

**PROJETO INOVAR**

**Análise Crítica do Plano de Testes**

Desenvolvimento de Sistemas

|  |  |
| --- | --- |
| **Professora:**  Roselene Fenske | **Equipe:**  Ana Clara de Souza  Eberth da Silva Rodrigues  Kaio Mazza  Rafael De Almeida de Magalhões |

**Junho/ 2025**

# Histórico de Revisões

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** |  | **Autor** |
| <30/jun/25> | <1.0> | Release Inicial | Eberth,Kaio,Rafe e Ana |  |

# Introdução & Objetivos

O sistema INOVAR (TIG – *Tourist Information Guide*) é uma solução desenvolvida com o objetivo de oferecer suporte informacional a turistas hospedados em hotéis parceiros. Por meio do sistema, os usuários poderão acessar dados relevantes sobre pontos turísticos, eventos locais, programações culturais, serviços disponíveis na cidade e informações pessoais, como o extrato de sua conta no hotel, incluindo consumo, ligações e diárias.

A aplicação será implantada localmente nos hotéis, operando através da intranet, o que garante maior controle sobre o ambiente de execução e reduz dependências externas, como conexões com a internet pública. A arquitetura do sistema também inclui integração com a base de dados interna do hotel, o que exige atenção redobrada com relação à segurança, à consistência dos dados e à performance das transações.

Dada a natureza crítica dessas funcionalidades — especialmente porque o sistema será utilizado diretamente por hóspedes e pela equipe administrativa dos hotéis —, a elaboração de um **plano de testes bem estruturado e abrangente** é indispensável. Esse plano deve garantir que todos os componentes do sistema operem corretamente em cenários reais, com foco na confiabilidade, usabilidade, segurança da informação e desempenho sob diferentes condições de uso.

Além disso, a validação por meio de testes proporciona segurança para o processo de implantação e manutenção do sistema, permitindo que erros sejam identificados e corrigidos de forma antecipada, reduzindo riscos operacionais e melhorando a satisfação dos usuários finais.

**Principais Objetivos**:

* Apoiar Turistas.
* Obter fluxo de informações entre os clientes para lucros dos clientes.
* Acomodação de clientes no local.
* Facilitar a navegação de turistas no sistema dando apoio legível ao cliente.
* Estabilidade, funcionalidade e segurança do sistema no ambiente real.
* Fluxo rápido entre o cliente e o sistema local.
* Verificar a integridade e consistência dos dados.
* Avaliar o desempenho e a escalabilidade (imagens e vídeos).
* Verificar a correta instalação e distribuição do sistema.

**Objetivos e Foco dos Tipos de Teste**

|  |  |  |
| --- | --- | --- |
| **Objetivos e Foco dos Tipos de Teste** | | |
| **Tipo de Teste** | **Tipo de Teste** | **Tipo de Teste** |
| Banco de Dado | Integridade e consistência | Inserção, atualização e recuperação de dados (turismo e hóspedes) |
| Funcional | Conformidade com os casos de uso | Fluxo de ações dos usuários e regras de negócio |
| Interface do Usuário | Usabilidade e navegação | Navegação, aparência, idiomas, acessibilidade |
| Performance | Resposta sob uso normal e máximo | Tempos de resposta (principalmente com dados multimídia) |
| Carga | Estabilidade sob múltiplos acessos | Comportamento com 10, 50, 100, 200 e 500 usuários |
| Segurança e Controle de Acesso | Restrição de funcionalidades | Garantir que usuários vejam apenas o que lhes é permitido |
| Instalação | Compatibilidade e replicabilidade | Instalar e validar funcionamento em máquinas novas e atualizadas |

# 2. Requisitos a Testar

A lista abaixo identifica aqueles itens – use cases, requisitos funcionais e não funcionais – que foram identificados como alvos de teste. Essa lista representa o que será testado.

## 2.1 Teste do Banco de Dados:

### 2.1.1 Operações Básicas (CRUD)

* Verificar se as informações podem ser cadastradas, alteradas e excluídas corretamente.
* Verificar se é possível realizar consultas com diferentes filtros (por cidade, local, gênero do evento, data, horário, etc).
* Verificar se eventos, pontos turísticos, hospedagens e cupons podem ser inseridos, atualizados e removidos.
* Verificar se os administradores e organizadores podem cadastrar novos eventos e hospedagens.
* Verificar se é possível editar os dados pessoais dos usuários.

### 2.1.2 Agrupamentos e Categorias

* Verificar se os eventos e hospedagens podem ser agrupados por categorias, como cidade, tipo de evento, avaliação etc.
* Verificar se os locais e eventos podem ser classificados e filtrados corretamente.
  + 1. **Validação de Dados**
* Verificar se as informações estão dentro dos formatos e padrões exigidos (ex: CPF no formato XXX.XXX.XXX-XX, data no formato DD/MM/AAAA).
* Verificar se o usuário pode alterar o proprietário de ingresso/hospedagem, respeitando o limite de 3 vezes.
* Verificar se o sistema realiza o reembolso automático em caso de cancelamento de evento ou hospedagem.
  + 1. **Consulta e Visualização de Dados**
* Verificar se o usuário pode consultar informações de eventos, hospedagens e pontos turísticos.
* Verificar se o usuário pode consultar ingressos e reservas disponíveis.
* Verificar se o usuário pode consultar cupons de desconto válidos.
* Verificar se o usuário pode consultar detalhes de sua reserva feita (evento ou hospedagem).
  + 1. **Atualização e Sincronização**
* Verificar se as informações de dados de usuários, eventos, hotéis e pontos turísticos estão sempre atualizadas.
* Verificar se as informações de ingressos e hospedagens estão sempre atualizadas.
  + 1. **Acesso e Login**
* Verificar divisão de usuários e administradores.
* Verificar se usuários e administradores podem acessar o sistema com login e senha.
* Validar o controle de acesso ao banco de dados de acordo com o perfil do usuário.
  + 1. **Confiabilidade e Desempenho**
* Verificar a disponibilidade e capacidade do banco de dados para suportar múltiplas requisições simultâneas.
* Verificar consistência dos dados relacionados a eventos, usuários e hospedagens (ex: ingresso ainda disponível após uma compra).

**2.2 Teste Funcional**

**2.2.1 Login e Senha**

* Verificar se o login com informações validas permite acesso.
* Verificar se o login com informações invalidas retorna mensagem de erro.
* Verificar se o usuário é redirecionado corretamente após login.
* Verificar se a recuperação de senha funciona via e-mail, ou telefone de login.
* Verificar se chaves de acesso (nome de usuário, e-mail e telefone) para login funcionam corretamente.
  + 1. **Cadastro e Gerenciamento de Usuários**
* Verificar se é possível cadastrar novos usuários, e se o cadastro funciona com dados válidos.
* Verificar se o sistema impede dados inválidos (ex: Formato de e-mail e CPF invalido).
* Verificar se o usuário pode alterar seus dados pessoais.
* Verificar se o usuário pode alterar seu login e senha.
* Verificar mensagens de erros e confirmações através de SMS e e-mail.
* Verificar se o usuário pode excluir sua conta.
* Verificar se o usuário é maior de 16 anos.
  + 1. **Reserva de Eventos e Hotéis**
* Verificar se o usuário pode realizar reservas de eventos ou hospedagens.
* Verificar se o usuário pode alterar o nome do proprietário da reserva (passar para outro nome/pessoa, respeitando o limite de 3 vezes).
* Verificar se o sistema atualiza o número de vagas disponíveis após uma reserva.
* Verificar se o cancelamento de uma reserva está funcionando em conjunto com seu devido reembolso automático.
* Verificar se o sistema impede reserva duplicada
  + 1. **Pagamento e Cupons**
* Verificar se o sistema aceita cupons válidos e se ele aplica o desconto.
* Verificar se o sistema rejeita cupons inválidos.
* Verificar se o sistema permite e efetua pagamento com diferentes formas de pagamento (pix, boleto, catão etc).
* Verificar se o sistema gera o meio de pagamento (ex: chave pix e boleto).
* Verificar se o sistema gera comprovante após pagamento e se ele notifica o usuário, através de notificação e mensagens via e-mail e SMS
  + 1. **Diferentes Acessos**
* Verificar se o administrador tem acesso a funcionalidades exclusivas (cadastrar eventos, ver estáticas, cancelar ou remover informações e dados, cadastrar um usuário (caso atendimento presencialmente), alterações de dados e informações, etc).
* Verificar se o usuário comum não consegue acessar funcionalidades administrativas.
* Verificar se o organizador do evento pode editar somente seus eventos.

**2.3 Interface**

**2.3.1 Navegabilidade e Usabilidade**

* Verificar se o sistema permite livre navegabilidade entre todas as informações e ações disponíveis (conta, eventos, hotéis, pontos turísticos etc).
* Verificar se o sistema permite traduções de diferentes línguas e com isso se ele funciona corretamente.
* Verificar se o sistema permite que o usuário pode realizar e visualizar um “carrinho”.
* Verificar se o sistema permite que o usuário pode visualizar seus gastos em hotéis e eventos.
* Verificar se o sistema permite que o usuário visualize suas compras e gastos passados.
* Verificar se o sistema pode gerar arquivos PDF para compras, e para gastos.
* Verificar se todos os botões e links funcionam corretamente.
* Verificar se menus e submenus estão acessíveis e usáveis.
* Verificar ações do sistema com determinado ato.
* Verificar se o sistema tem um suporte de ajuda.

**2.3.2 Adaptação para diferentes telas**

* Verificar se o sistema se adapta em diferentes tamanhos de telas (computador, tablet, celular).
* Verificar se todos os elementos ficam visíveis em resoluções menores, e se não ficam improporcional em resoluções menores ou maiores.
* Verificar se não há quebra de layout (imagens distorcidas, textos sobrepostos etc).

**2.3.3 Visual**

* Verificar se há hover aonde deve-se ter.
* Verificar se o sistema está de fácil manuseio e se as informações estão claras.
* Verificar se os textos, botões e imagens estão visíveis e não estão distorcidas.
* Verificar paleta de cores do sistema (para um visual bonito, e não causar dores de vista, cabeça ou tontura nos usuários).
* Verificar se o sistema está padronizado.
* Verificar se determinadas informações estão em destaque com seus tamanhos de fontes e cores destacadas (ex: Valor de hospedagem ou evento estar em destaque).

**2.3.4 Mensagens de Erro e Validação**

* Verificar se ao deixar um campo obrigatório vazio, o sistema exibe uma mensagem clara informando ao usuário a necessidade de preencher o campo.
* Verificar se as mensagens de erro estão visíveis e compreensíveis.
* Verificar se as mensagens de erro estão no local correto.
* Verificar se o sistema só permite determinada ação quando todos os campos tiverem preenchidos ou quando todas as necessidades forem compridas.

**2.3.5 Interações com o Usuário**

* Verificar se carregamentos tem indicador visual (ex: ícone de carregando).
* Verificar se há confirmações após ações importantes (ex: reservas, cancelamento, alterações, pagamento etc).
* Verificar se sistema envia para o usuário notificações e mensagens via SMS e e-mail nas funcionalidades onde são exigidas.

**2.4 Performance**

**2.4.1 Tempo de Resposta**

* Verificar tempo de resposta para carregar as telas.
* Verificar tempo de resposta para realizar login.
* Verificar tempo de resposta para realizar cadastro.
* Verificar o tempo de resposta para exibir a lista de eventos, hotéis, pontos turísticos etc.
* Verificar tempo de resposta para finalizar um pagamento.
* Verificar tempo de resposta para carregar mídia (imagens, mapas, vídeos etc).
* Verificar tempo de resposta para uma geração de PDF.
* Verificar tempo de resposta para um suporte.
* Verificar se tempo de resposta para ações comuns (ex: clicar em um botão, trocar de pagina etc) não passam de 3 segundos ou 4 segundos.

**2.4.2 Teste de Carga**

* Simular o uso simultâneo de 10, 50, 100, 200, 500, 1000, 2000, 5000, e 10000 usuários simultâneos.
* De acordo com os testes/simulações de cargas, verificar performance do sistema (se funciona corretamente, não apresenta grande lentidão ou travamentos, e se não derruba o servidor).

**2.4.3 Teste de Stress**

* Forçar o sistema além do esperado de usuários simultâneos para testar até quantos usuários ele suporta.
* Verificar se os dados não são corrompidos.
* Verificar recuperação após falha.
* Verificar tempo de travamento.
* Verificar acesso ao banco de dados.
* Verificar se o sistema volta ao normal, quanto tempo, ou se precisa reiniciar ele.
* Verificar rede e servidor.
* Verificar segurança de dados.

**2.4.4 Teste de Estabilidade**

* Verificar estabilidade do sistema (deixando rodar por horas ou dias).
  + Verificar se o sistema continua com todas as suas informações.
  + Verificar se o sistema não trava.
  + Verificar conexão com banco de dados.
  + Verificar segurança de dados.
  + Verificar consumo de memória.
  + Verificar tempo de resposta.
  + Verificar se rede ou servidor não cai.
  + Verificar tempo de resposta com a rede.
  + Verificar conexão com intranet/internet.

**2.5 Teste de Volume**

* Verificar quantos dados o sistema aceita (eventos, hotéis, pontos turísticos etc).
  + Verificar degradação de performance com a quantidade de dados no sistema

**2.6 Teste de Segurança**

* Verificar limitações de acesso aos dados.
* Verificar se usuários comuns não tem acesso a páginas, informações e ações restritas.
* Verificar se cada usuário tem acesso á apenas suas informações e dados.
* Verificar se os dados de cartões, e dados pessoais estão criptografados.
* Verificar autenticidade de hotéis, eventos, avaliações etc.
* Verificar antivírus.
* Verificar segurança do banco de dados.
* Verificar segurança dos sub sistemas.
* Verificar segurança ao ato de tentativa de recuperação de conta.
* Verificar segurança das formas de pagamentos.
* O sistema deve verificar corretamente a identidade do usuário e manter a sessão protegida.
* Verificar se login e senha são obrigatórios para acessar o sistema.
* Verificar se as senhas estão criptografadas no banco de dados.
* Verificar se o sistema encerra a sessão automaticamente após X minutos de inatividade.
* Verificar se as senhas não aparecem visíveis em formulários ou na URL.
* Tentar acessar URLs diretas de páginas administrativas sem login.
* Tentar alterar parâmetros na URL.
* Tentar enviar dados falsos nos formulários.
* Testar **cookies e tokens** de sessão para ver se podem ser reutilizados em outra conta.
* Verificar direitos do usuário.
* Tentar inserir scripts maliciosos em campos como nome, descrição, etc.
* Testar com comandos SQL maliciosos.
* Verificar se o sistema valida todos os campos de entrada.
* Verificar se apenas arquivos permitidos podem ser enviados (ex: proibir .exe, .php, .bat).
* Verificar se dados pessoais dos hóspedes estão acessíveis somente via login.
* Verificar se os extratos e reservas não podem ser acessados por outros usuários.
* Verificar se **dados privados não aparecem no código-fonte da página.**
* Verificar se atualizações no sistema só podem ser feitas pela rede interna.
* Verificar se apenas administradores podem atualizar ou excluir dados críticos.

**2.7 Testes de Instalação**

* Verificar se a instalação do sistema ocorre normalmente em todas as plataformas.
* Verificar se o sistema funciona normalmente quando se é instalado.
* Verificar atualizações.
* Verificar dados após atualização
* Verificar espaço de armazenamento.
* Verificar arquivos.
* Verificar acesso via atalhos.
* Verificar conectividade com o banco de dados.
* Verificar conectividade com a rede.
* Verificar conectividade com o servidor.
* Verificar se instala de forma independente (sem dependência de arquivos externos, programas etc).
* Verificar versão do sistema.
* Verificar se o sistema checa o espaço disponível em disco antes da instalação.
* Verificar se o sistema avisa o usuário se o espaço for insuficiente.
* Verificar se o espaço ocupado não ultrapassa o limite previsto.
* Verificar se o desinstalador remove todos os arquivos do sistema.
* Verificar se as configurações de sistema, entradas de registro (Windows) e atalhos também são removidos.
* Verificar se nenhum dado do usuário é apagado sem aviso (ex: reservas, histórico, relatórios).

**2.8 Testes do ciclo de negócios**

* Testar uso completo real do começo ao fim do sistema, testar como ele funcionaria em uso real
  + Cadastro
  + Login
  + Pesquisa de hotéis, eventos, pontos turísticos etc.
  + Visualização das informações.
  + Reservas de eventos e hospedagem.
  + Aplicação do cupom de desconto.
  + Avaliações.
  + Atualizar dados pessoais.
  + Recuperar senha.
  + Visualização do histórico de compras.
  + Geramento de PDF e comprovantes.
  + Carrinho.
  + Tudo que estiver no sistema.

1. **Identificação dos Testes Mais Críticos do Sistema**

Com base na documentação do plano de testes, os testes considerados mais críticos para o sistema são:

**3.1 Teste do Banco de Dados**

* Verifica o cadastro, atualização, consulta e remoção de dados dos usuários, hóspedes e informações turísticas.
* Avalia a integridade dos dados e a correta execução dos métodos de acesso ao banco.
* Garante que os dados apresentados aos usuários estejam consistentes e atualizados.

**3.2 Teste de Performance**

* Mede o tempo de resposta do sistema em diferentes situações.
* Avalia o desempenho ao lidar com dados multimídia (como imagens e vídeos).
* Verifica a estabilidade e velocidade do sistema com múltiplos usuários simultâneos.
* Observa a comunicação entre servidor e terminais usados no ambiente real.

1. **Revisão ou dúvidas:**

“Documentação clara e objetiva em relação aos testes, expondo de forma correta como cada teste vai ser executado, em qual ambiente, com qual ordem ou procedência será feito, esses detalhes e outros que fizeram a diferença e trouxeram essa tal clareza e objetividade. Um caso que me ocorreu algumas vezes durante a leitura deste documento, foi um breve pensamento de algum erro ou teste que poderia ser complementado, porém, que veio a ser respondido ou solucionado logo em seguida. Sem sombra de dúvidas, clareza e objetividade definem o documento realizado.

Vale ressaltar ao cliente o seguinte: Os testes foram realizados em máquinas com a configuração interna (hardware) semelhante/equivalente ao que eles usariam e que essas máquinas seriam usadas somente para nossa aplicação, o que nos leva ao seguinte: Se uma máquina de configuração inferior for usada ou uma máquina que exerce não somente nossa aplicação, erros podem ocorrer, na qual afetarão nosso sistema. Logo, isso não está em nossa responsabilidade a partir do consentimento do cliente. Grato.”

- ATT: Kaio Gomes do Nascimento Mazza – Analista de Testes Junior, pela corporação e empresa de Software.