

**CENTRO UNIVERSITÁRIO OPET - UniOpet**

**ADILSON GOMES DA SILVA**

**MATHEUS LOURENÇO**

**THAIS FERNANDES XAVIER DE PAULA DA SILVEIRA**

**WALLACE DE LIMA CABRAL**

**PROJETO INTEGRADOR**

**ISY Condomínios**

**CURITIBA**

**2023**

**ADILSON GOMES DA SILVA**

**MATHEUS LOURENÇO**

**THAIS FERNANDES XAVIER DE PAULA DA SILVEIRA**

**WALLACE DE LIMA CABRAL**

## **PROJETO INTEGRADOR**

### **ISY Condomínios**

Projeto integrador, apresentado como requisito parcial para aprovação nas disciplinas de: Arquitetura de software, Fundamentos data science, Análise e projeto orientado a objetos e Engenharia de requisitos do Centro Universitário Opet - UniOpet.

**CURITIBA**

**2023**

## LISTA DE ILUSTRAÇÕES

### Figuras

FIGURA 1: DIAGRAMA ARQUITETURA ISY CONDOMÍNIOS.....	7
FIGURA 2: DIAGRAMA DE CASO DE USO - ISY CONDOMÍNIOS.....	12
FIGURA 3: DIAGRAMA DE SEQUÊNCIA - ISY CONDOMÍNIOS.....	14
FIGURA 4: DIAGRAMA DE CLASSE - ISY CONDOMÍNIOS. ....	15

### Quadros

QUADRO 1: REQUISITOS FUNCIONAIS PARA O 3º SEMESTRE.....	8
QUADRO 2: REQUISITOS FUNCIONAIS PARA PRÓXIMOS SEMESTRES.....	9
QUADRO 3: ESPECIFICAÇÃO UC5 - DIAGRAMA DE CASO DE USO .....	12

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>4</b>
<b>2. SOFTWARE.....</b>	<b>5</b>
<b>3. ARQUITETURA .....</b>	<b>5</b>
3.1. ARQUITETURA MONOLÍTICA .....	5
3.2. ISY CONDOMÍNIOS .....	6
<b>4. REQUISITOS .....</b>	<b>8</b>
4.1. FUNCIONAIS .....	8
4.1.1. Para os próximos semestres .....	9
4.2. NÃO FUNCIONAIS .....	9
4.2.1. Para os próximos semestres .....	11
<b>5. DIAGRAMAS .....</b>	<b>12</b>
5.1. DIAGRAMA DE CASO DE USO .....	12
5.1.1. Especificação do diagrama de caso de uso .....	12
5.2. DIAGRAMA DE SEQUÊNCIA .....	14
5.3. DIAGRAMA DE CLASSE .....	15
<b>6. SOLUÇÃO TÉCNICA .....</b>	<b>15</b>
6.1. PROGRAMAS UTILIZADOS .....	15
<b>7. LINKS DE HOSPEDAGEM E GITHUB.....</b>	<b>16</b>
<b>8. CONSIDERAÇÕES FINAIS.....</b>	<b>17</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>18</b>
<b>GLOSSÁRIO .....</b>	<b>19</b>

## 1. INTRODUÇÃO

Este trabalho tem como objetivo apresentar a terceira etapa do projeto em desenvolvimento denominado ISY Condomínios.

A ISY Condomínios é um software desenvolvido especificamente para atender às demandas e desafios da gestão de condomínios, oferecendo uma solução integrada e eficiente para otimizar processos, promover a transparência e melhorar a comunicação entre os moradores e a administração.

Ao longo deste trabalho, iremos abordar em detalhes as suas principais funcionalidades, arquitetura e requisitos. Abordaremos também eventuais desafios e limitações do software.

Ressaltamos que este projeto se encontra em fase de desenvolvimento e aperfeiçoamento contínuo. Nosso objetivo é criar uma solução robusta e adaptável, capaz de atender às necessidades específicas de diferentes tipos de condomínios.

Esperamos que este trabalho desperte interesse e contribua para a compreensão dos benefícios e potencialidades do projeto ISY Condomínios.

## 2. SOFTWARE

A ISY Condomínios é um sistema desenvolvido para uso na administração moderna e consolidada de condomínios em plataforma única.

## 3. ARQUITETURA

Nesta etapa, foi analisado e escolhido a arquitetura que melhor se adequa ao software.

Sistemas de software são compostos por muitas estruturas e nenhuma estrutura única pode ser chamada de arquitetura (BASS; CLEMENTS; KAZMAN, 2013, tradução própria)<sup>1</sup>

Conforme citado acima, os softwares são compostos de muitas estruturas e estas estruturas podem ou não ser elementos arquitetônicos neste software. Ainda de acordo com os autores anteriores, *"A arquitetura de software de um sistema é o conjunto de estruturas necessárias para raciocinar sobre o sistema, que compreende os elementos de software, as relações entre eles e as propriedades de ambos"* (BASS; CLEMENTS; KAZMAN, 2013, tradução própria)<sup>2</sup>.

Ao se planejar um sistema, um dos pontos importantes é definir qual será o modelo arquitetônico que será usado, considerando suas vantagens e desvantagens, de acordo com o tipo de aplicação que será criada. Entre os modelos arquitetônicos existentes, para o nosso projeto, iremos utilizar a arquitetura monolítica.

### 3.1. ARQUITETURA MONOLÍTICA

Richards e Ford (2020) no livro *Fundamentals of Software Architecture: An Engineering Approach* propõe que a arquitetura monolítica é um dos estilos

---

<sup>1</sup> "Software systems are composed of many structures, and no single structure holds claim to being the architecture"

<sup>2</sup> "The software architecture of a system is the set of structures needed to reason about the system, which comprise software elements, relations among them, and properties of both"

arquitetônicos mais comuns devido a sua familiaridade, facilidade e baixo custo, além de ser a forma natural de desenvolver um sistema.

Ainda segundo Richards e Ford (2020) este estilo é composto de camadas horizontais, em que cada camada tem uma função específica na aplicação. A quantidade de camadas não está especificada, pois depende de cada aplicação desenvolvida, porém, as mais comuns de serem encontradas são as camadas de apresentação onde o usuário terá a interação com a aplicação, a camada de negócios onde está a regra do negócio, a camada de persistência que é destinada para a comunicação com o banco de dados e a camada de banco de dados.

Como toda arquitetura, o formato monolítico apresenta vantagens e desvantagens em seu uso. Conforme Pablo Santos (2021) no blog da empresa AR Phoenix Tecnologias, as vantagens e desvantagens são:

### **Vantagens da Arquitetura Monolítica**

- Estruturação simplificada;
- Poucos recursos tecnológicos;
- Um único profissional técnico;
- Baixa integração.

### **Desvantagens da Arquitetura Monolítica**

- Manutenção;
- Difícil de testar;
- Difícil de escalar;
- Não há flexibilidade em linguagens de programação.

## **3.2. ISY CONDOMÍNIOS**

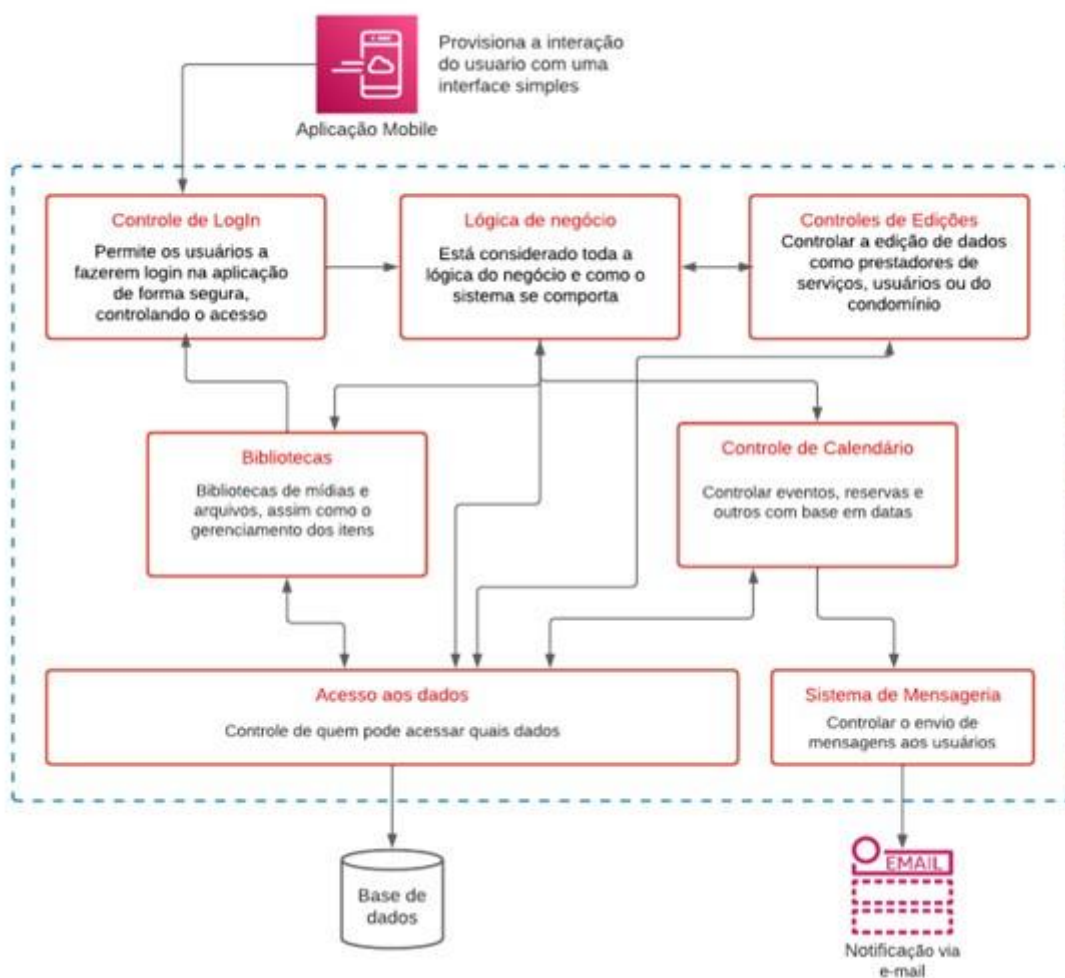
O projeto ISY condomínios será desenvolvido em uma arquitetura monolítica, em que todo o código, lógica de negócio e interface do usuário estarão

combinados em uma única camada sem uma separação clara entre as funcionalidades.

Foi escolhido este modelo arquitetônico pois é útil em projetos menores, onde a simplicidade é relevante, porém, com a consciência de que a escalabilidade deste projeto, pode ser um desafio considerando o modelo escolhido. Outros motivos que levaram a escolha deste modelo são: a facilidade de desenvolvimento, já que permite o trabalho mais rápido e com menos complexidade e a facilidade de implantação já que todo o sistema estará em um único executável, na nuvem.

Na imagem abaixo é possível observar o diagrama do modelo arquitetônico escolhido, aplicado ao projeto.

Figura 1: Diagrama arquitetura Isy Condomínios.





## 4. REQUISITOS

### 4.1. FUNCIONAIS

Quadro 1: Requisitos funcionais para o 3º semestre

<b>Identificador: RF01</b>	<b>Requisito:</b> Manter dados do condomínio.
O síndico(a) / administrador(a) irá registrar o nome e endereço completo do condomínio para inserção dos dados no sistema. Ele(a) poderá realizar a alteração destes dados caso seja necessário.	
<b>Identificador: RF02</b>	<b>Requisito:</b> Manter dados dos usuários.
O síndico(a) / administrador(a) e condôminos irão registrar os seus dados pessoais completos, dados de login, selecionar o seu condomínio e tipo de usuário. Caso seja síndico(a) / administrador(a) irá adicionar o documento comprovante do seu cargo. Ambos os usuários poderão realizar alterações destes dados caso seja necessário.	
<b>Identificador: RF03</b>	<b>Requisito:</b> Permitir o acesso dos usuários ao sistema.
O usuário irá digitar seu e-mail (login) e senha cadastrados, para acessar o sistema e todas as suas funções.	
<b>Identificador: RF04</b>	<b>Requisito:</b> Manter os dados dos prestadores de serviços.
O usuário poderá indicar um prestador de serviço, registrando no sistema dentro da opção “Eu indico”, o nome, tipo de serviço prestado, telefone completo (ddd + número), tipo de telefone, avaliação e comentário. Ficará registrado o nome e a data em que o usuário realizou esse registro.	
<b>Identificador: RF05</b>	<b>Requisito:</b> Armazenar e gerenciar documentos.
O usuário síndico(a) / administradora realizará o upload de arquivos relacionados ao condomínio (Atas, Regime interno, normas e regras do condomínio e outros), esses arquivos estarão disponíveis para os condôminos no aplicativo para visualização.	

#### 4.1.1. Para os próximos semestres

Quadro 2: Requisitos funcionais para próximos semestres

Identificador: RF06	Requisito: Manter os dados do mural de sugestões/reclamações.
O usuário irá selecionar o assunto e digitar uma descrição. Podendo alterar e excluir posteriormente. Ao salvar, o sistema irá mostrar aos condôminos em um mural.	
Identificador: RF07	Requisito: Manter os dados de recados/comunicados.
O usuário síndico(a)/ administrador(a) irá selecionar o assunto, a urgência do recado/comunicado e digitar uma descrição. Podendo alterar ou excluir posteriormente. Ao salvar, o sistema irá mostrar aos condôminos em um mural.	
Identificador: RF08	Requisito: Manter um sistema de mensageria para enviar notificações (e-mail, sms).
Envio de e-mails e sms de forma automática, sobre novas normas, regras, comunicados, e demais informações registradas no aplicativo.	
Identificador: RF09	Requisito: Manter calendário atualizado anualmente e permitir que o usuário realize reservas.
O usuário irá selecionar uma data, horário e tipo de local para lazer que gostaria de reservar. Ao salvar, o sistema irá registrar seu nome e unidade para realizar o agendamento. Ficando visível para os outros usuários.	

#### 4.2. NÃO FUNCIONAIS

- **Segurança:** O sistema deve garantir a proteção dos dados do condomínio, implementando mecanismos de autenticação, controle de acesso e criptografia para evitar acesso não autorizado ou divulgação indevida das informações.
- **Segurança (Usuário):** O sistema deve garantir a segurança das credenciais de login dos usuários, protegendo-as contra acesso não autorizado e ataques

de força bruta. Isso inclui o armazenamento seguro das senhas. Além disso, o sistema deve fornecer recursos de gerenciamento de contas, como a possibilidade de redefinir senhas esquecidas.

- **Conformidade (Privacidade de dados):** O sistema deve garantir a privacidade e a proteção dos dados pessoais dos usuários, cumprindo regulamentos de privacidade de dados, como o Regulamento Geral de Proteção de Dados (GDPR). Isso inclui a adoção de práticas de segurança adequadas, como criptografia e controle de acesso, além do consentimento explícito dos usuários para coleta e processamento de seus dados.
- **Conformidade (Integridade dos dados):** O sistema deve garantir a integridade dos dados dos fornecedores e prestadores de serviços, evitando a perda, corrupção ou modificação indevida das informações. Isso pode ser alcançado por meio de mecanismos de controle de transações e backup regular dos dados, além de restrições adequadas de acesso e permissões de edição somente para usuários autorizados.
- **Escalabilidade:** O sistema deve ser capaz de lidar com um grande volume de documentos, suportando o armazenamento e gerenciamento eficiente de uma quantidade significativa de arquivos. Isso inclui a capacidade de escalonamento horizontal ou vertical do sistema, garantindo que ele possa lidar com o aumento da carga de trabalho sem comprometer o desempenho ou a disponibilidade.
- **Disponibilidade:** Inicialmente, não será possível a intercomunicação entre condôminos através do sistema, não haverá chat. Podendo ser implementado futuramente conforme demanda e necessidade.

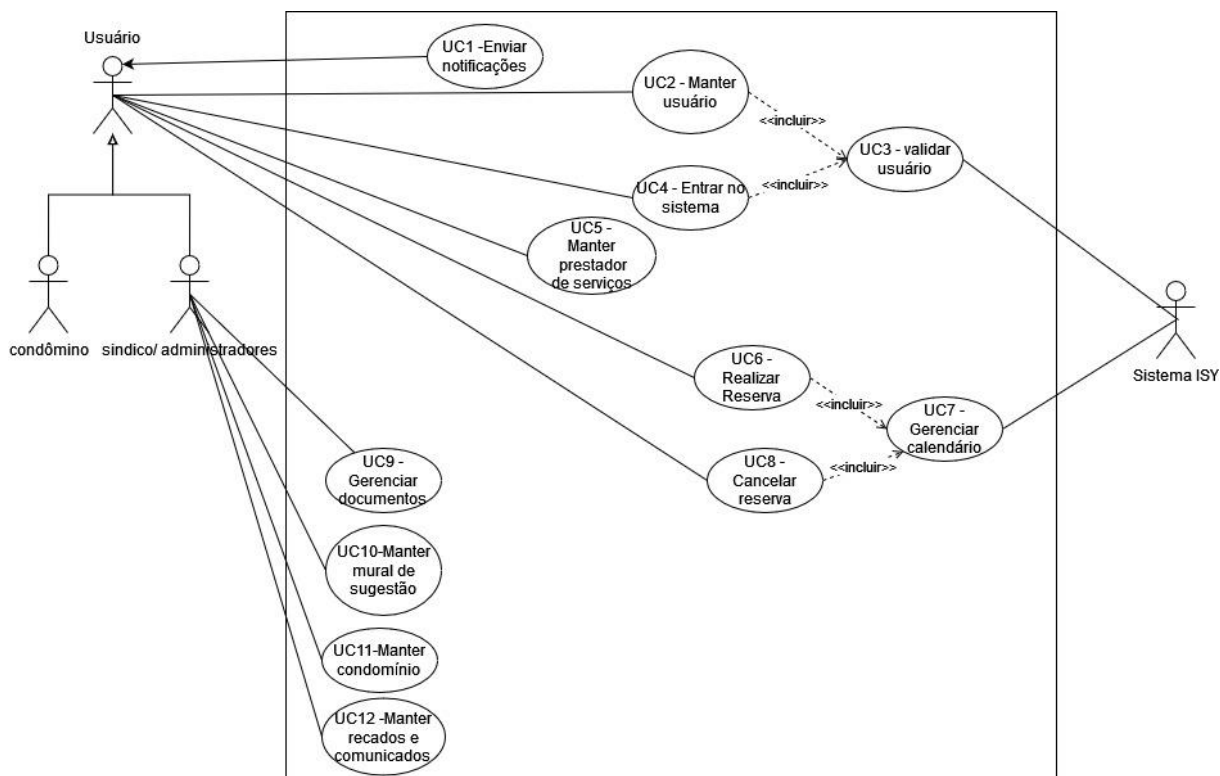
#### 4.2.1. Para os próximos semestres

- **Acessibilidade:** O sistema deve ser acessível e permitir que os usuários com diferentes necessidades possam interagir e utilizar o mural de sugestões/reclamações de forma efetiva. Isso pode incluir recursos como suporte, opções de tamanho de fonte ajustável e boa legibilidade de cores e contraste.
- **Atuação:** O sistema deve ser capaz de enviar notificações em tempo real para os usuários relevantes sempre que houver inclusão, alteração ou exclusão de recados/comunicados. Isso garante que os usuários sejam informados imediatamente sobre as atualizações e mudanças nos recados/comunicados, possibilitando uma comunicação eficiente e oportuna.
- **Escalabilidade:** O sistema de mensageria deve ser capaz de lidar com um grande volume de notificações, suportando o envio eficiente de mensagens para grande número de destinatários simultaneamente. Isso inclui a capacidade de escalonamento horizontal ou vertical do sistema de mensageria, garantindo que ele possa lidar com o aumento da carga de trabalho sem comprometer o desempenho ou a disponibilidade.
- **Confiabilidade:** O sistema de calendário e reservas deve ser altamente confiável, garantindo que as informações do calendário estejam sempre atualizadas e que as reservas dos usuários sejam processadas corretamente. Isso envolve a implementação de mecanismos de sincronização e atualização adequados para manter o calendário atualizado, bem como a realização de validações e verificações para garantir que as reservas sejam registradas e confirmadas com precisão.

## 5. DIAGRAMAS

### 5.1. DIAGRAMA DE CASO DE USO

Figura 2: Diagrama de caso de uso - Isy Condomínios.



#### 5.1.1. Especificação do diagrama de caso de uso

Quadro 3: Especificação UC5 - diagrama de caso de uso

#### ESPECIFICAÇÃO DO CASO DE USO UC5 – MANTER PRESTADOR DE SERVIÇOS

<b>IDENTIFICADOR ÚNICO DE CASO DE USO</b>	UC5 – Manter prestador de serviços
<b>DESCRIÇÃO</b>	Permite ao usuário o cadastro/gerenciamento de prestadores de serviço
<b>ATOR PRINCIPAL</b>	Usuário
<b>PRÉ-CONDIÇÕES</b>	O caso de uso UC4 – Entrar no sistema ter sido executado com sucesso e o usuário estar logado

<b>GATILHO</b>	Usuário logado selecionar Prestador de serviço
----------------	--

#### FLUXO PRINCIPAL

1. USUÁRIO ACESSA A TELA DE PRESTADOR DE SERVIÇO.
2. SISTEMA IRÁ REALIZAR UMA CONSULTA PARA EXIBIR OS PRESTADORES DE SERVIÇOS CADASTRADOS.
3. O USUÁRIO INDICA A OPERAÇÃO DESEJADA: INCLUSÃO, EXCLUSÃO, ALTERAÇÃO.

#### FLUXO ALTERNATIVO (2): CADASTRO INEXISTENTE

CASO NÃO EXISTA PRESTADORES CADASTRADOS NO SISTEMA, O SISTEMA DEVE EXIBIR UMA MENSAGEM INFORMANDO QUE NÃO EXISTE CADASTROS.

#### FLUXO ALTERNATIVO (3): INCLUSÃO

- A. O USUARIO SELECIONA A OPÇÃO DE ADICIONAR PRESTADOR DE SERVIÇO.
- B. SISTEMA MOSTRA UM FORMULARIO EM BRANCO PARA QUE OS DADOS DO PRESTADOR (NOME FORNECEDOR, TELEFONE, SERVIÇO PRESTADO, AVALIAÇÃO, COMENTARIO) SEJAM INCLUIDOS.
- C. O SISTEMA VERIFICA A VALIDADE DOS DADOS. SE OS DADOS FOREM VALIDOS, INCLUI O NOVO PRESTADOR. CASO CONTRÁRIO, O SISTEMA REPORTA O FATO.

#### FLUXO ALTERNATIVO (3): REMOÇÃO

- A. O USUARIO SELECIONA A OPÇÃO DE EXCLUIR NO PRESTADOR DESEJADO.
- B. APENAS O USUARIO QUE CADASTROU E O SÍNDICO/ADMINISTRADOR PODERAM EXCLUIR O PRESTADOR.

#### FLUXO ALTERNATIVO (3): ALTERAÇÃO

- A. O USUÁRIO SELECIONA A OPÇÃO DE ALTERAÇÃO.
- B. O SISTEMA MOSTRA UM FORMULARIO EM BRANCO PARA INSERÇÃO DE DADOS.
- C. O USUÁRIO MODIFICA OS CAMPOS DESEJADOS.
- D. O SISTEMA VERIFICA A VALIDADE DOS DADOS. SE OS DADOS FOREM VALIDOS, ALTERA O PRESTADOR. CASO CONTRÁRIO, O SISTEMA REPORTA O FATO.

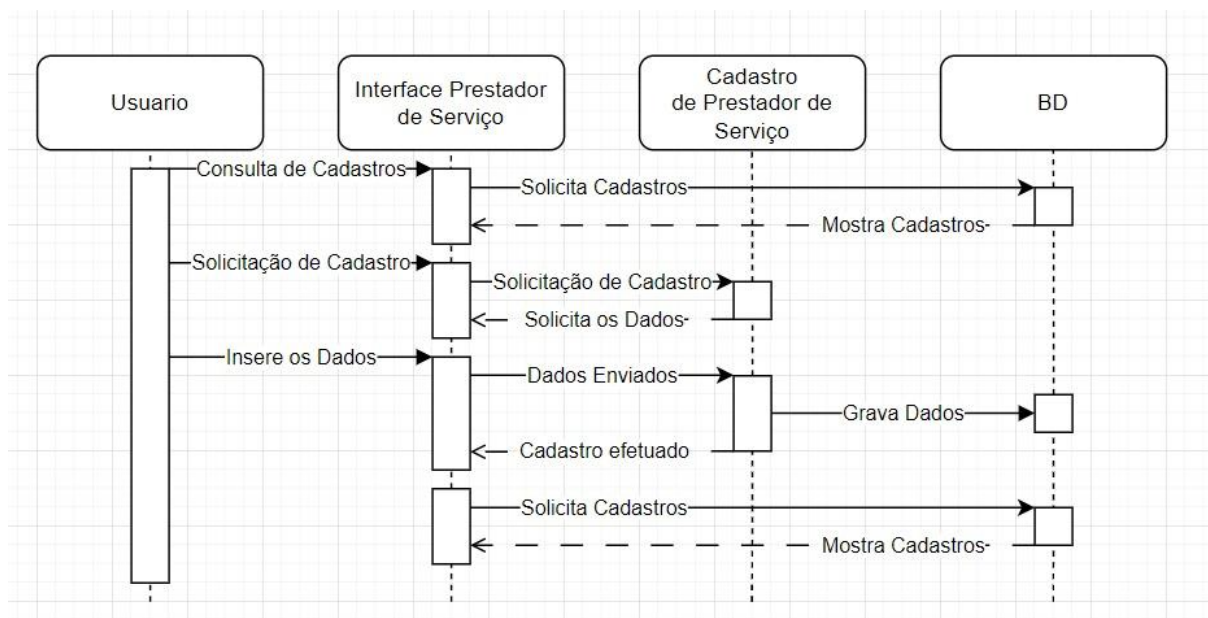
## 5.2. DIAGRAMA DE SEQUÊNCIA

O diagrama de sequência é uma das várias ferramentas de modelagem usadas para visualizar e descrever a interação entre objetos em um sistema.

O diagrama de sequência é especialmente útil para descrever o comportamento dinâmico de um sistema, enfatizando as interações entre os objetos e os eventos que desencadeiam essas interações. Ele é frequentemente usado para modelar casos de uso, cenários e fluxos de trabalho dentro de um sistema.

Neste diagrama foi utilizado a especificação UC5 - Manter prestador de serviços, retirado do diagrama de caso de uso mostrado na sessão anterior.

Figura 3: Diagrama de sequência - Isy Condomínios.

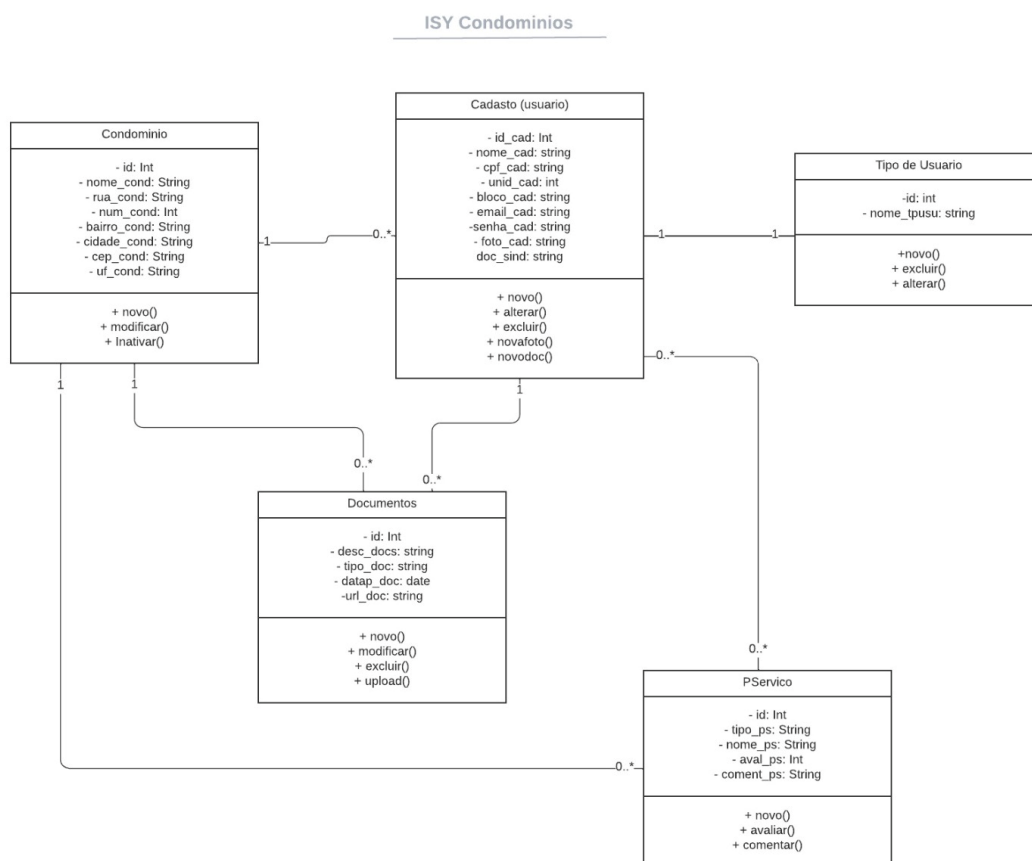


### 5.3. DIAGRAMA DE CLASSE

Este diagrama oferece uma visão geral da estrutura e dos relacionamentos das classes do sistema, permitindo uma melhor compreensão do design e da organização dos seus componentes.

No diagrama abaixo foi utilizado a especificação UC5 - Manter prestador de serviços, podemos visualizar nele as classes, atributos, métodos e os relacionamentos desta funcionalidade.

Figura 4: Diagrama de classe - Isy Condomínios.



## 6. SOLUÇÃO TÉCNICA

### 6.1. PROGRAMAS UTILIZADOS

Para a realização da etapa 3 do projeto foram utilizados os seguintes programas na parte de desenvolvimento WEB:



- Software Visual Studio Code (HTML, CSS, BOOTSTRAP, PHP);
- XAMPP;

Para banco de dados:

- Maria Db (phpMyAdmin)

Diagramas UML:

- Draw.io
- Lucichart

## **7. LINKS DE HOSPEDAGEM E GITHUB**

GITHUB: <https://github.com/agsilva86/ISY>

BANCO DE DADOS: <https://github.com/agsilva86/ISY>

HOSPEDAGEM: <https://grupo-isy.000webhostapp.com/login.php>

## 8. CONSIDERAÇÕES FINAIS

Desenvolver um projeto em grupo de um software de condomínios é um desafio que requer um esforço conjunto, coordenação, habilidades técnicas e interpessoais.

Como o grupo já vinha de duas etapas trabalhadas, nesta terceira etapa a maioria dos integrantes já se conhecia, e, portanto, ficou mais fácil entender como cada um poderia contribuir com o projeto.

Além de conhecimentos técnicos, os membros da equipe também desenvolveram habilidades interpessoais importantes. A comunicação efetiva foi fundamental para o sucesso do projeto, permitindo que as ideias fossem compartilhadas e compreendidas por todos. A colaboração entre os integrantes, a disposição para ouvir e considerar diferentes pontos de vista, também a capacidade de trabalhar em equipe foram essenciais para a resolução de conflitos e para o alcance dos objetivos comuns.

Um dos desafios apontados pelo grupo nesta etapa, foi a dificuldade de encontrar uma plataforma de hospedagem gratuita e compatível com o sistema feito com php, o que exigiu um pouco mais de atenção dos membros da equipe.

Durante todo o processo, reconhecemos que o trabalho em grupo proporcionou uma oportunidade significativa para aprendizado e desenvolvimento pessoal. Os membros da equipe tiveram a chance de aprimorar suas habilidades de liderança ao assumir responsabilidades, guiar e motivar os colegas de equipe. A capacidade de lidar com prazos apertados e gerenciar efetivamente o tempo também foi aprimorada, contribuindo para a eficiência e produtividade do grupo.

## REFERÊNCIAS BIBLIOGRÁFICAS

BASS, Len; CLEMENTS, Paul; KAZMAN, Rick. **Software Architecture in Practice**. 3. ed. Upper Saddle River, NJ: Pearson Education, Inc, 2013. ISBN 978-0-321-81573-6.

NORMAS ABNT.org. **Normas ABNT 2022**: Confira um guia com todas as regras. 18 maio 2022. Disponível em: <https://www.normasabnt.org/normas-abnt-2022/>. Acesso em: 10 out. 2022.

RICHARDS, Mark; FORD, Neal. **Fundamentals of Software Architecture: An Engineering Approach**. 1. ed. Sebastopol, CA: O'Reilly Media Inc, 2020. ISBN 978-1-492-04345-4.

SANTOS, Pablo. **Quais as diferenças entre Arquitetura Monolítica e Microserviços, suas vantagens e desvantagens**. Toda Matéria, 2021. Disponível em: <https://arphoenix.com.br/quais-as-diferencas-entre-arquitetura-monolitica-e-microservicos-suas-vantagens-e-desvantagens/>. Acesso em: 29 abr. 2023.

VAZQUEZ , Carlos Eduardo; SIMÕES, Guilherme Siqueira. **Engenharia de Requisitos: Software orientado ao negócio**. 1 . ed. [S. l.]: Editora Brasport, 2016. ISBN 9788574527963. Disponível em: <https://plataforma.bvirtual.com.br/Acervo/Publicacao/160193#>. Acesso em: 5 maio 2023.

## GLOSSÁRIO

**Software** - é um serviço computacional utilizado para realizar ações nos sistemas de computadores.

**Nuvem** - é a disponibilidade sob demanda dos recursos de computação como serviços na Internet.

**Login** - é um método que serve para permitir o acesso a uma plataforma.

**Dados** - conjunto de informações.

**Backup** - cópia de segurança.

**Chat**- funcionalidade de troca de mensagens.

**Visual studio code** - ferramenta para o desenvolvimento de software.

**Html** - linguagem de programação utilizado para organizar o conteúdo visual da aplicação.

**Css** - linguagem de programação utilizado para decorar a parte visual da aplicação.

**Bootstrap** - um framework front-end que fornece estruturas de CSS para a criação de sites e aplicações responsivas de forma rápida e simples.

**Xampp** - pacote de servidores.

**PhpMyAdmin** -uma ferramenta de software livre que se destina a lidar com a administração de um servidor de banco de dados MySQL ou MariaDB.

**Maria Db** - base de dados.

**Draw.io** - é uma ferramenta online para criação de diagramas e fluxogramas colaborativos.

**Lucichart** - é uma ferramenta online para criação de diagramas e fluxogramas colaborativos.

**Github**- repositório de dados compartilhado.

**Hospedagem** - é um espaço online reservado para seu site na internet.