

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS CORNÉLIO PROCÓPIO
PROGRAMAÇÃO ORIENTADA A OBJETOS 2 – TURMA ES41

ÁLISON CHRISTIAN REBOUÇAS VIDAL DE CARVALHO
JOÃO PEDRO CORREIA LEITE MOREIRA
MATHEUS GUILHERME MADUREIRA

SMART CONDOS

CORNÉLIO PROCÓPIO
FEVEREIRO, 2025

Álison Christian Rebouças Vidal De Carvalho;

João Pedro Correia Leite Moreira; e

Matheus Guilherme Madureira.

SMART CONDOS

Projeto elaborado na disciplina de Programação Orientada a Objetos 2 do curso de Engenharia da Computação, do Campus Cornélio Procópio da Universidade Tecnológica Federal do Paraná.

Professora: Gisele Alves Santana

CORNÉLIO PROCÓPIO

FEVEREIRO, 2025

RESUMO

O Smart Condos é um sistema de gerenciamento de condomínios que visa facilitar a organização e a comunicação entre moradores, síndicos e funcionários. O projeto foi desenvolvido utilizando a linguagem Java, com interface e funcionalidades implementadas no NetBeans. O sistema permite centralização das informações, substituindo métodos convencionais como o livro de ocorrências. Embora não automatize os processos, ele melhora a eficiência administrativa e a comunicação entre os condôminos.

Palavras-chave: Gerenciamento de condomínios, Java e NetBeans.

SUMÁRIO

1	Introdução	3
2	Ferramentas e Tecnologias	3
3	Desenvolvimento	4
3.1	Levantamento dos Requisitos	4
3.2	Diagrama de Casos de Uso	5
3.3	Especificação dos Casos de Uso	6
3.4	Diagrama de Classes	7
3.5	Boas Práticas	8
3.6	Banco de Dados	8
3.7	Telas do sistema	9
3.8	Link do vídeo da apresentação e GitHub	9
4	Conclusões	10

1 INTRODUÇÃO

O gerenciamento de condomínios tem sido tradicionalmente realizado por meios manuais, como livros de ocorrências, avisos impressos e registros descentralizados. Essa abordagem pode gerar dificuldades na organização das informações e no acesso rápido aos dados necessários.

Nesse revés, o desenvolvimento de sistemas digitais para a gestão condominial tem ganhado espaço nos últimos anos, proporcionando maior controle e segurança na administração desses ambientes. Diferentes abordagens são utilizadas, desde sistemas automatizados até soluções que buscam apenas facilitar a interação entre os condôminos e a administração.

O Smart Condos se insere nesse contexto como uma solução para facilitar a administração de condomínios, permitindo, por exemplo, que moradores visualizem avisos, que o síndico gerencie informações e que os funcionários registrem encomendas. O sistema foi desenvolvido em Java, utilizando o NetBeans como IDE e o PostgreSQL para armazenamento de dados. O objetivo do projeto é substituir os meios convencionais de registro e comunicação, centralizando todas as informações em uma plataforma digital acessível.

2 FERRAMENTAS E TECNOLOGIAS

- Linguagem: Java
- Ambiente de Desenvolvimento: NetBeans IDE
- Controle de Versão: GitHub
- Banco de Dados: PostgreSQL
- Ferramenta CASE: Astah UML

3 DESENVOLVIMENTO

Esta seção apresenta todos os componentes essenciais desenvolvidos para a construção do sistema Smart Condos. São descritos os requisitos levantados, os diagramas utilizados para modelagem do sistema, os scripts implementados e demais aspectos técnicos relevantes. Para facilitar a compreensão, utilizamos diagramas ilustram a estrutura e o funcionamento do sistema.

3.1 Levantamento dos Requisitos

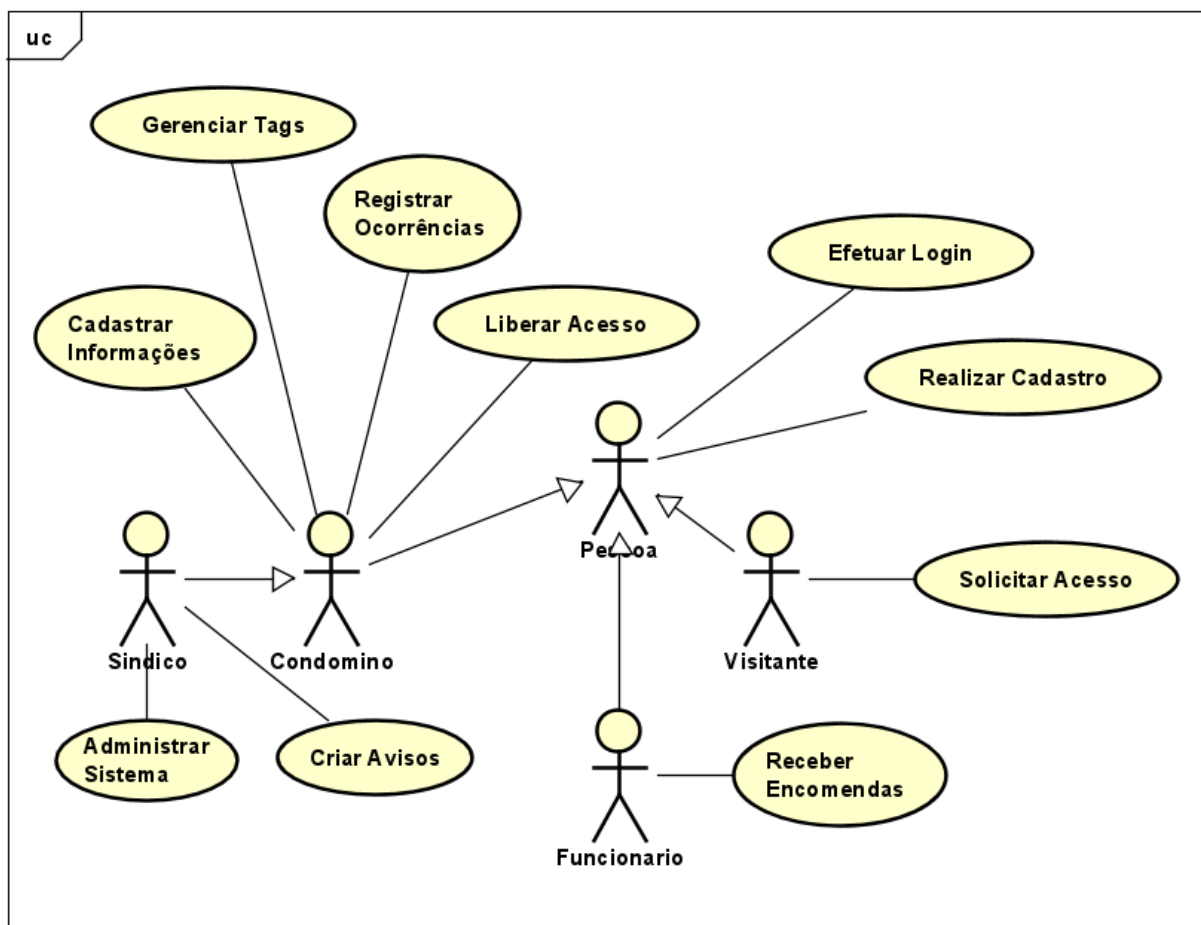
Funcionais

Identificador	Descrição	Prioridade
RF01	O usuário deve ser capaz de cadastrar seu CPF, nome, e-mail e condomínio.	Alta
RF02	O usuário deve ser capaz de fazer login através de seu e-mail e senha.	Alta
RF03	Cada condomínio terá um administrador (síndico), o qual terá privilégios especiais.	Alta
RF04	O síndico deve ser capaz de registrar funcionários do seu condomínio e atribuir funções a eles no aplicativo (como registro de manutenção e recebimento de encomendas).	Alta
RF05	O aplicativo terá uma sessão para registro de ocorrências pelos moradores.	Alta
RF06	O condômino poderá gerenciar as suas tags no aplicativo.	Média
RF07	O aplicativo terá uma sessão para aviso de recebimento de encomendas e de manutenção.	Alta
RF08	O condômino poderá fazer a liberação de visitantes através do aplicativo.	Alta
RF09	O aplicativo terá uma sessão de avisos feitos pelo síndico.	Alta
RF10	O aplicativo terá uma sessão para reservas (churrasqueira, salão de festas e afins).	Alta
RF11	O aplicativo terá uma sessão onde o condômino poderá inserir documentos e demais informações referentes ao seu imóvel.	Alta
RF12	O condômino poderá registrar seu automóvel no aplicativo (entrada facilitada ou não).	Média

Não Funcionais

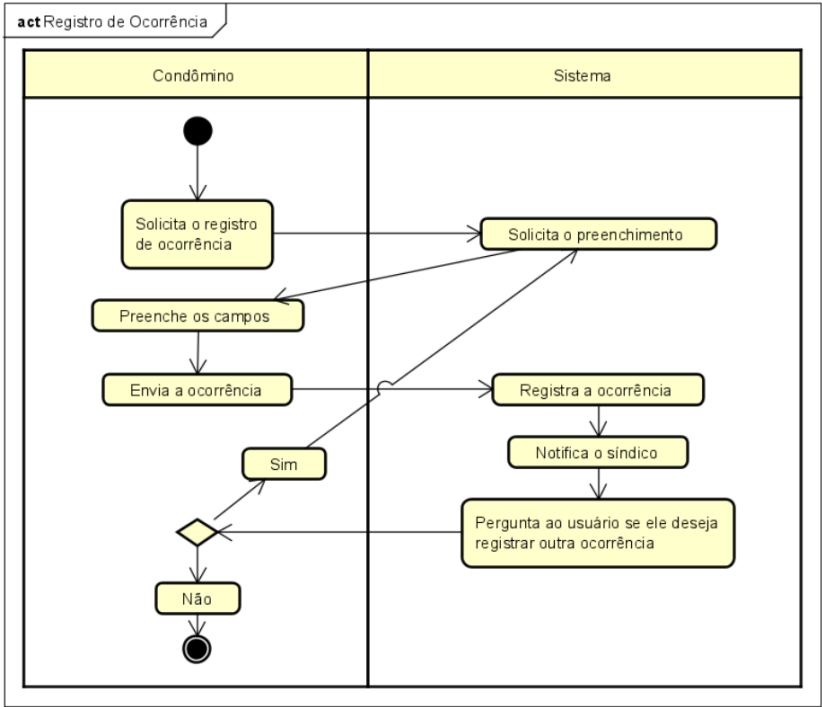
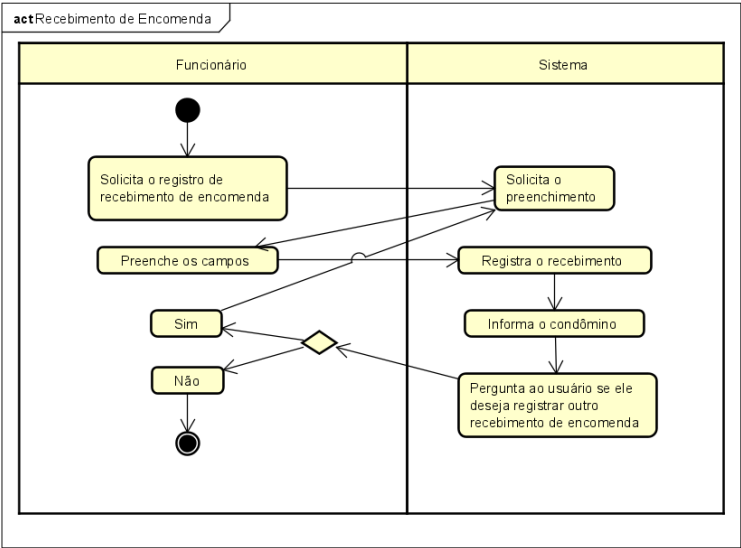
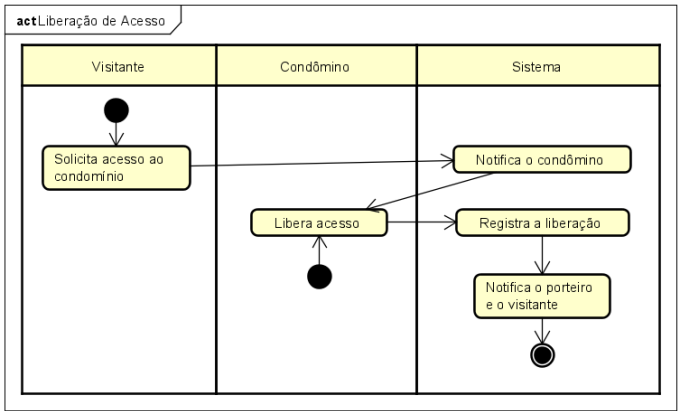
Identificador	Descrição	Tipo
RNF01	O sistema deve ter uma resposta das requisições de no máximo 400ms.	Desempenho
RNF02	O sistema deve armazenar dados dos usuários de maneira criptografada.	Segurança
RNF03	O usuário deve realizar login previamente para armazenamento de logs e, assim, acessar as rotas protegidas.	Rastreabilidade
RNF04	Os usuários comuns não podem acessar rotas de administrador.	Segurança
RNF05	O sistema deve estar disponível a todo momento para os usuários.	Disponibilidade

3.2 Diagrama de Casos de Uso



3.3 Especificação dos Casos de Uso

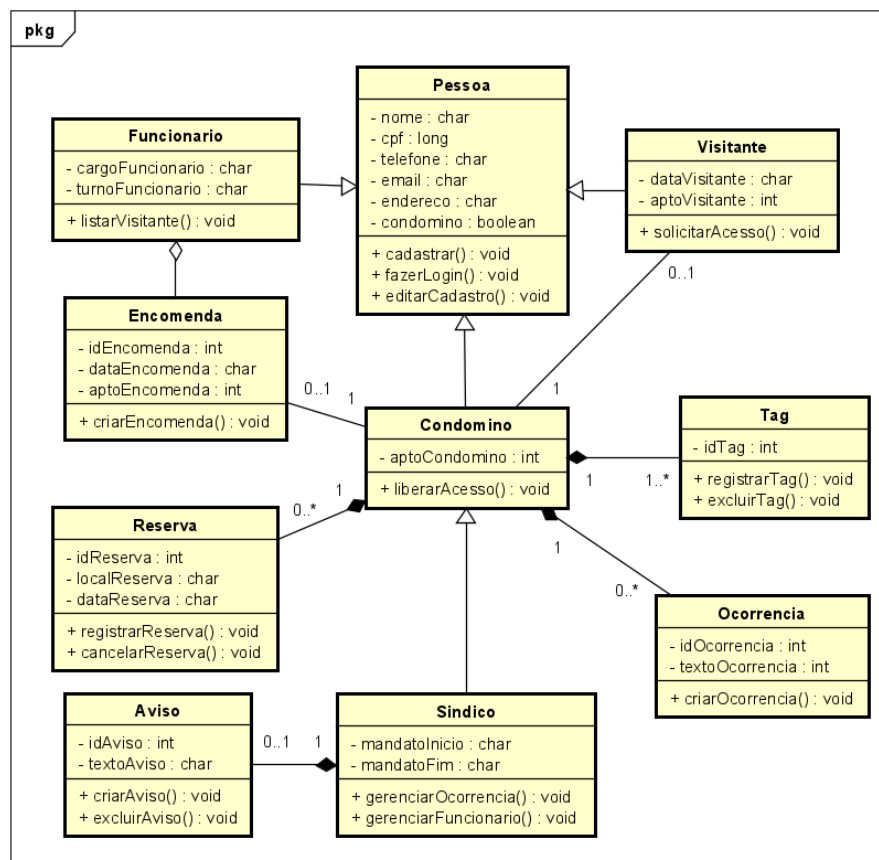
Fluxo Normal



Caminhos Alternativos

ID do caminho alternativo	Descrição
1	Caso alguma das informações inseridas estejam incompletas, o sistema alertará o usuário para que ele as preencha corretamente
2	Durante o cadastro, caso o CPF do usuário já esteja cadastrado, o sistema irá alertá-lo para recuperar seu cadastro.
3	Caso o acesso do visitante seja recusado pela terceira vez, ele terá seu cadastro eliminado e seu CPF bloqueado.

3.4 Diagrama de Classes



3.5 Boas Práticas

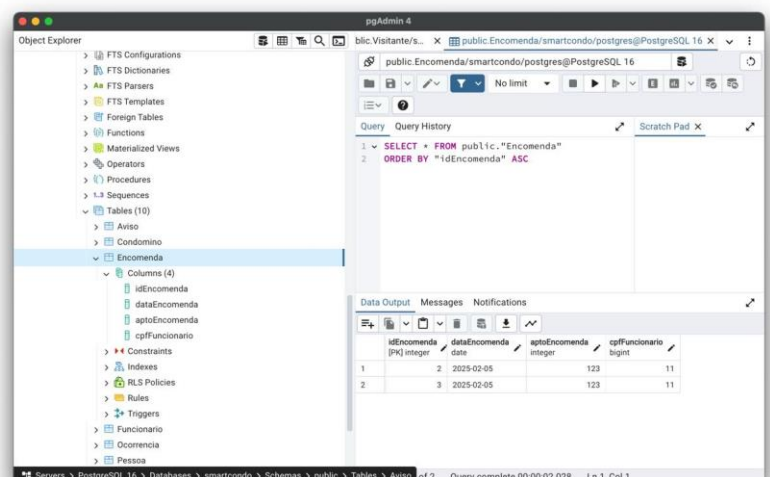
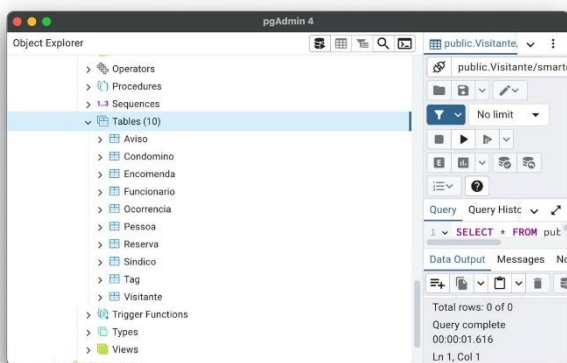
O projeto seguiu boas práticas de desenvolvimento, como o uso do **padrão arquitetural MVC** para organização da aplicação. O MVC (Model-View-Controller) separa a lógica do negócio da interface do usuário, tornando o sistema mais modular e manutenível.



Além disso, aplicamos o **princípio DIP** (Dependency Inversion Principle), que promove a inversão de dependência, garantindo que módulos de alto nível não dependam diretamente de módulos de baixo nível, aumentando a flexibilidade e reutilização do código.



3.6 Banco de Dados



3.7 Telas do sistema

The 'Tela do Síndico' interface includes a top navigation bar with 'Início' and 'Opções'. Below this is a 'Tela do Síndico' header with an 'Atualizar' button. A 'Principais avisos' section contains an 'Adicionar Aviso' button and a text area with the message 'Aviso bom dia a todos heelo'. At the bottom, there are six buttons arranged in a 3x2 grid: 'Reservas', 'Encomendas', 'Visitantes', 'Tags', 'Ocorrências', and 'Funcionários'.

The 'Tela do Funcionário' interface features a top navigation bar with 'Início' and 'Opções'. Below is a 'Tela do Funcionário' header with an 'Atualizar' button. An 'Encomendas' section includes an 'Adicionar Encomenda' button and a table with columns 'Id', 'Apartamento', and 'Data'. At the bottom, a 'Visitantes Liberados' section displays a table with columns 'Data', 'Nome', 'CPF Visitante', and 'Apartamento', showing a single entry for '2025-02-21' by 'Tiago Alcantra' with CPF '43428654387' in apartment '401'.

The 'CRIAR CONTA' form contains input fields for 'Nome', 'CPF', 'Telefone', 'Email', and 'Endereço'. It also includes a 'Condomínio' dropdown set to 'Sim' and an 'Apartamento' input field. At the bottom are 'Voltar' and 'Criar Conta' buttons.

The 'ADICIONAR RESERVA' form has a 'Local' dropdown set to 'Salão de Festas'. The 'Data' section includes a date picker for 'fevereiro' 2025 and a calendar grid. The calendar shows days from 05 to 09, with columns for 'dom.', 'seg.', 'ter.', 'qua.', 'qui.', 'sex.', and 'sáb.'. The 'Adicionar Reserva' button is at the bottom right.

3.8 Link do vídeo da apresentação e GitHub

Apresentação: shorturl.at/pe6C4 ou [clique aqui](#).

GitHub: github.com/jotasoftware/SmartCondos

4 CONCLUSÕES

O desenvolvimento do Smart Condos trouxe resultados satisfatórios dentro dos objetivos propostos. O sistema se mostrou funcional e eficaz na facilitação da comunicação e gestão condominial. A principal dificuldade encontrada foi a integração com o banco de dados, especialmente na recuperação e envio de informações. No entanto, essa barreira foi superada com ajustes na conexão e consultas SQL.

A experiência proporcionou um grande aprendizado na prática do desenvolvimento de sistemas, desde a implementação da interface até a estruturação do banco de dados. A utilização do NetBeans facilitou o processo, tornando a construção das telas intuitiva e eficiente. Como melhoria futura, sugere-se a ampliação do sistema com recursos de automação, como notificações automáticas e integração com dispositivos inteligentes, para aumentar ainda mais a eficiência e segurança do condomínio.

REFERÊNCIAS

HALLIDAY, D. RESNICK, R. Fundamentos de Física: Eletromagnetismo. 8ª Ed. Rio de Janeiro: Livros Técnicos e Científicos, 2009. Vol 3. 396 p.

TIPLER, P. A. Física. Rio de Janeiro: Editora Guanabara Dois S.A., 1978. vol. 2.

ULABY, F. T. Eletromagnetismo para Engenheiros. 1ª Ed. Porto Alegre: Bookman Companhia Editora Ltda, 2007. 382 p.

WILLIAMS, H. SMITH. A History of Science Volume II, Part VI: The Leyden Jar Discovered. BiblioBazaar, 2009.

HOUSTON, EDWIN J. Electricity in Every-day Life. [S.l.]: P. F. Collier & Son, 1905.

SEARS, Francis Weston; ZEMANSKY, Mark Waldo; YOUNG, Hugh D.; FREEDMAN, Roger A. Física III: Eletromagnetismo. 12. ed. São Paulo: Pearson Addison Wesley, 2009.