# 4. RESULTADOS E ANÁLISE DE MODELOS DE DADOS

No capítulo, será apresentado a análise e os modelos de dados do protótipo ‘CHECK-IN TOUR’.

# 4.1 DESCRIÇÕES DO SISTEMA

Visando ajudar a melhorar o modo de trabalho dos pontos turísticos o protótipo ‘CHECK-IN TOUR’ vem com intuito de:

A recepcionista fará o cadastro de todos os visitantes com os seguintes dados: nome, identidade, tipo de visitante, além disso, na hora do cadastro registra automaticamente a hora de entrada.

Poderá gerar relatórios específicos, podendo ser com informações diárias, mensais ou anuais.

# 4.1.1 Diagrama de caso de uso do sistema CHECK-IN TOUR

**Figura 1 – Diagrama de Caso de Uso.**

# 

Fonte: Autoria Própria, 2022.

# 4.1.2 Documentação do caso de uso efetuar login

|  |  |
| --- | --- |
| Casos de uso: | UC01: Efetuar Login. |
| Objetivo: | Acessar o Sistema. |
| Atores: | Recepcionista. |
| Prioridade: | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 🞎 | Essencial | 🞎 | Importante | 🞎 | Desejável | |
| Pré-condições: | 1. Login tem que estar criado. |
| Fluxo Principal: | 1. A recepcionista preenche os campos de login e senha com seus dados. 2. A recepcionista clica no botão de entrar. 3. O sistema executa o Fluxo Alternativo FA01 (Validar Login). 4. Esse caso de uso se encerra. |
| Fluxo Alternativo 01 (Validar Login): | 1. O sistema CHECK-IN TOUR executa o Fluxo de Exceção 01. 2. O sistema CHECK-IN TOUR valida os campos login e senha na tabela user no banco de dados Cadastro. 3. Se os dados preenchidos não estiverem corretos com a base de dados, o sistema mostrara uma mensagem de dados incorretos. 4. Se os dados preenchidos estiverem corretos com a base de dados, o sistema passa para a tela entrada do sistema CHECK-IN TOUR. 5. Esse caso de uso se encerra. |
| Fluxo de Exceção 01: | 1. Caso os campos estejam em branco o sistema apresenta a mensagem que é obrigatório os campos estarem preenchido. 2. Esse caso de uso se encerra. |
| Regras de negócio: | 1. Só será possível acessar o sistema quem tiver Login. |

# 4.1.3 Documentação do caso de uso cadastrar novo visitante

|  |  |
| --- | --- |
| Casos de uso: | UC02: Cadastrar Novo Visitante. |
| Objetivo: | Registrar um novo visitante. |
| Atores: | Recepcionista. |
| Prioridade: | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 🞎 | Essencial | 🞎 | Importante | 🞎 | Desejável | |
| Pré-condições: | 1. Recepcionista tem que estar logada no sistema CHECK-IN TOUR. |
| Fluxo Principal: | 1. A recepcionista irá preencher os campos de nome, identidade, escolher o tipo de visitante. 2. A recepcionista clica no botão de gravar. 3. O sistema CHECK-IN TOUR executa o Fluxo de Exceção 01. 4. O sistema CHECK-IN TOUR valida as informações e salva na tabela visitante no banco de dados cadastro. 5. Na página apresenta a mensagem: Entrada Gravada com Sucesso. 6. Esse caso de uso se encerra. |
| Fluxo Alternativo: | 1. Não Aplicável. |
| Fluxo de Exceção 01: | 1. Caso os campos estejam em branco o sistema apresenta a mensagem que é obrigatório os campos estarem preenchido. 2. Esse caso de uso se encerra. |
| Regras de negócio: | 1. Todo visitante deverá ser cadastrado. |

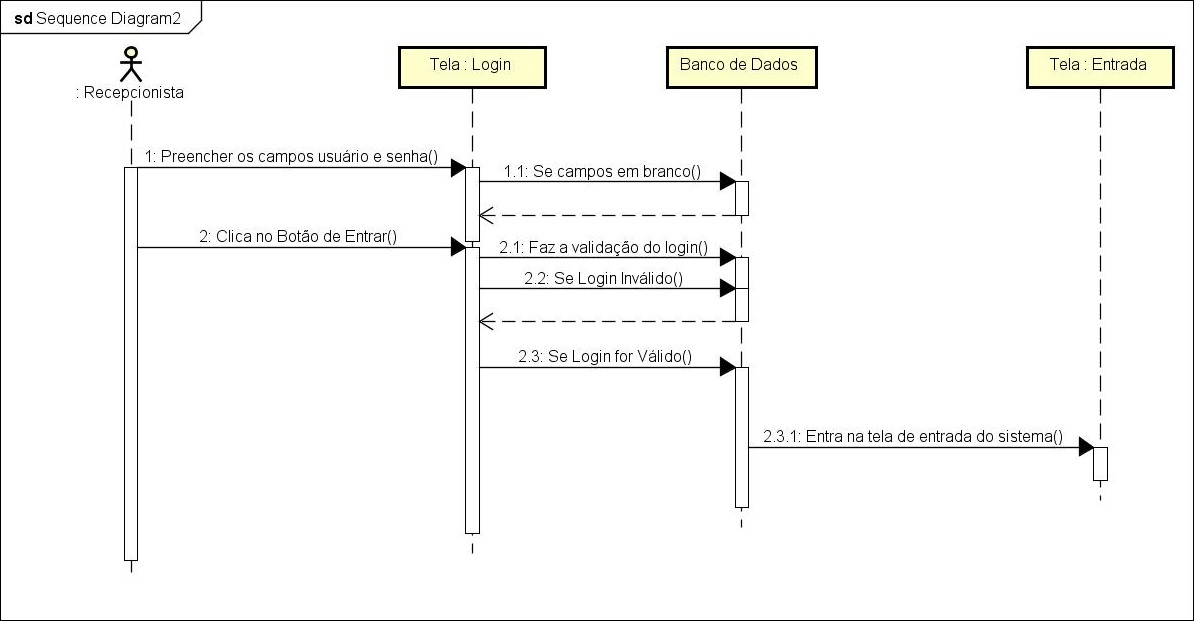
# 4.1.4 Documentação do caso de uso gerar relatório geral de visitante

|  |  |
| --- | --- |
| Casos de uso: | UC03: Gerar Relatório Geral de Visitante. |
| Objetivo: | Gerar relatório geral de visitante. |
| Atores: | Recepcionista. |
| Prioridade: | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 🞎 | Essencial | 🞎 | Importante | 🞎 | Desejável | |
| Pré-condições: | 1. Recepcionista tem que estar logada no sistema CHECK-IN TOUR. |
| Fluxo Principal: | 1. A recepcionista clicar na aba de relatório. 2. A recepcionista clica na opção gerar relatório geral de visitante. 3. O sistema CHECK-IN TOUR valida as informações e faz a busca na tabela visitante no banco de dados cadastro. 4. O sistema CHECK-IN TOUR apresenta na tela o relatório. 5. Esse caso de uso se encerra. |
| Fluxo Alternativo: | 1. Não Aplicável. |
| Regras de negócio: | 1. Pode ser gerado vários relatórios. |

# 

# 4.1.5 Diagrama de sequência efetuar login

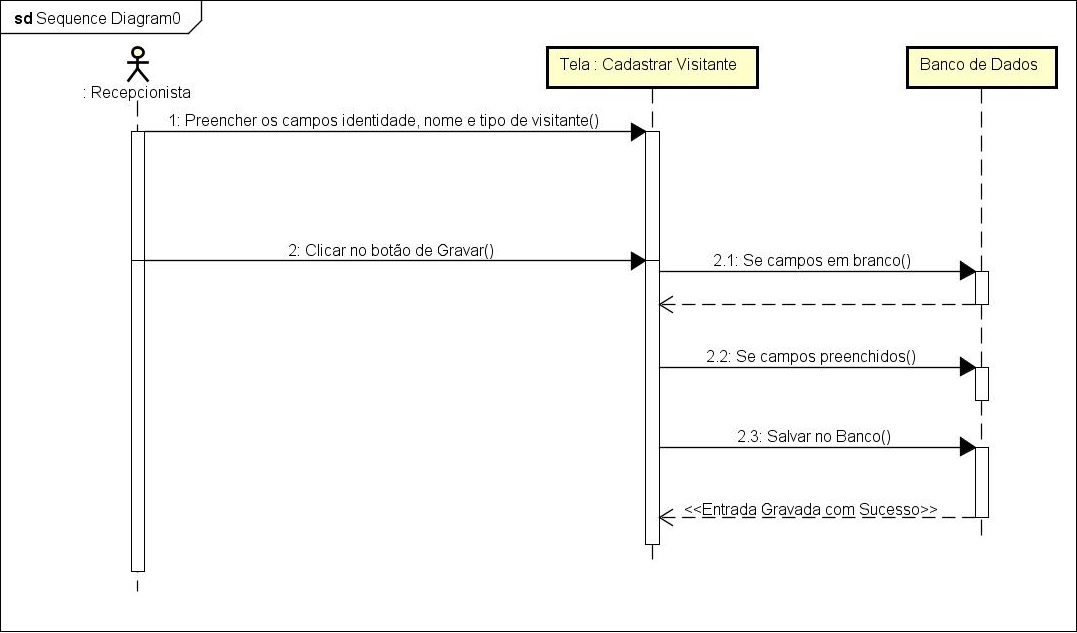
**Figura 2 – Diagrama de sequência efetuar login.**



Fonte: Autoria Própria, 2022.

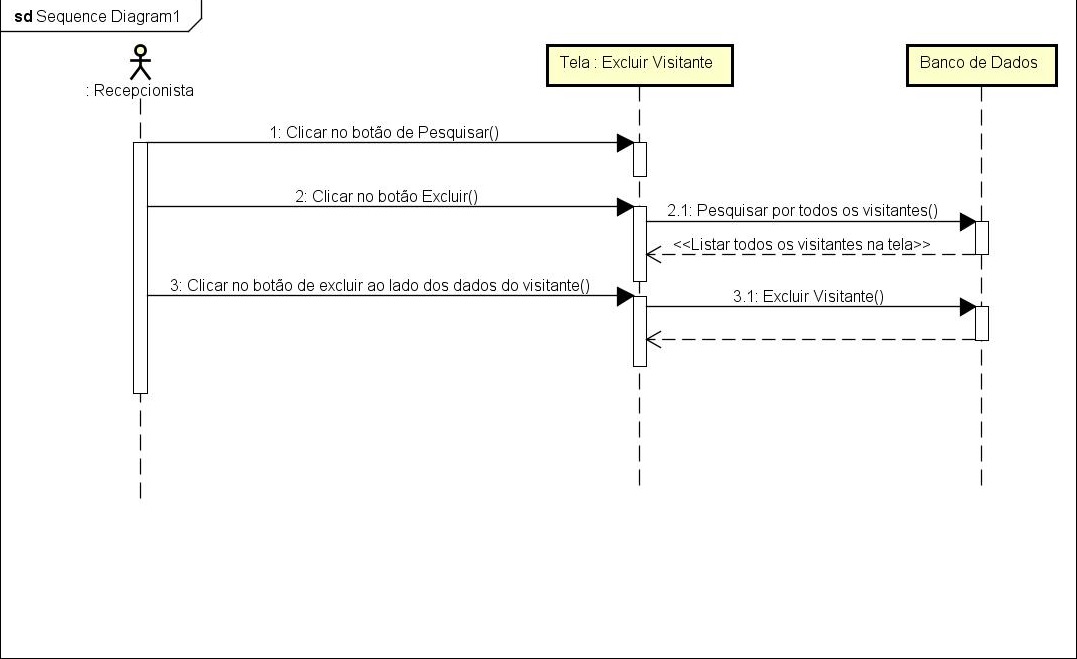
# 4.1.6 Diagrama de sequência cadastrar novo visitante

**Figura 3 – Diagrama de sequência cadastrar novo visitante.**



Fonte: Autoria Própria, 2022.

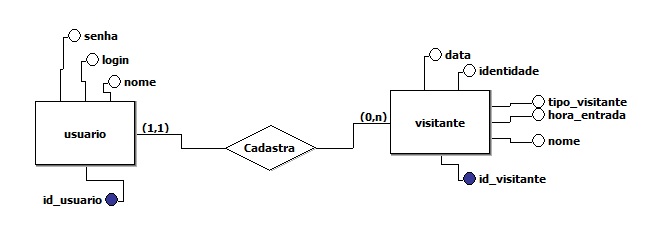
# 4.1.7 Diagrama de sequência excluir visitante

**Figura 4 – Diagrama de sequência excluir visitante**. Fonte: Autoria Própria, 2022.

# 4.2 MODELO DE ENTIDADE E RELACIONAMENTO - MER

# 4.2.1 Modelo conceitual

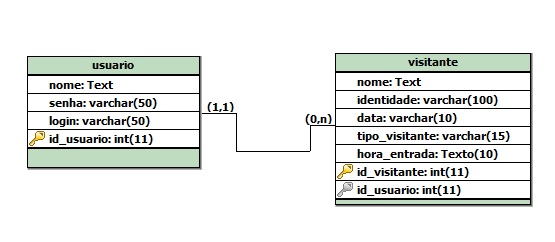
**Figura 5 – Modelo conceitual.**



Fonte: Autoria Própria, 2022.

# 

# 4.2.1 Modelo lógico

**Figura 6 – Modelo lógico**.

Fonte: Autoria Própria, 2022.

# 4.2.2 Modelo físico

CREATE DATABASE cadastro;

CREATE TABLE usuario (

nome Text,

senha varchar(50),

login varchar(50),

id\_usuario int(11) AUTO INCREMENT PRIMARY KEY);

CREATE TABLE visitante (

nome Text,

identidade varchar(100),

data varchar(10),

tipo\_visitante varchar(15),

hora\_entrada Texto(10),

id\_visitante int(11) AUTO INCREMENT PRIMARY KEY,

id\_usuario int(11),

FOREIGN KEY (id\_usuario) REFERENCES usuario (id\_usuario));

# 4.3 PROJETO DE BANCO DE DADOS

# 4.3.1 Dicionário de dados

**Tabela**: Usuario

**Objetivos:** Armazenar dados referentes à tabela usuário.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nome Coluna** | **Tipo** | **Tam** | **Null** | **PK** | **AI** | **FK** | **Tab. Origem** | **Descrição**  **Domínio** |
| Id | Int | 11 | N | S | S | N | N | Atributo único que define o usuário. |
| Nome | Text | - | N | N | N | N | N | Nome do usuário. |
| Login | Varchar | 50 | N | N | N | N | N | Identificador do login do usuário. |
| Senha | Varchar | 50 | N | N | N | N | N | Identificador da senha do usuário. |

**Tabela**: Visitante

**Objetivos:** Armazenar dados referentes à tabela visitante.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Nome Coluna** | **Tipo** | **Tam** | **Null** | **PK** | **AI** | **FK** | **Tab. Origem** | **Descrição**  **Domínio** |
| Id | Int | 11 | N | S | S | N | N | Atributo único que define o visitante. |
| Nome | Text | - | N | N | N | N | N | Nome do visitante. |
| Identidade | Varchar | 100 | N | N | N | N | N | Identificador da identidade do visitante. |
| data | Varchar | 10 | N | N | N | N | N | Identificador da do cadastro do visitante. |
| Hora\_entrada | Varchar | 10 | N | N | N | N | N | Identificador da hora de entrada do visitante. |
| Tipo\_visitante | varchar | 15 | N | N | N | N | N | Identificador do tipo de visitante. |

# 4.3.2 Mapeamento

|  |  |
| --- | --- |
| **PK(#)** – PRIMARY KEY (CHAVE PRIMÁRIA). | |
| TABELA: | Usuario. |
| ATRIBUTOS: | (Id#, nome, login, senha). |
| Obs: | O # representa a chave primária da tabela usuário. |

|  |  |
| --- | --- |
| **PK(#)** – PRIMARY KEY (CHAVE PRIMÁRIA). | |
|  | |
| TABELA: | Visitante. |
| ATRIBUTOS: | (Id#, nome, identidade, data, hora\_entrada, tipo\_visitante). |
| Obs: | O # representa a chave primária da tabela usuário. |

# CONSIDERAÇÕES FINAIS

No período do Projeto Final ll, foi desenvolvido o protótipo ‘CHECK-IN TOUR’, para facilitar o controle de entrada de visitantes nos pontos turísticos, devido à grande demanda de visitantes, muitos pontos turísticos não têm o controle de quantas pessoas passam por ali, pois o controle é feito manualmente.

Levando-se em consideração, foi desenvolvido o protótipo utilizando um padrão MVC (Modelo-Visão-Controle). No processo de desenvolvimento foi utilizado o SCRUM (Metodologia Ágil para Gestão e Planejamento de Projetos de Software), para agilizar o processo sem perder o foco. O protótipo foi feito utilizando a linguagem UML (Linguagem de Modelagem Unificada) e seu desenvolvimento foi através da linguagem de programação PHP (Personal Home Page).

Através do protótipo ‘CHECK-IN TOUR’, os pontos turísticos passam a ter um controle maior de seus visitantes podendo assim gerar relatórios precisos para saber em qual dia, mês ou ano obteve o maior ou menor número de visitas.

# REFERÊNCIAS BIBLIOGRÁFICAS

ALVAREZ, Miguel, **O que é HTML?**, 2004, Disponível em: <http://www.criarweb.com/artigos/7.php>. Acesso em: 19 de setembro de 2022.

PEREIRA, Ana P., **O que é CSS?**, 2009. Disponível em: <https://www.tecmundo.com.br/programacao/2705-o-que-e-css-.htm>. Acesso em: 15 de setembro de 2022.

CARVALHO, Flávia, **WAMPSERVER - Autoria e Designer na Internet I**, 1. ed. Rio Grande sdo SUl: Elsevier, 2010.

COAR, Ken, **Apache Guia Prático**, 1. Ed. Rio de Janeiro: Alta Books,2008.

DATE, C.J.; **Introdução a Sistemas De Bancos De Dados**, 8. ed. Rio de Janeiro: Elsevier, 2004.

DATE, C.J.; **Introdução a Sistemas de Bancos de Dados**, tradução da 4. ed. norte-americana, Editora Campus, 1991.

ELMASRI, Ramez; NAVATHE, ShamkantB, **Sistemas de Banco de Dados**, 6. ed. São Paulo: Addison Wesley, 2011.

BOLZAN, Willian, **DbDesigner**, 2009, Disponível em: <http://www.devmedia.com.br/dbdesigner-modelagem-e-implementacao-de-banco-de-dados/30897>. Acesso em: 15 de setembro de 2022.

GUEDES, Gilleanes T. A.; **Uml 2 – Guia Prático**, 2. ed. São Paulo: Novatec, 2014.

KORTH, H.F. e SILBERSCHATZ, A.; **Sistemas de Bancos de Dados**, Makron Books, 2. Ed. revisada, 1994.

LAMIM, Jonathan, **MVC - O padrão de arquitetura de software**, Santa Cruz do Sul: Oficinadanet, 2012, Disponível em: <https://www.oficinadanet.com.br/artigo/1687/mvc\_-\_o\_padrao\_de\_arquitetura\_de\_software>. Acesso em: 15 de setembro de 2022.

MILANI, Andre, **Construindo aplicações Web em PHP e MySql**, 1. ed. São Paulo: Novatec, 2010

MILANI, Andre, **Construindo aplicações Web em PHP e MySql,** 2. ed. São Paulo: Novatec, 2016.

PINTO, Pedro, **Sublime**, 2013, Disponível em: <https://pplware.sapo.pt/software/sublime-text-e-este-o-melhor-editor-de-texto-do-mundo/>. Acesso em: 15 de setembro de 2022.

SCHWABER, Ken, **Scrum**, 2004, Disponível em: <http://www.desenvolvimentoagil.com.br/scrum/>. Acesso em: 15 de setembro de 2022.

SEABRA, Bruno, **O QUE É O ASTAH?,** 2015, Disponível em:<http://www.startupsstars.com/2015/10/o-que-e-o-astah-posttecnico-por-bruno-seabra/>. Acesso em: 15 de setembro de 2022.

SOUZA, Hermes, **Criando, Inserindo e Exibindo Php + MySql**, 2008. Disponível em: <www.argohost.net>. Acesso em: 15 de setembro de 2022.

WELLCARE, **Controle de entrada e saída de funcionários e visitantes**,Disponível em: <http://www.wellcare.com.br/solucoes/controle-de-entrada-e-saida-de-funcionarios-e-visitantes>. Acesso em: 15 de setembro de 2022.

# APÊNDICE

**Apêndice A – Cronograma do Desenvolvimento do Projeto.**

|  |
| --- |
| ANO 2022 |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| MÊS | Jul | | | | AGO | | | | SET | | |
| SEMANA | **1°** | **2°** | **3°** | **4°** | **1°** | **2°** | **3°** | **4°** | **1°** | **2°** | **3°** |
| Escolha do Tema | **x** | **x** |  |  |  |  |  |  |  |  |  |
| Justificativa |  | **x** |  |  |  |  |  |  |  |  |  |
| Objetivos Gerais e Específicos |  | **x** |  |  |  |  |  |  |  |  |  |
| Referenciais Teóricos (Ferramentas) |  | **x** | **x** |  |  |  |  |  |  |  |  |
| Metodologia |  | **x** | **x** |  |  |  |  |  |  |  |  |
| Análise do Sistema |  |  |  | **x** | **x** |  |  |  |  |  |  |
| Projeto do Banco de Dados |  |  |  |  |  | **x** | **x** |  |  |  |  |
| Projeto de Interfaces |  |  |  |  |  | **x** | **x** |  |  |  |  |
| Desenvolvimento do Sistema |  |  |  |  |  |  | **x** | **x** | **x** | **x** | **x** |

# Atualizações:

# Modelo Lógico

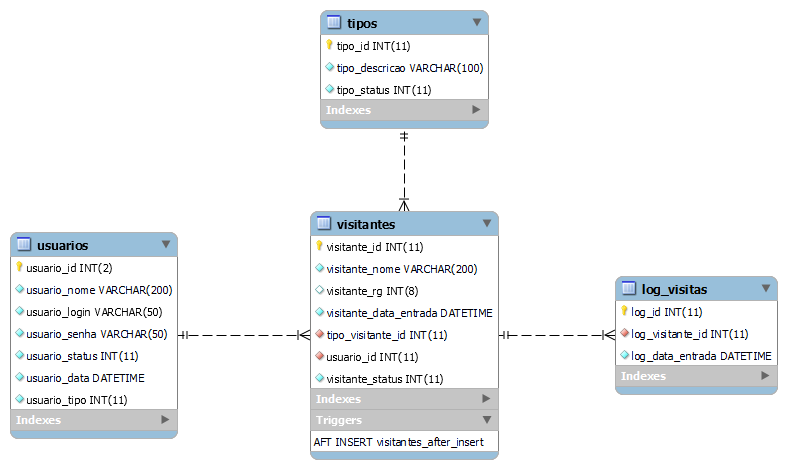


Figura de represantação de relacionamento.