



**UNIVERSIDADE ESTATUAL DE MARINGÁ**

<b>FERNANDO YOCHIMI NICI</b>	<b>RA 68540</b>
<b>LEONARDO RAMPAZZO</b>	<b>RA 68066</b>
<b>LUCIO MASSAO G. ITIKAWA</b>	<b>RA 65135</b>

# **SISTEMA DE GERENCIAMENTO DE CONDOMÍNIO**

Trabalho apresentado para obter nota na disciplina de Projeto de Sistema de Software, sob orientação da professora Juliana, para o curso de Informática na Universidade Estadual de Maringá.

**Maringá, 31 de agosto de 2015**

## SUMÁRIO

<b>1. DESCRIÇÃO GERAL DO SISTEMA.....</b>	<b>3</b>
<b>2. REQUISITOS.....</b>	<b>4</b>
2.1. Requisitos funcionais .....	4
2.2. Requisitos não funcionais .....	17
<b>3. CASOS DE USO .....</b>	<b>19</b>
<b>4. GLOSSÁRIO DE TERMOS .....</b>	<b>36</b>
4.1. Quanto à viabilidade técnica .....	36
4.2. Avaliação dos Riscos .....	37
4.3. Modelo objeto de negócio .....	38
4.4. Estilos Arquiteturais .....	39
<b>5. Framework.....</b>	<b>43</b>
5.1. Grails.....	43
5.2. Hibernate.....	43
5.3. Diagrama de classes do projeto .....	44
<b>REFERÊNCIAS.....</b>	<b>46</b>

## **1. DESCRIÇÃO GERAL DO SISTEMA**

O sistema consiste em um software para gestão de condomínios, onde rotinas referentes a cobranças, solicitações (obras, comprar e serviços) e pagamentos serão controladas. O sistema é administrado pelo síndico ou funcionários da área administrativa, e utilizado por todos os condôminos a partir de um portal web.

Dentre os dados que o sistema controla, destacam-se, por exemplo: aluguel de áreas comuns (salão de festas, áreas de lazer), despesas habituais (contas de água, luz, administração), e controle de um caixa do síndico.

Por parte do síndico, que é o principal administrador do sistema, é de sua responsabilidade a gestão financeira e a aprovação ou não de solicitações dos moradores. Mensalmente, o sistema gera um boleto bancário para cada condômino, para que seja efetuado o pagamento de todas as taxas e eventuais cobranças extras, como multas, serviços, taxas de aluguéis etc. E cabe ao condômino acessar sua interface no sistema e fazer o download deste boleto.

Através do sistema o condômino também pode solicitar serviços, fazer reclamações e como já dito, gerar seu boleto de cobrança. As solicitações feitas pelo condômino ficam sujeitas à aprovação ou não do síndico, e o condômino pode acompanhar o status de sua solicitação a partir do próprio sistema.

## 2. REQUISITOS

### 2.1. Requisitos funcionais

#### [RF01] Manutenção de usuário – Fernando

Descrição: Criar uma opção onde será possível fazer o cadastro de usuários e suas devidas senhas para o acesso ao sistema. Também envolve a recuperação da senha.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, funcionário</i>					
<b>Requisitos associados:</b>						

#### [RF02] Manutenção do Condômino - Fernando

Descrição: Cadastro os dados básicos dos condôminos como nome, RG, CPF.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, funcionário</i>					
<b>Requisitos associados:</b>						

#### [RF03] Manutenção Apartamento - Fernando

Descrição: Cadastro os dados básicos dos apartamentos como numero do apartamento, numero da garagem, nome do proprietário.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, funcionário</i>					
<b>Requisitos associados:</b>						

**[RF04] Manutenção Funcionários - Fernando**

Descrição: Cadastros dados básicos dos funcionários como nome do funcionário, função, salário, RG.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, funcionário</i>					
<b>Requisitos associados:</b>						

**[RF05] Manutenção de Produtos - Fernando**

Descrição: Cadastro os dados básicos dos produtos que foram comprados como nome do produto, preço, tipo do produto, data de compra.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, funcionário</i>					
<b>Requisitos associados:</b>						

**[RF06] Criar Ocorrência - Fernando**

Descrição: Cria uma ocorrência, que pode ser algum relato de algum lugar do condomínio que esteja danificada, ou de uma manutenção agendada. Onde essa ocorrência é um aviso destinado ao síndico, essa aplicação estará no modulo WEB para visualização.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário, Condômino.</i>					
<b>Requisitos associados:</b>						

**[RF07] Criar Solicitação - Fernando**

Descrição: Criar uma solicitação, onde solicitação é um pedido realizado por qualquer usuário. Agendamento de área de lazer, reuniões, pedido de mudança. Após isso, a solicitação será encaminhada ao síndico para análise. Depois de a solicitação estar fechada, o usuário não poderá mais excluir ou altera-la. Esta aplicação estará também em modulo WEB para a criação.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário, Usuário</i>					
<b>Requisitos associados:</b>						

**[RF08] Consulta Ocorrência e Solicitações - Fernando**

Descrição: Tela onde se tem a opção de verificar as ocorrências e solicitações que foram abertas ou que estão pendentes.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário, Usuário</i>					
<b>Requisitos associados:</b>	<i>RF06; RF07;</i>					

**[RF09] Contas a receber - Fernando**

Descrição: Criar uma opção para lançamento de contas a receber manualmente, onde será especificado referente à que a conta se trata, e terá a opção de emitir boleto caso seja necessário, caso não seja a conta deverá ficar registrada no sistema em aberto ate que seja feito a baixa ou recebimento desta mesma.

<b>Prioridade:</b>	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>						

#### [RF10] Cadastrar Feriados e DRS - Leonardo

Descrição: Dentro do controle ponto haverá opção onde será possível marcar os feriados, folga e DSR, "descanso semanal remunerado".

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF04; RF10;</i>					

#### [RF11] Controlar Ponto - Leonardo

Descrição: Os dias que não estiveram marcados como feriado, folga ou DSR, se não houver registro de ponto, devera ser contabilizado uma falta.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF04; RF10;</i>					

#### [RF12] Justificar Falta - Leonardo

Descrição: Criar uma opção para registro de Justificativa de falta, "atestado" ou simples justificativa mesmo.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF04;RF10;RF12;</i>					

### **[RF13] Cadastro da área de lazer - Leonardo**

Descrição: Cadastro das áreas de lazer do condomínio, tendo elas um nome, bloco e um número de identificação.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF02;</i>					

### **[RF14] Agendar área de lazer - Leonardo**

Descrição: Criar opção onde através de uma solicitação possa ser feito um agendamento de uma área de lazer, por exemplo o salão de festa, após isso será encaminhado ao síndico para aprovação.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF14</i>					

### **[RF15] Criar e Consultar Calendário de Agendamento - Leonardo**

Descrição: Após a aprovação do síndico para a solicitação de agendamento o dia e o horário feito no pedido será arquivada em um calendário virtual que estará no site ou no sistema off-line para consulta.



<b>Prioridade:</b>		Essencial	<b>X</b>	Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF14; RF15;</i>					

#### **[RF16] Criar Formulário de mudança - Leonardo**

Descrição: Criar um formulário que possa ser preenchido por um novo condômino afim de se ter as informações necessárias sobre aquele condômino. Esta opção pode ser virtual através de um usuário e senha do novo condômino.

<b>Prioridade:</b>		Essencial		Importante	<b>X</b>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF01; RF02;</i>					

#### **[RF17] Cancelar Solicitação - Leonardo**

Descrição: Usuário pode cancelar uma solicitação

<b>Prioridade:</b>		Essencial	<b>X</b>	Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF07; RF15;</i>					

#### **[RF18] Consulta solicitações - Lúcio**

Descrição: Criar um campo no sistema WEB para mostrar todas as solicitações que estão pendentes ou aprovadas.

<b>Prioridade:</b>		Essencial		Importante	<b>X</b>	Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF07 e RF08</i>					

### [RF19] Manutenção de Tipo de despesas - Lúcio

Descrição: Criar uma opção de manutenção de Tipo de despesas.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF01; RF02;</i>					

### [RF20] Manutenção de despesas - Lúcio

Descrição: Manutenção das Despesas.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>						

### [RF21] Tipos de pagamentos - Lúcio

Descrição: Criar uma opção para cadastrar tipos de pagamento. Ex: Boleto, dinheiro, cheque, cartão, etc.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF01;</i>					

**[RF22] Tipos de caixa - Lúcio**

Descrição: Criar uma opção para cadastrar tipos de caixas, onde cada tipo de caixa poderá ter movimentações financeiras de debito ou credito.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>	<i>RF01;</i>					

**[RF23] Históricos de Movimentações - Lúcio**

Descrição: Criar uma opção de cadastros de históricos que especificará a movimentação realizada em um determinado caixa. Ex.: histórico 01 recebimento de aluguel de área de lazer; histórico 02 pagamentos de supermercado.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>						

**[RF24] Contas bancarias - Lúcio**

Descrição: Criar uma opção para cadastrar as contas bancarias onde devera funcionar como espelho da conta real bancária, e todas os pagamentos e recebimentos que influenciarem no caixa poderão ou não ser descontado da conta bancária.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>						

**[RF25] Contas a Pagar - Lúcio**

Descrição: As contas a pagar lançadas no controle de despesas, poderão ser baixadas no controle financeiro, devesa ficar gravado no processo o tipo de pagamento, o histórico de movimentação, e os devidos lançamentos efetivados nos devidos caixas de acordo com a organização estabelecida pelo síndico.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>						

**[RF26] Emissão de boletos - Lúcio**

Descrição: criar uma opção para emissão de boletos, que poderá ser usada após o síndico aprovar o aluguel de alguma área de lazer, ao final do mês na cobrança de condomínio, além da opção de impressão devesa ser possível enviar via e-mail os boletos.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Ator(es):</b>	<i>Síndico, Funcionário</i>					
<b>Requisitos associados:</b>						

**[RF27] Aprovar solicitações - Fernando**

Descrição: O síndico de marcar caso uma solicitação foi aprovada ou não.

<b>Prioridade:</b>		Essencial		Importante	<b>X</b>	Desejável
<b>Requisitos funcionais associados:</b>	<i>RF08;</i>					

**[RF28] Montar Relatório de Condôminos - Leonardo**

Descrição: Criar relatório para mostrar os dados todos os condôminos atuais do condômino, importante para fins de planejamentos.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	RF02;					

**[RF29] Criar Relatório com opções de filtros para os aptos - Leonardo**

Descrição: Criar um relatório onde é possível escolher se quero buscar apenas os dados dos apartamentos de um condômino, de um bloco, ou que estão em situação de devedores.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	RF04;					

**[RF30] Criar relatório de funcionários - Leonardo**

Descrição: Criar relatório com filtro onde possa buscar os dados dos funcionários por nome, salario inicial ou função.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	RF04;					

**[RF31] Criar relatório de funcionários - Leonardo**

Descrição: Criar relatório com filtro onde possa buscar os dados dos funcionários por nome, salário inicial ou função.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	RF04;					

**[RF32] Criar Relatório de Produtos - Leonardo**

Descrição: Criar relatório de produtos onde possa ser feita uma busca através de uma data inicial até uma data final e de quais tipos de produto incluir no relatório, afim de organização de gastos.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	RF05;					

**[RF33] Alterar Dados dos Condôminos. - Leonardo**

Descrição: Alterar os dados básicos dos condôminos ou deletar os dados para entrada de um novo condômino.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	RF02;					

**[RF34] Gerar relatórios de serviços terceirizados - Leonardo**

Descrição: Gerar relatório que apresente os serviços terceirizados que foram contratados em certo período.

<b>Prioridade:</b>		Essencial		Importante	<b>X</b>	Desejável
<b>Requisitos funcionais associados:</b>						

**[RF35] Gerar relatório de registro de ponto - Lúcio**

Descrição: Criar um relatório por funcionário, onde sairá uma planilha com os horários registrados, faltas, e total de horas trabalhadas em um certo período determinado pelo usuário.

<b>Prioridade:</b>		Essencial	<b>X</b>	Importante		Desejável
<b>Requisitos funcionais associados:</b>	<i>RF10, RF11 e RF12</i>					

**[RF36] Relatório de despesas - Lúcio**

Descrição: Criar um relatório com opções de filtros por período e fornecedor de despesas, onde será listado todas as despesas de acordo com os filtros definidos.

<b>Prioridade:</b>		Essencial	<b>X</b>	Importante		Desejável
<b>Requisitos funcionais associados:</b>	<i>RF21</i>					

**[RF37] Emissão dos relatórios na opção de PDF ou DOC. - Lúcio**

Descrição: Poder gerar relatórios em PDF ou DOC afim de ter uma melhor compatibilidade com os sistemas operacionais atuais.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>						

**[RF38] Relatório de movimentação de caixa - Lúcio**

Descrição: O relatório proposto listará as movimentações e os históricos de acordo com o filtro por período do caixa, ou por histórico.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	<i>RF23, RF24, RF25 e RF27</i>					

**[RF39] Relatório de contas a receber. - Lúcio**

Descrição: Criar um relatório que mostrará todas as contas a receber de um determinado período.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	<i>RF23, RF24, RF25 e RF27</i>					



## 2.2. Requisitos não funcionais

### [NF01] Criptografar senhas - Fernando

Descrição: As senhas dos usuários devem ter essa opção para questão de segurança.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Requisitos funcionais associados:</b>	RF01;					

### [NF02] Banco de dados deve ser atualizado em tempo real - Fernando

Descrição: Atualizar o banco de dados com novos cadastros ou alterações em tempo real.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Requisitos funcionais associados:</b>						

### [NF03] Especificar nível de acesso para cada tipo de usuário - Fernando

Descrição: Gerencia o que cada usuário poderá fazer dentro do sistema, por exemplo, um condômino não pode ter acesso ao Caixa do prédio.

<b>Prioridade:</b>	<b>X</b>	Essencial		Importante		Desejável
<b>Requisitos funcionais associados:</b>	RF01;					

**[NF05] Os relatórios devem ter a sua consulta por período previamente preenchido. – Fernando**

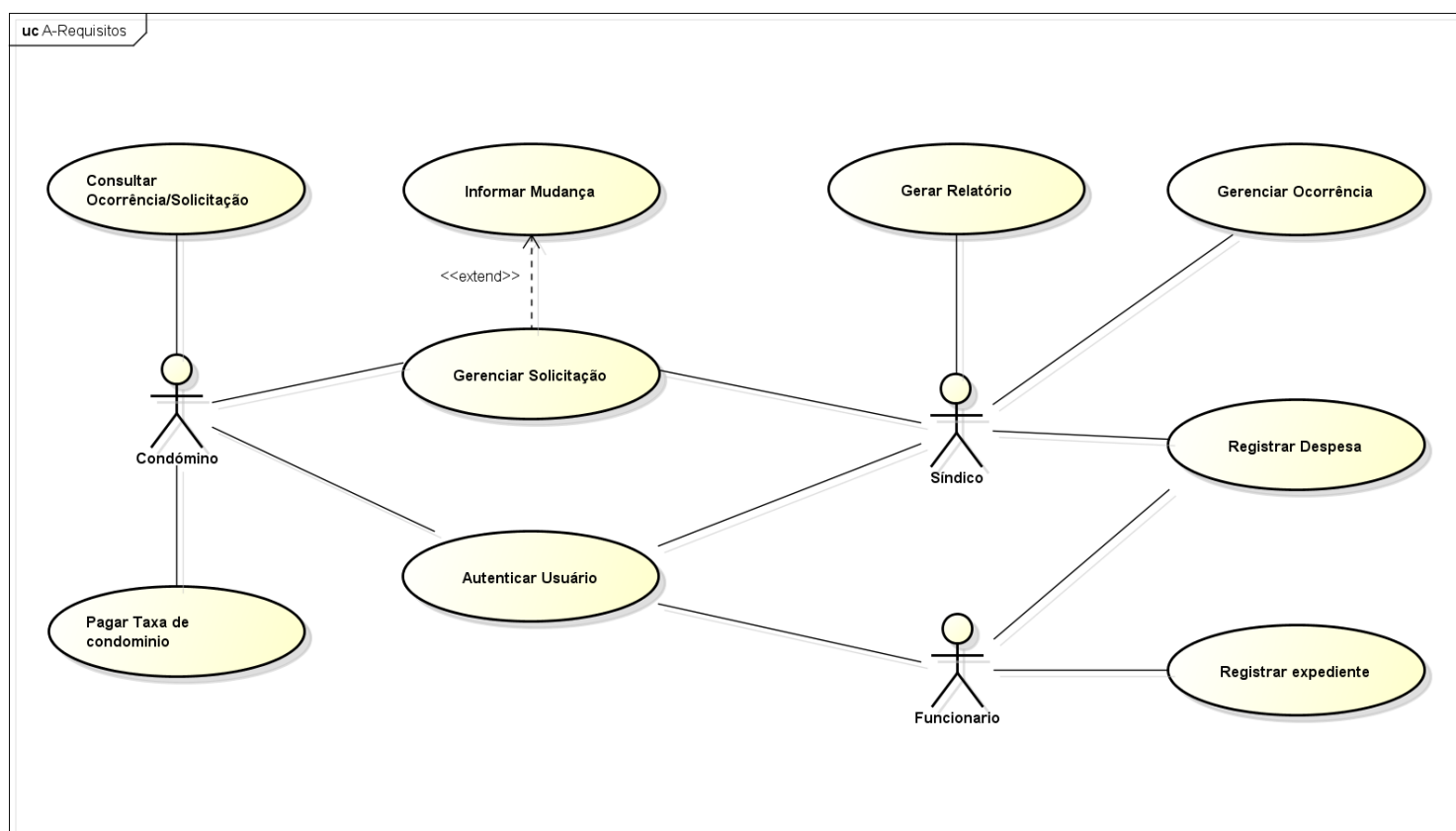
Descrição: A administração condominial é visada para ser feita mensalmente, sendo assim, o período de cada filtro para emissão de relatórios que possuem essa opção deve trazer o campo preenchido com o primeiro dia até o ultimo dia do mês.

<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/>	Desejável
<b>Requisitos funcionais associados:</b>	<i>Referências dos relatórios...</i>					

### 3. CASOS DE USO

A seguir estão representadas as principais funcionalidades do sistema por meio do diagrama de casos de uso.

#### UC001 – Informar Mudança - Fernando



powered by Astah

**Objetivo:** Solicitar requisição para mudança

**Atores:** Condômino, Síndico, Funcionário

**Pré-condição:**

- a) O usuário deve estar logado no sistema,
- b) Condômino estar financeiramente sem restrição

**Fluxo principal**

- a) O Usuário acessa sistema (condômino)
- b) Acessa a opção para solicitar mudança (condômino)
- c) Preenche o formulário de mudança (condômino)
- d) Envia o formulário de mudança (condômino)
- e) Valida o formulário (condômino)
- f) Retornar formulário enviado com sucesso (condômino)

**Fluxo alternativo:**

- a) O usuário acessa o sistema (condômino)
- b) O usuário acessa a pagina de solicitações
- c) O usuário pode cancelar a solicitação desde que esta não esteja aprovada pelo sindico.

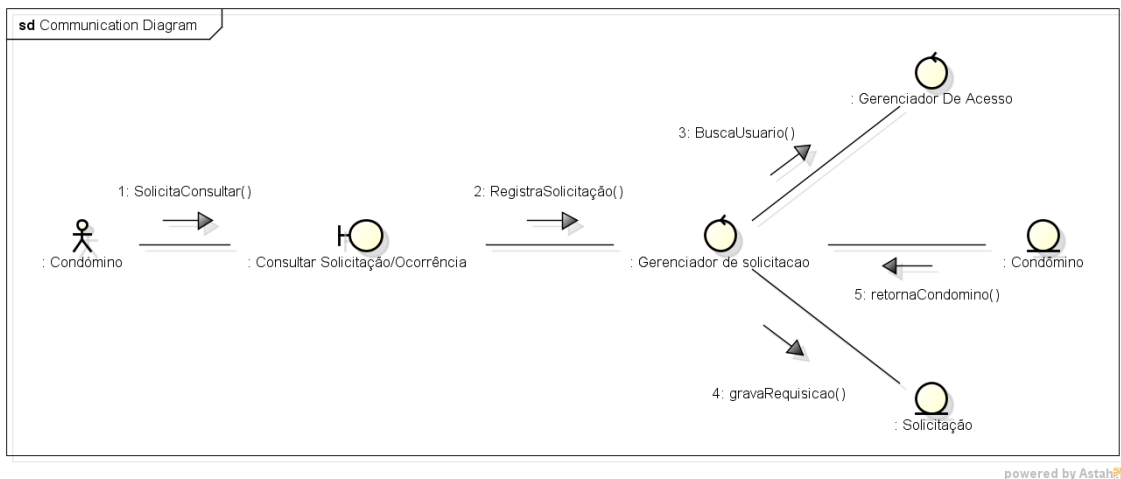
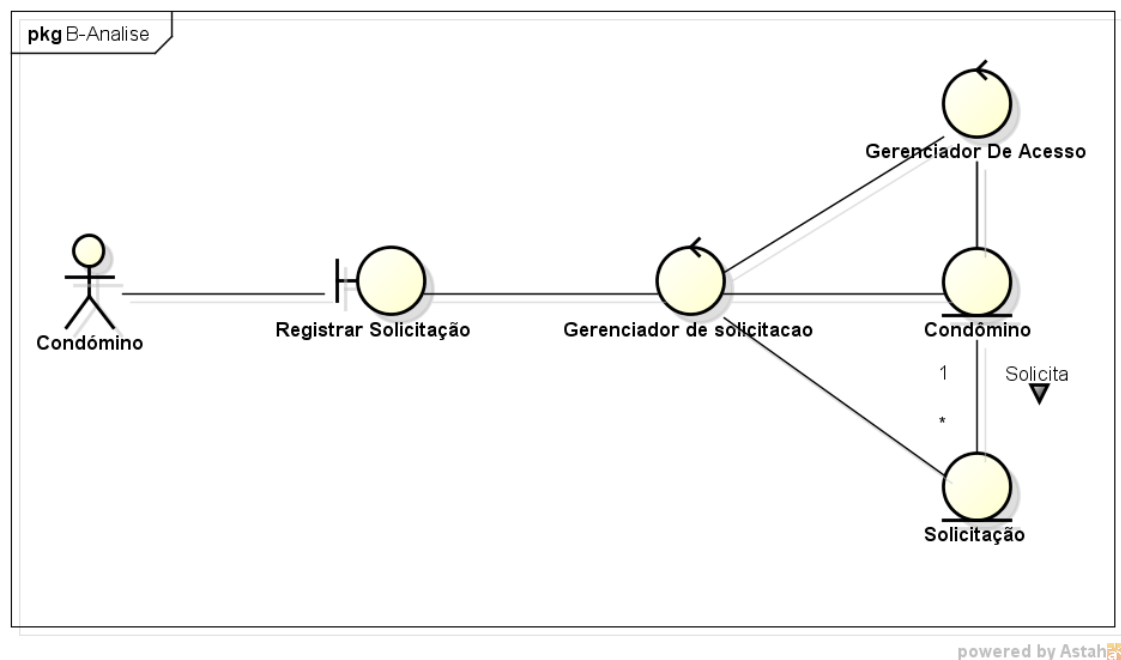
**Fluxo de exceção:**

- a) O usuário envia o formulário preenchido incorretamente
- b) O sistema retorna uma mensagem de erro informando os campos que devem ser verificados

**Pós-condição:**

- a) Solicitação é computada no sistema
- b) o Sindico recebe uma notificação de solicitação no sistema

**Requisitos relacionados:** RF01, RF03, RF07.



## UC002 – Gerenciar Solicitação (Reforma, Reclamação, Limpeza, Agendamento) (Leonardo)

**Objetivo:** Criar uma solicitação.

**Atores:** Condômino, Síndico, Funcionário.

**Pré-condição:**

a) O usuário deve estar logado no sistema.

**Fluxo principal:**

- a) O Usuário acessa sistema (condômino).
- b) Acessa a opção para Criar solicitação de (Reforma, Reclamação, Limpeza, Agendamento) (condômino).
- c) Preenche o formulário da Solicitação (condômino).
- d) Envia o formulário da solicitação (condômino).
- e) Valida o formulário (condômino).
- f) Retornar formulário enviado com sucesso (condômino).

**Fluxo alternativo:**

- a) O usuário acessa o sistema (condômino).
- b) O usuário acessa a pagina de solicitações.
- c) O usuário pode cancelar a solicitação desde que esta não esteja aprovada pelo síndico.

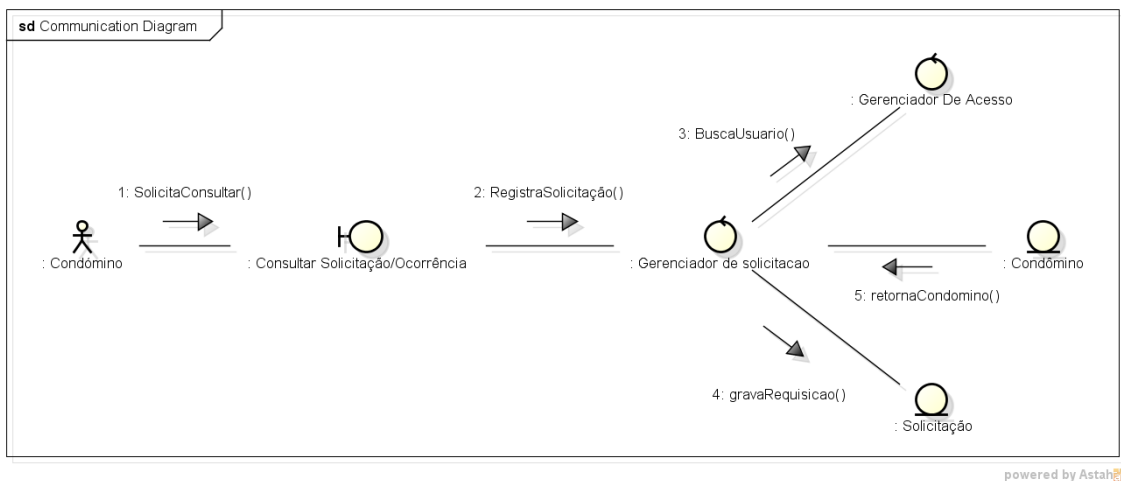
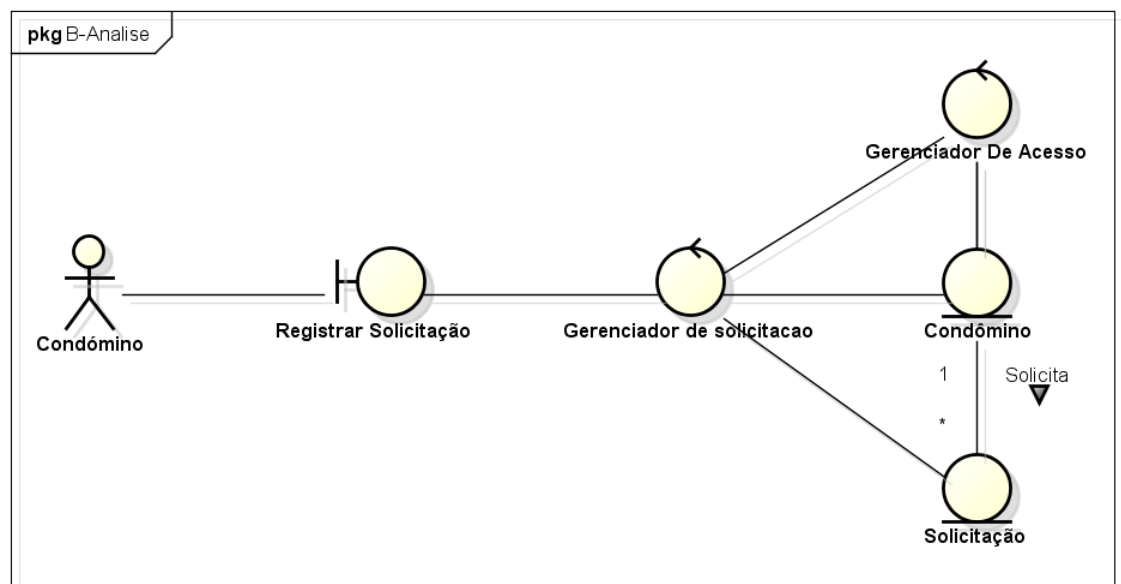
**Fluxo de exceção:**

- a) O usuário envia o formulário preenchido incorretamente.
- b) O sistema retorna uma mensagem de erro informando os campos que devem ser verificados.

**Pós-condição:**

- a) Solicitação é computada no sistema.
- b) O Síndico recebe uma notificação de solicitação no sistema.

**Requisitos relacionados:** RF17, RF08, RF17.



### UC003 – Pagar taxa de condomínio (Leonardo)

#### Objetivo:

Gerar um boleto para o pagamento da taxa do condomínio.

**Atores:** Condômino

#### Pré-condição:

a) O usuário deve estar logado no sistema.

- b) As contas para o mês de referência da taxa do condomínio devem estar fechadas (o sistema informará isso ao usuário).

**Fluxo principal:**

- a) O Usuário acessa sistema (condômino).
- b) O Usuário acessa a página de pagamento.
- c) O Usuário gera o boleto e imprime.

**Fluxo alternativo:**

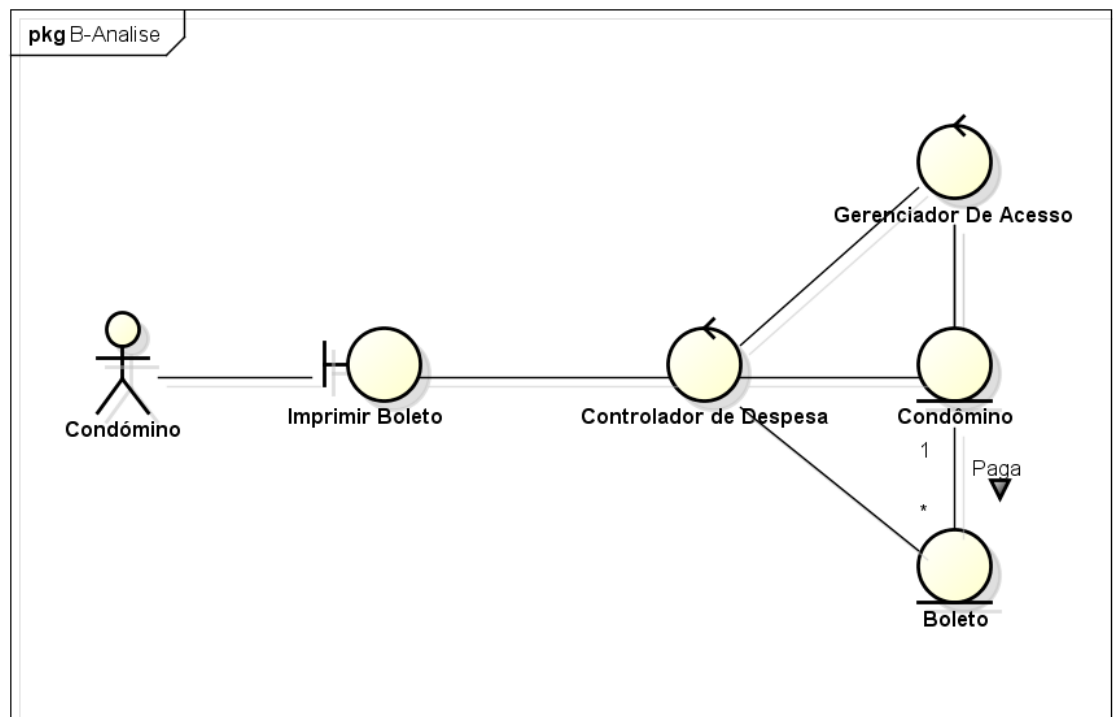
- a) Caso o condômino tenha contas atrasadas ele irá na opção de recalcular a taxa vencida.
- b) O usuário gera o boleto e imprime.

**Pós-condição:**

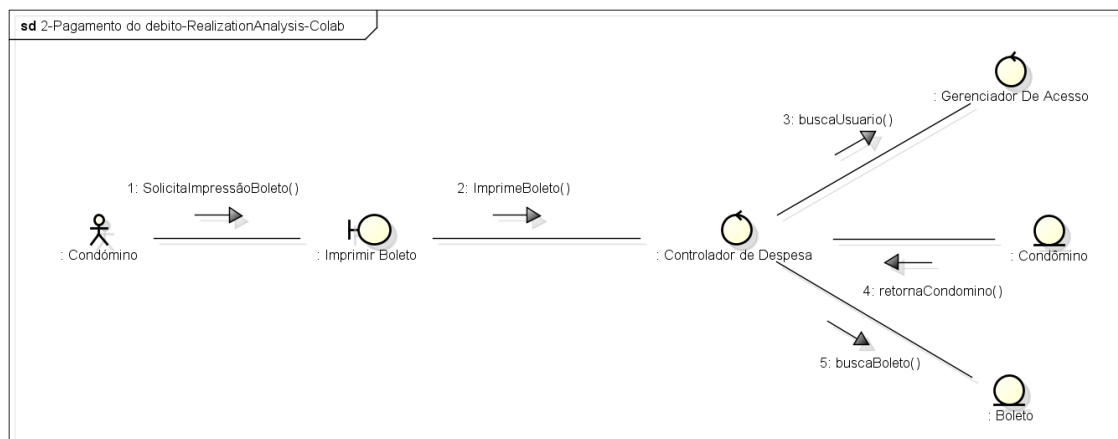
- a) O boleto é gerado
- b) O sistema computa que o condômino gerou o boleto

**Requisitos relacionados:** RF07 R08 RF17.





powered by Astah



powered by Astah

## UC005 – Consultar Ocorrência/Solicitação - Fernando

**Objetivo:** O usuário pode ver quais ocorrências e solicitações estão abertas para ele.

**Atores:** Usuários (Síndico, condômino e funcionário).

**Pré-condição:**

O usuário deve ter passado pela validação de acesso e ter feito a requisição para abertura desta ferramenta.

**Fluxo principal:**

- a) Usuário faz pedido para visualização das ocorrências e solicitações.
- b) (A1) (A2) O sistema retorna uma página contendo as informações das ocorrências e solicitações (A3).
- c) Fim do caso de uso.

**Fluxo alternativo:**

A1 – Usuário pode verificar ocorrências/solicitações que foram finalizadas.

- a) Usuário faz pedido de ocorrências/solicitações que foram finalizadas.
- b) Sistema retorna informações de ocorrências/solicitações finalizadas.
- c) Retorna ao passo “b” do fluxo principal.

A2 – Usuário pode verificar ocorrências/solicitações pendentes.

- a) Usuário faz pedido de ocorrências/solicitações que estão pendentes.
- b) Sistema retorna informações de ocorrências/solicitações pendentes.
- c) Retorna ao passo “b” do fluxo principal.

A3 – Usuário cancela uma solicitação.

- a) Usuário pede para cancelar uma solicitação pendente.
- b) Sistema retorna mensagem pedindo a confirmação da ação.
- c) Usuário pode então confirmar ou não.
- d) Retorna ao passo “b” do fluxo principal.

**Fluxo de exceção:**

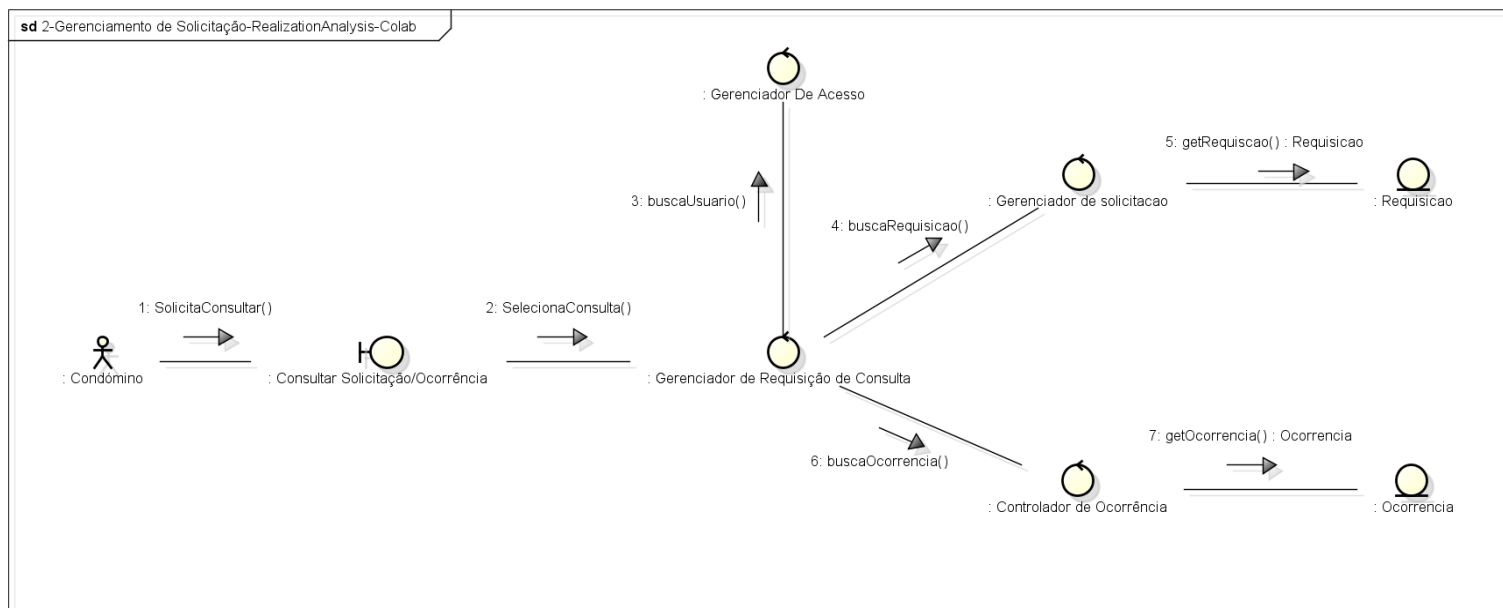
E1 – Sistema não encontra nenhuma ocorrência ou solicitação criada.

- a) Sistema retorna mensagem de que não existe nenhuma informação criada.
- b) Retorna ao passo “c” do fluxo principal.

**Pós-condição:**

O sistema deve retornar as informações ou uma mensagem dizendo que não existem informações.

**Requisitos relacionados:** RF06, RF07, RF08.

**UC006 – Registrar despesa (Leonardo)**

**Atores:** Síndico

**Pré-condição:**

- O usuário deve ter as permissões de síndico, ou de funcionário para efetuar a operação.
- O usuário deve estar logado no sistema com seu usuário e senha.

**Fluxo principal:**

- Na tela de entradas de notas -> entrada de produtos o usuário informara o fornecedor dos itens a serem registrados no sistema.
- Informará a data da nota, e a data de entrada da mesma e o sistema cairá na tela

de para informar os produtos a serem digitados.

c) Poderá fazer a digitação dos produtos informando um filtro com o nome do produto, código de barras, id do produto, ou selecionado em uma lista de produtos.

d) Após digitar o produto, o usuário digitara a quantidade, e o preço de custo do mesmo.

e) Após terminar a digitação de todos os produtos da nota, ira finalizar a nota, e selecionar o tipo de pagamento e prazo caso haja um.

### **Fluxo alternativo:**

#### **A1- Fornecedor não cadastrado.**

a) O usuário entrara na tela entrada de notas -> cadastro de fornecedores.

b) Fará um cadastro do fornecedor que for necessário para realizar a entrada de notas.

#### **A2- Produto não cadastrado.**

a) Na tela de entrada de notas -> cadastro de produto, poderá ser cadastrado um novo produto.

b) No momento do lançamento da nota, na digitação dos produtos, poderá ser acionada a tecla insert e inserido um novo produto no cadastro.

### **Fluxo de exceção:**

#### **E1 – Produtos sem quantidade ou preço de custo.**

a) Caso o cliente tente realizar a digitação de um produto sem terminar de inserir a quantidade ou o preço do produto que esteja sendo digitado o sistema apresentara uma mensagem de erro “digite a quantidade” ou “digite o preço” do produto, ou exclua o produto para que possa ser inserido um novo.

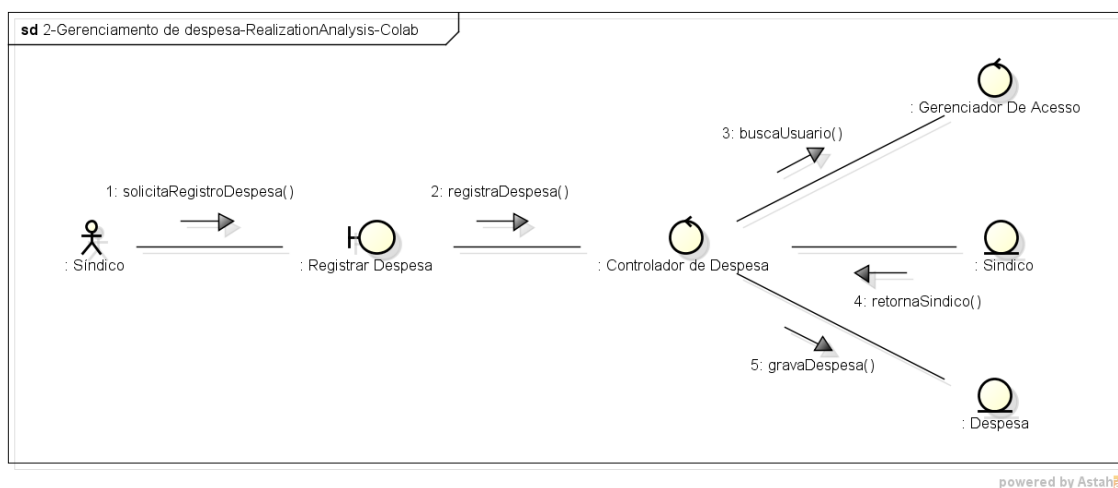
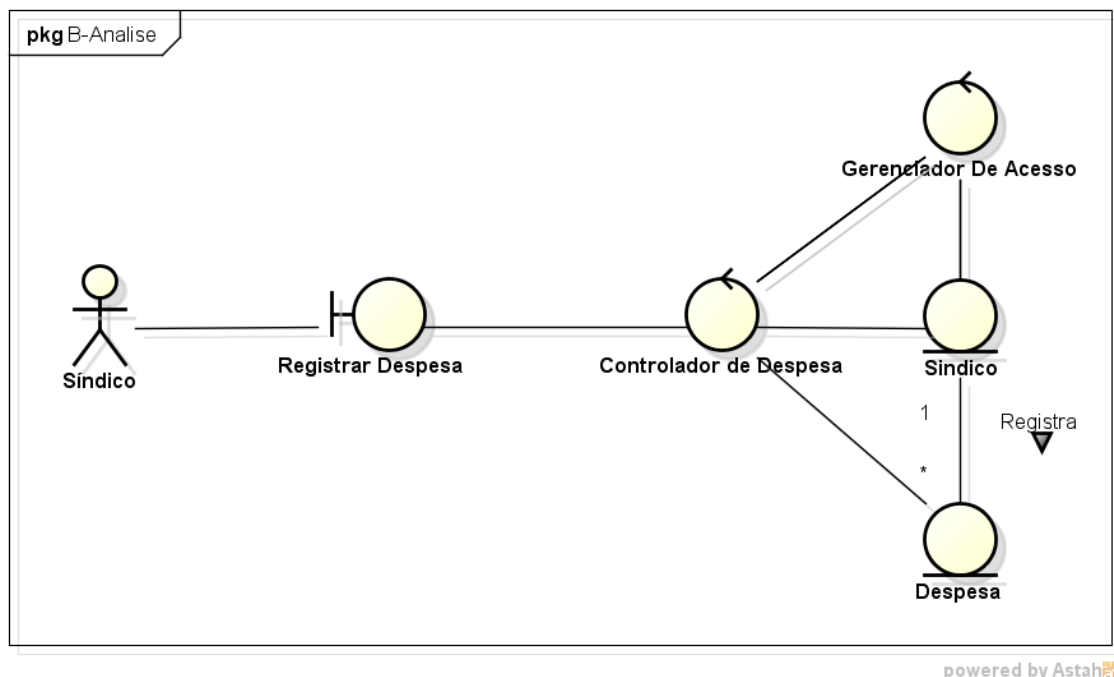
#### **E2 – Sair do sistema no processo de entrada de notas**

a) Caso o cliente queira sair do sistema após iniciado o processo de entrada de notas, o sistema apresentara uma mensagem “Os dados lançados até agora serão perdidos. Confirma? Sim/não”.

### Pós-condição:

Os dados financeiros da nota ficaram gravados no controle de despesas, poderão ser baixados, e os que forem a vista também movimentaram o caixa com a saída de dinheiro para o pagamento da nota.

**Requisitos relacionados:** RF0034, RF0033, RF0013;



**UC007 – Registrar Expediente - Fernando**

**Objetivo:** Registrar o expediente

**Atores:** Síndico, Funcionário

**Pré-condição:**

- a) O usuário deve estar logado no sistema.
- b) O usuário deve ser Funcionário.

**Fluxo principal:**

- a) O usuário registra o horário de entrada de expediente.
- b) O usuário registra o horário de saída de expediente.

**Fluxo Alternativo:**

- a) O usuário chega atrasado para registra a entrada de expediente.
- b) O sistema pede uma justificativa para o atraso.
- c) O usuário registra o horário de saída antes do horário do seu expediente.
- d) O sistema pede uma justificativa para a antecipação.

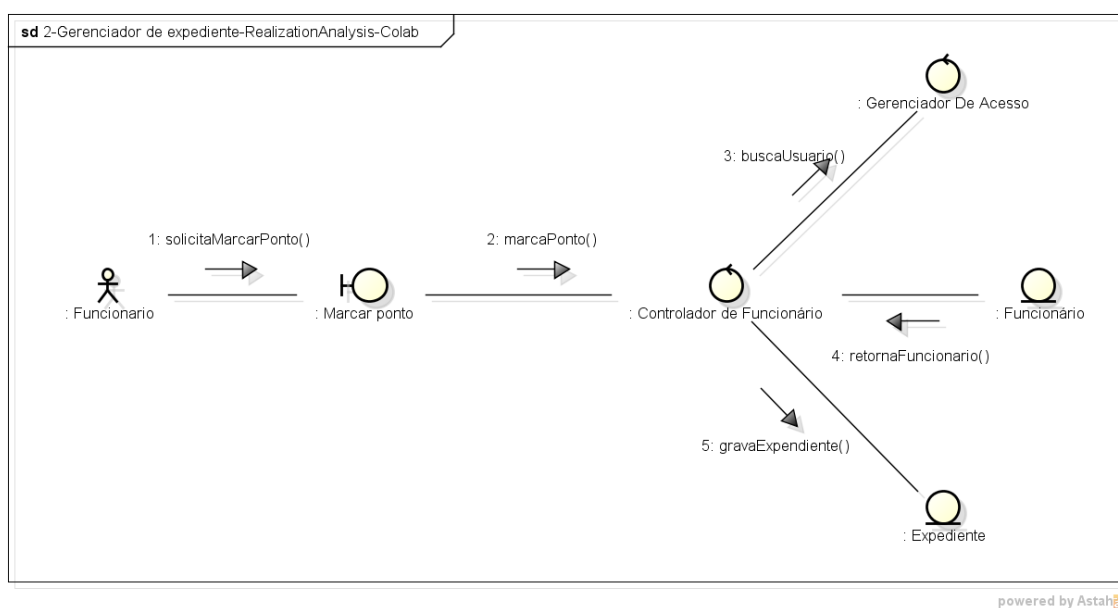
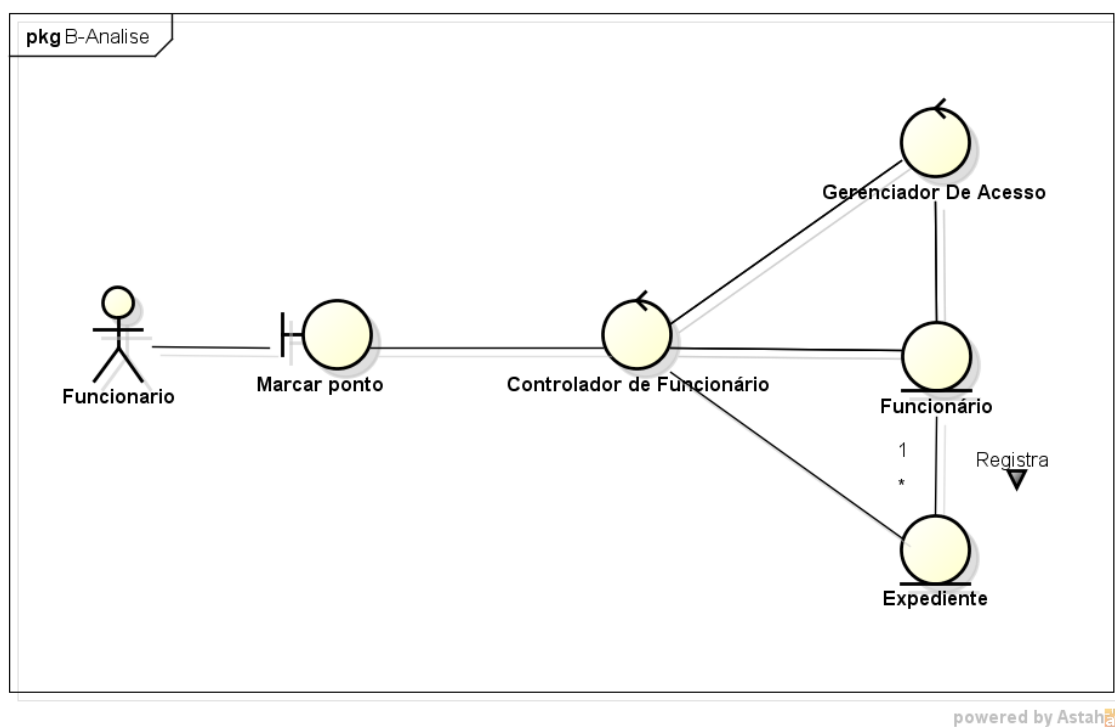
**Fluxo de exceção:**

- a) O usuário não comparece para registrar o expediente.

**Pós-condição:**

- a) Se não houver registro de expediente para aquele dia, este será descontado do banco de horas.

**Requisitos relacionados:** RF01, RF10, RF11, RF12, RF13.



## UC008 – Autenticar Usuário - Fernando

**Objetivo:** O usuário deve acessar o sistema com um login e uma senha.

**Atores:** Síndico, Funcionário, Condômino.

**Pré-condição:**

a) O usuário deve ter uma login e uma senha de acesso, gerado no cadastro do condômino, ou do funcionário ou do síndico.

**Fluxo principal:**

- a) Usuário solicita o acesso ao sistema.
- b) Sistema solicita uma verificação de acesso.
- c) Usuário (A1) digita seu login e senha de acesso.
- d) Sistema valida login e senha (E1).
- e) Usuário tem acesso ao sistema.

**Fluxo Alternativo:**

**A1-** Usuário faz pedido para recuperar a senha.

- a) Usuário digita seu e-mail.
- b) Sistema valida e-mail e envia uma opção de redefinição de senha.
- c) Retorna ao passo “c” do fluxo principal.

**Fluxo de exceção:**

**E1** – Login ou senha inválidos.

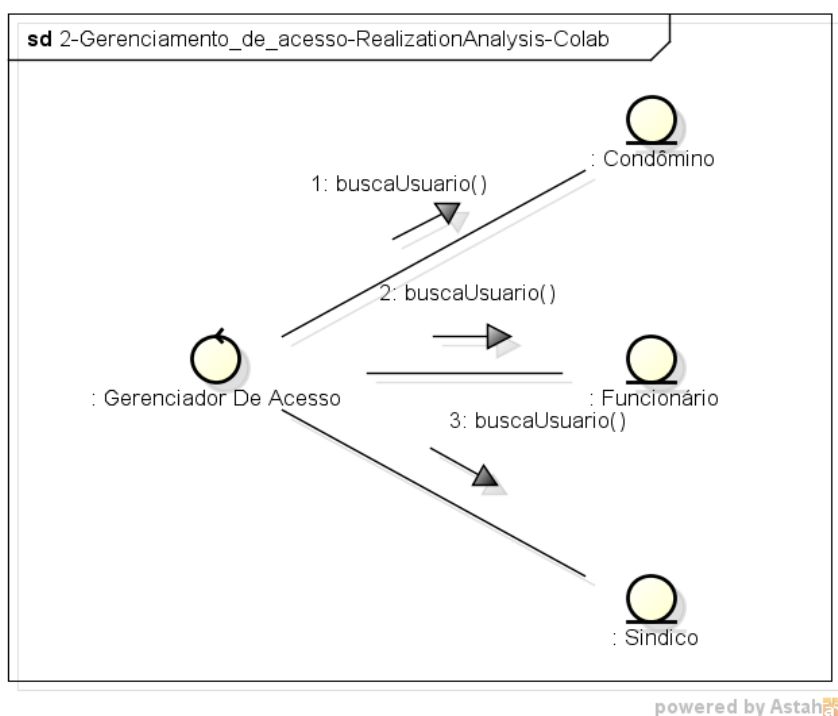
- a) Sistema valida login ou senha de acessos como não autorizado e retorna mensagem de erro para o usuário.
- b) Retorna ao passo “c” do fluxo principal

**Pós-condição:**

O usuário terá acesso ao sistema conforme a sua prioridade de acesso. Exemplo: no caso do síndico todas as ferramentas do sistema estarão liberadas.

**Requisitos relacionados:** RF01.





## UC009 – Gerenciar Ocorrência- Fernando

**Objetivo:** Manutenção da Ocorrência

**Atores:** Síndico

**Pré-condição:**

- a) O usuário deve estar logado no sistema.
- b) O usuário deve ser do tipo Funcionário.

**Fluxo principal:**

- a) O Usuário acessa sistema
- b) Acessa a opção para Criar Ocorrência que pode ser um simples aviso de manutenção predial, licitações que o condomínio gerou, informativo de multa para o condômino que desrespeitar as regras do condomínio.
- c) Preenche o formulário da Ocorrência
- d) Envia o aviso
- e) Valida o aviso

f) Retornar aviso enviado com sucesso

### Fluxo Alternativo:

- a) O usuário acessa o sistema
- b) O usuário acessa a pagina de ocorrência
- c) O usuário pode cancelar a ocorrência

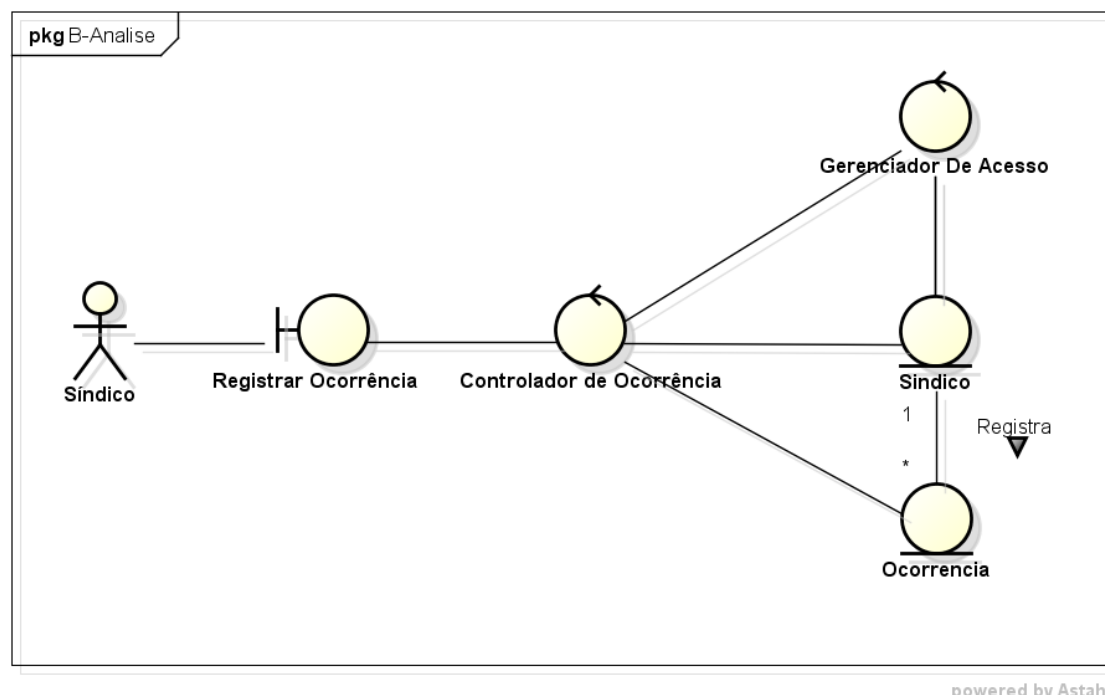
### Fluxo de exceção:

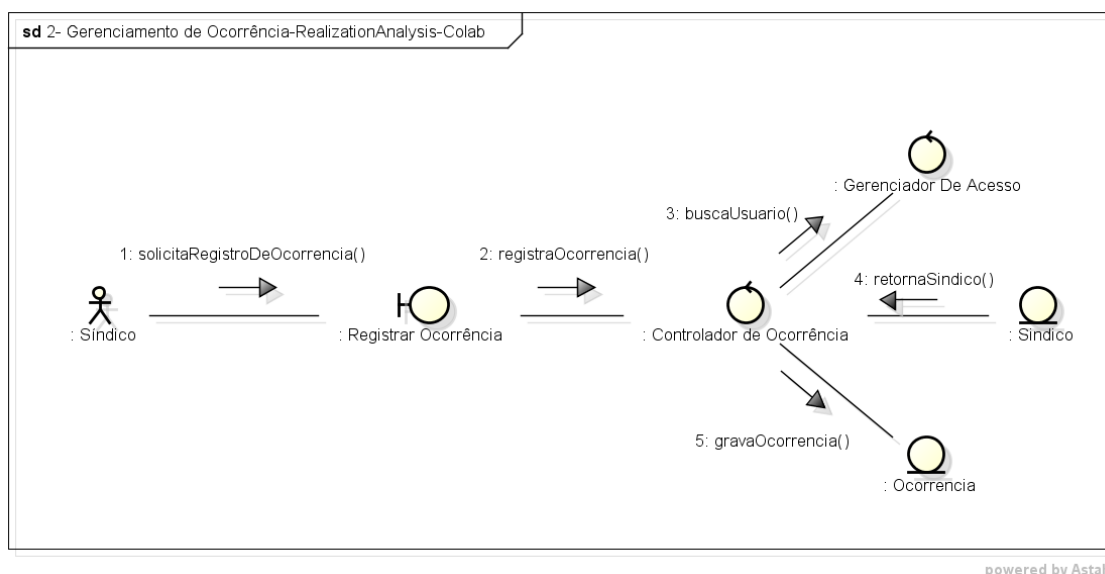
- a) O usuário envia o aviso preenchido incorretamente
- b) O sistema retorna uma mensagem de erro informando os campos que devem ser verificados

### Pós-condição:

- a) Ocorrência é computada no sistema
- b) Os condôminos recebem uma notificação de ocorrência no sistema

**Requisitos relacionados:** RF01, RF03, RF06.





#### 4. GLOSSÁRIO DE TERMOS

TERMOS	DESCRIÇÃO
Sindico	Responsável pelo Sistema. Gerenciador do Condomínio.
Condômino	Morador do prédio, Condomínio.
Requisição	É uma solicitação do condômino para o sindico.
Ocorrência	Registro do sindico para um evento, reparo ou reclamação.
Funcionário	Operador do sistema e funcionário do condomínio.
Expediente	São as jornadas de trabalho do funcionário.
Boleto	Documento que o condômino paga para o banco que o repassa para o caixa do condômino.
Caixa do Condomínio	É onde é armazenado o dinheiro do condomínio e é quem recebe os pagamentos.
Fornecedor	É quem fornece os produtos para o condomínio
Despesa	São registros de gastos com o condomínio.
Produto	São os produtos ou serviços que estão no registro de despesas.

##### 4.1. Quanto à viabilidade técnica

A tecnologia atual oferece uma gama de recursos que pode contribuir com a construção do projeto. Estamos em um momento que está sendo muito discutido desenvolvimento web, armazenamento e tantos outros assuntos relativos ao tema. Em relação a alguma restrição de custo de processamento, cremos que não haverá maiores problemas, no nosso entendimento os requisitos para o uso do sistema são básicos, praticamente nulo – necessitando apenas acesso à internet e um navegador web.

A questão do prazo é um ponto importante, pois como se trata de um evento com data estabelecida, existe um prazo válido para a entrega do projeto. Mas considerando as necessidades e o tempo hábil, cremos que o tempo é suficiente.

## **4.2. Avaliação dos Riscos**

### **• Risco 1**

Falta de tempo, uma vez que somos todos universitários e trabalharemos no contra turno. O sistema pode ser complexo no caso de mudança de síndico, necessitando de treinamento, para entender o fluxo do sistema.

### **• Risco 2**

Falta de informação, caso aconteça de não conseguirmos informações com tempo suficiente análise de viabilidade.

O sistema irá ajudar na comunicação entre o síndico e o condômino, devido ao síndico quase sempre estar indisponível, através da rede o condômino pode solicitar algum requerimento, reclamações, sugestões. E assim o síndico pode realizar a verificação dos registros.

Deixar todas as contas do condomínio disponíveis para maior transparência, devido as contas serem divididas entre os condôminos.

Controle de contas. O síndico não precisará mais enviar o boleto para todos os condôminos. O boleto ficará disponível na rede para download.

### **• Risco 3**

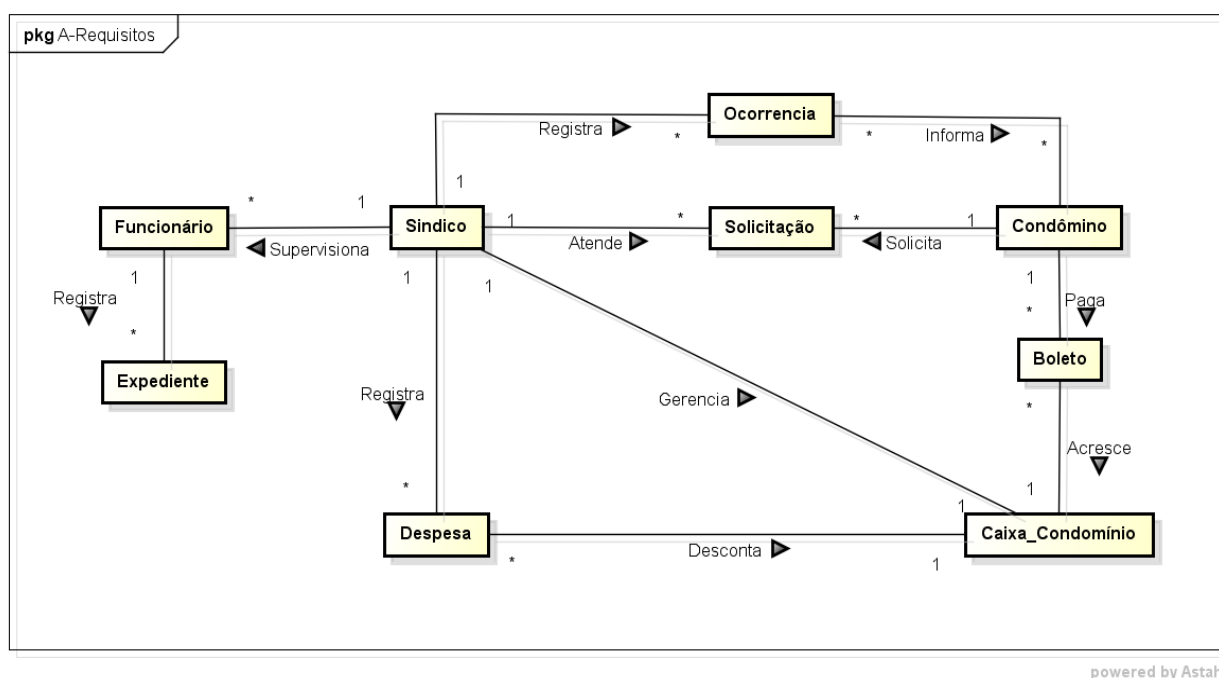
Saída de um ou mais integrantes do projeto. O risco pode variar de médio a grave, dependendo do nível de influência que o integrante tem no desenvolvimento do projeto. Uma tentativa de resolução seria a contraproposta salarial, e estratégia motivacional.

### **• Risco 4**

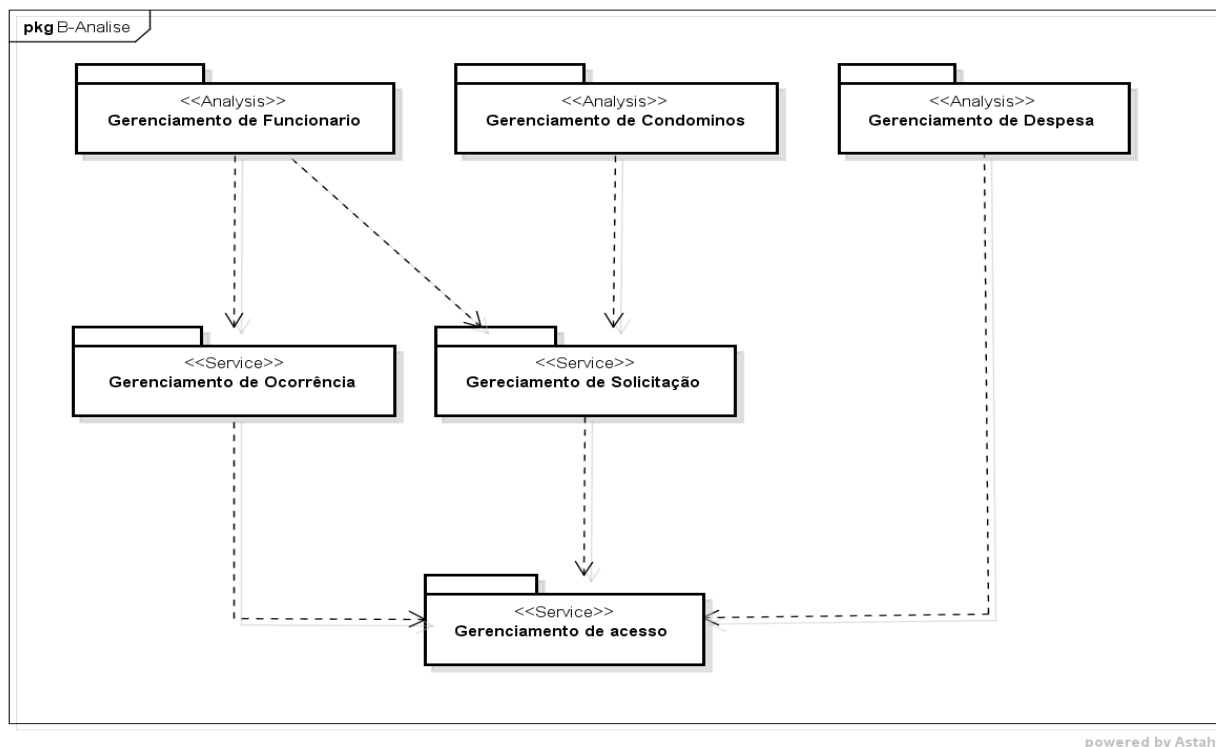
Baixa experiência no desenvolvimento de projetos de software dos membros da

equipe. Risco é considerado de baixa gravidade. Uma tentativa de resolução seria pesquisar para sanar possíveis dúvidas existentes. Existem inúmeras fontes de informação gratuitas disponíveis na internet.

### 4.3. Modelo objeto de negócio



#### 4.4. Diagrama de pacotes da área de negócio



powered by Astah

#### 4.4. Estilos Arquiteturais

##### Pipes and Filters (descartado):

Por ser uma aplicação que possivelmente terá várias mudanças em seus componentes durante o desenvolvimento, este fator será desvantajoso caso escolhamos este estilo, visto que alterações frequentes podem limitar e deixar a manutenção mais custosa do que o esperado. Outro fator é que o fluxo de dados não poderá ser unidimensional para o nosso sistema, visto que o usuário fará requisições e receberá as notificações das mesmas.

##### Camadas (utilizado parcialmente):

Não usaremos a arquitetura de camadas pura pois essa apresenta algumas desvantagens como o de uma camada só poder se comunicar com a sua adjacente ou sua posterior, no nosso sistema queremos que todas as camadas troquem mensagens com uma camada controladora.

**Padrões de Camadas MVC (utilizado):**

Com o aumento da complexidade dos sistemas/sites desenvolvidos hoje, essa arquitetura tem como foco dividir um grande problema em vários problemas menores e de menor complexidade. Dessa forma, qualquer tipo de alterações em uma das camadas poderá não interferir nas demais, facilitando a atualização de layouts, alteração nas regras de negócio e adição de novos recursos. Em caso de grandes projetos, o MVC facilita muito a divisão de tarefas entre a equipe.

Abaixo serão listadas algumas das vantagens em utilizar MVC em seus projetos:

- Facilita o reaproveitamento de código;
- Facilidade na manutenção e adição de recursos;
- Maior integração da equipe e/ou divisão de tarefas;
- Diversas tecnologias estão adotando essa arquitetura;
- Facilidade em manter o seu código sempre limpo;

**Padrão Cliente servidor (utilizado):**

O padrão de Cliente servidor, será utilizado parcialmente em nossa aplicação, na parte em que o usuário fará requisições para o Síndico por meio de uma página web.

Usaremos uma versão do cliente servidor compatível com o MVC, uma em que o browser troca mensagens com a view e com o nosso controlador.

**Estilo orientado a Objetos (Utilizado parcialmente):**

O Estilo Orientado a objetos se assemelha muito com o conceito de orientação a objeto, esse estilo não impõe muitas regras ao seu desenvolvimento, e não exemplifica alguns fluxos além do que conhecemos como orientação a objetos, usaremos os conceitos de orientação a objeto assim como o encapsulamento, polimorfismo e herança, junto com o framework hibernate espelharemos nossas classes para persisti-las em banco de dados.

**Invocação implícita (Descartado):**

- **Baseado em Eventos:**

Pretendemos controlar toda regra do sistema no modulo de controle, e a arquitetura implícita baseada em eventos não nos proporciona a visibilidade para estrutura o



nosso sistema dessa maneira, os eventos que serão disparados parece descentralizar a regra de negócio, caso os métodos de acesso ao modelo sejam feitos nesses gatilhos.

Uma forma pura desse estilo é raramente utilizada, se fossemos utilizar, iríamos utilizar uma maneira que fosse compatível com o MVC.

- **Broadcast:**

Considerando que teríamos que desenvolver um evento, as sub-rotinas teriam que ficar aguardando esse evento, essa quantidade de processamento seria desnecessária em nossa aplicação.

- **Orientado a Interrupção:**

O modelo orientado a interrupção é mais utilizado em aplicações de comunicação em tempo real, não utilizaremos nenhum tipo de comunicação em tempo real.

**Black Board (descartado):**

O Estilo arquitetural, black-board, também conhecido como quadro-negro consistem em manter uma central de dados, que é acessada por um conjunto de componentes (células de conhecimento), essas células de conhecimento percorrem essa central de dados e fazem o processamento, retornando os resultados obtidos, para outras células de conhecimento processarem ou retornar essa informação para o agente requerido. Essa arquitetura costuma ser mais utilizada por aplicações de inteligência artificial, e também provedores de pesquisa.

**Ponto A Ponto (descartado):**

Arquitetura ponto a ponto costuma ser mais utilizada em sistemas distribuídos, esses tipos de sistemas dividem o processamento de suas aplicações em seus nós formando uma rede de processamento parcializado. Essa estrutura é mais utilizada em aplicações que compartilham arquivos, ou as que subdividem o seu processamento em outros pontos.

**Sistemas Orientados a Serviços (descartado):**

Implementar padrões como o SOAP necessita de muito processamento, e servidores muito fortes. Sistemas orientados a serviço costumam ser sistemas distribuídos, que

oferecem um serviço específico e recebem muitas requisições feitas por outros sistemas, assim como uma busca de CEP, endereço, ou informações sobre um CPF como busca no Serasa etc.

**Computação na Nuvem (descartado):**

Essa arquitetura, consiste em manter as aplicações, serviços e servidores sendo virtualizados em nuvem, qualquer que seja a parte da sua aplicação que responde a requisições pode ser armazenada e acessa pelos seus usuários através da internet. Mesmo considerando o custo uma economia na hora da implantação utilizar um serviço de nuvem para manter os servidores a longo prazo ficaríamos dependentes desse serviço, uma vez que ficássemos sem internet, o que acontece com frequência devido à má qualidade das provedoras brasileiras e o descaso com a qualidade de transmissão para o consumidor, o acesso as aplicações seria impossível, mesmo que isso não ocorresse estaríamos nas mãos das empresas que mantem os serviços.

## **5. Framework**

### **5.1. Grails**

Grails é um framework para Groovy, uma linguagem orientada a objetos com tipagem dinâmica que roda na JVM para desenvolvimento web. Possui como sua maior característica o ganho em produtividade, basicamente reduzindo a quantidade de decisões que o desenvolvedor precisaria tomar, o framework, baseia-se em convenções - Convention-over-configuration (Programação por convenção). Usa por padrão MVC e utiliza o framework Spring.

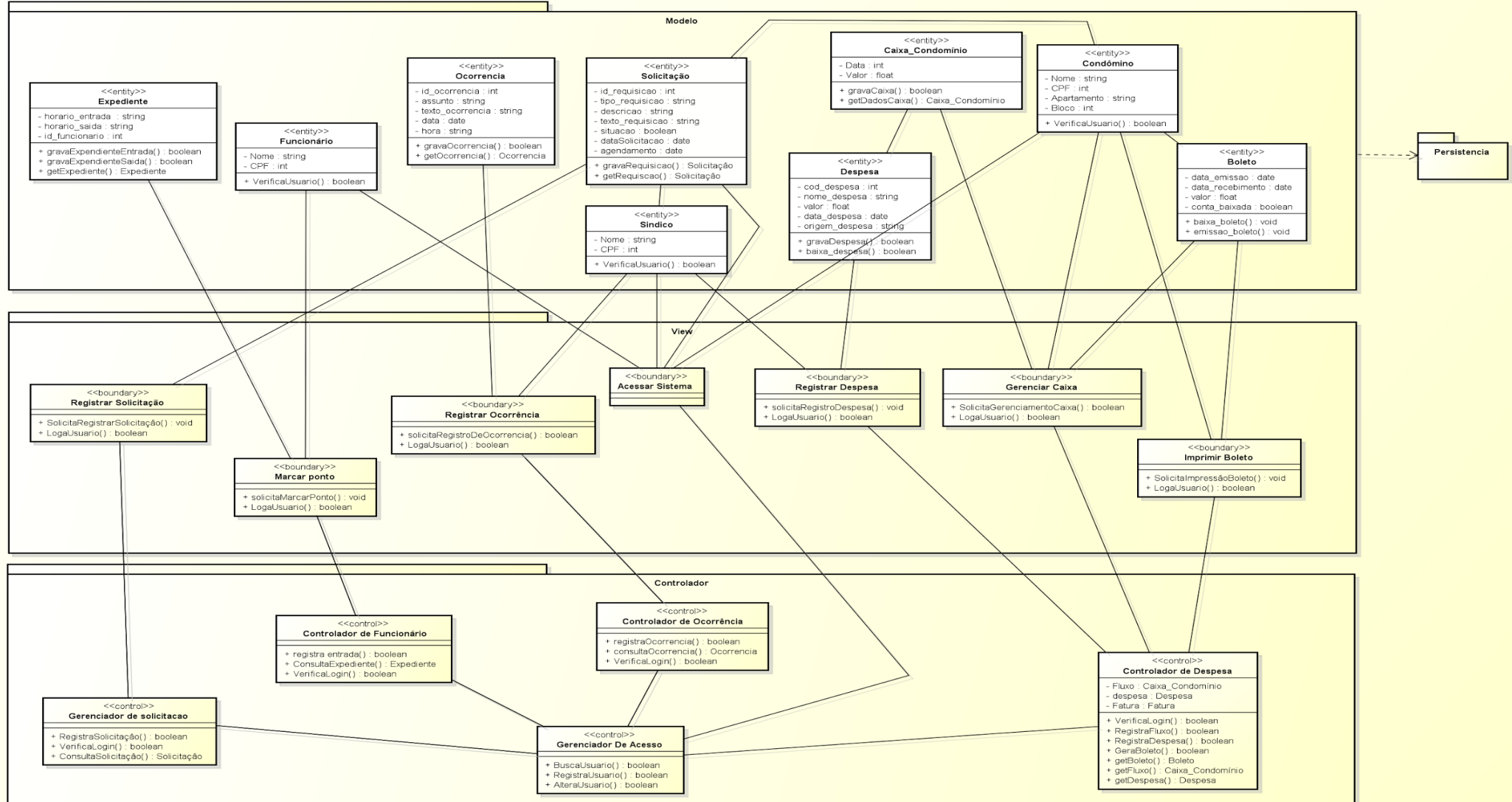
### **5.2. Hibernate**

Hibernate é um framework Java responsável por realizar a persistência dos dados no Banco de Dados utilizando o Mapeamento Objeto-Relacional. Basicamente ele trabalha como um middleware, mantendo-se entre a aplicação e o banco de dados, intermediando e reduzindo a complexidade para o desenvolvedor. Utilizando arquivo XML, facilita o mapeamento dos atributos entre a base de dados relacionais e o modelo de objeto. Tem como principal característica a transformação das classes e tipos de dados da linguagem Java para tabelas e os tipos de dados da linguagem SQL. Também é responsável por gerar as chamadas da linguagem SQL, tais quais Select, Insert, Update, Delete, dentre outras. Em resumo, dispensa ao desenvolvedor o trabalho manual da conversão dos tipos de dados e mantém a aplicação portátil para qualquer banco de dados que utilize a linguagem SQL.

Iteração	Descrição	Responsável
1ª Iteração	<ul style="list-style-type: none"> <li>• Cadastro de funcionários</li> <li>• Cadastro de apartamento</li> <li>• Validação de acesso</li> <li>• Cadastro de condômino</li> </ul>	Fernando, Leonardo, Lúcio
2º Iteração	<ul style="list-style-type: none"> <li>• Solicitar mudança</li> </ul>	Fernando
3º Iteração	<ul style="list-style-type: none"> <li>• Criar Solicitação</li> </ul>	Leonardo
4º Iteração	<ul style="list-style-type: none"> <li>• Pagar Taxa de Condomínio</li> </ul>	Leonardo
5º Iteração	<ul style="list-style-type: none"> <li>• Consultar Ocorrências</li> </ul>	Fernando
6º Iteração	<ul style="list-style-type: none"> <li>• Lançar Despesas de Produto</li> </ul>	
7º Iteração	<ul style="list-style-type: none"> <li>• Gerenciar Caixa</li> </ul>	Lúcio
8º Iteração	<ul style="list-style-type: none"> <li>• Registrar Expediente</li> </ul>	Fernando

### 5.3. Diagrama de classes do projeto

Grails



## REFERÊNCIAS

CAFÉ, Adriel. **INTRODUÇÃO AO HIBERNATE.**  
<<http://adrielcafe.com/cafelog/hibernate/45-introducao-ao-hibernate>> Acessado em 15/08/2015.

KIKO, Dev. **POR QUE GRAILS É IMPORTANTE PARA OS DESENVOLVEDORES JAVA?** <<http://www.itexto.net/devkico/?p=336>>  
Acessado em 15/08/2015.