

Grupo BANDEJÃO: Beatriz Dellatorre, Bruno Cotel, Denerson Berudio, Luana Vidal e Mairon Azevedo.

Disciplina: Qualidade e Teste  
Curso: Sistemas de informação

Professora: Vânia de Oliveira Neves  
Semestre: 23.1



# Trabalho Prático

## Plano de Testes

### Sistema API Eventos



#### Registro de Mudanças

Versão	Data de Mudança	Por	Descrição
1.0	19/06/2023	Todos os integrantes	Versão inicial do documento, especificando quais testes seriam utilizados, quais ferramentas, e requisitos funcionais e não funcionais.
2.0	09/07/2023	Todos os integrantes	Versão final do documento, implementando o que cada integrante teve de responsabilidade, e implementação dos tópicos que estavam em branco.

# Índice

<b>1.1 Estruturação do trabalho.....</b>	<b>2</b>
1.1.1 No escopo.....	2
1.1.2 Fora do escopo.....	3
<b>1.2 Objetivos de Qualidade.....</b>	<b>3</b>
<b>1.3 Papéis e Responsabilidades.....</b>	<b>3</b>
<b>2.1 Visão Geral.....</b>	<b>4</b>
<b>2.2 Fases de Teste.....</b>	<b>5</b>
<b>2.3 Completude do Teste.....</b>	<b>5</b>
<b>2.4 Critérios de Suspensão e Requisitos de Retomada.....</b>	<b>5</b>
<b>2.5 Atividades do projeto, estimativas e cronograma.....</b>	<b>5</b>
<b>4.1 Ferramentas de Teste.....</b>	<b>6</b>
<b>4.2 Ambiente de Teste .....</b>	<b>6</b>

# 1 Introdução

O Plano de Teste é projetado para descrever o escopo, abordagem, recursos e cronograma de todas as atividades de teste do projeto Sistema API Eventos.

O plano identifica os itens a serem testados, os recursos a serem testados, os tipos de teste a serem realizados, o pessoal responsável pelo teste, os recursos e o cronograma necessários para concluir o teste e os riscos associados ao plano.

## 1.1 Estruturação do trabalho

---

Nesta parte, iremos apresentar como fizemos o processo de desenvolvimento do trabalho, e como o estruturamos, para assim conseguir informar tudo que é necessário para entender o processo de produção do produto final. Mostrando ferramentas de comunicação, metodologias para melhor organização de tarefas, e os requisitos funcionais e não funcionais do sistema.

### 1.1.1 No escopo

Todos do Sistema API Evento que foram definidos nas especificações de requisitos de software precisam ser testados.

Nome do módulo	Papéis aplicáveis	Descrição
Consultar Status da edição.	Administrador, cliente	Ambos: Poder consultar o status de uma edição de evento cadastrada no sistema
Consultar avaliações feitas em um evento.	Administrador, cliente	Ambos: Poder consultar as avaliações feitas em um evento
Consultar ingressos comprados por um cliente.	Administrador, cliente	Ambos: Poder consultar os ingressos que um usuário possui.
Calcular média de avaliações	Administrador, cliente	Ambos: Poder calcular a média de avaliações de um evento específico
Cancelar a compra de um ingresso	Administrador, cliente	Ambos: Poder cancelar um ingresso comprado.
Verificar cargo de um usuário	Administrador	Administrador: Poder verificar um usuário para saber se possui a chave de Administrador ou não.

### 1.1.2 Fora do escopo

- Interfaces de hardware;
- Interfaces de software;
- Lógica de banco de dados;
- Interfaces de comunicação;
- Segurança e desempenho do site;
- Front End de um cliente.

## 1.2 Objetivos de Qualidade

---

Os objetivos do teste são verificar a funcionalidade da API de Eventos, o projeto deve testar o fluxo de cadastro de eventos, edições, ingressos e chaves dos usuários, para garantir que todas essas operações possam funcionar normalmente.

## 1.3 Papéis e Responsabilidades

---

Cargo	Membro	Tarefas
Aluno	Beatriz Dellatorre	Criação de testes unitários das classes de serviço.  Análise do sistema das especificações ISO.  Sugestão de novas idéias de funcionalidades para ter novas implementações de teste.  Implementação de novas funcionalidades de serviço do sistema.  Apoio na criação da documentação do Readme.md do github.
Aluno	Bruno Cotelro	Criação de testes unitários das classes de serviço.  Análise do sistema das especificações ISO.  Sugestão de novas idéias de funcionalidades para ter novas implementações de teste.  Implementação de novas funcionalidades de serviço do sistema.  Apoio na criação da documentação do Readme.md do github.
Aluno	Denerson Eduardo	Criação de testes unitários das classes de serviço.  Análise do sistema das especificações ISO.

		<p>Sugestão de novas idéias de funcionalidades para ter novas implementações de teste.</p> <p>Implementação de novas funcionalidades de serviço do sistema.</p> <p>Apoio na criação da documentação do Readme.md do github.</p>
Aluno	Luana Vidal Leão	<p>Criação de testes unitários das classes de serviço.</p> <p>Análise do sistema das especificações ISO.</p> <p>Sugestão de novas idéias de funcionalidades para ter novas implementações de teste.</p> <p>Implementação de novas funcionalidades de serviço do sistema.</p> <p>Apoio na criação da documentação do Readme.md do github.</p>
Aluno	Mairon Azevedo	<p>Criação de testes unitários das classes de serviço.</p> <p>Sugestão de novas idéias de funcionalidades para ter novas implementações de teste.</p> <p>Sugestão de novas idéias de funcionalidades para ter novas implementações de teste.</p> <p>Implementação de novas funcionalidades de serviço do sistema.</p> <p>Documentação do Readme.md do github.</p> <p>Criação da Pipeline do Projeto.</p>

## 2 Metodologia de Teste

### 2.1 Visão Geral

---

Utilizamos as formas de teste ensinadas em sala de aula para aprimorar nossos conhecimentos e conseguirmos ter uma visualização melhor da qualidade do nosso código.

### 2.2 Fases de Teste

---

- Teste de unidade (métodos das classes de serviço são testados individualmente) utilizando JUnit;
- Teste de integração (Teste de requisições utilizando Postman);
- Teste de sistema: teste do sistema utilizando Selenium;
- Teste de mutação utilizando Pit Runner;
- Teste Funcional.

## 2.3 Completude do Teste

---

- A taxa de 100% de execução dos testes é obrigatória, a menos que um motivo claro seja fornecido.
- Mínimo de 80% de cobertura feita.
- A taxa de 80% de aprovação dos testes é obrigatória.

## 2.4 Critérios de Suspensão e Requisitos de Retomada

---

Se os membros da equipe relatarem que há 40% dos casos de teste com falha, suspenda o teste até que a equipe de desenvolvimento corrija todos os casos com falha.

## 2.5 Atividades do projeto, estimativas e cronograma

---

Tarefa	Membros	Estimativa de esforço
Criar testes utilizando JMeter	Desenvolvedores	8 horas
Executar os testes	Testador, administrador de testes.	2 horas
Criar relatório de testes	Testador	4 horas
Implementação de Pipeline de CI/CD	Desenvolvedores	2 horas
Implementar Testes Unitários	Desenvolvedores	5 horas
Implementar Testes de Requisitos Funcionais	Desenvolvedores	6 horas
Total		27 horas

## 3 Entregáveis de Teste

Os entregáveis de teste que serão fornecidos, são:

1. Plano de teste.
2. Casos de teste.
3. Relatório de erros.
4. Simuladores de ferramentas de teste.
5. Dados de teste.

## 4 Necessidades de Recursos e Ambiente

### 4.1 Ferramentas de Teste

---

- Necessário um servidor de banco de dados com MySQL instalado e um servidor Web com Apache instalado;
- Necessário uma ferramenta para teste de requisições (Postman, por exemplo);
- Necessário uma IDE para conseguir abrir o código (IntelliJ, por exemplo);
- Necessário a ferramenta Selenium para realização dos testes automatizados.

### 4.2 Ambiente de Teste

---

Os seguintes softwares são necessários, além de softwares específicos do cliente.

- Windows 8 ou superior
- Office 2013 ou superior
- IntelliJ Community Edition

## 5 Termos / Acrônimos

TERMO ACRÔNIMO	DEFINIÇÃO
API	<i>Application Program Interface</i>
IDE	<i>Integrated Development Environment</i>
CI/CD	<i>Continuous Integration / Continuous Delivery</i>