** INSTITUTO POLITÉCNICO DOM DAMIÃO FRANLIN – Nº 8028**

**COORDENAÇÃO DO CURSO DE INFORMÁTICA**

**CURSO DE TÉCNICO DE INFORMÁTICA**

**DESENVOLVIMENTO DE UMA APLICAÇÃO WEB PARA VENDA ONLINE DE INGRESSOS PARA EVENTOS EDUCACIONAIS E CULTURAIS.**

**Elaborado por:**

Délcio Jacob

Luisa Paula Valente

Kaike Gaspar Bartolomeu

**Orientado por:** Eng. Helmer de J. Fumuassuca Capassola

Trabalho Final de Curso apresentado ao Instituto Politécnico Dom Damião Franklin (IPDDF) para obtenção do grau de técnico Médio: de Informática

**LUANDA, 2023**

****

**INSTITUTO POLITÉCNICO DOM DAMIÃO FRANLIN – Nº 8028**

**DESENVOLVIMENTO DE UMA APLICAÇÃO WEB PARA VENDA ONLINE DE INGRESSOS PARA EVENTOS EDUCACIONAIS E CULTURAIS.**

**LUANDA, 2023**

# **DEDICATÓRIA**

Agradecemos primeiramente а Deus que iluminou todo о nosso caminho durante esta caminhada e nós trouxe até aqui, aos nossos pais que não mediram esforços para que nós concluíssemos mais esta etapa importante de nossas vidas e pela força paciência, compreensão e pelo apoio incondicional, e aos colegas e amigos de longos anos de batalha.

# **AGRADECIMENTOS**

Agradecemos primeiramente a Deus. Aos pais e irmãos, que sempre nos apoiaram e não nós deixaram desistir. Ao nosso orientador Professor :Helmer de J.Fumuassuca Capassola, pela dedicação e paciência recebidas.

Aos nossos amigos e a todos que de uma forma ou de outra estiveram presentes no decorrer desta caminhada. E a todo corpo docente deste curso e sem esquecer todos os colegas que caminharam ao meu lado. A todos, o meu carinho e agradecimentos.

A todos os professores que estiveram presentes em nossa formação e de certa forma seus ensinamentos, e experiência transmitidos ao longo deste trajecto, foram cruciais para a conclusão do nosso ensino médio.

# **RESUMO**

O mercado de vendas online está vivendo um dos seus melhores momentos,ajudando assim a impulsiona as vendas e a alcançar um público diferente .

Com as vendas online o seu negócio pode ganhar mais credibilidade e visualidade, tanto o organizador como o cliente, depois de se cadastrarem no nosso site, poderão vender como comprar ingressos via online.

Fazer compras online é comprar qualquer artigo sem nenhum contacto entre o vendedor e o cliente que resultará em uma compra rápida. O nosso projeto permitirá listar todos eventos que estarão disponíveis, disponibilizará as informações dos eventos, depois da compra podendo assim receber os seus ingressos em suas casas sem precisar se deslocar. O desenvolvimento da aplicação web se sucedeu utilizando tecnologias, técnicas e ferramentas adequadas. O protótipo foi desenvolvido em um ambiente JavaScript, pois JavaScript é a linguagem de programação do lado do cliente (Client-side) mais utilizada e é usada também para o lado do servidor (Server-side), permitindo inserir vários efeitos, fazendo com que o site fique mais dinâmico, e permite executar instruções como resposta às ações do usuário.

PALAVRAS-CHAVES: aplicação, venda online, bilhete, eventos, compras online, participante e organizador.

# **ABSTRATO**

# **LISTA DE ABREVIAÇÕES E SIGLAS**

HTML - HyperText Markup Language;

CSS - Cascading Style Sheets;

SQL - Structured Query Language;

RF – Requisitos Funcionais;

UML - Unified Modeling Language;

JS- JavaScritp

TS- TypeScript

# **LISTA DE TABELAS**

Tabela 1.1- Requisitos Funcionais

Tabela 1.2- Requisitos nã Funcionais

# **LISTA DE FIGURAS**

# Figura 1.1-Diagrama de Entidade Relacionamento

Figura 1.2-Modelo relaciomento

Figura 1.3-Diagrama de Atividade

Figura 1.4-Diagrama de Classe

Figura 1.5-Página principal

Figura 1.6-Tela de Login

Figura 1.7-Tela de criação de evento

Figura 1.8-Conexão com a base de dados

# 

# **ÍNDICE**

[**DEDICATÓRIA** I](#_Toc129345854)

[**AGRADECIMENTOS** II](#_Toc129345855)

[**RESUMO** III](#_Toc129345856)

[PALAVRAS-CHAVES: III](#_Toc129345857)

[**ABSTRATO** IV](#_Toc129345858)

[**LISTA DE ABREVIAÇÕES E SIGLAS** IV](#_Toc129345859)

[**LISTA DE TABELAS** IV](#_Toc129345860)

[**LISTA DE FIGURAS** IV](#_Toc129345861)

[Figura 1.1-Diagrama de Entidade Relacionamento IV](#_Toc129345862)

[**ÍNDICE** V](#_Toc129345863)

[**INTRODUÇÃO** 1](#_Toc129345864)

[Contextualização do problema. 1](#_Toc129345865)

[Formulação do Problema 1](#_Toc129345866)

[Justificativa 1](#_Toc129345867)

[Objetivo geral 2](#_Toc129345868)

[Objectivos específicos 2](#_Toc129345869)

[**CAPÍTULO 1 – FUNDAMENTAÇÃO TEÓRICA** 3](#_Toc129345870)

[1.1 Venda online de ingressos para eventos 3](#_Toc129345871)

[1.2 Importância da venda de ingressos online 3](#_Toc129345872)

[1.3 Engenharia De Software 4](#_Toc129345873)

[1.4 Análise e Desenvolvimento de Sistemas 4](#_Toc129345874)

[1.4.1 Análise e Especificação de Requisitos 4](#_Toc129345875)

[1.5 Linguagem de Modelagem Unificada 4](#_Toc129345876)

[1.5.1 Diagramas 5](#_Toc129345877)

[1.5.1.1 Diagramas Estruturais 5](#_Toc129345878)

[1.5.1.2 Diagrama Comportamental 5](#_Toc129345879)

[1.6 Paradigmas de Programação 5](#_Toc129345880)

[1.6.1 Programação Modular 6](#_Toc129345881)

[1.7 Linguagem de Programação 6](#_Toc129345882)

[1.7.1 TypeScript 6](#_Toc129345883)

[1.7.2 JavaScript 7](#_Toc129345884)

[1.7.3 Reactjs 7](#_Toc129345885)

[1.7.4 Nodejs 7](#_Toc129345886)

[1.10 Html 7](#_Toc129345887)

[1.11 Css 7](#_Toc129345888)

[1.12 Boostrap 8](#_Toc129345889)

[1.13 Desenvolvimento da Estrutura de Dados 8](#_Toc129345890)

[1.13.1 Base de dados 8](#_Toc129345891)

[1.13.2 Sistema De Gerenciador De Base De Dados 8](#_Toc129345892)

[1.13.4 SGBD Mysql 8](#_Toc129345893)

[1.13.5 Modelo de Dados 9](#_Toc129345894)

[1.13.6 Dicionário de Dados 9](#_Toc129345895)

[1.13.7 Linguaguem de Base de Dados 9](#_Toc129345896)

[1.13.8 Modelo Entidade e Relacionamento 10](#_Toc129345897)

[1.14 API 10](#_Toc129345898)

[1.15 Visual Studio Code 10](#_Toc129345899)

[1.16 MySQL 10](#_Toc129345900)

[1.17 Astah Community 10](#_Toc129345901)

[1.18 Sistema de Informação 11](#_Toc129345902)

[1.19 Arquitetura cliente-servidor 11](#_Toc129345903)

[1.19.1 Cliente 11](#_Toc129345904)

[1.19.2 Servidor 11](#_Toc129345905)

[1.20 Metodologias De Desenvolvimento De Software 11](#_Toc129345906)

[1.21 Tabelas 12](#_Toc129345907)

[**CAPÍTULO II – METODOLOGIA DO ESTUDO** 13](#_Toc129345908)

[**2 Metodologia de Investigação Cientifica** 13](#_Toc129345909)

[**2.1 Metodologia de Desenvolvimento** 13](#_Toc129345910)

[**CAPÍTULO III – APRESENTAÇÃO, ANÁLISE E DISCUSSÃO DOS RESULTADOS OBTIDOS** 14](#_Toc129345911)

[3.1 Requisitos Funcionais 14](#_Toc129345912)

[3.2 Requisitos Não Funcionais 15](#_Toc129345913)

[Tabela 1.2 - Requisitos nã Funcionais 16](#_Toc129345914)

[3.3 Diagramas 16](#_Toc129345915)

[**CONCLUSÕES** 22](#_Toc129345916)

[**RECOMENDAÇÕES** 22](#_Toc129345917)

[**REFERÊNCIAS BIBLIOGRÁFICAS** 23](#_Toc129345918)

[Bibliografia 23](#_Toc129345919)

[**GLOSSÁRIO** 24](#_Toc129345920)

[**APÊNDICES** 24](#_Toc129345921)

[**ANEXOS** 24](#_Toc129345922)

# **INTRODUÇÃO**

Hoje em dia é muito mais fácil fazer compras via online,no passado voce precisava deslocar-se e isso condicionava no seu tempo e dinheiro. Hoje as empresas usam uma tecnologia moderna para engajar os clientes e otimizar operações.

Eventos diz respeito a um conjunto de pessoas que se reunem em um espaço específico para tratar de um assunto de interesse geral.

No nosso projecto iremos nós focar um dois tipos de eventos: Eventos educacionais e eventos culturais.

Eventos educacionais está divdido em duas partes: A primeira diz respeito a pequenas produções feitas em instituições com intuito de engajar os alunos em alguma atividade ou assunto, enquanto que a segunda pode ser grandes produções feitas em congressos,seminários etc. Eventos culturais são concebidas para entreterimento,conhecimento relacionado a arte,moral ,costumes e gozam de um público mais ou menos amplo .

## Contextualização do problema.

Com a expansão dos eventos em angola. Surge a necessidade de adapta-los para que estes consigam satisfazer as necessidades dos clientes passando assim a levar os eventos cada vez mais próximo do cidadão.

## Formulação do Problema

Devido às dificuldades enfrentadas pelos organizadores de eventos na venda de ingressos e frequentadores de eventos na obtenção deste surge o seguinte problema: Qual é o impacto da aplicação Web para venda online de ingressos de eventos?

## Justificativa

Para esta franja da sociedade, o nosso projeto irá permitir redução dos custos e enchentes no processo de vendas dos bilhetes. Permitirá ainda que criadores de eventos publiquem seus eventos sem muita burocracia.Maior facilidade para o acesso e compra de bilhetes, mais agilidade na entrada do evento, comodidade aos convidados e rapidez e melhor ainda, sem precisar se descolocar;

Como estudantes desta instituição acadêmica, do curso de Técnico de Informática e como membro da sociedade angolana,cremos que este tema seja de grande relevância pelas seguintes razões:

* Para a área cientifica é de grande importância, pois irá contribuir de uma forma positiva no diálogo científico que existe sobre o tema, e permitirá que discentes e os docentes universitários tenham mais argumentos nas suas explanações, baseadas nos resultados encontrados na pesquisa feitas, assim também enriquecendo o conhecimento do público em geral sobre o tema apresentado;
* Para os profissionais da área é importante porque os resultados obtidos da pesquisa vão permitir sanar as inquietações dos profissionais e fornece-lhes um vasto leque de conhecimento em relação ao tema, com o propósito de permitir que esses possam responder positivamente a todas dificuldades encontradas em aplicações e contribuir para o seu desenvolvimento;

### Objetivo geral

Desenvolver uma aplicação Web para venda online de ingressos de eventos;

### Objectivos específicos

* Definir áreas de vendas de ingressos;
* Permitir o cadastramento de usuários;
* Listar os eventos;
* Disponibilizar as informações dos eventos;
* Permitir a venda online de ingressos;

# **CAPÍTULO 1 – FUNDAMENTAÇÃO TEÓRICA**

## 1.1 Venda online de ingressos para eventos

Vendas online são operações comerciais feitas pela internet, isto é, produtos e serviços vendidos e distribuídos sem nenhum contato físico entre o vendedor e o cliente final. (Vázquez, 2023)

Nesse tipo de modalidade, o consumidor acessa a loja virtual de uma empresa por meio de um navegador ou por um aplicativo. Feito isso, ele escolhe o produto que deseja comprar, faz o pagamento online e recebe a mercadoria em casa. (Varella, 2022)

Podemos considerar como um evento qualquer acontecimento que reúna pessoas em um espaço físico em torno de um objetivo. De forma mais acadêmica, o livro Manual Prático de Eventos define os eventos como “todos os acontecimentos previamente planejados, organizados e coordenados de forma a contemplar o maior número de pessoas em um mesmo espaço físico e temporal, com informações, medidas e projetos sobre uma ideia, acção ou produto, apresentando os diagnósticos de resultados e os meios mais eficazes para se atingir determinado objetivo.” (Martin, 2017)

Com tantas possibilidades e objetivos diferentes, os eventos são cada vez mais utilizados por profissionais e empresas de todos os setores. Com isso, o público conta com opções variadas para se divertir, se informar, aprender habilidades novas, se movimentar e, é claro, viver novas experiências. Para aceder aos eventos, o publico devera adquirir ingressos. (Alves, s.d.)

Ingresso é um bilhete usado para ingressar em algum estabelecimento, como boates, estádios de futebol, festas, cinema, etc. O ingresso pode ser eletrônico ou físico e deve ser entregue a um trocador antes de se entrar em um estabelecimento.

## 1.2 Importância da venda de ingressos online

Actualmente, estima-se que aproximadamente 90% dos ingressos para grandes eventos, como shows e festivais, são vendidos de forma totalmente online. Oferecendo mais facilidade, comodidade e rapidez na transação, a venda de ingressos pela internet tem se transformado no formato favorito dos produtores de eventos nos mais variados segmentos ao redor do mundo.

## 1.3 Engenharia De Software

Sistemas para computadores, com aplicações práticas e tecnológicas de Gestão de Projetos, entre outras disciplinas, com objetivo na qualidade e na produtividade. Nos dias atuais, essas práticas e tecnologias abrangem uma variedade de categorias, como: Processo e qualidade de software, banco de dados, linguagens de programação, plataformas, padrões de projeto de software e bibliotecas digitais. Além disso, umas das principais finalidades da Engenharia de Software é oferecer as ferramentas necessárias para planejar e gerir todo o processo de desenvolvimento de sistemas computacionais de qualidade e que sejam capazes de atender às necessidades de seu requisitante, isto é, a demanda do mercado. (Stoodi, 2020)

## 1.4 Análise e Desenvolvimento de Sistemas

A Análise e Desenvolvimento de Sistemas é a área profissional responsável por dar uma solução informatizada a um problema, como a criação de um software, um aplicativo ou um banco de dados. Esse problema pode ser originado de uma demanda de uma empresa privada, de uma startup de tecnologia, ou de uma escola que deseja criar um aplicativo para seus alunos, por exemplo. O profissional responsável por executar essas atividades é o analista de sistemas. (Clara, 2018)

### **1.4.1** Análise e Especificação de Requisitos

A analise é o processo de observação e levantamento dos elementos do ambiente onde o software será implantado. Devem-se identificar as pessoas que terão contato com o software, quer seja um contato operacional ou para o fornecimento de informações relevantes para o seu desenvolvimento. A especificação é a forma de comunicação direta entre o analista e a equipe de desenvolvimento do software. Os requisitos compõem o conjunto de necessidades estabelecido pelo cliente/usuário que define a estrutura e o comportamento do software que esta sendo desenvolvido. (Silva, s.d.)

## ****1.5 Linguagem de Modelagem Unificada****

É uma linguagem padrão usada para desenhar a estrutura de projeto de software. A sigla UML significa Unified Modeling Language, que traduzida para o português é exatamente igual ao título deste blog post: Linguagem de Modelagem Unificada. A UML é uma forma de representação gráfica do projeto. Através dela é possível gerar visualizações, construções e especificações de cenários, além da documentação do que foi executado. A UML, portanto, resulta em um diagrama que pode ser dividido em dois tipos, como veremos a seguir.

## 1.5.1 Diagramas

Existem dois tipos de UML: estruturais e comportamento. Os diagramas estruturais são os elementos constantes no software, divididos em sete tipos. Já os comportamentais que também são cinco desenham como o sistema irá operar. (Lucidchart, 2020)

### Diagramas Estruturais

Diagrama de classes É o diagrama UML mais usado, e a principal base de qualquer solução orientada a objetos. Classes dentro de um sistema, atributos e operações, e a relação entre cada classe. Classes são agrupadas para criar diagramas de classes quando há uma diagramação de grandes sistemas. Diagrama de sequência Mostra como objetos interagem entre si, e a ordem de ocorrência. Representam interações para um determinado cenário.

### 1.5.1.2 Diagrama Comportamental

Diagrama de caso de uso Representa uma determinada funcionalidade de um sistema, e foi criado para ilustrar a forma como as funcionalidades se relacionam e seus controladores internos e externos (atores). Diagramas de atividade exibem o fluxo processual do controle entre objetos de classe, juntamente com processos organizacionais, como fluxos de trabalho empresariais. Estes diagramas são feitos de formas especializadas, e então conectados com setas. A notação definida para diagramas de atividade é semelhante aos dos diagramas de estados.

## 1.6 Paradigmas de Programação

Um paradigma pode ser entendido como um tipo de estruturação ao qual a linguagem deverá respeitar. A depender do objetivo proposto, a solução que a linguagem oferecerá obedece a um tipo de paradigma. Portanto, o que vai definir o paradigma utilizado será a tratativa dada ao problema. (Noleto, 2020). Basicamente, existem seis principais tipos de paradigmas. A depender da linguagem de programação utilizada, pode-se usar mais de um deles. Os primeiros foram os paradigmas dos tipos imperativo e declarativo. A partir deles é que todos os outros foram derivados. Paradigma orientado a eventos: Toda linguagem que faz uso de interface gráfica é baseada nesse paradigma. Nele, o fluxo de execução do software é baseado na ocorrência de eventos externos, normalmente disparados pelo usuário. Paradigma orientado a objetos: Esse é, entre todos, talvez o mais difundido. Nesse paradigma, ao invés de construirmos nossos sistemas com um conjunto estrito de procedimentos, assim como se faz em linguagens “fortemente imperativas” como o Cobol, Pascal etc, na orientação a objetos utilizamos uma lógica bem próxima do mundo real, lidando com objetos e estruturas.

### 1.6.1 Programação Modular

A programação modular é o processo de subdividir um programa de computador em subprogramas separados. Um módulo é um componente de software separado. Muitas vezes, pode ser usado em uma variedade de aplicativos e funções com outros componentes do sistema. Funções semelhantes são agrupadas na mesma unidade de código de programação e funções separadas são desenvolvidas como unidades de código separadas, para que o código possa ser reutilizado por outros aplicativos. A programação orientada a objetos (OOP) é ​​compatível com o conceito de programação modular em grande medida. A programação modular permite que vários programadores dividam o trabalho e depuram partes do programa independentemente. (pt.theastrologypage, 2021)

## 1.7 Linguagem de Programação

É uma linguagem formal que, através de uma série de instruções, permite que um programador escreva um conjunto de ordens, ações consecutivas, dados e algoritmos para criar programas que controlam o comportamento físico e lógico de uma máquina. (Carrer, 2019). Programador e máquina se comunicam por meio dessa linguagem, permitindo especificar, com precisão, aspectos como: Quais dados um software deve operar, como esses dados devem ser armazenados ou transmitidos, quais ações o software deve executar, de acordo com cada circunstância variável.

### 1.7.1 TypeScript

TypeScript é uma linguagem de programação de código aberto desenvolvida pela Microsoft. É um superconjunto sintático estrito de JavaScript e adiciona tipagem estática opcional à linguagem. Tipos fornecem uma maneira de descrever a forma de um objeto, fornecendo melhor documentação e permitindo que o TypeScript valide se seu código está funcionando corretamente. Como TypeScript é um superconjunto de JavaScript, os programas JavaScript existentes também são programas TypeScript válidos. (Rosenwasser, 2022)

### 1.7.2 JavaScript

JavaScript (frequentemente abreviado como JS) é uma linguagem de programação interpretada estruturada, de script em alto nível com tipagem dinâmica fraca e multiparadigma (protótipos, orientado a objeto, imperativo e funcional). Juntamente com HTML e CSS, o JavaScript é uma das três principais tecnologias da World Wide Web. JavaScript permite páginas da Web interativas e, portanto, é uma parte essencial dos aplicativos da web. A grande maioria dos sites usa, e todos os principais navegadores têm um mecanismo JavaScript dedicado para executá-lo. (d'Ávila, 2017)

### 1.7.3 Reactjs

React é uma biblioteca JavaScript para construção de interfaces de usuário. (Bhandwalkar, 2018)

### 1.7.4 Nodejs

Node.js é um software de código aberto, multiplataforma, baseado no interpretador V8 do Google e que permite a execução de códigos JavaScript fora de um navegador web. (Metz, 2011). A principal característica do Node.js é sua arquitetura assíncrona e orientada por eventos. O runtime do Node.js é single-thread uma única thread (chamada de Event Loop) é responsável por executar o código Javascript, sem a necessidade de criar novas threads, o que torna o código mais simples de manter. Chamadas que seriam bloqueantes, como entrada/saída, são realizadas de forma assíncrona usando a libuv.

## 1.10 Html

HTML é uma linguagem de marcação utilizada na construção de páginas na Web. Documentos HTML podem ser interpretados por navegadores. A tecnologia é fruto da junção entre os padrões HyTime e SGML. HyTime é um padrão para a representação estruturada de hipermídia e conteúdo baseado em tempo. (Dubost, 2008)

## 1.11 Css

CSS é um mecanismo para adicionar estilos (cores, fontes, espaçamento, etc.) a uma página web, aplicado diretamente nas tags HTML ou ficar contido dentro das tags <style>. Também é possível, adicionar estilos adicionando um link para um arquivo CSS que contém os estilos. Assim, quando se quiser alterar a aparência dos documentos vinculados a este arquivo CSS, basta modifica-lo. (Cederholm, Dan; Marcotte, Ethan, 2010)

## 1.12 Boostrap

Bootstrap é um framework web com código-fonte aberto para desenvolvimento de componentes de interface e front-end para sites e aplicações web, usando HTML, CSS e JavaScript, baseado em modelos de design para a tipografia, melhorando a experiência do usuário em um site amigável e responsivo. (Mark, 2019)

## 1.13 Desenvolvimento da Estrutura de Dados

### 1.13.1 Base de dados

Uma base de dados é uma ferramenta de recolha e organização de informações. As bases de dados podem armazenar informações sobre pessoas, produtos, encomendas ou qualquer outro assunto. Muitas bases de dados começam por ser uma lista num programa de processamento de texto ou numa folha de cálculo. À medida que a lista cresce, começam a aparecer inconsistências e repetições nos dados. Os dados tornam-se difíceis de compreender num formato de lista e as maneiras para efetuar pesquisas ou selecionar subconjuntos de dados para revisão são limitadas. Assim que estes problemas começam a aparecer, convém transferir os dados para uma base de dados criada por uma sistema de gestão de bases de dados (DBMS) como o Access. (Microsoft, s.d.)

### 1.13.2 Sistema De Gerenciador De Base De Dados

Um banco de dados é uma coleção de tabelas relacionadas que são geralmente integradas, vinculadas ou referenciadas a um outro. A vantagem de um banco de dados é que os dados e registros contidos em tabelas diferentes podem ser facilmente organizadas e recuperadas utilizando software de gestão especializado chamado de sistema gerenciador de banco de dados (SGBD) ou gerente de banco de dados. (Ehgomes, s.d.)

### 1.13.4 SGBD Mysql

O MySQL foi criado na Suécia, por David Axmark, Allan Larsson e o finlandês Michael Widenius. Eles começaram o projeto em 1980. O MySQL é um SGBD, um Sistema de gerenciamento de banco de dados, que usa a linguagem SQL como interface. (Pacievitch, s.d.)

Este banco de dados é conhecido por sua facilidade de uso, sendo ele usado pela NASA, HP, Bradesco, Sony, e muitas outras empresas. Sua interface simples, e também sua capacidade de rodar em vários sistemas operacionais, são alguns dos motivos para este programa ser tão usado atualmente, e seu uso estar crescendo cada vez mais.

### 1.13.5 Modelo de Dados

A modelagem de dados consiste em criar um diagrama simplificado de um sistema de software e dos seus elementos de dados. Isso é feito usando texto e símbolos para representar os dados e como eles fluem. Portanto, os modelos de dados fornecem um plano para projetar um novo banco de dados ou fazer a reengenharia de um aplicativo legado. De modo geral, saber o que são modelos de dados ajuda uma organização a usar seus dados de maneira eficaz para atender às necessidades de negócios de informações. Os modelos de dados são fluxograma que ilustram as entidades de dados, seus atributos e os relacionamentos entre as entidades. Ou seja, ele permite às equipes de gerenciamento e análise de dados documentar os requisitos de dados para aplicativos. Dessa forma, podem identificar erros nos planos de desenvolvimento antes que qualquer código seja escrito. (Gaidargi, 2022)

### 1.13.6 Dicionário de Dados

O dicionário de dados consiste numa lista organizada de todos os elementos de dados que são pertinentes para o sistema. Sem o dicionário de dados o modelo não pode ser considerado completo, pois este descreve entradas, saídas, composição de depósitos de dados e alguns cálculos intermédios. O DD consiste num ponto de referência de todos os elementos envolvidos na medida em que permite associar um significado a cada termo utilizado. (Estgv, s.d.)

### 1.13.7 Linguaguem de Base de Dados

SQL é uma linguagem padrão para trabalhar com bancos de dados relacionais. Ela é uma linguagem declarativa e que não necessita de profundos conhecimentos de programação para que alguém possa começar a escrever queries, as consultas e pedidps, que trazem resultados de acordo com o que você está buscando. SQL significa Standard Query Language, literalmente a linguagem padrão para realizar queries. (Silveira, 2022).

### 1.13.8 Modelo Entidade e Relacionamento

O Modelo Entidade Relacionamento de um banco de dados é um tipo de modelagem conceitual, o qual procura representar, de maneira abstrata, os objetos de um domínio de negócios, descrevendo as suas características e relacionamentos. (Estrategiaconcursos, 2022). Os principais elementos em uma modelagem entidade relacionamento de um banco de dados são as entidades, os seus atributos e os relacionamentos entre elas.

## 1.14 API

As APIs são um conjunto de padrões que fazem parte de uma interface. As APIs permitem a criação de plataformas de maneira mais simples e prática para desenvolvedores. A partir de APIs, é possível criar softwares, aplicativos, programas e plataformas diversas. Por exemplo, apps desenvolvidos para celulares Android e iPhone (iOS) são criados a partir de padrões definidos e disponibilizados pelas APIs de cada sistema operacional. (Fabro, 2020)

## 1.15 Visual Studio Code

O Visual Studio Code é um editor de código-fonte desenvolvido pela Microsoft para Windows, Linux e macOS. Ele inclui suporte para depuração, controle de versionamento Git incorporado, realce de sintaxe, complementação inteligente de código, snippets e refatoração de código. (Lardinois, 2015).

## 1.16 MySQL

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês Structured Query Language) como interface. É atualmente um dos sistemas de gerenciamento de bancos de dados mais populares da Oracle Corporation, com mais de 10 milhões de instalações pelo mundo. (MySQL, 2007)

## 1.17 Astah Community

Astah Community é um software para modelagem UML (Unified Modeling Language – Linguagem de Modelagem Unificada) com suporte a UML 2, desenvolvido pela Change Vision, Inc e disponível para sistemas operacionais Windows 64 bits. Anteriormente conhecido por JUDE, um acrônimo de Java and UML Developers Environment (Ambiente para Desenvolvedores UML e Java). (Techtudo, s.d.)

## 1.18 Sistema de Informação

Sistema de Informação, sigla S.I., é um conjunto de componentes inter-relacionados (pessoas, hardware, software, redes de comunicações e recursos de dados) que coletam (ou recuperam), processam, armazenam e distribuem informações destinadas a apoiar a tomada de decisões, a coordenação e o controle de uma organização. Os sistemas de informação computadorizados (SI) utilizam hardware, software, redes de telecomunicações, técnicas de administração de dados computadorizadas e outras formas de tecnologia de informação (TI) para transformarem recursos de dados em produtos de informação. Estes produtos oferecem informações para a tomada de decisão feita pelos gestores.

## 1.19 Arquitetura cliente-servidor

Arquitetura cliente-servidor ou modelo cliente-servidor é uma arquitetura na qual o processamento da informação é dividido em módulos ou processos distintos. Existe um processo que é responsável pela manutenção da informação (servidores) e outro responsável pela obtenção dos dados (os clientes). (Canalti, 2018)

### 1.19.1 Cliente

Solicitam um determinado serviço, através do envio de uma mensagem ao servidor. Enquanto o processo servidor está trabalhando a solicitação, o cliente está livre para realizar outras tarefas.

### 1.19.2 Servidor

Oferecem serviços a processos usuários, ou seja, executam a tarefa solicitada e enviam uma resposta ao cliente que se traduz nos dados solicitados.

### 1.20 Metodologias De Desenvolvimento De Software

As metodologias de desenvolvimento de software consistem, basicamente, no conjunto de abordagens que podem ser utilizadas para a criação de sistemas de processamento de dados. As principais metodologias disponíveis no mercado: Metodologia Ágil trata-se de uma metodologia de desenvolvimento de software com foco no próprio projeto ou produto. Ela visa a realização de melhorias e alterações constantes, baseadas no feedback dos usuários, dos próprios clientes e até do time interno de criação. Sem estruturas rígidas, o processo visa períodos curtos de desenvolvimento, para que os resultados e seus respectivos feedbacks sejam obtidos de maneira rápida. A medologia cascata considerada oposta à metodologia Ágil, a abordagem em Cascata visa a adoção de controles e processos lineares muito rigorosos. Nela, uma nova etapa só pode ser começada quando as anteriores forem concluídas, para que as ações sejam orientadas gradualmente até as próximas responsabilidades previstas. Funcional, gradual e analítico, esse método também é considerado simples, já que impõe uma sequência restrita e bem dimensionada de tarefas.

### 1.21 Tabelas

As tabelas são projetadas para organizar e exibir informações, com dados organizados em colunas e linhas. As tabelas facilitam a compreensão, a leitura e a comparação de números e texto. (Infogram, 2020)

# **CAPÍTULO II – METODOLOGIA DO ESTUDO**

# **2 Metodologia de Investigação Cientifica**

Para elaboração desse trabalho, seguiram-se as seguintes etapas:

O tipo de pesquisa utilizada no presente trabalho foi descritiva e exploratória em relação aos objetivos. Os procedimentos de coleta dos dados supracitados, foi através de pesquisa bibliográfica e documental, com abordagem quantitativa e qualitativa, com o intuito de relacionar os dados para a interpretação. Os dados analisados foram transformados em uma aplicação web para melhor visualização. Assim, os dados foram cruzados e interpretados tanto em quantidade como em qualidade para se constatar a venda online de ingressos.

# **2.1 Metodologia de Desenvolvimento**

Metodologia de desenvolvimento de software representa um conjunto de atividades ou processos que são usados para construir um software.

Para o desenvolvimento deste trabalho, foi adotado o ciclo de vida cascata como metodologia de desenvolvimento de software.

De forma resumida iremos detalhar as linguagens, ferramentas e as tecnologias usadas no projeto: AstahUML (Para a modelagem de software), Linguagem JavaScript (Para o desenvolvimento da aplicação, Nodejs e Reactjs ), Servidor local (Servidor da Aplicação), Linguagem de marcação (HTML), Linguagem de folhas de estilo(CSS).

Após a conclusão dessa etapa, foram especificados os requisitos funcionais e não funcionais. Em seguida, fez-se a modelagem lógica e física do banco de dados e a implementação do aplicativo.

# **CAPÍTULO III – APRESENTAÇÃO, ANÁLISE E DISCUSSÃO DOS RESULTADOS OBTIDOS**

## 3.1 Requisitos Funcionais

Requisitos Funcionais descrevem uma interação entre o sistema e o seu ambiente, são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações.

|  |  |  |
| --- | --- | --- |
| ID | Descrição | Classificação |
| RF01 | A aplicação deverá permitir cadastro de usuários | Essencial |
| RF02 | A aplicação deverá permitir ao usuário consultar eventos. | Essencial |
| RF03 | A aplicação deverá permitir a compra de ingresso para evento. | Essencial |
| RF04 | Permite a autenticação do usuário para entrada na aplicação. | Essencial |
| RF05 | A aplicação deve controlar a situação de um usuário, podendo estar cadastrado. | Essencial |

## 3.2 Requisitos Não Funcionais

Os requisitos não funcionais são restrições aos serviços ou funções oferecidas pelo sistema. Incluem restrições de timing, restrições no processo de desenvolvimento e restrições impostas pelas normas. Ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, muitas vezes, aplicam-se ao sistema como um todo.

|  |  |  |
| --- | --- | --- |
| ID | Atributos | Descrição |
| RF01 | Usabilidade | O sistema terá uma interface amigável, imediata, isto é, suas informações e funcionalidades deverão estar bem visíveis e disponíveis bem como um para os usuários. |
| RF02 | Segurança | O acesso ao sistema será por meio da validação de usuários. O sistema não terá problemas de conflitos de perfis dos usuários, pelo facto do uso de sessões e a distribuição dos ficheiros do sistema nos perfis adequados. |
| RF03 | Sistema | O sistema deverá ser de fácil acesso, um sistema que possa ser utilizado a partir de diferentes plataformas, tanto de hardware como de software, Sistemas Operativos e Navegadores. |
| RF04 | Alertas | As mensagens de alerta serão simples e explicativas sobre os erros gerados. |
| RF05 | Confiabilidade | O sistema devera estar sempre disponível. |
| RF06 | Segurança | O acesso a área administrativa será restrito. |

## Tabela 1.2 - Requisitos nã Funcionais

## 3.3 Diagramas

**3.3.1** Diagrama Entidade Relacionamento - DER

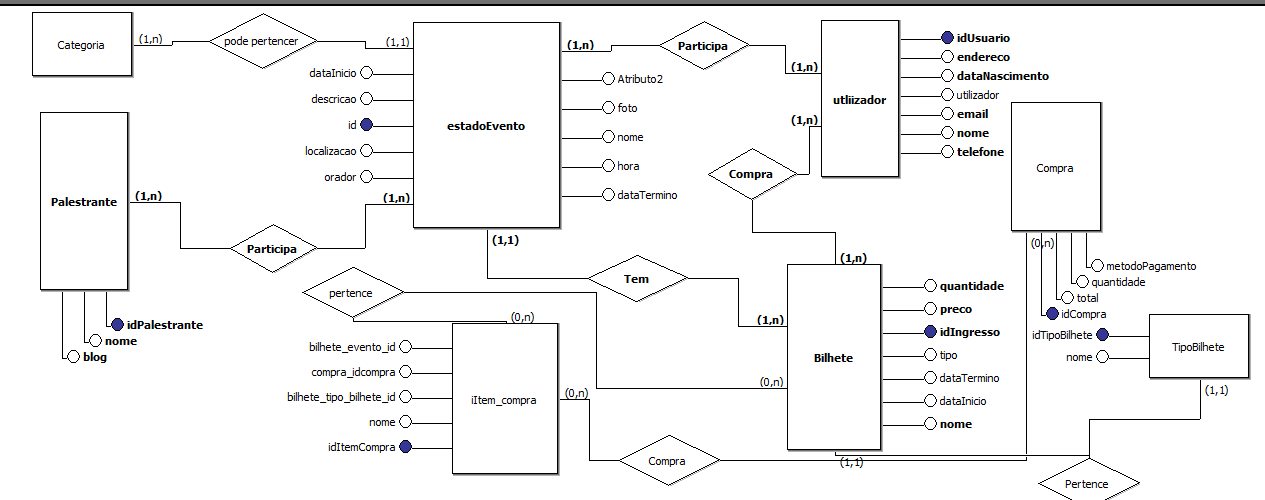
****Ilustra como “entidades”, por exemplo, pessoas, objetos ou conceitos, se relacionam entre si dentro de um sistema.

Figura 1.1 - Diagrama Entidade Relacionamento - DER

**3.3.2 Modelo Relacional**

Representação gráfica do sistema que descreve os relacionamentos entre entidades ou pessoas, objetos, lugares, conceitos ou eventos dentro do sistema.

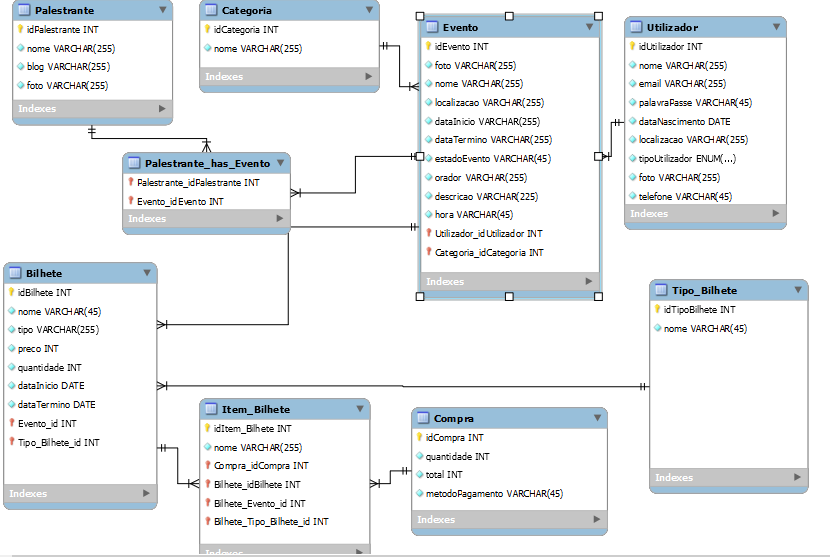


Figura 1.2 - Modelo relaciomento

**2.4.1 Diagramas de Caso de Uso**

São informações que os actores passam para o sistema e informações que o sistema passa par os actores.

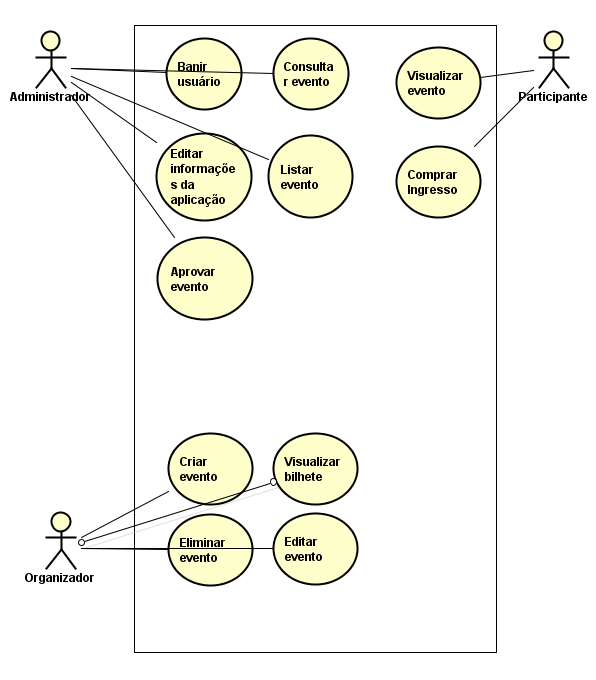


Figura 1.3 - Diagrama de Caso de Uso

**2.4.2 Diagrama de Atividade**

Mostra o fluxo de controle de cada atividade.

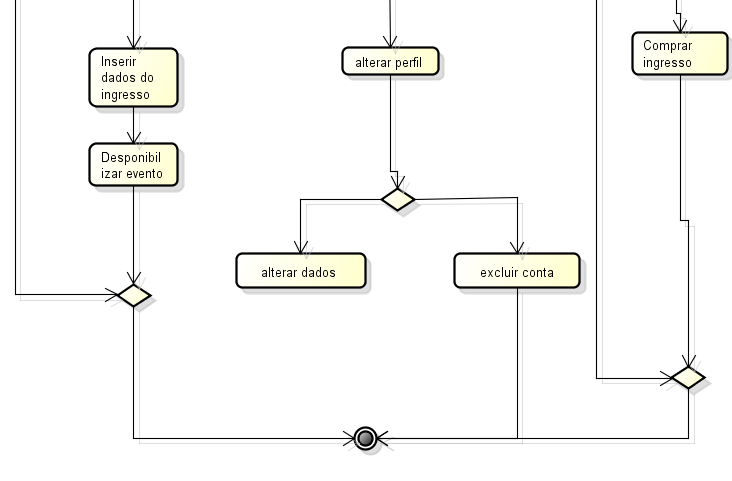
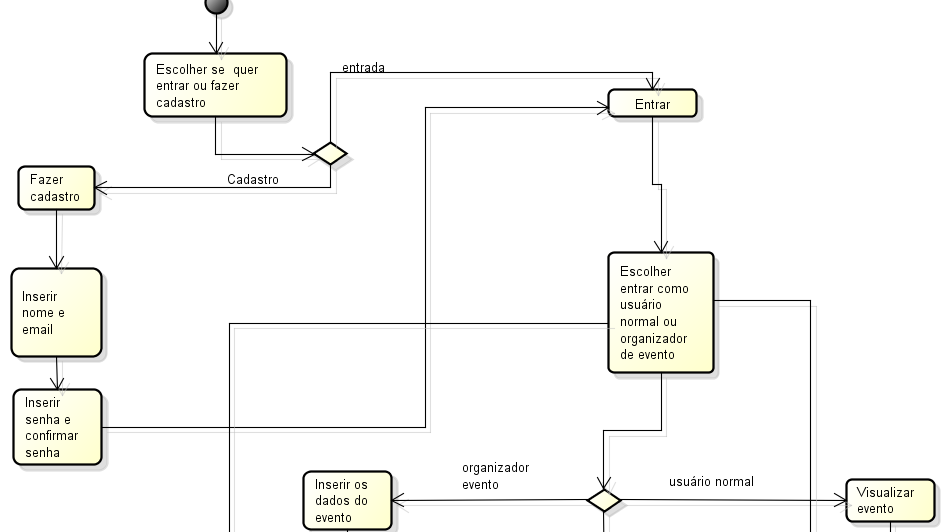


Figura 1.3-Diagrama de Atividade

**2.4.5 Diagrama de Sequência**

Mostra interação entre os objetos de exemplo as mensagens que são passadas entre esses objetos dentro do caso de uso.

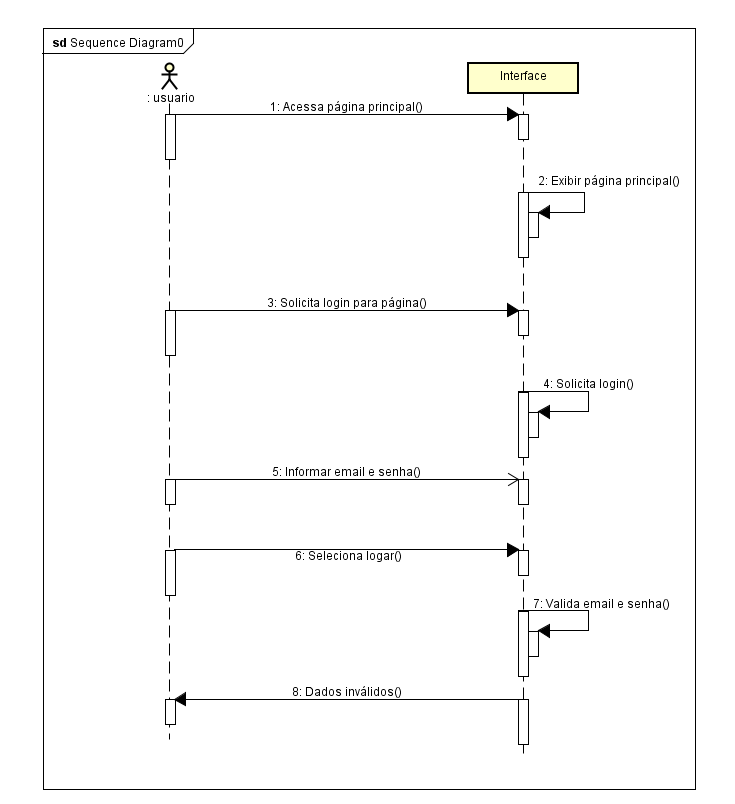


Figura 1.4 - Diagrama de Seguência

**2.4.6 Diagrama de Classe**

Modela os objetos que compõem o sistema, exibindo o relacionamento entre os objetos ou classes, atributos e descrevendo os objetos que fazem e os serviços que eles oferecem.

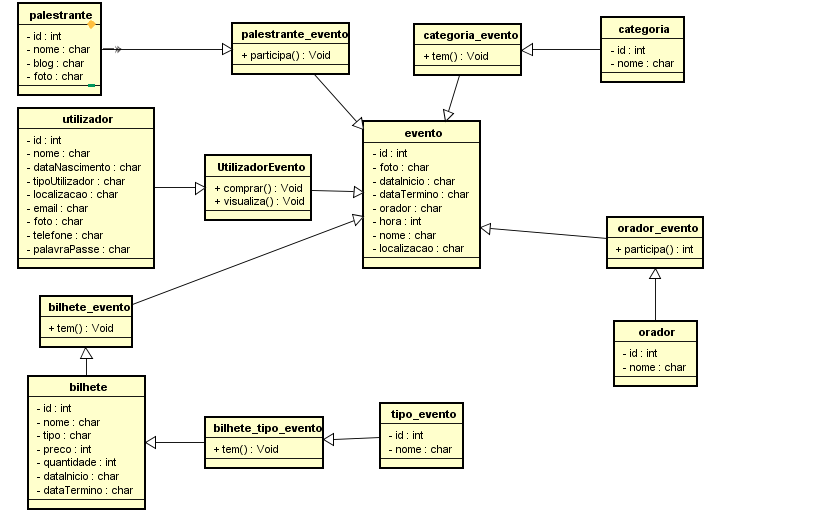


Figura 1.4-Diagrama de Classe

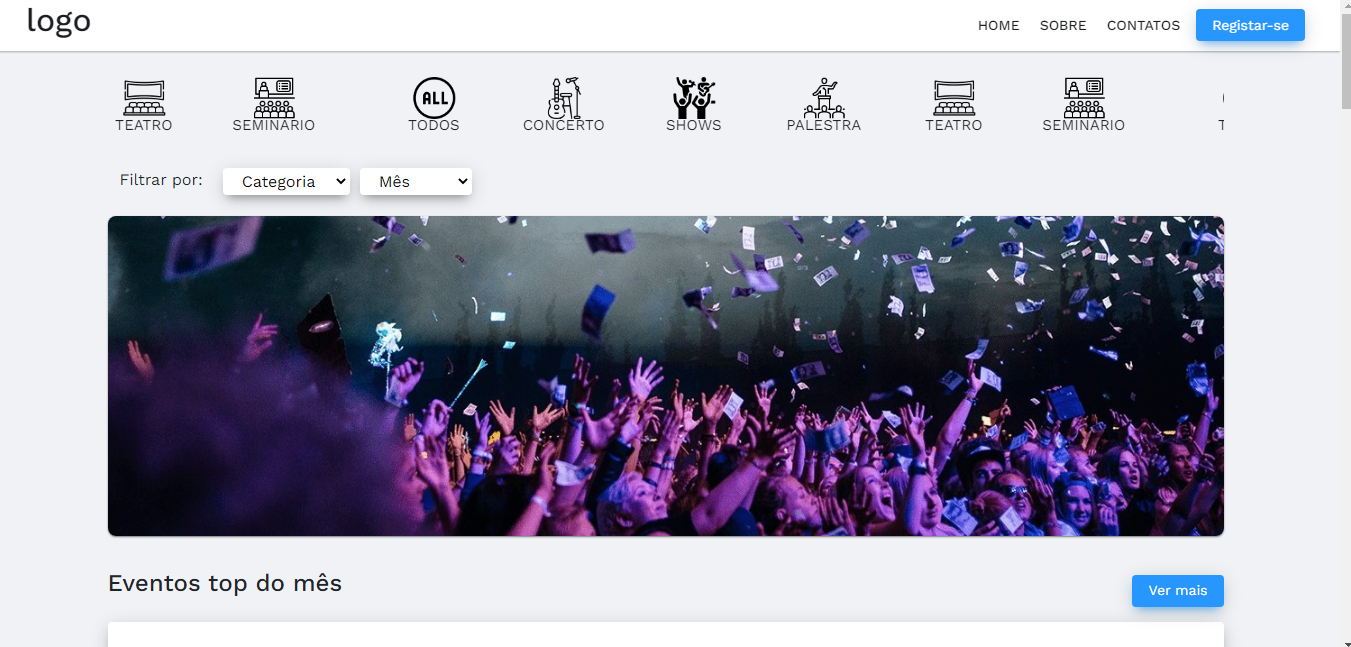


Figura 1.5- Página principal

****

Figura 1.6-Tela de Login

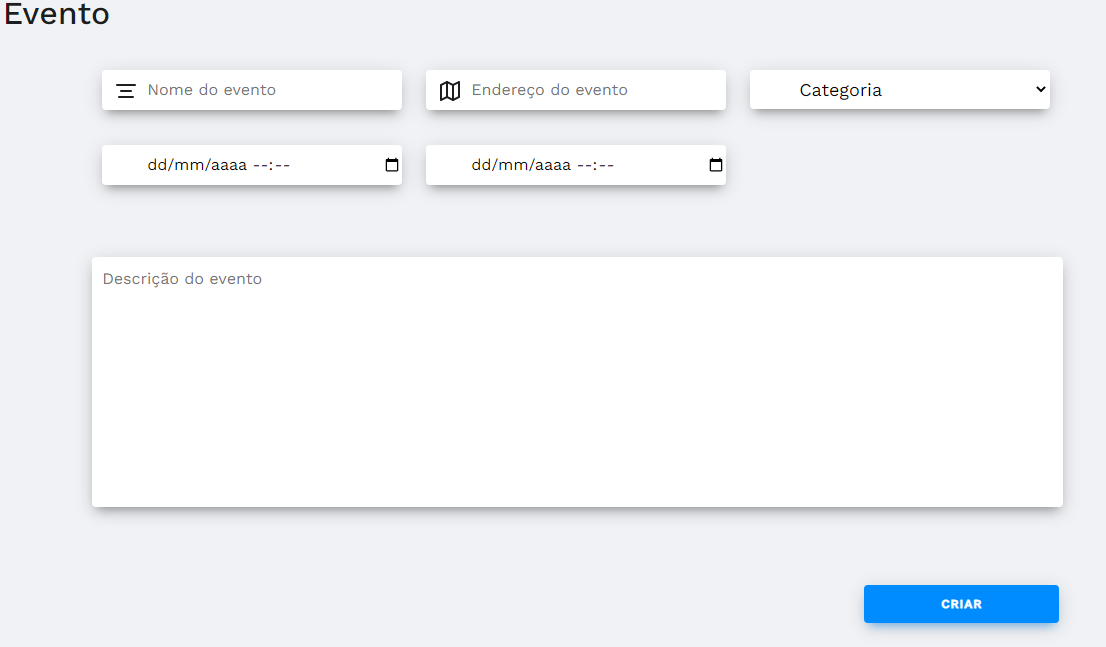


Figura 1.7-Tela de criação de evento

# C:\Users\Kaike Bartolomeu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\cone.png

# **C:\Users\Kaike Bartolomeu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\conex.png**

Figura 1.8-Conexão com a base de dados

# 

# **CONCLUSÕES**

Em suma o nosso sistema de vendas online de ingressos para eventos educacionais e culturais, facilitará a vida dos organizadores de eventos como dos clientes na hora de venda e compra dos ingressos, melhorando a comunicação entre o vendedor e o consumidor.

Sendo que o nosso site pussui uma pulítica de troca que é claramente um especto importante, o consumidor poderá entrar em conctaco com a empresa caso haja algum problema no seu artigo e troca-lo.Cultivando assim uma confiança com público em geral,que gerará um aumentando a credibilidade do seu evento e o crescimento de suas vendas é com certeza um bom motivo pra fazer compras neste site.

O nosso projeto permitirá ainda que os organizadores de evento publiquem seus eventos sem muitas burogracias.Oferecendo assim agilidade na entrada dos seus eventos e comodidade para os seus clientes.

# **REFERÊNCIAS BIBLIOGRÁFICAS**

# Bibliografia

Alves, N. (s.d.). *Sympla*. Fonte: Sympla: https://blog.sympla.com.br/blog-do-produtor/o-que-e-um-evento/

Bhandwalkar, A. (2018). *Cronj*. Fonte: Cronj: https://www.cronj.com/blog/portals-react-js/

Canalti. (27 de 09 de 2018). *Canalti*. Fonte: Canalti: https://www.canalti.com.br/arquitetura-de-computadores/arquitetura-cliente-servidor/

Carrer, F. (14 de 04 de 2019). *Rockcontent*. Fonte: Rockcontent: https://rockcontent.com/br/blog/linguagem-de-programacao/

Cederholm, Dan; Marcotte, Ethan. (09 de 04 de 2010). *Jigsaw*. Fonte: Jigsaw: https://jigsaw.w3.org/css-validator/

Clara, M. (02 de 10 de 2018). *Tuiuti*. Fonte: Tuiuti: https://www.tuiuti.edu.br/blog-tuiuti/o-que-se-faz-em-analise-e-desenvolvimento-de-sistemas

d'Ávila, M. (2 de 02 de 2017). *Mhavila*. Fonte: Mhavila: http://www.mhavila.com.br/topicos/web/window\_modal.html

Dubost, K. (15 de 01 de 2008). *W3*. Fonte: W3: http://www.w3.org/QA/2008/01/html5-is-html-and-xml.html

Ehgomes. (s.d.). *Ehgomes*. Fonte: Ehgomes: http://ehgomes.com.br/disciplinas/bdd/sgbd.php

Estgv. (s.d.). *Estgv*. Fonte: Estgv: http://www.estgv.ipv.pt/paginaspessoais/ajas/AS/Apontamentos%20Te%C3%B3ricos/as\_3\_4.pdf

Estrategiaconcursos. (2022). *Estrategiaconcursos*. Fonte: Estrategiaconcursos: https://www.estrategiaconcursos.com.br/blog/modelo-entidade-relacionamento/

Fabro, C. (15 de 06 de 2020). *TechTudo*. Fonte: TechTudo: https://www.techtudo.com.br/listas/2020/06/o-que-e-api-e-para-que-serve-cinco-perguntas-e-respostas.ghtml

Gaidargi, J. (01 de 10 de 2022). *Infonova*. Fonte: Infonova: https://www.infonova.com.br/gestao-de-ti/o-que-sao-modelos-de-dados/

Lardinois, F. (29 de 04 de 2015). *techcrunch*. Fonte: techcrunch: http://techcrunch.com/2015/04/29/microsoft-shocks-the-world-with-visual-studio-code-a-free-code-editor-for-os-x-linux-and-windows

Lucidchart. (13 de 04 de 2020). *Lucidchart*. Fonte: Lucidchart: https://www.lucidchart.com/pages/pt/o-que-e-uml

Mark, O. (19 de 08 de 2019). *Blog.getbootstrap*. Fonte: Blog.getbootstrap: https://blog.getbootstrap.com/2013/08/19/bootstrap-3-released/

Martin, V. (2017). *Manual Pratico de Eventos.* Elsevier Editora Ltda.

Metz, C. (1 de 03 de 2011). *theregistertheregister*. Fonte: theregister: https://www.theregister.com/2011/03/01/the\_rise\_and\_rise\_of\_node\_dot\_js/

Microsoft. (s.d.). *Support.microsoft*. Fonte: Support.microsoft: https://support.microsoft.com/pt-pt/office/no%C3%A7%C3%B5es-b%C3%A1sicas-da-base-de-dados-a849ac16-07c7-4a31-9948-3c8c94a7c204

MySQL. (22 de 06 de 2007). *mysql*. Fonte: mysql: https://www.mysql.com/why-mysql/case-studies

Noleto, C. (26 de 07 de 2020). *Blog.betrybe*. Fonte: Blog.betrybe: https://blog.betrybe.com/tecnologia/paradigmas-de-programacao/

Pacievitch, Y. (s.d.). *Infoescola*. Fonte: Infoescola: https://www.infoescola.com/informatica/mysql/

pt.theastrologypage. (12 de 09 de 2021). *pt.theastrologypage*. Fonte: pt.theastrologypage: https://pt.theastrologypage.com/modular-programming

Rosenwasser, D. (24 de 04 de 2022). *devblogs.microsoft.com*. Fonte: devblogs.microsoft.com: https://devblogs.microsoft.com/typescript/announcing-typescript-4-7/

Silva, J. G. (s.d.). *Linhadecodigo*. Fonte: Linhadecodigo: http://www.linhadecodigo.com.br/artigo/3224/analise-e-especificacoes-de-requisitos.aspx

Silveira, P. (07 de 10 de 2022). *Alura*. Fonte: Alura: https://www.alura.com.br/artigos/o-que-e-sql

Stoodi. (21 de 10 de 2020). *Stoodi*. Fonte: Stoodi: https://www.stoodi.com.br/blog/carreira/engenharia-de-software/

Techtudo. (s.d.). *Techtudo*. Fonte: Techtudo: https://www.techtudo.com.br/tudo-sobre/astah-commmunity/

Varella, I. (12 de 04 de 2022). *Shopify*. Fonte: Shopify: https://www.shopify.com/br/blog/comercio-eletronico

Vázquez, A. (19 de 01 de 2023). *Nuvemshop*. Fonte: Nuvemshop: https://www.nuvemshop.com.br/blog/como-vender-pela-internet/

# **GLOSSÁRIO**

# **APÊNDICES**

# **ANEXOS**