Uma imagem com texto, Tipo de letra, captura de ecrã, Azul elétrico

Descrição gerada automaticamente

**Instituto Politécnico do Cávado e do Ave**

**Escola Superior de Tecnologia**

**Programação Orientada a Objetos**

**Licenciatura em Engenharia de Sistemas Informáticos**

**1º Trabalho Prático**

Gestão de Condomínios

Fábio Alexandre Gomes Fernandes– a22996

Pedro Lourenço Morais Rocha – a23009

dezembro de 2023

**Resumo**

**Índice**

[1. Introdução 7](#_Toc150970729)

[2. Classes 8](#_Toc150970730)

[2.1 Gestão 9](#_Toc150970731)

[2.1.1 Descrição 9](#_Toc150970732)

[2.1.2 Implementação 10](#_Toc150970733)

[2.2 Condomínio 10](#_Toc150970734)

[2.2.1 Descrição 10](#_Toc150970735)

[2.2.2 Implementação 11](#_Toc150970736)

[2.3 Imóvel 11](#_Toc150970737)

[2.3.1 Descrição 11](#_Toc150970738)

[2.3.2 Implementação 12](#_Toc150970739)

[2.4 Proprietário 12](#_Toc150970740)

[2.4.1 Descrição 12](#_Toc150970741)

[2.4.2 Implementação 13](#_Toc150970742)

[2.5 Despesa 13](#_Toc150970743)

[2.5.1 Descrição 13](#_Toc150970744)

[2.5.2 Implementação 14](#_Toc150970745)

[2.6 Reunião 14](#_Toc150970746)

[2.6.1 Descrição 14](#_Toc150970747)

[2.6.2 Implementação 15](#_Toc150970748)

[2.7 Documento 15](#_Toc150970749)

[2.7.1 Descrição 15](#_Toc150970750)

[2.7.2 Implementação 16](#_Toc150970751)

[3. Conclusão 17](#_Toc150970752)

[4. Referências 18](#_Toc150970753)

**Índice de Ilustrações**

[**Figura 1:** Diagrama de Classes 8](file:///C:\Users\ferna\OneDrive\Ambiente%20de%20Trabalho\poo.docx#_Toc150970754)

**Índice de Acrónimos**

# Introdução

No âmbito da cadeira de Programação Orientada a Objetos, lecionada por Luís Ferreira no curso de Engenharia de Sistemas Informáticos, apresentamos este relatório sobre o desenvolvimento de um programa de gestão de condomínios. A gestão eficiente de condomínios é um desafio que envolve a coordenação de diversos elementos, desde a administração de propriedades e despesas até a organização de reuniões e documentação. Com o objetivo de simplificar e otimizar esse processo, desenvolvemos uma solução intuitiva que se alinha aos princípios da Programação Orientada a Objetos. Este sistema não é apenas uma ferramenta tecnológica, mas sim uma aplicação prática dos conceitos aprendidos na disciplina. Ao longo deste relatório, exploraremos as principais funcionalidades e estruturas do programa, destacando como cada componente está interligado para criar uma ferramenta eficaz de gestão condomínios. Desde a representação de diferentes tipos de utilizadores até a hierarquia de classes e relações entre elas, examinaremos de perto as decisões de design que fundamentam a arquitetura do software.

Este trabalho busca não apenas demonstrar a aplicação prática dos conceitos de Programação Orientada a Objetos, mas também oferecer uma solução valiosa para um contexto real, contribuindo para uma compreensão mais aprofundada dos desafios e oportunidades associados à implementação desses princípios.

# Classes

Uma imagem com texto, captura de ecrã, diagrama, Paralelo

Descrição gerada automaticamenteO sistema de gestão de condomínios apresenta uma estrutura robusta e abrangente, centrada em classes que contém informações vitais para a administração eficiente e colaborativa de condomínios. A classe central, “Gestão”, assume um papel importante ao coordenar e executar operações necessárias para o bom funcionamento do condomínio. Esta classe serve como um ponto central para a execução de processos importantes e para a manutenção do fluxo de informações entre as diversas classes do sistema. Pela figura 1, percebe-se que existe mais classes além da classe “Gestão”, sendo estas: “Condomínio”, “Imóvel”, “Proprietário”, “Despesa”, “Reuniao” e “Documento”.

**Figura 1:** Diagrama de Classes

Com base no diagrama de classes (figura 1), é possível observar as heranças entre as classes. Assim, no contexto das heranças, temos:

* **“Gestão” → “Condomínio”:** A classe “Condomínio” herda características e operações da classe “Gestão”. Isso significa que a gestão do condomínio é realizada através da classe “Gestão”, que possui métodos e atributos essenciais,

e a classe “Condomínio” estende esses recursos para lidar com as operações específicas de condomínio.

* **“Gestão” → “Imóvel”, “Proprietário”, “Despesa”, “Reunião” e “Documento”:** Cada uma delas herda diretamente da classe “Gestão”. Isso significa que cada classe no sistema (Imóvel, Proprietário, Despesa, Reunião e Documento) tem acesso direto aos métodos e atributos da classe “Gestão”, facilitando a coordenação centralizada dessas entidades.

## Gestão

### Descrição

A classe “Gestão” desempenha um papel central no sistema de gestão de condomínios, sendo esta responsável por coordenar e executar operações necessárias para que haja um bom funcionamento do condomínio. Ela atua como o ponto central para a execução de processos importantes e para a manutenção do fluxo de informações entre as diversas entidades do sistema.

**Atributos**

1. **CondominioAtual:** Referência ao objeto “Condominio” que está sendo gerenciado pelo gestor;
2. **HistoricoOperacoes:** Registo das operações executadas pelo gestor ao longo do tempo.

**Métodos**

1. **RealizarPagamento():** Iniciar o processo de registo de pagamento para um proprietário específico;
2. **AgendarReuniao():** Agendar uma nova reunião no condomínio;
3. **AdicionarDocumento():** Adicionar um novo documento ao sistema;
4. **ExecutarOperacaoPersonalizada():** Permite executar operações personalizadas no contexto do condomínio.

### Implementação

…fase 2 do trabalho…

## Condomínio

### Descrição

A classe “Condomínio” representa a peça fundamental no sistema de gestão de condomínios, contendo todas as informações relevantes sobre um condomínio específico. Seu papel é essencial na organização e administração dos diversos aspetos que compõem a gestão de condomínios. Esta classe serve como uma classe central, concentrando dados importantes que são essenciais para o bom funcionamento e colaborativo do condomínio.

**Atributos**

1. **Nome:** Representa o nome do condomínio;
2. **Endereco:** Endereço físico do condomínio;
3. **Imoveis:** Contém uma lista de imóveis existentes no condomínio;
4. **Despesas:** Contém uma lista de despesas associadas ao condomínio (água, eletricidade, limpeza, manutenção…);
5. **Proprietarios:** Contém uma lista de proprietários que fazem parte do condomínio;
6. **Reunioes:** Contém informações sobre reuniões agendadas (data, hora, local e pauta);
7. **Documentos:** Contém documentos relevantes para o condomínio (atas de reunião, regulamentos internos, comunicados…).

**Métodos**

1. **AdicionarDespesa():** Permite adicionar uma nova despesa à lista de despesas do condomínio;
2. **AdicionarImovel():** Adicionar um novo imóvel à lista de imoveis do condomínio;
3. **AdicionarProprietario():** Adicionar um novo proprietário à lista de proprietários do condomínio;
4. **AgendarReuniao():** Agendar uma nova reunião para o condomínio;
5. **AdicionarDocumento():** Adicionar um novo documento à lista de documentos do condomínio.

### Implementação

…fase 2 do trabalho…

## Imóvel

### Descrição

A classe “Imóvel” é uma representação fundamental no sistema de gestão de condomínios, conter informações específicas sobre um imóvel dentro do condomínio. Esta classe é essencial para seguir as responsabilidades financeiras e associativas de cada imóvel.

**Atributos**

1. **IdImovel:** Identificação do imóvel dentro do condomínio através do número ou nome;
2. **Proprietario:** Identificação do proprietário do imóvel em questão;
3. **Despesas:** Lista das despesas associadas ao imóvel;
4. **Quotas:** Registo das quotas pagas.

**Métodos**

1. **AdicionarDespesa():** Adicionar uma nova despesa ao imóvel;
2. **PagarQuota():** Registar o pagamento de uma quota, atualizando o histórico de quotas pagas;
3. **ObterSaldo():** Calcular e retornar o saldo atual da unidade, tendo atenção às despesas e quotas pagas.

### Implementação

…fase 2 do trabalho…

## Proprietário

### Descrição

A classe “Proprietário” representa as pessoas que possuem imóveis do condomínio, sendo essenciais para a gestão participativa e colaborativa. Esta classe contém informações sobre os proprietários ou inquilinos dos imóveis, facilitando o acompanhamento de dados pessoais e contribuições para o condomínio.

**Atributos**

1. **Nome:** O nome do proprietário;
2. **Contato:** O número de telemóvel e email;
3. **Imovel:** Referência ao imóvel à qual o proprietário está associado;
4. **HistoricoPagamentos:** Lista das contribuições financeiras do proprietário ao longo do tempo;
5. **PresencasReunioes:** Lista das presenças do proprietário em reuniões do condomínio.

**Métodos**

1. **RealizarPagamento():** Registar o pagamento do proprietário, atualizando o histórico de pagamentos;
2. **VisualizarHistorico():** Mostrar todo o histórico de pagamentos realizados pelo proprietário;
3. **ParticiparReuniao():** Registar a presença do proprietário em uma reunião especifica.

### Implementação

…fase 2 do trabalho…

## Despesa

### Descrição

A classe “Despesa” é fundamental no sistema de gestão de condomínios, representando cada uma das obrigações financeiras que o condomínio deve suportar. Esta classe contém informações detalhadas sobre as despesas, ajudando assim na gestão e na distribuição justa dos custos entre os imóveis.

**Atributos**

1. **Tipo:** Indica o tipo de despesa (água, eletricidade, limpeza, manutenção…)
2. **Valor:** Indica o montante financeiro associado à despesa;
3. **DataVencimento:** Indica a data-limite para realizar o pagamento da despesa;
4. **EstadoPagamento:** Regista se a despesa foi paga ou está pendente;
5. **Imovel:** Referência ao imóvel à qual a despesa está associada.

**Métodos**

1. **AtualizarEstadoPagamento():** Atualizar o estado de pagamento da despesa;
2. **ObterValorTotal():** Mostrar o valor total da despesa;
3. **DetalhesDespesa():** Mostrar detalhadamente a despesa.

### Implementação

…fase 2 do trabalho…

## Reunião

### Descrição

A classe “Reunião” desempenha um papel central no sistema de gestão de condomínios, proporcionando um meio estruturado para discussões, decisões e comunicação entre os proprietários dos imoveis. Esta classe contém informações essenciais sobre as reuniões realizadas no âmbito do condomínio.

**Atributos**

1. **Data:** Indica a data em que a reunião esta agendada para ocorrer;
2. **Hora:** Indica a hora em que a reunião esta marcada para começar;
3. **Local:** Indica o local onde a reunião será realizada;
4. **Intervenientes:** Lista dos proprietários que vão participar na reunião;
5. **Ata:** Documento que regista as decisões e discussões ocorridas durante a reunião.

**Métodos**

1. **AgendarReuniao():** Agendar uma nova reunião com os detalhes fornecidos;
2. **AdicionarInterveniente():** Adicionar um proprietário à lista de intervenientes da reunião;
3. **RegistarAta():** Associar um documento do tipo ata à reunião, registando as decisões e discussões.

### Implementação

…fase 2 do trabalho…

## Documento

### Descrição

A classe “Documento” é uma classe crucial no sistema de gestão de condomínios, visto que este representa os registos formais e informações essenciais. Esta classe contém diversos tipos de documentos, como atas de reuniões, regulamentos internos, comunicados e outros, fornecendo uma estrutura padrão para o armazenamento e recuperação de informações importantes.

**Atributos**

1. **Nome:** O nome do documento;
2. **Tipo:** Indica o tipo de documento (ata, regulamento, comunicado…);
3. **DataCriacao:** A data em que o documento foi criado ou registado;
4. **Conteudo:** O conteúdo informativo do documento.

**Métodos**

1. **AdicionarDocumento():** Adicionar um novo documento ao sistema;
2. **RemoverDocumento():** Remover um documento existente com base no nome fornecido;
3. **AtualizarConteudo():** Atualizar o conteúdo de um documento específico.

### Implementação

…fase 2 do trabalho…

# Conclusão

# Referências