Sumário

[1 INTRODUÇÃO 3](#_Toc163740761)

[1.1 Objetivo 3](#_Toc163740762)

[1.2 Escopo 3](#_Toc163740763)

[1.3 Definições, Siglas e Abreviações 3](#_Toc163740764)

[1.4 Referências 3](#_Toc163740765)

[1.5 Visão Geral 3](#_Toc163740766)

[2 DESCRIÇÃO GERAL DO PRODUTO 4](#_Toc163740767)

[2.1 Perspectiva do Produto 4](#_Toc163740768)

[2.1.1 INTERFACES DE SISTEMA 4](#_Toc163740769)

[2.1.2 INTERFACES DE USUÁRIO 4](#_Toc163740770)

[2.1.3 INTERFACES DE HARDWARE 4](#_Toc163740771)

[2.1.4 INTERFACES DE SOFTWARE 4](#_Toc163740772)

[2.1.5 INTERFACES DE COMUNICAÇÃO. 5](#_Toc163740773)

[2.1.6 LIMITES DE MEMÓRIA 5](#_Toc163740774)

[2.1.7 OPERAÇÕES 5](#_Toc163740775)

[2.1.8 EXIGÊNCIAS DE ADAPTAÇÃO AO LOCAL 5](#_Toc163740776)

[2.2 Funções do produto 6](#_Toc163740777)

[2.3 Características do utilizador 6](#_Toc163740778)

[2.4 Restrições 6](#_Toc163740779)

[2.5 Assunções e dependências 6](#_Toc163740780)

[2.6 Divisão e atribuição das exigências 7](#_Toc163740781)

[3 Exigências específicas 7](#_Toc163740782)

[3.1 Interfaces externas 7](#_Toc163740783)

[3.2 Requisitos Funcionais 7](#_Toc163740784)

# INTRODUÇÃO

## Objetivo

A finalidade deste documento é fornecer uma base clara e abrangente para o desenvolvimento do sistema, garantindo que todas as partes interessadas tenham uma compreensão comum dos requisitos.

## Escopo

O produto de *software* obtidos no fim do processo de desenvolvimento será conhecido como ResidencialSync e é projetado para facilitar a administração de condomínios residenciais. O sistema permite o agendamento de áreas de lazer, cálculo das taxas condominiais, emissão de boletos de cobrança das taxas condominiais, agendamento de visitas com geração de QrCode pro visitante, e geração de relatórios áreas de lazer, moradores e suas unidades residenciais. O objetivo principal é otimizar a gestão do condomínio e oferecer uma experiência satisfatória aos moradores.

## Definições, Siglas e Abreviações

API - (Application Programming Interface)

E - (Evidente)

## Referências

## Visão Geral

O Capítulo 2 apresenta uma visão panorâmica do produto, expondo de forma simplificada sua perspectiva, incluindo as interfaces, funcionalidades, características dos usuários, bem como as restrições, suposições e dependências do produto.

No Capítulo 3, os requisitos expostos no Capítulo 2 são detalhados, fornecendo informações precisas que serão fundamentais durante as fases de projeto, teste e manutenção do software. Este capítulo inclui a especificação detalhada dos casos de uso do produto, garantindo uma compreensão abrangente de suas funcionalidades e interações.

# DESCRIÇÃO GERAL DO PRODUTO

## **Perspectiva do Produto**

Nesta seção o produto é colocado em perspectiva em relação a outros produtos relacionados e descrevemos sua operação dentro de várias restrições. Esta subseção é crucial para compreender como o software se integra em um sistema maior e como suas funcionalidades se relacionam com o ambiente ao seu redor. Além disso, são destacadas as interfaces críticas que o software deve suportar.

### Interfaces de Sistema

Como este software será utilizado em um ambiente independente, não há necessidade de interfaces com outros sistemas externos.

### Interfaces de Usuário

Software deve ter uma estrutura de navegação lógica e intuitiva, com menus bem organizados. Um funcionário treinado por 30 minutos deve conseguir realizar qualquer operação do software em menos de 5 minutos.

### Interfaces de Hardware

Não há interfaces de hardware relevantes para este software, pois as funcionalidades são executadas exclusivamente através de interações de software e não requerem dispositivos de hardware específicos.

### Interfaces de Software

#### Apache PDFBox

* Nome: Apache PDFBox.
* Mnemônica: PDFBox.
* Número de versão: 3.0.2
* Fonte: <https://pdfbox.apache.org/>

O PDFBox é uma biblioteca de código aberto em Java que permite a criação, manipulação e extração de conteúdo de arquivos PDF. No contexto do nosso software, ele é usado para criar documentos PDF que contêm informações sobre as reservas de áreas de lazer realizadas pelos moradores do condomínio. Esses documentos são então disponibilizados para impressão como comprovantes da reserva.

#### Caelum Stella

* Nome: Caelum Stella.
* Mnemônica: Stella.
* Número de versão: 2.1.2
* Fonte: [Home · caelum/caelum-stella Wiki (github.com)](https://github.com/caelum/caelum-stella/wiki)

O Stella é um projeto que visa suprir as necessidades do programador Java nas suas necessidades do dia a dia de trabalhar com o domínio brasileiros. Validadores para CEP, CPF, CNPJ, inscrição estadual, gerador de boleto e a integração desses com a JPA, JSF, bean validation e outros. No contexto do ResidencialSync, será usado para a geração de boletos das taxas condominiais. Documentação do módulo de geração de boletos: [Stella boleto · caelum/caelum-stella Wiki (github.com)](https://github.com/caelum/caelum-stella/wiki/Stella-boleto)

### Interfaces de Comunicação.

Não há interfaces de comunicação relevantes para este sistema, pois não há necessidade de interação com outros sistemas ou dispositivos externos.

### Limites de Memória

Não há requisitos específicos de memória para este sistema, pois não há exigências relacionadas ao uso de memória que afetem o seu funcionamento.

### Operações

Todas as operações do sistema são iniciadas e realizadas pelo funcionário que usa o software do condomínio. O software opera de forma interativa durante todo o tempo de uso, pois requer a interação direta dos usuários para realizar suas funcionalidades.

### Exigências de adaptação ao local

Não há exigências específicas de adaptação ao local para este software.

## Funções do produto

O sistema de gerenciamento de condomínios oferece diversas funcionalidades essenciais para a administração eficiente do condomínio. Isso inclui o agendamento de áreas de lazer disponíveis, emissão de comprovantes de reserva para os moradores, agendamento de visita, cálculo e gestão das taxas condominiais, além de gerar os boletos das taxas, para que possam ser enviados aos moradores e proprietários. Essas características garantem um suporte abrangente para todas as necessidades administrativas do condomínio.

## Características do utilizador

Os funcionários do condomínio que irão ter acessão ao software podem ter uma variedade de formações educacionais, desde ensino médio até formação técnica ou superior. No entanto, é seguro assumir que eles têm pelo menos um nível básico de educação e habilidades de leitura e escrita. Embora alguns funcionários possam ter mais experiência com tecnologia do que outros, é razoável supor que eles têm alguma familiaridade com o uso de computadores e software básico.

## Restrições

Com exceção das API´s descritas na seção 2.1.4, o software não possui outras restrições.

## Assunções e dependências

Este software depende das seguintes APIs externas para fornecer funcionalidades específicas:

1. **API de Geração de Boletos**: O sistema utiliza uma API de terceiros para gerar boletos de cobrança para os moradores do condomínio. A disponibilidade e funcionalidade contínua desta API são críticas para o processo de cobrança e gestão financeira do condomínio.
2. **API de Geração de PDF**: O software utiliza uma API externa para gerar documentos PDF dos comprovantes de reservas de áreas de lazer.

É importante observar que, caso essas APIs se tornem indisponíveis, descontinuadas ou depreciadas, será necessário buscar alternativas viáveis para substituí-las. Qualquer interrupção ou mudança nas APIs mencionadas acima pode afetar o funcionamento e a integridade das funcionalidades do software.

## Divisão e atribuição das exigências

Não foram identificadas exigências que possam ser adiadas para versões futuras do sistema. Todas as exigências listadas foram consideradas essenciais para a versão inicial do software.

# Exigências específicas

## Interfaces externas

O Apache PDFBox é uma interface de software que será utilizada pra gerar os comprovantes da reserva em PDF. Os dados de entrada serão o nome do morador, o local reservado e a data/hora da reserva. Já a saída será um PDF contendo esses dados.

Já o Caelum Stella é uma API que contém um módulo de geração de boletos, que será usado pra gerar os boletos contendo as taxas condominiais. Os dados de entrada são alguns dados do morador como Nome, CPF, Endereço da residência. E mais algumas informações, como a data de vencimento do boleto, juros que serão aplicados, valor da multa de atraso. E por fim, dados de quem irá receber o valor pago, como o Nome, CPF e o nome do banco.

## Requisitos Funcionais

R0. O sistema deve permitir o agendamento de áreas de lazer disponíveis, sendo a reserva podendo ser durante um período de até 5 horas. (E)

R1. O sistema deve impedir a sobreposição de reservas em uma mesma área de lazer para o horário. (E)

R3. O sistema deve verificar se a pessoa que está requisitando a reserva é mesmo um morador ou proprietário de terreno, através de seu CPF, validando também se a propriedade associada a reserva, é realmente a residência, ou terreno que o reservante diz.

R4. O sistema deve fornecer relatório das áreas disponíveis para agendamento em uma data/hora especificada. (E)

R5. Após a reserva de uma área de lazer ser feita por um funcionário com os dados sendo informados pelo morador ou algum proprietário de terreno, o sistema deve produzir um comprovante em PDF, a propriedade associada a reserva, o local reservado e a data/hora da reserva, e esse comprovante deve ser baixado na máquina que roda o sistema.

R6. O sistema deve produzir relatórios contendo todas as informações (Nome, Sexo, CPF, RG, Data de Nascimento) dos habitantes de uma determinada unidade residencial. (E)

R7. O sistema deve permitir a emissão de boletos de cobrança para taxas condominiais para todas as propriedades do condomínio, após o funcionário inserir juros e multa de atraso, taxa base e conta de água do mês atual. O boleto deve ser emitido no nome do proprietário da unidade residencial ou do terreno, e o beneficiário do boleto deve ser o condomínio. (E)

R8. Após a emissão deve ser feito o download de todos os boletos na máquina que o sistema roda.

R11. O sistema deve calcular juros e multas para boletos em atraso, conforme as regras estabelecidas pelo condomínio. (E)

R12. O sistema deve calcular o valor da taxa condominial com base no rateio de água dividindo o valor total da conta de água pelo número de unidades residenciais, sendo o valor total da conta de água informada pelo funcionário somada com uma taxa base definida pelo condomínio e também informada pelo funcionário, além da data de vencimento, e por fim somar o resultado da multiplicação da taxa fixa de R$ 10,00 referente ao uso das áreas de lazer do condomínio, pela quantidade de agendamentos feitas pelos moradores da respectiva unidade residencial durante os últimos 30 dias. Exemplo: ((Conta de Água/Quantidade de Residências) + Taxa Base) + (10,00 \* qtdReservasResidencia) (E)

R13. O cálculo pro rateio de água não deve incluir os terrenos, apenas as unidades residenciais, sendo assim, para os terrenos, é apenas somada a taxa base com o resultado da multiplicação da taxa fixa de R$ 10,00 referente ao uso das áreas de lazer do condomínio, pela quantidade de agendamentos feito pelo mesmo proprietário do respectivo terreno durante os últimos 30 dias. Exemplo: Taxa Base + (10,00 \* qtdReservasTerreno)

R17. O sistema deve permitir o agendamento de visitas para uma unidade residencial, gerando um QR code único para entrada e saída do visitante. O QR code gerado deve ser carregado na máquina em que roda o sistema, para ser enviado ao visitante. Para o agendamento o morador deve informar o dia da visita, CPF do visitante, número do celular do visitante e placa do veículo do visitante. (E)

R18. O QRcode gerado deve usar alguns dados pra ser gerado, como dados da visita, e a data que expira, possuindo 1 dia de validade, sendo assim a data de expiração é 1 dia após a data em que a visita vai acontecer.

R20. O sistema deve possibilitar a geração de relatórios contendo todas as informações de um morador a partir da placa de algum de seus veículos. (E)

R21. Durante a inicialização do sistema, o sistema deve carregar os dados que serão usados durante a utilização do ResidencialSync, inclusive dados do Condomínio onde o sistema vai ser instalado, já que os dados serão usados na geração dos boletos, dados como: Nome, Agência, Dígito Agência, Código de Beneficiário, Dígito Código Beneficiário, Número Convênio, Número Carteira, Logradouro, CEP, Cidade, Bairro, UF, CNPJ e o Banco.

## Exigências de desempenho

O software não possui requisitos específicos de desempenho. No entanto, espera-se que o sistema opere de forma eficiente e responsiva para atender às necessidades dos usuários.

## Exigências lógicas da base de dados

O software deve conter um a base de dados relacional, para armazenar os dados de forma organizada.