



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

# **Plano de Medição de Qualidade**

***Jeferson Gonçalves Noronha Soriano -  
471110***

**Responsável: Doe**

## GLOSSÁRIO

Siglas	Definição
UFC	Universidade Federal do Ceará

## 1. INTRODUÇÃO

O documento destina-se a equipe que está desenvolvendo o software, desde de suas primeiras etapas as últimas, buscando centralizar informações sobre qualidade para que depois se outras pessoas tenha acesso ao documento, possam entender todo o processo de medição que o software foi submetido

### 1.1. Descrição do Produto

O produto analisado é o aplicativo denominado doe, um app mobile que rodará nos principais aplicativos Android.

O app tem como intuito facilitar o encontro entre 2 partes no âmbito de doação, a pessoa que doa e que é beneficiada com a doação, só que invertendo um pouco a lógica comum atual onde a pessoa que está precisando de ajuda posta o que ela ta precisando e invertendo para quem tem algo para doar, posta no aplicativo e quem precisa de doação entre em contato com o doador

### 1.2. Objetivos da Avaliação

O objetivo da avaliação é validar a usabilidade (eficácia e eficiência da interação) das principais funcionalidades do aplicativo doe

Um usuário será submetido a tarefas no software para analisarmos se a interface compre com seu papel e também como e também terá uma análise internamente como o software está desempenhando

Analisar	o doe
Para o propósito de	entender o que esta de acordo com o planejado
Com respeito a	interface e como ele se comunica com o usuario
Do ponto de vista	de usuário
No contexto de	uso real

## 2. MÉTODO

### 2.1. Participantes

Para a avaliação participaram 4 usuários reais que possuem certo nível de experiência uso de smartphone e contado com tecnologia , os participantes da avaliação possuem idade no intervalo de 15 a 30 anos, com gêneros variados. Dentre os participantes há indivíduos que possuem mais familiaridade com TI e outros não

### 2.2. Contexto de Uso

- Tarefas:
  - Realizar cadastro
  - Realizar login
  - Cadastrar produto.
    - No ponto de vista do usuário beneficiado
  - ver detalhe de um produto.

- entrar em contato com usuário doador pelo whatsapp
- Ambientes:
  - Ambiente real 1, celular: android 9, ASUS\_A001D.

### 2.3. Procedimentos da Avaliação

Ao chegar, os participantes foram informados de que a usabilidade do produto doo seria avaliada para descobrir se o produto satisfaz suas necessidades. Os usuários foram informados de que não era um teste de suas habilidades e sim da usabilidade da aplicação. Em seguida, o avaliador explicou para os usuários como seria a avaliação, apresentando o dispositivo a ser utilizado, o laboratório e informou que sua interação seria registrada.

### 2.4. Medidas de Software Coletadas

#### Métricas Externas

##### 2.3.1 Eficácia

Nome	Descrição	Função de Medição	Método
Frequência de erros	Qual é a frequência de erros cometidos pelo usuário em relação a um valor-alvo	$X = A/B$ A = número de erros cometidos pelos usuário B= número de tarefas (ou pode ser o tempo)  Note: O número de erros cometidos pelo usuário pode incluir todos os erros, ou apenas erros não corrigidos, ou apenas erros que fazem com que a tarefa não seja concluída corretamente.	Mede o desempenho do usuário.

##### 2.3.2 Eficiência

Nome	Descrição	Função de Medição	Método
Completo da tarefa	Qual é a frequência de erros cometidos pelo usuário em relação a um valor-alvo?	$X = A/B$ A = número de erros cometidos pelos usuário B= número de tarefas (ou pode ser o tempo) Note: O número de erros cometidos pelo usuário pode incluir todos os erros, ou apenas erros não corrigidos, ou apenas erros que fazem com que a tarefa não seja concluída corretamente.	Mede o desempenho do usuário.

#### Métricas Internas

### 2.3.3 Manutenibilidade

Nome	Descrição	Função de Medição	Método
Quantidade de linhas de uma classe	Mede a quantidade de linhas que uma classe tem para determinar se uma classe num ta fazendo mais do que deveria	$X = A - B$ A = Número de linha desejável (entre 100 e 400) por classe B = Número de linha encontradas na classe	Mede se as classes estão com a quantidade de linhas adequadas

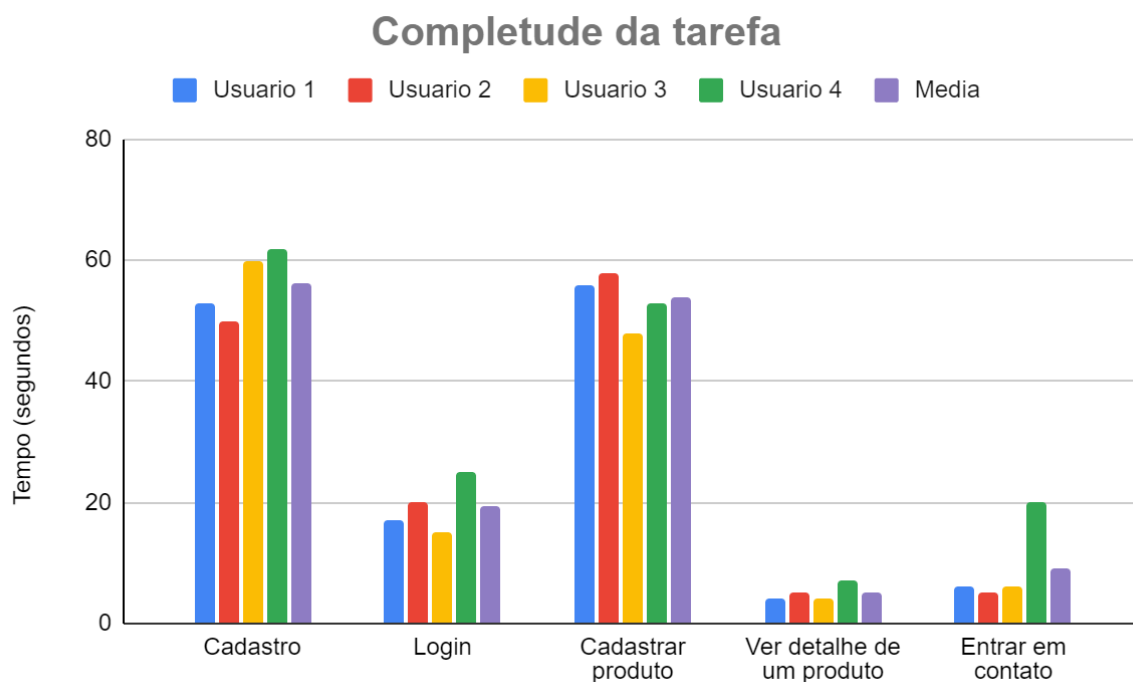
### 2.3.4 Manutenibilidade

Nome	Descrição	Função de Medição	Método
Quantidade de métodos de uma classe	Mede a quantidade de métodos que uma classe tem para determinar se uma classe num ta fazendo mais do que deveria	$X = A - B$ A = Número de métodos desejável (entre 13 a 16) por classe B = Número de métodos encontrados na classe	Mede se as classes estão com a quantidade de métodos adequadas

## 3. RESULTADOS

### 3.1. COMPLETEDE DA TAREFA

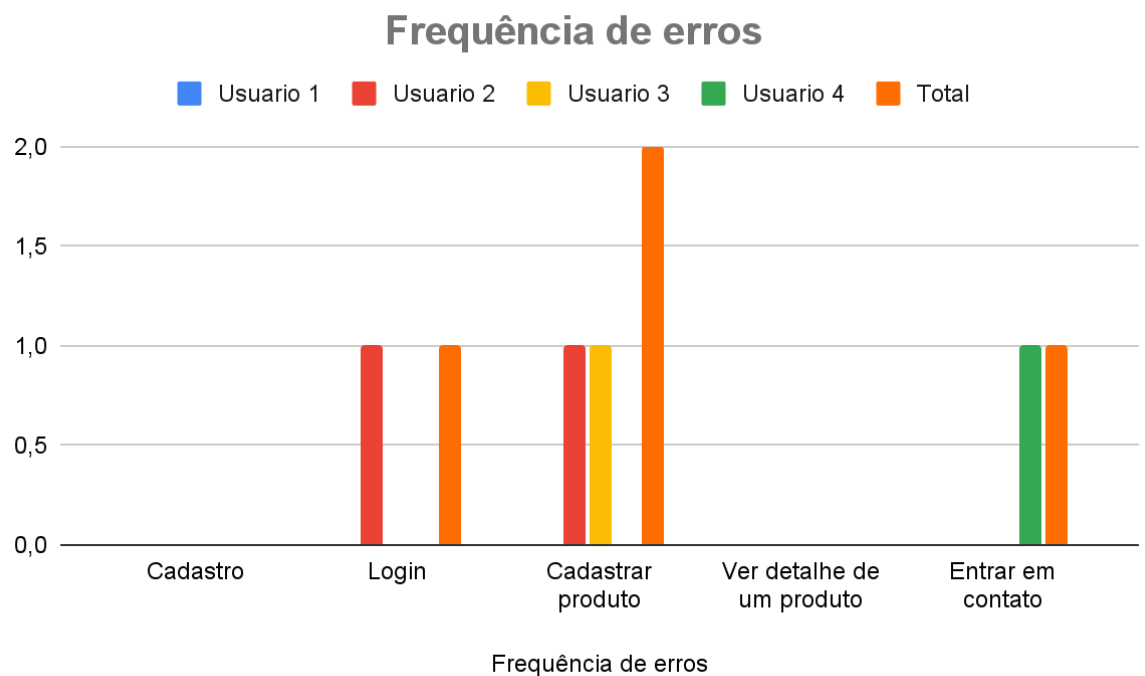
Como foi proposto na métrica de completude da tarefa, avaliamos o tempo que o usuário levou para completar uma tarefa, desde o momento que o pedido da tarefa foi pedido, até o entendimento do usuário que completou a tarefa, a seguir vemos os resultados obtidos.



Assim no gráfico temos as tarefas pedidas aos usuários e o tempo que cada usuário levou para concluir as tarefas, e logo depois uma média dos usuários para concluir a tarefa em questão, podemos perceber que as tarefas de cadastro são as que mais demoram, isso se dá pelo fato do preenchimento dos dados, mas apesar do tempo, é algo justo, não é nada que atrapalhe a experiência.

### 3.2. FREQUÊNCIA DE ERROS

Como foi proposto na métrica de Frequência de erros, avaliamos a quantidade de erros cometidos pelo usuário até completar uma tarefa proposta, a seguir vemos os resultados obtidos

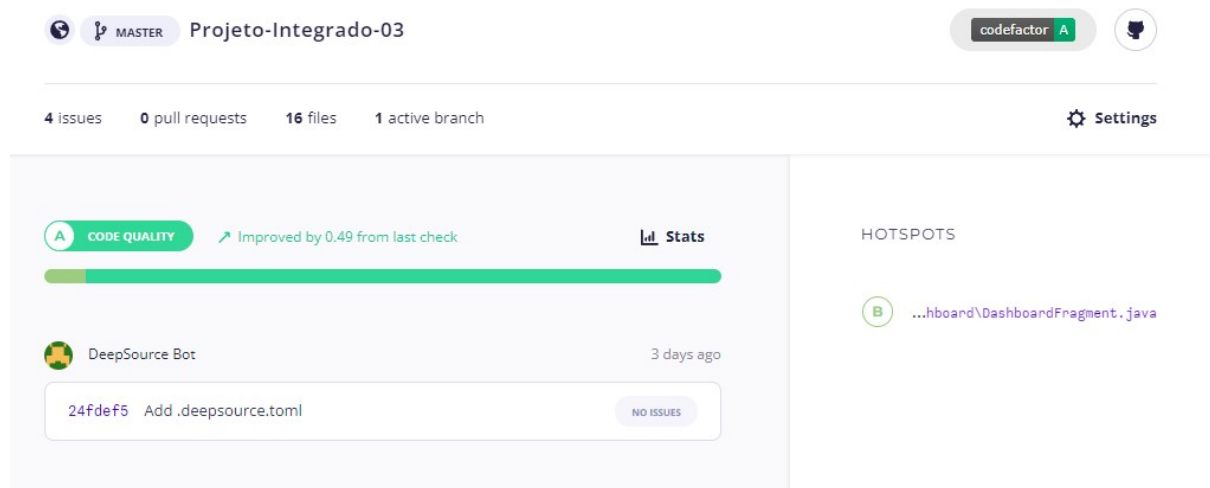


A Partir da análise do gráficos podemos notar quantos usuários erraram em uma tarefa e quantidade de erros cometidos naquela tarefa, que seria o “total” no gráfico, então podemos notar que a tarefa que mais obteve erros e usuários distintos errando foi a de cadastrar produto, isso se deu pelo fato que esse 2 usuários clicaram em outras coisas antes de ir de fato no adicionar, o que para a análise consideramos como um erro, isso significa que o adicionar talvez não esteja tão claro para todos os usuários

Os usuários que erraram coisas no login, foi um erro na credencial(colocaram a senha errado) então percebesse a falta de uma opção de visualizar a senha para conferir se está correta

### 3.3. MÉTRICAS INTERNAS

Para as métricas internas usei uma ferramenta chamada codefactor que para usar basta conectar com o repositório que deseja realizar essas medidas



alguns detalhes sobre o codefactor é essa nota que ele dá para seu código, seguindo o ranking podemos ver como está a qualidade do seu código

CodeFactor uses a modified metric that used letter indicators combined with 1-10 numeric range:

Performance	Letter	Range
Excellent (no issues)	A+	10
Excellent	A	9.4-10
Excellent	A-	9.0-9.3
Good	B+	8.7-8.9
Good	B	8.3-8.6
Good	B-	8.0-8.2
Satisfactory	C+	7.7-7.9
Satisfactory	C	7.3-7.6
Satisfactory	C-	7.0-7.2
Poor	D+	6.7-6.9
Poor	D	6.3-6.6
Poor	D-	6.0-6.2
Failing	F	0-5.9

Fazendo a análise de cada classe obtivemos as métricas colocadas no documentos, como por exemplo a classe que mais tinha linhas de código, tinha 224 sendo assim todas as classes estavam de acordo com a métrica descrita

essa mesma classe também era a que mais tinha métodos, tendo 17 métodos, assim não alcançando o desejável estabelecido, até no codefactor a classe recebe um nota B para a qualidade


app/src/main/java/com/example/Doe/ui/dashboard/DashboardFragment.java


29 Complexity
1.53 Complexity / M
0 Duplication
1 Churn
298 Lines
224 Lines of Code
19 Methods

11.79 LOC / Method

## 4. 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