



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**Campus de Quixadá**

# **Plano de Medição de Qualidade**

Davi dos Santos Freitas, Guthyerri Alexandrino Barbosa, Maria Alice de Sousa Macedo  
Profº Camilo Camilo Almendra

# ÍNDICE

GLOSSÁRIO	3
HISTÓRICO DE REVISÕES	3
1. INTRODUÇÃO	4
1.1. Descrição dos produtos a serem avaliados	4
1.2. Objetivos da avaliação	4
2. MÉTODO	5
2.1. Ambiente de avaliação	5
2.2. Procedimentos da Avaliação	5
2.3. Medidas de Software	6
2.3.1. Usabilidade	6
2.3.2. Segurança	7
2.3.1. Manutenibilidade	8
2.3.2. Eficiência de Desempenho	10
2.4. Procedimentos de Interpretação	11
3. REFERÊNCIAS	11

## GLOSSÁRIO

<b>Siglas</b>	<b>Definição</b>
UFC	Universidade Federal do Ceará

## HISTÓRICO DE REVISÕES

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Responsável</b>
13/10/22	1.0	Criação do documento.	Davi dos Santos, Guthyerri Alexandrino e Maria Alice
15/10/22	1.1	Preenchimento da parte de Introdução e Método	Davi dos Santos, Guthyerri Alexandrino e Maria Alice
24/10/22	1.2	Revisão geral da documentação e reescrita de informações gerais.	Davi dos Santos, Guthyerri Alexandrino e Maria Alice
25/10/22	1.3	Finalização da documentação e revisão final	Davi dos Santos, Guthyerri Alexandrino e Maria Alice

## **1. INTRODUÇÃO**

Na finalidade de conceder uma visão mais abrangente do sistema em desenvolvimento e analisado, o documento de plano de medição contém como objetivo o fornecimento de análises técnicas e detalhadas conforme os aspectos de qualidade internos e externos escolhidos. Dessa forma, a destinação do documento refere-se àqueles cujo interesse é focado em conhecer o projeto detalhadamente sob a visão de qualidade de software.

### **1.1. Descrição dos produtos a serem avaliados**

O presente documento apresenta em seu conteúdo o fornecimento de um plano de medição a respeito do sistema de alcinha Ramirez, cujo propósito engloba o apoio aos profissionais da área da fotografia na finalidade de elaboração de seus portfólios pessoais, relações de contratação, descrição de currículo com informações gerais de sua carreira de trabalho, retorno de avaliação por parte dos clientes sobre a contratação.

Nesse íterim, por tratar-se de um sistema móvel e funcional, o ambiente a ser analisado no sistema será, nas partes referentes às características de não-usabilidade, de desenvolvimento para duas camadas de implementação: front-end e back-end. Por isso, a avaliação percorrerá através de cada nova versão do sistema versionada a partir da ferramenta Git, tendo o repositório alocado na plataforma Github. Portanto, a partir de cada versão lançada na plataforma citada, as análises de qualidade detalhadas neste documento serão realizadas para validar a qualidade final do produto no que tange ao desenvolvimento de software.

### **1.2. Objetivos da avaliação**

A avaliação visa avaliar as qualidades de Usabilidade, Segurança, Manutenibilidade e Eficiência de Desempenho.

Usabilidade visa avaliar o aspecto de operabilidade do sistema, logo concentrando o seu foco avaliativo em questões voltadas à conclusão de atividades, tempo de realização de operações para cada funcionalidade mapeada para o projeto e compreensão das atividades fornecidas pelo sistema.

Segurança apoia-se na subcaracterística de integridade, uma vez que, como o sistema apresenta dois tipos de usuários finais possíveis – cliente e fotógrafo –, o acesso a áreas do sistema devem ser autenticadas conforme o tipo de cadastro de cada usuário. Dessa forma, a garantia de acesso restrito faz-se de suma importância no objetivo de manter a segurança da aplicação em conformidade aos requisitos.

Manutenibilidade abrange os aspectos de manutenção de código, isto é, o grau de eficiência e eficácia com que o sistema em questão possa ser modificado. Dentre as subcaracterísticas contidas em sua definição de atuação, temos: testabilidade e reusabilidade. A primeira consiste na medição da quantidade de funcionalidades testadas e validadas pelos testes ao sistema, enquanto a segunda indica o grau que uma implementação pode ser reutilizada na construção de novos produtos.

Para Eficiência de Desempenho, com uma funcionalidade de busca e filtros, contido no sistema, é de grande importância que o tempo de resposta das consultas realizadas apresentem alto grau de eficiência no objetivo de que os dados sejam retornados em tempo hábil na garantia de qualidade conforme os requisitos.

Analisar	Usabilidade
Para o propósito de	avaliar as subcaracterísticas desse atributo de qualidade presentes no sistema
Com respeito a	adequação do objeto medido em relação às métricas de qualidade
Do ponto de vista	do usuário que utiliza o sistema
No contexto de	ambiente de utilização conforme os requisitos

Analisar	Segurança
Para o propósito de	avaliar a cobertura de acesso seguro dos usuários a certas partes do sistema
Com respeito a	adequação da medição conforme os requisitos relacionados a característica de segurança
Do ponto de vista	dos tipos de usuários possíveis no sistema - fotógrafo ou cliente -
No contexto de	uso do produto em um ambiente mobile

Analisar	Manutenibilidade
Para o propósito de	avaliar e medir aspectos técnicos de manutenção
Com respeito a	reusabilidade e testabilidade
Do ponto de vista	do desenvolvedor responsável pela implementação das funcionalidades do projeto e execução de testes unitários
No contexto de	ambiente de aplicação de integração contínua (SonarQube)

Analisar	Eficiência de Desempenho
Para o propósito de	medir aspectos de eficiência das funcionalidades implementadas com conexões diretas a API
Com respeito a	pesquisa de dados
Do ponto de vista	dos desenvolvedores e testadores do sistema
No contexto de	ambiente de aplicação de integração contínua (SonarQube)

## 2. MÉTODO

### 2.1. Ambiente de avaliação

As medidas coletadas durante o processo de avaliação são: Grau de atividades concluídas, Grau de Autenticação, Conjunto Total de Componentes, Cobertura de Testes.

Para a execução e validação das análises, será realizada em um ambiente de desenvolvimento em um computador Windows 10 a partir do emulador da aplicação pelo Android Studio, além de teste de usabilidade feito por dispositivo móvel acessível ou utilizado pelo participante com a versão do android igual ou superior a 8.1.

### 2.2. Procedimentos da Avaliação

Preparado o local de avaliação, os participantes – os envolvidos na contribuição do desenvolvimento do produto, sendo eles desenvolvedores e testadores – deverão ser informados a respeito sobre o sistema de avaliação nos requisitos de tempo e regras de participação – informações a serem registradas, dispositivo que deve ser utilizado e o ambiente que o participante estará inserido –, bem como as atividades que deverão ser

executadas, e que o teste corresponde às características da aplicação e não a habilidade dos participantes em executar determinadas atividades; dessa forma procurando melhorar o código do projeto e desenvolver novas possíveis melhorias.

Dessa forma, o avaliador explica que os pontos a serem avaliados serão Usabilidade (Operabilidade), Segurança (Integridade), Manutenibilidade (Testabilidade e reusabilidade) e Eficiência de Desempenho (Comportamento). Destarte.

As tarefas propostas para os participantes executarem no decorrer do plano de mediação incluem as seguintes atividades:

- Realizar cadastro de usuário pelos dois fluxos – cliente e fotógrafo;
- Realizar o login no sistema a partir da sua conta recém-criada;
- Utilizar a funcionalidade de busca a partir dos seguintes filtros possíveis:
  - Nome do fotógrafo;
  - Especialização;
  - Faixa de preço: valor mínimo e valor máximo;
  - Local: cidade ou estado;
  - Ordenar por:
    - Visualizações.
    - Curtidas;
- Visitar perfil de fotógrafo;
- Visualizar postagem;
- Criar comentário ou curtir postagem;
- Deletar postagem ou comentário;
- Postar nova foto no sistema, com ou sem preço de venda

## 2.3. Medidas de Software

### 2.3.1. Usabilidade:

<b>Nome</b>	Grau de atividades concluídas
<b>Objetivo</b>	Medir grau em que um conjunto de atividades da aplicação é concluída pelo usuário.
<b>Descrição</b>	Qual a quantidade de atividades do sistema que foram concluídas pelo usuário? A medida relacionada às atividades apresenta os seguintes 2 estados: <ul style="list-style-type: none"> <li>● Concluída;</li> <li>● Não concluída;</li> </ul>
<b>Fórmula</b>	x = quantidade de funcionalidades concluídas y = quantidade total de funcionalidades mapeadas no sistema; $z = \frac{x}{y} \times 100$ (valor percentual de funcionalidades concluídas)
<b>Parâmetros</b>	Se $\leq 50\%$ indica um baixo grau de atividades concluídas pelo usuário; Se $x > 50\%$ , indica que grau moderado ou alto de conclusão de atividades.

<b>Procedimento de coleta</b>	<p><b>Como coletar:</b> as funcionalidades são coletadas a partir de seus registros na avaliação de usabilidade pelo avaliador.</p> <p><b>Como armazenar:</b> através de um documento de Teste de Usabilidade.</p>
<b>Análise</b>	<p><b>Como analisar:</b> analisar a quantidade de funcionalidades concluídas pelo usuário e o meio e o tempo considerados que apresentem grau moderado ou alto de conclusão de atividades.</p> <p><b>Ações corretivas:</b> caso ocorra uma longa demora na realização de alguma funcionalidade (mais de 10 minutos), o avaliador deve orientar o participante para uma nova atividade.</p> <p><b>Onde armazenar:</b> a ocorrência deve ser descrita no documento de teste de usabilidade com os devidos registros de impedimentos da conclusão da atividade pelo usuário.</p> <p><b>Como executar a revisão da efetividade do indicador:</b> revisão do design do projeto com o UX Designer e notificar os desenvolvedores das possíveis novas alterações</p>

### 2.3.2. Segurança:

<b>Nome</b>	Grau de Autenticação
<b>Objetivo</b>	Medir grau em que um produto ou sistema garante que os dados são acessíveis somente por pessoas autorizadas.
<b>Descrição</b>	<p>Qual a quantidade de rotas que exigem autenticação para que seja restrito a um tipo de usuário cujo acesso não seja permitido?</p> <p>A medida relacionada às autenticações apresenta os seguintes 3 estados:</p> <ul style="list-style-type: none"> <li>• Não implementada;</li> <li>• Parcialmente implementada;</li> <li>• Totalmente implementada;</li> </ul>
<b>Fórmula</b>	<p><math>x</math> = quantidade de funcionalidades com autenticação implementadas;</p> <p><math>y</math> = quantidade total de autenticações mapeadas para o projeto;</p> <p><math>z = \frac{x}{y} \times 100</math> (valor percentual de acessos a funcionalidades já autenticadas)</p>
<b>Parâmetros</b>	<p>Se <math>z = 0</math>, indica que nenhuma autenticação a todas funcionalidades do projeto foram implementadas;</p> <p>Se <math>z \leq 25\%</math> indica que <math>\frac{1}{4}</math> das autenticações foram implementadas;</p> <p>Se <math>z = 50\%</math>, indica que <math>\frac{1}{2}</math> dos acessos tiveram as autenticações implementadas;</p>

	<p>Se <math>x &gt; 75\%</math>, o valor informa uma implementação de <math>\frac{3}{4}</math> das autenticações;</p> <p>Se <math>x = 100\%</math>, há total implementação das autenticações no sistema.</p>
<b>Procedimento de coleta</b>	<p><b>Como coletar:</b> as funcionalidades são coletadas a partir de seus registros em uma planilha;</p> <p><b>Como armazenar:</b> através de indicadores, na própria planilha, de que a funcionalidade foi totalmente implementada.</p>
<b>Análise</b>	<p><b>Como analisar:</b> avaliar a quantidade de indicadores que indicam as funcionalidades cujas autenticações ainda não foram implementadas.</p> <p><b>Ações corretivas:</b> caso ocorra alteração nos requisitos a respeito do acesso a uma funcionalidade, registrar a mudança nos indicadores da planilha.</p> <p><b>Onde armazenar:</b> o resultado total é calculado e armazenado na planilha que contém os dados gerais das funcionalidades implementadas.</p> <p><b>Como executar a revisão da efetividade do indicador:</b> revisão de código nas partes relativas às funcionalidades cujas autenticações são exigidas de forma que haja certificação do compromisso com os requisitos.</p>

### 2.3.3. Manutenibilidade:

<b>Nome</b>	Conjunto Total de Componentes
<b>Objetivo</b>	Essa medida visa mapear a quantidade total de implementações no projeto que podem ser transformadas em componentes para evitar retrabalho.
<b>Descrição</b>	<p>Quantas implementações podem ser transformadas em componentes?</p> <p>A medida relacionada aos componente apresenta os seguintes 3 estados:</p> <ul style="list-style-type: none"> <li>• Reutilizável;</li> <li>• Não reutilizável</li> </ul>
<b>Fórmula</b>	<p><math>n</math> = quantidade total de implementações no projeto;</p> <p><math>m</math> = quantidade de implementações utilizadas em outras etapas;</p> <p><math display="block">z = \sum_{x=m}^n x \quad (\text{valor total de implementações para serem transformadas em componentes})</math></p>



<b>Parâmetros</b>	<p>Se <math>\frac{z}{n} \geq 50\%</math>, as implementações apresentam alta possibilidade de serem componentes;</p> <p>Se <math>\frac{z}{n} \geq 50\%</math>, as implementações apresentam alto grau para se transformarem em componentes.</p>
<b>Procedimento de coleta</b>	<p><b>Como coletar:</b> avaliar implementações a nível de código que são constantemente implementadas desnecessariamente para que se tornem componentes reutilizáveis.</p> <p><b>Como armazenar:</b> arquivar todas as implementações em um repositório comum para que possam ser mapeados.</p>
<b>Análise</b>	<p><b>Como analisar:</b> mapear cada implementação que esteja sendo implementada constantemente em mais de um caso no projeto.</p> <p><b>Ações corretivas:</b> caso haja bastante diferença entre as implementações, elaborar relatório para analisar suas conformidades aos requisitos.</p> <p><b>Onde armazenar:</b> criar relatório com o detalhamento de cada implementação para que seja colocada em um repositório compartilhado.</p> <p><b>Como executar a revisão da efetividade do indicador:</b> elaborar encontros com o time de desenvolvimento para análise e revisão das implementações.</p>

<b>Nome</b>	Cobertura de Testes
<b>Objetivo</b>	Essa medida visa mapear a quantidade total de testes no projeto implementados, testados e validados
<b>Descrição</b>	<p>Quantos testes foram implementados e validados no projeto para as funcionalidades totais do sistema?</p> <p>A medida relacionada aos testes apresentam os seguintes 3 estados para as funcionalidades mapeadas no sistema:</p> <ul style="list-style-type: none"> <li>• Não testado;</li> <li>• Testado e não validado;</li> <li>• Testado e validado;</li> </ul>
<b>Fórmula</b>	<p><math>n</math> = quantidade total de funcionalidades testadas e validadas;</p> <p><math>m</math> = quantidade de funcionalidades;</p> <p><math>z = \frac{m}{n} \times 100</math> (valor de cobertura de funcionalidades testadas)</p>
<b>Parâmetros</b>	<p>Se <math>z = 0</math>, indica que nenhum teste foi implementado</p> <p>Se <math>z &lt; 50</math> indica que <math>\frac{1}{4}</math> dos testes foram validados, portanto uma apresentação baixa de cobertura;</p>

	<p>Se <math>z \geq 50</math> e <math>z &lt; 75</math>, indica que <math>\frac{1}{2}</math> dos testes tiveram foram validados, logo havendo uma cobertura parcial.</p> <p>Se <math>z \geq 75</math>, o valor informa uma cobertura de <math>\frac{3}{4}</math> a 1 de testes para as funcionalidades total, logo apresentando uma alta taxa de cobertura.</p>
<b>Procedimento de coleta</b>	<p><b>Como coletar:</b> avaliar implementações a nível de código que necessitam serem testadas e validadas.</p> <p><b>Como armazenar:</b> arquivar todos os testes realizados em uma planilha contendo a taxa total de cobertura e os testes pendentes.</p>
<b>Análise</b>	<p><b>Como analisar:</b> mapear cada funcionalidade não testada ou testada e não validada no sistema. Verificar o grau final de cobertura que alcance a meta acima de ou igual a 75%.</p> <p><b>Ações corretivas:</b> caso haja presença de teste cuja funcionalidade não esteja conforme os requisitos, iniciar sua remoção e de semelhantes. Além disso, caso a meta não seja atingida, mapear os testes que não foram validados ou nunca testados e iniciar uma ação de mitigação para uma próxima análise de medição.</p> <p><b>Onde armazenar:</b> criar relatório com o detalhamento de cada teste executado para cada funcionalidade a fim de serem descritos na planilha final de testes.</p> <p><b>Como executar a revisão da efetividade do indicador:</b> elaborar encontros com o time de desenvolvimento para análise e revisão dos testes executados.</p>

#### 2.3.4. Eficiência de Desempenho:

<b>Nome</b>	Média de Tempo de Resposta
<b>Objetivo</b>	Essa medida visa medir o tempo médio das requisições realizadas para cada funcionalidade do projeto cuja sua utilização é exigida para atingir a sua completude.
<b>Descrição</b>	<p>Qual é o tempo médio que o sistema leva para responder às exigências requisitadas pelo usuário?</p> <p>O seguinte dado deve fornecer o valor relativo a tempo na unidade de <i>ms.</i> conforme o padrão SI, para cada requisição realizada no período de avaliação.</p>
<b>Fórmula</b>	<p><math>n</math> = quantidade total de funcionalidades</p> <p><math>m_i</math> = média do tempo de resposta da funcionalidade <math>i</math></p> $z = \frac{\sum_{i=1}^n m_i}{n} \text{ (média do tempo de resposta do sistema)}$

<b>Parâmetros</b>	<p>Se <math>z \leq 1000_{ms}</math> indica que o tempo de resposta do sistema corresponde a um comportamento hábil.</p> <p>Se <math>z &gt; 1000_{ms}</math> e <math>z \leq 5000_{ms}</math>, indica que o tempo de resposta do sistema corresponde a um comportamento moderado.</p> <p>Se <math>z &gt; 5000_{ms}</math> e <math>z \leq 5000</math>, indica que o tempo de resposta do sistema corresponde a um comportamento lento ou demorado.</p>
<b>Procedimento de coleta</b>	<p><b>Como coletar:</b> para cada funcionalidade cuja requisição é necessária, realizar a execução da chamada à API até 10 vezes e, a partir disso, calcular o valor da média do tempo de resposta na grandeza de milissegundos (<i>ms</i>). Com todas as médias calculadas, calcula-se a média das médias de tempo de execução.</p> <p><b>Como armazenar:</b> arquivar todos os dados relacionados à média do tempo de execução, somatório e resultado em uma planilha de avaliação na aba restrita para medida de <b>Eficiência de Desempenho</b>.</p>
<b>Análise</b>	<p><b>Como analisar:</b> avaliar se o tempo médio de resposta do sistema indica um comportamento conforme o estabelecido pelos parâmetros. Desse modo, o resultado indicará a natureza do sistema para o comportamento de eficiência.</p> <p><b>Ações corretivas:</b> caso a meta desejada não seja atingida – sistema com tempo de resposta hábil –, faz-se necessário elaborar um plano de ação com o propósito de reduzir o tempo médio de resposta.</p> <p><b>Onde armazenar:</b> descrever em um documento de plano de ação as novas ações a serem tomadas para que a meta seja atingida junto ao time de desenvolvimento.</p> <p><b>Como executar a revisão da efetividade do indicador:</b> realizar uma reexecução da medida para cada funcionalidade e realizar o cálculo final e comparar o resultado atual com o anterior para que o novo valor seja avaliado e possivelmente validar a medição.</p>

## 2.4. Procedimentos de Interpretação

As análises de todas as medidas representam valores quantitativos cujo conteúdo proporciona uma análise mais técnica e direta das qualidades do sistema de maneira que o procedimento de análise seja executado sem impasse algum. Por isso, as tabelas de medição correspondem ao padrão GQM – Goal Question Metrics –, haja vista que cada campo preenchido corresponde às ações a serem tomadas para cada etapa do processo de medição.

Para o caso do teste de **Eficiência de Desempenho**, há contido em sua medição um valor hipotético de 10, correspondendo somente a quantidade de repetições que uma requisição será realizada na finalidade de atingir uma média do tempo de resposta para uma única funcionalidade.

Os outros valores para as métricas restantes estão descritas em cada tabela correspondente.

### **3. REFERÊNCIAS**

ISO/IEC 25000. Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE. v. 2005, 2005.

ISO/IEC 9126. Software Engineering – Product Quality – Part 1. 2001