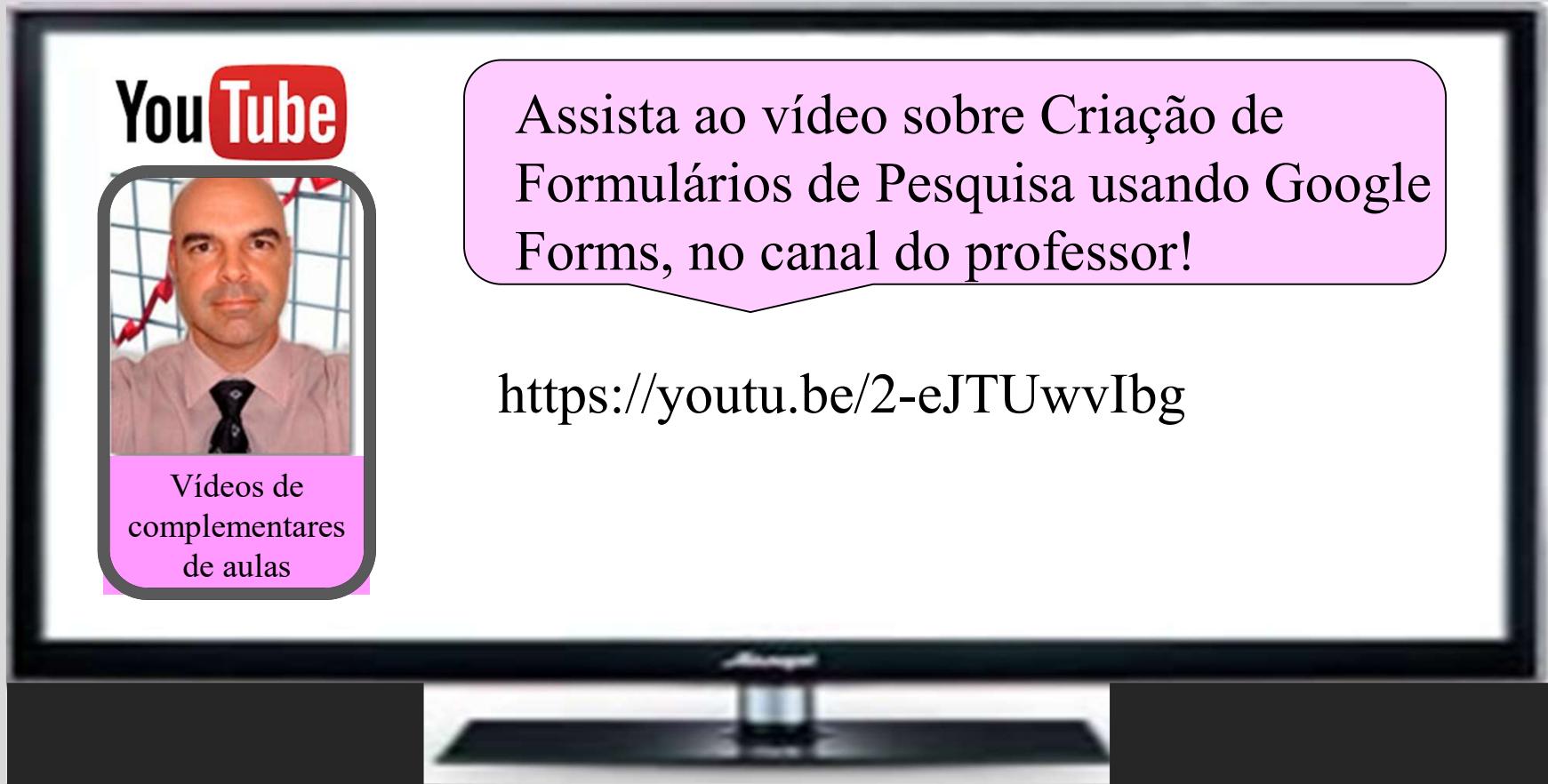
Exemplo de um formulário aplicado a confirmação de requisitos de um sistema de suporte – help desk de TI:

Requisito	Detalhes do requisito	Área de negócio não pode operar sem ele	Área de negócio pode operar sem ele mas terá maior custo e/ou menor desempenho	Requisito é opcional – melhora a experiência do usuário mas não afeta os negócios
Registrar chamado de suporte	Gerar protocolo de atendimento			X
Registrar chamado de suporte	Guardar data e hora da abertura do chamado		X	
Consultar chamado	Apresentar o status	X		
Consultar chamado	Apresentar o nome e ramal do atendente responsável	X		

Exemplo de um formulário aplicado a confirmação de requisitos de um sistema de compras de mercadorias de uma empresa:

Requisito	Detalhes do requisito	Área de negócio não pode operar sem ele	Área de negócio pode operar sem ele mas terá maior custo e/ou menor desempenho	Requisito é opcional – melhora a experiência do usuário mas não afeta os negócios
Cadastrar fornecedor	Classificar por tipo de material fornecido	X		
Cadastrar fornecedor	Guardar data e hora do cadastro			X
Cadastrar fornecedor	Guardar dados de contato comercial	X		
Consultar compras	Total das compras no mês por fornecedor		X	
Consultar compras	Total das compras no mês por mercadoria adquirida		X	

## CANAL DO PROFESSOR



## ATIVIDADE PRÁTICA



Crie um formulário para confirmar os requisitos que precisam ser atendidos em seu projeto de sistema para administração do estacionamento.

Você quer confirmar quais requisitos são mais prioritários para os usuários que devem pontuá-los com: 1-Impressindível; 2-Opcional.

Desenvolva o formulário no Google forms:

<https://www.google.com/forms>



Em seguida, execute o formulário com a sua equipe de projeto e observe a tabulação dos resultados no painel de respostas da ferramenta!



Anote os resultados (recorte e cole) em um documento e guarde no GITHUB com o nome de Formulario.

Na pesquisa em massa:



Prós:

- Alcança um grande público;
- Objetividade (controle e direcionamento);
- Facilidade de interpretação;
- Facilidade de tabulação de resultados.



Contras:

- Nível de retorno baixo, caso os formulários sejam remetidos para preenchimento e retorno pelo colaborador (em torno de 2% a 5% caso não exista um estímulo adicional para a devolução do formulário preenchido ou exigência por chefia);
- Necessidade de conhecer previamente o processo investigado.

## DINÂMICAS DE LEVANTAMENTO DE REQUISITOS DE SOFTWARE

### Reunião de trabalho



Várias pessoas participam, podendo existir vários responsáveis por levantar requisitos e vários entrevistados que darão suas sugestões para o escopo do projeto.

Deve seguir uma pauta de assuntos a qual foi divulgada antecipadamente, sendo o debate aberto.

Deve contar com um responsável por registrar as participações, discussões e deliberações em uma ata.

Um mediador deve ser escolhido para manter a ordem e o foco na pauta.



## ATIVIDADE PRÁTICA



O dono do estacionamento quer que seja avaliada a possibilidade de não existirem mais manobristas e que o estacionamento trabalhe com self-parking, existindo apenas um vigia em cada portaria de acesso.

Vamos fazer uma reunião coletiva onde a sala será dividida em:

- Três pessoas que são supervisores que ficam no estacionamento durante o dia, tarde e noite (3 turnos – 1 responsável por turno);
- Pessoas que são vigias;
- O dono do estacionamento;
- Três pessoas que representam o RH.

Objetivo: adicionar ou não escopo ao projeto e revisar o escopo até então traçado, chegando a um acordo final sobre o que será desenvolvido.

Nas reuniões de trabalho:



Prós:

- Exploração do tema sob várias perspectivas simultaneamente (catálise de opiniões, visões e conhecimentos);
- A própria interação os entrevistados alimenta os entrevistadores de conhecimento sobre o tema;
- A linguagem corporal dos participantes enfatiza o que é relevante na discussão e demonstra a relação de poder entre as pessoas e suas forças no patrocínio do projeto.



Contras:

- Dificuldade para manter o controle sobre a reunião e seu foco central de discussão;
- Registro do conteúdo da reunião é difícil (ideal que ocorra uma gravação de áudio ou vídeo, sendo obrigatório gerar um documento de entendimentos-ATA);
- A tabulação de resultados é difícil;
- Não esgota o entendimento de um tema de pesquisa;
- Dificuldade de agendamento.

## DINÂMICAS DE LEVANTAMENTO DE REQUISITOS DE SOFTWARE

Reunião de trabalho



Para melhorar as relações e resultados de reuniões, você pode aplicar JAD: Joint Application Design

JAD funciona como um Workshop, porém com regras específicas.

- Cada um tem um papel bem definido na reunião:
  - Principais *stakeholders*
  - Equipe de projeto
  - Facilitador
  - Documentador
  - Observadores

- Fluxo de execução:



## Preparação do JAD

### 1. Examinar se é adequada a utilização do JAD

- Perspectiva de sucesso?
- Tamanho do projeto?
- Domínio da técnica?
- Espírito da equipe e do cliente?

### 2. Planejar as sessões

- Quantas?
- Finalidade?

### 3. Elaborar a perspectiva gerencial para nivelamento e informação sobre as expectativas da alta administração

- Objetivos?
- Escopo?
- Premissas e Restrições?

### 4. Familiarizar-se com a área de negócio

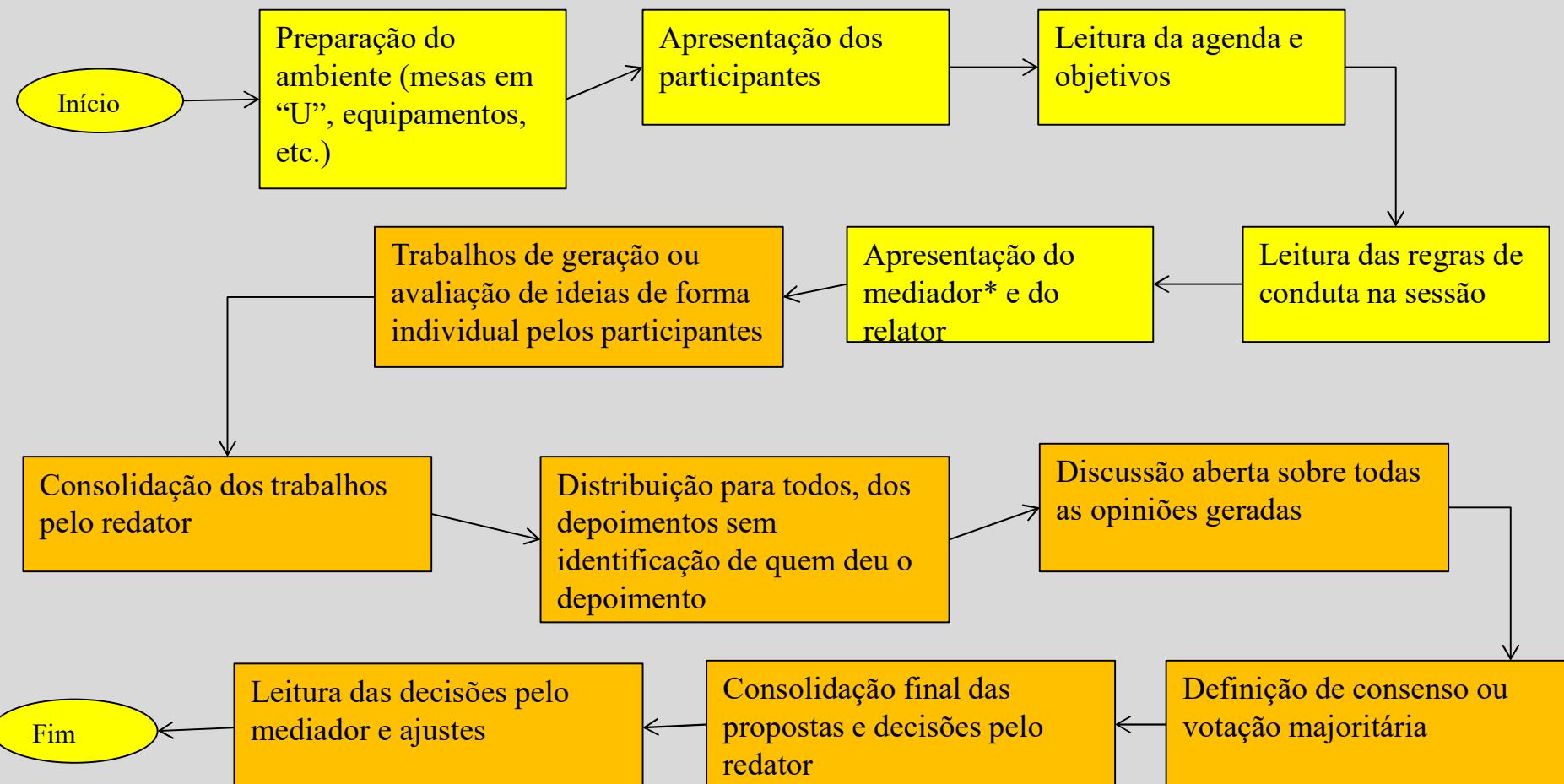
- Documentação?
- Pesquisa dos envolvidos?
- Convocação?

### 5. Preparar a agenda da sessão

- Elemento básico !
- Regras?
- Abordagem?

## Execução da Sessão JAD

\*O mediador também é conhecido como facilitador



**Revisão JAD**

Tem objetivo de avaliar o sucesso da aplicação do método e explorar melhorias que possam ser aplicadas numa próxima sessão.

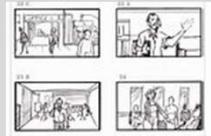
Confirma o alcance de objetivos e define a necessidade de novos encontros para concluir os trabalhos.

## ENGENHARIA DE REQUISITOS

Técnicas complementares de levantamento e validação de requisitos



-Protótipo



-Storyboard

## TÉCNICAS COMPLEMENTARES PARA ESCLARECER REQUISITOS

## Uso de protótipos

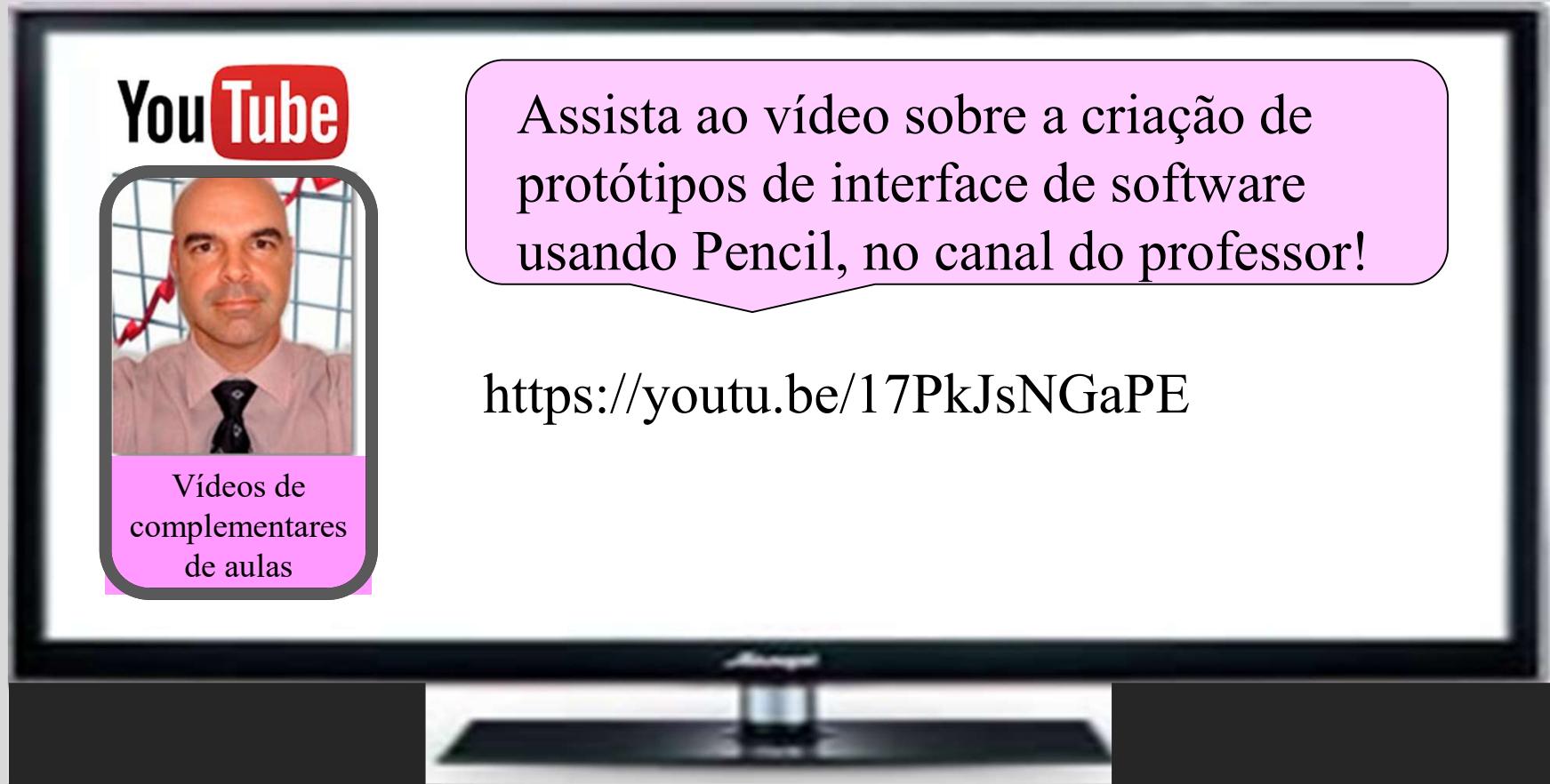


Protótipos podem ser aplicados, seja qual for o Ciclo de Vida de Processo de Software que você escolheu, com a finalidade de confirmar as exigências do projeto.

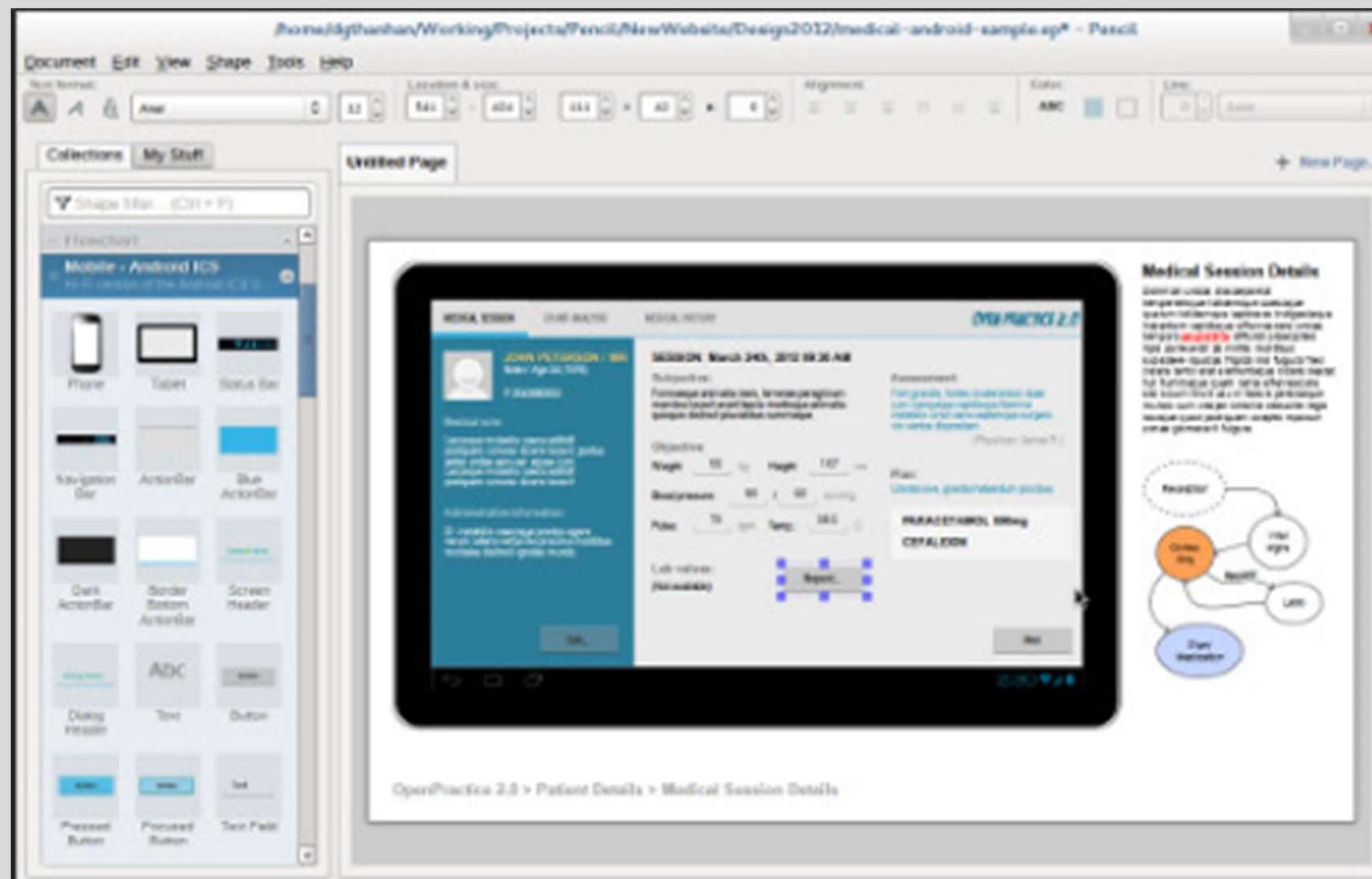
Os protótipos são provas de conceito. Em geral envolvem a produção de desenhos de telas e relatórios de operação sobre os dados armazenados no sistema.

O protótipo é uma “casca” e não tem “alma”, ou seja, ele trata aspectos de interface com o usuário para validação de uma proposta de interatividade mas não contém nenhuma lógica de programação ou arquivo digital de dados.

## CANAL DO PROFESSOR



Exemplo de protótipo de tela de sistema:



## ATIVIDADE PRÁTICA



Crie um protótipo para validar as necessidades de uma tela do sistema de controle de estacionamento – função de registro de entrada de veículo por digitação.

Utilize o aplicativo instalado no seu computador:

- **PENCIL**



Guarde esse documento no GITHUB.

## TÉCNICAS COMPLEMENTARES PARA ESCLARECER REQUISITOS

## Uso de storyboard

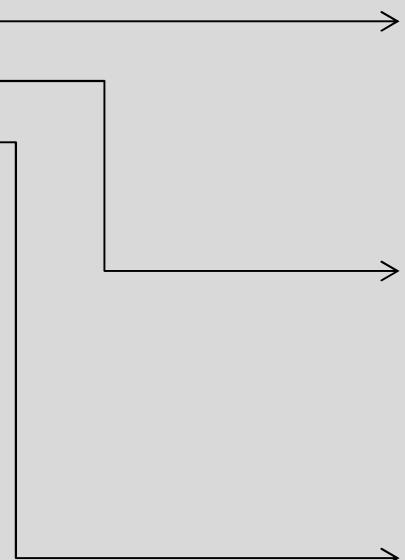


Storyboards são como histórias em quadrinhos que ilustram uma operação, uma atividade de pessoas de uma empresa ou na sua rotina cotidiana.

A proposta é não deixar dúvidas sobre os principais pontos a serem explorados em uma solução.

Os storyboards compõem simulações ou demonstrações que possibilitem a confirmação dos requerimentos de forma:

- Passiva
- Ativa
- Interativa

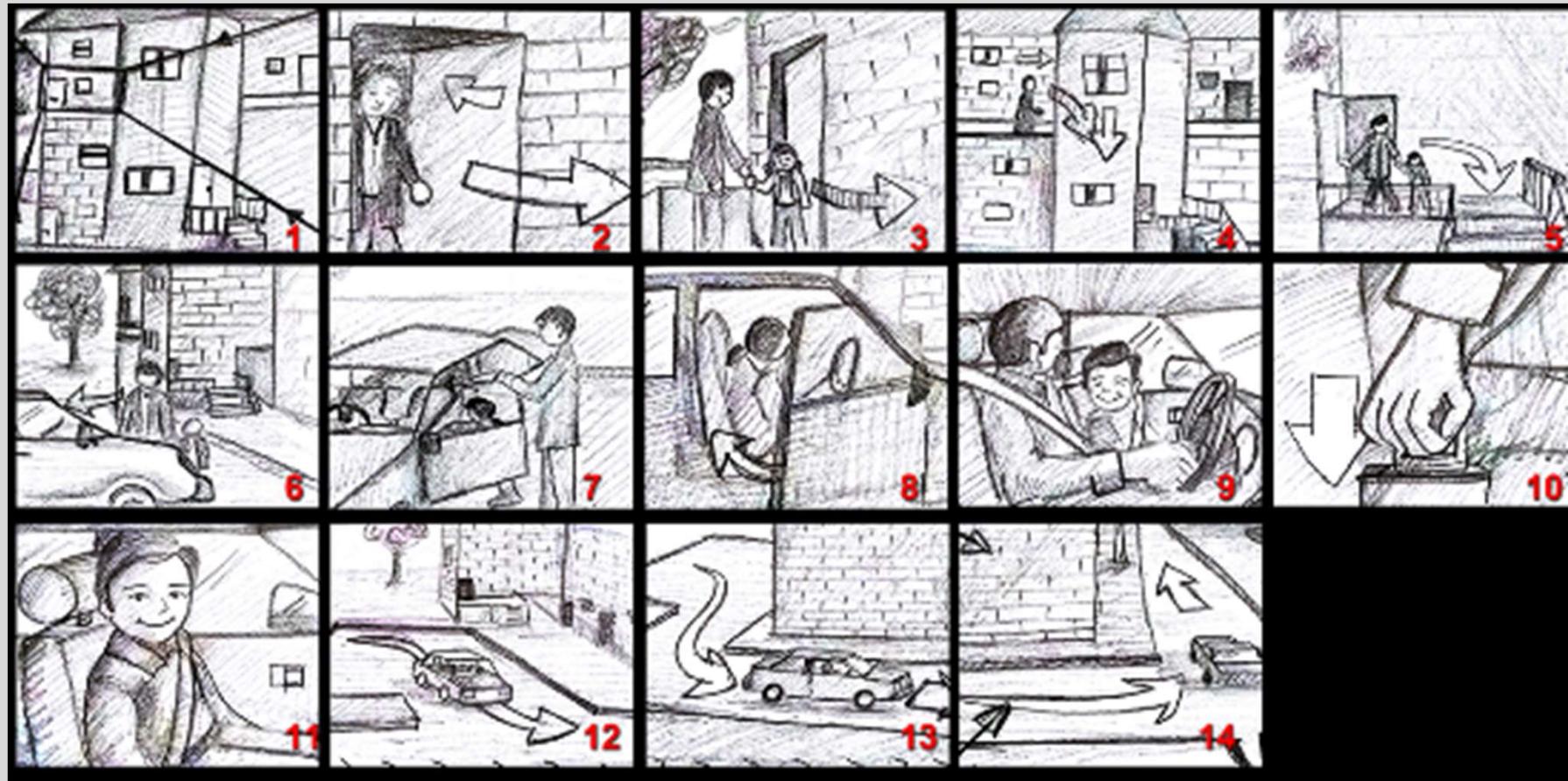


Conta-se uma história para o cliente em forma de figuras, apresentações, vídeos, demonstrações de **protótipos**.

Permite-se que o cliente complemente uma história ou apresentação com suas informações.

Permite-se que o cliente interaja com o protótipo para simular uso e operação e confirmar requerimentos.

Exemplo de storyboard:



## ATIVIDADE PRÁTICA



Faça um Storyboard para demonstrar a operação que você viu (lhe foi contada) sobre como o estacionamento funciona.

Tente resumir em no máximo 10 slides/cenas.

Utilize o MS-PowerPoint para criar os slides!

Suba o documento no GITHUB.



**Os registros do levantamento de requisitos podem conter:**

- Cópias de documentos coletados com usuários;
- Endereços da internet, vídeos e material em diversas mídias que ilustram, exemplificam e conta histórias sobre o que precisa ser desenvolvido;
- Nomes de produtos e soluções de referências (já existentes no mercado e que devem ser seguidos como exemplo ou base inicial para o projeto);
- Filmagens ou gravações de voz obtidas no ambiente de trabalho, durante as observações de campo, reuniões ou entrevistas (sempre sob autorização daquele que é gravado);
- Atas de reuniões;
- Formulários de pesquisa preenchidos (digitais ou em papel);
- Relatórios de resumo dos levantamentos e análises feitas pela equipe técnica.



## PROJETO INTEGRADO DE CURSO



Faça um brainstorm sobre possíveis projetos para o seu AM.

Em seguida, pesquise produtos existentes no mercado, equivalentes aos que você quer desenvolver.

Depois, consolide as ideias iniciais e deixe cada um pensar em detalhes do escopo de requisitos em sua casa.

Promova uma reunião de trabalho com o grupo, alguns dias depois, para alinhar as ideias e tentar fechar um escopo inicial de projeto.

Tente fazer protótipos ou histórias sobre as suas ideias.

## ATIVIDADE EXTRA

Junto com o seu colega de grupo das atividades em sala de aula...

Identifique os requisitos de um jogo de FORCA (aquele jogo em que temos que adivinhar palavras para não sermos virtualmente enforcados).

1º) Descreva como funciona o jogo da FORCA em um texto que reflete a História do Usuário pesquisando sobre o jogo.

2º) Liste os requisitos que você identificou a partir do texto, debatendo com seu colega, de forma a criar um resumo em tópicos que representa o Escopo obtido à partir da História do Usuário.

3º) Inclua na lista de requisito a necessidade de o jogo restringir o tempo de espera por uma resposta (ação) do usuário para cada letra a ser digitada como proposta.

## ATIVIDADE EXTRA

2º) Considere que o jogo deverá ser desenvolvido para funcionar em um computador padrão PC ou Mobile e descreva as características que a interface deve ter e como o usuário deve interagir com a aplicação, complementado a História do Usuário – pode usar desenhos para prototipar ou descreva em formato textual, como se estivesse coletando um depoimento.

Por fim, conclua a lista resumo dos requisitos que definem os atributos que o software deve atender e tipifique-os.

## ATIVIDADE EXTRA

Este projeto opcional vale 1 ponto a mais na última PS do ano se:

- A lista de requisitos estiver correta e completa;
- A modelagem em UML estiver correta e completa;
- O código fonte estiver escrito, respeitando os modelos de engenharia;
- O programa esteja funcionando.

Guarde o seu material:





Material de aula estará no site após a aula.

BONS ESTUDOS!

## Bibliografia

- PRESSMAN, R. S. Engenharia de software. São Paulo: Editora McGraw-Hill, 2002.
- SOMMERVILLE, IAN. Engenharia de software. Editora Pearson, 9.ed. - São Paulo, 2014.
- BEZERRA, EDUARDO. Princípios de análise e projeto de sistemas com UML. Alta Books, Rio de Janeiro, 2006.