

**UFRJ**

**ENGENHARIA DE COMPUTAÇÃO E INFORMAÇÃO**

**RELATÓRIO DO TRABALHO FINAL**

**COS482 - QUALIDADE DE SOFTWARE**

**PLANEJAMENTO DE TESTES DO SISTEMA SAFE**

**Alunos:**

<b>Nome</b>	<b>DRE</b>
Carlos Henrique Ferreira Brito Filho	120081409
Enzo Fernandes Vieira	122043774
Guilherme Oliveira Rolim Silva	122076696
João Pedro Regazzi Ferreira da Silva	120030115

**Professor:**

Guilherme Horta Travassos

15 de dezembro de 2025

## Resumo

Este relatório descreve o planejamento de testes do sistema SAFE, com foco na preparação de um conjunto completo de planos de teste, casos de teste e roteiros que permitam a execução posterior por uma equipe externa e independente. O sistema SAFE é um sistema de software IoT composto por subsistemas que incluem um dispositivo de coleta (SAFE BIoT), um broker de mensagens MQTT, um subsistema central de gerência e persistência (SAFE Manager) e um painel de monitoramento (SAFE Dashboard). O trabalho foi conduzido com base na especificação de requisitos do SAFE e no template adotado na disciplina, buscando maximizar a cobertura de requisitos e funcionalidades com esforço mínimo. Como resultado, foram produzidos artefatos finais por subsistema contendo estratégia de testes, casos de teste funcionais e não funcionais e roteiros detalhados. Além disso, esta entrega prevê, para todos os subsistemas, um modelo explícito de registro de resultados de execução, a ser preenchido pelo time externo quando os testes forem executados.

**Palavras-chave:** testes de software, plano de testes, requisitos, IoT, SAFE, verificação e validação.

## Sumário

<b>1</b>	<b>Introdução</b>	<b>3</b>
<b>2</b>	<b>Base do trabalho e artefatos produzidos</b>	<b>3</b>
<b>3</b>	<b>Abordagem metodológica</b>	<b>3</b>
<b>4</b>	<b>Critério de cobertura e priorização</b>	<b>4</b>
<b>5</b>	<b>Visão geral do planejamento por subsistema</b>	<b>4</b>
5.1	SAFE Manager . . . . .	4
5.2	SAFE Dashboard . . . . .	4
5.3	SAFE BIoT . . . . .	4
<b>6</b>	<b>Preparação para execução por equipe externa</b>	<b>5</b>
<b>7</b>	<b>Documentação do uso de modelos de linguagem</b>	<b>5</b>
<b>8</b>	<b>Conclusão</b>	<b>5</b>
<b>A</b>	<b>Lista de artefatos entregues</b>	<b>6</b>
<b>B</b>	<b>Registro de Resultados dos Testes</b>	<b>6</b>
B.1	SAFE BIoT . . . . .	7
B.1.1	Instruções de registro . . . . .	7
B.1.2	Resumo de Testes Funcionais . . . . .	7
B.1.3	Resumo de Testes de Integração . . . . .	7

B.1.4	Resumo de Testes de Regressão . . . . .	7
B.1.5	Critérios de Entrada . . . . .	8
B.1.6	Critérios de Saída . . . . .	8
B.2	SAFE Dashboard . . . . .	9
B.2.1	Registro de execução . . . . .	9
B.2.2	Resumo de Testes Funcionais . . . . .	9
B.2.3	Resumo de Testes Não Funcionais . . . . .	9
B.3	SAFE Manager . . . . .	11
B.3.1	Registro de execução . . . . .	11
B.3.2	Resumo de Testes Funcionais . . . . .	11
B.3.3	Resumo de Testes Não Funcionais . . . . .	11

# 1 Introdução

O sistema SAFE é uma solução desenvolvida no contexto da pandemia da COVID-19 com o objetivo de apoiar o monitoramento do uso seguro das instalações físicas da UFRJ, considerando níveis de risco e condições do ambiente, como temperatura, concentração de CO<sub>2</sub> e número de pessoas por instalação, conforme diretrizes de biossegurança. Arquiteturalmente, o SAFE é composto por subsistemas interconectados por comunicação via MQTT e por integrações entre serviços e interface web, o que caracteriza um cenário típico de software IoT em que parte relevante das entradas e eventos do sistema não depende exclusivamente de uma interface humano-computador.

Neste trabalho, elaboramos planos de teste por subsistema, com casos de teste e roteiros prontos para execução independente, mantendo rastreabilidade entre requisitos e testes e registrando as condições necessárias para que a execução seja reproduzível por um time externo. Todos os artefatos desenvolvidos estão disponíveis no repositório GitHub do projeto, acessível em: <https://github.com/carloshenriquebf/software-quality>.

## 2 Base do trabalho e artefatos produzidos

O planejamento foi elaborado a partir da especificação de requisitos do SAFE e do padrão de documentação adotado na disciplina para planos de teste. A organização do trabalho privilegia rastreabilidade e execução independente, com separação por subsistemas e com casos de teste descritos de forma objetiva, incluindo pré-condições, passos e resultados esperados.

Os artefatos produzidos pelo grupo foram organizados por subsistema, com uma visão geral do sistema e materiais comuns reutilizados na elaboração e revisão dos planos. Foram produzidos um documento geral do sistema SAFE, um documento de persona para padronização de instruções, uma versão do template em Markdown e, para cada subsistema (Manager, Dashboard e BIoT), um documento de requisitos organizado, um plano de testes e um documento final consolidado, além dos prompts utilizados.

## 3 Abordagem metodológica

A abordagem adotada combinou práticas tradicionais de engenharia de testes com uso controlado de modelos de linguagem, com validação humana. O processo seguido foi adaptado de (TRAVASSOS et al., 2025) e foi o seguinte:

1. Definição de uma estrutura comum para os documentos e organização dos materiais por subsistema.
2. Elaboração dos documentos de requisitos por subsistema, de forma coerente com a visão geral do sistema.
3. Construção dos planos de teste e dos casos de teste, incluindo roteiros detalhados e prioridade de execução.

4. Revisão crítica e padronização dos documentos, garantindo consistência, viabilidade e clareza para execução por terceiros.
5. Consolidação final dos artefatos na estrutura de entrega do grupo.

O uso de modelos de linguagem foi aplicado como ferramenta de produtividade para apoiar a redação e a estruturação inicial dos planos, mas a responsabilidade pela qualidade e aplicabilidade dos testes permaneceu com o grupo.

## **4 Critério de cobertura e priorização**

Buscou-se maximizar a cobertura de requisitos e funcionalidades com esforço mínimo por meio de cobertura por requisitos com priorização. Na prática, os documentos finais registram prioridades nos casos de teste, permitindo execução incremental e foco inicial nos cenários essenciais e de maior impacto.

## **5 Visão geral do planejamento por subsistema**

### **5.1 SAFE Manager**

O SAFE Manager é o subsistema central do SAFE, responsável por orquestrar ações de monitoramento, gerenciar usuários, instalações e dispositivos BIoT, processar e armazenar dados ambientais, controlar estados das instalações e se comunicar com o broker MQTT e com o Dashboard. O plano do Manager contempla testes funcionais e testes não funcionais e descreve os casos com pré-condições, passos e resultados esperados, com rastreabilidade por requisito.

### **5.2 SAFE Dashboard**

O SAFE Dashboard é o painel de monitoramento do SAFE, oferecendo interface visual para acompanhamento dos indicadores de biossegurança e estados de utilização das instalações. O plano do Dashboard cobre requisitos funcionais de apresentação de dados e alertas e prevê testes não funcionais relacionados a comunicação com o Manager e comportamento em situações de indisponibilidade e atualização.

### **5.3 SAFE BIoT**

O SAFE BIoT é um dispositivo IoT projetado para coletar dados ambientais em ambientes internos e enviá-los via Wi-Fi e MQTT para o broker. Essa natureza impõe particularidades de teste, pois os dados de entrada são medições de sensores e eventos físicos e a comunicação é assíncrona e dependente de rede. O plano do BIoT contempla testes funcionais, de integração e não funcionais e descreve roteiros voltados a simulação de falhas e validação de publicação contínua.

## 6 Preparação para execução por equipe externa

Como a execução será realizada por um time externo e independente, os artefatos foram preparados com roteiros autoexplicativos, rastreabilidade por requisito e priorização. Para que a entrega esteja efetivamente pronta para execução em todos os subsistemas, também foi previsto um modelo explícito de registro de resultados, contemplando status de execução, data e observações. No subsistema SAFE BIoT, esse modelo já aparece de forma detalhada no documento final, incluindo tabelas de resumo e critérios de entrada e saída. Para os subsistemas SAFE Dashboard e SAFE Manager, adota-se o mesmo padrão de registro, consolidado neste relatório e apresentado no **Anexo B**.

## 7 Documentação do uso de modelos de linguagem

O uso de modelos de linguagem foi documentado por subsistema, incluindo o prompt utilizado e o modelo empregado, permitindo transparência e auditoria.

Subsistema	Modelo utilizado	Artefatos associados
SAFE Manager	ChatGPT	<prompt-manager.md>, <resultado-manager.md>
SAFE Dashboard	DeepSeek	<prompt-dashboard.md>, <resultado-dashboard.md>
SAFE BIoT	Claude	<prompt-biot.md>, <resultado-biot.md>

## 8 Conclusão

Este relatório apresentou o planejamento de testes do sistema SAFE, orientado à execução futura por equipe externa e independente. A estratégia adotada buscou maximizar cobertura de requisitos com esforço mínimo por meio de cobertura por requisitos com priorização, separação por subsistemas e roteiros detalhados. Foram produzidos planos por subsistema, casos de teste funcionais e não funcionais e documentação de prompts e modelos utilizados. Por fim, foi consolidado um modelo padronizado de registro de resultados para apoiar a execução externa e tornar a entrega pronta para execução em todos os subsistemas.

## Referências

TRAVASSOS, G. et al. Lessons learned from the use of generative ai in engineering and quality assurance of a web system for healthcare. In: *Anais do XXIV Simpósio Brasileiro de Qualidade de Software*. Porto Alegre, RS, Brasil: SBC, 2025. p. 278–288. ISSN 0000-0000. Disponível em: <<https://sol.sbc.org.br/index.php/sbqs/article/view/39016>>.

## A Lista de artefatos entregues

A estrutura de arquivos foi organizada por subsistema e por documentos gerais. A seguir, registra-se a lista dos principais artefatos:

- `<geral.md>`: visão geral do sistema SAFE e descrição de subsistemas.
- `<persona.md>`: persona e instruções usadas na elaboração dos planos.
- `<template.md>`: versão do template em Markdown.
- `<requisitos-manager.md>`, `<plano-de-testes-manager.md>`, `<prompt-manager.md>`, `<resultado-manager.md>`.
- `<requisitos-dashboard.md>`, `<plano-de-testes-dashboard.md>`, `<prompt-dashboard.md>`, `<resultado-dashboard.md>`.
- `<requisitos-biot.md>`, `<plano-de-testes-biot.md>`, `<prompt-biot.md>`, `<resultado-biot.md>`.
- `<README.md>`: descrição do contexto, abordagem e organização dos artefatos.

## B Registro de Resultados dos Testes

Esta seção consolida o formato de registro de resultados a ser preenchido após a execução. O objetivo é registrar, de forma padronizada, o que foi executado, quando foi executado e quais observações foram identificadas, preservando rastreabilidade com os casos e requisitos.

## B.1 SAFE BIoT

### B.1.1 Instruções de registro

Para cada tipo de teste e caso de teste, deve ser documentado: data de execução, resultado (Passou/Falhou/Bloqueado), observações e problemas encontrados, ações corretivas tomadas e data da resolução.

### B.1.2 Resumo de Testes Funcionais

ID do Caso	Descrição	Status	Data	Observações
TC_RF01_001	Obtenção dos limites de temperatura, CO <sub>2</sub> e ocupação na inicialização			
TC_RF02_001	Contabilização de entrada de pessoas			
TC_RF02_002	Contabilização de saída de pessoas			
TC_RF03_001	Coleta de temperatura			
TC_RF04_001	Coleta de CO <sub>2</sub> (eCO <sub>2</sub> )			
TC_RF05_001	Apresentação do estado da instalação			
TC_RF06_001	Coleta de umidade			
TC_RF07_001	Coleta de VOCs			

### B.1.3 Resumo de Testes de Integração

ID do Caso	Descrição	Status	Data	Observações
TC_RNF2_001	Formato JSON das mensagens SAFE_IAQ			
TC_RNF10_001	Intervalo de publicação de dados de qualidade do ar			
TC_RNF6_001	Retomada automática após perda de energia			
TC_RNF7_001	Reconexão automática à internet			
TC_RNF9_001	Armazenamento local em caso de falha de conexão			
TC_RNF13_001	Coleta contínua durante a semana			
TC_RNF22_001	Reportagem de falhas através de etiquetas de erro			

### B.1.4 Resumo de Testes de Regressão

ID do Caso	Descrição	Status	Data	Observações
TC_REG_001	Suite de regressão – Testes críticos			



### **B.1.5 Critérios de Entrada**

Para iniciar os testes de regressão e não funcional, os seguintes critérios devem ser atendidos:

- [ ] Todos os testes funcionais foram executados e aprovados
- [ ] Todos os testes de integração foram executados e aprovados
- [ ] Nenhum defeito crítico está pendente
- [ ] O firmware foi compilado com sucesso e está pronto para teste
- [ ] Ambiente de teste está totalmente configurado e validado

### **B.1.6 Critérios de Saída**

Para considerar os testes de regressão e não funcional como concluídos com sucesso:

- [ ] Todos os casos de teste essenciais foram executados
- [ ] Taxa de sucesso mínima de 95% em todos os testes
- [ ] Taxa de sucesso em testes de longa duração (RNF13)  $\geq 99\%$
- [ ] Nenhum defeito crítico está aberto
- [ ] Número aceitável de defeitos maiores: máximo 2 abertos
- [ ] Todos os defeitos corrigidos foram validados por regressão
- [ ] Documentação de testes está completa
- [ ] Concordância da equipe sobre a qualidade do produto para deploy
- [ ] Testes de regressão final passaram na versão de produção

## B.2 SAFE Dashboard

### B.2.1 Registro de execução

Para cada caso de teste executado no SAFE Dashboard, devem ser registrados: data de execução, resultado (Passou/Falhou/Bloqueado), observações e problemas encontrados, bem como ações corretivas quando aplicável.

### B.2.2 Resumo de Testes Funcionais

ID do Caso	Descrição	Status	Data	Observações
TC_RF01	Verificar exibição da temperatura por instalação			
TC_RF02	Verificar exibição de CO <sub>2</sub> em ppm			
TC_RF03	Verificar exibição do número de pessoas			
TC_RF04	Verificar classificação de risco			
TC_RF05	Verificar estado de utilização			
TC_RF06	Verificar localização e unidade			
TC_RF07	Verificar gráficos de evolução na última hora			
TC_RF08	Verificar valores máximos na última hora			
TC_RF09	Verificar exibição de umidade			
TC_RF10	Verificar exibição de VOCs			
TC_RF11	Verificar alerta de limite atingido			
TC_RF12	Verificar legenda no rodapé			

### B.2.3 Resumo de Testes Não Funcionais

ID do Caso	Descrição	Status	Data	Observações
TC_RNF01	Verificar comunicação com API do Manager			
TC_RNF02	Verificar obtenção de limites de biossegurança			
TC_RNF05	Verificar atualização de dados a cada INTER-VALO_ATUALIZAR_DADOS			
TC_RNF06	Verificar atualização de limites a cada INTER-VALO_ATUALIZACAO_LIMITES			
TC_RNF07	Verificar atualização de estados a cada INTER-VALO_ATUALIZACAO_ESTADO			

TC_USB01	Verificar responsividade em diferentes telas
TC_USB02	Verificar organização em cards
TC_SEC01	Verificar alerta de dados desatualizados
TC_SEC02	Verificar alerta de indisponibilidade do Manager

---

## B.3 SAFE Manager

### B.3.1 Registro de execução

Para cada caso de teste executado no SAFE Manager, devem ser registrados: data de execução, resultado (Passou/Falhou/Bloqueado), observações e problemas encontrados, bem como ações corretivas quando aplicável.

### B.3.2 Resumo de Testes Funcionais

ID do Caso	Descrição	Status	Data	Observações
TC_RF01	Gerenciamento de Gestores das Instalações			
TC_RF02	Gerenciamento de Dispositivos BIoT			
TC_RF03	Gerenciamento de Instalações			
TC_RF04	Solicitação de Manutenção			
TC_RF05	Notificação de Manutenção			
TC_RF08	Solicitação de Limpeza			
TC_RF12	Solicitação de Uso da Instalação			
TC_RF15	Atualização do Estado da Instalação			
TC_RF17	Login no Sistema			
TC_RF18	Alteração de Senha			

### B.3.3 Resumo de Testes Não Funcionais

ID do Caso	Descrição	Status	Data	Observações
TC_RNF01	Comunicação com Broker MQTT			
TC_RNF05	Alerta de Falta de Dados			
TC_RNF09	Disponibilidade do Sistema			
TC_RNF15	Encerramento Automático de Sessão			
TC_RNF18	Responsividade da Interface			