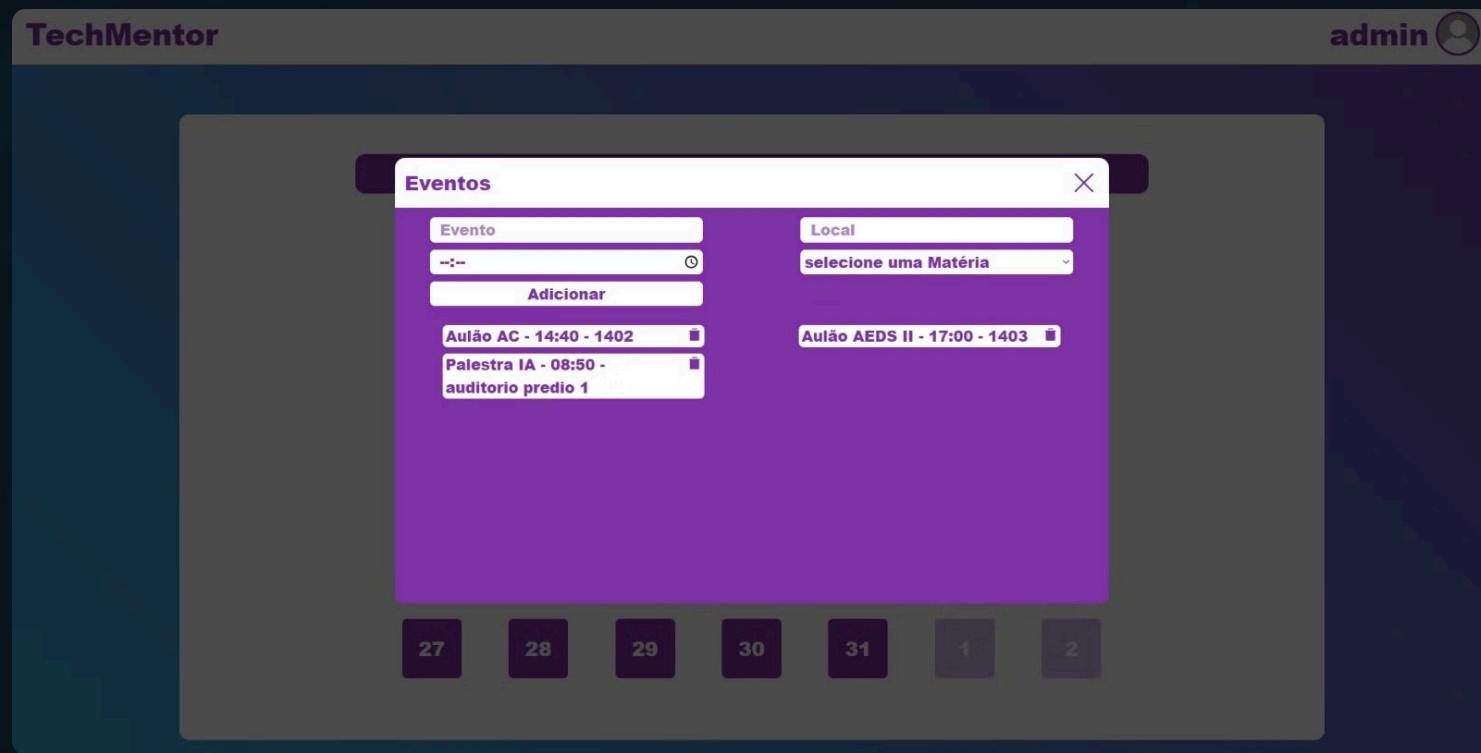


# Techmentor

Implementação do backend, conexão com o banco de dados e especificação de sistemas inteligentes

# Frontend e Backend Vinculados

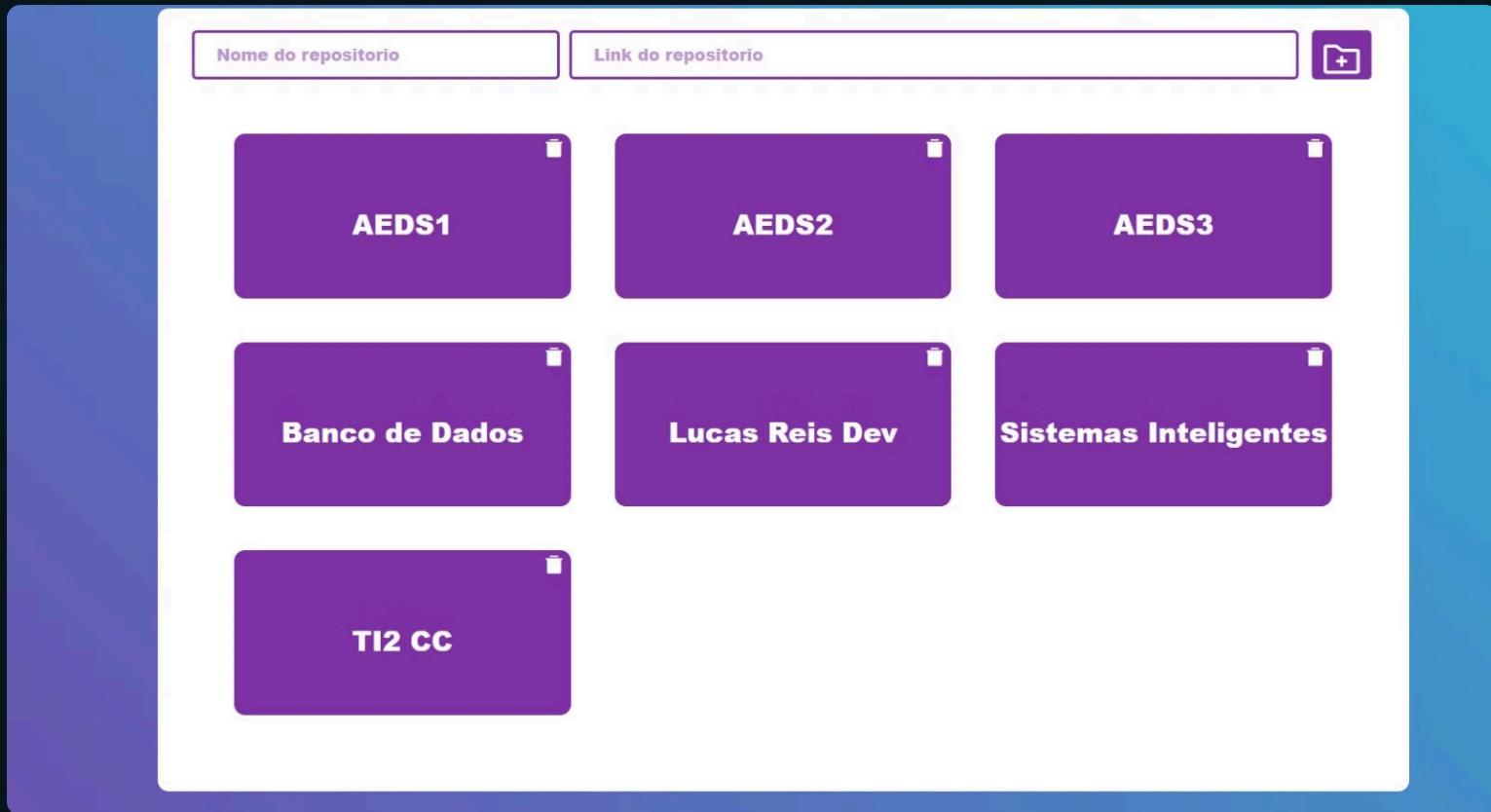


A screenshot of a web application interface titled "Monitores". The page displays a list of monitor assignments:

Monitor	Curso	Remoção
Felipe	Introdução aos Sistemas Inteligentes	-
Lucas Marinho	Algoritmos e Estruturas de dados I	-
Lucas Teixeira	Algoritmos e Estruturas de dados I	-
Luisa Jardim	Cálculo I	-
Matue	Algoritmos e Estruturas de dados III	-
Natsumi	Eletrônica para Computação	-

Below this section is another titled "Alunos" which contains two entries:

Aluno	Matéria	Ação
Alvim	selecione uma Matéria	+
Augusto Bizarria	selecione uma Matéria	+



## Formulário de Acompanhamento de Monitoria 2024/2

**Formulário para preenchimento pelos monitores para acompanhamento dos atendimentos na monitoria do segundo semestre de 2024.**

**1. Disciplina abordada:**

**2. Data do atendimento:**

**3. Matricula do aluno:**

**4. Dúvida do aluno: Indique os temas separados por ponto e vírgula (Ex.: ponteiros; derivada; sistemas de numeração).**

**. Dúvida do aluno: Insira uma breve descrição a respeito da dúvida do aluno (Ex.: Dúvida sobre declaração e aritmética de ponteiros; Aluno não sabia aplicar a regra da cadeia; Dificuldades**

# Back-end e Banco de Dados Vinculados

```
public void registrarPessoa(Pessoa pessoa) throws SQLException{
    try(Connection conn = DriverManager.getConnection(url, user, password))<
        String sql = "INSERT INTO pessoa (id, nome, email, senha, tipo_usuario) VALUES (?, ?, ?, ?, ?)";
        PreparedStatement pstmt = conn.prepareStatement(sql);

        pstmt.setInt(1, pessoa.getId());
        pstmt.setString(2, pessoa.getNome());
        pstmt.setString(3, pessoa.getEmail());
        pstmt.setString(4, pessoa.getSenha());
        pstmt.setInt(5, pessoa.getTipoUsuario());

        pstmt.executeUpdate();
    }
}
```

0	admin	admin@admin.com	0	admin	[NULL]
123	Capanema	capanema@gmail.com	1	123	[NULL]
321	Rommel	rommel@gmail.com	1	123	[NULL]
1.524.619	Natsumi	usuario@example.com	2	senhaDoUsuario	[NULL]
12.131.415	Felipe	felipe@gmail.com	2	teste	[NULL]
1.269.484	Lucas Teixeira Reis	luquinhas@cinema	2	senhadocinema	[NULL]
1.275.782	Luisa Jardim	luisa@jardim	2	luisasenha	[NULL]
1.523.672	Lucas Grossi	lucas@grossi	3	luquinhas	[NULL]
1.528.669	Alvim	alvim@gmail	3	alvimsenha	[NULL]

# Estrutura de Pastas do Projeto

```
Codigo
└── src
    └── main
        ├── java
        │   ├── app
        │   ├── config
        │   ├── controller
        │   ├── dao
        │   ├── model
        │   └── service
        ├── resources
        │   ├── config
        │   │   └── config.properties
        │   ├── db
        │   │   └── schema.sql
        │   └── public
        │       ├── assets
        │       ├── css
        │       ├── html
        │       └── js
        └── test
exec
pom.xml
README.md
```

# DAO, Model, Service, App e Controller

## 1 DAO

O DAO será responsável pela interação direta com o banco de dados, encapsulando as operações CRUD.

## 2 Model

O Model definirá as estruturas de dados utilizadas pela aplicação.

## 3 Service

O Service implementará a lógica de negócio, coordenando as interações entre as camadas de DAO e Controller.

## 4 Controller

O Controller será responsável por receber e processar as requisições do front-end, interagindo com o Service.

## 5 App

O App é basicamente o responsável por instanciar os services e controllers e chamar todos os métodos de setup.

# Especificação de sistemas inteligentes



## Fornecedor

O principal fornecedor de tecnologia de IA que será adotado no projeto é a Azure.



## API

A API que será utilizada é a Azure Face AI, responsável por processar imagens faciais.



## Serviço

O serviço utilizado será o reconhecimento facial para identificação e autenticação.



## AI Service



Aliraijee

