Aula 03

Formação - Programação Java

Aula 03

Pesquisa de perfil de cliente e público-alvo

Requisitos funcionais, requisitos não funcionais e regras de negócio

Arquitetura da informação: mapa do site e mapa de navegabilidade

Aula 03

Pesquisa de perfil de cliente e público-alvo

Requisitos funcionais, requisitos não funcionais e regras de negócio

Arquitetura da informação: mapa do site e mapa de navegabilidade

Pesquisa de perfil de cliente e público-alvo

Você teve uma excelente ideia para um **produto de software**, que resolve todos os seus problemas e/ou de seus clientes.

Mas, antes de colocar a mão na massa e sair passando dias e noites programando, cabem alguns questionamentos que são muito importantes para esta empreitada ter sucesso.

Você realmente conhece seu cliente com profundidade?

Aula 03

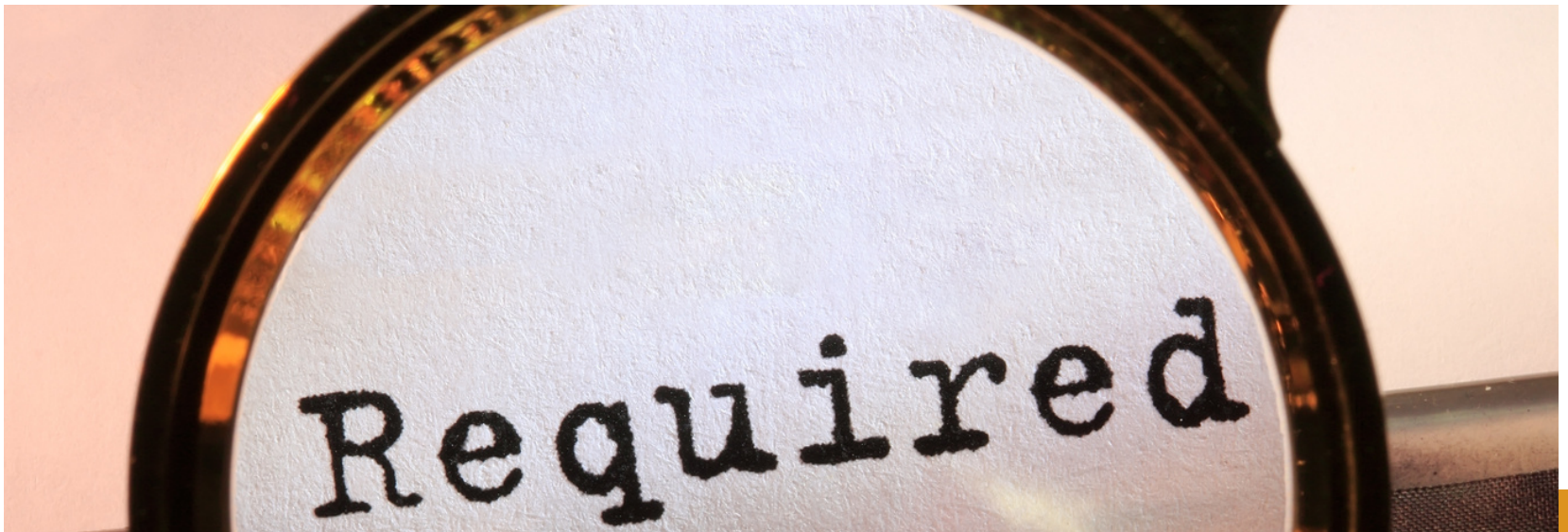
Pesquisa de perfil de cliente e público-alvo

Requisitos funcionais, requisitos não funcionais e regras de negócio

Arquitetura da informação: mapa do site e mapa de navegabilidade

Requisitos funcionais, requisitos não funcionais e regras de negócio

Requisitos funcionais e **requisitos não funcionais** são duas categorias distintas de requisitos que são especificados durante o processo de desenvolvimento de um sistema de software. Vamos entender a diferença entre eles:



Requisitos funcionais

- **Definição**: São as **funcionalidades específicas que o sistema deve fornecer.** Em outras palavras, são as ações que o sistema deve ser capaz de realizar em resposta a entradas específicas.

- **Exemplos**: Um sistema de comércio eletrônico pode ter requisitos funcionais, como permitir que os usuários façam login, adicionem itens ao carrinho, realizem pagamentos online, etc.

- **Medição**: Podem ser medidos e testados de forma objetiva, pois estão relacionados ao comportamento do sistema em relação às entradas do usuário.

Requisitos não funcionais

- **Definição: São atributos ou propriedades do sistema que não estão relacionados a funcionalidades específicas**, mas sim a características globais que afetam seu desempenho, usabilidade, segurança, entre outros aspectos.

- **Exemplos**: Requisitos de desempenho, como tempo de resposta máximo aceitável; requisitos de usabilidade, como a facilidade de aprendizado do sistema; requisitos de segurança, como controle de acesso; requisitos de escalabilidade, como a capacidade de lidar com um grande número de usuários simultâneos.

- **Medição**: Muitas vezes são mais subjetivos e podem ser difíceis de medir diretamente. Podem envolver critérios de aceitação mais qualitativos.

Requisitos funcionais e requisitos não funcionais

Em resumo, **enquanto os requisitos funcionais descrevem o que o sistema deve fazer em termos de funcionalidades e comportamentos específicos, os requisitos não funcionais tratam das características que afetam o sistema como um todo,** independentemente das funcionalidades específicas.

Ambos são cruciais para o sucesso do desenvolvimento de software, pois garantem que o sistema atenda às **expectativas dos usuários em termos de comportamento e desempenho.**

Regras de negócio

Refere-se a **condições, restrições ou operações específicas** que aplicam-se ao domínio do negócio para o qual um sistema de software está sendo desenvolvido.

Essas regras são fundamentais para garantir que o sistema atenda aos **requisitos e expectativas do negócio**. As regras de negócio geralmente encapsulam **políticas**, **procedimentos** e **práticas específicas** que orientam o funcionamento da organização ou atividade para a qual o sistema está sendo desenvolvido.

Pontos-chave sobre regras de negócio

1. **Natureza:**
   - As regras de negócio podem abranger uma ampla variedade de aspectos, desde regras de validação de dados até políticas de tomada de decisão específicas do domínio do negócio.
2. **Independência:**
   - As regras de negócio são frequentemente formuladas de maneira independente das implementações tecnológicas. Elas devem ser entendidas e aplicadas independentemente da plataforma ou tecnologia utilizada.
3. **Impacto no Sistema:**
   - As regras de negócio influenciam diretamente a lógica e o comportamento do sistema. Elas podem ser implementadas em diferentes camadas do software, desde a camada de interface do usuário até a camada de acesso a dados.

Pontos-chave sobre regras de negócio

4. **Alterações Dinâmicas:**
   - As regras de negócio podem mudar com o tempo à medida que o ambiente de negócios evolui. Portanto, é importante que o sistema seja flexível e capaz de acomodar alterações nas regras de negócio sem exigir uma revisão significativa da estrutura do software.
5. **Documentação:**
   - É essencial documentar claramente as regras de negócio para que os desenvolvedores, analistas de negócios e outros stakeholders possam compreendê-las e garantir que o sistema seja desenvolvido de acordo com essas regras.

Regras de negócio

Exemplos de regras de negócio podem incluir **restrições de validação de dados** (por exemplo, um campo não pode estar vazio), **regras de autorização** (quem tem permissão para acessar determinadas informações), **cálculos específicos de negócios**, entre outros.

Entender e aplicar efetivamente as **regras de negócio** é crucial para o sucesso de um s**istema de software**, pois elas desempenham um papel fundamental na criação de soluções que realmente atendam às **necessidades e objetivos do negócio.**

Aula 03

Pesquisa de perfil de cliente e público-alvo

Requisitos funcionais, requisitos não funcionais e regras de negócio

Arquitetura da informação: mapa do site e mapa de navegabilidade

Arquitetura da informação: mapa do site e mapa de navegabilidade

O **mapa do site** e o **mapa de navegabilidade** são ferramentas importantes no design e desenvolvimento de websites, ambos visando melhorar a experiência do usuário e a estrutura de informações do site.

Mapa do Site (Sitemap):

- **Definição**: O mapa do site é uma representação visual ou lista hierárquica de todas as páginas que compõem um site. Ele fornece uma visão geral da estrutura e da organização do conteúdo, mostrando como as páginas estão interconectadas.

- **Objetivo**: Facilitar a compreensão da arquitetura do site, auxiliar no planejamento do conteúdo e na navegação do usuário. Pode ser utilizado como uma ferramenta de comunicação entre membros da equipe de desenvolvimento, designers e stakeholders.

Mapa de Navegabilidade (Wireframe ou Flowchart):

- **Definição**: O mapa de navegabilidade é uma representação visual da interação entre as páginas de um site. Ele mostra como os usuários podem navegar de uma página para outra, indicando links, botões e elementos de navegação.

- **Objetivo**: Ajuda a entender o fluxo de interação do usuário, a lógica de navegação e a disposição dos elementos interativos. Pode ser apresentado como wireframes (esboços de páginas sem detalhes visuais) ou fluxogramas que representam os caminhos que um usuário pode seguir.

Arquitetura da informação: mapa do site e mapa de navegabilidade

Ambos os mapas são complementares e desempenham papéis importantes em diferentes fases do desenvolvimento de um site:

- **Fase Inicial (Planejamento):** O mapa do site é valioso para entender a estrutura global do conteúdo, enquanto o mapa de navegabilidade pode ser útil para planejar a experiência interativa.

- **Design e Desenvolvimento:** O mapa de navegabilidade, especialmente na forma de wireframes, é essencial para representar visualmente a disposição dos elementos nas páginas e as interações possíveis.

- **Testes e Melhorias:** Ambos os mapas podem ser ferramentas úteis para realizar testes de usabilidade e identificar possíveis melhorias na estrutura e na navegação.

Arquitetura da informação: mapa do site e mapa de navegabilidade

Em resumo, **o mapa do site oferece uma visão geral da estrutura de conteúdo**, enquanto o **mapa de navegabilidade se concentra na representação visual das interações entre as páginas**. Ambos são cruciais para criar uma experiência de usuário coesa e eficaz em um site.

Aula 04

Formação - Programação Java

Aula 04

GIT

GITHUB

GIT com VSCODE

Diferenças entre GIT e GITHUB

GIT

É um **sistema de controle de versão,** gratuito e de código aberto. Ou seja, é um software que controla versões de arquivos.

GITHUB

É um **serviço de hospedagem** na **CLOUD**, para projeto que utilizam o GIT.

Instalando o GIT

Ir no site: https://git-scm.com/downloads

Instalando o GIT

Ao instalar o **GIT** em nossa máquina, ele já instala um novo terminal chamado **GIT BASH**

Para verificar se o **GIT** está instalado e simultaneamente verificar a versão atual coloque **git --version** no CMD

Arquivos do projeto HTML

A pasta necessária para essa parte está dentro do **drive**, **Aula04**.

Inicializando seu primeiro repositório

Informar ao **GIT** qual ver ser o **diretório** (directory) que ele irá subir.
Precisaremos do endereço aonde está o diretório.

Uma dica e arrastar o diretório para dentro do terminal que ele irá mostrar o caminho.

CD -> Entrar no diretório
DIR -> Informa os arquivos presentes no diretório

Arquivos do projeto HTML

git init -> Inicializar o diretório

git status -> Informa o status dos arquivos

Realizando o tracking dos arquivos

Nesse momento, todos os arquivos estão no modo **untracked.** Por isso, não conseguimos inserir um **snapshot**. Ou seja, existe um **NO SNAPSHOOT.**

git add . -> Adiciona todos os arquivos para SNAPSHOOT.
git add –all

STAGED -> Está pronto para fazer o commit.
UNSTAGE -> Não está pronto para fazer o commit.

Para tirar algum arquivo do **STAGED** , você poderá utilizar o seguinte comando: **git rm –cached index.html'**

Realizando o primeiro commit

Irá pegar todos os arquivos que estão em **STAGED** e irá gerar uma **versão(snapshoot).**

Material Complementar

Criar um GitHub Pages
https://docs.github.com/pt/pages/getting-started-with-github-pages/creating-a-github-pages-site

Guia Instalação do Git
https://www.hostinger.com.br/tutoriais/tutorial-do-git-basics-introducao

Vídeo: Como colocar meu site na internet com Github Pages
Como colocar meu site na internet com Github Pages (Tutorial Completo)