

## Exercício - FURPS

Classificando Atributos de Qualidade

Nome : Vítor Ian dos Santos Gonçalves

RA: 321140627

Nome: Larissa Rayane Braga da Paz

RA: 321115521

Nome : Dayane Fonseca Neves

RA: 321122433

Nome: Ivan Kabuto

RA: 320136231

Nome : Bruno Vinicios de Castro Faria

RA: 322128777

Nome: Gabriel Jorge Benevides

RA: 321222943

- **Atividade em Grupo:** 6 participantes
- Escolham um representante do grupo para criar um repositório no Github, e, adicionar os outros integrantes do grupo como colaboradores  
<https://docs.github.com/pt/account-and-profile/setting-up-and-managing-your-github-user-account/managing-access-to-your-personal-repositories/inviting-collaborators-to-a-personal-repository>
- Atualize a planilha [Grupos - Gerência de Qualidade](#)
- Após terminar a atividade:
  - Adicione esse arquivo no formato PDF no repositório;
  - Cada integrante do grupo, poste o arquivo PDF no ulife.

O documento de *Especificação Suplementar de Requisitos* captura os requisitos de sistema que não são capturados imediatamente nos casos de uso do modelo de casos de uso. Entre os requisitos estão incluídos os seguintes atributos de qualidade do sistema: usabilidade, confiabilidade, desempenho e suportabilidade. Tais requisitos fazem parte da FURPS, que é um modelo de classificação de atributos de qualidade de software, desenvolvido na Hewlett-Packard (HP) e publicado pela primeira vez por Grady e Caswell.

Considerando um sistema qualquer, preencha cada uma das seções abaixo do documento de Especificação Suplementar de Requisitos referente à FURPS. A atividade deverá ser feita em sala, com equipes de 6 alunos até o final da aula.

**Nome do Sistema** Tools.net

### Descreva resumidamente o sistema e suas principais funcionalidades

O sistema website Tools.net, tem o intuito de unir instrumentos diários utilizados por desenvolvedores, designers e usuários comuns. O sistema conta com funções matemáticas, geradores, ferramentas ortográficas e conversores de imagem, vídeo, áudio, pdf, etc... Tools.net estará disponível em todas as versões de navegadores, com suporte para Desktop, e dispositivos mobile.

## 1. Funcionalidade

Descreva abaixo os requisitos funcionais do seu sistema, colocando o nome da funcionalidade e a descrição dela, por exemplo:

- **E-mail:** fornecer serviços que permitam que os usuários enviem e recebam mensagens;
- **Ajuda Online:** disponibilizar ajuda online para os usuários;
- **Segurança:** Proporcionar serviços para proteção de acesso a determinados recursos ou informações.

**Conversores:** Converter documentos, imagens, vídeos, áudio, etc...  
**Funções matemáticas:** Resolução de funções básicas da matemática.  
**Ortografia:** Correção e tradução de textos.  
**Geradores:** Geradores de lorem ipsum, nomes, senhas, etc...  
**Imagens:** Manipulação geral de imagem (Compactar, remover fundo, etc...)

## 2. Usabilidade

Descreva nesta sessão

todos os requisitos de qualidade relacionados a usabilidade, tais como: facilidade de uso, facilidade de aprendizado, padrões de usabilidade e localização; Por exemplo:

- Tempo de treinamento necessário para que usuários comuns ou avançados se tornem produtivos em operações específicas no sistema;
- Especifique períodos de tempo mensuráveis para tarefas típicas no sistema;
- UI/UX Design Patterns.

Os requisitos de usabilidade podem incluir as seguintes subcategorias: fatores humanos, estética, consistência na interface com o usuário, ajuda on-line sensível ao contexto, assistentes e agentes, documentação do usuário e materiais de treinamento.

**Portabilidade:** o sistema deve estar disponível para diferentes tipos de navegadores e telas.  
**Tempo de treinamento:** Lendo por média de 5 minutos, conseguirá compreender as operações específicas do sistema, tanto para usuários comuns e/ou avançados.  
 O site poderá ser usado para realizar suas tarefas típicas 24 horas por dia.  
**UI/UX Design Patterns:** Facilidade na navegação, realizando as ações que o usuário deseja, interagindo com a interface, com um layout moderno, mais limpo, sendo criativo, sem poluição visual e pensando em cores que sejam muito quentes, que podem atrapalhar a visualização do conteúdo. Além de uma URL específica, direta e sem muito caracter especial.

## 3. Confiabilidade

Os requisitos de confiabilidade a serem considerados são: frequência e gravidade de falha, capacidade de recuperação, possibilidade de previsão, precisão e tempo médio entre falhas (MTBF). Por exemplo:

- A. Disponibilidade:** especifique a porcentagem de tempo disponível ( xx.xx%), as horas de uso, o acesso à manutenção, as operações de modo degradado, etc.
- B. Tempo Médio entre Falhas (MTBF):** normalmente especificado em horas, mas também poderá ser especificado em termos de dias, meses ou anos.
- C. Tempo Médio para Reparo (MTTR):** quanto tempo o sistema poderá ficar sem funcionar após uma falha?
- D. Exatidão:** especifique a precisão (resolução) e exatidão (através de algum padrão conhecido) necessárias na saída do sistema.
- E. Taxa máxima de erros ou defeitos:** geralmente expressa em termos de erros / KLOC (thousands of lines of code, milhares de linhas de código) ou de erros / ponto de função.
- F. Taxa de erros ou defeitos (categorizada em termos de erros de pouca, média ou muita importância):** os requisitos devem definir o que se entende por erro "crítico"(por exemplo, perda total de dados ou total incapacidade de usar determinadas partes da funcionalidade do sistema).

Descreva abaixo os requisitos de confiabilidade para seu sistema referente aos itens A, B e C da lista acima:

**Disponibilidade:** O sistema tem que funcionar próximo dos 100%, durante 24 horas por dia e 7 dias na semana e sua manutenção deve ser feita por versionamentos.

**Tempo Médio entre Falhas (MTBF):** +/- 2 Semanas

**Tempo Médio para Reparo (MTTR):** +/- 2 Horas

**Exatidão:** Próximo aos 99%

**Taxa máxima de erros ou defeitos:** +/- 10%

**Taxa de erros ou defeitos (categorizada em termos de erros de pouca, média ou muita importância):** +/- 1% para erros de muita importância, +/- 3% para erros de média importância e +/- 10% para erros de pouca importância.

## 4. Desempenho

Descreva as características de desempenho do seu sistema, tais como: velocidade, eficiência, disponibilidade, precisão, produtividade, tempo de resposta, tempo de recuperação e uso de recursos. Por exemplo:

- **Tempo de resposta de uma transação:** tempos médio e máximo;
- **Taxa de transferência:** quantidade de transações por segundo;

- **Capacidade:** o número de clientes ou de transações que o sistema pode acomodar;
- **Modos de degradação:** o modo aceitável de operação quando o sistema tiver sido degradado de alguma maneira;
- **Uso de recursos:** memória, disco, comunicações, etc

**Tempo de resposta de uma transação:** médio de 1s e máximo de 5s.

**Taxa de transferência:** 100 por segundo.

**Capacidade:** 10.000 usuários simultâneos.

**Modos de degradação:** será degradado por funcionalidades, ou seja, a falha de uma função não interfere na outra.

**Uso de recursos:** internet, desempenho do computador.

## 5. Suportabilidade

Descreva todos os requisitos que aprimoraram a *suportabilidade* ou *manutenibilidade* do seu sistema, tais como: padrões de codificação, convenções de nomeação, bibliotecas de classes, acesso à manutenção e utilitários de manutenção. Ainda, os requisitos de suportabilidade podem incluir as possibilidades de teste, adaptação, manutenção, compatibilidade, configuração, serviço, instalação e localização (internacionalização)

**Codificação:** O código deverá ser separado por funções, e indentado de forma correta, conforme os padrões adotados. Será utilizado HTML, CSS, JavaScript e PHP.

**Frameworks:** Bootstrap, React e jQuery.

**Manutenção:** testes unitários com JUnit, para identificarmos falhas, correções, degradações, para que ocorra uma manutenção mais prática.

**Suporte:** utilizar o método de perguntas frequentes, e caso o usuário ainda tenha dúvidas, disponibilizar um formulário com contato via e-mail.