

---

**Plano de Teste**

**Hoje Não!**

**Versão 1.0**

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

## Histórico de Alterações

Data	Versão	Descrição	Autor

## Sumário

1. Introdução .....	3
1.1 Finalidade .....	3
1.2 Escopo.....	3
1.3 Público-alvo .....	4
1.4 Terminologia e Acrônimos do Documento .....	4
1.5 Referências .....	4
1.6 Estrutura do Documento .....	4
2. Missão de Avaliação e Motivação dos Testes .....	5
2.1 Informações Detalhadas .....	5
2.2 Missão de Avaliação .....	5
2.3 Motivadores dos Testes .....	5
3. Itens-alvo dos Testes .....	6
3.1 Teste de Configuração.....	6
3.2 Teste de Unidade.....	6
3.3 Teste de Carga .....	6
3.4 Teste de Usabilidade e compatibilidade .....	6
3.5 Teste de Banco de Dados .....	6
3.6 Teste de Segurança.....	6
4. Resumo dos Testes Planejados.....	6
4.1 Resumo das Inclusões dos Testes.....	7
5.1 Técnicas e Tipos de Teste.....	8
6. Critérios de Entrada e de Saída .....	10
6.1 Plano de Teste .....	10
6.2 Ciclos de Teste .....	10
7. Fluxo de Trabalho de Teste .....	11
8. Necessidades Ambientais .....	11
8.1 Hardware Básico do Sistema.....	11
8.2 Elementos de Softwares Básicos do Ambiente de Teste .....	12
8.3 Configurações do Ambiente de Teste .....	12
9. Responsabilidades, Perfil da Equipe e Necessidades de Treinamento .....	12
9.1 Pessoas e Papéis .....	12

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

# Plano de Teste

## 1. Introdução

### 1.1 Finalidade

A finalidade do Plano de Teste de Iteração é reunir todas as informações necessárias ao planejamento e ao controle do esforço de teste referente a uma iteração específica. Ele descreve a abordagem dada ao teste do software é o plano gerado e usado pelos gerentes para coordenar o esforço de teste.

Este Plano de Teste referente ao Hoje Não! Onde suporta os seguintes objetivos:

- Elucidar o testador quanto as ferramentas necessárias para a realização dos testes cabíveis;
- Identificar erros e assegurar suas correções tanto no sistema quanto na sua documentação;
- Identificar os requisitos para o testador realizar os devidos testes com a finalidade de melhorias;
- Mostrar inúmeras formas aos testadores para a utilização dos testes;
- Identificar os resultados dos testes
- Explicar os recursos necessários para a execução dos testes;
- Adquirir estimativa dos esforços de teste;
- Verificar possíveis riscos por meios dos testes;

### 1.2 Escopo

O Projeto Hoje Não!, tem como objetivo criar um aplicativo para facilitar a exposição de provas e facilitar nas denúncias. Uns de seus principais objetivos e a rápida comunicação com o DP (Departamento de Polícia) mais próximo que receberá um relatório para atender os pedidos de socorro do Usuário. Assim visando a diminuição ou tentativa de violência contra as mulheres. Onde ele vai ter a necessidade de fazer testes de unidade, integração do sistema. Os testes unitários e de integração vai lidar com a qualidade funcional, base de dados, interface gráfica e controle de acesso, já o teste do sistema irá verificar o funcionamento, desempenho, e a usabilidade do sistema.

Durante todo o processo, poderá haver mudanças de acordo com o ciclo de vida do software, como inclusões dos requisitos ou até mesmo exclusões.

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

### 1.3 Público-alvo

Este documento de plano de teste é destinado aos seguintes envolvidos, o Gerente de Testes, Analista de Sistema, Testador e Gerente de Projeto, com o intuito de orienta-los nos testes a serem desenvolvidos. Onde cada um tem sua responsabilidade e sua parte ao todo, focando nos objetivos do projeto Hoje Não!

### 1.4 Terminologia e Acrônimos do Documento

As definições utilizadas neste documento do sistema Hoje Não! Serão abordadas posteriormente no glossário.

Termo	Descrição
RF	Requisito funcional
RNF	Requisito não funcional
MySQL	

### 1.5 Referências

- Documento de Visão: Descrição do Sistema HOJE NÃO!
- Documento de Requisitos: Levantamento de Requisitos do Sistema

### 1.6 Estrutura do Documento

- As seções seguintes estão organizadas como descritas abaixo:

Parte 1 – Introdução:

Parte 2: Missão de Avaliação e Motivação dos testes.

Parte 3: Itens-alvo dos Testes.

Parte 4: Resumo dos Testes Planejados.

Parte 5: Abordagem dos Testes:

Parte 6: Catálogos Iniciais de Ideias de Teste e outras Fontes de Referência.

Parte 7: Critérios de Entrada e de Saída.

Parte 8: Produtos Liberados.

Parte 9: Fluxo de Trabalho de Teste.

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

Parte 10: Necessidades Ambientais.

Parte 11: Responsabilidades, Perfil da Equipe e Necessidades de Treinamento.

Parte 12: Marcos as Interação.

Parte 13: Riscos, Dependências, Suposições e Restrições.

Parte 14: Procedimento e Processos de Gerenciamento.

## **2. Missão de Avaliação e Motivação dos Testes**

O Teste de software estabelece de um procedimento realizado pelo testador, do qual tem como objetivo detectar possíveis falhas que venham apresentar durante todo o processo no sistema, sendo assim passando a ser corrigidos o mais rápido possível, para obter uma maior qualidade no produto em desenvolvimento podendo ocorrer maior satisfação do cliente com o mesmo, analisar ou verificar se todos os requisitos estão de acordos, e se o que foi pedido está condicente com o resultado final esperado.

### **2.1 Informações Detalhadas**

O principal problema está na dificuldade de agilidades nos processos, falta de segurança e, contudo, afetando muito a mão de obra na instituição, acarretando em perda de tempo e podendo até causar perdas de informações e podendo gerar vários outros problemas sérios. Com o decorrer, os resultados, busca e as atividades estão cada vez mais precisas tornando-o para o profissional uma grande dificuldade e um mal desempenho no seu desenvolvimento.

Uma boa solução seria desenvolver um sistema que possa agilizar todos os processos da instituição, para conseguir minimizar todos os métodos tradicionais oferecendo mais segurança e mais precisão nas informações. Obtendo de forma mais rápida e eficaz tendo um maior resultado nas suas tarefas ou consultas.

### **2.2 Missão de Avaliação**

No teste de software tem como objetivo:

- Localizar o maior número de erros possíveis, e em seguida corrigindo com habilidade, diminuindo ao máximo possível os custos que foram nos reparos anteriores.
- Conferir e verificando se toda as documentações do sistema estão de acordo com a necessidade do cliente, propondo maior qualidade nos processos e no produto final.

### **2.3 Motivadores dos Testes**

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

O teste de software é muito importante, para se ter um bom sucesso desse projeto, pois ele contém uma grande responsabilidade pela qualidade do produto e garanti que todo o projeto está de acordo com a documentação está descrita. O testador verifica e executa todas as suas tarefas com o intuito de corrigir possíveis falhas que possa haver no projeto, evitando retrabalhos para o futuro. Com a conclusão do projeto ele deve estar de forma correta e dentro do prazo e como foi planejado, trazendo uma boa satisfação ao cliente.

### 3. Itens-alvo dos Testes

A listagem abaixo identifica os itens software, hardware e elementos de suporte do produto que foram identificados como alvo dos testes. Essa lista representa os itens que serão testados.

#### 3.1 Teste de Configuração

- Fazer uma análise e depois verificar se todas as funcionalidades do sistema está funcionando corretamente em pelo menos um ou dois navegadores.
- Fazer uma análise e depois verificar se o sistema está executando as configurações mínimas de hardware;

#### 3.2 Teste de Unidade

- verificar cada componente do sistema para ver se o seu funcionamento está funcionando de forma correta.

#### 3.3 Teste de Carga

- Verificar e Avaliar o desempenho do sistema com 1 usuário conectado no mesmo;
- Verificar e Avaliar o desempenho do sistema com 10 usuários conectados no mesmo;

#### 3.4 Teste de Usabilidade e compatibilidade

- Fazer o teste se a interface do ambiente se adapta a diferentes resoluções de tela e diferentes tipos de navegadores;

#### 3.5 Teste de Banco de Dados

- Analisar se o banco de dados não possui informações inseguros ou incoerentes;
- Testado para verificar integridade do banco de dados;

#### 3.6 Teste de Segurança

- Verificar se tem a possibilidade de captura de dados de uma transação, ou que seja disponível em tela, e ter acesso ao banco de dados por MySQL.

### 4. Resumo dos Testes Planejados

Todos os testes mostrados aqui serão executados, os que não serão executados não precisa ser aplicado.

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

#### 4.1 Resumo das Inclusões dos Testes

Principais testes planejados para a interação atual:

- Teste de caixa branca;
- Teste de caixa preta;
- Teste de carga;
- Teste de unidade;
- Teste de usabilidade.

#### 5. Abordagem dos Testes:

Há abordagem de teste será realizada manual e/ou automatizada, e uma profunda análise a ser realizada. Em outras palavras ela define se o que interessa e somente respostas geradas pelo item em teste e o seu comportamento.

##### Teste de configuração:

Verifica a operação do sistema em diferentes configurações de software e hardware para que haja um bom desempenho em todas as funcionalidades de software e hardware, evitando mudanças em questão, sem afetar o desempenho do software a ser desenvolvido. Essa técnica verifica se o alvo de teste funciona corretamente nas configurações exigidas pelo software e hardware. Faz parte desse teste o sistema operacional de rede, os servidores de redes, banco de dados e outros recursos.

##### Teste de unidade:

Tem por objetivo a especificação de explorar a menor unidade do projeto procurando por defeitos ou falhas de lógicas implementadas em cada módulo individualmente, são testados os métodos dos objetos e/ou pequenos trechos de códigos. Isso validando todas as assinaturas de entradas e saídas sendo elas válidas ou inválidas garantindo que não haverá falhas no sistema.

##### Teste de usabilidade e compatibilidade:

Verifica se o sistema possui uma boa usabilidade para todos os públicos, se adaptando a telas de diversas resoluções, rodando em diversos navegadores em diversos sistemas operacionais, sendo compatível assim com o máximo de dispositivos possível.

##### Teste de carga:

Verificar o comportamento do sistema após o mesmo ser submetido a uma certa quantidade de dados, verificar se os dados inseridos de forma continuam e automatizada foram realmente inseridos no banco de dados.

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

### Teste de segurança:

O objetivo é garantir a segurança do funcionamento do sistema averiguando o seu comportamento mediante às diversas solicitações de acesso ilegais nos três ambientes, visando a existência de vulnerabilidades testando os mecanismos de proteção inclusos no sistema. Perante a esse teste, irá ver a possibilidade de dúvidas sobre prováveis falhas de vulnerabilidades do software a serem eliminadas.

## 5.1 Técnicas e Tipos de Teste

### 5.1.1 Teste de Integridade de Dados e de Banco de Dados

Objetivo da Técnica:	Assegurar que todos métodos e processos de acesso ao Banco de Dados e a integridade dos dados funcionem de forma correta e conseguindo um desempenho aceitável e que de forma alguma apresente nenhuma inconsistência de dados. O principal objetivo é executar todos métodos na base de dados e a regra de negócio é ser de total independência da interface do usuário, portanto, possibilitando a observação dos registros do comportamento do mesmo mediante aos testes.
Técnica:	Haverá um conjunto de testes na base de dados como: Carga, e desempenho.
Estratégias:	Construir de forma organizada a estrutura do sistema junto com a execução dos testes para encontrar erros que venham a comprometer o andamento do mesmo.
Critérios de Êxito:	A base de dados deverá fazer transações sem que haja perda de performance, com os dados 100% perfeitos, sem vestígios de incoerência de dados.
Considerações Especiais:	Para a modificação diretamente na base de dados será utilizado SGBD como PHPMyAdmin, proporcionando assim um melhor acompanhamento e manuseio do mesmo, fazendo testes lentamente, com o objetivo de encontrar possíveis falhas no mesmo durante todo o processo do teste.

### 5.1.2 Teste de Função

Objetivo da Técnica:	O principal objetivo do teste tem como apoio os requisitos funcionais do sistema para que possa verificar-se a consistência de dados da interface testando-as todas as entradas e analisar como o sistema está agindo, de forma incorreta serão testadas as entradas para analisar se não haverá nenhuma falha de entrada de dados.
----------------------	---



Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

Estratégias:	As técnicas executadas têm como garantir o desempenho do sistema diante dos processos validando o funcionamento correto, no entanto, os dados de entradas são fornecidos, o teste será executado e o resultado obtido é comparado a um resultado aguardado. Todos os componentes testados pode ser um método, uma função, um componente etc.
Ferramentas Necessárias:	Teste funcional ou teste de caixa preta.
Crêterios de Êxito:	Ao cumprir-se esse teste, o testador deve ter como início o guia de casos de teste ou as especificações de casos de uso, neles estarão todos expostos detalhadamente todo o comportamento do sistema, onde deverá buscar entender erros que o usuário poderá cometer ou que fogem da especificação.
Considerações Especiais:	Assegurar que todo o desempenho ou funcionalidade do sistema para que o usuário final se sinta confortável e que o manuseio do sistema será sem falhas de interfaces.

#### 5.1.3 Teste da Interface do Usuário

Objetivo da Técnica:	O objetivo é analisar a usabilidade, acessibilidade da interface em que o sistema disponibiliza para o usuário, oferecendo elementos visuais de total facilidade de manuseio e aprendizado, proporcionando melhor manuseio do sistema e dos recursos que ele oferece.
Técnica:	Serão utilizadas aplicações web voltadas a teste de interface bem como teste de compatibilidade e expansividade.
Estratégias:	Avaliando a usabilidade do sistema durante todo o processo do desenvolvimento a fim de fazer alterações necessárias para otimizar a usabilidade do sistema antes da conclusão.
Crêterios de Êxito:	Para ter uma maior interação com o sistema tem que haver ambientes aceitáveis pelo usuário final e que proporcione maior usabilidade e acessibilidade do mesmo.
Considerações Especiais:	Depois da conclusão o sistema tem por obrigação passe por todos os testes onde validará todo seu desempenho, usabilidade e acessibilidade para que o usuário se sinta confortável.

#### 5.1.4 Teste de Carga

Objetivo da Técnica:	Assegurar que o sistema funcione com certa quantidade de dados sendo inserida de forma simultânea e continua.
----------------------	---

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

Técnica:	Será feito os testes de carga e de performance para ver se o sistema pode suportar um alto índice que será utilizado por vários usuários ao mesmo tempo.
Estratégias:	Verificar a capacidade do sistema durante todo o processo de desenvolvimento, podendo assim ter um aumento na escala do sistema antes da conclusão.
Critérios de Êxito:	10 usuários conectados ao mesmo tempo no sistema sem que apresente perda de performance.
Considerações Especiais:	Garantir que durante todo o processo tenha um bom desempenho no processamento utilizado por 10 usuários, que não venha aparecer a corrupção de dados ou perda.

## 6. Critérios de Entrada e de Saída

### 6.1 Plano de Teste

#### 6.1.1 Critérios de Entrada de Plano de Teste

O plano de teste inicializará ao término da implementação dos requisitos especificados no documento de requisitos.

#### 6.1.2 Critérios de Saída de Plano de Teste

A conclusão dos testes se dará quando todos os requisitos forem testados, e não podendo ser encontrado erros (bugs). Em casos de erros no processo, os testes serão suspensos até que os problemas sejam concluídos e assim possamos continuar com os testes.

#### 6.1.3 Critérios de Suspensão e de Reinício

No caso de problemas que impossibilitem a execução da lista de requisitos a serem testados, será suspenso os testes até que se solucione o problema, só assim os demais requisitos serão novamente testados e validados. Caso tenha um problema que possa continuar com teste, terá continuidade, mas quando solucionar o problema o teste será realizado novamente. Quando os todos requisitos forem testados sem nenhum erro, os mesmos serão dados como concluídos, e assim será feito até que termine toda lista com sucesso.

### 6.2 Ciclos de Teste

#### 6.2.1 Critérios de Entrada de Ciclo de Teste

O ciclo de teste seguirá a ordem especificada do plano de testes.

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

### 6.2.2 Critérios de Saída de Ciclo de Teste

Os testes deverão passar por os ciclos, sendo que todos os defeitos e exceções terão que estar corrigidos.

### 6.2.3 Término Anormal do Ciclo de Teste

Só ocorrerá se tiver falhas no planejamento do plano de teste.

## 7. Fluxo de Trabalho de Teste

Todos os resultados dos testes devem ser registrados em um documento que deverá conter um histórico para futuras análises.

## 8. Necessidades Ambientais

Os conjuntos de tabelas a seguir apresentam os recursos do sistema necessários ao esforço de teste descrito neste Plano de Teste.

Recursos do Sistema		
Recurso	Quantidade	Nome e Tipo
Nome do Banco de Dados	01	formulario
Nome do Servidor	01	Localhost / Apache
PCs de Desenvolvimento de Teste	02	Notebook
Rede ou Sub-rede	01	DHCP
Servidor de Banco de Dados	01	MySQL

### 8.1 Hardware Básico do Sistema

Os conjuntos de tabelas a seguir apresentam os recursos do sistema necessários ao esforço de teste descrito neste Plano de Teste.

Recursos do Sistema		
Recurso	Quantidade	Nome e Tipo
Notebook	2	Samsung Intel Core-i3, HD 1T, Processador Intel(R) Core(TM) i3-1070 E CPU @ 2.20GHz 2,20 GHz RAM: 4,00GB
		Samsung Intel Core-i3, HD 500GB, Processador Intel(R) Core(TM) i3-270 E CPU @ 2.20GHz 2,20 GHz RAM: 4,00GB
Nome do Servidor	01	Localhost / Apache
Rede ou Sub-rede	01	DHCP

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

Servidor de Banco de Dados	01	MySQL
----------------------------	----	-------

## 8.2 Elementos de Softwares Básicos do Ambiente de Teste

São necessários os seguintes elementos de softwares básicos no ambiente de teste deste *Plano de Teste*.

Nome do Elemento de Software	Versão	Tipo e Outras Observações
Windows 10	6.3	Sistema Operacional
Chrome	56	Navegador da Internet

## 8.3 Configurações do Ambiente de Teste

Devem ser fornecidas e suportadas as seguintes Configurações de Ambiente de Teste para este projeto.

Nome da Configuração	Descrição	Implementada na Configuração Física
Configuração do usuário comum	Terminal de uso com navegador Firefox	1 Computador, com máquinas virtuais implantadas
Mínima configuração suportada	Configuração mínima para garantir um bom funcionamento.	HD de 8 GB e 512MB de RAM, Firefox 12.0

## 9. Responsabilidades, Perfil da Equipe e Necessidades de Treinamento

### 9.1 Pessoas e Papéis

Esta tabela mostra as suposições referentes ao perfil da equipe do esforço de teste.

Recursos Humanos		
Papel	Recursos Mínimos Recomendáveis	Responsabilidades ou Comentários Específicos

Hoje_Não!	Versão: 1.0
<Iteração/Mestre> Plano de Teste	Data:05/11/2021
HojeNão!_Plano_Testes	

Testador	João Victor Penha Alves e Carlos Ariel	<ul style="list-style-type: none"> <li>• adquirir recursos apropriados</li> <li>• avaliar a eficiência do esforço de teste</li> <li>• determinar os resultados dos testes</li> <li>• definir a arquitetura de automação de teste</li> <li>• definir os elementos de testes</li> <li>• estruturar a implementação dos testes</li> </ul>
Analista de Sistema	João Victor Penha Alves e Carlos Ariel	<ul style="list-style-type: none"> <li>• documentação Solicitação da Mudança</li> <li>• revisão e Implementação caso haja mudanças</li> <li>• necessária na documentação.</li> </ul>
Gerente de Projeto	João Victor Penha Alves e Carlos Ariel	<ul style="list-style-type: none"> <li>• verificar as técnicas de teste</li> <li>• identificar ideias de teste</li> <li>• combinar missão</li> <li>• identificar motivadores</li> <li>• defender os interesses do teste</li> </ul>
Analista de Sistema	João Victor Penha Alves e Carlos Ariel	<ul style="list-style-type: none"> <li>• documentar solicitações de mudança</li> <li>• avaliar a qualidade do produto</li> </ul>
Testador	João Victor Penha Alves e Carlos Ariel	<ul style="list-style-type: none"> <li>• implementar os testes e os conjuntos de testes</li> <li>• executar os conjuntos de testes</li> <li>• registrar os resultados</li> <li>• analisar as falhas dos testes e possibilitar a recuperação posterior documentar incidentes</li> </ul>