

Gral Imóveis

Plano de Testes

Desenvolvimento de Sistemas

Professora:

Elisabete Tie Hato

Equipe:

Aires Ribeiro Da Silva

Rogério Tadeu Dos Reis

Gabriel Cunha Pereira Dos Santos

Lucas França Fagundes

Histórico de Revisões

Data	Versão	Descrição	Autor
03/03/2021	1.0	Release Inicial	Aires, Gabriel, Rogério e Lucas
06/03/2021	1.1	Inserção dos itens 2.4, 3.2 e 1.1	Aires
06/03/2021	1.2	inserção item 2.3 e 4.2	Gabriel
06/03/2021	1.3	Inserção dos item 3.1	Rogério Reis
06/03/2021	1.4	inserção item 2.1	Lucas
06/03/2021	1.5	Adicionados termos e acrônimos	Rogério Reis

Índice

1. INTRODUÇÃO	4
1.1 TERMOS E ACRÔNIMOS	4
1.2 OBJETIVOS	4
1.3 SISTEMA EXEMPLO	5
1.4 ESCOPO	5
1.5 IDENTIFICAÇÃO DE PROJETO	5
2. REQUISITOS A TESTAR	6
2.1 TESTE DA INTERFACE DO USUÁRIO	6
2.2 TESTE FUNCIONAL	6
2.3 TESTE DE DESEMPENHO	6
2.4 TESTE DO BANCO DE DADOS	6
3. ESTRATÉGIA E FERRAMENTAS DE TESTE	7
3.1 ESTRATÉGIAS	7
3.1.1 <i>Teste Funcional</i>	7
3.1.2 <i>Teste de Interface</i>	8
3.2 FERRAMENTAS	8
4. RECURSOS	9
4.1 EQUIPE DE TESTE	9
4.2 SISTEMA	10
5. CRONOGRAMA	11

1. Introdução

Este documento apresenta o planejamento das atividades de testes do sistema **Gral imóveis** o qual será utilizado como base para as atividades de acompanhamento, revisão, verificação e validação do projeto, desde seu início até sua conclusão, a fim de garantir a análise comparativa do resultado real versus planejado. Desta forma, ações corretivas e preventivas poderão ser tomadas, sempre que resultados reais desviarem significativamente do planejado.

1.1 Termos e acrônimos

Esta seção explica o conceito de um subconjunto de termos importantes que serão mencionados no decorrer deste documento, sendo que os termos devem ser apresentados em ordem alfabética. Todo e qualquer termo, convenção adotada ou abreviações deveriam ser apresentadas nessa tabela a fim de comunicar às partes envolvidas e interessadas (i.e. os stakeholders) o seu significado. Isto visa prover as denominações corretas empregadas no documento.

Termo	Descrição
Clicável	Designativo do elemento de uma interface que, quando sobre ele se clica, desencadeia a execução de um comando ou de uma opção.
Browser	Um navegador de rede, navegador web, navegador da internet ou simplesmente navegador.
API	Application Programming Interface (Interface de programação de aplicação)
back-end	Parte de retaguarda
front-end	Interface frontal
SGBD	Sistema de Gerenciamento de Banco de Dados
Open Source	Código aberto
BD	Base de dados ou Banco de dados

--	--

1.2 Objetivos

Esta seção contém o conjunto de objetivos que orientam as atividades de testes do sistema a ser desenvolvido.

Esse documento do Plano de Testes do Sistema **Gral imóveis** possui os seguintes objetivos:

- Levantar as informações de projeto pertinentes e os componentes de software a serem testados.
- Definir o conjunto de requisitos a serem testados (alto nível).
- Definir e detalhar as estratégias de teste a serem utilizadas.
- Definir os recursos necessários e obter uma estimativa dos esforços das atividades de teste.
- Identificar os artefatos resultantes das atividades de testes.

1.3 Sistema Gral imóveis

Este documento especifica o plano de teste de um sistema que deve prover cadastro e pesquisa de anúncios de imóveis, conteúdo online, denominado de Sistema **Gral Imóveis**, a ser desenvolvido para a Empresa xpto. Seu propósito é prover anúncio e pesquisa de imóveis, permitindo acesso integral dos conteúdos cadastrados no sistema.

1.4 Escopo

Este projeto aborda um Sistema de informação online, com foco em prover conteúdo online. Este Sistema de imóveis necessitará fazer testes unitários, de integração e de sistema. Os testes unitários e de integração visam tratar a qualidade funcional, a interface gráfica e o controle de acesso. Por outro lado, os testes de sistema tratarão as questões de desempenho. Para a execução dos testes serão utilizadas máquinas o mais idênticas possível, em termos de hardware, àquelas que serão implantadas no cliente (provedor de conteúdo online, como um portal de notícias), a fim de garantir a previsibilidade de desempenho e compatibilidade. Outros testes como os testes de stress, de volume e de falha/recuperação não serão realizados por se considerar que o ambiente de implantação do sistema não está sujeito a esse tipo de ocorrência, as quais podem ser facilmente previstas e tratadas pelo cliente.

1.1 Identificação de Projeto

Esta seção contém informações sobre o conjunto de documentos disponíveis e respectiva situação de revisão, os quais são importantes no plano de teste.

Documento	Disponível	Revisado
Especificação de Requisitos	Sim	Sim
Plano de Projeto	Sim	Sim
Modelo de Análise	Sim	Sim
Modelo de Projeto	Sim	Sim
Documento de Arquitetura	Sim	Sim
Protótipo	Não	Não
Manual do Usuário	Não	Não
Lista de Riscos	Sim	Sim

1. Requisitos a Testar

Esta seção apresenta um conjunto de requisitos funcionais e não funcionais que foram identificados durante o levantamento de requisitos e para os quais os testes abaixo são considerados como necessários (representam um extrato do que será testado).

OBS.: Note que um sistema possui diversos requisitos que podem ser verificados através de outros tipos de testes como teste de carga, teste de instalação, teste de volume, teste de stress, teste de configuração. Por exemplo, num teste de segurança e de controle de acesso, você pode verificar se usuários não cadastrados podem acessar informações restritas aos cadastrados. Adicionalmente, você pode checar se, além do administrador, ninguém mais pode inserir, atualizar ou remover dados do sistema

1.2 Teste da Interface do Usuário

- Ao navegar pelas telas de interface gráfica o usuário deverá identificar o que deve ser preenchido e compreender o uso de forma rápida.
- Identificar se os recursos clicáveis estão facilmente disponíveis como os agregados a fotos e links.

- Averiguar se os campos de autenticação de usuário estão disponíveis de forma clara, no qual os campos de preenchimento obrigatório sejam facilmente utilizados.
- Identificar se existe variação na diagramação e navegação de telas em diferentes browsers de navegação.
- Verificar se existe comunicação de telas por meio de itens clicáveis de modo que o usuário não percorra longo caminho para acessar componentes.
- Identificar se as opções do menu atendem as principais requisições dos usuário, ou as utilizadas com maior frequência.
- Em caso de seleção de opção que redireciona para outra página identificar se existe uma etapa de recuperação de dados ao sair. Averiguar a possibilidade de salvar dados preenchidos como o de cadastro de imóveis, por exemplo.
- Verificar se a resolução das imagens de divulgação é satisfatória em diferentes configurações de dispositivos que acessam o sistema.
- Verificar se o usuário é notificado por meio de mensagens informativas em caso de erros de preenchimento.
- Identificar se estilos de ícones clicáveis atendem às preferências dos usuários.
- Verificar se ícones são autoexplicativos para o usuário.
- Averiguar a facilidade de navegação vertical das páginas com a existência de botão com função de volta ao topo.
-

1.3 Teste Funcional

- Verifique se qualquer usuário pode acessar o sistema podendo realizar uma pesquisa de imóveis.
- Verifique que as inserções de anúncios (cadastro dos imóveis), foram realizadas com sucesso.
- Verifique que as consultas tragam todas informações de tais cadastros hora salva na base de dados
- Verifique que a operação de alterar um registro, de fato alterou na base de dados.

- Verifique que a operação de excluir um registro, realmente foi excluído na base de dados.
- Verifique que ao preencher o formulário para contato com o vendedor e salvá-lo, seja realizado com sucesso na base de dados.
- Verifique que ao consultar o relatório de formulários, seja apresentado na tela de forma consistente em relação aos dados inseridos.

1.4 Teste de Desempenho

- Front-end – site comprador e área administrativa vendedor
 - Verificar o tempo de carregamento das páginas e funcionalidades do site comprador.
Exemplo: landing page, filtros de pesquisa.
 - Verificar o tempo de resposta das requisições dos formulários das páginas do comprador.
 - Verificar tempo de resposta na manipulação dos dados pelo comprador na área administrativa. Exemplo: criação de anúncio, alteração de status de mensagem de contato.
- API back-end
 - Verificar o tempo de inicialização do servlet container da API no servidor de aplicação.
 - Verificar o desempenho das respostas da API para as requisições feitas pelo front-end para todos endpoints. Swagger fornece o tempo.
 - Verificar o desempenho das instruções SQL feitas pela implementação da JPA. Exemplo: log no console das instruções SQLs.
- Banco de Dados
 - Verifique o desempenho das instruções efetuadas pela API, no banco de dados.
Exemplo ferramenta IWatch
 - Verifique se foram criados os índices adequados nas tabelas que o projeto utiliza.

1.5 Teste do Banco de Dados

- Verifique se as informações de conteúdos podem ser inseridos, atualizados e consultados pelo administrador de sistema.

- Verifique que as informações úteis obtidas pelo subsistema responsável podem ser atualizadas e que as mesmas podem ser apresentadas.
- Verifique que as informações inseridas, sejam consistentes e coerentes de acordo com os dados de tal cadastro.

2. Estratégia e Ferramentas de Teste

Esta seção apresenta um conjunto de tipos de testes a serem realizados onde, para cada tipo de teste adotado, definem-se as técnicas utilizadas, o critério de finalização do teste e outras suposições ou considerações específicas para o teste.

1.6 Estratégias

As estratégias compreendem um conjunto de testes empregados com o objetivo de verificar aspectos específicos do sistema em desenvolvimento (abaixo, apresenta-se um extrato de possíveis testes que podem ser adotados).

1.6.1 Teste Funcional

Objetivo do Teste:	Assegurar a funcionalidade adequada do teste, incluindo navegação, entrada de dados, processamento e recuperação.
Técnica:	Executar cada caso de uso e fluxo de caso de uso, usando dados válidos e inválidos a fim de verificar:

	<p>§ Se os resultados esperados ocorrem quando dados válidos são usados</p> <p>§ Se mensagens de erro ou aviso apropriadas são exibidas quando dados inválidos são usados.</p> <p>§ Se cada regra de negócio está sendo aplicada de maneira apropriada</p>
<p>Critério de Finalização:</p>	<p>§ Todos os testes planejados foram executados.</p> <p>§ Todos os defeitos identificados foram tratados.</p>
<p>Considerações Especiais:</p>	Nenhuma

1.1.1 Teste de Interface

Objetivo do Teste:	Verificar se a navegação através das funcionalidades testadas refletem as funções e os requisitos do negócio apropriadamente, incluindo janela-a-janela, campo-a-campo, e o uso de métodos de acesso (movimentos do mouse, teclas aceleradoras) Checar se os objetos e características da janela, tais como menus, tamanho, posição, estado e foco conformam-se aos padrões.
Técnica:	Criar ou modificar os testes para cada janela a fim de verificar a navegação e os estados de objeto adequados para cada janela e objetos da aplicação.
<p>Critério de Finalização:</p>	É verificado que cada janela permanece consistente com a versão de comparação ou dentro de padrões aceitáveis de usabilidade.
<p>Considerações Especiais:</p>	Todas as propriedades para objetos devem ser acessadas.

1.7 Ferramentas

As seguintes ferramentas serão empregadas para esse projeto:

Atividade	Ferramenta	Fornecedor
Gerenciamento de Teste	Microsoft Word	Microsoft
Projeto de Teste	Astah UML	Astah
Gerenciamento de Projeto	Microsoft Project	Microsoft
	Microsoft Word	Microsoft
	Trello	Atlassian
Ferramentas do SGBD	Postgres	BDs Open Source
	Dbeaver	Open Source (ASL)

2. Recursos

Esta seção apresenta a equipe de testes, identificando-se os papéis e responsabilidades. Além disso, a infra-estrutura existente é listada.

2.1 Equipe de Teste

A Tabela abaixo descreve um conjunto de papéis e respectivas responsabilidades na equipe de teste.

Papel	Responsável	Responsabilidades
Gerente de Teste, Gerente do Projeto de Teste	Aires Ribeiro	Fornece supervisionamento gerencial. Responsabilidades: <ul style="list-style-type: none">• provê direcionamento técnico• adquire recursos apropriados• fornece relatórios de gerenciamento
Test Designer	Gabriel Cunha	Identifica, prioriza, e implementa os casos de teste. Responsabilidades:

		<ul style="list-style-type: none"> • gera o plano de teste • cria o modelo de teste • avalia a efetividade do esforço de teste
Testador	<p>Lucas França E Rogério Reis</p>	<p>Executa os testes.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • executar os testes • registrar os resultados • reestabelecer-se dos erros • documentar solicitações de mudança
Administrador do Sistema de Teste	Integrante 1	<p>Garante que o ambiente e os bens de teste sejam gerenciados e mantidos.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • administrar o sistema de gerenciamento teste • instalar e gerenciar o acesso do trabalhador ao sistema de testes
Gerente do Banco de Dados, Administrador do Banco de Dados	Integrante 4	<p>Garante que o ambiente e bens de teste de dados (banco de dados) sejam gerenciados e mantidos.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • administrar os dados de teste (base de dados)
Designer	Integrante 3	<p>Identifica e define as operações, atributos, e associações das classes de teste.</p> <p>Responsabilidades:</p> <ul style="list-style-type: none"> • identificar e definir as classes de teste • identificar e definir os pacotes de teste
Implementador	Integrante 2	<p>Implementa e faz os testes unitários das classes e pacotes de teste.</p> <p>Responsabilidades:</p>

		<ul style="list-style-type: none"> cria as classes e pacotes de teste implementados no modelo de teste
--	--	---

2.2 Sistema

A Tabela abaixo expõe os recursos disponíveis para o projeto de teste.

Servidores:

- Servidor de Aplicação
 - Site Front-end
 - API de integração
- Servidor de Banco de Dados
 - PostgreSQL DataBase Server

3. Cronograma

O cronograma abaixo contempla as atividades de testes, marcos, e respectivas datas de início e término.

Milestone	Data de Início	Data de Término
Planejar Teste		
Projetar Teste		
Implementar Teste		
Executar Teste		
Avaliar Teste		