



BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**Felipe Dos Santos Nves
Paulo Eduardo Bolher**

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

Pizza's Truffle

Presidente Epitácio – SP
2022

Sumário

Capítulo 1 - DESCRIÇÃO GERAL DO PRODUTO	3
1.2 Escopo do Produto.....	3
1.3 Funções do Produto.....	3
1.3 Regras do Negócio	6
1.4 Perspectiva do produto / Requisitos não funcionais	6
1.5 Restrições, Suposições e Dependências.....	6
Capítulo 2 REQUISITOS ESPECÍFICOS	7
2.1 Diagrama de Casos de Uso	7
2.2 Especificações de Casos de Uso e Diagramas de Atividades	8
Subfluxo Consultar Cliente	16
Capítulo 3 PROJETO DE SOFTWARE.....	28
3.1 Arquitetura Lógica de Software.....	28
3.2 Diagrama de Classes	29
3.3 Diagramas de Sequência.....	31
3.4 Mapeamento OO-Relacional (Modelo Lógico de Dados).....	41

Capítulo 1 - DESCRIÇÃO GERAL DO PRODUTO

1.2 Escopo do Produto

O sistema Pizza 's Truffle (Sistema de Delivery de pizza) tem como objetivo permitir que os administradores da empresa tenham um melhor gerenciamento de seu negócio, obtendo uma melhor performance dos serviços prestados.

O Atendente (Usuário) será responsável pelo cadastro de entregadores e clientes. O mesmo também terá permissão para excluir um determinado entregador ou um cliente específico. Além disso, poderá efetuar vendas, e, logo após, gerar cupom fiscal. Lançar pagamentos, gerenciar ingredientes, abrir caixa e cadastrar tanto pizzas como bebidas.

O Pizzaiolo, este poderá ter acesso a uma tela específica onde após um pedido ser registrado, ele aparecerá nela e poderá ser alterado o estado do pedido para pronto.

O Gerente, após acessar o sistema terá permissão para realizar todas as tarefas de um atendente, também poderá gerenciar os usuários, as despesas e os fornecedores, efetuar compra, fechar o caixa, pagar o entregador como as despesas e os fornecedores, além de imprimir relatórios e acessar a tela do Pizzaiolo.

A implantação do Pizza 's Truffle beneficiará a empresa em sua parte administrativa, contribuirá para que seja realizada de forma mais rápida e precisa o atendimento aos seus clientes.

1.3 Funções do Produto

Funções de Básica:

RF_FB01 – Manter clientes. Para manter cliente é necessário as seguintes informações: código, nome, endereço, cidade, estado, contato, CPF e E-mail.

RF_FB02 – Manter usuário. Para manter usuário é necessário as seguintes informações: código, nível de acesso, nome, data nascimento, E-mail, senha, endereço, cidade, estado e contato .

RF_FB03 – Manter entregador. Para manter Entregador é necessário as seguintes informações: código, nome, endereço, cidade, estado, CPF, E-mail e contato.

RF_FB04 – Manter pizza. Para manter Pizza é necessário as seguintes informações: código, nome, preço e ingredientes.

RF_FB05 – Manter ingredientes. Para manter ingredientes é necessário as seguintes informações: código e nome.

RF_FB06 – Manter bebida. Para manter bebida é necessário as seguintes informações: código, nome, preço e quantidade.

RF_FB07 – Manter fornecedor. Para manter fornecedor é necessário as seguintes informações: código, nome, endereço, cidade, estado, CNPJ, E-mail, contato.

RF_FB08 – Manter borda. Para manter a borda é necessário as seguintes informações: descrição.

RF_FB09 – Manter despesa. Para manter a despesa é necessário as seguintes informações: Caixa, nome, descrição, valor, data de vencimento e data de pagamento.

Funções Fundamentais:

RF_FF01 – Efetua venda. Quando o usuário for realizar uma venda, o sistema deve registrar os dados relacionados com o tal registro. Para isso, são necessários os seguintes itens de informação: Qual caixa está sendo realizando a venda, para qual cliente está sendo vendido, data da venda, total da venda, e o valor do desconto.

RF_FF02 – Registrar compra do fornecedor. Ao ser realizada a compra, o sistema deve registrar os dados relacionados com o tal registro. Para isso, são necessários os seguintes itens de informação: Qual caixa está fazendo a compra, de qual fornecedor está sendo comprado, data da compra e valor total da compra.

RF_FF03 – Abrir caixa. Para realizar a abertura do caixa, o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Código do usuário, nome do usuário, data da abertura do caixa, data do fechamento do caixa, horário de abertura do caixa, horário de fechamento do caixa, valor inicial do caixa, valor final do caixa.

RF_FF04 – Fechar caixa. Para realizar o fechamento do caixa, o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Código do usuário, nome do usuário, data da abertura do caixa, data do fechamento do caixa, horário de abertura do caixa, horário de fechamento do caixa, valor inicial do caixa, valor final do caixa.

RF_FF05 – Liberar pedido para entrega. Para realizar a mudança de estado do preparo, o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Estado novo.

RF_FF06 – Efetuar pagamento do entregador. Para efetuar o pagamento do entregador sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Total de entregas feita e o total por entrega.

RF_FF07 – Pagar o fornecedor. Para efetuar o pagamento do fornecedor o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Data de pagamento e valor pago.

RF_FF08 – Lançar pagamento. Para efetuar o pagamento do pedido o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Total do pedido e data de pagamento.

RF_FF09 – Efetuar login. Para efetuar login o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Email e senha.

RF_FF10 – Gerar cupom fiscal. Para efetuar o cupom fiscal o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Código do produto , nome dos produtos comprados, quantidade dos produtos, valor por unidade, valor total, data e hora da compra, cpf do cliente caso seja requisitado.

RF_FF11 – Pagar despesa. Para efetuar o pagamento da despesa o sistema deve registrar os dados para essa operação. Para isso, são necessários os seguintes itens de informação: Data de pagamento.

Funções de saída:

RF_FS01 – Gerar relatório de movimentação de caixa.

Filtro: período.

Itens de informação exibidos: código, caixa, valor, motivo e tipo.

RF_FS02 – Gerar relatório de compra.

Filtro: período.

Itens de informação exibidos: código, caixa, fornecedor, data e total.

RF_FS03 – Gerar relatório de venda.

Filtro: período.

Itens de informação exibidos: código, caixa, cliente, data e total.

RF_FS04 – Gerar relatório de usuário.

Filtro: código.

Itens de informação exibidos: código, nível de acesso, nome, data de nascimento, email, senha, endereço, cidade, estado, contato.

RF_FS05 – Gerar relatório de cliente.

Filtro: código.

Itens de informação exibidos: código, nome, endereço, cidade, estado, CPF, email e contato.

1.3 Regras do Negócio

Estas serão as regras de negócio:

RN_001 – O cliente receberá um cartão fidelidade a cada compra feita, em sua 11ª pizza, ganhará um desconto de 5%.

RN_002 – Não serão aceitos pagamentos em cheque.

RN_003 – Todo último domingo do mês será lançado um desconto promocional de 5% de desconto a todos os pedidos do dia.

RN_004 – O estabelecimento receberá pedidos até às 23:59.

1.4 Perspectiva do produto / Requisitos não funcionais

O cliente optou por esta abordagem pelo fato da interface de usuário ter as telas de Crud's padronizadas, isto favorece muito a usabilidade do sistema, facilitando o processos de Crud. Além disto, todas as telas de pesquisa utilizando de filtros para realizar suas buscas, o que também ajuda o usuário a ter um melhor uso destas funcionalidades.

Outro fator importante para decisão é que esta abordagem delimita dois níveis de acesso:

Atendente – Tem acesso ao gerenciamento de entregadores, cadastro de pizzas, realização de vendas e compras;

Gerente – Além realizar todas atividades do atendente, poderá gerenciar os atendentes e imprimir relatórios como de venda, compra, entre outros.

Para interface de hardware será necessário um computador com um monitor, tendo como requisitos mínimos de execução: 4G RAM, processador dual core ou superior e 500 GB de HD. Para a interface de software será adotado o banco de dados MySQL Workbench. Afim de que esta abordagem funcione de maneira correta foi esclarecido que é de extrema importância a configuração do relógio do servidor web para que esteja de acordo com o horário real.

1.5 Restrições, Suposições e Dependências

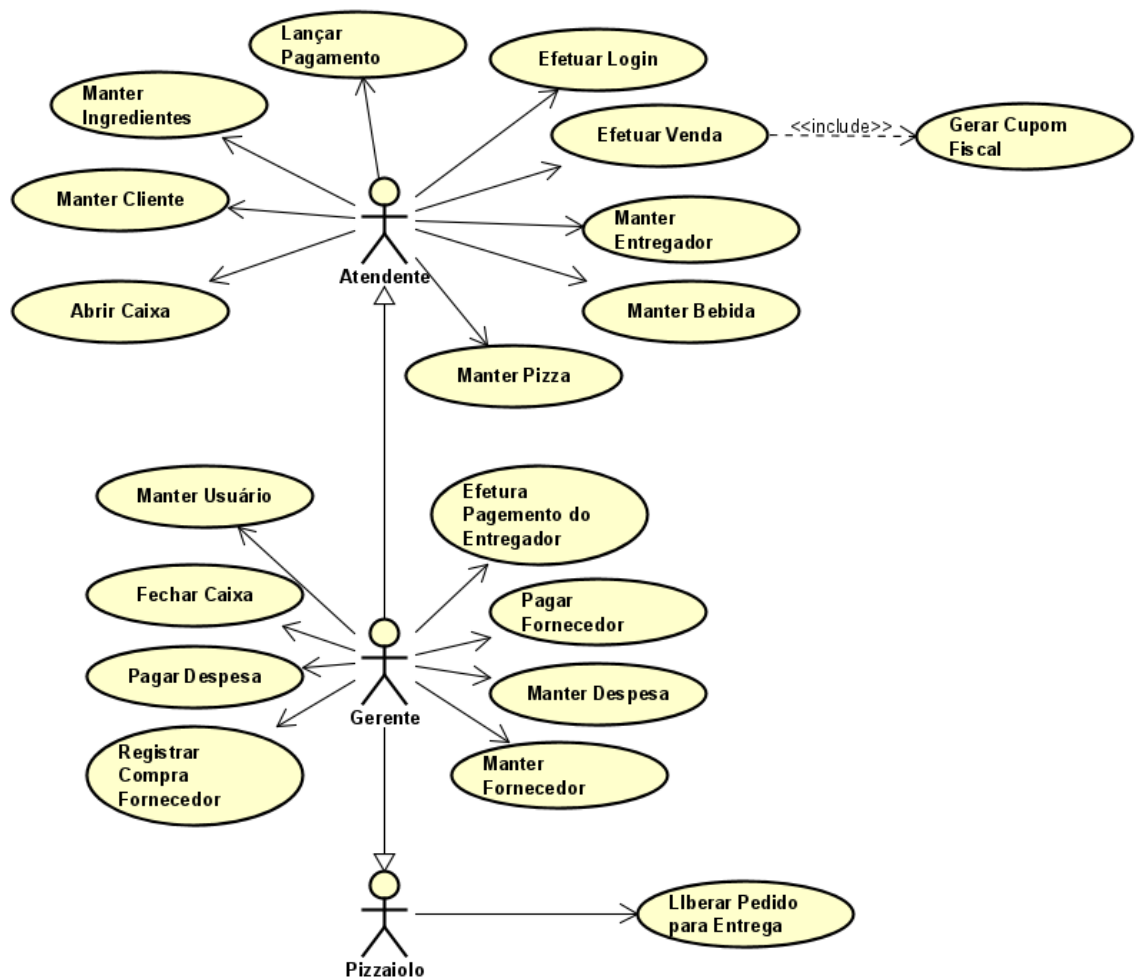
Qualquer restrição à utilização do software oriunda ao mau funcionamento da máquina, é responsabilidade do cliente, e indicado que seja feita manutenção preventiva do hardware periodicamente para que não ocorra o mesmo. Deve ser de

responsabilidade do cliente manter o banco de dados do antivírus atualizado.

A capacidade de armazenamento do banco de dados do software é responsabilidade do cliente, a capacidade deve atender a quantidade de dados armazenados.

Capítulo 2 REQUISITOS ESPECÍFICOS

2.1 Diagrama de Casos de Uso



2.2 Especificações de Casos de Uso e Diagramas de Atividades

Especificação do Caso de Uso: Registrar compra do fornecedor
Referências Cruzadas: RF_FF02
Ator Principal: Gerente
Interessados e Interesses: <ul style="list-style-type: none"> - Gerente: deseja gravar uma compra realizada. - Fornecedor: deseja vender seu produto.
Pré-condições: O Gerente é identificado e autenticado.
Garantia de sucesso (pós-condições): Os dados da compra são armazenados no sistema.
Fluxo Básico: <ol style="list-style-type: none"> 1. O gerente entra em contato com o fornecedor. 2. O gerente verifica se o fornecedor está cadastrado no sistema. 3. O gerente seleciona os produtos da compra. 4. O sistema registra a compra.
Fluxos Alternativos: <ol style="list-style-type: none"> 2. Caso o fornecedor não esteja registrado no sistema. <ol style="list-style-type: none"> 2.1 O gerente solicita os dados do fornecedor. 2.2 O sistema grava o novo fornecedor. 2.3 Retorna ao passo 3. 3. Caso o produto não esteja registrado no sistema. <ol style="list-style-type: none"> 3.1 O gerente cadastra o novo produto solicitado na compra. 3.2 Retorna ao passo 3.

Especificação do Caso de Uso: Pagar o fornecedor
Referências Cruzadas: RF_FF07
Ator Principal: Gerente
Interessados e Interesses: <ul style="list-style-type: none"> - Gerente: deseja realizar o pagamento do fornecedor.
Pré-condições: O Gerente é identificado e autenticado.

Garantia de sucesso (pós-condições): O pagamento é efetuado registrando a data do mesmo.
Fluxo Básico: <ol style="list-style-type: none"> 1. O gerente abre as compras pendentes. 2. O gerente seleciona a compra pendente para o pagamento. 3. O sistema registra a data de pagamento. 4. O sistema debita o caixa. 5. O gerente é notificado da debitação.
Fluxos Alternativos:

Especificação do Caso de Uso: Pagar despesa
Referências Cruzadas: RF_FF11
Ator Principal: Gerente
Interessados e Interesses: - Gerente: deseja pagar a despesa.
Pré-condições: O gerente é identificado e autenticado.
Garantia de sucesso (pós-condições): É registrado na despesa a data de pagamento.
Fluxo Básico: <ol style="list-style-type: none"> 1. O gerente abre as despesas pendentes. 2. O gerente seleciona a despesa pendente para o pagamento. 3. O sistema registra a data de pagamento. 4. O sistema debita o caixa. 5. O gerente é notificado da debitação.
Fluxos Alternativos:

Especificação do Caso de Uso: Liberar pedido para entrega
Referências Cruzadas: RF_FF05
Ator Principal: Pizzaiolo

Interessados e Interesses: - Pizzaiolo: deseja liberar o pedido para entrega. - Entregador: deseja realizar a entrega.
Pré-condições: O Pizzaiolo é identificado e autenticado.
Garantia de sucesso (pós-condições): O pedido é registrado e liberado para entrega.
Fluxo Básico: 1. O pizzaiolo abre os pedidos em preparo. 2. O pizzaiolo seleciona o novo estado do pedido para pronto. 3. O sistema grava o novo estado do pedido. 4. O sistema muda o estado da entrega para em andamento. 5. O entregador recolhe o pedido para entrega.
Fluxos Alternativos:

Especificação do Caso de Uso: Lançar pagamento
Referências Cruzadas: RF_FF08
Ator Principal: Atendente (Usuário)
Interessados e Interesses: - Atendente: deseja lançar o pagamento.
Pré-condições: O Atendente é identificado e autenticado.
Garantia de sucesso (pós-condições): O pagamento é concluído e registrado no sistema com sua forma de pagamento.
Fluxo Básico: 1. O entregador entrega o comprovante de pagamento da entrega. 2. O atendente lança o pagamento no sistema. 3. O sistema muda o estado da entrega para "entregue" 4. O sistema registra a data do pagamento na venda. 5. O sistema suplementa o caixa. 6. O sistema notifica a suplementação.
Fluxos Alternativos:

Especificação do Caso de Uso: Gerar cupom fiscal
Referências Cruzadas: RF_FF10
Ator Principal: Atendente (Usuário)
Interessados e Interesses: - Atendente: deseja finalizar a venda.
Pré-condições: O Atendente é identificado e autenticado.
Garantia de sucesso (pós-condições): O sistema imprime o cupom fiscal.
Fluxo Básico: <ol style="list-style-type: none"> 1. O Atendente pede para gerar cupom fiscal. 2. Sistema registra no cupom o nome da empresa. 3. Sistema registra no cupom o endereço da empresa. 4. Sistema registra no cupom o CNPJ da empresa. 5. Sistema registra no cupom a data da operação. 6. Sistema registra no cupom o valor total da operação. 7. Sistema registra no cupom a descrição do produto. 8. Sistema gera o cupom fiscal.
Fluxos Alternativos:

Especificação do Caso de Uso: Fechar caixa
Referências Cruzadas: RF_FF04
Ator Principal: Gerente
Interessados e Interesses: - Gerente: deseja realizar o fechamento do caixa para finalizar a diária.
Pré-condições: O Gerente é identificado e autenticado.
Garantia de sucesso (pós-condições): Os dados do caixa são armazenados no sistema.
Fluxo Básico: <ol style="list-style-type: none"> 1. O gerente abre a tela de caixa. 2. O gerente seleciona o caixa aberto. 3. O sistema altera o estado caixa.

4. O sistema guarda os dados do caixa.
Fluxos Alternativos:

Especificação do Caso de Uso: Efetuar pagamento do entregador
Referências Cruzadas: RF_FF06
Ator Principal: Gerente
Interessados e Interesses: - Gerente: deseja realizar o pagamento do entregador.
Pré-condições: O Gerente é identificado e autenticado.
Garantia de sucesso (pós-condições): O pagamento é efetuado registrando a data do mesmo.
Fluxo Básico: <ol style="list-style-type: none"> 1. O gerente seleciona o entregador. 2. O gerente realiza o pagamento do entregador selecionado. 3. O sistema registra a data de pagamento. 4. O sistema debita o caixa. 5. O sistema notifica o gerente.
Fluxos Alternativos:

Especificação do Caso de Uso: Efetuar login
Referências Cruzadas: RF_FF09
Ator Principal: Atendente (Usuário), Gerente e Pizzaiolo
Interessados e Interesses: - Atendente: deseja logar no sistema. - Gerente: deseja logar no sistema. - Pizzaiolo: deseja logar no sistema
Pré-condições: O usuário está cadastrado no sistema.
Garantia de sucesso (pós-condições): O usuário entre no sistema.

Fluxo Básico:

1. O usuário digita o email.
2. O usuário digita a senha.
3. O sistema verifica o usuario e a senha.
4. O usuário está liberado para utilizar o sistema.

Fluxos Alternativos:

3. O usuário ou a senha está incorreto.
 - 3A O usuário não está cadastrado no sistema.
 - 3A.1 O gerente cadastra o usuário.
 - 3A.2 Vá para o caso de uso manter usuário.
 - 3A.3 Retorna ao passo 1.
 - 3B O sistema relata ao usuário que algo está errando em seu email ou senha.
 - 3B.1 Retorna ao passo 1.

Especificação do Caso de Uso: Efetua venda

Referências Cruzadas: RF_FF01

Ator Principal: Atendente (Usuário)

Interessados e Interesses:

- Funcionário: deseja gravar uma venda realizada.
- Cliente: deseja obter o pedido.

Pré-condições: O Atendente é identificado e autenticado.

Garantia de sucesso (pós-condições): Os dados da venda são armazenados no sistema.

Fluxo Básico:

1. O cliente entra em contato com a pizzaria.
2. O atendente solicita ao cliente sua identificação e informa ao sistema.
3. O sistema exibe os dados do cliente.
4. O atendente solicita o produto do pedido ao cliente.
5. O sistema exibe dados do produto.
6. O sistema registra o produto na venda.
7. Atendente pergunta ao cliente se deseja mais produto.
8. Sistema registra a venda.

Fluxos Alternativos:

2. O cliente não tem identificação.
 - 2.1 O atendente pergunta ao cliente se deseja cadastrar-se no sistema.
 - 2A. O cliente aceita cadastrar-se no sistema.

- 2A. 1 O cliente entrega seus dados ao atendente.
- 2A. 2 Chama “Manter cliente”.
- 2A. 3 Avança ao passo 3.
- 2B. O cliente recusa cadastrar-se no sistema.
- 2B. 1 Avança ao passo 4.
- 4. O produto solicitado para o pedido está em falta no estoque.
- 4.1 O atendente informa ao cliente a falta do insumo.
- 4.2 Retorna ao passo 4.
- 7. Atendente pergunta se o cliente quer mais produtos.
- 7A. O cliente deseja mais produtos.
- 7A. 1 Retorna ao passo 4.
- 7B. O cliente não deseja mais produtos.
- 7B. 1 Avançar ao passo 8.

Especificação do Caso de Uso: Abrir caixa

Referências Cruzadas: RF_FF03

Ator Principal: Atendente (Usuário)

Interessados e Interesses:

- Atendente: deseja realizar a abertura do caixa para que então possa utilizar o sistema..

Pré-condições: O Atendente é identificado e autenticado.

Garantia de sucesso (pós-condições): Os dados do novo caixa são armazenados no sistema.

Fluxo Básico:

1. O sistema abre a tela de caixa.
2. O atendente fornece o valor inicial do caixa.
3. O sistema guarda os dados do novo caixa.
4. O sistema é liberado para uso.

Fluxos Alternativos:

2. O atendente insere um valor inválido.
 - 2.1 O sistema verifica o valor inserido.
 - 2.2 O sistema apresenta uma notificação ao usuário.
 - 2.3 Retorna ao passo 2.

Fluxo básico

1. O caso de uso inicia quando o Operador de caixa necessita fazer a manutenção (inclusão, alteração, exclusão ou consulta) de um cliente.
2. De acordo com o tipo de operação de manutenção desejado pelo Operador de

caixa, um dos subfluxos é executado:

1. Se o Operador de caixa deseja incluir um novo cliente, o subfluxo “Incluir cliente” é executado.
2. Se o Operador de caixa deseja alterar informações de um cliente já cadastrado, o subfluxo “Alterar cliente” é executado.
3. Se o Operador de caixa desejar excluir um cliente já cadastrado, o subfluxo “Remover cliente” é executado.
4. Se o Operador de caixa deseja consultar informações sobre um ou mais clientes cadastrados, o subfluxo “Consultar cliente” é executado.

Subfluxo Incluir Cliente

1. Este subfluxo inicia quando o *Operador de caixa* solicita incluir um *cliente*;
2. O sistema solicita ao *Operador de caixa* o preenchimento dos seguintes atributos:
 - Nome *
 - Endereço. Descreve a rua ou a avenida em que o cliente reside;
 - Cidade
 - Estado (campo de escolha fechada. Valores possíveis: todos os estados cadastrados no sistema);
 - CPF *
 - Conta
 - Email
3. O *Operador de caixa* preenche os atributos anteriores e confirma a inclusão;
4. O sistema realiza a inclusão dos dados informados pelo *Operador de caixa* no passo 3;
5. O sistema exibe uma mensagem informando que a inclusão do cliente foi efetivada com sucesso;

(*) atributos obrigatórios

Subfluxo Alterar Cliente

1. Este subfluxo inicia quando o *Operador de caixa* solicita alterar um *cliente*;
2. O *Operador de caixa* seleciona um único *cliente*;
3. O sistema solicita a alteração dos atributos listados no passo 2 do subfluxo “Incluir cliente”.
4. O *Operador de caixa* 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 *cliente* foi efetivada com sucesso;

Subfluxo Remover Cliente

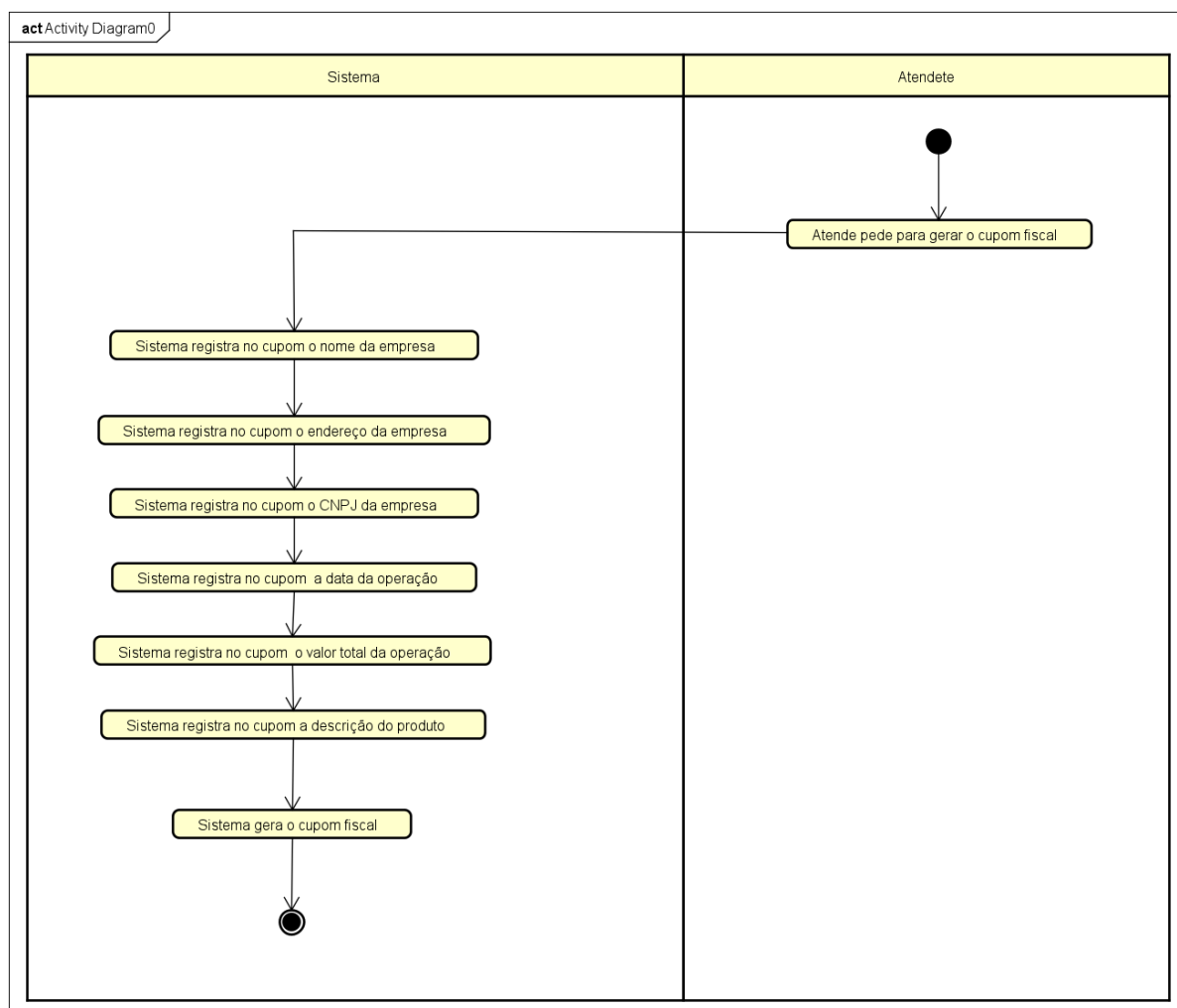
1. Este subfluxo inicia quando o *Operador de caixa* solicita remover um *clientes*;
2. O *Operador de caixa* seleciona qual *cliente* deseja remover e solicita a remoção;
3. O sistema solicita a confirmação para remoção;
4. O *Operador de caixa* confirma a remoção;
5. O sistema remove o *cliente* confirmado;
6. O sistema exibe uma mensagem informando que a remoção do *cliente* foi efetivada com sucesso;

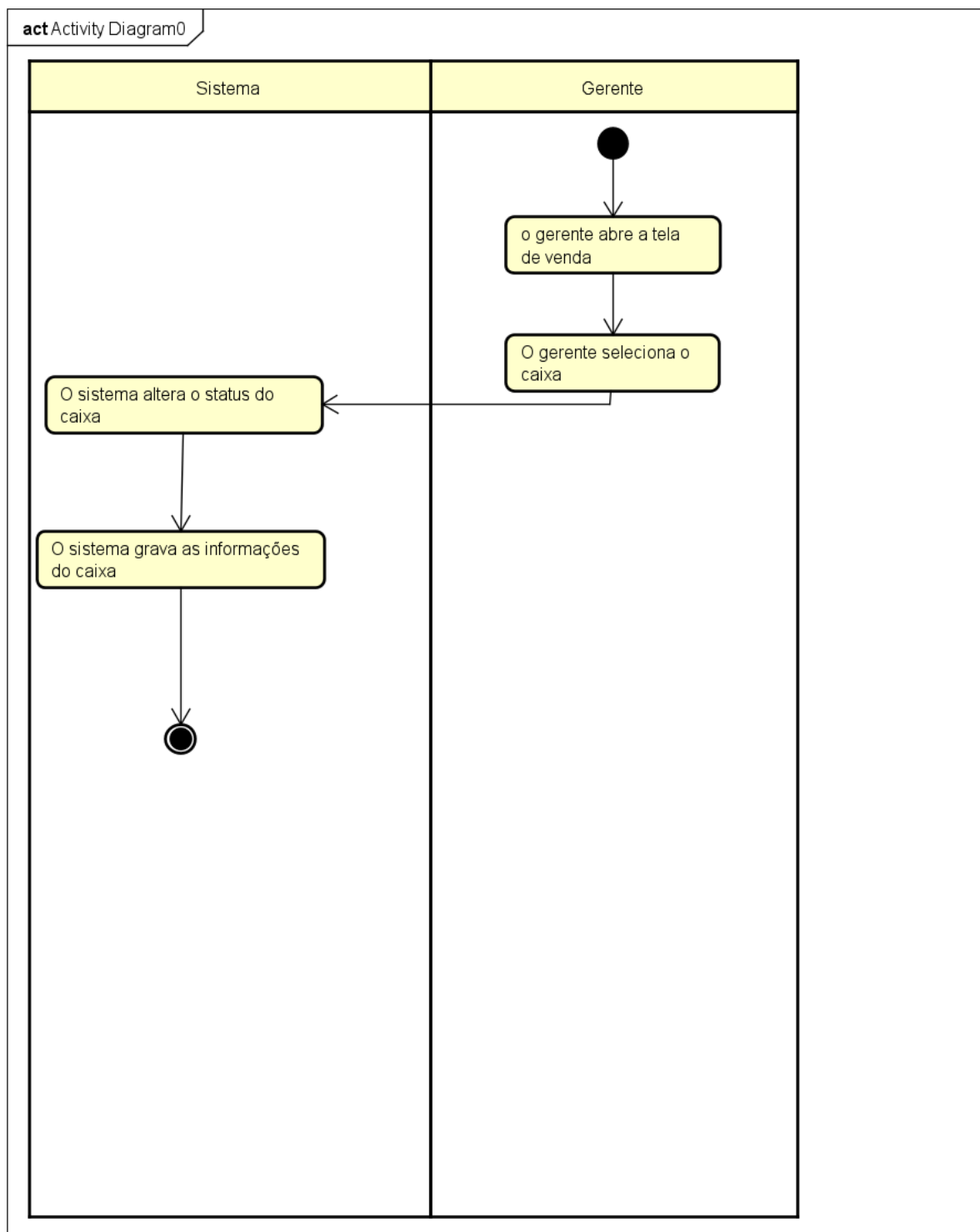
Subfluxo Consultar Cliente

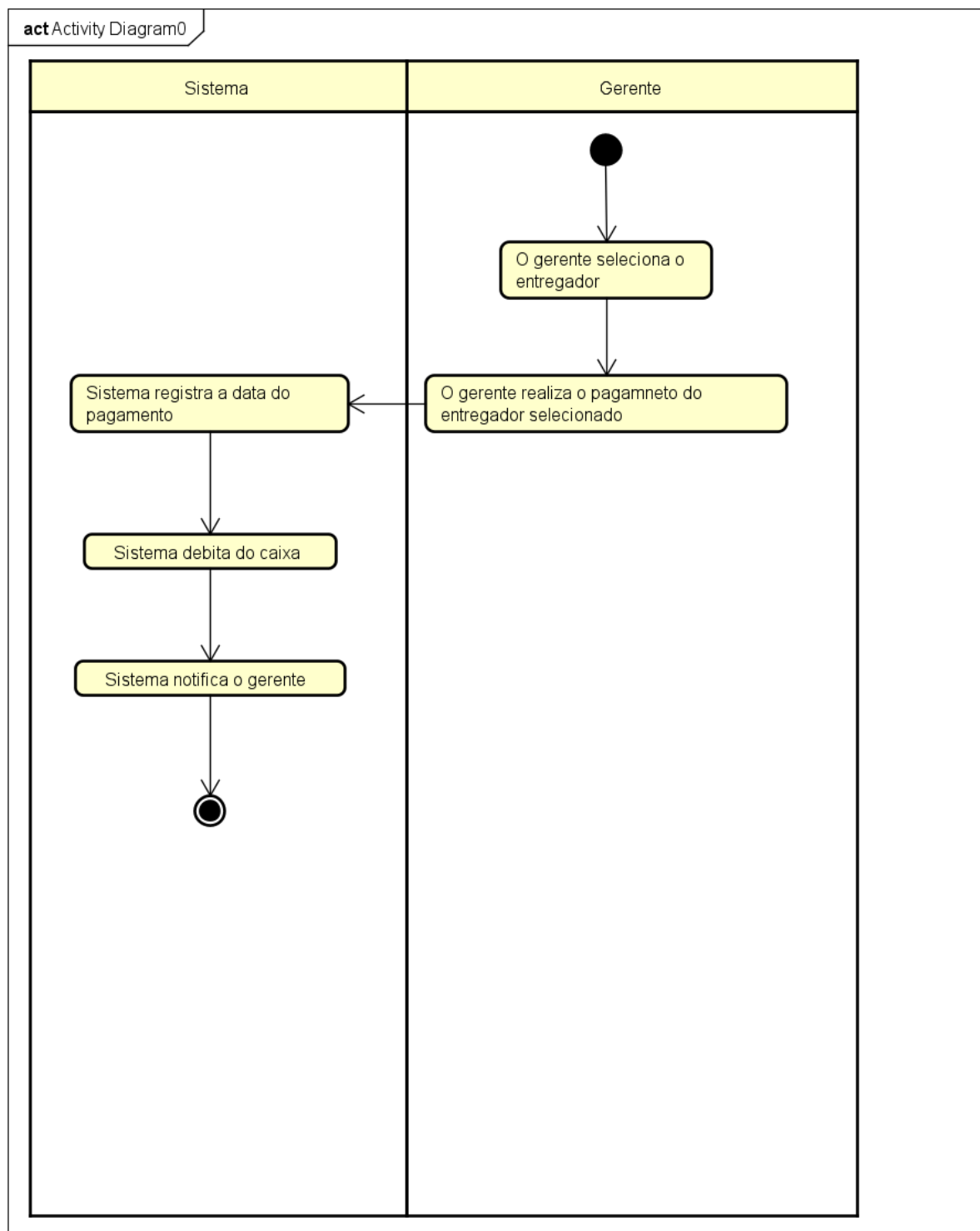
1. Este subfluxo inicia quando o *Operador de caixa* solicita consultar *clientes*;
2. O sistema solicita o preenchimento dos seguintes filtros:
 - Nome
3. O *Operador de caixa* preenche os filtros e solicita a consulta;
4. O sistema apresenta as seguintes informações dos *clientes* obtidos na consulta:
 - Nome, endereço, cidade, estado, CPF, email, contato.

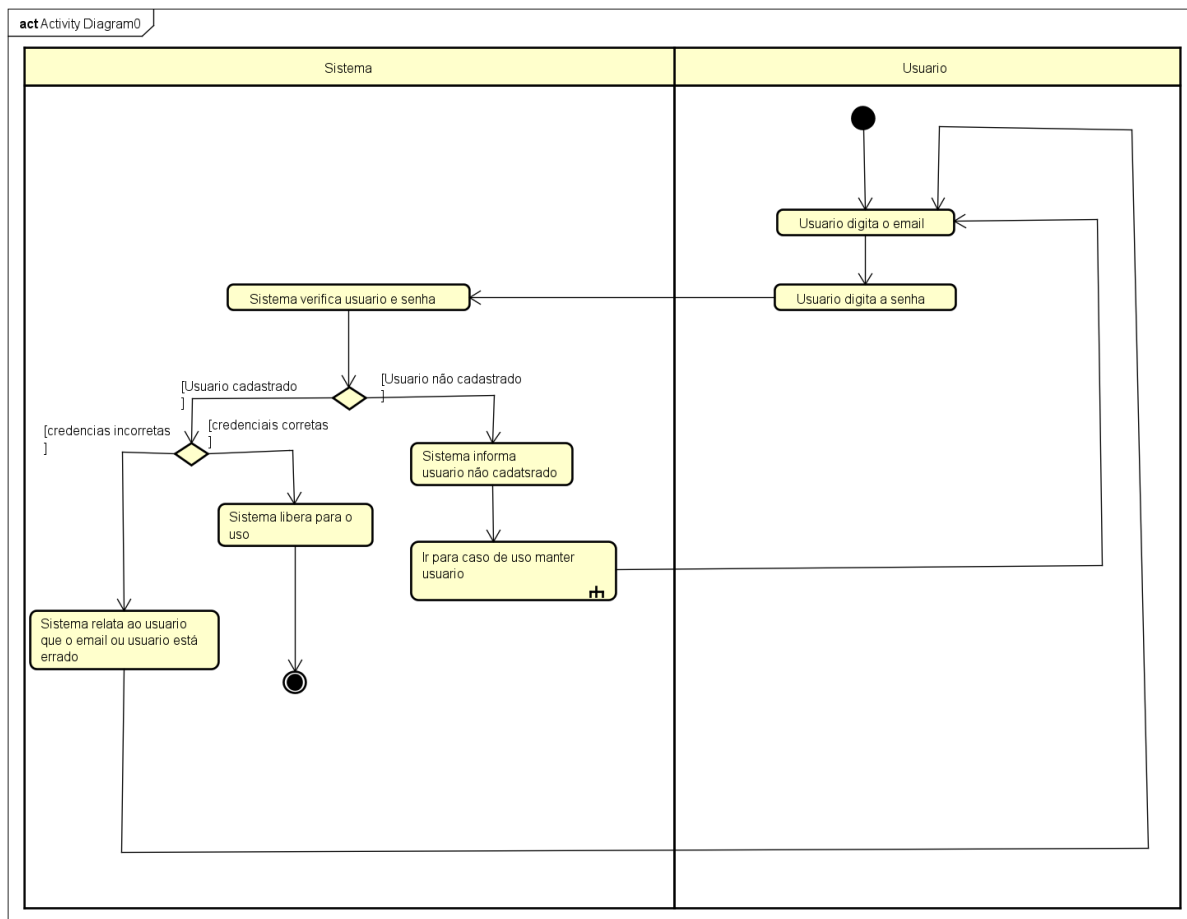
Validações e regras de negócio

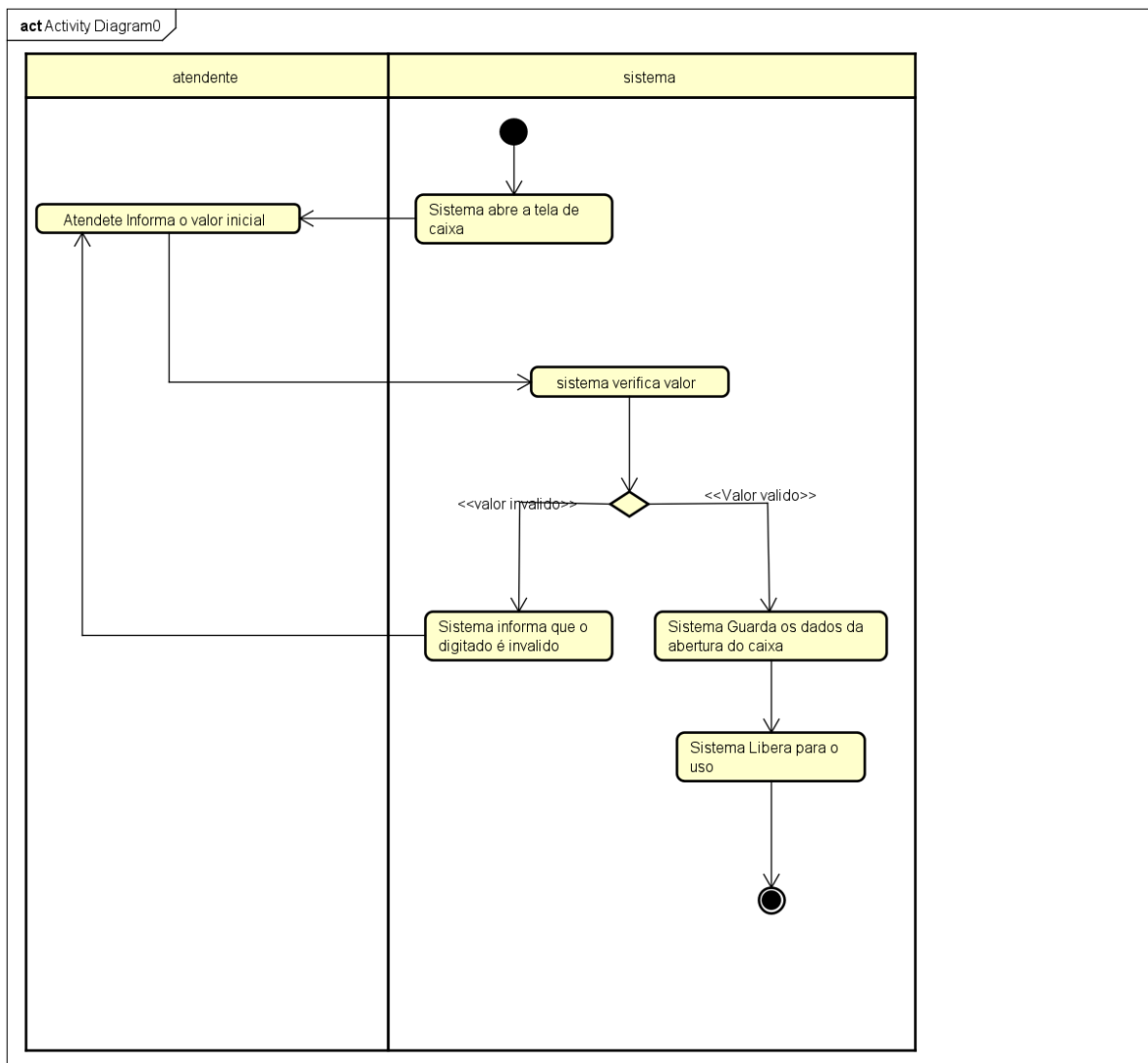
- Esta regra se aplica a todos os subfluxos. Atributos obrigatórios. Se algum atributo obrigatório não tiver sido preenchido, o sistema não completará a operação e notificará ao *Operador de caixa*, informando que os campos obrigatórios não foram preenchidos e solicitando o preenchimento dos mesmos;
- 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 *Operador de caixa*, informando quais campos foram preenchidos com valores inválidos e solicitando o preenchimento correto;

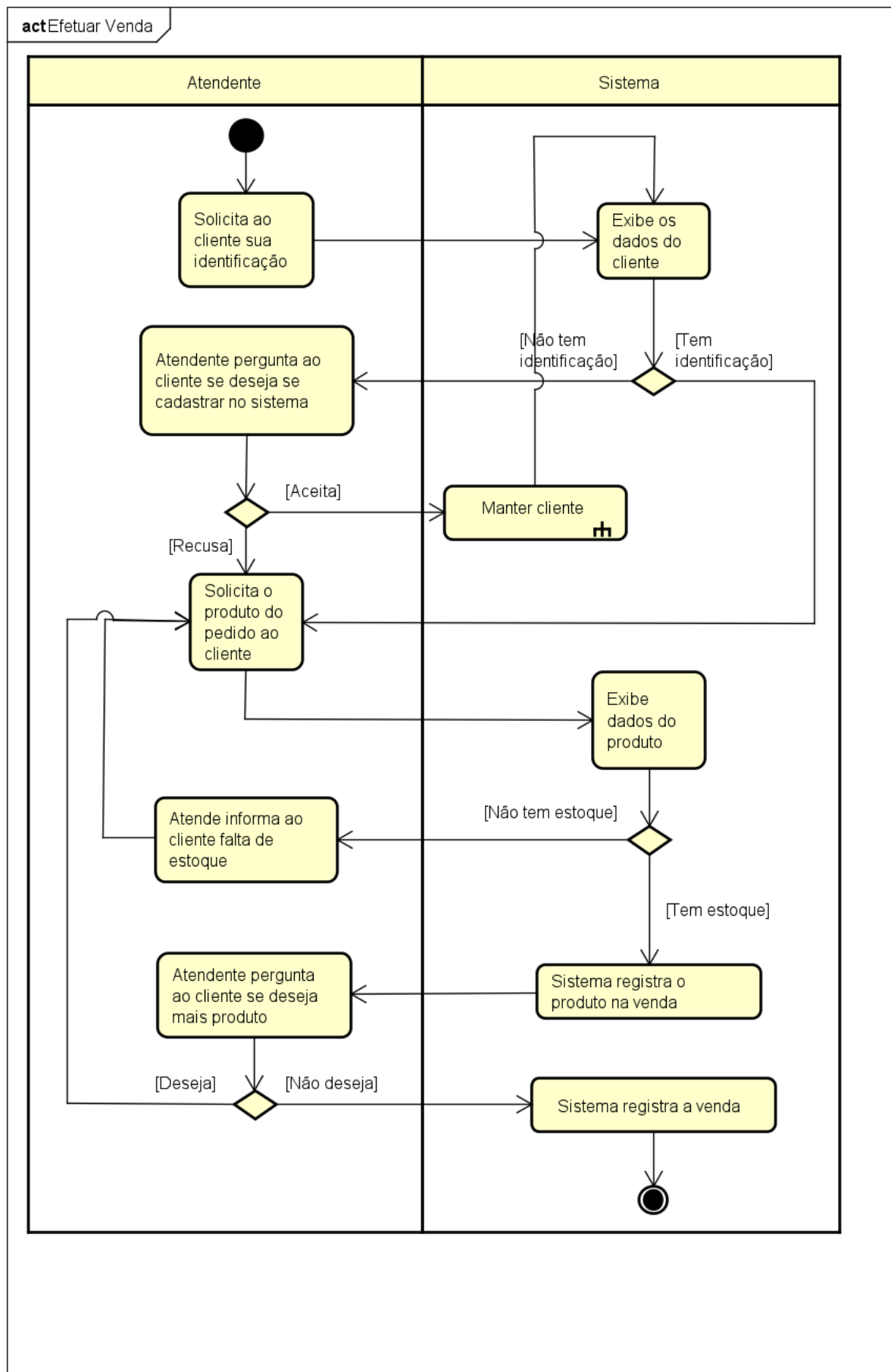


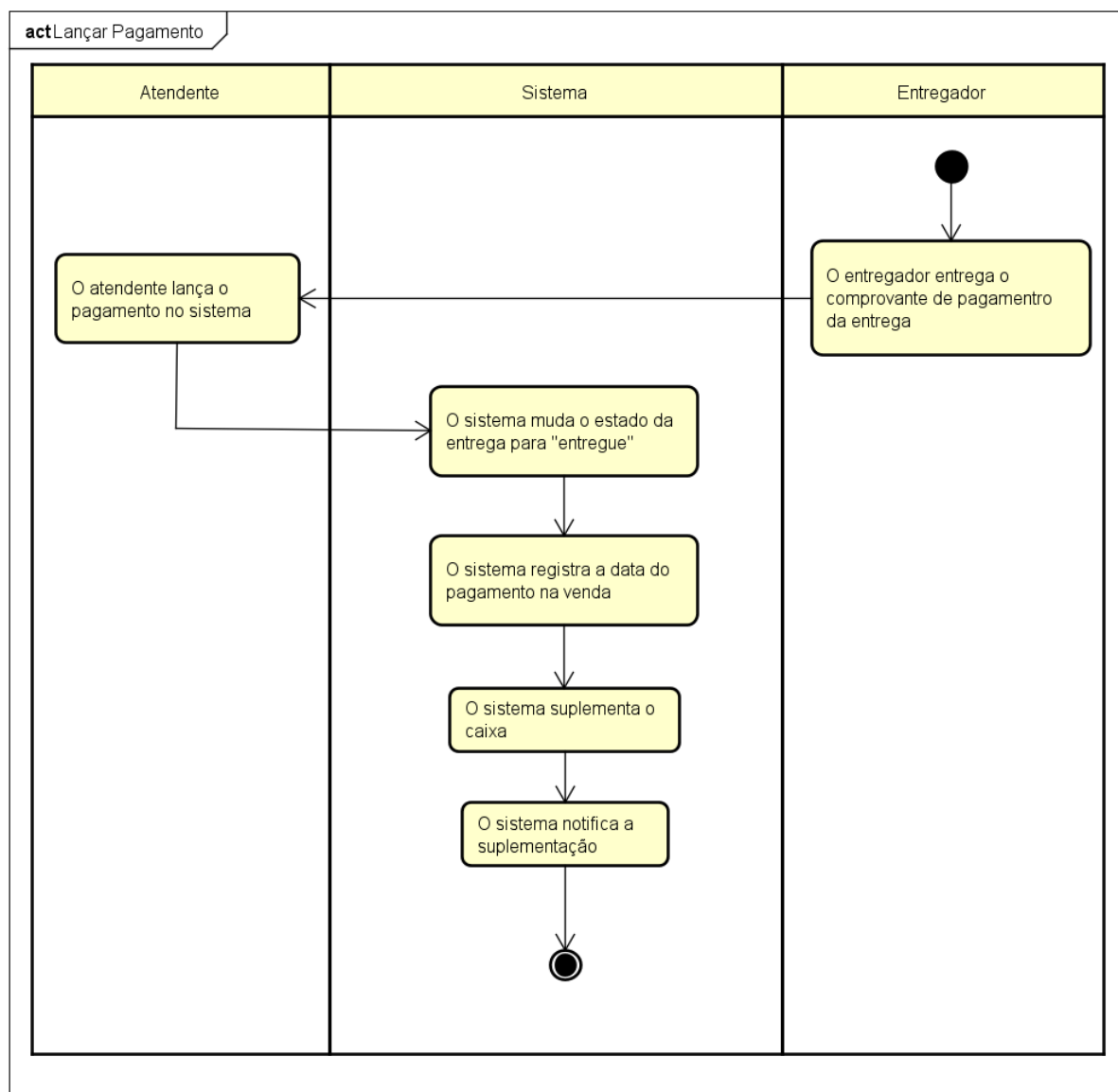


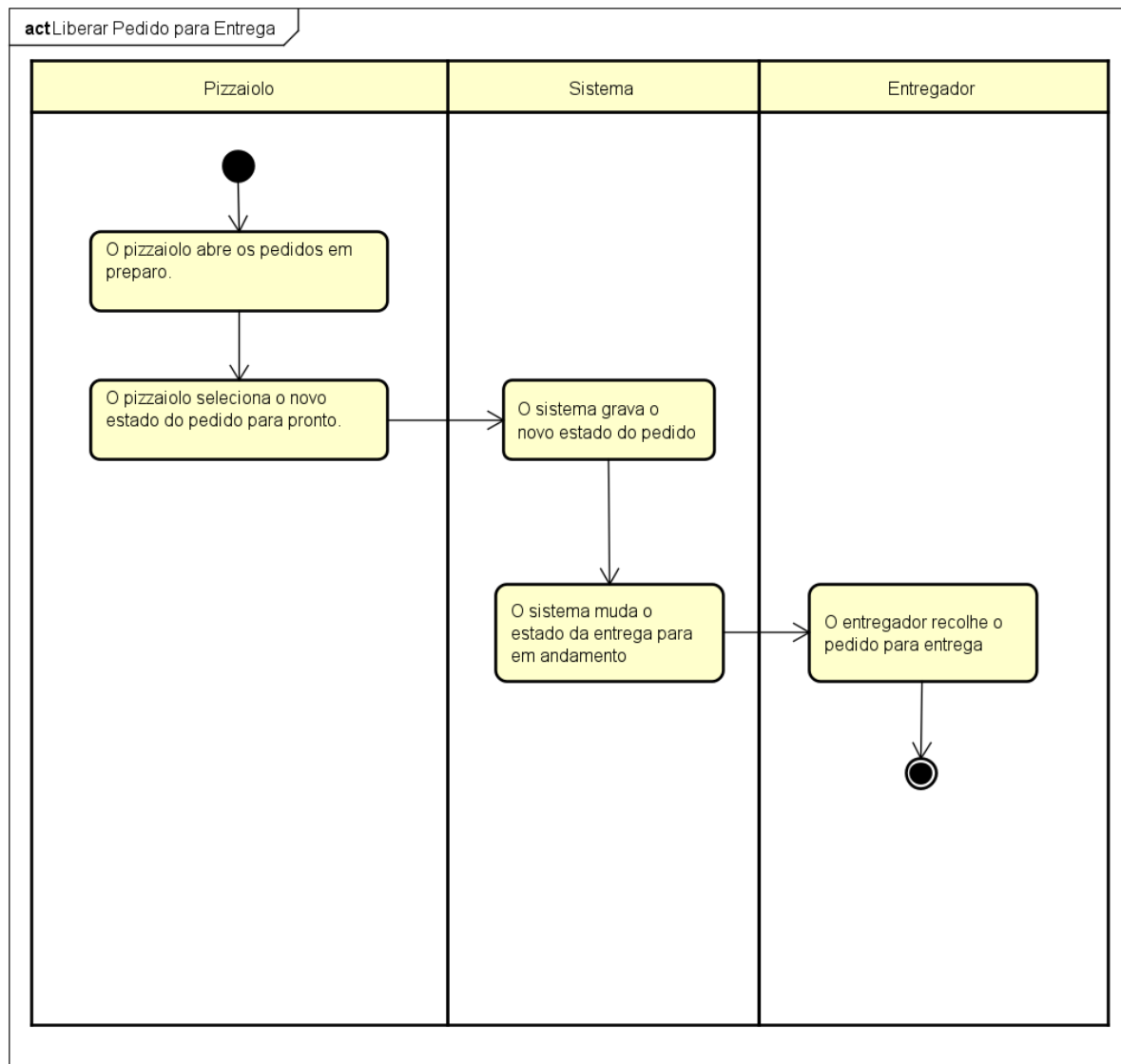


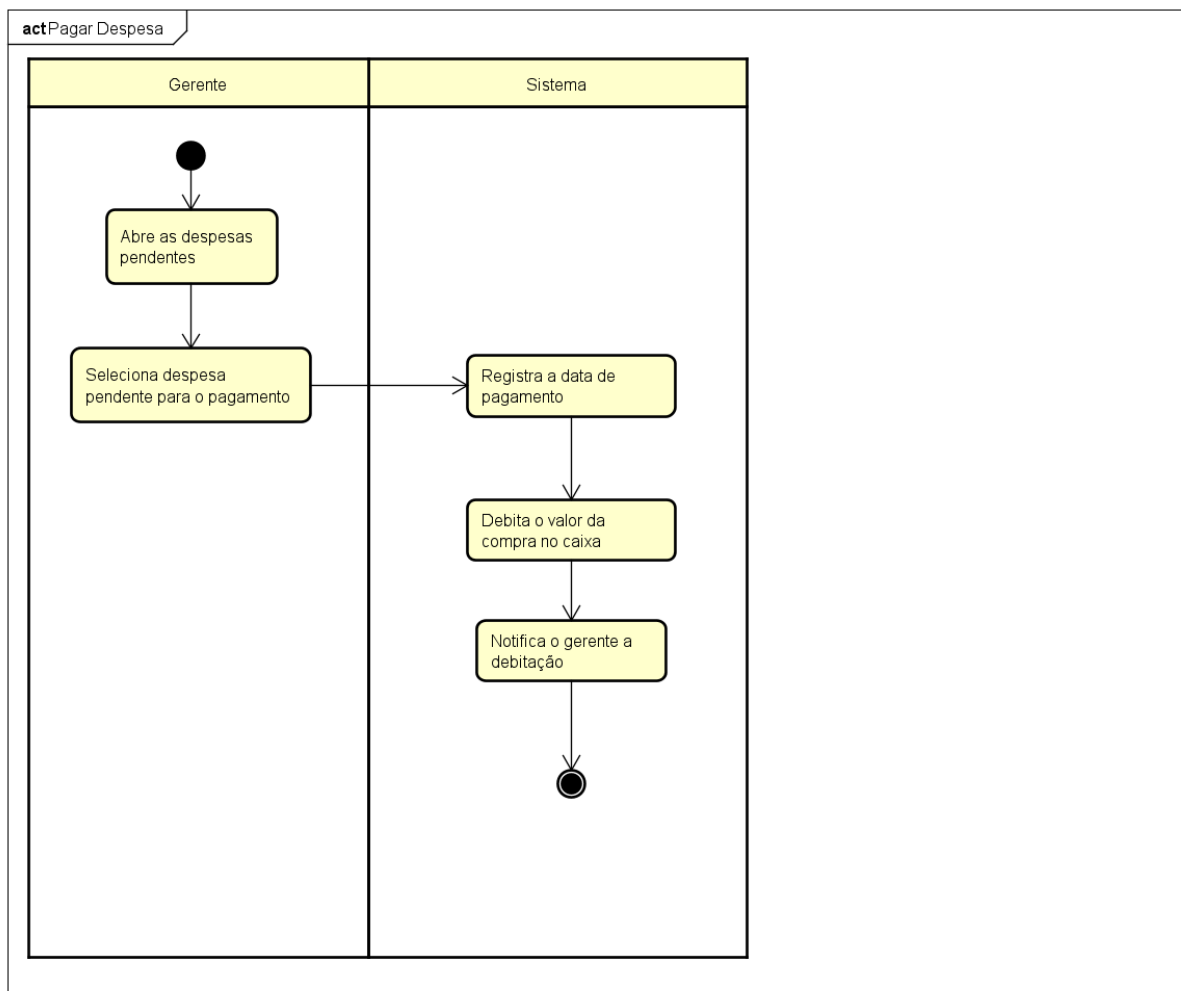


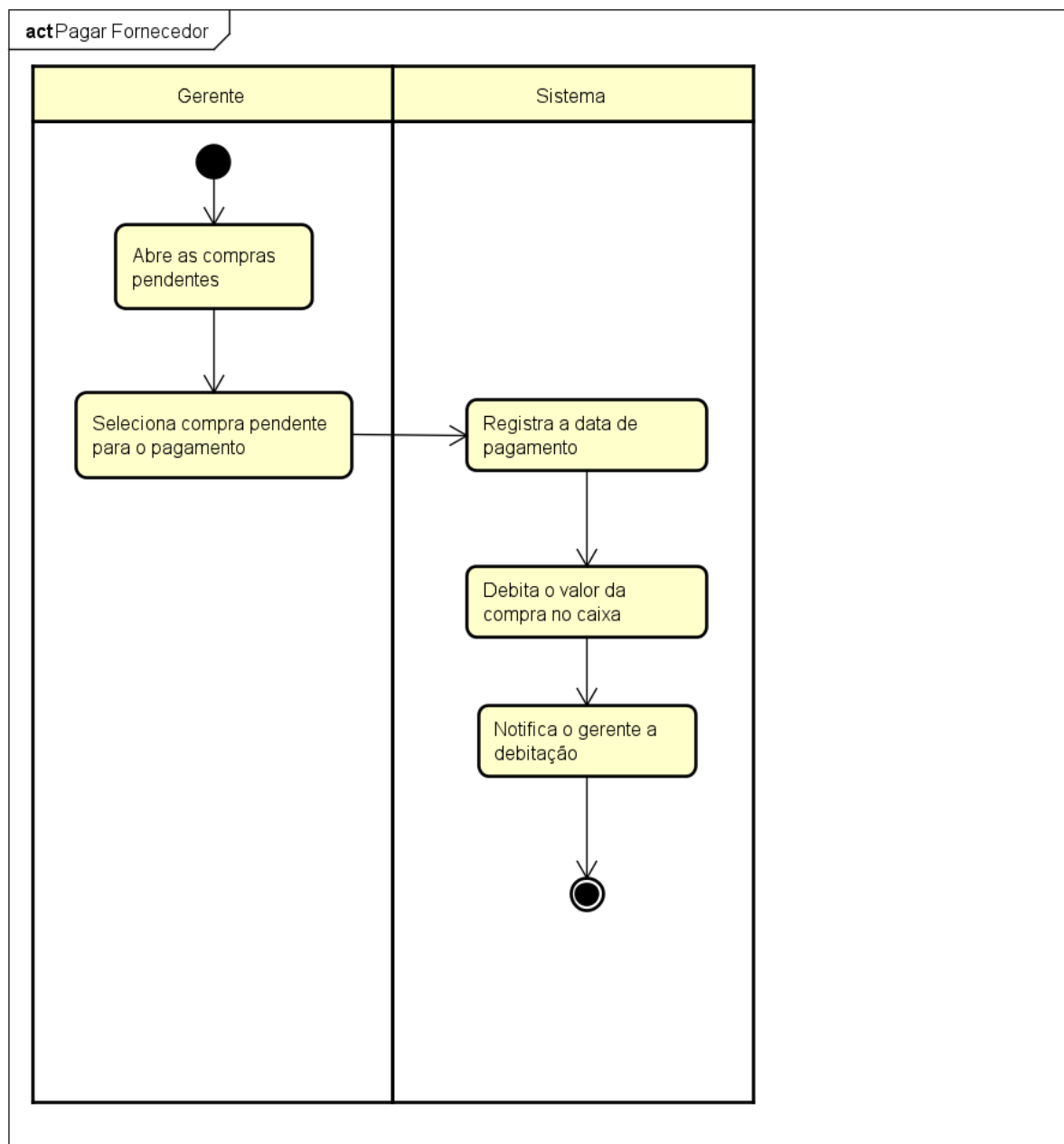


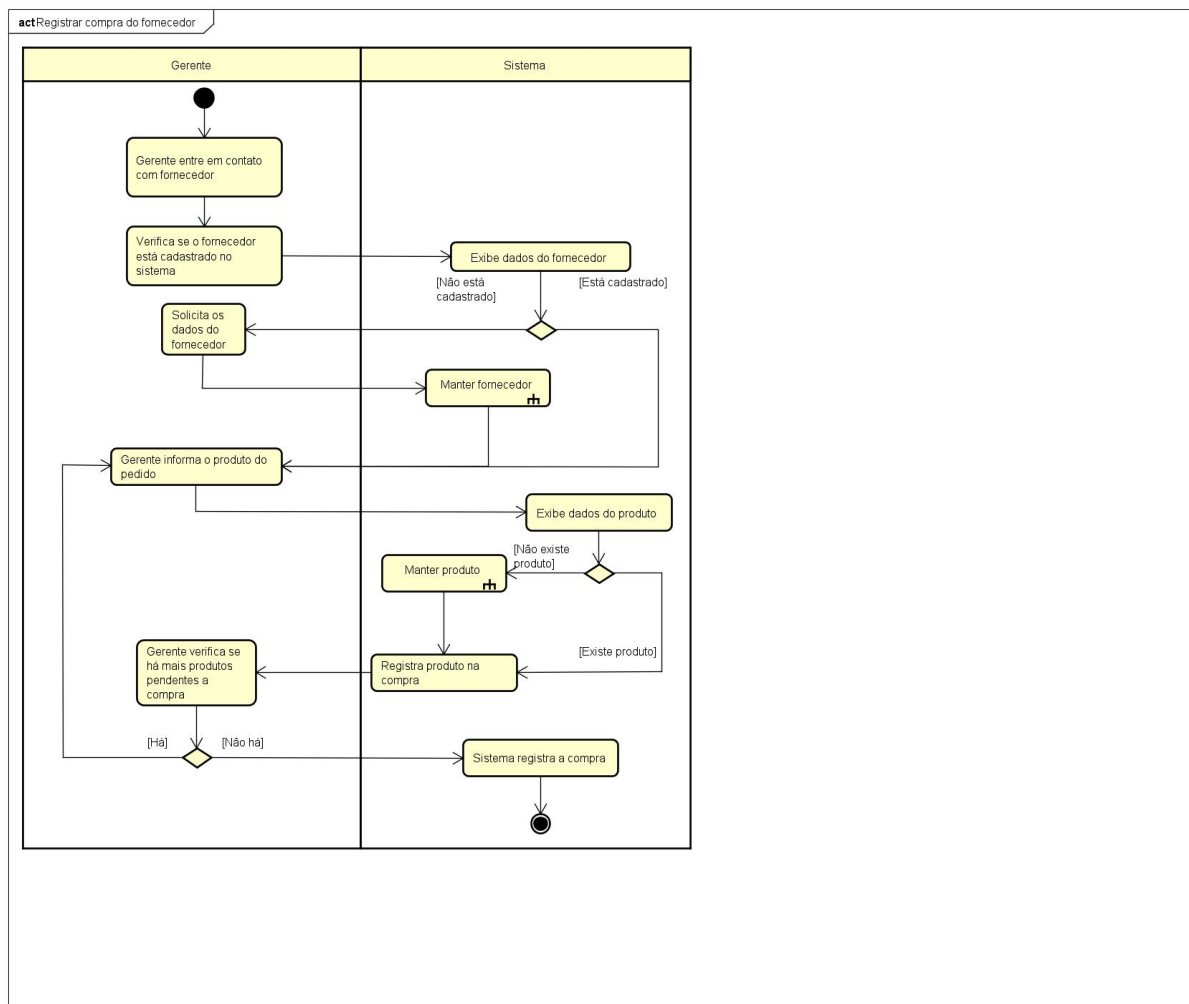












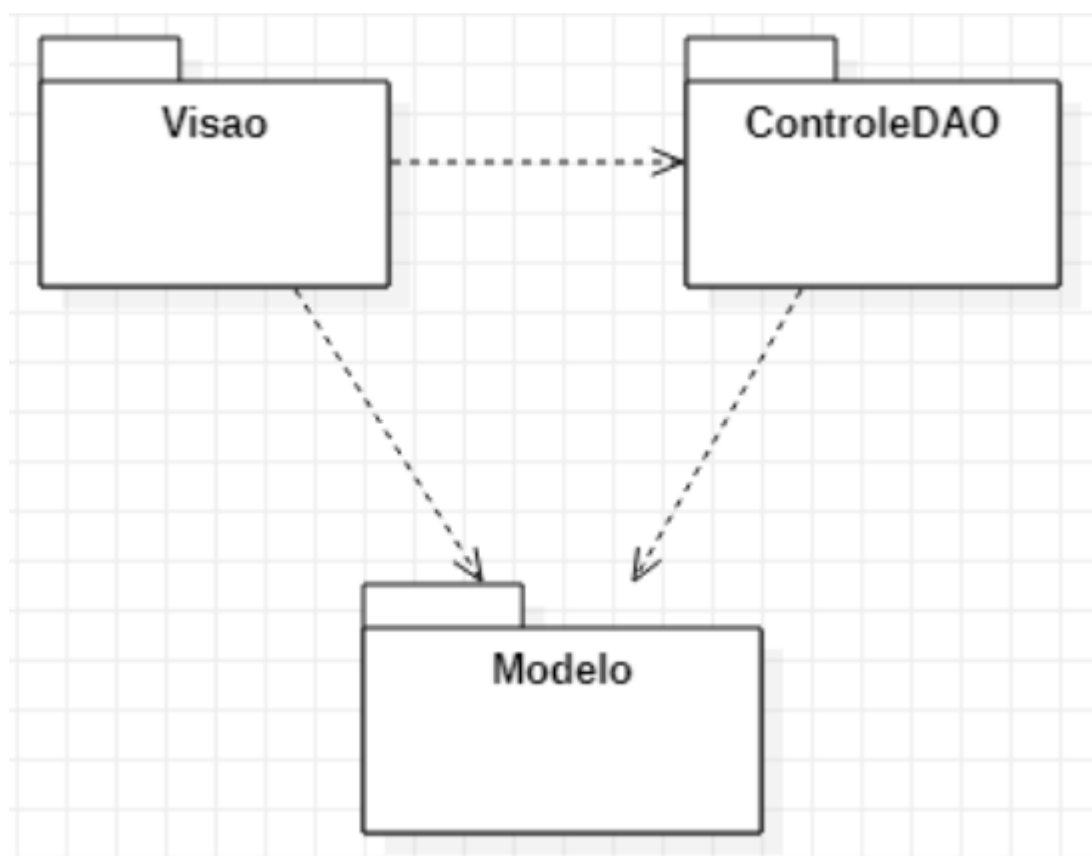
Capítulo 3 PROJETO DE SOFTWARE

3.1 Arquitetura Lógica de Software

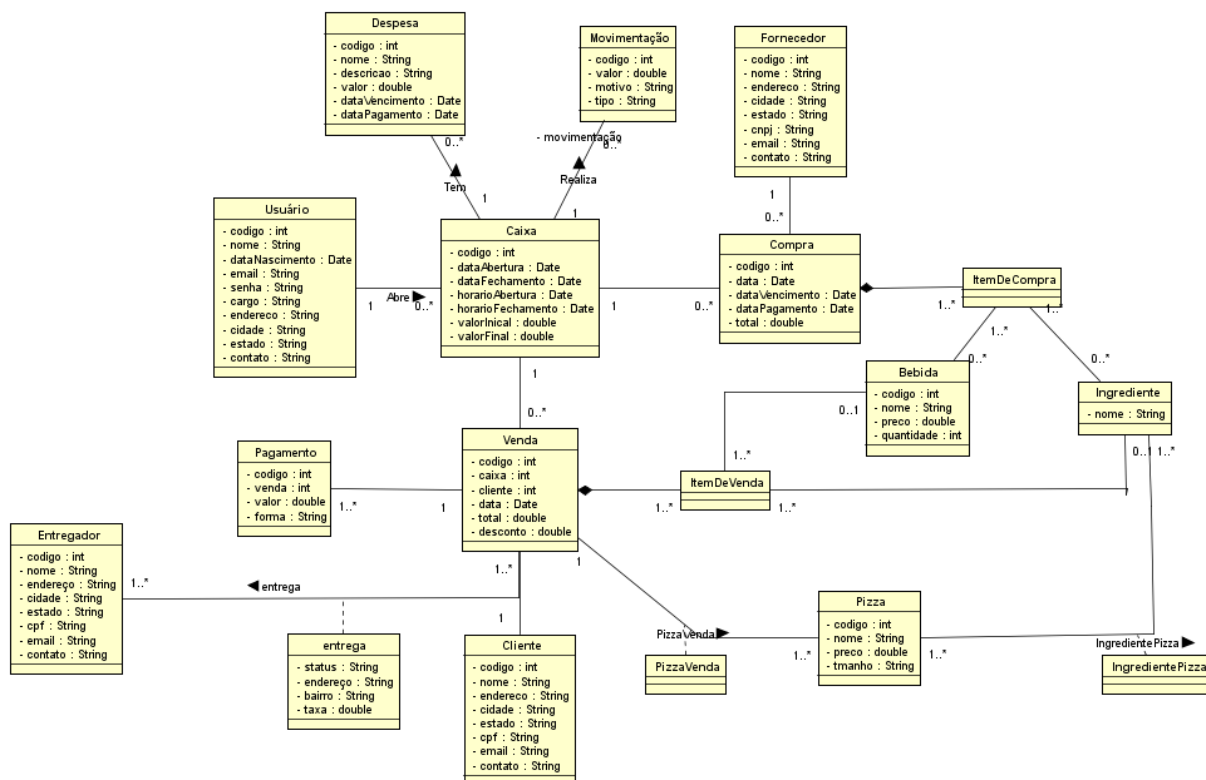
MVC é um padrão de arquitetura de software, separando sua aplicação em 3 componentes:

- interação do usuário (view): apresenta as informações de forma visual ao usuário. Está na linha de frente da comunicação com usuário e é responsável transmitir questionamentos ao controller e entregar as respostas obtidas ao usuário.
- manipulação dos dados (model): gerencia e controla a forma como os dados se comportam por meio das funções, lógica e regras de negócios estabelecidas; responsável pelas regras de negócios, persistência com o banco de dados e as classes de entidades.
- controle (controller): intermedia as requisições enviadas pelo view com as

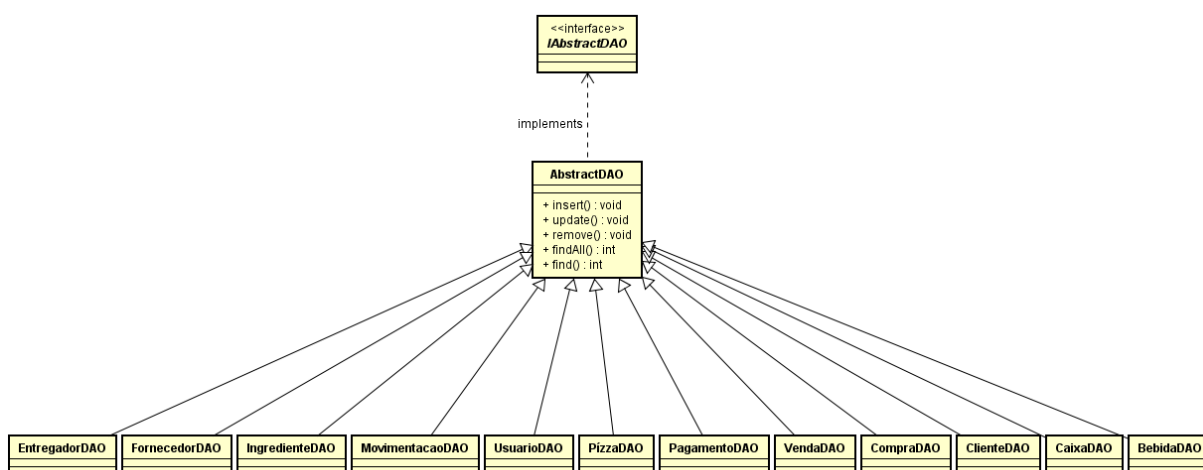
respostas fornecidas pelo model



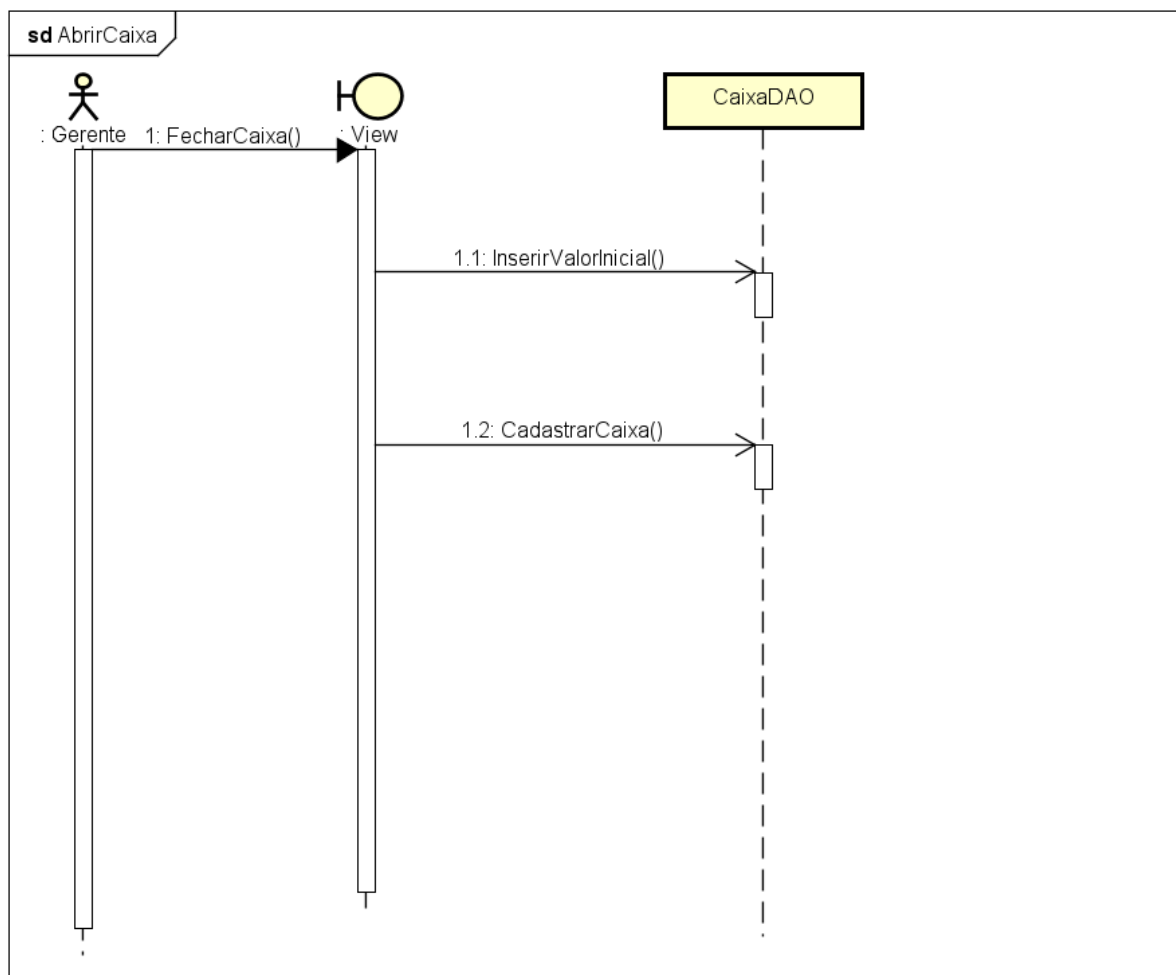
3.2 Diagrama de Classes

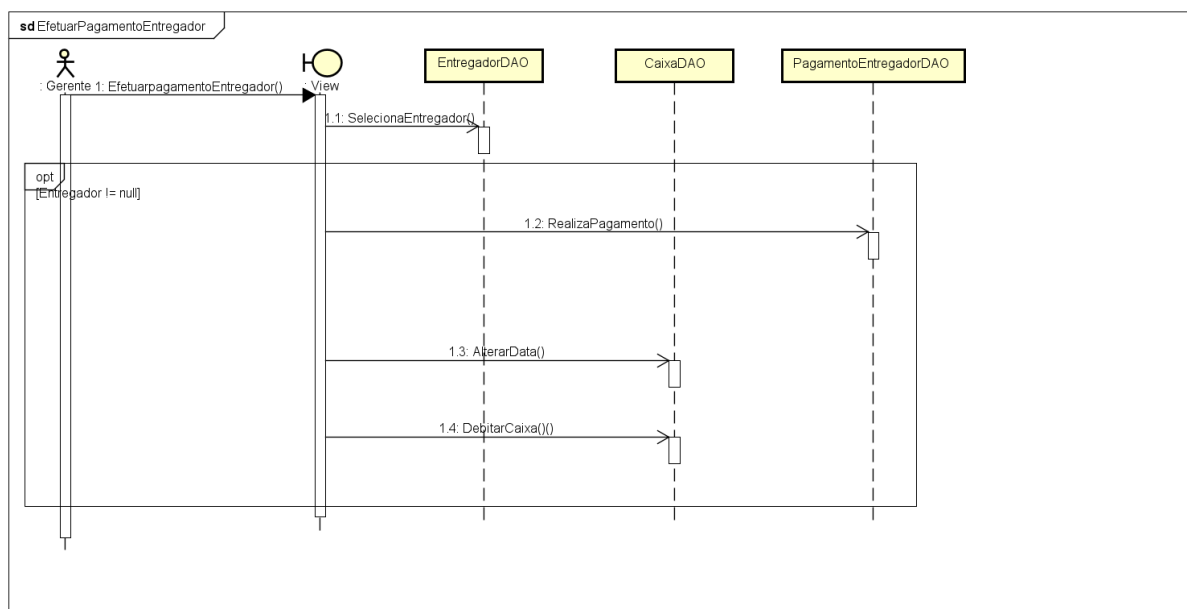
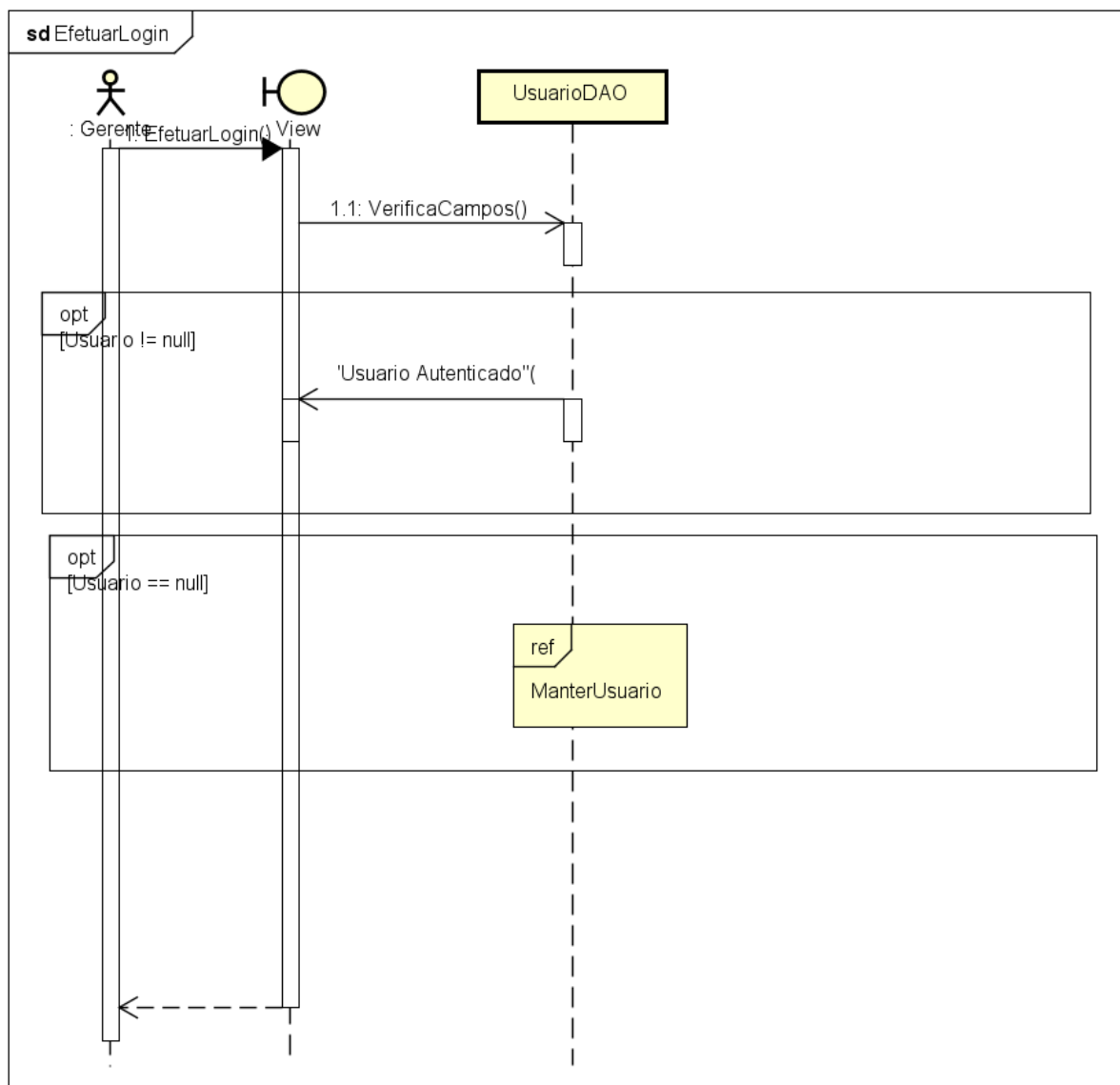


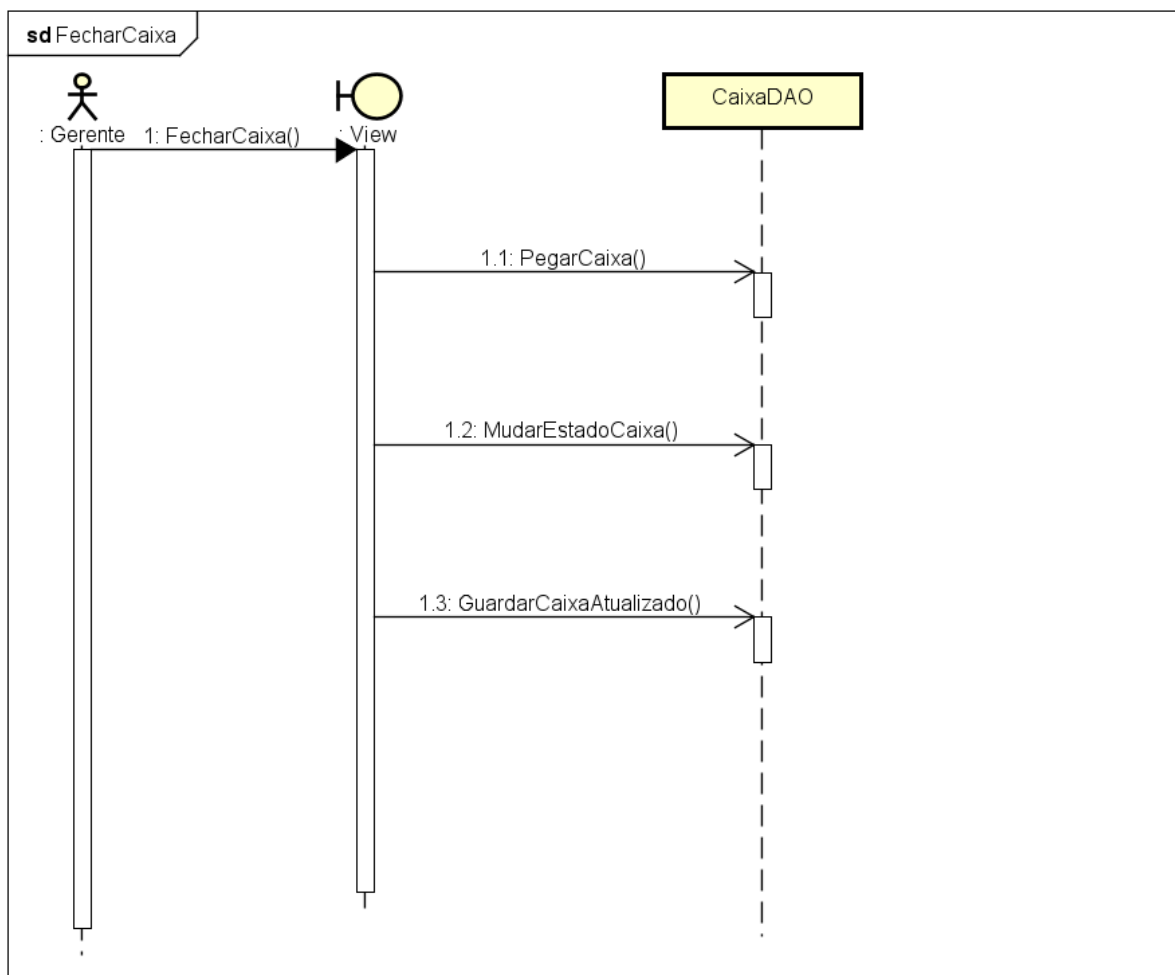
3.2.2 Diagrama de Classes do DAO

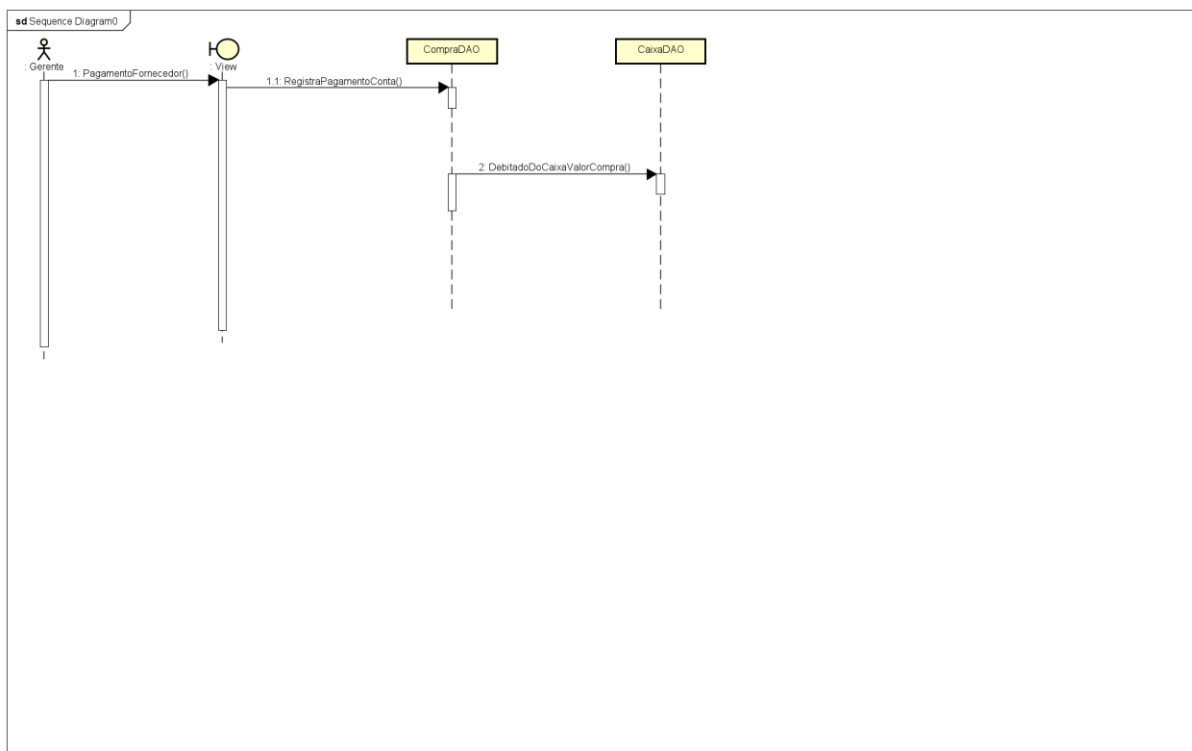
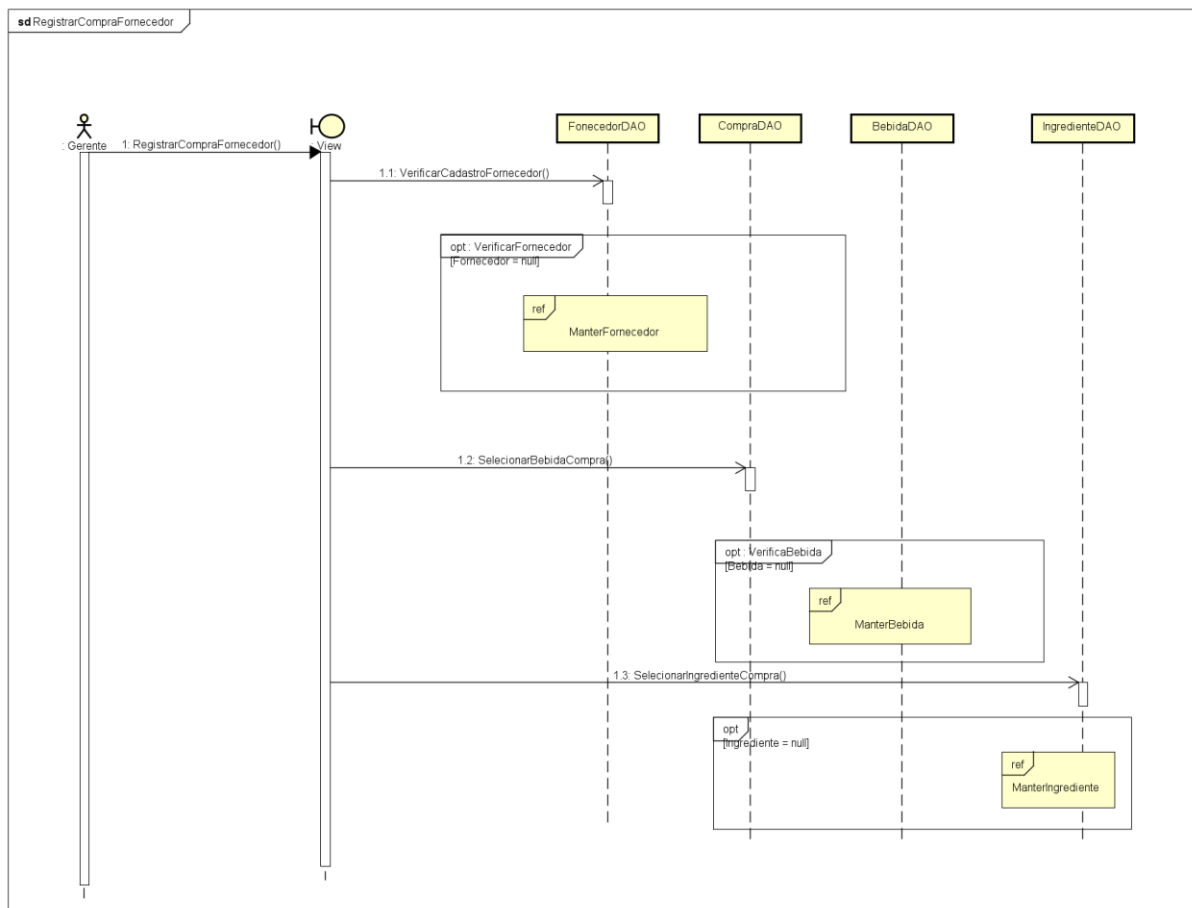


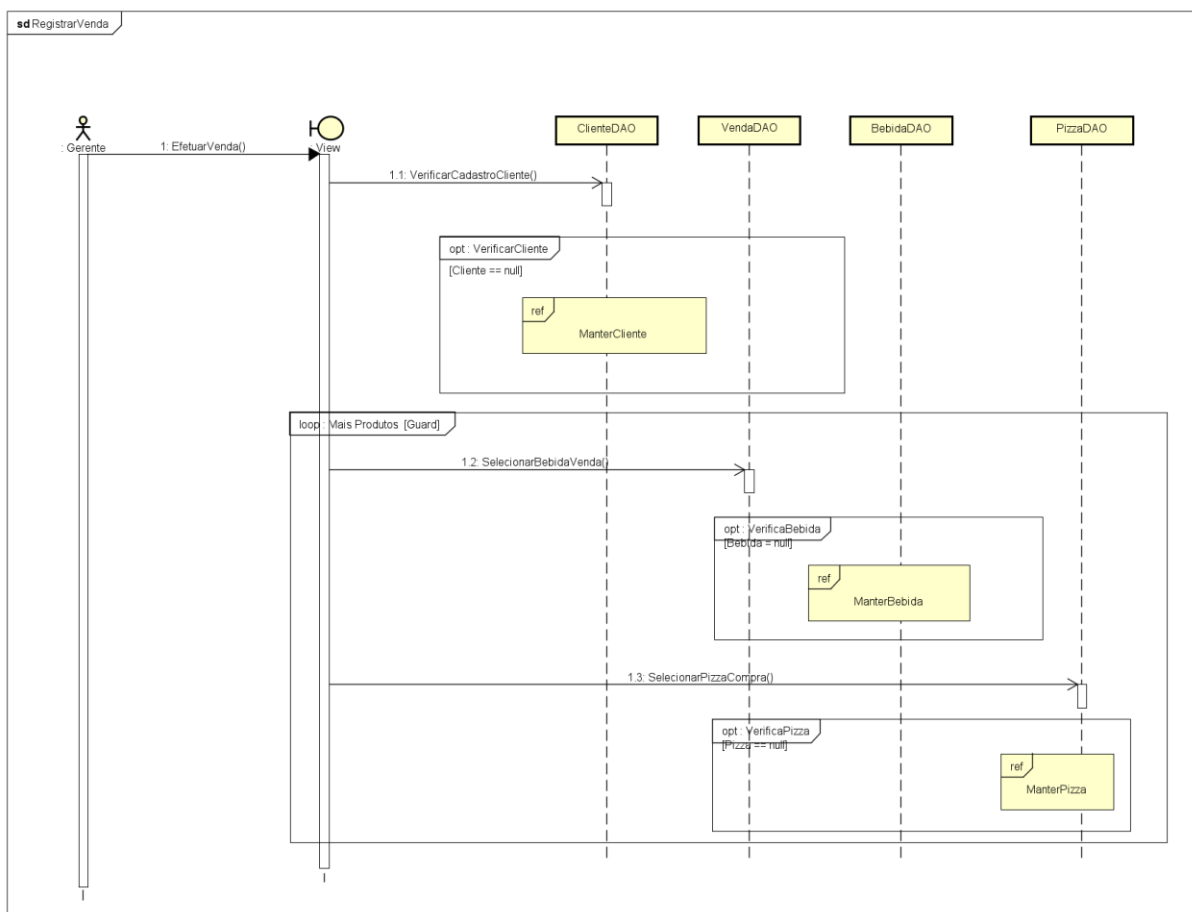
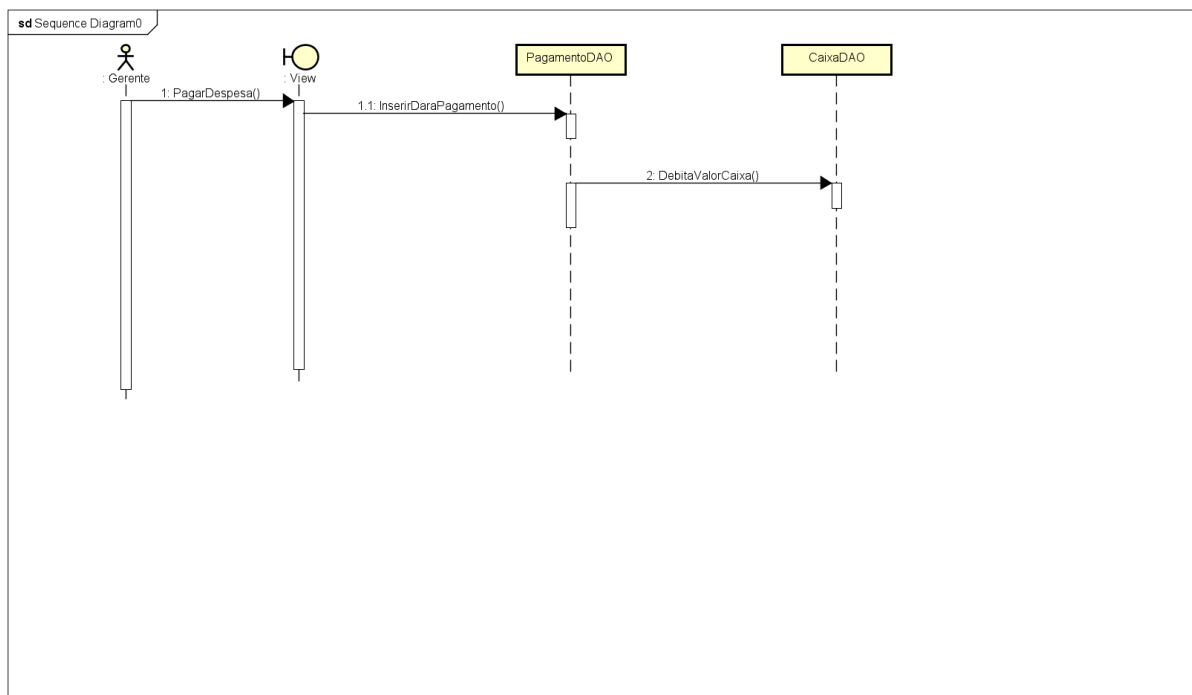
3.3 Diagramas de Sequência

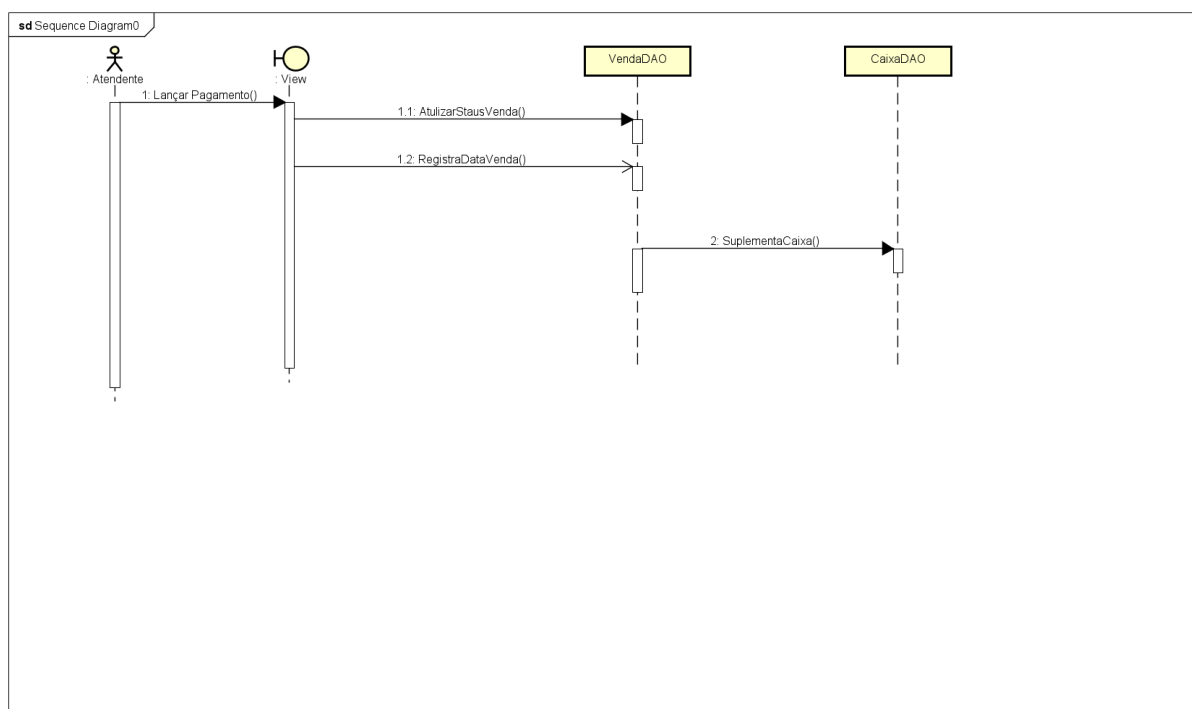
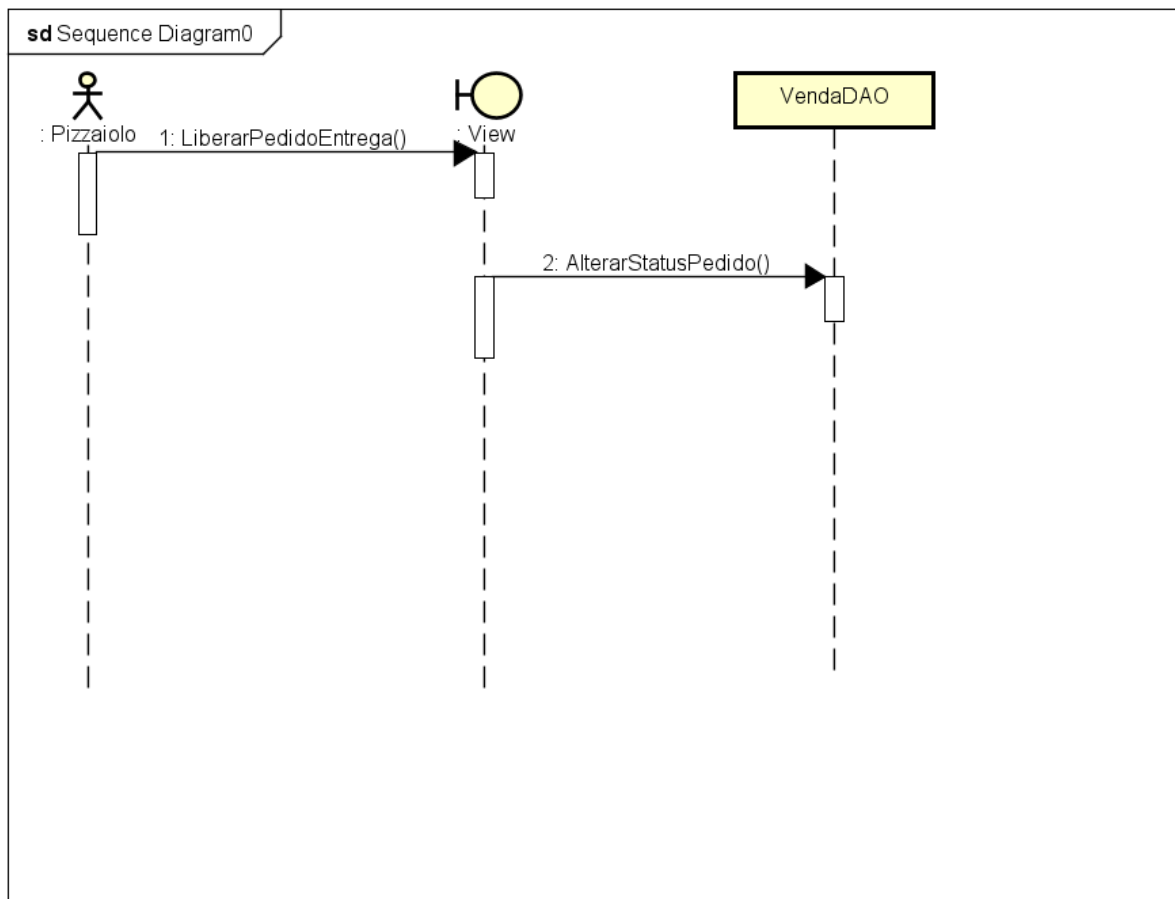


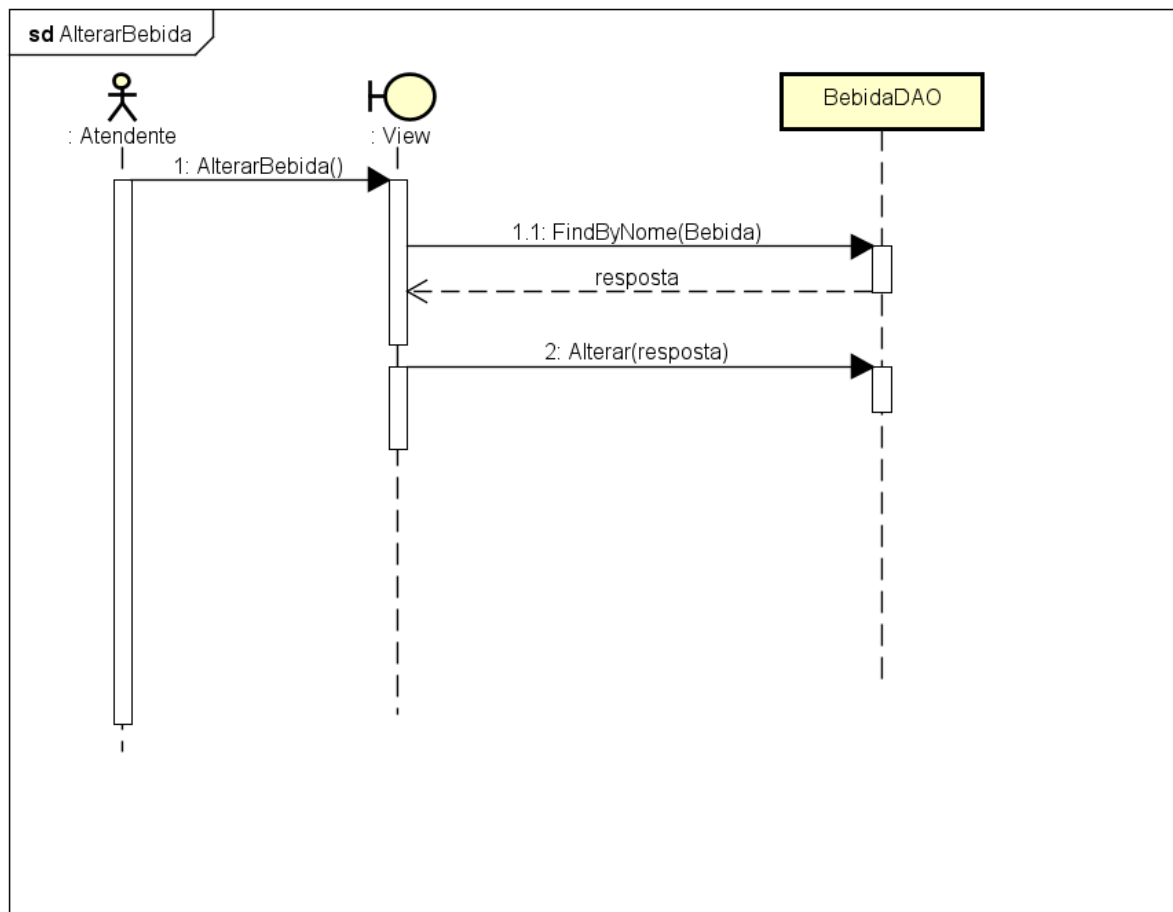


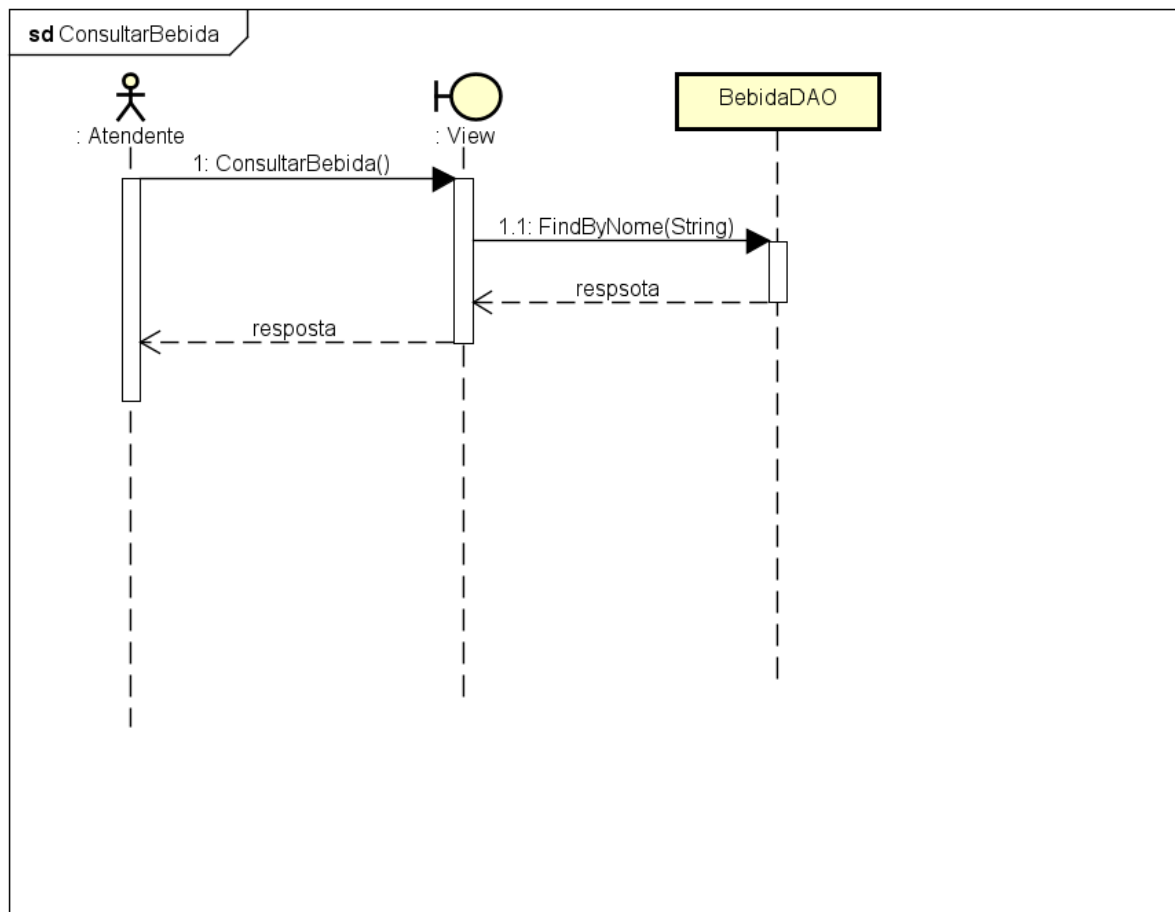


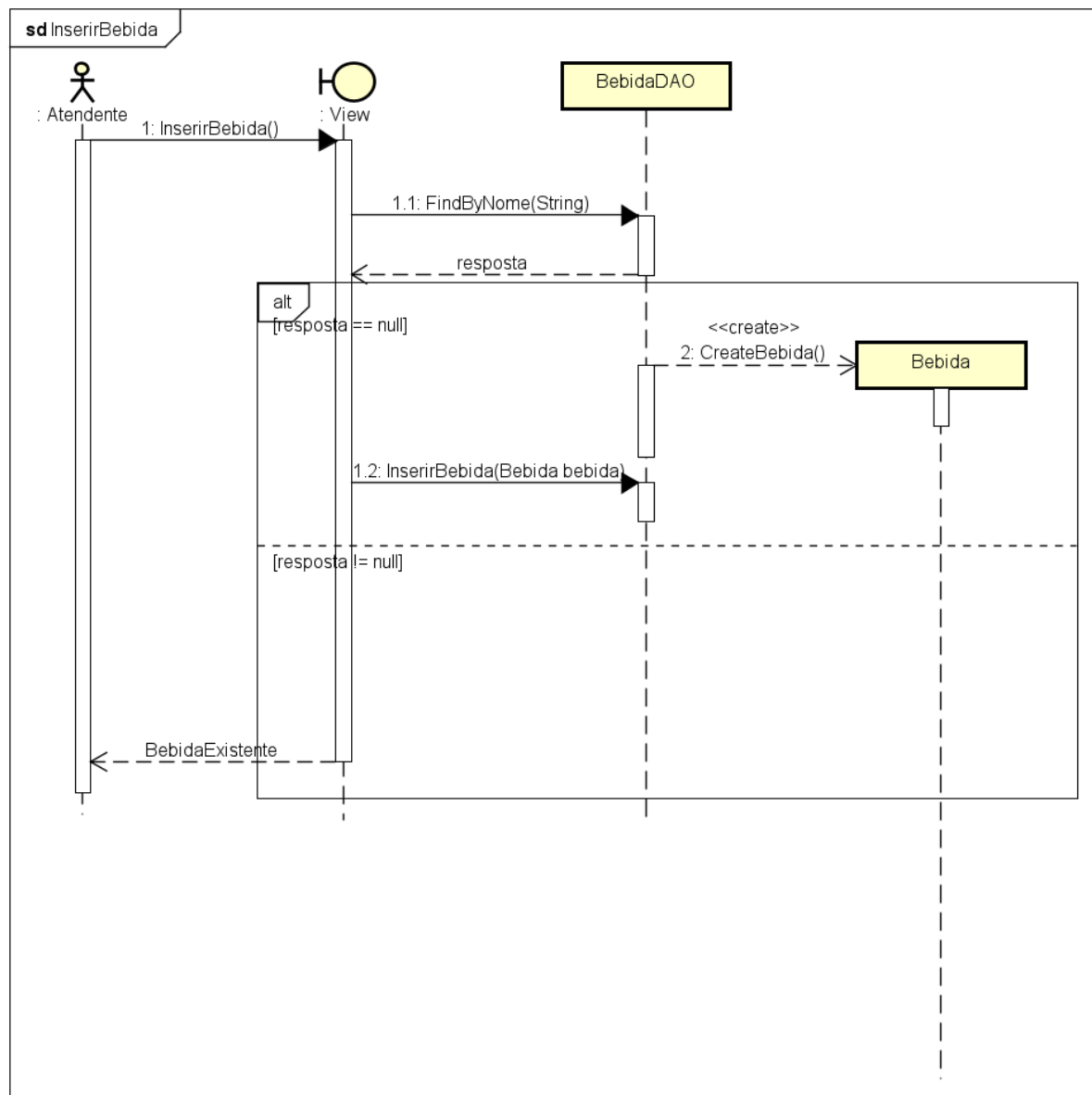


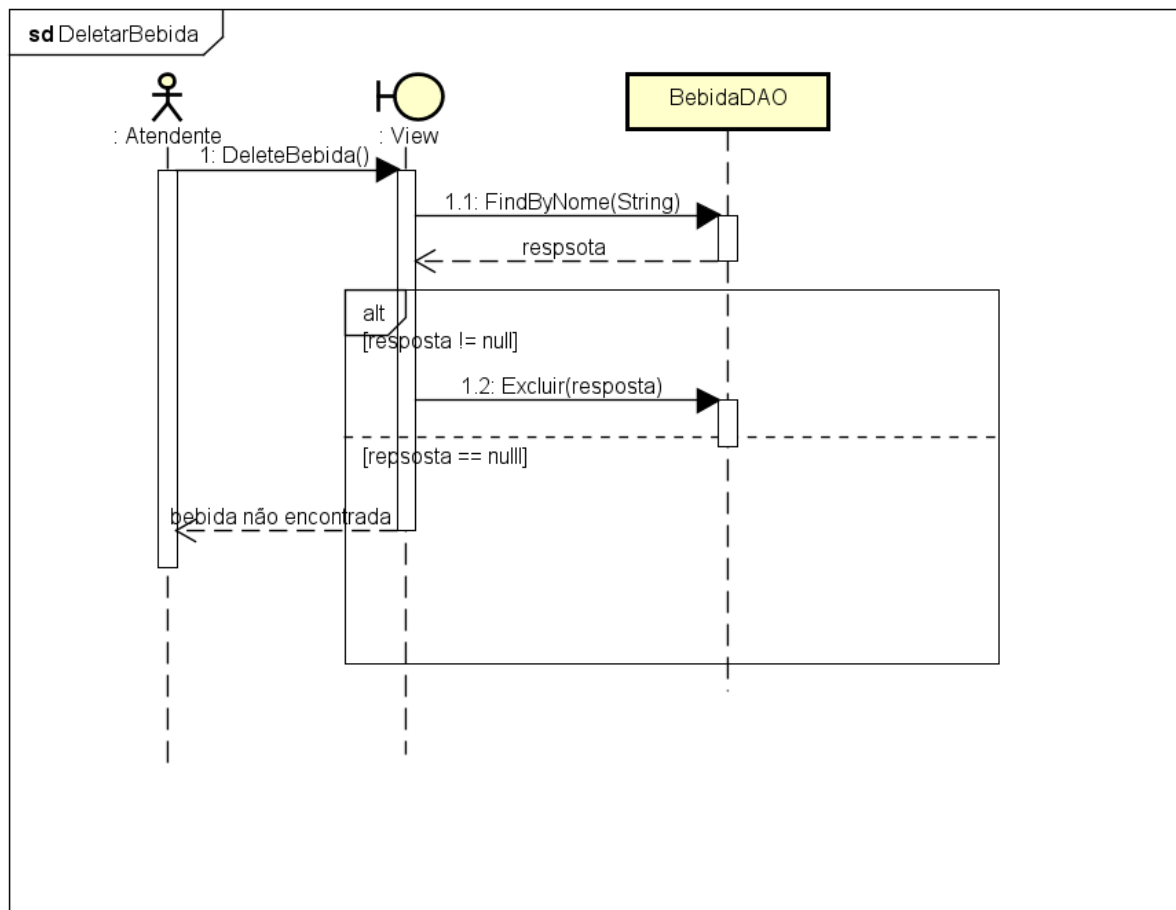












3.4 Mapeamento OO-Relacional (Modelo Lógico de Dados)

Bebida = (PK_codigo, nome, preco, quantidade)

BebidaCompra = (PK_compra, PK_bebida, quantidade, subtotal)

BebidaVenda = (PK_venda, PK_bebida, quantidade, subtotal)

Caixa = (PK_codigo, FK_usuario, dataAbertura, dataFechamento, horarioAbertura, horarioFechamento, valorInicial, valorFinal)

Cliente = (PK_codigo, nome, endereco, cidade, estado, cpf, email, contato, cupom)

Compra = (PK_codigo, FK_caixa, FK_fornecedor, data, dataPagamento, dataVencimento, total)

Despesa = (PK_codigo, nome, descricao, valor, dataVencimento, status)

Entrega = (PK_cliente, PK_venda, status, endereco, bairro, taxa, FK_entregador)

Entregador = (PK_codigo, nome, endereco, cidade, estado, cpf, email, contato)

Fornecedor = (PK_codigo, nome, endereco, cidade, estado, cnpj, email, contato)

Ingrediente = (nome)

IngredienteCompra = (quantidade, subtotal)

IngredientePizza = (PK_pizza, PK_ingrediente)

Movimentacao = (PK_ingrediente, PK_compra, PK_codigo, FK_caixa, valor, motivo, tipo)

Pagamento = (PK_codigo, FK_venda, valor, forma)

PagamentoDespesa = (dataPagamento)

PagamentoEntregador = (PK_codigo, entregador, valor, data)

Pizza = (PK_codigo, nome, tamanho, preco)

PizzaVenda = (PK_pizza, PK_venda, FK_borda, quantidade, subtotal, descricao)

Usuario = (PK_codigo, nome, dataNascimento, email, senha, endereco, cidade, estado, contato, cargo)

Venda = (PK_codigo, FK_caixa, FK_cliente, data, total, desconto, status)