Conteúdo

[Projeto 1](#_Toc158111714)

[Título e objetivo 1](#_Toc158111715)

[Autor, data, contexto do trabalho 1](#_Toc158111716)

[Descrição do problema 1](#_Toc158111717)

[Ferramentas e versões 1](#_Toc158111718)

[Análise de dados 1](#_Toc158111719)

[Descrição de processos 2](#_Toc158111720)

[Dicionário 2](#_Toc158111721)

[String connection, instâncias, bases de dados 2](#_Toc158111722)

[Lista de tabelas, campos e tipos 2](#_Toc158111723)

[Namespaces 3](#_Toc158111724)

[Classes e métodos 3](#_Toc158111725)

[Outras informações 3](#_Toc158111726)

# Projeto

## Título e objetivo

Gestão Escolar – Implementação de um Sistema Integrado de Gestão para Otimização e Eficiência na Administração Educacional

## Autor, data, contexto do trabalho

Autor: Camila Santos, Data Início: 03/04/2024, Data Final: 18/04/2024.

O sistema visa integrar e automatizar processos acadêmicos e administrativos, melhorando a comunicação entre a instituição de ensino, seus funcionários e seus alunos, este projeto busca promover a interatividade e a participação ativa dos alunos, através de uma plataforma. O objetivo é implementar um sistema que seja crítico para a captura eficaz e o uso inteligente de dados e informações sobre alunos, funcionários e notas. Este projeto é parte de um esforço maior para implementar um novo Sistema de Gestão Educacional, que substituirá um sistema legado em seu fim de vida, para melhorar o sistema operacional da escola.

## Descrição do problema

Temos uma escola primaria que precisa de um sistema para gerir professores, turmas, alunos e notas. Este sistema permitira a cada utilizador de acordo com o seu nível de acesso uma consulta, adição, remoção e edição.

1. O Diretor com todos os privilégios pode gerir professores, turmas, alunos e notas;
2. os professores poderão consultar turmas e alunos e adicionar, remover e editar notas;
3. e os alunos tem acesso somente a lista de alunos da sua turma e a consulta somente das suas notas.

Estes são objetivos fundamentais para a gestão eficaz de uma escola, permitindo que os funcionários, alunos e pais acedam e atualizem informações relevantes de forma segura e eficiente.

## Ferramentas e versões

Visual Studio

Tipo de projeto: Windows Form App

Versão: .NET Framework 4.7.2

# Análise de dados

A base de dados "bd\_projeto\_form\_camila" é estruturada em torno de várias entidades, incluindo login, funcionario, turma, aluno, nota e disciplina. Cada entidade é representada por uma tabela que contém colunas específicas para armazenar informações relevantes. Por exemplo, a tabela aluno armazena detalhes sobre os alunos, como nome, apelido, idade, e-mail, e a tabela nota regista as notas atribuídas aos alunos em diferentes disciplinas.

Uma imagem com texto, captura de ecrã, diagrama, Tipo de letra

Descrição gerada automaticamenteO projeto é concebido para atender a uma ampla gama de necessidades dentro da instituição educacional, incluindo o gerenciamento de logins para acesso ao sistema, a gestão de informações sobre funcionários e turmas, e a avaliação do desempenho dos alunos através de notas e disciplinas. A estruturação da base de dados permite uma fácil integração e análise de dados, facilitando a identificação de tendências, a avaliação do desempenho dos alunos e a otimização dos recursos educacionais.

# Descrição de processos

A descrição de processos para o sistema de gestão escolar é detalhada da seguinte forma:

1. Autenticação de Usuários:
   * O processo começa com a autenticação do usuário, que é realizada através da tabela login. Os utilizadores inserem seu nome de usuário e senha, que são verificados contra os registos na tabela login. Se as credenciais correspondem, o utilizador é autenticado e tem acesso ao sistema.
2. Gestão de Funcionários:
   * Após a autenticação, os funcionários podem aceder suas informações pessoais e detalhes de trabalho, que são armazenados na tabela funcionário. Isso inclui o nome do funcionário e o ID de login associado, permitindo a personalização da experiência do utilizador.
3. Gestão de Turmas:
   * Os diretores podem gerenciar as turmas através da tabela turma, que contém informações sobre cada turma, incluindo a designação da turma e o ID do professor responsável. Isso permite a criação, atualização e exclusão de turmas, bem como a atribuição de professores a turmas específicas.
4. Gestão de Alunos:
   * Os alunos são gerenciados através da tabela aluno, que armazena informações como nome, apelido, idade, e-mail, e o ID da turma a qual o aluno pertence. Isso permite a adição, atualização e exclusão de alunos, bem como a visualização de detalhes do aluno.
5. Gestão de Notas:
   * As notas dos alunos são registadas na tabela nota, que inclui o ID do aluno, o ID da disciplina, a data de atribuição da nota e a nota em si. Isso permite a entrada, atualização e visualização de notas para cada aluno em cada disciplina.
6. Gestão de Disciplinas:
   * As disciplinas são gerenciadas através da tabela disciplina, que contém informações sobre cada disciplina, incluindo a designação da disciplina. Isso permite a criação, atualização e exclusão de disciplinas, bem como a atribuição de disciplinas a turmas.

# Dicionário

## String connection, instâncias, bases de dados

* String connection:

string connectionString = "Server=62.28.39.135,62444;Database=bd\_projeto\_form\_camila;User Id=EFACAMILA;Password=123.Abc##;";

* Instância: “lagostin”
* Base de dados: “bd\_projeto\_form\_camila”

## Lista de tabelas, campos e tipos

1. Entidades:
   1. login:
      1. id\_login(int)
      2. user(varchar)
      3. password(varchar)
      4. role(varchar)
   2. funcionario:
      1. id\_funcionario(int)
      2. nome(varchar)
      3. login\_id(int)
   3. turma:
      1. id\_turma(int)
      2. designação(varchar)
      3. professor\_id(int)
   4. aluno:
      1. id\_aluno(int)
      2. nome(varchar)
      3. apelido(varchar)
      4. idade(int)
      5. email(varchar)
      6. turma\_id(int)
      7. login\_id(int)
   5. nota:
      1. id\_nota(int)
      2. aluno\_id(int)
      3. disciplina\_id(int)
      4. data\_atribuicao(datetime)
      5. nota(decimal)
   6. disciplina:
      1. id\_disciplina(int)
      2. designação(varchar)

## Namespaces

projeto\_form\_camila

## Classes e métodos

1. Classes:
   1. Login (Form) => Formulario para receber o utilizador e a senha
      1. FormLogin\_Load() => Inicializa o formulário
      2. verificarLogin() => Verifica qual a role do utilizador
      3. direcionarUtilizador() => Redireciona para o formulário especifico de acordo com a role
   2. Funcionario
      1. CRUD => Conjunto de métodos que gere os funcionários como: adicionar, editar remover e detalhar
      2. listarFuncionarios() => Busca todos os funcionário e dispõe em uma lista
   3. Professor (Form) => Formulário para o professor gerir os alunos pertencentes a sua turma
      1. FormProfessor\_Load() => Inicializa o formulário
      2. CRUD => Conjunto de métodos que gere as notas dos alunos como: adicionar, editar remover e detalhar
      3. preencherGridAlunos() => Busca todos os alunos de uma turma específica e dispõe em uma grid
   4. Aluno (Form) => Formulário de acesso dos alunos para consultar a lista de alunos da sua turma e suas próprias notas
      1. FormAluno\_Load()=> Inicializa o formulário
      2. listarAlunosPorTurma() => Busca os alunos de uma turma especifica e dispõe em uma lista
      3. preencherGridAlunos() => Busca as notas do aluno que efetuou o login e dispõe em uma grid
   5. Turma
      1. CRUD => Conjunto de métodos que gere as turmas como: adicionar, editar remover e detalhar
      2. listarTurmas() => Busca todas as tumas cadastradas e dispôs em uma lista/grid
   6. Disciplina
      1. CRUD => Conjunto de métodos que gere as disciplinas como: adicionar, editar remover e detalhar
      2. listarDisciplinas() => Busca todas as disciplinas cadastradas e dispõe em uma lista/grid
   7. Nota
      1. CRUD => Conjunto de métodos que gere as disciplinas como: adicionar, editar remover e detalhar
      2. listarNotasPorTurma() => Busca e lista as notas de uma turma especifica
      3. listarNotasPorAluno() => Busca e lista as notas de um aluno especifico
      4. listarNotasPorDisciplina() => Busca e lista as notas de uma determinada disciplina
   8. AppDbContext
   9. Diretor (Form) => Formúlario do Diretor que administra todas as entidades
      1. FormDiretor\_Load()=> Inicializa o formulário
      2. Este formulário acede a vários métodos das outras classes

## Outras informações

