

# **Criação E Implementação De Um Sistema De Gestão Escolar**

**Edilson Miguel de A. Filho<sup>1</sup>, Henrique Egberto C. da Costa<sup>1</sup>, Rodrigo de Araújo L. Rodrigues<sup>1</sup>, Alex Michel F. de Andrade<sup>2</sup>**

<sup>1</sup>Graduandos em Engenharia da Computação do Departamento de Tecnologia e Computação – Universidade Potiguar do Rio Grande do Norte (UNP) - Caixa Postal CEP 59056-000 - Natal – RN – Brasil

<sup>2</sup>Mestre em Engenharia Elétrica e docente do Departamento de Tecnologia e Computação – Universidade Potiguar do Rio Grande do Norte (UNP) - Caixa Postal CEP 59056-000 - Natal – RN – Brasil

dilsonmiguel25@gmail.com, henriquee.costa2010@gmail.com,  
rodrigo.rodrigues.ufrn@outlook.com, alex.andrade@ulife.com.br

**Resumo.** *O sistema de gestão escolar é um produto planejado com o objetivo de alcançar o ambiente educacional no ensino básico no formato de website, sendo um material destinado a facilitar o acesso às informações da instituição, dos gestores e dos alunos. A plataforma permitirá que os pais possam buscar e analisar o status de matrícula, bem como o desempenho individual por disciplina, além de proporcionar um contato direto com a equipe pedagógica. Para alcançar essa meta, serão utilizadas ferramentas de desenvolvimento web, como o Node.js, que será responsável pelo gerenciamento do backend, e a biblioteca Javascript React para a interface de usuário. Ambos serão integrados a um banco de dados programado em MySQL. Com essa abordagem moderna e organizada, o projeto de gerenciamento escolar é uma ferramenta inovadora atendendo aos avanços tecnológicos.*

**Palavras-chave:** Desenvolvimento. Web. Banco de Dados.

## 1. Introdução

O ambiente educacional está se modernizando e as escolas atendendo a um avanço tecnológico estão em busca de ferramentas que proporcione agilidade e melhoria em seus processos internos de gestão escolar, visando o melhoramento das atividades internas com foco na eficiência e na transparência dos resultados.

É importante acompanhar as novas tendências para tornar o projeto eficaz e de fácil manutenção quando na necessidade de atualização e aplicação de novos códigos.

Analisar e determinar qual sistema gerenciador de banco de dados será aplicado, antes de iniciar o desenvolvimento do projeto, é de suma importância para apresentação de um programa eficaz.

Este documento aborda sobre o desenvolvimento e as ferramentas aplicadas na elaboração do projeto de gerenciamento de um banco de dados escolar e a utilidade no mercado educacional. Visando atender um mercado promissor, onde tornasse possível o acesso remoto via *web* por meio de controle de usuário, utilizando dados de login e senha, em interface amigosa que permitirá o manuseio em diferentes plataformas e hardware.

## 2. Objetivo Geral

O objetivo geral desse trabalho é desenvolver um sistema distribuído, capaz de realizar a comunicação cliente-servidor, para uma aplicação de gerenciamento escolar. Foi utilizado ferramentas para a elaboração de uma plataforma de acesso remoto que, permitirá por meio de um servidor, a busca e ou inserção de informações em um banco de dados afastado.

### 2.1. Objetivo Específico

1. Elaboração do formato do banco de dados e definição das ferramentas que serão aplicadas como MySQL, JavaScript, TypeScript, Node.js, e React.js;
2. Elaboração de esboço de interface web como telas de login e dos usuários na ferramenta Figma;
3. Elaboração do banco de dados com tabelas, chaveamento entre tabelas e interações com o servidor por meio do software MySQL;
4. Desenvolvimento da interface de usuário através da ferramenta React.js;
5. Desenvolvimento do link de serviço por API usando o backend entre banco de dados e interface web pela ferramenta Node.js.

### 2.2. Objetivo Motivacional

A motivação da elaboração deste artigo se justifica pela oportunidade de inserir no mercado um produto que possibilite a modernização no setor. Proporcionando ao público mobilidade ao acesso a informações, antes engessadas em arquivos físicos, ou que necessitam de estar presente nos locais escolares para obter informações básicas, como notas, boletos financeiros, relatório e ementas.

Esta mobilidade proporcionará aos professores preenchimento de informações de forma distante, acesso remoto, permitindo ao usuário ganho de tempo para elaborações de outras atividades acadêmicas.

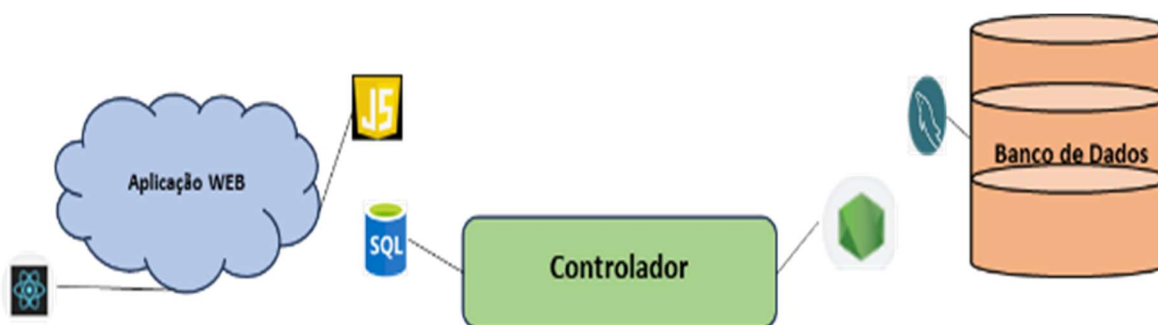
### 3.0 Metodologia de desenvolvimento do software

A metodologia adotada para a elaboração deste projeto foi desenvolvida a partir de experiências realizadas no sistema de banco de dados, como cadastramento de perfis e inserção de dados para criação de um sistema de gestão escolar. Portanto, a partir da realização de algumas configurações foi possível separar os tipos de perfil e quais seriam as permissões para cada usuário. (GIL 2002)

### 4.0. Ferramentas de desenvolvimento

Nesta etapa do documento são apresentados os conceitos básicos de ferramentas e aplicações definidas para a construção do sistema de gerenciamento escolar com acesso via *web* fazendo comunicação através do servidor ao banco de dados.

**Figura 1** – Ferramentas de desenvolvimento



Fonte: Autoral

### 4.1. Arquitetura do Sistema

A arquitetura do sistema é uma estrutura organizada de aplicativos de software estabelecida em três camadas de computação física e lógica independente. Essas camadas são formadas pelas camadas de interface, camada de negócio e camada de dados na **Figura 1** ilustra a formatação das camadas. (IBM, 2023)

Essas camadas foram estabelecidas com as seguintes funções para esse projeto:

- Camada de interface: É a apresentação do software ao usuário do sistema correspondente ao servidor web, desenvolvida para ambiente o dinâmico com as ferramentas de JavaScript e CSS;
- Camada de negócio: É responsável pelo controle das regras estabelecidas, ou seja, processamento e validação dos dados, a partir da lógica de negócio configurada para tratamento das entradas. Desenvolvida no ambiente JavaScript e executada numa estrutura Node.js;
- Camada de dados: É responsável pelo gerenciamento dos dados de forma a estabelecer conectividade entre a base de dados sendo, responsável pela

função de servidor de banco de dados, executada pelo software de gerenciamento de banco de dados MySQL.

#### **4.1.1. Modelos de Padrão de Arquitetura**

Existem uma grande quantidade de modelo de padronização das arquiteturas de sistemas que atendem para cada necessidade de aplicação. Entretanto, serão apresentados modelos de aplicação que sejam destinados especificamente para aplicação web.

- a. O modelo MVC (Model View Controller) é um padrão estrutura com uma abordagem no modelo de controlador, ou seja, esse modelo não permite comunicação entre o *view* e o *model* para acesso a dados e regra do negócio sem que seja passado por ele. Por ser um padrão dividido em três partes distintas permite que seja desenvolvido separadamente. Neste modelo, a manutenção dos sistemas é facilitada devido ser possível ser realizada separadamente sem comprometer a funcionalidade, além da possibilidade de reaproveitamento das aplicações. (ARAÚJO 2007)
- b. O modelo MVP (Model View Presenter) é um padrão com uma abordagem de modelo mediador, onde o “Presenter” responsável por atualizar o “View” quando o “Model” é alterado e de sincronizar o Model com relação ao View. As tratativas de validação dos dados são realizadas no “Presenter” porém, as chamadas para as validações estão configuradas no “View” causando uma maior dependências entre ele. (SANTOS, 2019)
- c. O modelo MVVM (Model View ViewModel) é um padrão com abordagem associativa entre o ViewModel e o View permitindo uma associação entre as propriedades reduzido a dependência de códigos para a execução de sincronismo entre dados e controle. Essa estrutura de dados fornece um acoplamento flexível onde mantém a interface web e os dados sincronizados permitindo que sejam traduzidos antes do envio ao Model. O Model é a camada de modelo do sistema responsável por padronizar os dados para sejam utilizados no ViewModel. (ARAÚJO 2007)

#### **4.2. Banco de Dados MySQL**

É um Sistema de Gerenciador de Banco de Dados (SGBD) relacional que surgiu na década de 1980, utiliza linguagem de consulta estruturada (SQL), sendo um sistema amplamente utilizado devido ter um código livre (Open-Source) e não haver custo para utilização. É um sistema adequado para pequenas e grandes aplicações com capacidade de suportar um volume enorme de dados em cada tabela, sendo, compatível com a maioria dos sistemas operacionais existentes no mercado. (ORACLE, 2023)

Como é um sistema desenvolvido em linguagem C e C++ é possível a portabilidade entre as diferentes plataformas, sistemas e compiladores além de possuir ferramentas de interface para múltiplas linguagens de programação logo, permitindo a manipulação de maneira customizada. Desta forma demonstra algumas vantagens frente a outros gerenciadores, pois, habilita uma melhor programação, disponibiliza funções mais simples e possibilidade de modificação, ou seja, permitindo maior adaptação ao usuário. (ORACLE, 2023)

#### 4.2.1. Script do banco de dados

O script são instruções de comando que determinam as configurações da tabela e a relação entre elas. MILANI (2016)

- CREATE: Instrução de criação da tabel.
- ALTER: Comando de alteração da estrutura da tabela.
- INSERT: Código que permite inserir linhas com dados dentro de uma tabela.
- DELETE: Instrução que permite que dados sejam excluídos da tabela.
- SELECT: É uma instrução que permite a seleção de dados para uma determinada consulta ou execução de uma tarefa.
- UPDATE: Instrução de atualizar de linhas numa tabela.
- KEY: Chave de acesso.
- PRIMARY KEY: tem a função de garantir a integridade da tabela e exclusividade dos dados.
- FOREIGN KEY: é aplicado para estabelecer e impor acesso, ou seja, permitir um link entre duas tabelas controlando os dados que podem ser armazenados na tabela da chave estrangeira.
- CONSTRAINT: Estabelece restrição de acesso à chave estrangeira.
- REFERENCES: tem função de estabelecer privilégio à tabela pai.

#### 4.3. JavaScript

É uma linguagem de programação em script orientada a objetos destinada amplamente a criação de páginas web, estabelecida por meio do padrão ECMAScript onde todos os navegadores mais modernos são orientados pelo padrão ECMAScript 5.1 desde 2012 e os mais antigos pelo padrão ECMAScript 3. É uma linguagem criada pelo programador Brendan Eich. FLANAGAN (2013)

Linguagem destinada à programação das front-end muitas vezes aplicadas em conjunto com HTML e CSS assim sendo uma linguagem de fácil manipulação e com bastante flexibilidade, dinamismo e peculiaridades, logo, se estendeu as aplicações de back-end que devido à alta funcionalidade em diversas aplicações dentre as quais estão as comunicações com banco de dados por de servidores chamados de “server-side”. FLANAGAN (2013)

##### 4.3.1. React.js

O React é um framework criado para desenvolvimento de componentes de interface de usuário em aplicações web, sendo, uma biblioteca do JavaScript declarativa, eficiente e flexível. Tem a funcionalidade de melhorar o processo de atualização do DOM, proporcionando ao programador que sejam elaborados pequenos e isolados códigos na resolução de complexos componentes a partir de chamadas classes (class) em extensão React (**extends** React.Component). (META OPEN SOURCE, 2023)

Nestes componentes estarão inclusas instruções e essas instruções criará uma herança para o `React.Component` que por sua vez permitirá acesso a função `React.Component`. (META OPEN SOURCE, 2023)

#### **4.3.2. Vite.js**

É uma ferramenta que permite criar aplicações front-end de maneira mais rápida e completa, sendo, extremamente útil para desenvolvedores front-end estabelecendo conquistas ao programador. Funciona dividindo o código em duas partes sendo uma em código fonte e a outra em dependências, está última são normalmente arquivos das bibliotecas do Javascript que não sofrerão alterações, logo, são pré-compiladas usando o `USBuild` que é um empacotador Javascript extremamente rápido. (EVAN YOU, 2023)

Contudo, o código fonte é servido como módulos do Javascript nativo, permitindo que o browser auxilie na execução do `Bundle`, portanto, estabelecendo que durante a alteração do arquivo o browser requisite apenas este arquivo, esta modalidade proporciona mais agilidade nas operações `Hot Module Replacement`, essa operação é uma tecnologia que permite alterações no ambiente de desenvolvimento em tempo real. (EVAN YOU, 2023)

#### **4.3.3. Node.js**

`Node.js` é um interpretador JavaScript rápido com vínculos para APIs de baixo nível permitindo trabalhar com processos, arquivos, soquetes de rede e outras aplicações como a compatibilidade em implementações de navegadores web em clientes APIs de servidor. Os vínculos APIs são assíncronas, ou seja, o Node conta com rotinas de tratamento de eventos permitindo que suas aplicações não sejam bloqueadas enquanto esperam o término de operações sendo uma característica que permite segurança ao sistema de gerenciamento durante processos de atualizações de dados entre a aplicação web e o banco. (THE OPENJS FOUNDATION, 2023)

#### **4.4. Figma**

É uma plataforma de desenvolvimento construtivo para auxiliar nos designs de interfaces WEB entre outras aplicações. Esta plataforma foi lançada em 2016 por Dylan Field e Evan Wallace, ambos da empresa Figma, Inc, tendo como objetivo oferecer uma ferramenta gratuita completa de fácil manipulação aos usuários. (FIGMA, 2023)

#### **4.5. Tailwind**

Tailwind é um framework de estilização da interface do usuário que tem como objetivo auxiliar na eficiência do desenvolvimento de aplicações web. Ao utilizar as classes pré-definidas do Tailwind, os desenvolvedores podem estilizar as páginas da web sem precisar escrever CSS do zero. Isso economiza tempo e ajuda a manter a consistência visual em todo o site. (TAILWIND LABS, 2023)

Além disso, o Tailwind oferece uma grande variedade de opções de personalização para atender às necessidades específicas de cada projeto. Com o Tailwind,

é possível criar um design exclusivo e personalizado sem precisar passar por todo o processo de escrever CSS do zero. (TAILWIND LABS, 2023)

## **5.0. Desenvolvimento**

Inicialmente foi elaborado um pequeno projeto para prática do conhecimento adquirido em banco de dados utilizando os métodos de criação de tabelas, chaveamento entre tabelas e tratamento de dados. Logo, este projeto evoluiu para criação de um sistema de gerenciamento de banco de dados escolar explorando o universo da web para acesso remoto do servidor deste banco, via internet a um portal com controle de acesso permitido aos administradores e usuários previamente cadastrados. Entretanto, foi necessário o desenvolvimento de página web que por meio de ferramentas da biblioteca do JavaScript customizou-se uma interface de interação para manipulação dos dados e do banco.

### **5.1. Requisito do Sistema**

- a. Permitir que os usuários administradores façam cadastro dos professores, responsáveis e alunos que de forma remota possam realizar por meio de login acesso ao sistema;
- b. Possibilitar aos usuários professores acesso remoto para consultar e inserir informações acadêmicas no sistema;
- c. Proporcionar aos responsáveis dos alunos acesso em diversas plataformas para consultar informações estudantil e financeira;
- d. Atualizar aos alunos os dados sobre nota das disciplinas ou eventos registrados pelos professores;

### **5.2. Característica do Sistema**

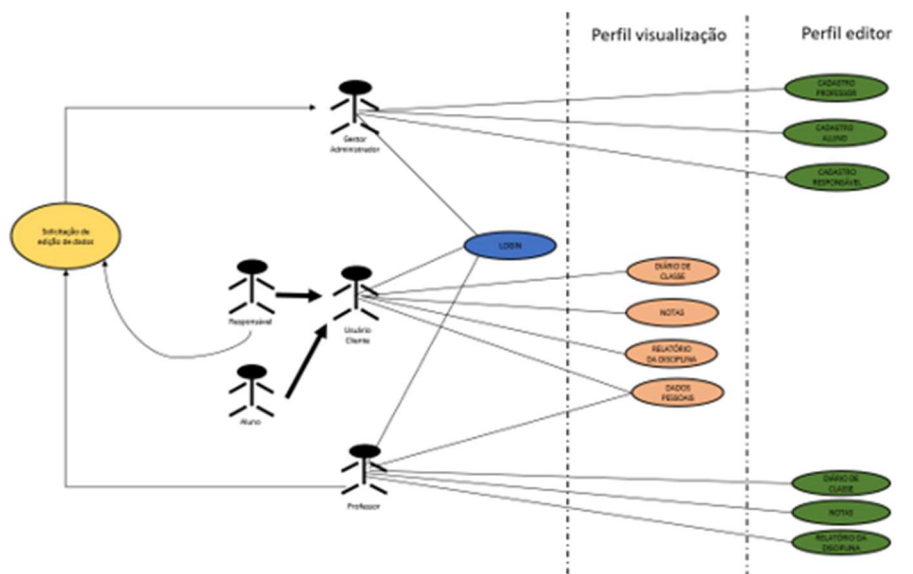
Este sistema foi desenvolvido para usuário com perfis administrador e operador, onde cada usuário tem característica preestabelecida durante o cadastramento ao sistema.

No perfil Administrador é possível o cadastramento de outros usuários como professor, aluno e responsável que por sua vez, cada usuário terá permissões distintas como visualizador de dados e editor parcial de dados.

No perfil de visualizador, as informações estarão estáticas sendo, possível a conferência das informações. Caso necessário correção, fara uma solicitação ao gestor para adequação das informações.

No perfil de edição de dados, estão os professores que terá o cadastro com permissão de adição de informações ao sistema sendo, dados introduzidos aos usuários alunos como nota, diário de classe e relatório da disciplina. Caso ocorra a necessidade de correção de informações, fara uma solicitação ao gestor para que seja realizada a ação. Este fluxo de ações e perfis estão ilustrados no diagrama de caso da **Figura 2**.

**Figura 2 - Diagrama de Caso**



Fonte: Autoral

### 5.3. Interface Web do Sistema de Gerenciamento do Banco de Dados

A interface *web* do **Sistema de Gestão Escolar** é um programa de natureza administrativa para o ambiente escolar com perfil de acesso restrito ao usuário gestor que por sua vez, poderá cadastrar professores, responsáveis e alunos afim, de possibilitar aos usuários o acesso remoto aos conteúdos contido neste sistema.

Na **Figura 3**, é apresentado a *homepage* de acesso com os links destintos de acordo com a funcionalidade, porém, no link LOGIN é possível o acesso ao sistema.

**Figura 3** – Homepage: Sistema de Gestão Escolar



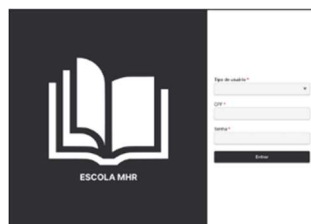
Fonte: Autoral

### 5.4. Interface de Login

A **Figura 4**, é a interface *Login* destinada ao acesso do usuário previamente cadastrado pelo administrador, que deverá selecionar a hierarquia a partir da seleção do *Tipo de Usuário*, preencher nas caixas de texto o número do CPF e a senha, logo ao clicar no botão “Entrar” o usuário será direcionado para a página do seu perfil onde terão acesso as funcionalidades do sistema.



**Figura 4** – Interface de acesso: Tela do Login



Fonte: Autoral

## 5.5. Interface do administrador

O usuário Administrador é o perfil de gestor do sistema, sendo responsável pelo cadastramento, atualização e exclusão dos usuários Professor, Aluno e Responsável. Nestas páginas estão disponíveis os comandos de acesso ao banco de dados de forma que será possível o envio de novas informações assim como consulta.

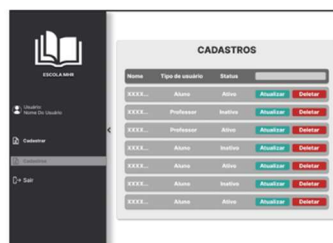
### 5.5.1 Telas da Interface Administrador

- Na **Figura 5**, são mostradas as telas disponíveis ao usuário de forma que cada uma, tem particularidade funcional respeitando a categoria distinta.
- Na **Figura 5.a**, é destinada a escolha do cadastramento de Aluno ou Professor através do botão de link, neste caso ao selecionar uma das funções abrirá uma nova tela destinada a função desejada.
- A **Figura 5.b**, é a tela esperada para a realização de busca de usuário cadastrado onde serão possíveis a visualização do **Nome, Tipo de Usuário e Status**, além da funcionalidade de “Atualizar” e “Deletar” cadastro.
- A **Figura 5.c**, é a tela do cadastro do **Aluno** que permitirá o preenchimento dos dados pessoais como nome, CPF, RG, data de nascimento, telefone, e-mail e sexo sendo, essas informações obrigatórias simbolizada com o asterisco (\*), após o término do preenchimento o usuário irá clicar no botão “Salvar” que os dados serão armazenados no banco.
- Na **Figura 5.d**, é a tela destinada ao cadastramento dos dados pessoais do **Responsável**, após término do preenchimento das informações, deve clicar no botão “Salvar” para encerrar baixar as informações para o banco e em seguida concluir o cadastro do Aluno.
- Nas **Figuras 5.e**, é apresentada a tela de cadastramento do professor. Nesta tela serão preenchidas as informações pessoais do “Professor” como nome, CPF, RG, data de nascimento, telefone, e-mail, sexo, disciplina e status. Assim, concluindo essa etapa e clicando no botão “Salvar” para encerrar baixar as informações para o banco e em seguida concluir o cadastro do Aluno.

**Figura 5** – Interface do usuário: Administrador



**Figura 5.a** – Tela Cadastro: Aluno e Professor



**Figura 5.b** – Tela Cadastro: Atualizar e Deletar

**Figura 5.c** – Tela Cadastro Aluno: Salvar

**Figura 5.d** – Tela Cadastro Responsável: Salvar

**Figura 5.e** – Tela Cadastro Professor: Salvar

Fonte: Autoral

## 5.6. Interface do Professor

Interface do usuário **Professor** com perfil de entrada de dados nas telas destinadas as notas, diário de classe e relatório de disciplina de cada aluno por turma, através da escolha por meio de um *select* de “Busca” afim de procurar o aluno ao qual serão preenchidas as informações distintas. Na tela de dados pessoais o usuário terá perfil de visualizador não sendo possível alteração das informações.

### 5.6.1. Telas da Interface Professor

Na **Figura 6** estão demonstradas as telas disponíveis ao perfil do usuário onde será possível o acesso separadamente a partir da seleção em link no menu verticalizado a esquerda da tela e no centro a direita estará a subtela escolhida.

#### a. Tela Notas

Na **Figura 6.a**, é a tela inicial onde será possível escolher a turma em que deseja realizar o preenchimento das notas. Ao clicar no botão “Digitar Notas” será aberto caixas de texto para permitir que sejam digitadas as notas, também será mostrado as informações da **Turma, Série e Disciplina**;

#### b. Tela Diário de Classe

No menu Diário de Classe abrirá a tela da **Figura 6.b**, onde estará listado as Turmas que o professor administra suas disciplinas, ao identificar a turma que necessita fazer o lançamento das informações, deve clicar sobre o botão “Diário Turma” para realizar o preenchimento das informações. Ao término deverá clicar no botão “Salvar” para registrar no banco de dados.

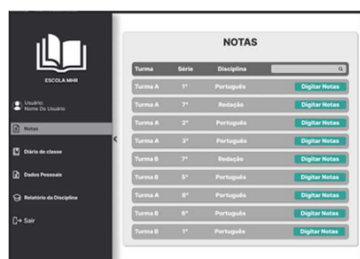
#### c. Tela Dados Pessoais

Na **Figura 6.c** é disponibilizado os Dados Pessoais em formato visualizador.

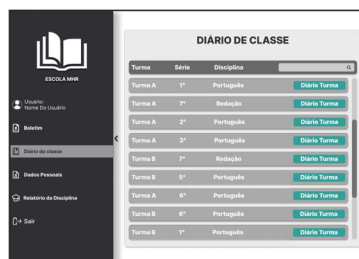
#### d. Tela Relatório da Disciplina

No menu Relatório da Disciplina abrirá a tela da **Figura 6.d**, onde estarão listados as Turmas que o professor administra suas disciplinas, ao identificar a turma que deseja fazer os lançamentos das informações, deve clicar sobre o botão “Digitar Relatório” para habilitar a opção de escolha do aluno da turma e série em que deseja registrar as informações.

**Figura 6** – Interface do usuário: Professor



**Figura 6.a** – Tela Professor: Notas



**Figura 6.b** – Tela Professor: Diário de Classe



**Figura 6.c** – Tela Professor: Dados Pessoais



**Figura 6.d** – Tela Professor: Relatório de Disciplina

**Fonte:** Autoral

## 5.7. Interface do Aluno

Interface destinada ao usuário aluno com perfil de visualizador sem permissão de alteração dos dados com o objetivo de acesso a informações como dados pessoais, notas das disciplinas, relatório da disciplina e diário de classe.

### 5.7.1. Telas da Interface Aluno

Na **Figura 7** estão demonstradas as telas disponíveis ao perfil do usuário onde será possível o acesso separadamente a partir da seleção em link no menu verticalizado a esquerda da tela e no centro a direita estará a subtela escolhida.

#### a. Tela Notas

Na **Figura 7.a** é possível o acesso à tela **NOTAS** onde serão apresentadas as notas preenchidas pelo usuário professor com resultados referentes as Nota 1, Nota 2, Nota 3 e Média de cada disciplina.

#### b. Tela Diário de Classe

Na **Figura 7.b** é apresentado a tela **DIÁRIO DE CLASSE** será possível acessos as informações preenchidas pelos professores de cada disciplina estando registrado na caixa de texto no canto superior esquerdo a data do preenchimento realizado educador.

#### c. Tela Dados Pessoais

A **Figura 7.c** é referente a tela dos **DADOS PESSOAIS**, logo, ficarão as informações pessoais registradas durante o processo de cadastro do aluno feito pelo usuário administrador, essas informações estarão disponíveis apenas

como leitura no usuário **Aluno**, porém estes dados estão sendo consultados no banco de dados do servidor.

d. **Tela Relatório da Disciplina**

Na **Figura 7.d** serão as informações do **RELATÓRIO DA DISCIPLINA**, o **Aluno** terá como possibilidade de consultar o relatório parcial de cada disciplina, onde na caixa de texto terá o resumo descrito pelo professor, sendo possível escolher a disciplina na caixa de seleção localizada no canto superior esquerdo.

**Figura 7** – Interface do usuário: Aluno



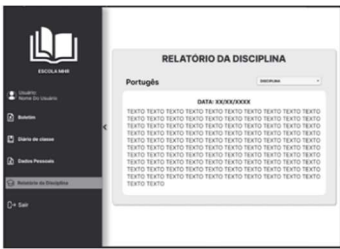
**Figura 7.a** – Tela Aluno: Notas



**Figura 7.b** – Tela Aluno: Diário de Classe



**Figura 7.c** – Tela Aluno: Dados Pessoais



**Figura 7.d** – Tela Aluno: Relatório de Disciplina

**Fonte:** Autoral

**5.8. Desenvolvendo as Tabelas do Banco de Dados**

Neste tópico são descritas as instruções necessárias para criar tabelas assim como inserir, consultar, atualizar, excluir, filtrar e ordenar dados, armazenados no banco de dado. Portanto, é necessário um planejamento sobre a relação entre as tabelas. Na **Figura 8**, é apresentado o digrama de relacionamento entre tabelas do banco de forma a estabelecer o fluxo dos dados os locais armazenados no sistema.

**Figura 8** – Diagrama Banco de Dados



estrangeira a tabela Endereço (tbendereco) de forma a completar as informações deste cadastro.

**e. Tabela Professor**

Na tabela tbprofessor apresentada na **Figura 8**, estão relacionadas informações referentes ao “Professor” cadastrado no sistema, onde no IdProfessor constará o dado único a ele respeitando a excepcionalidade daquele cadastro que terão informações pessoais sensíveis ao indivíduo. Esta tabela por sua vez está vinculada por meio de acesso restrito em chave estrangeira as tabelas Endereço (tbendereco) e Disciplina (tbdisciplina) de forma a completar as informações deste cadastro.

**f. Tabela Diário**

A tabela tbdiario, mostrada na **Figura 8** é destinada ao armazenamento dos dados transcritas pelos professores, a fim de informar a turma diariamente sobre atividades realizadas ou lembretes de cobrança, esta ferramenta servirá como meio de comunicação com a turma. Essa tabela está configurada para manter um relacionamento entre as tabelas Disciplina (tbdisciplina), Professor (tbprofessor) e Turma (tbturma) através das tags IdDisciplina, IdProfessor e IdTurma.

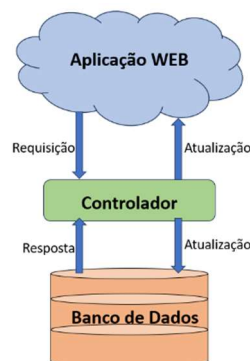
**g. Tabela Registro Diário**

A tabela tbregistrodiario, tem como finalidade registrar o preenchimento realizado pelo professor em Diário de Classe na tabela Diário (tbdiario) na tag IdDiario, assim como os vínculos com as tabelas Disciplina (tbdisciplina), Professor (tbprofessor) e Turma (tbturma) através das tags IdDisciplina, IdProfessor e IdTurma. Na **Figura 8**, estão listados os relacionamentos entre tabelas.

## 5.9. Padrão de Arquitetura Escolhido para o Sistema

O modelo MVC foi escolhido devido a possibilidade de manutenção mesmo com o sistema em funcionamento. Esta arquitetura modular permitiu aos desenvolvedores que de forma distintas que elaborassem as configurações do sistema independentemente sem comprometer o tempo de elaboração do projeto. (SANTOS, 2019)

**Figura 9 – Arquitetura do Sistema**

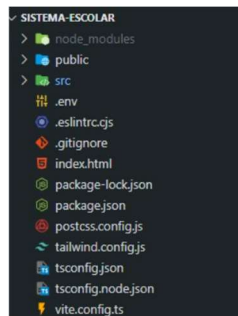


**Fonte:** Autoral

### 5.9.1. Estrutura da aplicação WEB

É a estrutura usada para a elaboração da interface do usuário onde são configurados por meio de lógica de relacionamento com o controller a fim de enviar e buscar dados do sistema de gerenciamento de banco de dados.

**Figura 10 - Estrutura web**

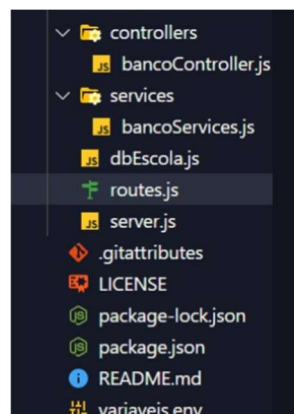


**Fonte:** Autoral

### 5.9.2. Estrutura de Controle

Esta estrutura é definida com o objetivo de tratar as interações e controlar a relação do *view* ao sistema do banco com a funcionalidade de fazer cumprir os critérios de relacionamento tratando as rotas de endereçamento dos dados a fim de evitar a quebra do link e perda dos dados.

**Figura 11 – Estrutura do Controlador**



**Fonte:** Autoral

### 5.9.3. Estrutura de Negócio

Na estrutura de negócio é a camada responsável pelo domínio da aplicação contendo as regras de relacionamento entre as diversas tabelas e a organização do armazenamento dos dados.

## 6. Conclusão

O projeto do sistema de gestão escolar foi uma oportunidade de desenvolver uma ferramenta destinada às rotinas administrativas no ambiente escolar, oferecendo diversos recursos úteis como o cadastro de alunos, responsáveis e professores, além da

possibilidade de registro de notas e diários de classe. Entretanto, as funcionalidades planejadas inicialmente estão parcialmente entregues devido, a complexidade das relações entre tabelas do banco de dados porém, as paginas de interface web destinada aos usuários estão prontas para serem acopladas ao Controller da arquitetura desenvolvida.

Os avanços dos relacionamento entre as tabelas concluídas no banco de dados e o desenvolvimento das interface web, permitiu que o sistema fosse integrado na arquitetura do software, para realização de testes e definição do modelo. O padrão MVC foi o modelo escolhido devido, possibilitar a manutenção e a integração das atualizações com o sistema em funcionamento. Permitindo que os desenvolvedores continuem trabalhando no aprimoramento do projeto.

Uma das melhorias que serão implementadas futuramente incluem a criação de relatórios mais detalhados e informativos sobre o desempenho dos alunos, assim como a adição de mais interfaces para analisar dados e visualizar as turmas de alunos. Tudo isso, com o objetivo de oferecer uma gestão ainda mais completa e eficiente para as instituições de ensino. Com essas melhorias, gestores e professores poderão tomar decisões mais assertivas e contribuir ainda mais para o sucesso dos alunos.

## 7. Referências

ARAÚJO, Kleitor Franklint Correa de. Java EE 5 - guia prático: Scriptlets, Servlets, JavaBeans. 2. ed. São Paulo: Editora Érica Ltda, 2007.

EVAN YOU. Documentação Vite.js. Elaborada por Evan You e contribuidores do Vite. Disponível em: <<https://vitejs.dev/>>. Acesso em: 20 maio 2023.

GIL, Antonio Carlos. **Como elaborar projeto de pesquisa**. 4. ed. São Paulo: Atlas, 2002.

FIGMA. Documentação Figma. Disponível em: <<https://www.figma.com/best-practices/guide-to-developer-handoff/>>. Acesso em: 20 maio 2023.

FLANAGAN, David. JavaScript O guia definitivo. 6. ed. Porto Alegre: Bookman, 2013.

META OPEN SOURCE. Documentação React. 2023. Disponível em: <<https://react.dev/>>. Acesso em: 20 maio 2023.

MILANI, André. CONSTRUINDO APLICAÇÕES WEB COM PHP E MYSQL. 2. ed. São Paulo: Novatec, 2016.



OLIVEIRA, Claudio Luis Vieira. ZANETTI, Humberto Augusto Piovesana. Node.js: Programe de forma rápida e prática. São Paulo: Expressa, 2022.

ORACLE. Introdução ao MySQL. Disponível em: <<https://dev.mysql.com/doc/>>. Acesso em: 25 maio 2023.

REZENDE, Ricardo. Conceitos de banco de dados. Disponível em: <[http://www.sqlmagazine.com.br/Colunistas/RicardoRezende/02\\_ConceitosBD.asp](http://www.sqlmagazine.com.br/Colunistas/RicardoRezende/02_ConceitosBD.asp)>. Acesso em: 01.06.2023

SANTOS, Emanuel Sistema de informação de gestão de encomendas da atividade de tratamento naval. 2019. 87 f. Dissertação (Mestrado) - Curso de Engenharia de Software, Instituto Politécnico de Setúbal, Setúbal, 2019.

TAILWIND LABS. Documentação Tailwind CSS. Disponível em: <<https://tailwindcss.com/>>. Acesso em: 20 maio 2023.

THE OPENJS FOUNDATION. Documentação Node.js. Disponível em: <<https://nodejs.org/en/docs>>. Acesso em: 20 maio 2023.

IBM.O que é arquitetura de três camadas? Disponível em: <<https://www.ibm.com/br-pt/topics/three-tier-architecture>>. Acesso em: 10 jun. 2023.