**Plano de Testes**

(modelo)

**Nome**

**do**

**Produto/Arte & Agenda**

**Elaborado**

**por**

**Data**

**25/06/2024**

**ÍNDICE**

**Gustavo Henrique, Mateus Azeredo**

# 1.0 INTRODUÇÃO

# 2.0 OBJETIVOS E TAREFAS

**2.1 Objetivos**

**2.2 Tarefas**

**3.0 ESCOPO**

**Geral**

**Tática**

**4.0 Estratégia de teste 4.1 Teste teste unitário 4.2 Teste de sistema e integração 4.3 Teste de desempenho e estresse 4.4 Testes de aceitação do usuário 4.5 Teste em lote/regressão Automatizados**

**5.0 Requisitos de hardware**

**6.0 Requisitos de Software 6.1 Ferramentas de Testes**

**7.0 Calendários de testes**

**8.0 Procedimentos de Controle**

**9.0 Recursos/Funções a serem testados**

**10.0 Recursos que não devem ser testados**

**11.0 Recursos/Papéis e Responsabilidades**

**12.0 Documentos/Evidências dos Testes**

**13.0 Dependências/Restrições**

**14.0 Riscos/Suposições**

**15.0 Aprovações**

# 1.0 INTRODUÇÃO

**Web site para organizar agendamentos de um salão de beleza que contem três entidades, administrador, funcionário e usuário. Administrador e responsável por cadastrar horários de agendamentos, serviços e funcionários. Usuários agendar serviços e conferir agendamento. Entidade Funcionário responsável por apenas conferir agenda.**

**.**

**2.0OBJETIVOS E TAREFAS**

# 2.1 Objetivos

Testar funcionalidades da entidade administrador, garantir que funcionalidade de agendamento funcione e que usuários e funcionários visualizem seus agendamentos

# Tarefas

* Testar funções de agendamento.
* Visualização de agendamento pelo usuário.
* Garantir funcionamento do sistema de agenda.
* Teste de Autenticação de usuário.
* Testar Manter administrador.

**3.0ESCOPO**

# Geral

Todas as funções do sistema de entidades agendamento, administrador, usuário e funcionário deverão ser testadas

# Táticas

Testar as funcionalidades CRUD de Manter Administrador, Manter Usuário, Manter agenda **e Manter Produto** identificar e encontrar erros e correções a serem feitas.

# 4.0ESTRATÉGIA DE TESTE

# Serão realizados os seguínte tipos de Testes, tanto no Back-End quanto Front-End.

* Unitários
* Teste de Integração
* Testes Funcionais
* Testes de Regressão
* Testes de Aceitação do Usuário.

# Em todas as funcionalidades testar as funções abaixo e as integrações internas e externas, se houverem:

# AdministradorServico:

* Verificar se o usuário é administrador.
* Obter dados do usuário.
* Listar todos os usuários.
* Listar usuários por função.

AgendamentoServico:

* Cadastrar um novo agendamento.
* Listar agendamentos por usuário.
* Listar agendamentos por data.
* Excluir agendamento.

AutorizacaoServico:

* Carregar usuário por nome de usuário.

# HorarioServico:

* Cadastrar um novo horário.
* Listar todos os horários.
* Listar horários ativos.
* Excluir horário.
* Editar horário.
* Atualizar status de ativo.
* Buscar horário por ID.

# ProdutoServico:

* Listar todos os produtos.
* Editar produto.
* Cadastrar ou alterar produto.
* Remover produto.
* Remover produto de usuário.

# 4.1 Teste Unitário

**Participantes:**

|  |  |
| --- | --- |
| **Testes Unitários - Responsáveis** |  |
| **Nome** | **Equipe** |
| Gustavo Henrique, Mateus | Mateus, Gustavo Henrique |

**Metodologia:**

**Especificação dos Testes:**

* Com base nos métodos implementados nos serviços.

**Desenvolvimento dos Scripts de Teste:**

* Junit será usado para escrever os testes unitários.
* Cada teste será estruturado para verificar uma condição específica de cada método.

**Execução dos Testes:**

Preparar o ambiente de teste com as dependências necessárias.

Executar os testes individualmente para cada método dos serviços.

**Registro e Correção de Defeitos:**

* Após as correções, os testes serão re-executados para garantir que os defeitos foram corrigidos e que não surgiram novos problemas.

**Revisão e Avaliação:**

A equipe de testes revisará os resultados para garantir que todas as funcionalidades foram validadas.

# 4.2 Teste de sistema e integração

* Assegurar que os serviços AdministradorServico, AgendamentoServico, AutorizacaoServico, HorarioServico e ProdutoServico funcionem corretamente quando integrados.
* Confirmar que os casos de uso do sistema, como a criação de agendamentos, a listagem de produtos, a autorização de usuários, etc., sejam executados corretamente em um fluxo integrado.
* Identificar e corrigir defeitos que possam surgir devido à interação entre diferentes serviços.

**Participantes:**

|  |  |
| --- | --- |
| **Testes de Sistema e Integração - Responsáveis** | |
| **Nome** | **Equipe** |
| Gustavo Henrique , Mateus Lima | Mateus Lima, Gustavo Henrique |

**Metodologia:**

**Especificação dos Testes:**

* Com base nos requisitos funcionais e nos casos de uso
* Dados necessários para simular diferentes cenários de teste.

**Desenvolvimento dos Scripts de Teste:**

* Desenvolvedores escreverão os scripts de teste unitário para validar funcionalidades individuais dos serviços.
* Desenvolveremos scripts de integração utilizando ferramentas como Postman, JUnit, ou frameworks de teste de integração.

**Execução dos Testes:**

* Validação de cada serviço individualmente
* Execução dos testes que validam a interação entre serviços, como a comunicação entre AdministradorServico e UserRepositorio.
* Postman para testes de API, JUnit para testes de unidade e integração, Jenkins para integração contínua e execução automatizada dos testes.

**Registro e Correção de Defeitos:**

* Após as correções, os testes serão re-executados para garantir que os defeitos foram corrigidos e que não surgiram novos problemas.

**Revisão e Avaliação:**

* A equipe de testes revisará os resultados para garantir que todas as funcionalidades foram validadas.

# 4.4 Teste de aceitação do usuário

* Garantir que todas as funcionalidades descritas nos requisitos foram implementadas corretamente e funcionam conforme o esperado.
* Verificar se o sistema é intuitivo e fácil de usar pelo usuário final.
* Assegurar que o sistema atende às necessidades e expectativas dos usuários finais.

**Participantes:**

|  |  |
| --- | --- |
| **Testes de Aceitação do Usuário - Responsáveis** | |
| **Nome** | **Equipe** |
| Gustavo Henrique , Mateus Lima | Mateus Lima, Gustavo Henrique |

**Metodologia:**

* Com base nos requisitos funcionais e não-funcionais do sistema.
* Escolha dos usuários finais que participarão dos testes
* Configuração do ambiente de teste similar ao ambiente de produção.
* Os usuários finais executarão os cenários de teste, utilizando o sistema em situações reais de uso.
* Cada cenário de teste deve produzir resultados que correspondem aos resultados esperados definidos nos requisitos.
* Após as correções, os testes serão re-executados para garantir que os defeitos foram corrigidos e que não surgiram novos problemas.

**4.5 Teste em Lote - Automatizados**

# Teste de regressão

O teste de regressão é o reteste seletivo de um sistema ou componente para verificar se as modificações não causaram efeitos indesejados e se o sistema ou componente ainda funciona conforme especificado nos requisitos.

**Participantes:**

|  |  |
| --- | --- |
| **Testes de Regressão - Responsáveis** | |
| **Nome** | **Equipe** |
|  |  |

**Metodologia:**

Indique como serão feitos os testes de Regressão automatizados, como serão geradas e armazenadas as evidências/resultados dos testes.

# 5.0REQUISITOS DE HARDWARE

Computadores para os testadores, servidores de testes para o sistema e banco de dados, conexão com a Internet e rede de computadores interligadas.

**6.0REQUISITOS DE SOFTWARE**

# 6.1 Ferramentas de Testes

* Testes Unitários: Junit, versão 5.6.0
* Testes de Regressão/Unitários: Junit, versão 5.6.0
* Testes de Integração/Regressão: Selenium, versão 4.22.0, Postman, versão 9.16.0
* Testes de Aceitação do Usuário, utilizar o próprio sistema.

# 7.0CALENDÁRIO DE TESTES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Teste** | **Unitários** | **Ferramenta s** | **Junit** | **Data Início** | **20/06/2024** |
| **Marco** |  |  |  | **Data Fim** | **23/06/24** |
| **Teste** | **Integração/Sistema** | **Ferramenta s** | **Selenium** | **Data Início** | **24/06/2024** |
| **Marco** |  |  |  | **Data Fim** | **24/06/2024** |
| **Teste** | **Regressão** | **Ferramenta s** | **Selenium** | **Data Início** | **24/06/2024** |
| **Marco** |  |  |  | **Data Fim** | **24/06/2024** |
| **Teste** | **Aceitação/Usuário** | **Ferramenta s** | **Próprio Sistema** | **Data Início** | **25/06/2024** |
| **Marco** |  |  |  | **Data Fim** | **25/06/2024** |

**8.0PROCEDIMENTOS DE CONTROLE**

# Relatório de problemas

# Erros Relacionados aos teste Selenium e Junit:

Teste de regressão da interface não foi possível realizar por causa do software selinum. O problema ocorrido acontece na classe produto(no ícone serviços), na qual para realizar o cadastramento completo e necessário sair do modal para escolher uma foto que e obrigatório, por isso e necessário sair da interface do site Arte & cabelo na qual o Selenium não consegui executar fora do site .

# Erro relacionados aos teste Junit e Postman:

Problemas na classe produtoControler em APIS remover por Ids, remover por usuário e editar por Ids. Nessas APIS ditadas não foram possíveis realizar os testes Junit e Postman.

# Pedidos de mudança

**Documente o processo de modificações no software**. Identifique quem assinará as alterações e quais seriam os critérios para incluí-las no produto atual. **Se as alterações afetarem os programas existentes, estes módulos terão de ser identificados.**

# 9.0RECURSOS/FUNÇÕES A SEREM TESTADOS

|  |  |
| --- | --- |
| **Recurso e/ou Funções** | **Testes** |
| Administrador | Verificar, Obter, Listar |
| ManterAgendamento | CRUD- Cadastro, ListaUsuario, ListaData, Exclusão, |
| autenticação | Verificar |
| ManterHorario | Cadastrar , Excluir, Listar, Atualizar, BuscarID |
| ManterProduto | Cadastra, Listar, Atualizar , Remover |

# 11.0RECURSOS/PAPÉIS E RESPONSABILIDADES

|  |  |  |
| --- | --- | --- |
| **Recursos** | **Papel/Função** | **Responsável** |
| Casos de testes de Aceitação | Usuária/Gestor | Gustavo Henrique, Mateus Azeredo |
| Realizar/executar os testes | Desenvolvedor/Tes ter | Gustavo Henrique, Mateus Azeredo |
| Projetar/definir rotinas de testes | Analista/Desenvolv edor | Gustavo Henrique, Mateus Azeredo |
| Realizar/executar testes de | Usuário/Gestor | Gustavo Henrique, Mateus Azeredo |
| aceitação |  |  |
| Ambiente de  testes/Servidores, etc | Analistas/Suporte | Gustavo Henrique, Mateus Azeredo |
| Problemas/Contingências | Gerente TI | Gustavo Henrique, Mateus Azeredo |

**12.0DOCUMENTOS/EVIDÊNCIAS DOS TESTES**

# Principais resultados

Identifique os documentos de entrega. Você pode listar os seguintes documentos:

* Plano de teste -> Este documento.
* Casos de teste
* Relatórios de incidentes de teste
* Relatórios de resumo de teste e/ou evidências dos testes (telas, logs,etc).

# 13.0DEPENDÊNCIAS/RESTRIÇÕES

Caso aconteçam restrições significativas para a execução dos testes como indisponibilidade de itens de teste e outros recursos computacionais ou Software de teste, estas restrições/problemas devem ser reportados neste item do Plano de Testes.

# 14.0RISCOS/PRESSUPOSTOS

Caso ocorram atrasos nas entregas dos itens (programas) a serem testados e/ou falta de ferramenta de testes, devem ser relatados aqui, indicando possíveis graus de risco para a entrega dos testes, conforme planejamento, bem como as contingências para cada caso

# 15.0APROVAÇÕES

Especifique os nomes e cargos de todas as **pessoas que devem aprovar este plano**. Forneça espaço para as assinaturas e datas.

|  |  |  |
| --- | --- | --- |
| **APROVAÇÕES** |  |  |
| **Nome/Responsável** | **Assinatura** | **Data** |
| Gustavo Henrique Nunes Bueno |  | 25/06/24 |
| Mateus Azeredo de Lima |  | 25/06/24 |