



**BACHARELADO EM  
CIÊNCIA DA COMPUTAÇÃO**

**GABRIEL ARAÚJO ZANARDO  
GUILHERME MORAES DA SILVA  
JOÃO PEDRO SOARES DA FRANCA**

**PROJETO INTEGRADOR ENTRE AS DISCIPLINAS:  
Análise e Projeto de Sistemas, Ferramentas de Programação I e  
Banco de Dados**

Sakai Burguer

## SUMÁRIO

Capítulo 1 -	DESCRIÇÃO GERAL DO PRODUTO .....	3
1.1	Escopo do Produto.....	3
1.2	Funções do Produto .....	4
1.2.1	Funções Fundamentais.....	4
1.2.2	Funções Básicas .....	5
1.2.3	Funções de Saída.....	5
1.3	Regras do Negócio .....	5
1.4	Perspectiva do produto / Requisitos não funcionais.....	6
1.5	Restrições, Suposições e Dependências .....	7
Capítulo 2 -	REQUISITOS ESPECÍFICOS .....	8
2.1	Diagrama de Casos de Uso.....	9
2.2	Especificações de Casos de Uso e Diagramas de Atividades.....	10
2.2.1	Diagramas de Atividades .....	10
2.2.2	Especificações de Casos de Uso .....	21
Capítulo 3 -	PROJETO DE SOFTWARE .....	29
3.1	Arquitetura Lógica de Software .....	29
3.2	Diagrama de Classes.....	30
3.3	Diagramas de Sequência.....	33
3.4	Mapeamento OO-Relacional (Modelo Lógico de Dados).....	40

## Capítulo 1 - DESCRIÇÃO GERAL DO PRODUTO

### 1.1 *Escopo do Produto*

O software será chamado Sakai Burger e terá como objetivo informatizar uma empresa do ramo alimentício, auxiliando o controle administrativo, atendimento ao cliente e gerenciamento de tarefas, bem como melhorar o desempenho geral da empresa.

O sistema contemplará a implementação de cadastro de clientes, funcionários, pedidos, entradas e saídas do caixa, cadastro e edição de cardápio, controle de estoque, onde será mensurada a quantidade de cada item no estoque, e uma série de relatórios que fornecerão informações para controle de caixa e de pedidos realizados pelos clientes.

Diante do problema exposto pelo cliente, que consiste na ausência total de controle de estoque e de pedidos, o que obriga o cliente a efetuar o gerenciamento de forma manual e um controle de estoque sem eficiência. A ausência do cadastro de clientes e a dificuldade da realização de pedidos via telefone, já que tudo é feito por um funcionário e arquivado em papel, dificultam a comunicação das atendentes com a cozinha e retardam o processo de produção do pedido. Também possui dificuldades no controle de caixa e de cardápio, pois é necessário citar o cardápio para o cliente e alterações se tornam mais complexas.

A solução propõe ao cliente um sistema de delivery para facilitar a comunicação entre a empresa e seus clientes. Já para a parte administrativa será proposto um controle total de estoque através da utilização de códigos para os produtos que utiliza. O controle de caixa será realizado automaticamente durante a confirmação dos pedidos feitos pelos clientes da empresa. Assim, melhorando o sistema de atendimento ao cliente, o gerenciamento do negócio e o controle financeiro de entradas e saídas.

O software terá muitos benefícios para a empresa, como por exemplo: um efetivo controle de caixa de entradas, saídas e contas pendentes, bem como o de estoque e de clientes, fornecendo à empresa relatórios, informações valiosas para a elaboração de estratégias comerciais e apoio as decisões em geral.

O software também será integrado com sistemas de pagamento para delivery e incluirá um módulo para geração de notas fiscais, automatizando as transações

fiscais.

## **1.2 Funções do Produto**

O sistema permitirá várias funções como: cadastro, consultas, relatórios e movimentações, conforme especificadas abaixo.

### **1.2.1 Funções Fundamentais**

RF\_FF01 – Fechar pedido. Ao confirmar o fechamento de um pedido, o sistema registra os dados relacionados a tal pedido e envia para o pagamento do pedido. Para isso são necessários os seguintes itens de informação: data do pedido, horário do pedido, Identificação do(s) Produto(s), Quantidade vendida do(s) Produto(s), Endereço a ser entregue (caso necessário).

RF\_FF02 – Pagar pedido. Ao fechar um pedido, o sistema permitirá o cadastro do pagamento do pedido. Para isso são necessários os seguintes itens de informação: Identificação do Pedido, Valor, Data, Hora, Forma de Pagamento, Quantidade de Parcelas (caso necessário).

RF\_FF03 – Registrar conta a pagar. O sistema permitirá o registro de despesas que a empresa terá que pagar como água, luz, telefone e fornecedores. Sua baixa se dará pela quitação do pagamento, onde será preenchido o valor e data do pagamento efetuado. Para isso são necessários os seguintes itens de informação: Data de Vencimento, Data de Pagamento, Descrição, Valor e Número da NFE que gerou a conta.

RF\_FF04 – Registrar suprimento no caixa. O sistema permitirá o registro de suprimento no caixa, poderá ser feito manualmente durante a abertura do caixa e/ou durante o dia caso necessário. Para isso são necessários os seguintes itens de informação: Data, Hora, Valor, Descrição, Identificação do Funcionário.

RF\_FF05 – Realizar Sangria. O sistema permitirá o registro de sangrias no caixa, poderá ser feito manualmente em casos de necessidade. Para isso são necessários os seguintes itens de informação: Data, Hora, Valor, Descrição, Identificação do Funcionário.

RF\_FF06 – Realizar entrega. O sistema permitirá que os funcionários terceirizados encarregados de realizar as entregas, visualizem os pedidos e alterem o seu status após a confirmação da entrega. Para isso são necessários os seguintes itens de informação: Data, Hora, Endereço, Identificação do Cliente e Identificação do Funcionário Terceirizado.

### **1.2.2 Funções Básicas**

Refere-se às operações CRUD (inserção, consulta, edição e exclusão) necessárias para a manutenção de dados. São elas:

RF\_FB01 – Manter Clientes. Itens de informação necessários: CPF, Nome, Endereço, Data de Nascimento, E-mail e Telefones.

RF\_FB02 – Manter Funcionário. Itens de informação necessários: CPF, Nome, Endereço, Data de Nascimento, Cargo, Salário, E-mail e Telefones.

RF\_FB03 – Manter Produtos. Itens de informação necessários: Código, Descrição, Tipo de Item, Nome do Item, Preço Unitário, Quantidade em Estoque e Em Promoção.

RF\_FB04 – Manter Pedido. Itens de informação necessários: Identificação do Pedido, Data, Hora do Pedido, Valor, Status, Endereço de Entrega e Hora da Entrega.

RF\_FB05 – Manter Carrinho. Itens de informação necessários: Identificação do Carrinho, Valor, Cliente Associado ao Carrinho e Quantidade de Itens.

RF\_FB06 – Manter Terceirizados. Itens de informação necessários: CPF, Nome, Endereço, Data de Nascimento, Data de Contratação, Data de Término, Salário, Função, E-mail e Telefone.

### **1.2.3 Funções de Saída**

RF\_FS01 – Gerar Relatório de Fluxo de Caixa. Filtro: Período. Itens de informação exibidos: Histórico de Entradas e Saídas, Relação de Ganhos e Despesas.

RF\_FS02 – Exibir Pedido. Itens de informação exibidos: Identificação do Pedido, Data, Hora, Valor, Status e Endereço a ser entregue.

RF\_FS03 – Exibir Informações de Pagamento. Itens de informação exibidos: Identificação do Pedido, Data, Hora, Valor, Forma de Pagamento.

## **1.3 Regras do Negócio**

O valor do frete será calculado de acordo com a distância, onde pedidos em até 10 quilômetros de distância terão um custo fixo de cinco reais, caso a distância exceda esse limite, será cobrada uma taxa de cinquenta centavos para cada quilômetro.

Os pedidos de delivery poderão ser pagos somente por meio virtual, impossibilitando que motoboys recebam o pagamento, seja ele via cédulas físicas ou

pagamento via cartão através de maquininhas.

Os alunos do IFSP receberão um desconto de dez por cento no valor final do pedido.

#### **1.4 Perspectiva do produto / Requisitos não funcionais**

O sistema da Sakai Burguer será desenvolvido em ambiente Java, com xHTML, CSS e Javascript para a parte visual e banco de dados em PostgreSQL. O produto possuirá comunicação com sistemas de pagamento para suporte dos pedidos via delivery. Além disso, haverá a utilização de um sistema para geração de notas fiscais.

O produto será o primeiro a ser desenvolvido e implementado para a empresa, possui formato padrão para um site do ramo alimentício, com barras de rolagens, relatórios com o nome da empresa e o título do relatório, mensagens com botões em português.

Quanto às limitações aos usuários, o sistema propiciará a execução em quatro níveis, sendo o 1º nível de acesso total que será para Sócios (Gerentes) e coordenadores.

O 2º nível será para os funcionários (recepcionistas e cozinheiros), que contemplará os recursos de cadastro, consultas e movimento de pedidos, clientes e carrinho e cadastro de suprimentos no caixa, não estando disponível o cadastro e edição de produtos, emissão de relatórios, cadastro e consulta de sangrias do caixa e contas a pagar, consultas de suprimentos do caixa, edição de cardápio, cadastro e edição de funcionários, administradores e terceirizados, que serão funcionalidades exclusivas do nível 1.

O 3º nível será para o cliente, que contemplará os recursos de cadastro de pedido, cadastro de usuário comum, acompanhamento de pedido, cadastro, exclusão e edição de carrinho e fechamento e pagamento de pedido, as demais funcionalidades somente estarão disponíveis para os níveis 1 e 2.

O 4º nível será destinado aos funcionários terceirizados, possuindo acesso somente à atividade de realização de entregas e visualização de pedidos via delivery.

A interface será amigável com o usuário, contendo menus de acessos as operações e avisos aos usuários quando houver ocorrência de erros.

Todas as telas terão o mesmo padrão, facilitando a comunicação

usuários/sistema. Além disso, o sistema será responsivo, sendo adaptável para as diversas resoluções que o usuário buscar.

### ***1.5 Restrições, Suposições e Dependências***

Para garantir a segurança e a integridade dos dados armazenados no sistema desenvolvido para a "Sakai Burguer", algumas medidas e requisitos técnicos precisam ser rigorosamente seguidos:

A utilização inadequada do sistema (preenchimento de informações incorretamente) poderá gerar problemas, sugere-se sempre que faça uma checagem após o preenchimento de informações, antes de salvar qualquer modificação.

Para que o sistema tenha um bom desempenho, é necessário no mínimo um processador de 2.3Ghz, HD de 256 Gb e periféricos (mouse, teclado). Não existem limites de memória para acesso ao sistema, recomenda-se no mínimo 4 Gb (RAM).

A funcionalidade completa do sistema depende da disponibilidade dos serviços de nuvem para hospedagem e armazenamento de dados. Qualquer interrupção nesses serviços pode afetar a operação do sistema.

Uma conexão de internet estável é crucial para a comunicação entre os componentes do sistema e para o acesso remoto pelos usuários autorizados.

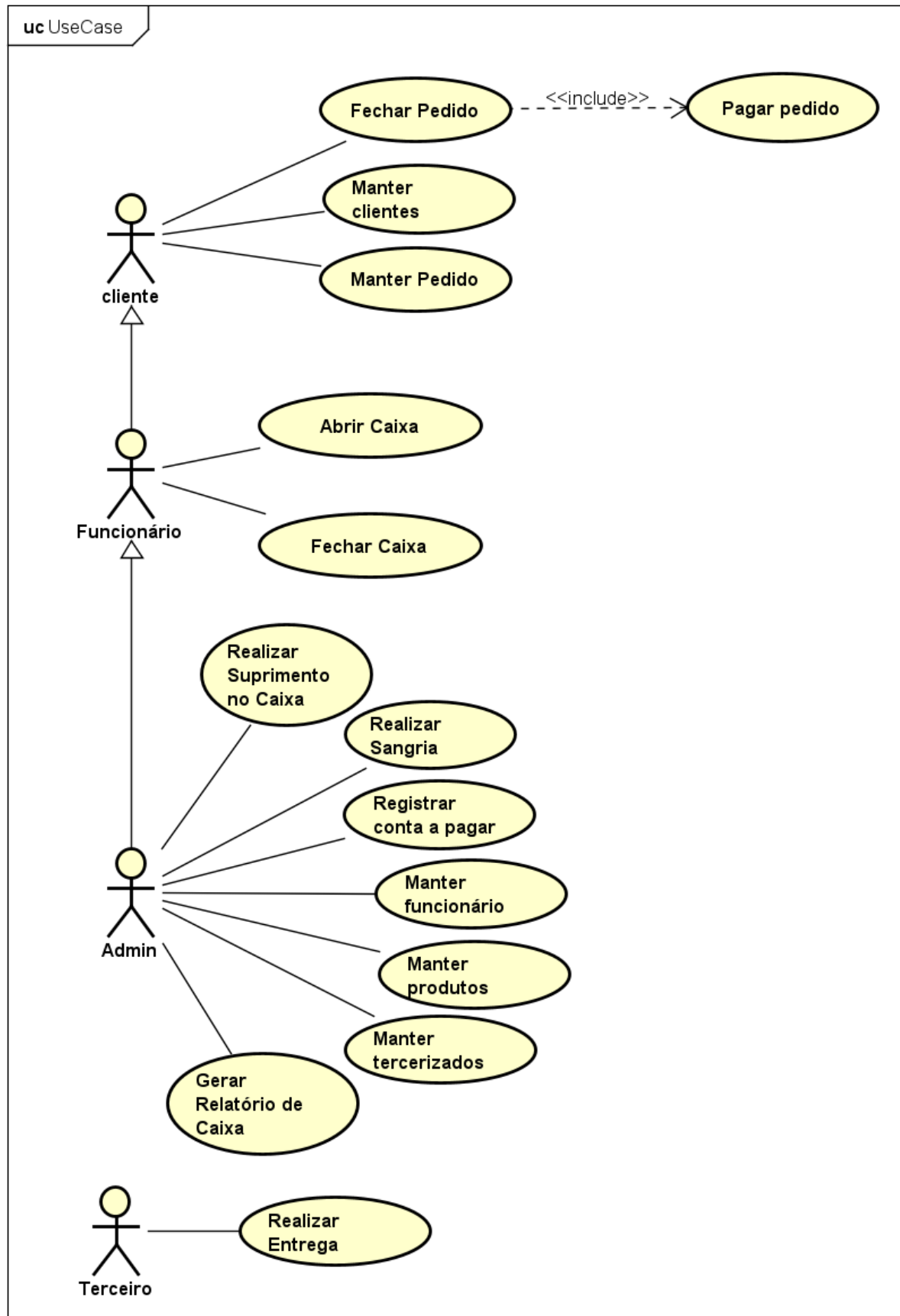
Presume-se que a manutenção regular do hardware e software será realizada para prevenir falhas e manter o sistema em funcionamento otimizado.

O cumprimento destas determinações é vital para o funcionamento eficiente e seguro do sistema desenvolvido para a "Sakai Burguer". Qualquer desvio dessas práticas pode comprometer seriamente a operação e a integridade dos dados da empresa.

## **Capítulo 2 - REQUISITOS ESPECÍFICOS**

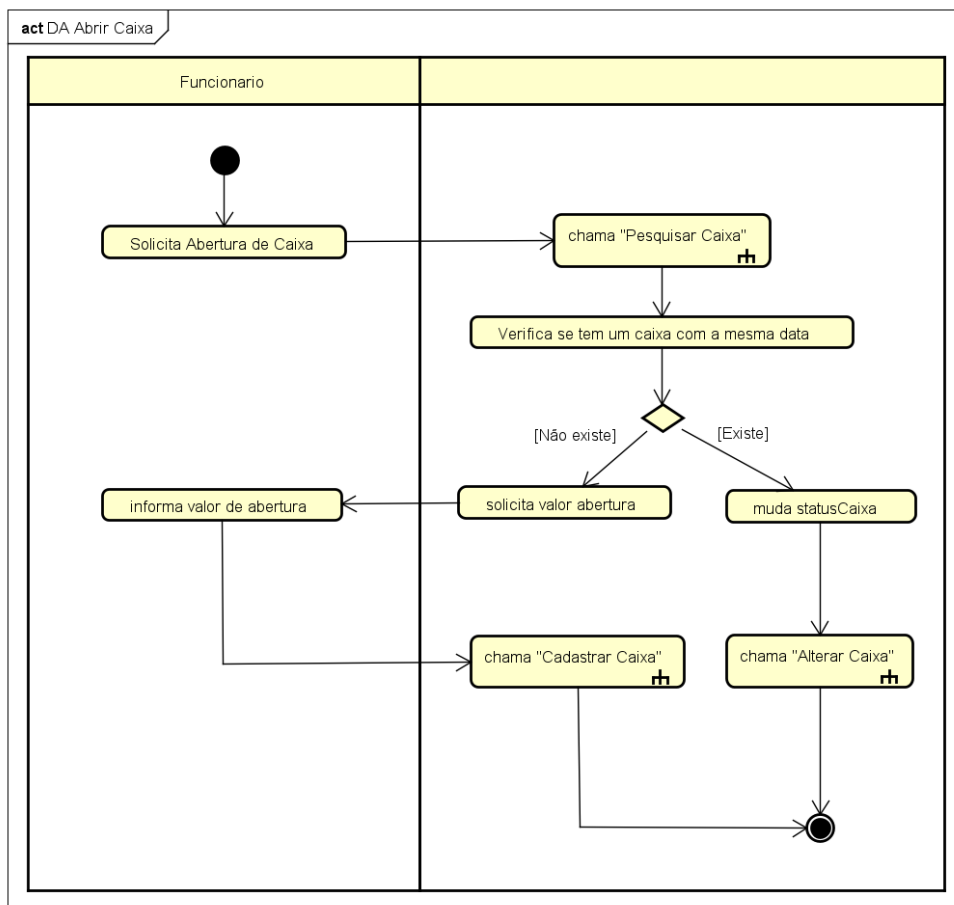


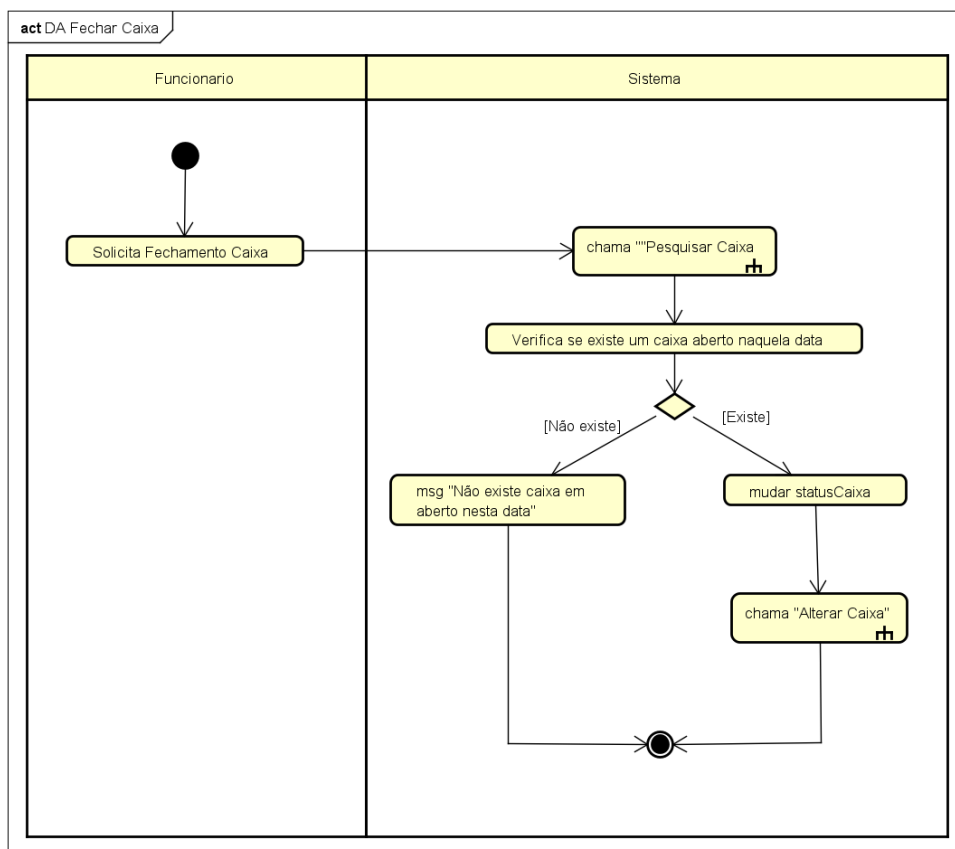
## 2.1 Diagrama de Casos de Uso

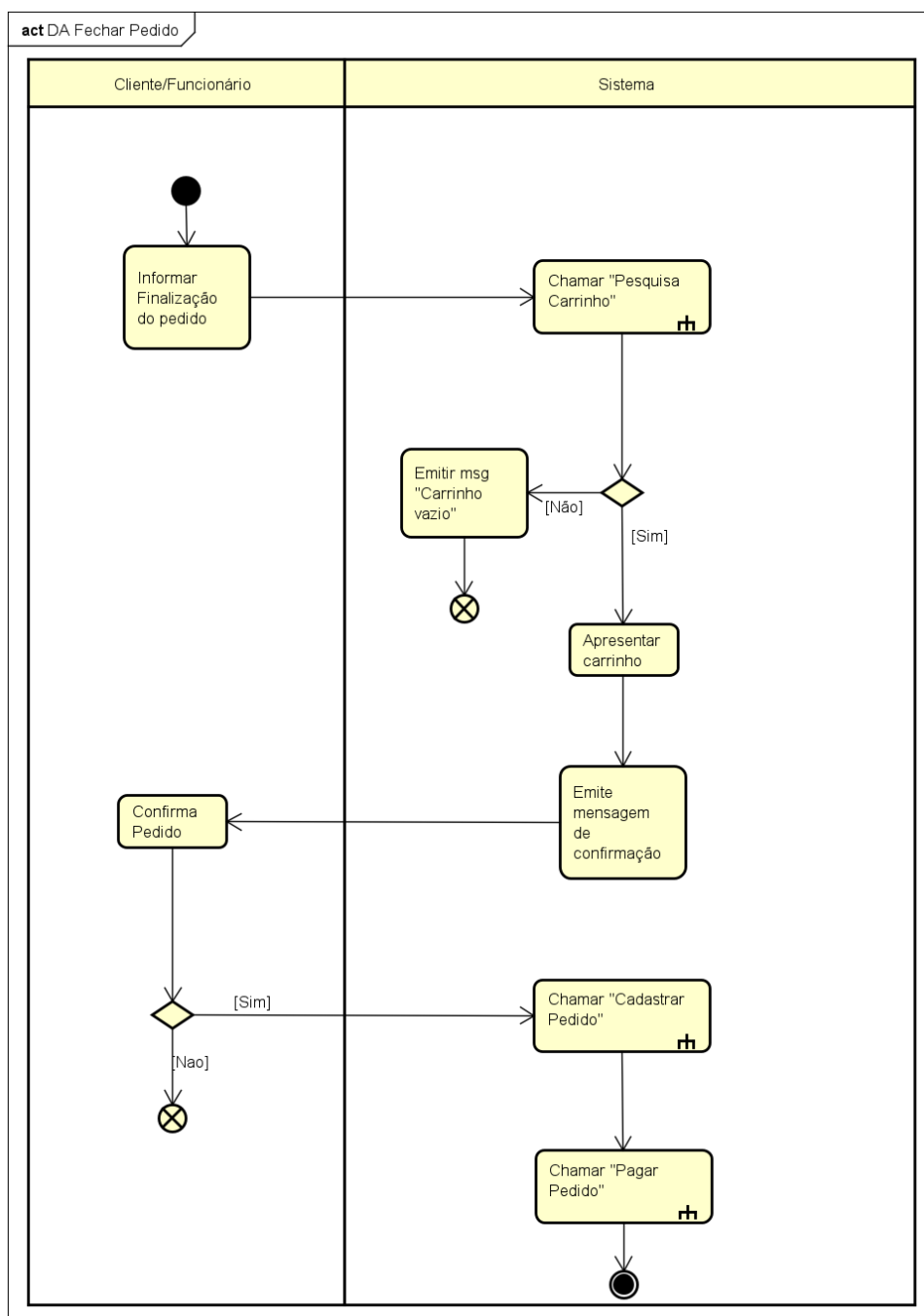


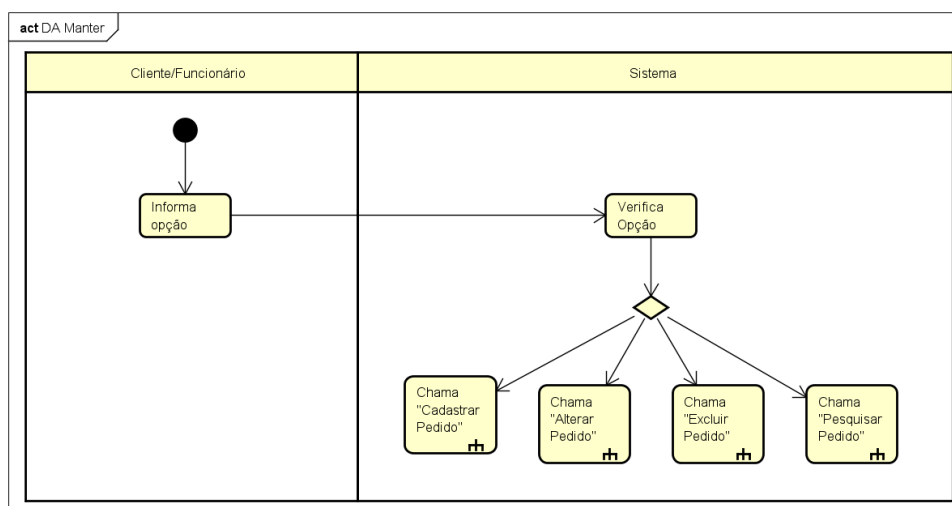
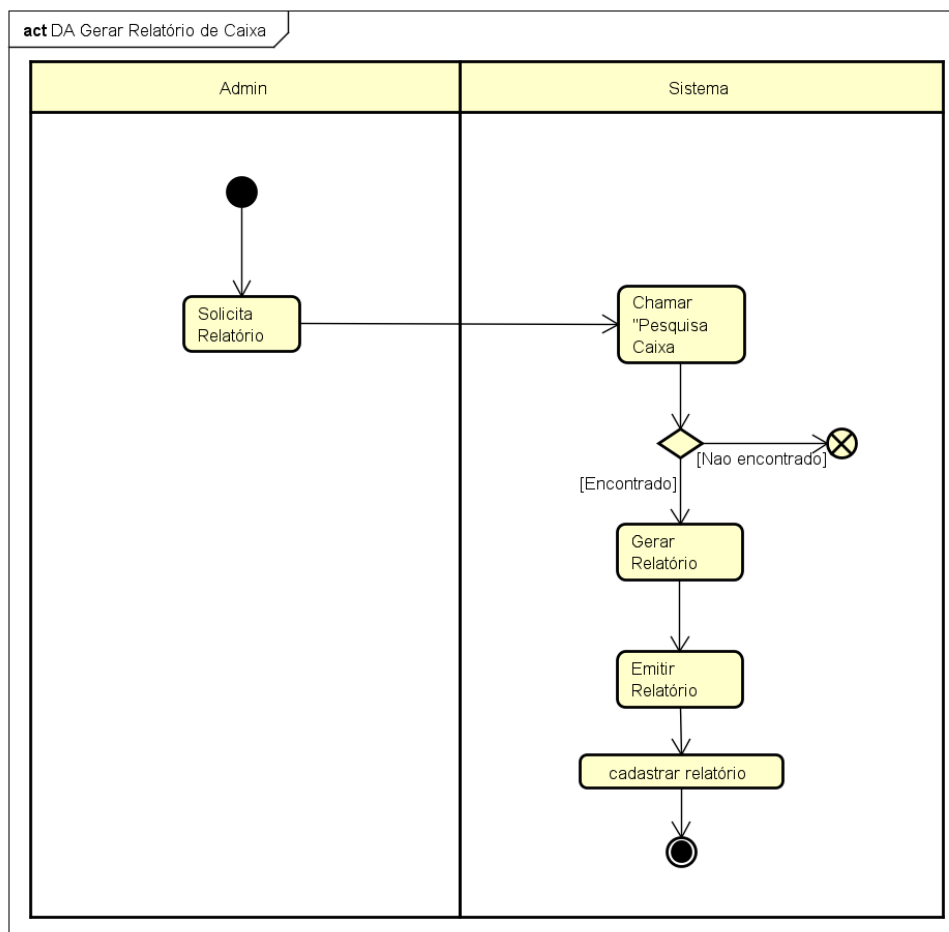
## 2.2 Especificações de Casos de Uso e Diagramas de Atividades

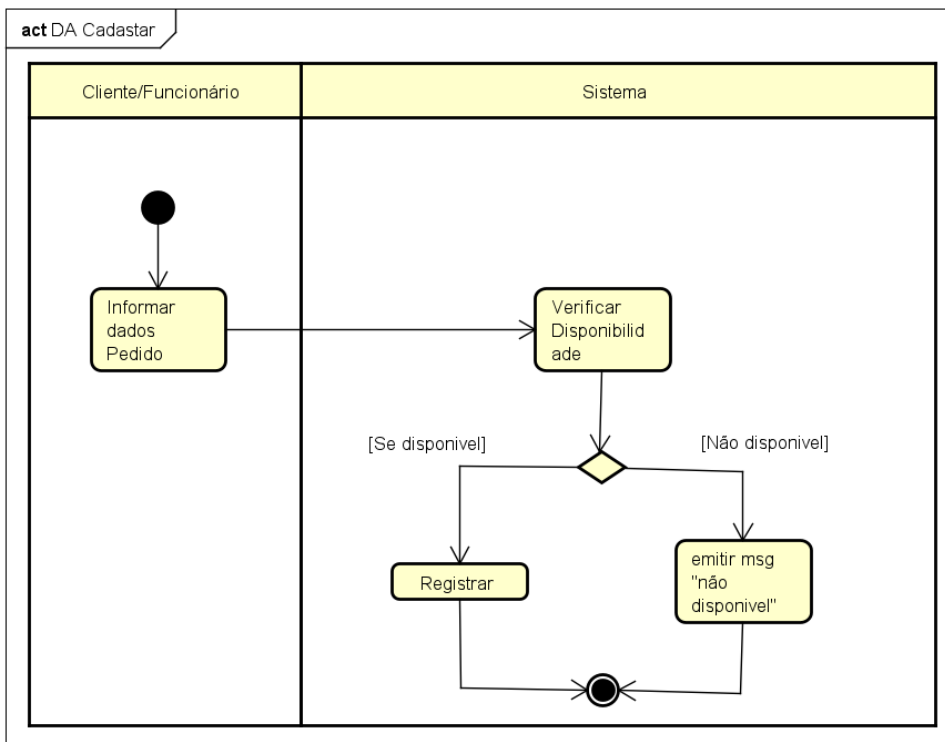
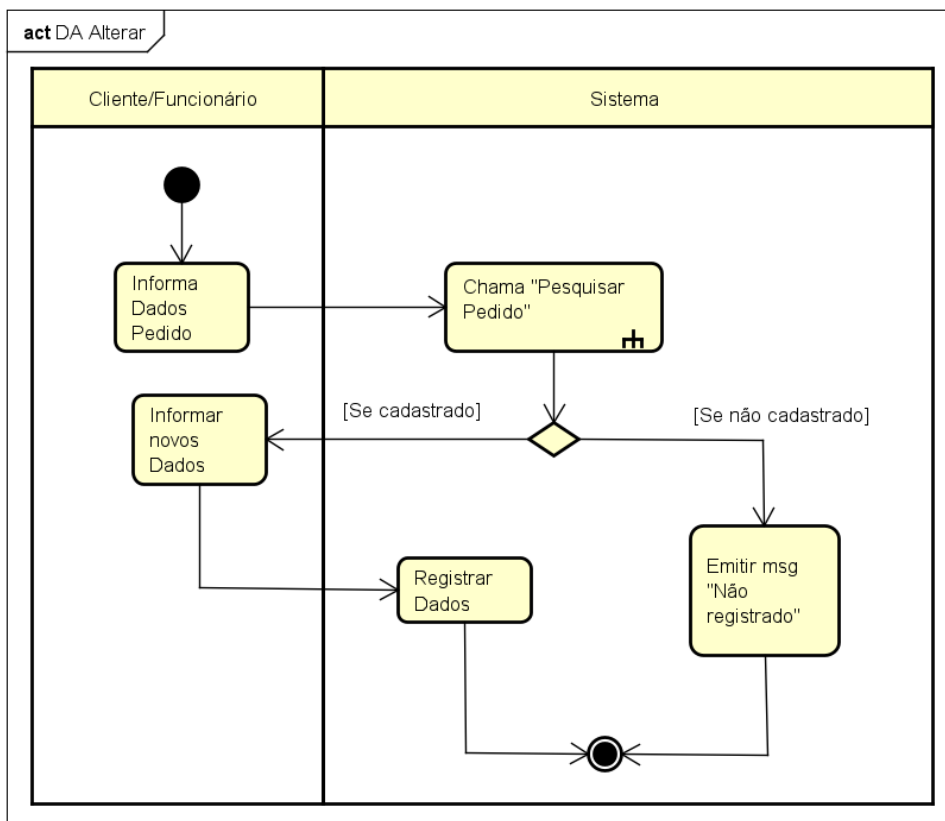
### 2.2.1 Diagramas de Atividades

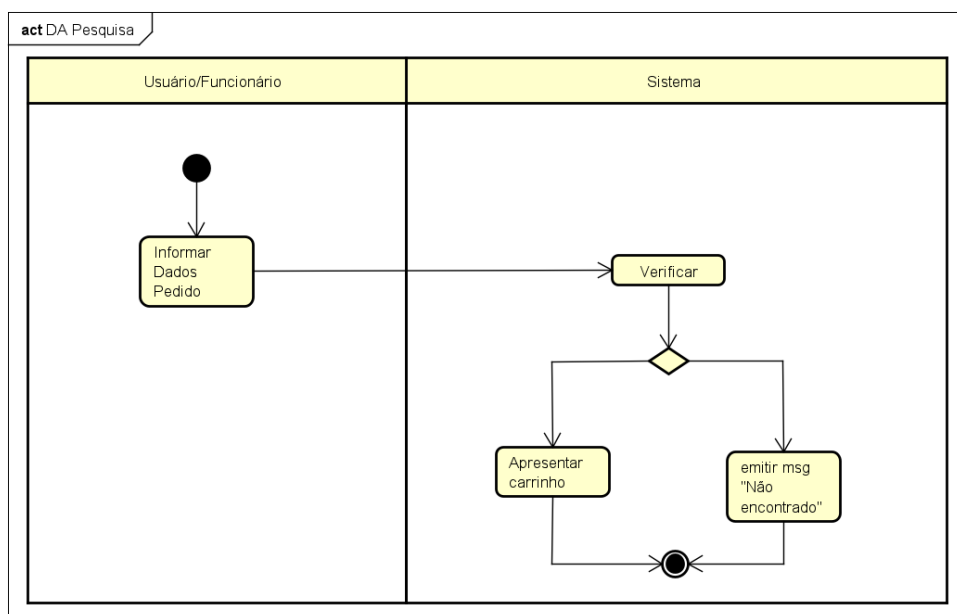
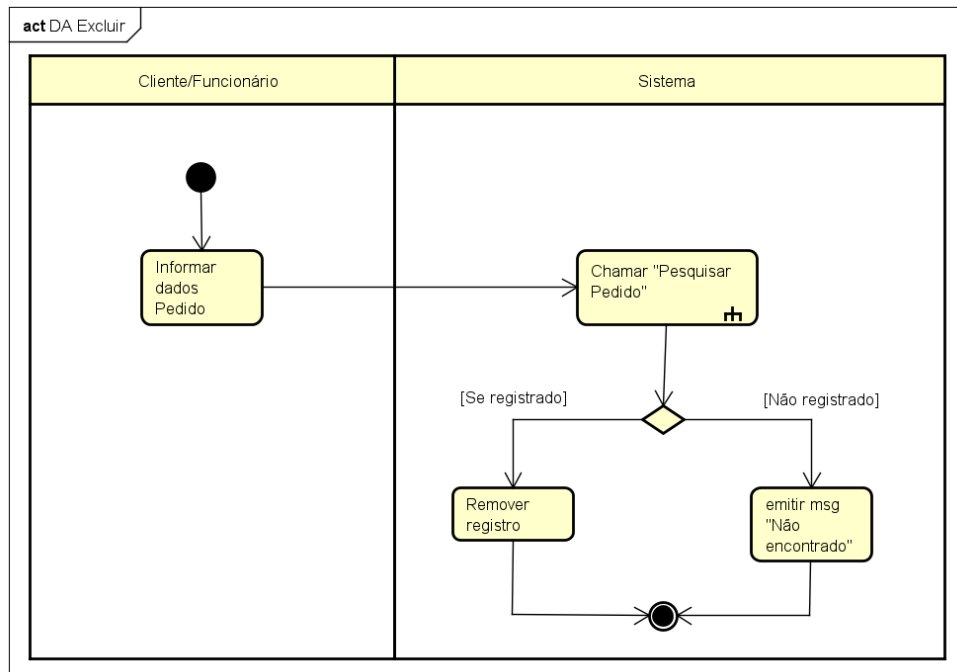


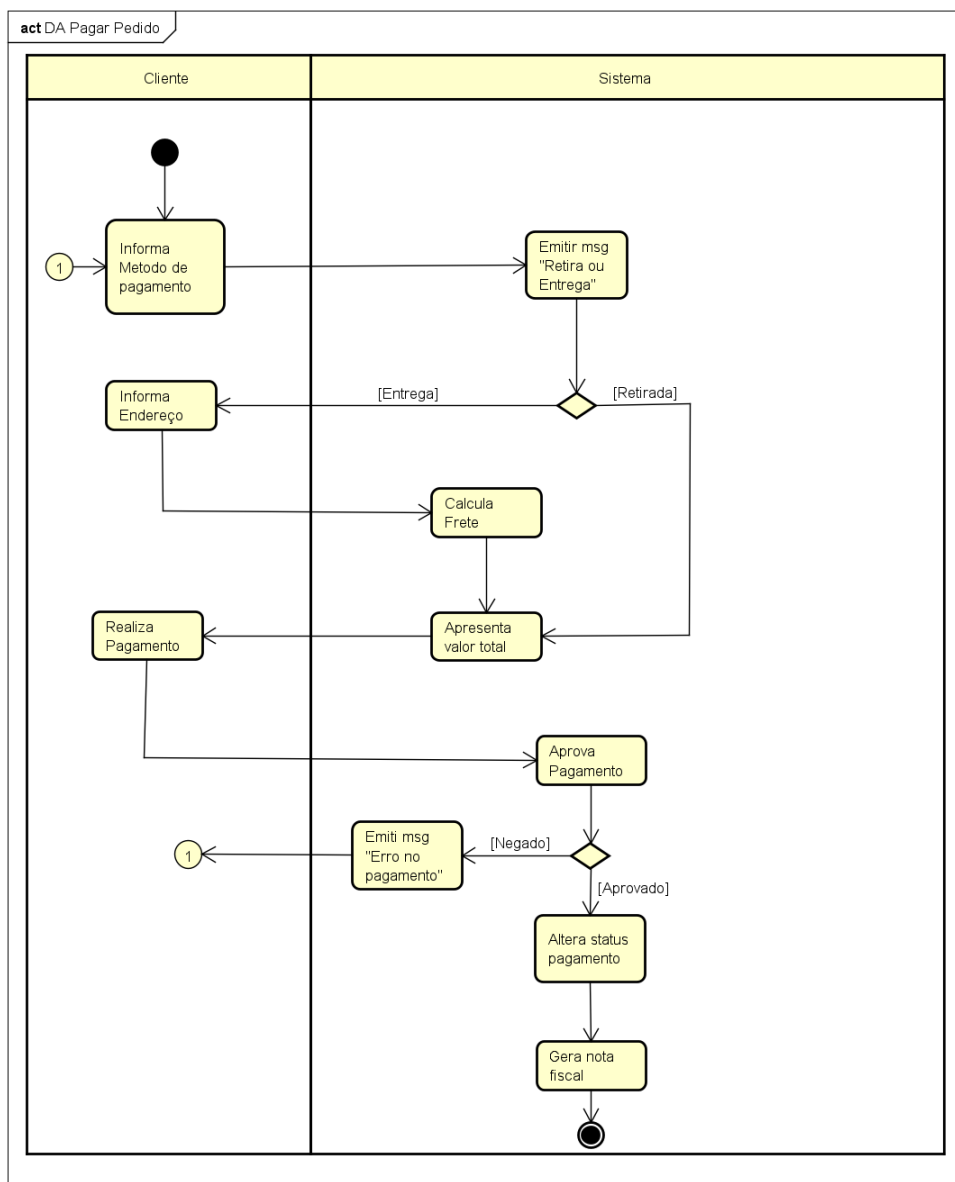




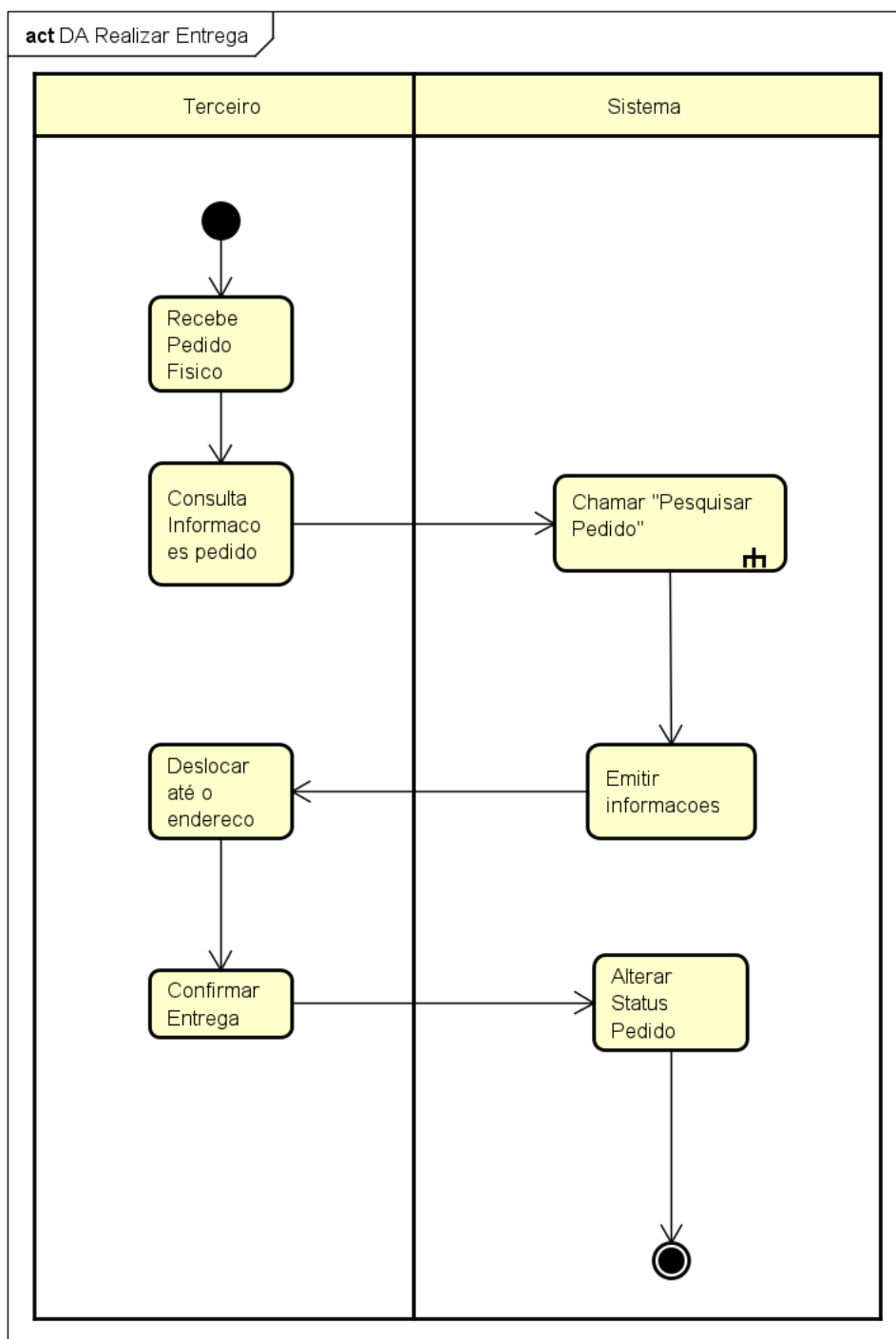




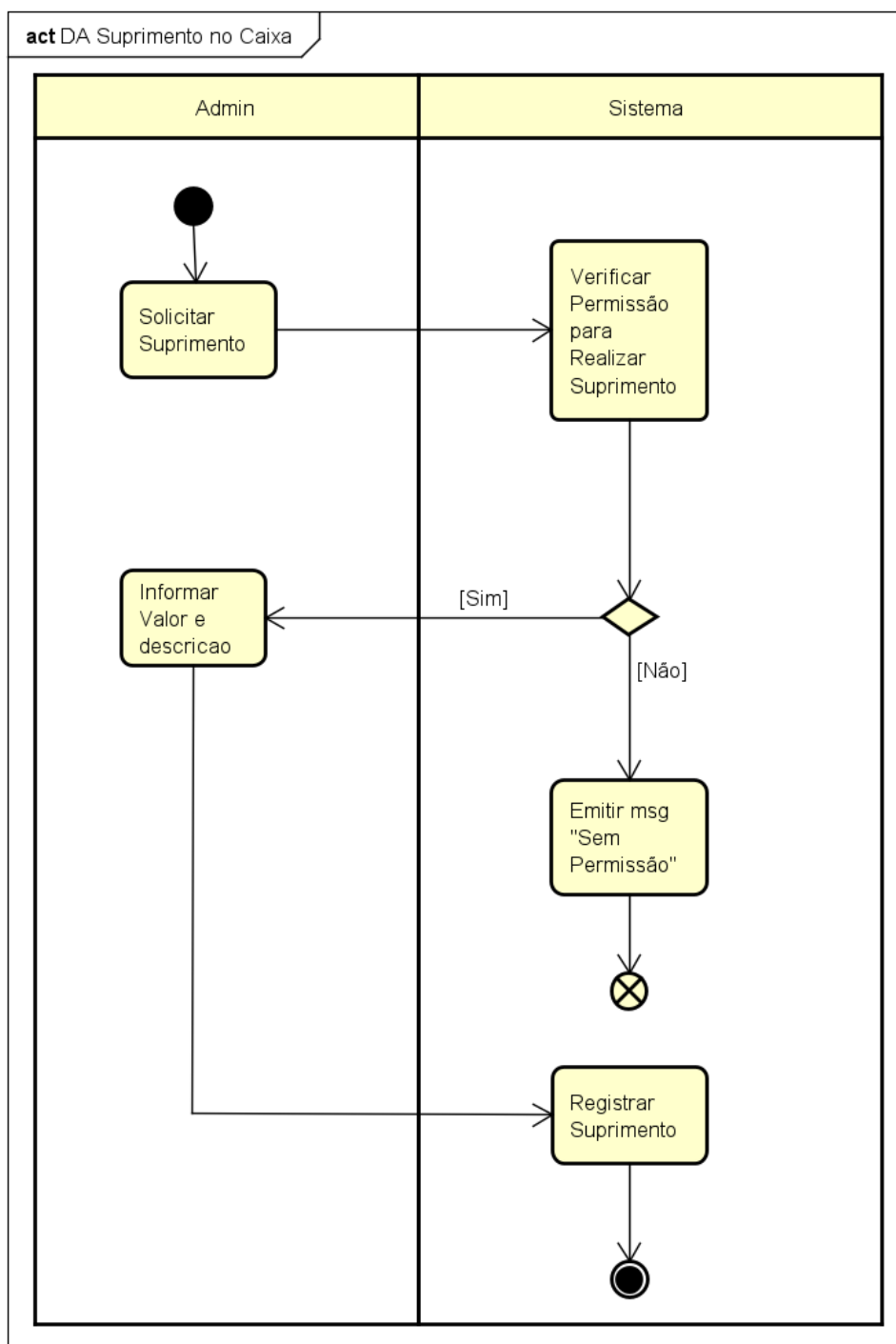


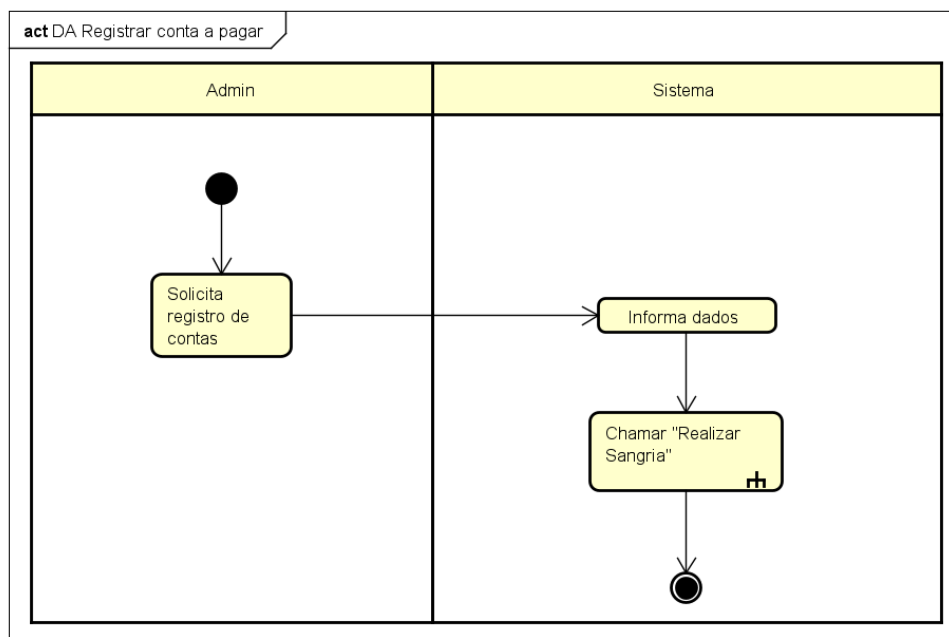


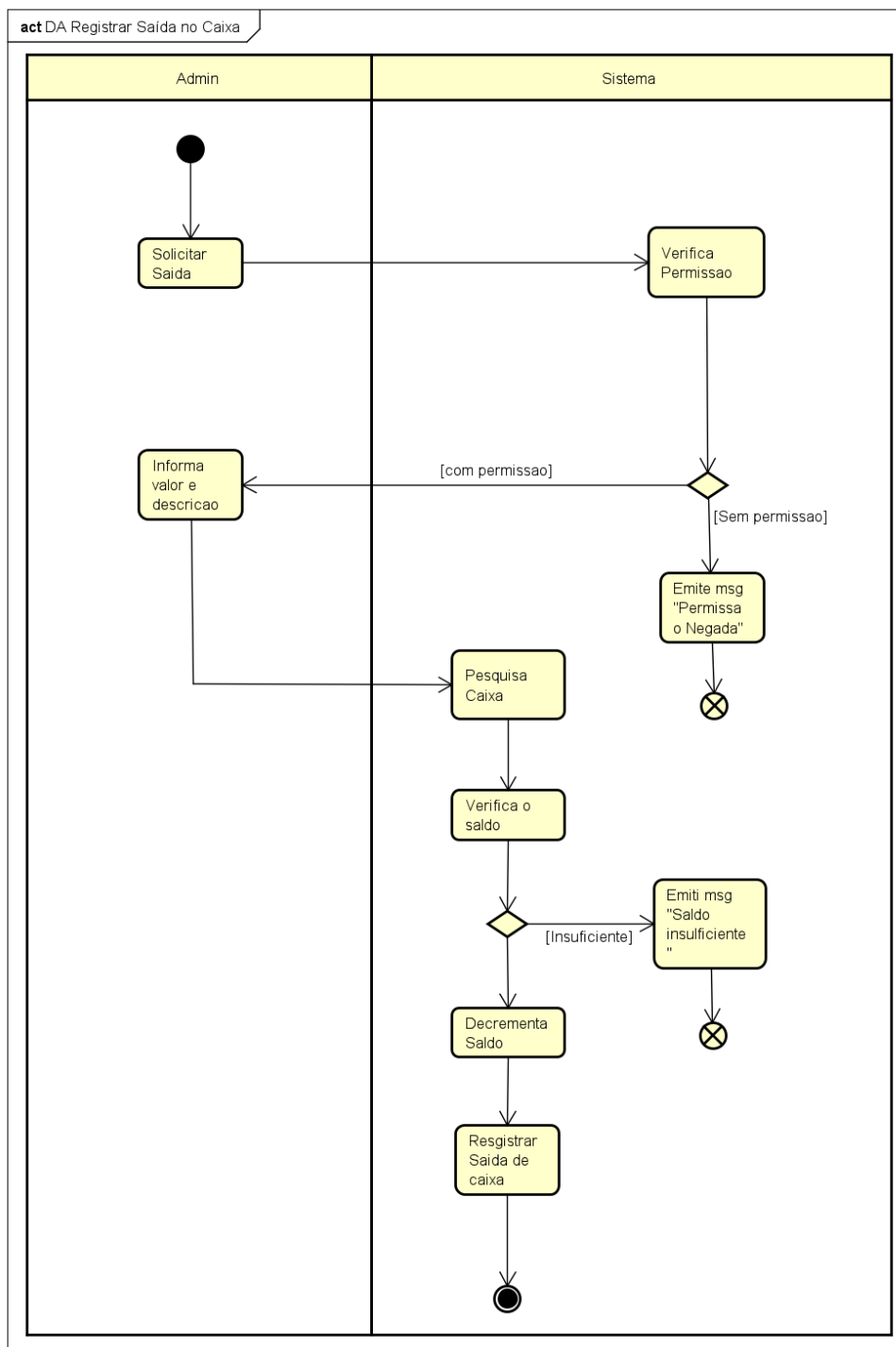












## 2.2.2 Especificações de Casos de Uso

**Caso de uso:** Fechar Pedido

**Ator Principal:** Cliente/Funcionário

**Pré-condições:** cliente autenticado no sistema, cliente ter cadastrado os produtos no carrinho

**Pós-condições:** fechamento do pedido, dados do pedido armazenado

**Fluxo principal:**

1. Este caso de uso se inicia quando o cliente deseja fechar o pedido
2. Estender ao caso de uso “pesquisa carrinho”
3. Sistema verifica existência e preenchimento do carrinho
4. Sistema lista produtos do carrinho
5. Sistema verifica confirmação do pedido
6. Estender para caso de uso “cadastrar pedido”
7. Estender para caso de uso “pagar pedido”

**Fluxo alternativo:**

2. Carrinho vazio
  - 2.1 Cancelar operação
5. Pedido não confirmado
  - 5.1 Cancelar operação

**Caso de uso:** Pagar Pedido

**Ator Principal:** Cliente

**Pré-condições:** Cliente autenticado no sistema, fechar pedido ter sido concluído

**Pós-condições:** status do pedido ser alterado para “PAGO”

**Fluxo principal:**

1. Este caso de uso se inicia quando o cliente acessa o sistema para pagar o pedido realizado
2. Cliente informa método de pagamento
3. Cliente escolhe entre: entrega ou retirada

**Seção: Entrega**

4. Sistema solicita endereço
5. Sistema calcula frete
6. Sistema apresenta o valor total
7. Cliente realiza pagamento
8. Sistema valida pagamento
9. Sistema altera status do pagamento
10. Sistema gera nota fiscal

**Seção: Retirada**

4. sistema apresenta valor total
5. Cliente realiza pagamento
6. Sistema valida pagamento
7. Sistema altera status do pagamento
8. Sistema gera nota fiscal

**Fluxo alternativo:**

8. Pagamento não aprovado

- 8.1 Sistema exibe mensagem de erro
- 8.2 Retornar ao passo 2

**Caso de uso:** Gerar relatório de caixa

**Ator Principal:** Administrador

**Pré-condições:** Administrador autenticado no sistema

**Pós-condições:** Relatório armazenado e emitido

**Fluxo principal:**

1. Este caso de uso se inicia quando o administrador acessa o sistema para gerar a emissão de relatório de caixa
2. Administrador solicita o relatório
3. Estender para caso de uso “pesquisa caixa”
4. Sistema gera relatório
5. Sistema emite relatório
6. Sistema cadastrar relatório

**Fluxo alternativo:**

3. Caixa não encontrado
- 3.1 Cancelar operação

**Caso de uso:** Registrar conta a pagar

**Ator Principal:** Administrador

**Pré-condições:** Administrador autenticado no sistema

**Pós-condições:** dados da conta paga registrados armazenados

**Fluxo principal:**

1. Este caso de uso se inicia quando o administrador acessa o sistema para registrar contas
2. Administrador informa dados da conta
3. Estender para o caso de uso “registrar saída de caixa”
4. Sistema registra conta paga

**Fluxo alternativo:**

**Caso de uso:** Realizar Entrega

**Ator Principal:** Terceiro (entregador)

**Pré-condições:** Pedido fechado no sistema.

**Pós-condições:** status do pedido ser alterado para “ENTREGUE”

**Fluxo principal:**

1. Este caso de uso se inicia quando o entregador recebe o pedido físico e o Status do pedido está “PRONTO PARA ENTREGA”
2. Entregador solicita endereço
3. Sistema fornece endereço de entrega
4. Entregador se desloca até o endereço
5. Entregador confirma a entrega
6. Sistema muda o status do pedido para “ENTREGUE”

**Fluxo alternativo:**

4. Endereço não localizado, sistema muda status do pedido para “Tentativa de entrega”.

**Caso de uso:** Realizar Sangria

**Ator Principal:** Administrador

**Pré-condições:** Administrador autenticado no sistema, saldo disponível no caixa

**Pós-condições:** Saldo atualizado

**Fluxo principal:**

1. Este caso de uso se inicia quando o administrador acessa o sistema para realizar uma sangria
2. O administrador informa o valor a ser retirado
3. Sistema verifica saldo disponível
4. Sistema registrar valor do saque
5. Sistema atualiza saldo do caixa
6. Sistema registra Sangria

**Fluxo alternativo:**

3. Saldo Insuficiente.
  - 3.1 Administrador retorna ao passo 2 e escolhe outro valor.
  - 3.2 Administrador cancela operação

**Caso de uso:** Abrir Caixa

**Ator Principal:** Funcionário

**Pré-condições:** Funcionário autenticado no sistema

**Pós-condições:** Caixa com Status Aberto

**Fluxo principal:**

1. Este caso de uso se inicia quando o funcionário solicita a abertura de um caixa
2. Sistema pesquisa se existe um caixa com a mesma data
3. Sistema solicita o valor de abertura



4. Funcionário informa valor de abertura
5. Estender para caso de uso “Cadastrar Caixa”
6. Sistema finaliza operação

**Fluxo alternativo:**

- 3.1 Sistema muda Status Caixa para Aberto
- 3.2 Estender para caso de uso “Alterar Caixa” e finaliza Operação

**Caso de uso:** Fechar Caixa

**Ator Principal:** Funcionário

**Pré-condições:** Funcionário autenticado no sistema

**Pós-condições:** Caixa com Status Fechado

**Fluxo principal:**

1. Este caso de uso se inicia quando o funcionário solicita o fechamento de um caixa
2. Sistema pesquisa se existe um caixa aberto naquela data
3. Sistema muda status Caixa para Fechado
4. Estender para caso de uso “Alterar Caixa”
5. Sistema finaliza Operação

**Fluxo alternativo:**

- 3.1 Sistema Manda mensagem “Não existe caixa em aberto nesta data” e finaliza operação

**Caso de uso:** Realizar Suprimento de Caixa

**Ator Principal:** Administrador

**Pré-condições:** Administrador autenticado no sistema

**Pós-condições:** Saldo do Caixa atualizado

**Fluxo principal:**

1. Este caso de uso se inicia quando o administrador solicita saída de caixa
2. Sistema verifica permissão
3. Funcionário informa valor e descrição
4. Sistema incrementa o valor

**Fluxo alternativo:**

- 2.1 Sistema emite mensagem “Sem permissão” e finaliza a operação.

Este padrão é utilizado para a documentação dos requisitos de operações de manutenção em sistemas de informação, por meio do uso de modelos e especificações de casos de uso. Os requisitos de operações de manutenção são caracterizados por operações de Inclusão, Consulta, Alteração e Exclusão.

**Fluxo básico**

1. O caso de uso inicia quando o Cliente necessita fazer a manutenção (inclusão, alteração, exclusão ou consulta) de um pedido.
2. De acordo com o tipo de operação de manutenção desejado pelo cliente, um dos subfluxos é executado:
  - a. Se o cliente deseja incluir novo pedido, o subfluxo incluir pedido é executado.
  - b. Se o cliente deseja alterar informações de um pedido já cadastrado, o subfluxo alterar pedido é executado.
  - c. Se o cliente deseja excluir um pedido já cadastrado, o subfluxo remover pedido é executado.
  - d. Se o cliente deseja consultar informações sobre um ou mais pedidos cadastrados, o subfluxo consultar pedido é executado.

**Subfluxo Incluir Pedido**

1. Este subfluxo inicia quando o cliente solicita incluir um pedido;
2. O sistema solicita ao cliente o preenchimento dos seguintes atributos:
  - Produtos\*
  - Quantidade\*
  - Método de Entrega\*
  - Método de Pagamento\*
  - Data do Pedido\*
  - Dados do Cliente\*
  - Observação
3. O cliente preenche os atributos anteriores e confirma a inclusão;
4. O sistema realiza a inclusão dos dados informados pelo cliente no passo 3;
5. O sistema exibe uma mensagem informando que a inclusão do pedido foi efetivada com sucesso;

### **Subfluxo Alterar Pedido**

1. Este subfluxo inicia quando o cliente solicita alterar um pedido;
2. O cliente seleciona um único pedido;
3. O sistema solicita a alteração dos atributos:
  - Produtos
  - Quantidade
  - Método de Entrega
  - Método de Pagamento
  - Data do Pedido
  - Dados do Cliente
  - Observação
4. O cliente altera os dados desejados e confirma a alteração;
5. O sistema realiza a alteração dos dados informados no passo 4;
6. O sistema exibe uma mensagem de confirmação informando que a alteração do pedido foi efetivada com sucesso;

### **Subfluxo Remover Pedido**

1. Este subfluxo inicia quando o cliente solicita remover um ou mais pedidos;
2. O cliente seleciona quais pedidos deseja remover e solicita a remoção;
3. O sistema solicita a confirmação para remoção;
4. O cliente confirma a remoção;
5. O sistema remove os pedidos confirmados;
6. O sistema exibe uma mensagem informando que a remoção dos pedidos foi efetivada com sucesso;

### **Subfluxo Consultar Pedido**

1. Este subfluxo inicia quando o cliente solicita consultar um pedido;
2. O sistema solicita o preenchimento dos seguintes filtros:
  - Produto
  - Método de Entrega
  - Método de Pagamento
  - Data do Pedido
  - Cliente
3. O cliente preenche os filtros e solicita a consulta;

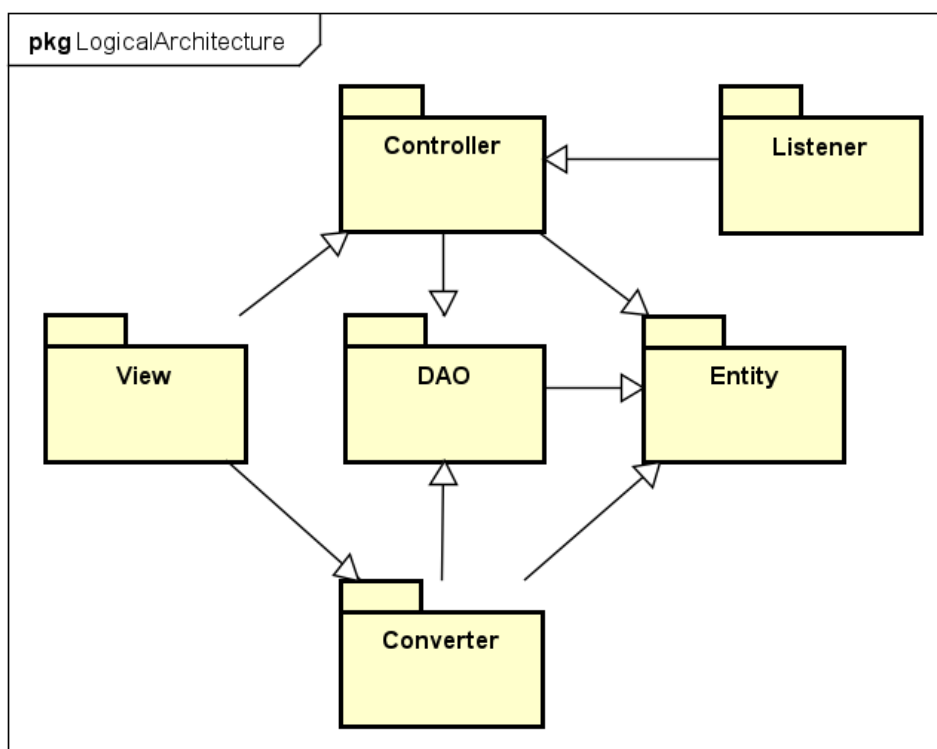
4. O sistema apresenta as seguintes informações dos pedidos obtidos na consulta:
- Produtos, quantidade (cada produto), observação, valor (soma de todos os itens do pedido), data do pedido, método de entrega, método de pagamento, cliente.

#### **Validações e regras de negócio**

- Esta regra se aplica a todos os subfluxos. Atributos obrigatórios. Se algum atributo obrigatório não tiver sido preenchido, o sistema não completará a operação e notificará ao cliente, solicitando o preenchimento;
- Esta regra se aplica a todos os subfluxos. Atributos com valores não permitidos. Se algum atributo for preenchido com valor não permitido, o sistema não completará a operação e notificará ao cliente, solicitando o preenchimento;
- No subfluxo **remover**, o sistema valida os pedidos selecionados de acordo com as seguintes regras:
  - o *Caso o pedido esteja pago, porém não tenha sido entregue ainda.*

## Capítulo 3 - PROJETO DE SOFTWARE

### 3.1 Arquitetura Lógica de Software



**View (Camada de apresentação):** É responsável pela interface e interação com o cliente, exibindo e recebendo informações. Interage com a camada de controle para obter e apresentar os dados, possui interação com a camada "**converter**" para casos especiais.

**Controller (camada e controle):** É responsável por traduzir as ações do usuário em operações de negócio, recebendo as ações do cliente por meio da camada de apresentação e utiliza da camada de acesso ao banco de dados para executar métodos.

**DAO (camada de acesso ao banco de dados):** Responsável pela comunicação com o banco de dados, realiza operações de persistência (CRUD). Interage com a camada de entidade para manipular os dados

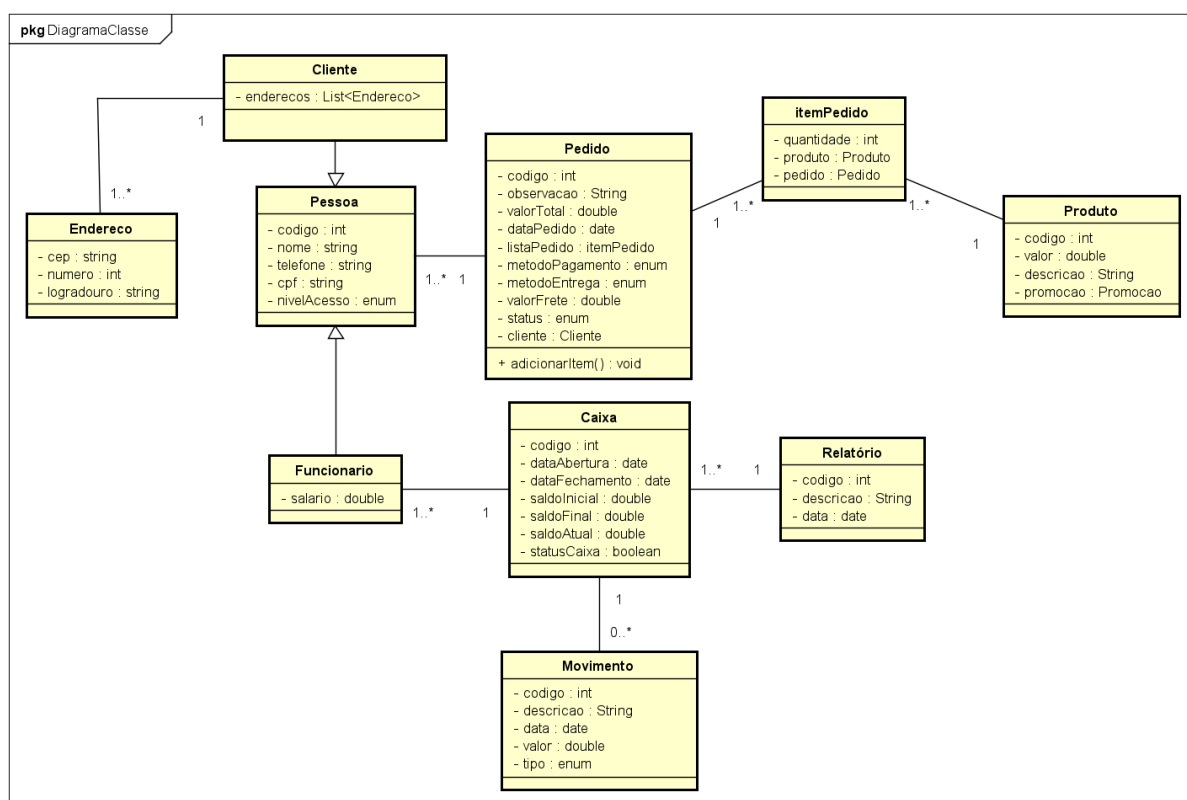
**Entity (camada de entidade):** Representa as entidades no sistema, possui os atributos relacionados a cada entidade.

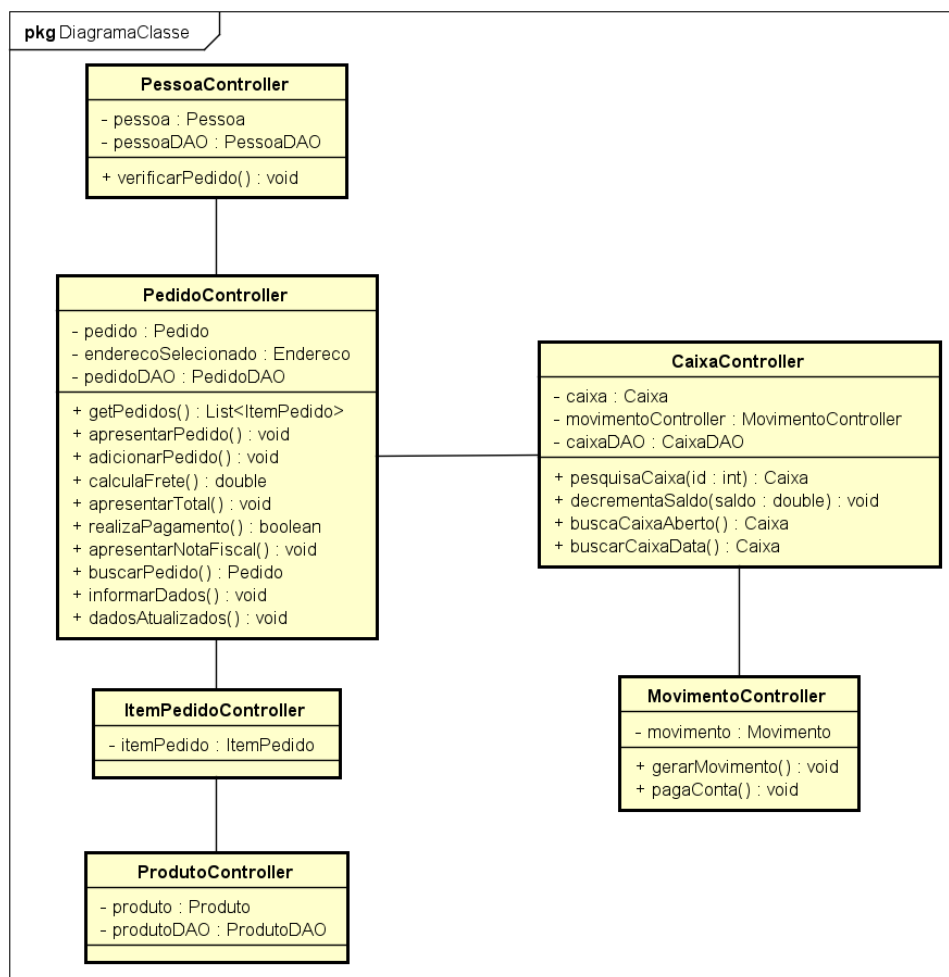
**Converter (camada de conversão):** Usado para casos especiais, converte

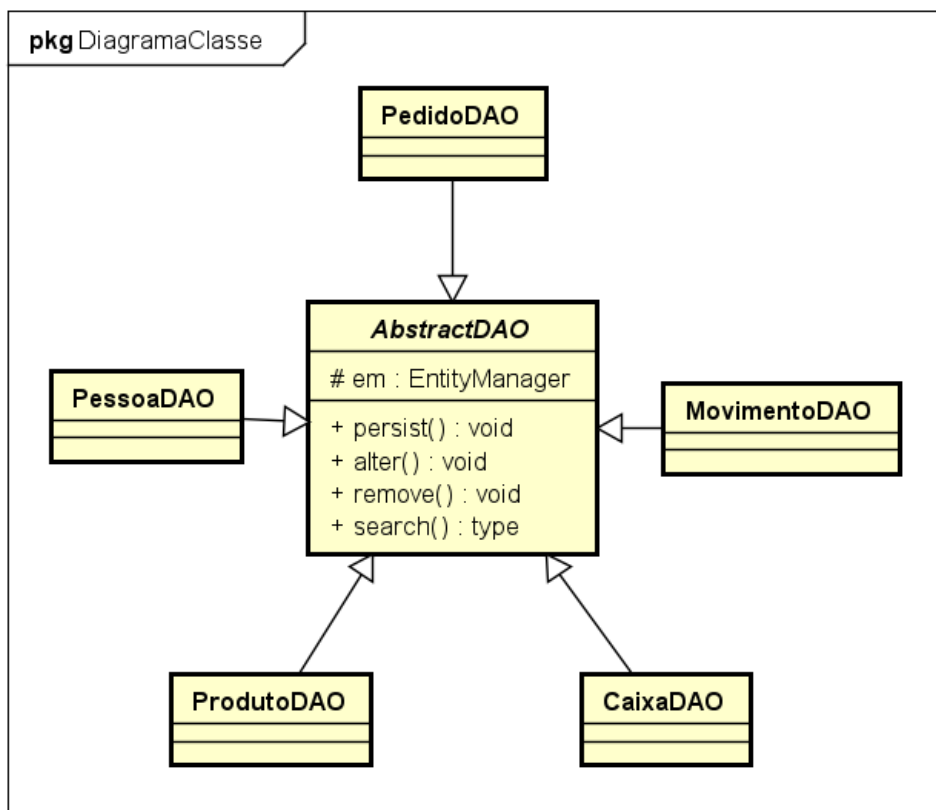
dados de um formato de persistência (**DAO**) para um formato de apresentação (**view**)

**Listener:** Recebe e processa eventos externos, como mensagens. Interage com a camada de controle para notificar eventos.

### 3.2 Diagrama de Classes

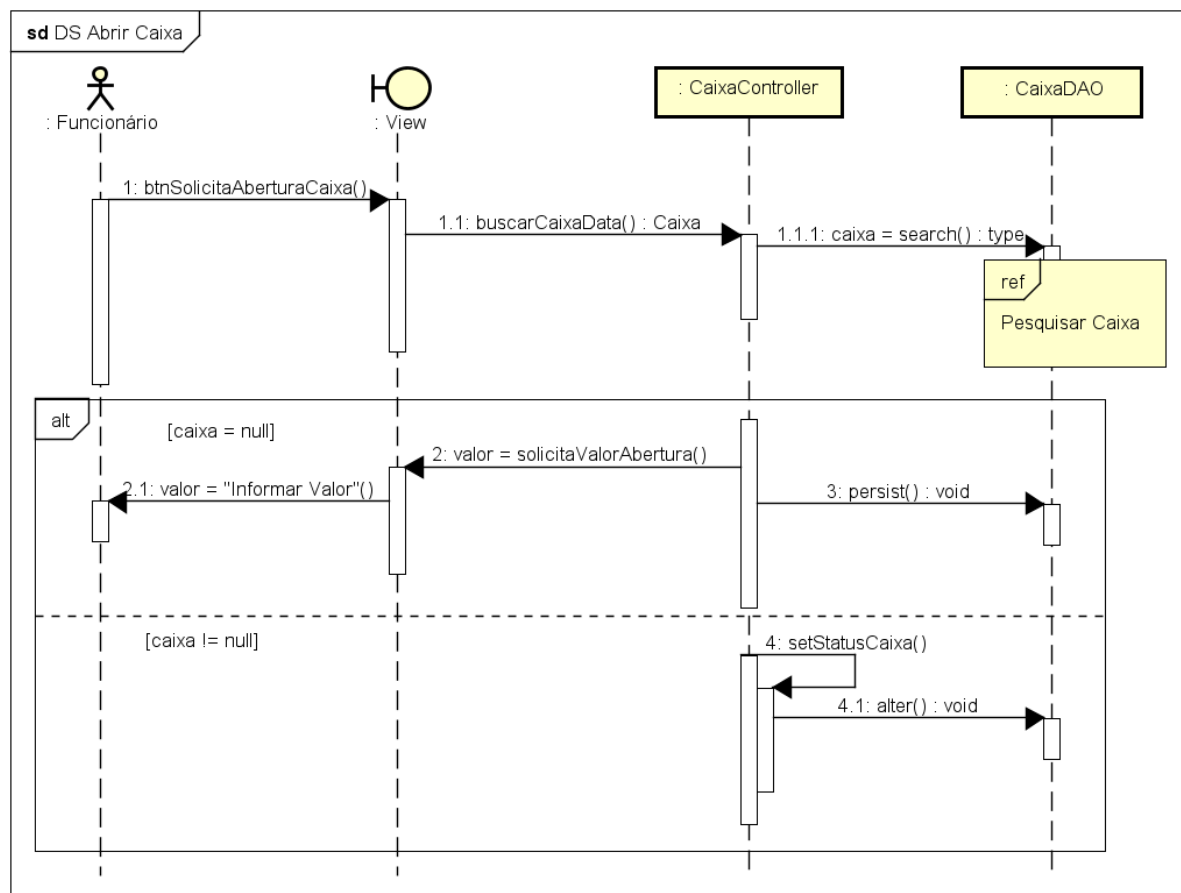


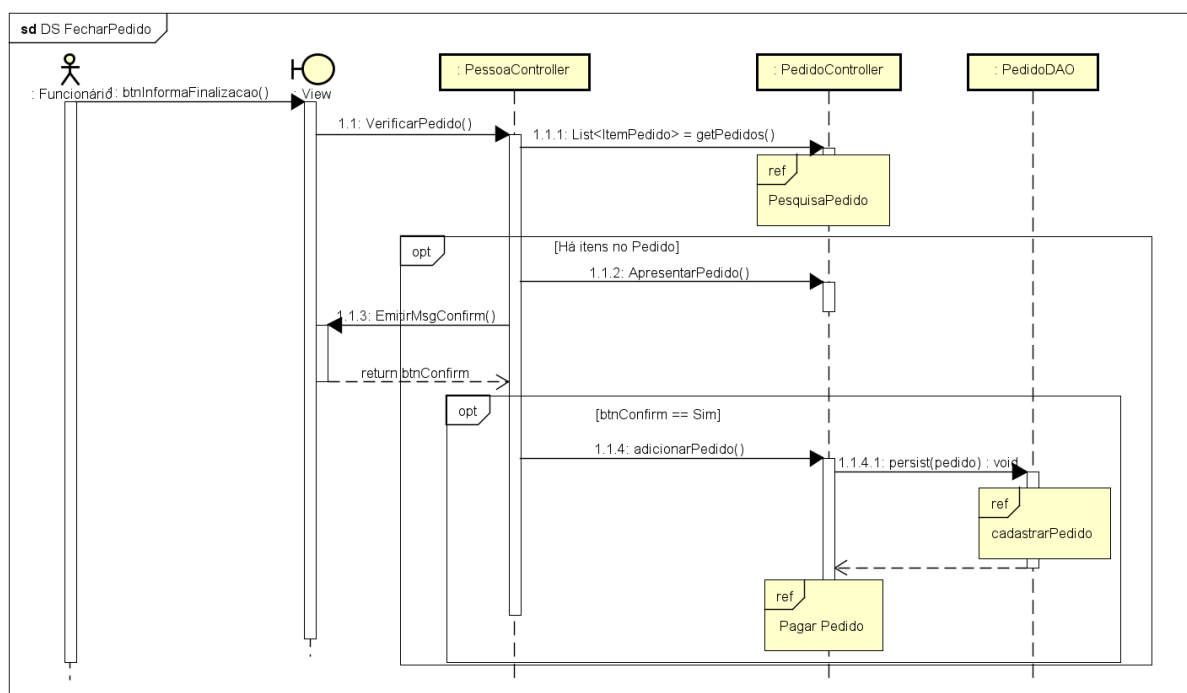
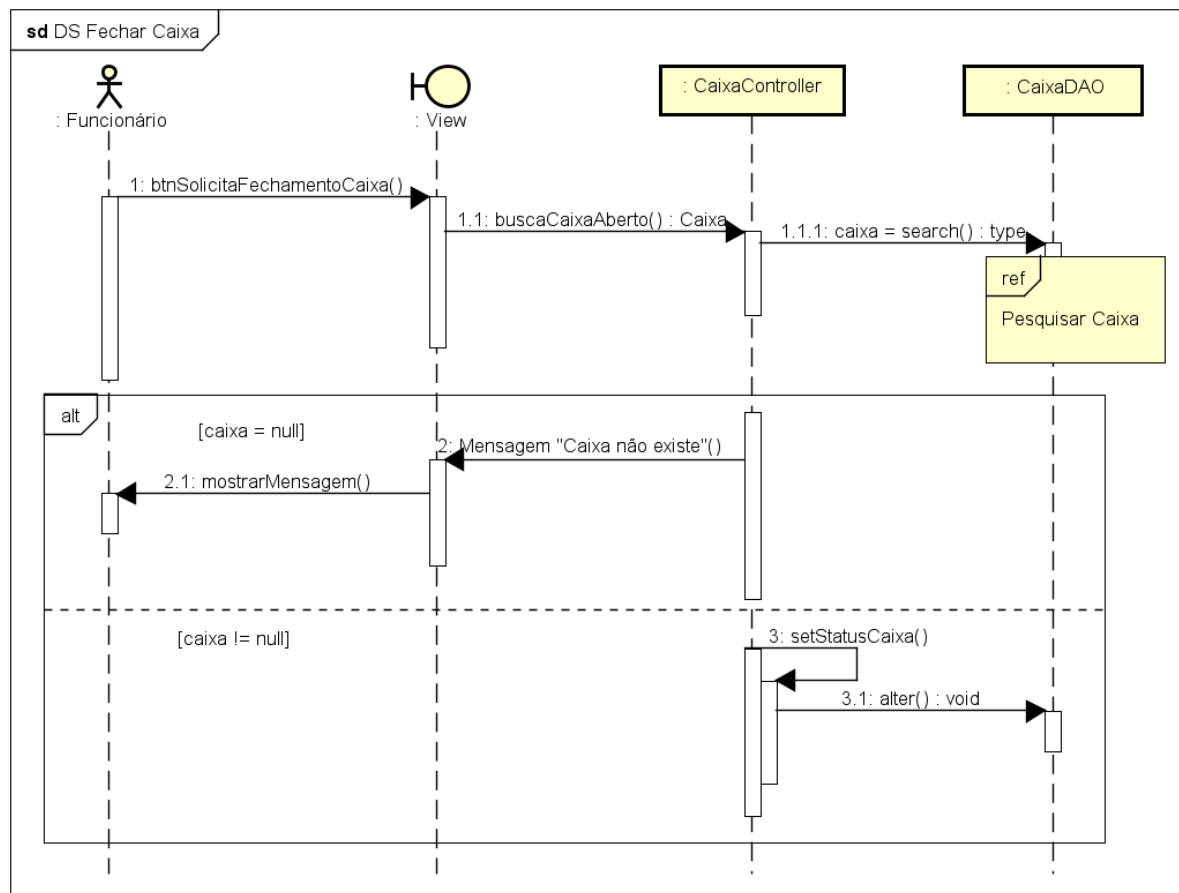


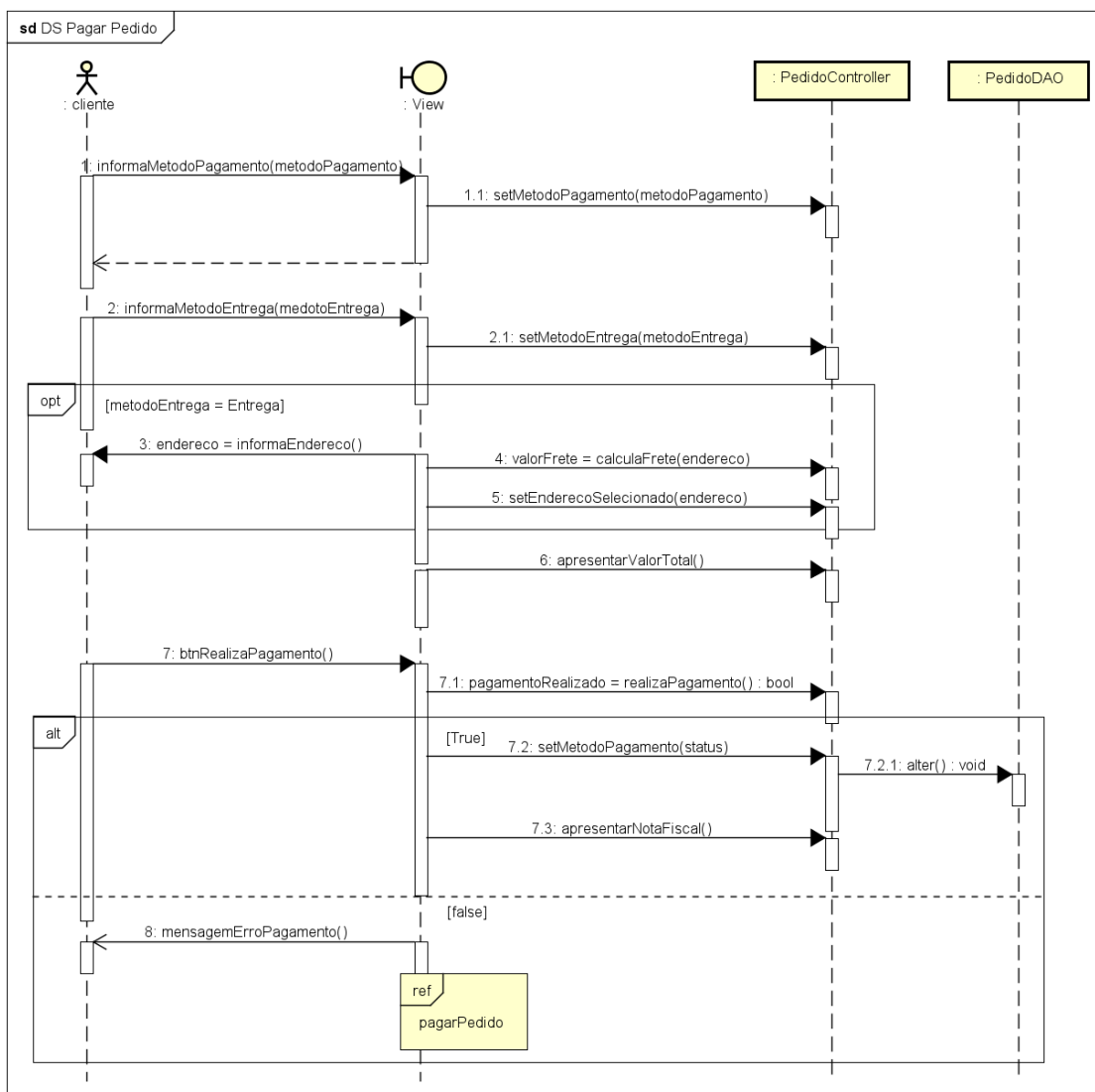
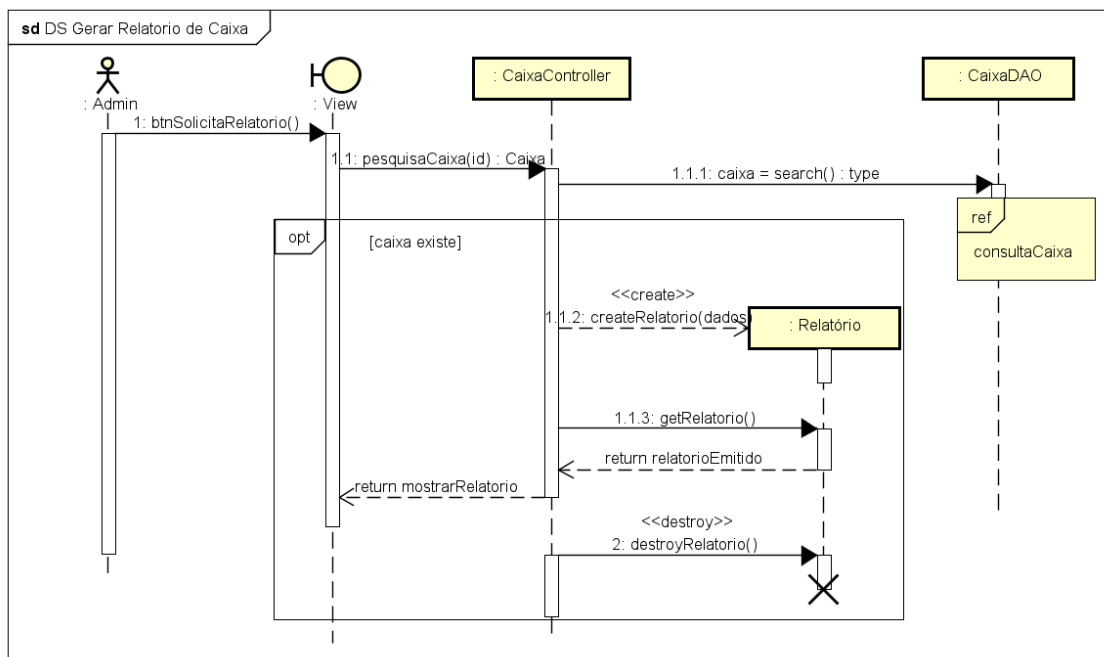


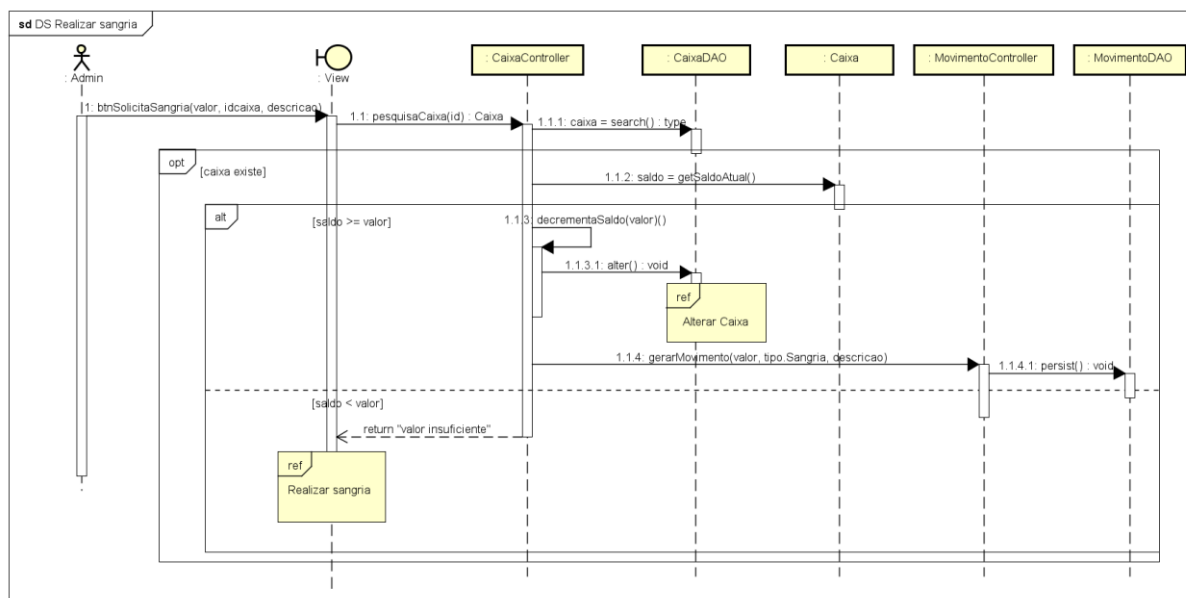
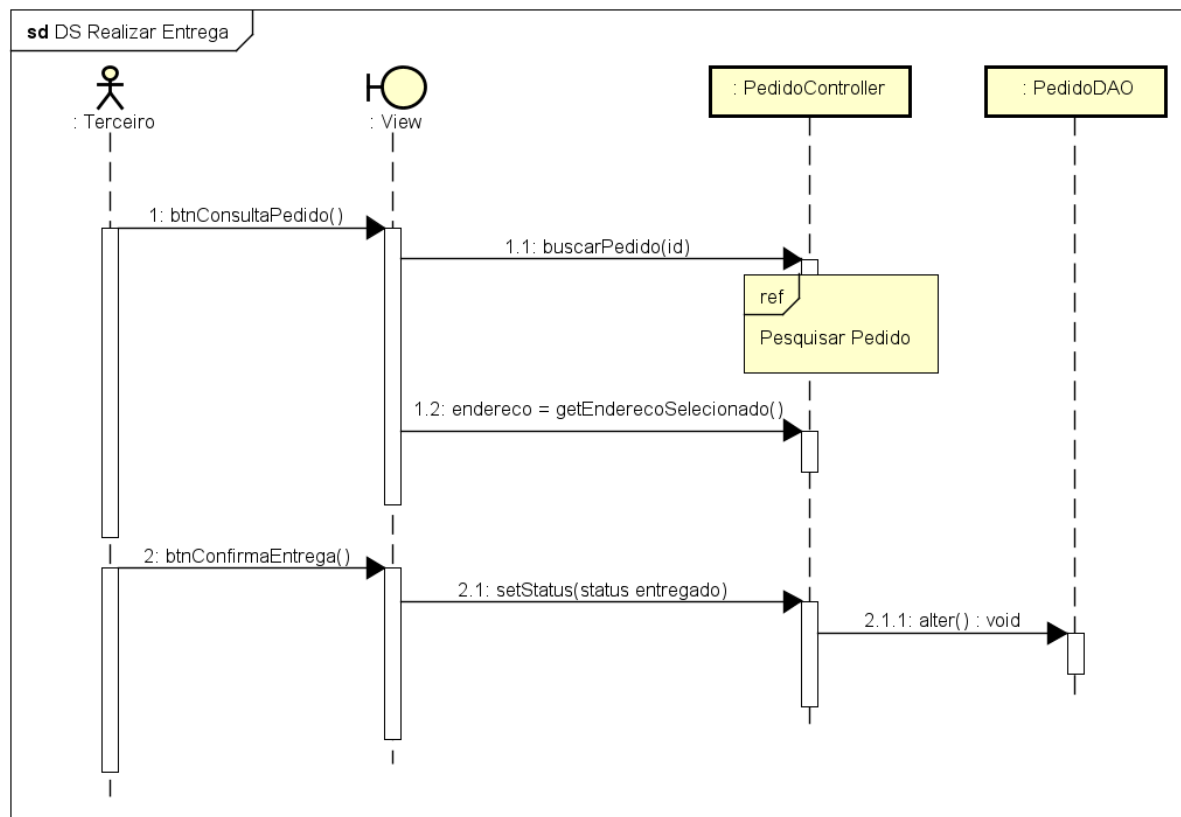


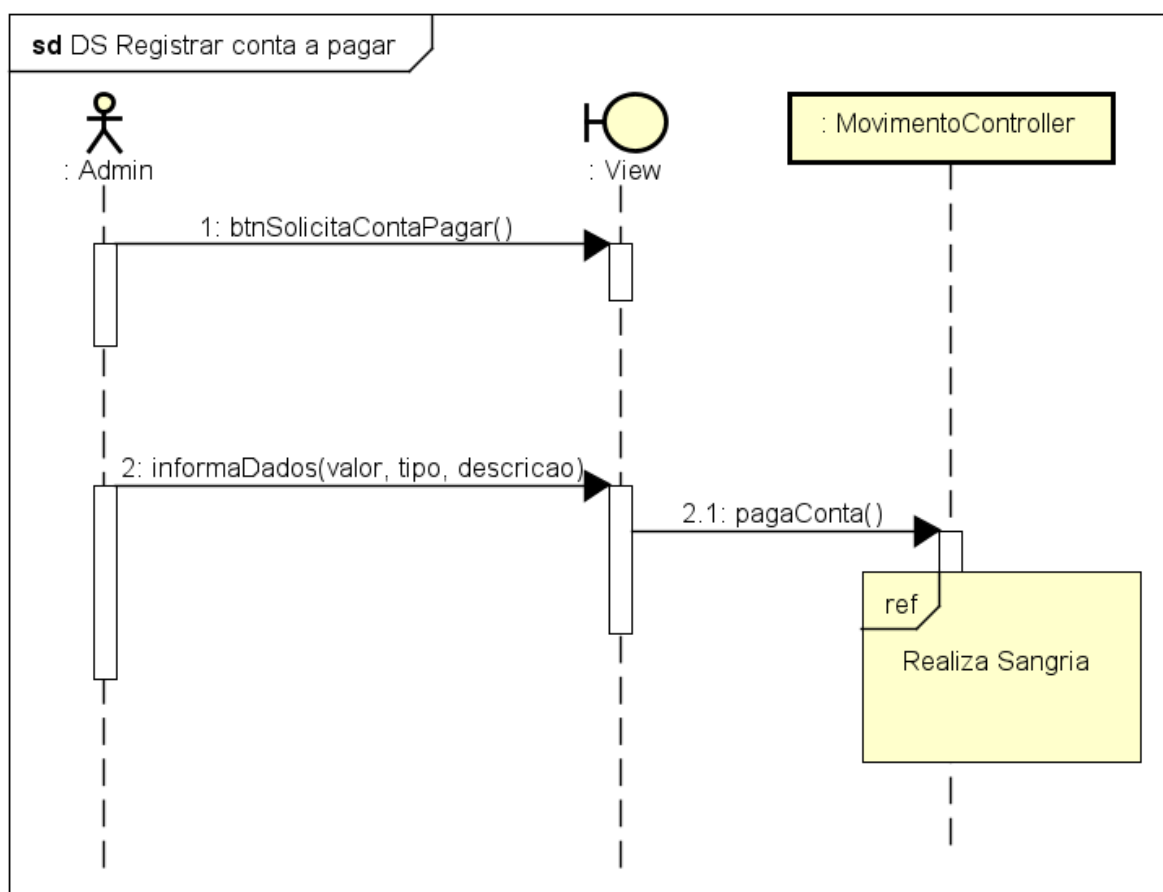
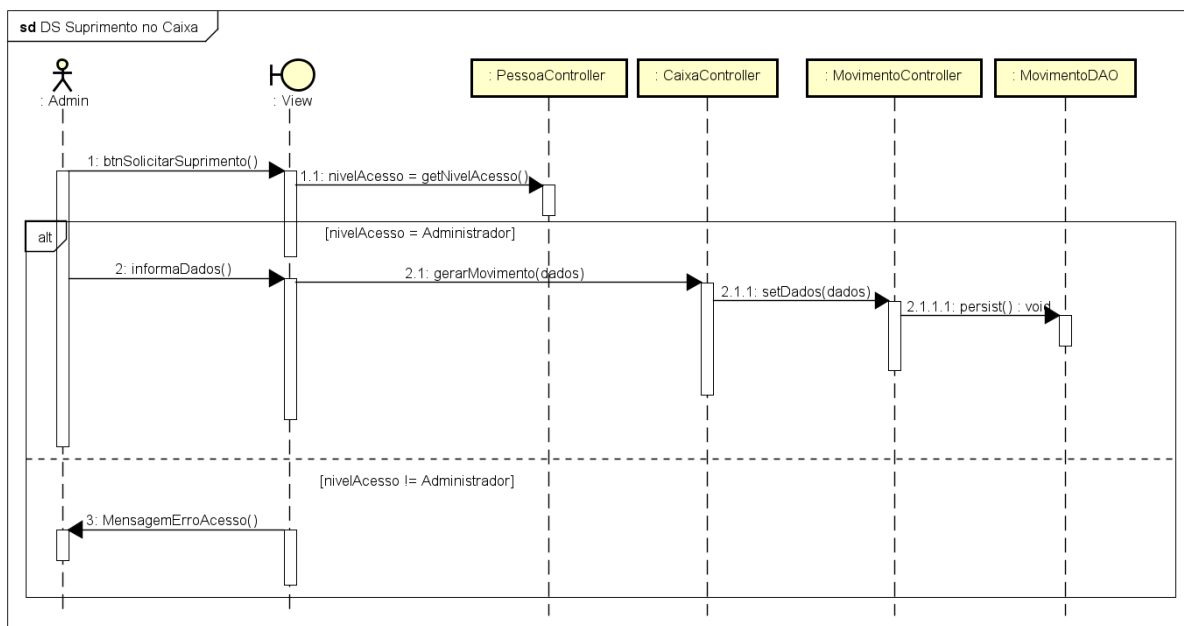
### 3.3 Diagramas de Sequência

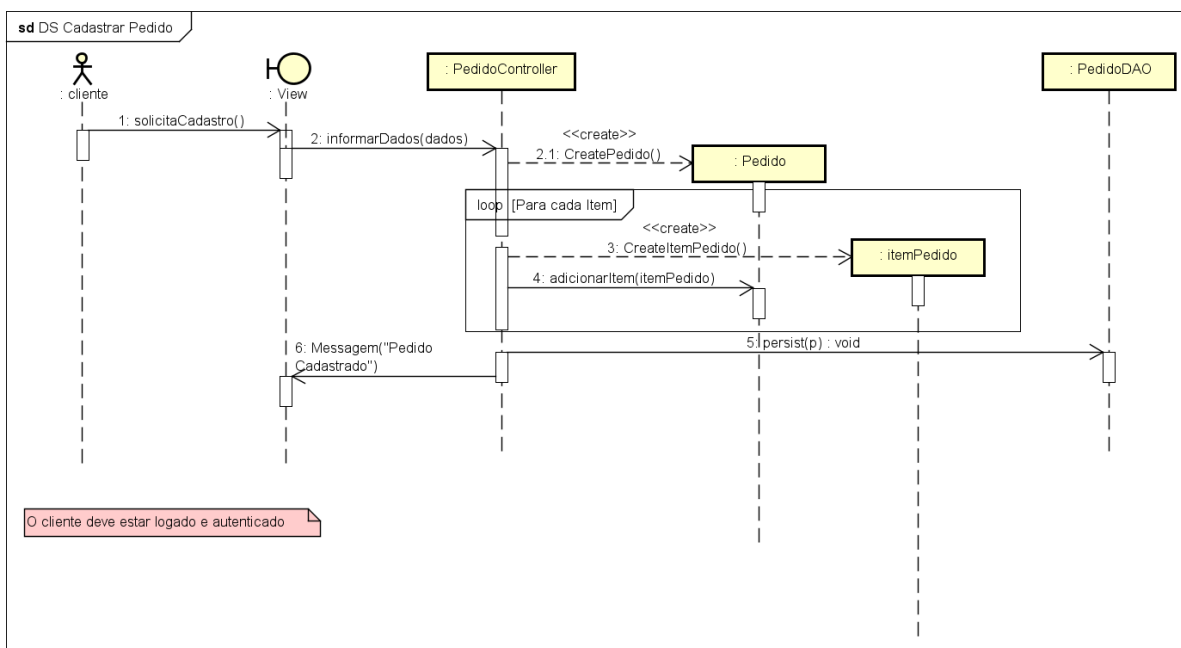
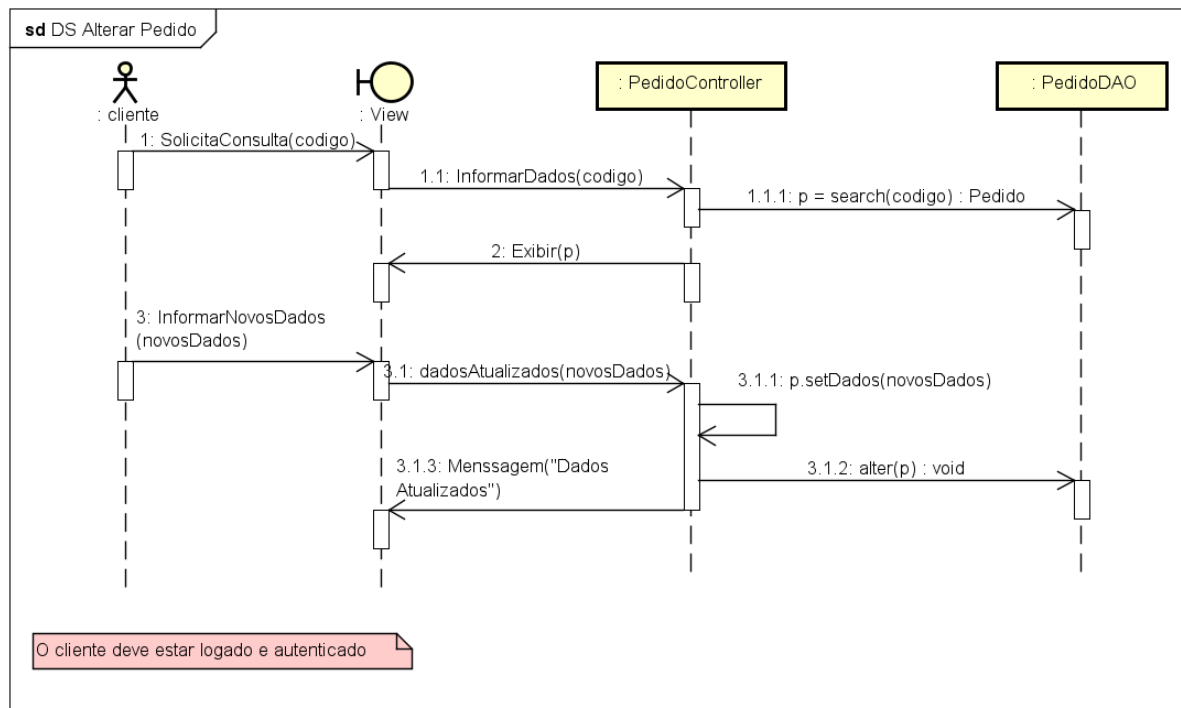


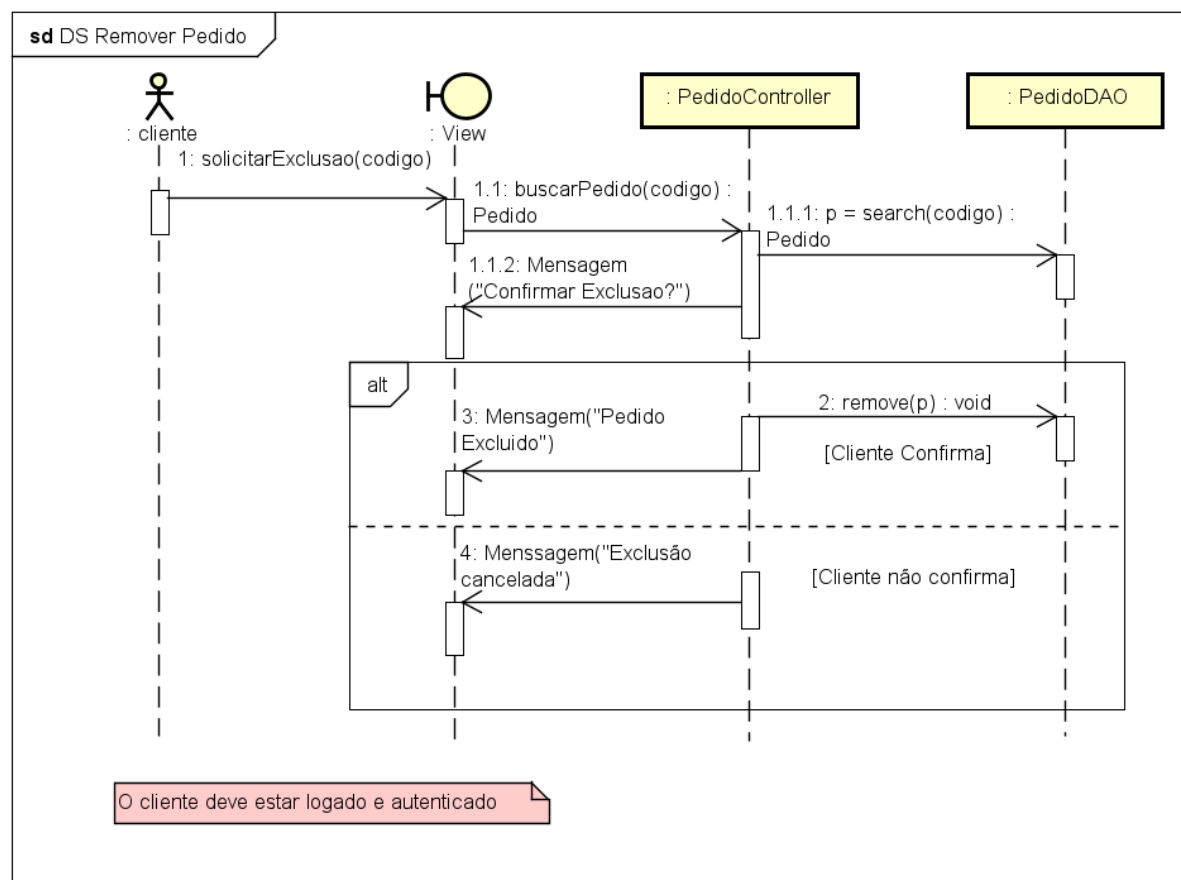
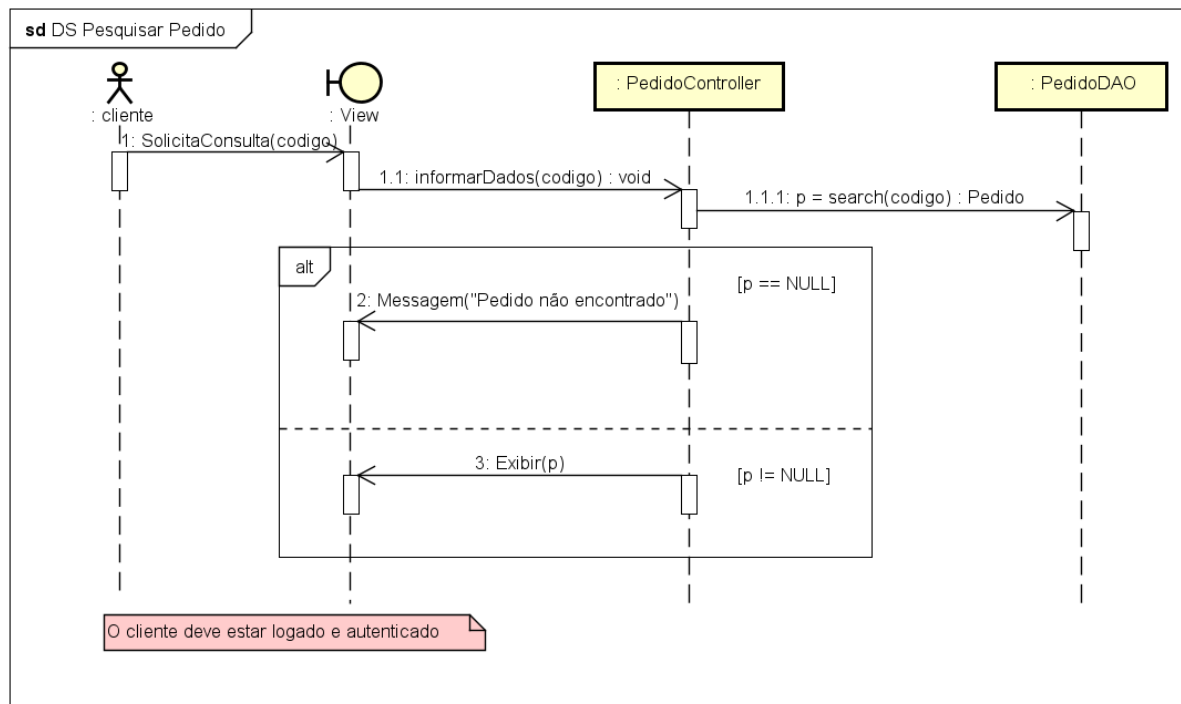












### 3.4 Mapeamento OO-Relacional (Modelo Lógico de Dados)

