**Exercício 1- Mapeando e Classificando Requisitos**

**Orientações:**

* **Objetivo:** Praticar os conceitos de requisitos vistos em aula digital.

**Atividades:**

Um sistema que controla o acervo e empréstimo de uma biblioteca qualquer possui uma série de processos organizacionais que são:

* Cadastro e atualização de obras do acervo
* Exclusão de obras do acervo
* Consulta ao acervo
* Empréstimo de obras do acervo
* Reserva de alguma obra do acervo
* Devolução de obras
* Cobrança de obras não devolvidas
* Reserva de obras
* Cadastro de usuários e perfis de acesso
* Relatórios de obras inexistentes
* Relatório de obras mais emprestadas
* Relatório de obras que não são emprestadas a mais de x dias
* Relatório de usuários com devoluções pendentes
* Relatório de giro de empréstimo mensal da biblioteca.

* Elaborar uma lista de 10 **requisitos funcionais** para este sistema e indique a qual processo organizacional que cada um deles se refere. Crie no mínimo um requisito para cada um dos processos organizacionais mencionados acima. (letras a-n)

O modelo da lista de requisitos que vocês deverão preencher se encontra abaixo.

|  |  |  |
| --- | --- | --- |
| **ID** | **Descrição do requisito** | **Processo Organizacional** (exemplos) |
| 01 | O sistema deve realizar um cadastro de um relatório em 5 segundos | Cadastro do acervo |
| 02 | O sistema deveria | Reserva de obras |
| 03 | O sistema deve ter cadastro de usuário e perfis de acesso | Cadastro de usuarios |
| 04 | O sistema deve permitir o empréstimo de obras, registrando a data de retirada e o  prazo de devolução | Empréstimo de obra  s do acervo |
| 05 | O sistema deve permitir a reserva de obras que estejam emprestadas, notificando  o usuário quando a obra estiver disponível | Reserva de alguma  obra do acervo |
| 06 | O sistema deve registrar a devolução de obras, atualizando o status no acervo e  liberando a obra para novo empréstimo. | Devolução de obras |
| 07 | O sistema deve enviar notificações de cobrança para obras que não foram  devolvidas dentro do prazo | Cobrança de obras  não devolvidas |
| 08 | O sistema deve permitir o cadastro de usuários e perfis de acesso com diferentes  níveis de permissão | Cadastro de usuários  e perfis de acesso |
| 09 | O sistema deve gerar relatórios de obras mais emprestadas, com filtro por período | Relatório de obras  mais emprestadas |
| 10 | O sistema deve gerar relatórios de usuários com devoluções pendentes, listando os  prazos devidos | Relatório de usuários  com devoluções  pendentes |

**Dicas :**

* Usar a linguagem de uma forma consistente. Use ‘deve’ para requisitos obrigatórios, e ‘deveria’ para requisitos desejáveis.
* Evitar o uso de jargões de computação
* Elaborar uma lista de 10 **requisitos não funcionais** para este sistema. Informe o tipo de requisito não funcional. Em caso de dúvida, consulte o material no Ulife, sobre os tipos de requisitos não funcionais. O modelo da lista de requisitos que vocês deverão preencher se encontra abaixo

|  |  |  |
| --- | --- | --- |
| **ID** | **Descrição do requisito não funcional** | **Tipo de Requisito Não Funcional** (Exemplos) |
| 01 | O sistema deve processar solicitações de  empréstimo em menos de 3 segundos | Desempenho |
| 02 | O sistema deve ter uma interface amigável e intuitiva, permitindo que os usuários naveguem facilmente entre as funcionalidades | Interface |
| 03 | O sistema deve estar disponível para uso 99,9% do tempo, garantindo alta disponibilidade | Confiabilidade |
| 04 | O sistema deve suportar até 10.000 usuários simultâneos sem degradação de desempenho | Escalabilidade |
| 05 | O sistema deve criptografar todos os dados sensíveis, como informações de usuários e históricos de empréstimo | Segurança |
| 06 | O sistema deve ser compatível com os navegadores mais utilizados, como Chrome, Firefox, e Safari | Compatibilidade |
| 07 | O sistema deve ser facilmente configurável para diferentes bibliotecas, permitindo ajustes nas regras de empréstimo | Flexibilidade |
| 08 | O sistema deve permitir a integração com sistemas externos, como bases de dados de livros online | Interoperabilidade |
| 09 | O sistema deve realizar backups automáticos diários, garantindo a recuperação de dados em caso de falha | Manutenção |
| 10 | O sistema deve realizar backups automáticos diários, garantindo a recuperação de dados em caso de falha | Modificação |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* Requisitos não funcionais podem comprometer os requisitos funcionais? Justifique e dê um exemplo:

R: Sim, os requisitos não funcionais podem comprometer os requisitos funcionais, isso ocorre porque os requisitos não funcionais especifiam as qualidades do sistema, como desempenho, segurança e usabilidade. Se um requisito não funcional não for atendido, ele pode impedir ou dificultar o funcionamento adequado dos requisitos funcionais.

* Porque é importante validar os requisitos antes de passarmos para a próxima fase do processo de desenvolvimento de software?

R: A validação dos requisitos é necessário para garantir que todos os requisitos coletados são precisos, completos e relevantes para o problema que o sistema deve resolver. A validação ajuda a evitar retrabalho, erros e falhas durante as fases posteriores do desenvolvimento.

* Quais os critérios podem ser utilizados para validar requisitos?

R: Viabilidade, testabilidade, Rastreabilidade, Completo.

* Quem participa da validação de requisitos?

R: Clientes e usuário finais, Analista de negócios, Devs, QA, DevOps