

UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE GRADUAÇÃO CIÊNCIA DA COMPUTAÇÃO

NADSON SILVA TIMBÓ

**MODELAGEM DE UM SISTEMA DE INFORMAÇÃO PARA GESTÃO DA
PRODUÇÃO DE IOGURTE NA INDÚSTRIA SABOR LATICÍNIOS DO
NORDESTE**

São Luís
2013

NADSON SILVA TIMBÓ

**MODELAGEM DE UM SISTEMA DE INFORMAÇÃO PARA GESTÃO DA
PRODUÇÃO DE IOGURTE NA INDÚSTRIA SABOR LATICÍNIOS DO
NORDESTE**

Monografia apresentada ao curso de
Ciência da Computação da
Universidade Federal do Maranhão,
como parte dos requisitos para a
obtenção do grau de Bacharel em
Ciência da Computação.

Orientador: Prof.º Dr. Sofiane Labidi

São Luís

2013

Timbó, Nadson Silva

Modelagem de um Sistema de Informação para Gestão da
Produção de logurte na Indústria Sabor Laticínios do Nordeste
/ Nadson Silva Timbó. – São Luís, 2013.

148 f.

Impresso por computador (Fotocópia)

Orientador: Prof.º Dr. Sofiane Labidi

Monografia (Graduação) – Universidade Federal do
Maranhão, Curso de Ciência da computação, 2013.

1. Sistema – Modelagem 2. Astah Professional 3. UML 4.

Objeto - Orientação I. Título.

CDU 004.414.23

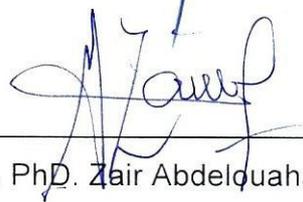
**MODELAGEM DE UM SISTEMA DE INFORMAÇÃO PARA GESTÃO DA
PRODUÇÃO DE IOGURTE NA INDÚSTRIA SABOR LATICÍNIOS DO
NORDESTE**

Aprovado em 26/06/2013

BANCA EXAMINADORA



Prof. Dr. Sofiane Labidi.
(Orientador)



Prof. PhD. Zair Abdelquahab



Prof. Dr. Nilson Santos Costa

AGRADECIMENTOS

A Deus por me dar força, saúde e coragem para enfrentar os momentos mais difíceis deste caminho.

Aos meus pais (Francisco Napoleão Camilo Timbó e Marilene Silva Timbó), irmão (Naylson Silva Timbó) que sempre me apoiaram e acreditaram em mim. familiares, que com muito carinho e apoio, não mediram esforços para que eu chegasse até esta etapa de minha vida. Esta vitória é nossa.

A minha namorada, Raissa Ferreira do Nascimento, por todo o amor, amizade, carinho, companheirismo e compreensão.

Ao Professor, orientador e amigo, Dr. Sofiane Labidi, por suas sábias palavras, seu apoio, incentivo e pela honra de poder ser seu aprendiz.

Ao Professor, Dr. Nilson Santos Costa nesta jornada pela dedicação, humildade e paciência, que me levaram a execução e conclusão desta Dissertação.

E em especial para o doutorando, amigo e “braço direito” Christian Diniz pelo acompanhamento, paciência, motivação. E por fim para Szana Grijó que sempre me motivou e auxiliou nas dúvidas.

*“...quem não vive pra servir,
não serve pra viver...”*

RESUMO

O Sistema de Produção representa um módulo de um sistema desatualizado, cujo foco é gerir todo o processo de produção de iogurte na indústria de laticínios, separando de outros módulos, tornando-o independente e autossuficiente. Para o desenvolvimento do sistema se faz necessário à utilização de uma modelagem UML visando apresentar uma visão, especificação, conceituação e documentação do sistema. A análise e captura de requisitos é necessária para a modelagem, feita através da ferramenta Astah Professional, sendo uma base sólida para a equipe de desenvolvimento. Demonstrando a necessidade de migração de sistemas desatualizados para uma arquitetura orientada a objeto.

Palavras-chave: Astah professional, UML, Orientação a objeto.

ABSTRACT

The production system represents an outdated module, which focus is to manage the entire production process of yogurt on the dairy industry, separating from others modules, rendering it as independent and self-sufficient. For the development of the system it is necessary use a modeling UML in order to present a vision, specification, conceptualization and system documentation. The analysis and requirements understanding is necessary for modeling, taken through the tool Astah Professional, being a solid base for the development team. Demonstrating the outdated migration need for an object-oriented architecture.

Key-words: Astah professional, UML, Object-oriented.

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 - Relacionamento de interdependência entre organização e sistema de informação | 27 |
| Figura 2 - Diagramas da UML | 39 |
| Figura 3 - Representação de um ator | 40 |
| Figura 4 - Representação de um caso de Uso | 41 |
| Figura 5 - Representação de uma associação..... | 41 |
| Figura 6 - Representação de herança..... | 42 |
| Figura 7 - Representação de uma inclusão..... | 42 |
| Figura 8 - Representação de um extend..... | 43 |
| Figura 9 - Representação de uma restrição..... | 43 |
| Figura 10 - Representação de multiplicidade..... | 43 |
| Figura 11 - Representação de uma classe..... | 44 |
| Figura 12 - Representação de associação unária..... | 45 |
| Figura 13 - Representação de associação binária..... | 45 |
| Figura 14 - Representação de uma associação ternária..... | 46 |
| Figura 15 - Representação de agregação..... | 46 |
| Figura 16 - Representação de composição..... | 46 |
| Figura 17 - Representação de Generalização/Especialização..... | 47 |
| Figura 18 - Representação de classe associativa..... | 47 |
| Figura 19 - Representação de restrição..... | 48 |
| Figura 20 - Representação de estereótipos..... | 49 |
| Figura 21 - Representação Objeto..... | 51 |
| Figura 22 - Representação de Vínculo..... | 51 |
| Figura 23 - Elementos do diagrama de sequência..... | 53 |
| Figura 24 - Fragmento de interação e operador de interação..... | 54 |
| Figura 25 - Componentes do diagrama de atividade..... | 57 |
| Figura 26 - Ferramenta Astah Professional..... | 61 |
| Figura 27 - Organograma..... | 70 |
| Figura 28 - Sistema desatualizado da Sabor Laticínios..... | 87 |
| Figura 29 - Solicitação de produção com nova modelagem..... | 88 |

Figura 30 - Abertura de solicitação.....89

LISTA DE TABELAS

| | |
|--|----|
| Tabela 1 - Tabela de produção de maio..... | 72 |
|--|----|

SUMÁRIO

| | |
|--|----|
| AGRADECIMENTOS..... | 5 |
| RESUMO..... | 7 |
| ABSTRACT..... | 8 |
| LISTA DE FIGURAS..... | 9 |
| LISTA DE TABELAS..... | 11 |
| SUMÁRIO..... | 12 |
| 1 INTRODUÇÃO..... | 16 |
| 1.1 Contextualização e Problemática..... | 16 |
| 1.1 Justificativa: importância do sistema WEB..... | 18 |
| 1.2 Objetivos..... | 20 |
| 1.2.1 Geral..... | 20 |
| 1.2.2 Específico..... | 20 |
| 2 ESTADO DA ARTE..... | 20 |
| 2.1 Sistemas de Informações gerenciais..... | 20 |
| 2.2 Sistema de informação para a Web..... | 24 |
| 2.3 Sistemas de informações nas organizações..... | 26 |
| 2.4 Orientação a Objetos..... | 29 |
| 2.4.1 Origem..... | 29 |
| 2.4.2 Conceitos..... | 30 |
| 2.5 UML: Unified Modeling Language..... | 33 |
| 2.5.1 Visão Geral da UML..... | 33 |
| 2.5.2 Modelagem com UML..... | 36 |
| 2.5.3 Diagramas da UML..... | 40 |
| 2.5.3.1 Diagramas de casos de uso..... | 40 |
| 2.5.3.2 Diagramas de classes..... | 44 |
| 2.5.3.3 Diagramas de objetos..... | 50 |
| 2.5.3.4 Diagramas de sequência..... | 52 |
| 2.5.3.5 Diagramas de atividade..... | 55 |
| 2.5.4 UML 2.0..... | 59 |
| 2.6 FERRAMENTA ASTAH PROFESSIONAL..... | 60 |

| | | |
|-----|--|-----|
| 3 | DESCRIÇÃO DO NEGÓCIO..... | 63 |
| 3.1 | A Empresa..... | 63 |
| 3.2 | Funcionamento..... | 65 |
| 3.3 | Estrutura organizacional..... | 67 |
| 3.4 | O processo de produção..... | 70 |
| 4 | MODELAGEM DO SISTEMA DE PRODUÇÃO..... | 73 |
| 4.1 | Sistema de controle de produção (apresentação)..... | 73 |
| 4.2 | Diagrama de caos de uso..... | 77 |
| 4.4 | Diagrama de classe..... | 78 |
| 4.3 | Diagrama de sequência..... | 80 |
| 4.5 | Diagrama de atividade..... | 83 |
| 5 | CONCLUSÃO..... | 86 |
| | REFERÊNCIAS..... | 90 |
| | APÊNDICES..... | 93 |
| | APÊNDICE A: Diagrama de caso de uso do Sistema de Produção..... | 93 |
| | APÊNDICE B: Descrição do caso de uso Selecionar Insumos Líquido | 94 |
| | APÊNDICE C: Descrição do caso de uso Selecionar Fórmula..... | 95 |
| | APÊNDICE D: Descrição do caso de uso Selecionar Capacidade..... | 96 |
| | APÊNDICE E: Descrição do caso de uso Registrar Avaria Líquido..... | 97 |
| | APÊNDICE F: Descrição do caso de uso Recuperar Código de Saída..... | 98 |
| | APÊNDICE G: Descrição do caso de uso Movimentar Saída Produto..... | 99 |
| | APÊNDICE H: Descrição do caso de uso Movimentar Saída Insumo Embalagem..... | 100 |
| | APÊNDICE I: Descrição do caso de uso Movimentar Entradas..... | 101 |
| | APÊNDICE J: Descrição do caso de uso Manter Unidade..... | 102 |
| | APÊNDICE K: Descrição do caso de uso Manter Produto..... | 103 |
| | APÊNDICE L: Descrição do caso de uso Manter Fórmula..... | 104 |
| | APÊNDICE M: Descrição do caso de uso Manter Estoque Mínimo..... | 105 |
| | APÊNDICE N: Descrição do caso de uso Gerar Código de Saída..... | 106 |
| | APÊNDICE O: Descrição do caso de uso Finalizar Solicitação..... | 107 |
| | APÊNDICE P: Descrição do caso de uso Definir Tipo..... | 108 |
| | APÊNDICE Q: Descrição do caso de uso Calcular Embalagem..... | 109 |
| | APÊNDICE R: Descrição do caso de uso Cadastrar Real Produzido..... | 110 |
| | APÊNDICE S: Descrição do caso de uso Associar Embalagem..... | 111 |

| | |
|--|-----|
| APÊNDICE T: Descrição do caso de uso Abrir Solicitação Produção..... | 112 |
| APÊNDICE U: Descrição do caso de uso Validar Estoque Insumos..... | 113 |
| APÊNDICE V: Diagrama de classes Modelo Conceitual..... | 114 |
| APÊNDICE W: Diagrama de classes Modelo Projeto..... | 115 |
| APÊNDICE X: Diagrama de Sequência Fazer Login..... | 116 |
| APÊNDICE Y: Diagrama de Sequência Cadastrar Capacidade..... | 117 |
| APÊNDICE Z: Diagrama de Sequência Buscar Capacidade..... | 118 |
| APÊNDICE AA: Diagrama de Sequência Cadastrar Produto..... | 119 |
| APÊNDICE BB: Diagrama de Sequência Associar Embalagem..... | 120 |
| APÊNDICE CC: Diagrama de Sequência Manter Estoque Mínimo..... | 121 |
| APÊNDICE DD: Diagrama de Sequência Buscar Produto..... | 122 |
| APÊNDICE EE: Diagrama de Sequência Editar Produto..... | 123 |
| APÊNDICE FF: Diagrama de Sequência Excluir Produto..... | 124 |
| APÊNDICE GG: Diagrama de Sequência Cadastrar Nota Fiscal..... | 125 |
| APÊNDICE HH: Diagrama de Sequência Cadastrar Despesa..... | 126 |
| APÊNDICE II: Diagrama de Sequência Editar Nota Fiscal..... | 127 |
| APÊNDICE JJ: Diagrama de Sequência Buscar Nota Fiscal..... | 128 |
| APÊNDICE KK: Diagrama de Sequência Excluir Nota Fiscal..... | 129 |
| APÊNDICE LL: Diagrama de Sequência Movimentar Entrada..... | 130 |
| APÊNDICE MM: Diagrama de Sequência Gerar Código de Entrada..... | 131 |
| APÊNDICE NN: Diagrama de Sequência Criar Fórmula..... | 132 |
| APÊNDICE OO: Diagrama de Sequência Buscar Fórmula..... | 133 |
| APÊNDICE PP: Diagrama de Sequência Editar Fórmula..... | 134 |
| APÊNDICE QQ: Diagrama de Sequência Abrir Solicitação de Produção..... | 135 |
| APÊNDICE RR: Diagrama de Sequência Validar Estoque..... | 136 |
| APÊNDICE SS: Diagrama de Sequência Movimentar Saída Insumo/Embalagem..... | 137 |
| APÊNDICE TT: Diagrama de Sequência Finalizar Solicitação..... | 138 |
| APÊNDICE UU: Diagrama de Sequência Buscar Solicitação..... | 139 |
| APÊNDICE VV: Diagrama de Sequência Cadastrar Real Produzido..... | 140 |
| APÊNDICE WW: Diagrama de Sequência Registrar Avaria..... | 141 |
| APÊNDICE XX: Diagrama de Sequência Recuperar Código de Saída..... | 142 |
| APÊNDICE YY: Diagrama de Atividade Cadastrar Produto..... | 143 |

| | |
|--|-----|
| APÊNDICE ZZ: Diagrama de Atividade Associar Embalagem..... | 144 |
| APÊNDICE AAA: Diagrama de Atividade Criar Fórmula..... | 145 |
| APÊNDICE BBB: Diagrama de Atividade Abrir Solicitação..... | 146 |
| APÊNDICE CCC: Diagrama de Atividade Selecionar Fórmula..... | 147 |
| APÊNDICE DDD: Diagrama de Atividade Finalizar Solicitação..... | 148 |
| APÊNDICE EEE: Diagrama de Atividade Cadastrar Real Produzido..... | 149 |
| APÊNDICE FFF: Diagrama de Atividade Recuperar Código de Saída..... | 151 |
| APÊNDICE GGG: Diagrama de Atividade Movimentar Entrada..... | 152 |

1 INTRODUÇÃO

1.1 Contextualização e problemática

Os computadores vêm ganhando mais espaço, principalmente nas empresas, sendo fonte de solução de problemas em conjunto com licenças de programas, pessoal qualificado, sistemas de telecomunicações, treinamentos, entre outros. Dessa forma, podemos conceituar Tecnologia da Informação (TI) como um conjunto de recursos tecnológicos e computacionais para a geração da informação, para seu uso e para o conhecimento (REZENDE, 2003). Nessa área há um investimento por várias razões, como exemplo, o aumento da produtividade, o posicionamento no mercado, a tentativa de ir ao encontro do cliente, manter competitividade, aumento na eficiência operacional, fornecimento de melhores informações para o gerenciamento, ganho de vantagens competitivas, aumento da receita/lucro, redução de custos e outros (SUWARDY, 2003).

Schaicoski (2002) responsabiliza a TI pelo sucesso ou fracasso de uma empresa, onde deve contribuir de forma a torná-la ágil, flexível e capaz de se adaptar aos revezes do cenário econômico, possuindo um papel estratégico, sustentando as operações de negócios existentes e viabilizando novas estratégias empresariais.

Segundo dados divulgados pela 23ª Pesquisa Anual de Tecnologia da Informação (2012), divulgada pelo Centro de Tecnologia de Informação Aplicada da Escola de Administração de Empresas de São Paulo, da Fundação Getúlio Vargas (FGV) no Brasil os investimentos na área da TI aumentaram, passando a ter um equipamento para cada dois habitantes. A estimativa é que o setor mantenha uma taxa de crescimento anual na faixa de 10% até 2014 (UOL ECONOMIA, 2011).

Com a velocidade que a tecnologia se desenvolve, é praticamente impossível imaginar o funcionamento de uma organização sem a existência de uma tecnologia de software que suporte as regras de negócios propostas pela empresa. No entanto, geralmente as atualizações não são simples e costumam ser grandes, podendo ser resultantes do cumprimento de leis, de mudanças na

economia nacional e internacional, alterações de leis, mudanças de gestão, de proprietário, de diretor, de custos e otimização de processos, visando sempre permanecer no mercado competitivo de forma mais agressiva.

Normalmente, empresas que possuem um grande conjunto de regras de negócio costumam investir muito dinheiro em sistemas de software e, para que haja um retorno financeiro, esse sistema é usado durante vários anos, mesmo ciente que o tempo de vida varia de sistema para sistema e de empresa para empresa (SOMMERVILLE, 2003).

1.2 Justificativa: importância do sistema WEB

O mercado vem sofrendo grandes mudanças, sendo atacado de informações e avanços tecnológicos, gerando assim novos comportamentos e desafios. Com essas mudanças, os clientes começaram a desenvolver maior sensibilidade quanto ao gosto e qualidade, resultando em uma autoanálise e revisão nas regras de negócios dos empreendedores, ou seja, uma reengenharia. A partir disso, inicia-se um processo de terceirização dos serviços, onde se obtém melhores resultados com serviços realizados por especialistas na área, que adotam melhores práticas, e acabam analisando o desempenho das empresas de nível mundial, assim como o seu próprio desempenho.

As empresas passaram a dar mais valor aos seus clientes, preocupando-se em construir um relacionamento duradouro e lucrativo com os mesmos. Visando chamar a atenção e fidelidade do cliente, as empresas repensaram suas filosofias, conceitos e ferramentas. Com enfoque nas ferramentas, novos sistemas de gerenciamento, controle de estoque, de vendas e de ordem de serviço, foram inseridos, pois até então não estavam atendendo as necessidades (velocidade e qualidade), sendo assim, começaram a aparecer novos métodos de desenvolvimento e modelagem que levaram as empresas a adquirirem sistemas que atendessem suas expectativas.

A decisão de manter um sistema desatualizado e não funcional acarreta algumas dificuldades para a empresa, como exemplo: pode-se assumir que gerou uma obsolescência das ferramentas e linguagem utilizada, além da dificuldade em encontrar profissionais com conhecimento especializado, requer gastos altos com manutenção, principalmente com hardware obsoleto e por fim pode citar a difícil de integração com novos sistemas, devido ao uso de tecnologias incompatíveis, tudo isso levaria qualquer mudança na regra de negócio a um enorme e demorado quebra cabeça.

Levando em consideração que um novo sistema atenderá melhor as necessidades e regras de negócios, acompanhando as evoluções tanto da empresa como da tecnologia, a empresa possuirá uma documentação apropriada e concreta para melhor entendimento futuro, um equilíbrio entre conhecimento da ferramenta e custo de manutenção. Tendo em vista que o objetivo empresarial é,

acima de tudo, continuar presente no mercado, geralmente decidi-se adotar em back-end, uma estratégia e uma política de modernização em seus sistemas.

Com a evolução da tecnologia Web, os sistemas passaram a ter um foco maior na “Era Web”, onde saíram de desktop para *Web Based* (sistemas baseados em web - executam em um browser), onde ganharam mais espaço e sofreram mudanças em alguns pontos. Logo, houve modificações na portabilidade, infraestrutura e segurança. Na portabilidade, devido o sistema possuir acesso de fora da empresa e encontrar-se disponível 24 horas por dia, podendo ser acessados em qualquer lugar que possua internet. A mudança na infraestrutura levou as empresas a não dependerem de servidores e de aquisição a hardware/software, reduzindo custos, pois o serviço seria terceirizado, sendo necessária somente conexão com a internet. A modificação na segurança se dá através de novas propostas como backup passando a ser automatizada pelo servidor, pois antes eram manuais e eventuais falhas no equipamento poderiam colocá-la em risco, sendo preparado para 100% de disponibilidade, ou próximo disso.

1.3 Objetivos

1.3.1 Geral

- Gerar uma modelagem para apoiar o desenvolvimento de um software, em que o fornecedor se responsabiliza por toda a estrutura necessária para a disponibilização do sistema e o cliente utiliza o software via internet.

1.3.2 Específico

- Analisar, ponderar e apresentar uma modelagem de um dos módulos do software desatualizado e não funcional de controle de estoque e operacional (módulo de produção);
- Gerar um software web based no âmbito estratégico, voltado para a produção, caracterizado por ser robusto e com possibilidade de evolução;
- Apresentar diagramas que compõem a linguagem de modelagem unificada (UML - Unified Modeling Language), visando auxílio na concepção de ideias referentes ao projeto;
- Apresentar por meio de diagramas as ideias de forma clara e objetiva;
- Utilizar a UML como apoio a um processo de desenvolvimento de sistema orientado a objeto;
- Proporcionar interação de toda a equipe, consolidando e documentando comportamental e estruturalmente;
- Capturar diferentes visões, dominando a complexidade do sistema, delimitando o escopo do problema, apoiado nas tomadas de soluções.

2. ESTADO DA ARTE

2.1 Sistemas de informações gerenciais

É de suma importância para uma empresa possuir e manipular dados para extrair conhecimento, passando a ser um diferencial nas negociações. Logo, os sistemas de informações surgem com essa finalidade e com intuito gerencial.

Segundo Davenport e Prusak (1999) para que se obtenha sucesso, as organizações precisam saber definir o conceito de dados, informações e conhecimento, pois a tomada de decisões e solução de problemas baseia-se nesses elementos. Os dados representam o elemento bruto, no qual precisa ser analisado. Oliveira (2002, p. 51) relata que “dado é qualquer elemento identificado em sua forma bruta que, por si só, não conduz a uma compreensão de determinado fato ou situação”.

O conceito de informação para Oliveira (1992) é apresentado de forma estruturada e de muita importância, auxiliando no processo decisório. Também pode ser considerado como um dado já processado e armazenado de forma lógica para que o receptor possa efetuar uma decisão (PADOVEZE, 2000). Por fim, Laudon e Laudon (1999, p. 10) afirmam, que o conhecimento “é o conjunto de ferramentas conceituais e categorias usadas pelos seres humanos para criar, colecionar, armazenar e compartilhar a informação”.

Depois de obter a definição de dado, informação e conhecimento, o conceito de sistema fica mais intuitivo. O biólogo alemão Ludwig Von Bertalanffy (2009, p. 63) no início dos estudos da década de 1920, define um sistema como um "conjunto de elementos de interação", logo podemos afirmar que todo sistema que manipule dado e gera uma informação, usando ou não a tecnologia, é considerado um sistema de informação (SI).

Para Laudon e Laudon:

“Um sistema de informação pode ser definido como um conjunto de componentes inter-relacionados trabalhando juntos para coletar, recuperar, processar, armazenar e distribuir informações com a finalidade de facilitar o planejamento, o controle, a coordenação, a análise e o processo decisório em organizações” (1999, p. 4).

Os fundamentos do SI são baseados nas seguintes atividades: entrada dos dados, o processamento com responsabilidade de transformar a entrada em uma saída desejada, útil e apropriada, através de tomada de decisões e por fim a saída representando o resultado desse processamento, com auxílio do computador, vídeo, documentos impressos ou digitalizados, relatórios, gráficos, cálculos para uso gerencial.

O objetivo dos SIs focam na resolução de problemas internos, visando uma preparação para tendências no mercado, dando condições para que as empresas consigam reagir, possuindo uma base forte no processo de decisão. Laudon e Laudon (1999, p. 26), afirmam que “a razão mais forte pelas quais as empresas constroem os sistemas, então, é para resolver problemas organizacionais e para reagir a uma mudança no ambiente”.

Segundo O'brien (2002, p.29), “quando os sistemas de informação se concentram em fornecer informação e apoio à tomada de decisão eficaz pelos gerentes, eles são chamados sistemas de apoio gerencial”. Alguns exemplos são citados como o Sistema de Suporte da Decisão (SSD), e o Sistema de Informação Gerencial (SIG). .

Podemos conceituar SIG como um tipo de sistema ou processo de apoio gerencial com eficácia, que gera produtos de informação para apoiar muitas das necessidades diárias de tomada de decisão das organizações e são resultados da interação colaborativa entre pessoas, tecnologia e procedimentos, visando atingir as metas. Um SIG fornece informações em forma de relatórios e demonstrativos pré-estipulados para os gerentes. Podem incluir software que auxilia na tomada da decisão, sistemas especialistas, gestão de projetos, recursos de dados e processos informatizados que permitem um funcionamento eficiente.

Para Batista, sistema de informação gerencial:

“É o conjunto de tecnologias que disponibilizam os meios necessários à operação do processamento dos dados disponíveis. É um sistema voltado para a coleta, armazenagem, recuperação e processamento de informações usadas ou desejadas por um ou mais executivos no desempenho de suas atividades. É o processo de transformação de dados em informações que são utilizadas na estrutura decisória da empresa e proporcionam a sustentação administrativa para otimizar os resultados esperados” (2004, p. 22).

Uma das características nos SIGs é a existência da multiplicidade de produtos de informação através de relatórios resumidos que auxiliam na tomada de decisão, podendo ser obtidos pela filtragem e análise de dados altamente detalhados em banco de dados de processamento de transações e apresentação de resultados aos administradores de forma conivente, pode ser citado como exemplo:

- Relatórios Programados: são tradicionais, levando informação para os gerentes, podendo citar relatórios diários e semanais ou demonstrativos financeiros;
- Relatórios de execução: relatórios onde gerente pode obter informações específicas, ou seja, um relatório que contém informações apenas sobre clientes que excedam os limites de créditos;
- Informes e respostas por solicitação: mostra qualquer informação que o gerente requisitar, capaz de obter respostas imediatas através das estações de trabalho;
- Relatórios em pilhas: informações empilhadas na estação de trabalho em rede do gerente.

Todas as funções de gestão, planejamento, organização, direção e controle são necessários para o bom desempenho organizacional. Os SIGs são fundamentais para suportar essas funções, especialmente de planejamento, controle e parte integrante da decisão tomada.

Oliveira (2002) informa que os SIGs podem trazer alguns benefícios para as empresas, como a redução dos custos das operações, melhoria no acesso às informações, proporcionando relatórios mais precisos e rápidos, com menor esforço, melhoria na produtividade e nos serviços realizados e oferecidos, na tomada de decisões, por meio do fornecimento de informações mais rápidas e precisas, estímulo de maior interação dos tomadores de decisão, fornecimento de melhores projeções dos efeitos das decisões, melhoria na estrutura organizacional, para facilitar o fluxo de informações na estrutura de poder, proporcionando maior poder para que os profissionais controlem os sistemas, redução do grau de centralização de decisões na empresa e por fim, a melhoria na adaptação da empresa para enfrentar acontecimentos não previstos.

Em síntese, um SIG será desenvolvido para apoiar as metas, usando seus relatórios para comparar resultados e novas estratégias.

2.2 Sistemas de informação para a Web

Surgiu na década de 1990 uma nova tecnologia para internet, Tecnologia Web, que se define como o conjunto de padrões para comunicação, endereçamento e apresentação de informações com o objetivo de ser um repositório de conhecimento humano (BERNERS-LEE, 1994), relativamente simples, formado por documentos eletrônicos localizados em servidores ligados à rede mundial (Internet), podendo ser visualizado em qualquer computador ligado a mesma. Os documentos são chamados de páginas Web, formando uma grande rede de informações chamada *World Wide Web* ou simplesmente Web, facilitando vários serviços.

Seu crescimento causou um impacto significativo no comércio, indústria, bancos, finanças, educação e setores de entretenimento e na vida particular das pessoas (GINIGE & MURUGESAN, 2001). A Web continua evoluindo com novos recursos e funções, passando a ser a interface de acesso dos usuários a informações dinâmicas, sendo utilizada para interação de clientes e fornecedores, permitindo a universalização do acesso à informação, podendo ser acessada pelos navegadores web gerando novas oportunidades de negócios, fazendo com que todas as empresas estejam à mesma distância, aumentando o comércio eletrônico, possibilitando ser citado o comércio negócio-para-consumidor (B2C) e negócio-para-negócio (B2B).

A tecnologia Web estimulou desenvolvimentos em sistemas interorganizacionais, colaborativos e impactou os SIs na forma como são construídos e na maneira como são utilizados, sendo considerados por alguns, a nova geração de SI, introduzindo desafios gerenciais e técnicos (ISAKOWITZ, BIEBER e VITALI, 1998), sendo que seu sucesso dependerá do desenvolvimento.

Várias metodologias de desenvolvimento vêm sendo propostas, assim, a web passou a ser o principal caminho para implantação e desenvolvimento de serviços computacionais, como transações bancárias, divulgação e acesso de informações, softwares como serviços para gestão, sistemas web, site web entre outros, atingindo cada vez mais usuários, aumentando a demanda por sistemas web based, pois adiciona funcionalidade de negócio ao objetivo almejado.

Segundo Pressman (2002) o processo de sistemas baseados na web possui algumas características como o imediatismo, segurança e estética. O imediatismo requer prazos mais curtos para colocação do produto no mercado, havendo assim a necessidade de usar métodos para planejamento, projeto, implementação e testes. A segurança requer medidas adicionadas na infraestrutura para proteger conteúdo reservado ou como modo seguro de transmissão de dados. Por fim a estética responsável para chamar atenção do produto.

Os sistemas baseados na web possuem as seguintes vantagens em sua utilização (CONALLEN, 2003):

- Execução a partir de qualquer navegador web;
- Acesso de qualquer lugar do mundo, necessitando apenas de um computador conectado à internet e um navegador, sendo desnecessária a presença física na empresa, apresentando melhor produção;
- Interface HyperText Markup Language (HTML), onde muitos usuários já estão acostumados;
- Atualização de dados e informações em tempo real;
- Desenvolvimento, manutenção e atualização centralizada da aplicação, não sendo necessária a instalação em equipamentos diferentes, somente em um servidor, minimizando os custos, pois quaisquer mudanças serão feitas somente no servidor, estabilizando o processamento;
- Exportação de dados entre usuários remotos usando o protocolo HyperText Transfer Protocol (HTTP) ;
- Não é necessário um poder grande de memória e processadores, pois é executado no servidor;
- Para o funcionamento de uma aplicação web, sua arquitetura necessita de um servidor web, uma conexão de rede e um navegador web que permita que a regra de negócio seja executada de forma fácil.

2.3 Sistemas de informações nas organizações

Na visão atual reconhecemos que os conceitos e processos importantes da organização devem ser considerados e apoiados pelos SIs, pois são construídos com intuito de estruturar organizacionalmente o modo como atingir suas metas. Organização, de acordo com Lacombe (2003), é um conjunto de duas ou mais pessoas que se reúnem de forma estruturada e deliberada e em associação, para realizar tarefas, compondo metas para alcançarem objetivos planejados, de forma coordenada e controlada.

Completando com Moraes:

“Organizações são instituições sociais e a ação desenvolvida por membros é dirigida por objetivos. São projetadas como sistemas de atividades e autoridade, deliberadamente estruturados e coordenados, elas atuam de maneira interativa com o meio ambiente que a cerca” (2004, p. 91).

Organizações podem ser ou não com fins lucrativos, mas tem como objetivo, alcançar produtos e/ou serviços, metas e/ou objetivos. Visam atender uma necessidade específica, (MAXIMIANO, 2007) de forma mais eficaz e eficiente. Existem três características comuns a todas as organizações: comportamento, estrutura e processos, afinal, todas elas têm pessoas com comportamento humano que satisfazem necessidades, desenvolvem atitudes, motivam e comandam. São organizadas, possuem uma estrutura, crescem, ampliam-se, alteram-se e exercem alguma atividade, através de processos que comunicam e tomam decisões.

Os SIs tem como papel duas visões: visão externa e visão interna. A Visão externa controla e monitora com objetivo de assegurar eficiência e eficácia por meio da coleta de informações a respeito de cada processo. Sendo que a eficiência está ligada a otimização dos recursos para atingir melhor o objetivo com qualidade nos serviços e produtos, economia e sem prejuízos na produtividade. Enquanto eficácia refere-se a alcançar os resultados planejados, metas e/ou objetivos. Maximiano (2007, p. 11) defende que de acordo como são administradas as organizações, elas podem ser “eficientes e eficazes” ou “ineficientes e ineficazes”, podendo ser fonte de solução ou de problemas.

A Visão interna passa a ser uma visão contemporânea onde defende que os SIs estão entrelaçados com o processo de valor, sendo parte do processo, auxiliando na entrada, transformação do produto ou na saída.

As organizações possuem funções básicas como manufatura e produção, vendas e marketing, finanças, contabilidade e recursos humanos, sendo distribuídas dentro dos níveis organizacionais (estratégico, gerencial, conhecimento, operacional) que por fim estão distribuídos dentro das estruturas organizacionais (tradicional e matricial).

Os SIs e as organizações possuem um relacionamento de interdependência, onde as organizações detêm o negócio, as regras, procedimentos e estratégias e os SIs são tradutores e intérpretes delas, onde qualquer mudança na estratégia, regra e processo de negócios depende de seus sistemas e de sua flexibilidade, como mostra na figura 1.

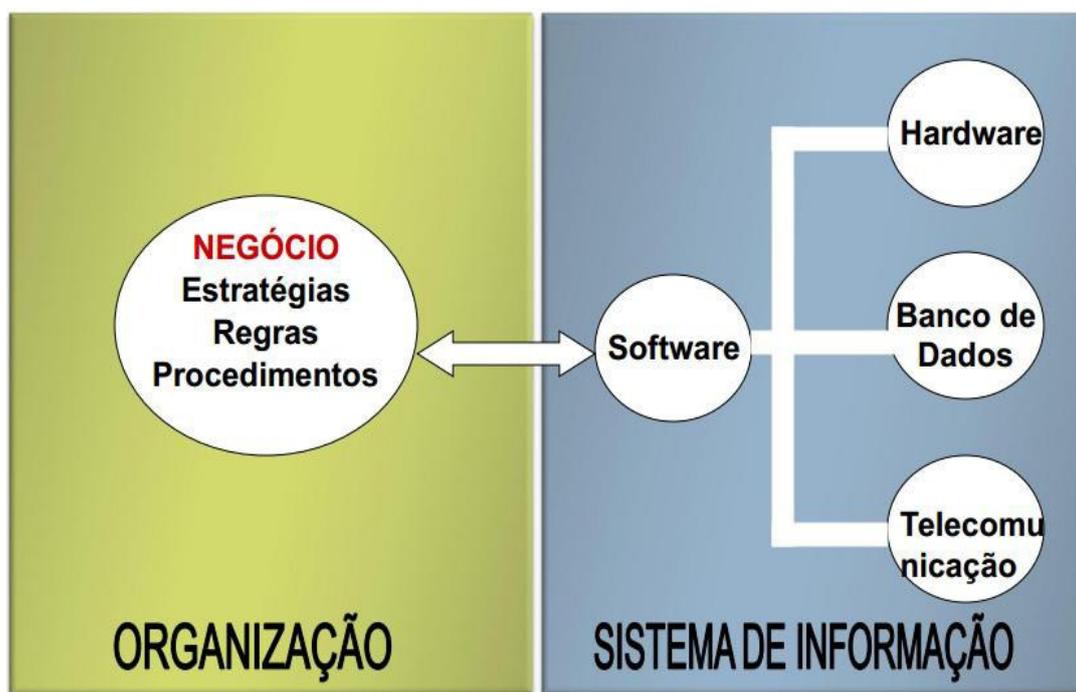


Figura 1 – Relacionamento de interdependência entre organização e sistema de informação

Os SIs tornaram-se mais presentes nas organizações, gerando transformação no modelo e na forma das empresas, criando a base para o comércio e empresas digitais, principalmente devido à revolução da rede e da internet, pois trouxe um ambiente extremamente elástico, tolerante a falhas e de alta escalabilidade. A consequência dessa transformação resultou em um achatamento das organizações, diminuindo o número de funcionários e burocracia, gerando níveis hierárquicos, desvinculando o trabalho de uma

localização, dando-lhes flexibilidade, pois as empresas grandes podem agir como pequenas e vice-versa, redefinindo as fronteiras organizacionais.

Portanto os SIs fazem parte da organização, devendo possuir uma estrutura definida sobre a qual irá desenvolver suas funções para atingir seus objetivos. Logo, administradores não podem ignorá-los, pois afetam na tomada de decisão e as evoluções organizacionais dependem mais da evolução dos SIs, devido gerar suporte às projeções.

2.4 Orientação a objetos

2.4.1 Origem

A orientação a objetos (OO) surgiu nos anos 60 na Noruega, com Kristen Nygaard e Ole-Johan Dahl, no Centro Norueguês de Computação. Através da linguagem Simula 67, foram introduzidos conceitos de classe e herança. Mais tarde, tendo Alan Curtis Kay como um dos líderes desse projeto, seu conceito foi se aperfeiçoando, pois começaram a programar em Simula, o primeiro editor gráfico orientado a objetos.

Mais tarde formulou-se uma analogia “algébrico-biológica”, defendendo que o computador deve funcionar como organismo vivo, onde cada “célula” se relaciona com outra, visando alcançar um objetivo e funcionando de forma autônoma, podendo agrupar-se para resolver problemas.

Alan Kay observou que o conceito de objetos tinha potencial como uma ferramenta cognitiva, havendo uma correspondência com a forma de pensar das pessoas sobre o mundo. Percebendo ainda que um substantivo isolado gera uma imagem na mente, porém um verbo não tem o mesmo efeito. Assim, Alan pensou em construir um sistema de softwares que interagissem entre si, com os seguintes princípios:

- Qualquer coisa é um objeto;
- Objetos realizam tarefas através da requisição de serviços;
- Cada objeto pertence a uma determinada classe;
- Uma classe agrupa objetos similares;
- Uma classe possui comportamentos associados ao objeto;
- Classes são organizadas em hierarquias;

Portanto Alan criou o paradigma da orientação a objeto, a mais eficiente forma de análise e programação da atualidade.

2.4.2 Conceitos

Gueres (2011, p. 43) relata que “o ser humano, no início de sua infância aprende a pensar de maneira orientada a objetos, representando seu conhecimento por meio de abstrações e classificações”. Sendo assim, as crianças aprendem conceitos simples, como pessoas, carros ou mesas, por exemplo, que são conceituados como grupos de objetos, onde os objetos possuem características e comportamentos de qualquer objeto do grupo em questão.

No desenvolvimento de software tradicional as estruturas de dados e rotinas são desenvolvidas de forma fracamente acopladas. Surgiu então uma abordagem diferente, a orientação a objeto, que propõe uma organização de software em termos de coleção de objetos discretos incorporando estrutura e comportamento próprios (dados e procedimentos).

Na programação orientada a objetos (POO) inicia-se uma abordagem onde o programador visualiza seu programa como uma coleção de objetos cooperantes que se comunicam. Podemos definir orientação a objeto como um modelo para desenvolvimento de software que utiliza várias técnicas para estruturar soluções para problemas computacionais, e esse paradigma é estruturado em objetos, que enfatiza os aspectos de abstração, encapsulamento, polimorfismo e herança. Assim sendo, os conceitos de OO são descritos em objetos, classes, encapsulamento, heranças e polimorfismo.

Um objeto é a unidade fundamental de qualquer sistema OO, é uma entidade do mundo real que tem uma identidade, e pode representar entidades concretas (carro, casa, computador) ou entidades conceituais (estratégia) e cada objeto é distinto do outro, mesmo que contenham as mesmas características. Objetos são instâncias (representa o objeto concretizado a partir de uma classe) de classes em tempo de execução e encapsula estado e comportamento, determinam quais informações possuem e quais podem ser manipuladas, sendo representadas em termos de atributos.

Classe é uma entidade que engloba atributos (variáveis ou campos que indicam possíveis informações armazenadas por um objeto de classe, como nome, idade, gênero) e métodos (funcionalidades da classe - representam operações que podem ser realizadas sobre os dados). Cada classe descreve um

conjunto de objetos individuais. É uma abstração que descreve propriedades importantes para uma aplicação e simplesmente ignora o resto.

Os objetos interagem entre si por meios de mensagens, uma chamada para que seja executado um de seus métodos, podendo alterar o estado interno ou mandar mensagens a outros.

Gueres (2011, p. 44) confirma que “uma classe representa uma categoria, e os objetos são os membros ou exemplos dessa categoria”.

Um conceito bastante usado em POO é o encapsulamento que representa a capacidade de acobertar seus aspectos internos, ou impedir o acesso de outros objetos, ocultando informações, sem a necessidade de saber o funcionamento interno de um método, sendo sua única preocupação passar os atributos. Com isso somente os métodos podem acessar seus próprios dados encapsulados, incentivando a modularidade do programa, levando mudanças no próprio método sem afetar outras classes.

Gueres (2011, p. 48) relata que “a herança é uma das características mais poderosas e importantes da orientação a objetos”. As classes são inseridas em hierarquia de especialização, onde uma classe mais específica herda todas as propriedades da classe mais genérica (um nível acima na hierarquia). GUERES (2011, p. 48) completa ainda que “o conceito de herança trabalha com os conceitos de superclasses e subclasses”. As superclasses são conhecidas também como classe-mãe e as subclasses como classe-filha ou derivadas, estendendo as características herdadas da superclasse.

Um das vantagens do uso de herança é o fato de se declarar uma classe com atributos e métodos e a classe derivada não necessitará redeclarar os mesmos atributos e métodos, pois herdará os mesmos, permitindo assim o reaproveitamento, o que otimizará o tempo de desenvolvimento, facilitando nas manutenções e permitindo também a diminuição de linhas de código.

Outro conceito que está diretamente ligado ao conceito de herança é o polimorfismo. Esse termo qualifica um objeto como capaz de assumir diferentes formas em diferentes classes. “Trabalha com a redeclaração de métodos previamente herdados por uma classe” GUERES (2011, p. 48). Ou seja, os métodos terão a mesma assinatura, no entanto precisam ser reimplementados, ou podem ser usados com mesmo nome da superclasse, logo possuindo dois ou mais métodos com a mesma nomenclatura.

No desenvolvimento de sistema em POO é utilizado a abstração que consiste em concentrar no que um objeto é e faz antes de decidir como será instalado, preservando a tomada de decisões. Ele é geralmente usado junto com polimorfismo e herança.

2.5 UML: Unified Modeling Language

2.5.1 Visão geral da UML

A UML (Unified Modeling Language ou Linguagem de Modelagem Unificada) é uma linguagem de modelagem que pode ser utilizada em todos os domínios da aplicação, para modelar softwares baseados no paradigma de orientação a objetos. Auxilia com a definição das características do sistema, como captura de requisitos, comportamento, estrutura lógica, dinâmica dos processos e outras.

“Surgiu da união de três métodos de modelagem: o método de Booch, o método OMT (*Object Modeling Technique*) de Jacobson, e o método OOSE (*Object-Oriented Software Engineering*) de Raumbaugh. (GUERES 2011, p. 19)”.

No projeto inicial Jim Rumbaugh e Grady Boocho tentaram combinar dois métodos de modelagem orientada a objeto: Booch e OMT (Object Modeling Language). Posteriormente Ivar Jacobson, criador do método Objectory se uniu criando o método Unificado no final de 1995. Logo depois Raumbaugh juntou-se na Rational Software. Através dessa união, em 1996, foi lançada a primeira versão da UML. Em seguida, muitas empresas atuantes na área de desenvolvimento e modelagem influenciaram, participaram e contribuíram com o projeto. Em 1997 a UML foi adotada pela “OMG (*Object Management Group* ou Grupo de Gerenciamento de Objetos), como uma linguagem-padrão de modelagem” GUERES (2011, p. 19).

A UML é uma linguagem (possui sintaxe e semântica) padrão da OMG para visualização, especificação, construção e documentação de software orientado a objetos. A visualização do modelo facilita a comunicação, trazendo ao grupo a mesma ideia do sistema, onde cada símbolo gráfico tem uma semântica bem definida. É uma ferramenta poderosa para a especificação de diferentes aspectos arquiteturais e de uso de um sistema. Na construção, torna-se uma geração automática de código a partir do modelo visual e permite a manutenção da consistência entre as duas visões. A UML concede um padrão para desenhar “plantas” do sistema, incluindo Deliverables, ou seja, documentos com

especificação de requisitos funcionais, planos de testes, que são de grande importância para controlar, medir e refletir sobre um sistema durante o seu desenvolvimento e implantação.

A necessidade para se modelar está ligada ao desenvolvimento de um projeto muito bem elaborado, para fornecer uma estimativa de custos, determinar prazos, avaliar a quantidade de profissionais necessários para o projeto, especificar o material, avaliar o local etc. Grandes projetos necessitam ser modelados com riqueza de detalhes, pois estão continuamente em evolução, aumentando proporcionalmente o tamanho, a complexidade e a abrangência, tendo em vista que constantemente os clientes solicitam modificações ou melhorias, e que o mercado está em contínua mudança, o que acaba obrigando novas estratégias, mudanças em leis ou alíquotas. A UML faz a comunicação entre a estrutura e o comportamento desejado do sistema, visualiza e controla a arquitetura e administra os riscos. Desta forma, os sistemas nunca estão finalizados. Portanto, um SI precisa ter documentação extremamente minuciosa, sendo atualizada com facilidade e rapidez, corrigindo o anterior, e é aí que a UML surge, sendo uma de suas vantagens.

A modelagem da UML está incluso dentro do processo de desenvolvimento de um sistema. Uma das primeiras fases no processo de desenvolvimento é o levantamento de requisitos que trabalham com o domínio do problema e tentam determinar “o que” o software deve fazer, e se é realmente possível desenvolver, buscando compreender as necessidades do usuário, através de reuniões e entrevistas para compreensão do funcionamento do processo para informatizá-lo. Não deve ter limites para a quantidade de reuniões, pois o objetivo é compreender as necessidades e os conceitos vagos e abstratos do usuário, auxiliando-o nas informações que deverão ser produzidas e fornecidas, e também com o desempenho do software. Essa fase contempla e separa requisitos funcionais (funcionalidades do software, o desejo do cliente) dos não funcionais (restrições, condições, consistências, validações impostas sobre os requisitos funcionais).

A próxima fase é chamada de análise dos requisitos, onde as necessidades apresentadas pelo cliente são analisadas, verificando se foram especificadas corretamente e se foram bem compreendidas, sendo determinadas as reais necessidades do SI. Pode apresentar um protótipo com intuito de

mostrar o comportamento do sistema em essência, facilitando a compreensão para o cliente, deixando claro que é apenas um rascunho desse sistema. Em seguida tem-se a maior fase da modelagem, a fase de projeto que trabalha com o domínio da solução, preocupando-se em como o sistema fará o que foi determinado na fase anterior, ou seja, a solução do problema identificado.

Enquanto na análise são identificadas as funcionalidades e restrições do software, no projeto será estabelecido como essas funcionalidades irão se realizar. E por fim, a fase de manutenção, considerada por alguns autores a mais importante, por ser quase inevitável, visto que as empresas são dinâmicas e mudam constantemente. Sendo assim, a modelagem auxilia na manutenção facilitando a compreensão do sistema.

2.5.2 Modelagem com UML

Um modelo é uma representação da realidade, uma ligação entre conceitos teóricos e observações. O desenvolvimento de um modelo envolve o conceito de abstração, onde seu propósito está na captura de uma visão de um sistema físico, determinando o que deve ser incluído no modelo. Visa uma melhor compreensão do sistema que se está desenvolvendo.

A UML é muito mais que a padronização de uma notação. É também o desenvolvimento de novos conceitos não usados normalmente. O bom entendimento da UML não é apenas em aprender a simbologia e o seu significado, mas também significa aprender a modelar orientado a objetos no estado da arte.

Com a modelagem, alcançam-se quatro objetivos (BOOCH, 2006): ajudam a visualizar o sistema como ele é ou como é desejado; permitem especificar a estrutura ou o comportamento de um sistema; proporcionam um guia para a construção do sistema e documentam as decisões tomadas.

Um modelo conceitual da linguagem pressupõe aprender três elementos principais: “os blocos de construção básicos da UML, as regras que determinam como esses blocos poderão ser combinados e alguns mecanismos comuns aplicados na UML” (BOOCH 2006, p. 17).

A literatura abrange três tipos de blocos de construção:

- Itens: “são abstrações identificadas como cidadãos de primeira classe em um modelo” BOOCH (2006, p. 18), subdividindo em quatro itens que constituem os blocos de construção básicos:
 - Itens Estruturais: substantivos aplicados em modelos UML que representam elementos conceituais ou físicos, parte menos mutável;
 - Itens Comportamentais: parte dinâmica onde ocorre a troca de mensagens entre objetos, representa o comportamento do modelo no tempo e no espaço;
 - Itens de Agrupamento: referem às partes organizacionais;
 - Itens Anotacionais: usados para descrever, esclarecer alguma anotação sobre o modelo, são as partes explicativas do modelo.

- Relacionamento: reúne e liga os itens entre si, estabelecendo as seguintes relações:
 - Associações;
 - Generalização;
 - Dependência;
 - Realização;
- Diagramas: representação gráfica de uma coleção de itens e seus relacionamentos.

Como os blocos construtivos não podem ser combinados de forma aleatória, às regras da UML especificam o que deverá ser um modelo bem formado. “Os modelos bem-formados são aqueles autoconsistentes semanticamente e em harmonia com todos os modelos a eles relacionados” (BOOCH 2006, p. 29).

A UML possui regras para:

- Quais nomes podem ser atribuídos aos blocos construtivos (nomes);
- O contexto que determina um significado específico para um nome (escopo);
- Como um bloco pode ser visto e utilizado por outros blocos (visibilidade);
- Como os blocos construtivos se relacionam entre si de forma adequada e consistente (integridade);
- O significado para executar ou simular um modelo dinâmico (execução).

Os mecanismos básicos da UML permitem adaptar ou estender a UML de forma controlada, tornando mais simples pela presença de quatro mecanismos básicos:

- Especificações: está por trás dos diagramas, que fornece uma declaração textual da sintaxe e da semântica do respectivo bloco de construção. É usada para determinar detalhes do sistema. Pode atribuir se a classe é abstrata ou não, e a visibilidade dos seus atributos e operações;
- Adornos: possuem notação gráfica única e direta, proporcionando interpretação visual nos aspectos mais importantes;

- Divisões Comuns: divide dois conceitos utilizados na UML (classes e objetos);
- Mecanismos de Extensão: pretendem estender a linguagem de uma maneira controlada (estereótipos, definição de etiquetas e restrições).

Um diagrama é a apresentação gráfica de um conjunto de elementos, agrupando coleções de itens e relacionando-os. Propõe-se permitir a visualização do sistema por diferentes perspectivas, múltiplas visões do sistema, sendo modelado sob vários aspectos, onde cada diagrama é complementado por outro. Também oferece uma visão parcial dos elementos que compõem o sistema, podendo esse mesmo elemento aparecer em diversos outros diagramas. O uso de diagramas diminui a possibilidade de ocorrência de erros futuros, e permite a descoberta de possíveis falhas.

Para Guedes:

“(...) os diversos diagramas permitem analisar o sistema em diferentes níveis, podendo focar a organização estrutural do sistema, o comportamento de um processo específico, a definição de um determinado algoritmo ou até mesmo as necessidades físicas para a implantação do sistema” (2011, p. 30).

A UML se divide em 13 diagramas, podendo ser diagramas estruturais e diagramas comportamentais, sendo que os últimos sofrem uma subdivisão representada pelo diagrama de interação.

- Diagrama Estrutural
 - Diagrama de Classes;
 - Diagrama de Objetos;
 - Diagrama de Componentes;
 - Diagrama de Estruturas Compostas;
 - Diagrama de Implantação;
 - Diagrama de Pacote.
- Diagrama Comportamental
 - Diagrama de Casos de Uso;
 - Diagrama de Atividades;
 - Diagrama de Máquina de Estados;
 - Diagramas de Interação
 - Diagrama de Sequências,

- Diagrama de Comunicações;
- Diagrama de Temporização.
- Diagrama de Visão Geral da Interação

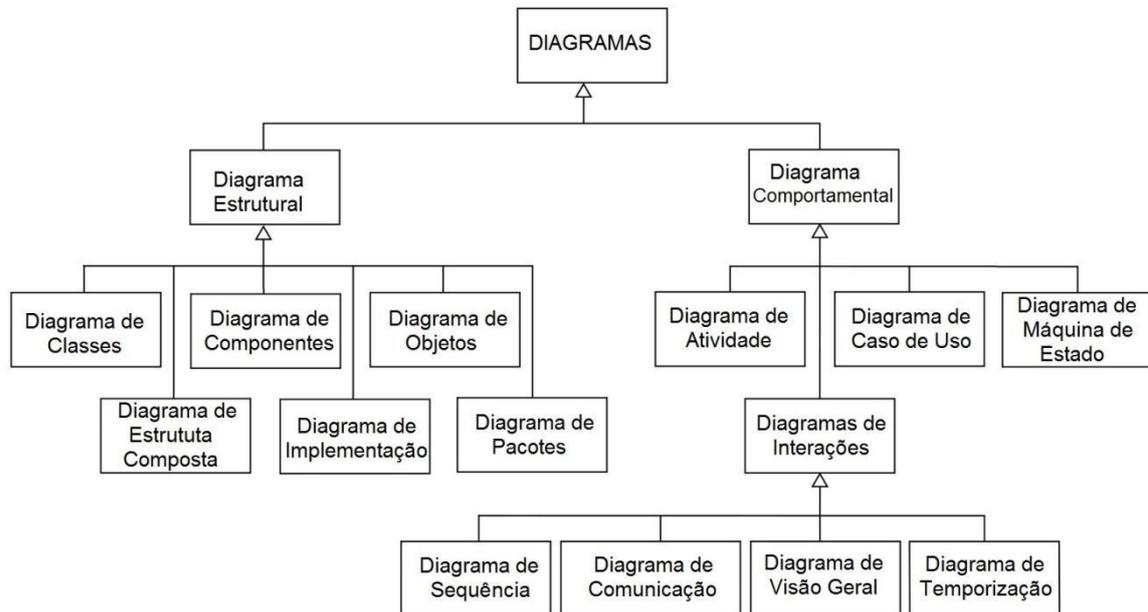


Figura 2 - Diagramas da UML

Diagramas estruturais são utilizados para visualizar, especificar, construir e documentar os aspectos estáticos do sistema, e os diagramas comportamentais são aqueles onde existe alguma alteração no comportamento das classes representadas. Este último se subdivide em diagrama de interação, que representa como os comportamentos de vários objetos interagem uns com os outros a partir de mensagens trocadas entre si em um contexto dentro do sistema, podendo esse contexto representar um caso de uso, modelando assim aspectos dinâmicos.

No tópico seguinte é comentado o conceito de alguns diagramas que foram usados para o estudo de caso desse trabalho.

2.5.3 Diagramas da UML

2.5.3.1 Diagramas de casos de uso

Como já foi observado, o diagrama de casos de uso fazem referência aos blocos de construção utilizando os itens estruturais. Também fazem parte dos diagramas comportamentais, logo por meio de linguagens simples possibilitam sua compreensão do comportamento externo do sistema por qualquer pessoa, sem se preocupar como as funcionalidades serão implementadas. Assim, é considerado o mais abstrato de todos os diagramas, portanto mais flexível e informal.

Costuma ser usado no início da modelagem, nas etapas de levantamento e análise de requisitos, no entanto é utilizado e modificado durante todo o processo.

Seguindo o conceito de UML, esse diagrama auxilia na identificação e compreensão dos requisitos do sistema, ajudando a especificar, visualizar e documentar as características, funções e serviços do sistema desejados pelo usuário, tendo como funcionalidade identificar os tipos de usuários que irão interagir com o sistema, os papéis e as funções específicas de um determinado usuário (GUEDES, 2011).

Os diagramas de casos de uso utilizam dois itens principais: atores e o caso de uso e algumas características descritas abaixo:

- **Atores:** representam os papéis que os usuários poderão utilizar, podendo ser um software, sistema externo ou integrado. São representados por bonecos com breve descrição.

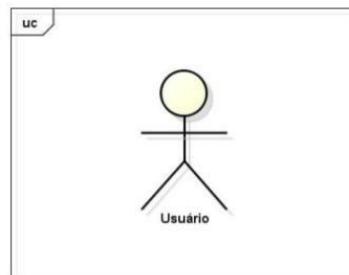


Figura 3 - Representação de um ator

- Caso de Uso: usados para capturar os requisitos do sistema, referindo-se aos serviços, tarefas ou funcionalidades do software e utilizado pelos atores. São representados por eclipses com um texto sucinto que descreve a funcionalidade.

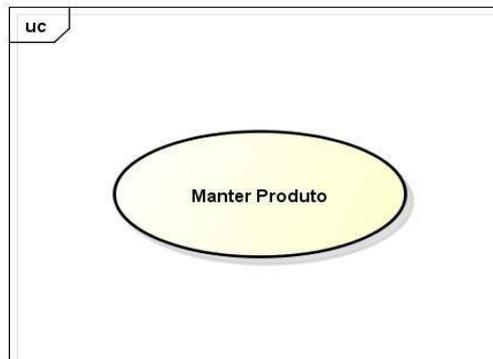


Figura 4 - Representação de um caso de Uso

- Documentação: fornecem instruções em linguagem simples do funcionamento do caso de uso, quais autores, restrições, exceções, validações e parâmetros estão envolvidos e quais etapas devem ser executadas, descrevendo as pré-condições e pós-condições. Possuem um fluxo principal e pode possuir um fluxo secundário Não possuem um padrão definido, logo o torna flexível e de fácil entendimento, logo deve ser evitado o uso de pseudocódigo;
- Associações: são representados pelos relacionamentos entre atores e os casos de uso ou casos de uso entre si. Uma associação representa que um ator utiliza uma funcionalidade, sendo simbolizada por uma linha ligando-os, podendo conter setas indicando o sentido que trafegam as informações. Os relacionamentos entre os casos de uso podem ser inclusão, extensão e generalização/especialização, descritos logo abaixo.

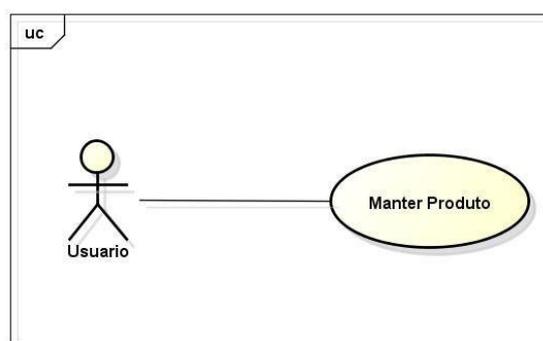


Figura 5 - Representação de uma associação

- **Generalização/Especialização:** Associação entre dois ou mais casos de uso com características semelhantes, diferindo em pequenos pontos. Define-se um caso de uso geral com características compartilhadas e outros casos de uso relacionados com características específicas. Os casos de usos especializados herdam as características dos generalizados. Esse relacionamento também pode ser aplicado sobre autores. Simbolizado com linha com seta na extremidade generalizada.

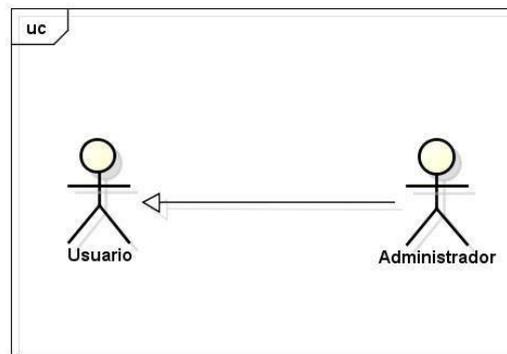


Figura 6 - Representação de herança

- **Inclusão:** usada quando existe um cenário ou rotina comum a mais de um caso de uso, assim outros casos de uso utilizam seu serviço. Esse relacionamento indica obrigatoriedade. É representada por uma linha tracejada com uma seta em uma das extremidades apontando para o caso de uso incluído, e vem acompanhado do estereótipo <<include>>.

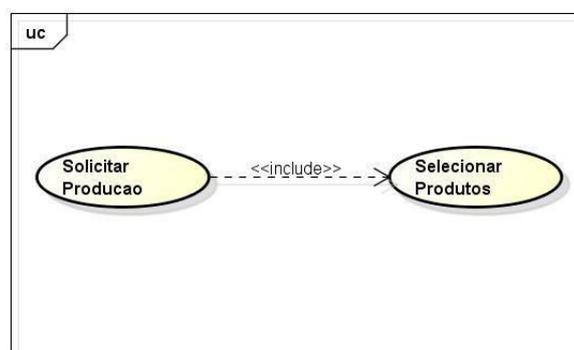


Figura 7 - Representação de uma inclusão

- **Extensão:** utilizados para descrever cenários opcionais. Necessidade de um teste, uma situação específica para executar um caso de uso

estendido ou não. Simbolizado por uma linha tracejada, apontando a seta para o caso de uso estendido, contendo um estereótipo <<extend>>.

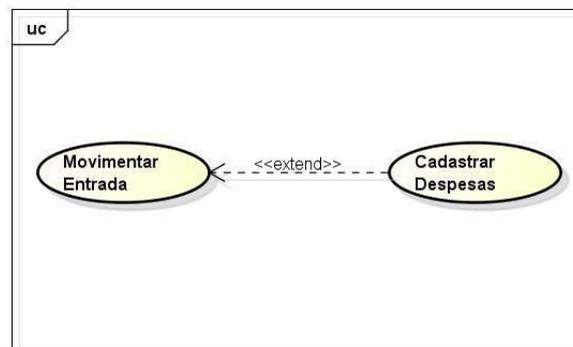


Figura 8 - Representação de um extend

- Restrições: especificado na associação por um texto entre chaves definindo validações, consistências e condições.

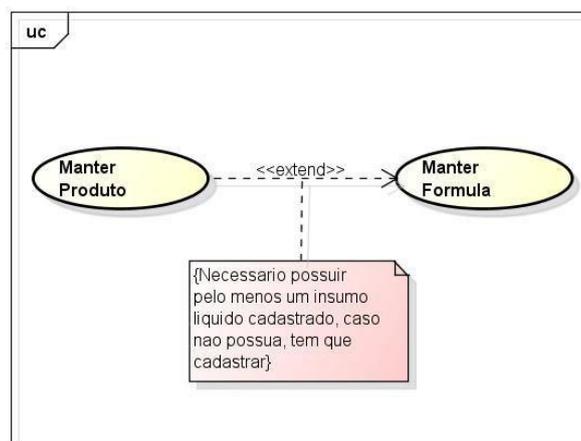


Figura 9 - Representação de uma restrição

- Multiplicidade: colocada nas extremidades da associação entre o ator e caso de uso, especificando o número de vezes que um ator pode utilizar um determinado caso de uso.

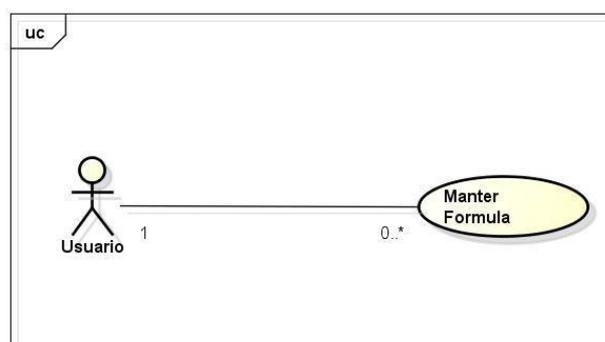


Figura 10 - Representação de multiplicidade

2.5.3.2 Diagrama de classes

O diagrama de classe é um diagrama estrutural que costuma ser considerado por muitos autores um dos mais importantes e mais utilizado da UML.

“Seu principal enfoque esta em permitir a visualização das classes que comporão o sistema com seus respectivos atributos e métodos, bem como em demonstrar como as classes do diagrama se relacionam, complementam e transmitem informações entre si.” (GUEDES (2011, p. 101)

Mostra uma visão estática, definindo a estrutura lógica, servindo como base para a maioria dos demais diagramas. É composto por classes e associações (relacionamentos) entre elas, representado por um retângulo podendo conter três divisões: nome da classe, seguidos pelos atributos e métodos. Abaixo segue um conjunto de características encontradas em diagramas de classes:

- **Atributos e Métodos:** armazenam dados dos objetos da classe e métodos, são as funções que uma instância pode executar. A visibilidade é colocada na frente dos atributos e métodos. São representadas por suspenso (#) que significa protegido, o mais (+) significando o público, e o menos (-) significando ser privada. Os atributos são seguidos pelo seu tipo, e os métodos informam entre parênteses os tipos de parâmetros recebidos e o tipo do retorno.

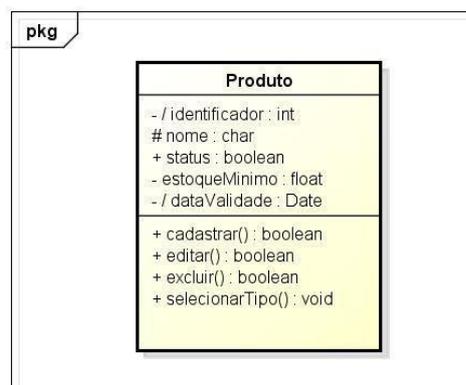


Figura 11 - Representação de uma classe

- **Relacionamento ou Associações:** descreve uma conexão entre os objetos de uma ou mais classes, sendo representadas por linhas que

os ligam. Pode ser colocado um nome na associação para melhor clareza, a navegabilidade (uma seta informando o tráfego da informação) e a multiplicidade informando a quantidade de objetos envolvidos.

Os tipos de associações são:

- Unária ou Reflexiva: essa associação ocorre quando o relacionamento está de um objeto de uma classe para objetos da mesma classe.

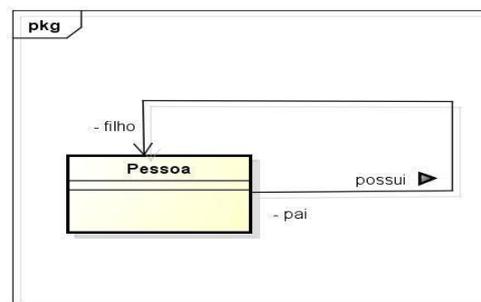


Figura 12 - Representação de associação unária

- Binária: os relacionamentos envolvem objetos de duas classes distintas, sendo a mais comum.

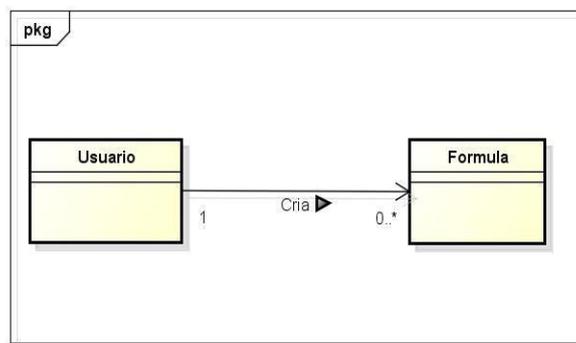


Figura 13 - Representação de associação binária

- Ternária ou N-ária: são associações que conectam objetos de mais de duas classes distintas. Representada por um losango convergindo todas as associações.

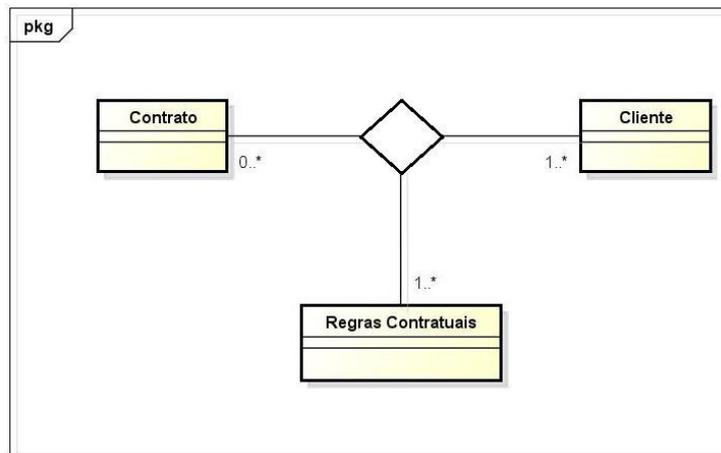


Figura 14 - Representação de uma associação ternária

- Agregação: essa associação tenta demonstrar uma relação de todo-parte, onde o objeto todo é completado pelo objeto parte, sendo simbolizado por um losango aberto na extremidade objeto-todo.

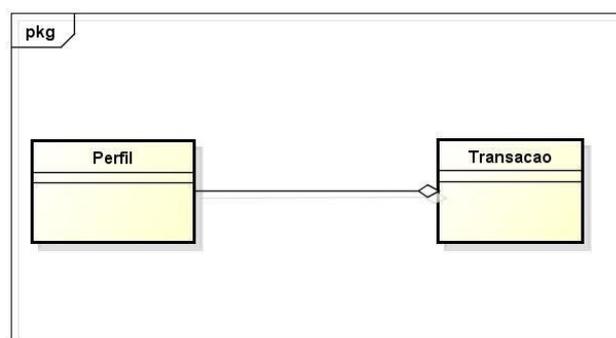


Figura 15 - Representação de agregação

- Composição: uma variação de composição, onde o objeto-parte não existe sem o objeto-todo, sendo simbolizado por um losango fechado na extremidade objeto-todo.

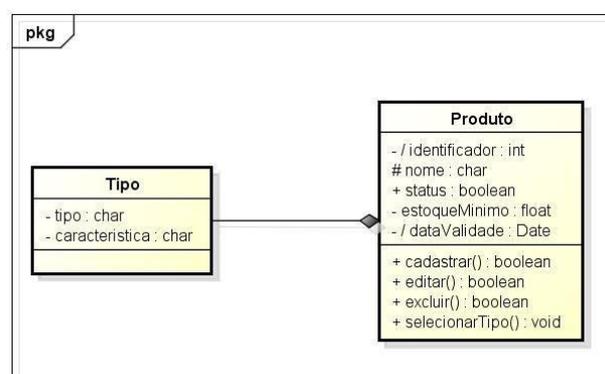


Figura 16 - Representação de composição

- Generalização/Especialização: segue o mesmo conceito utilizado no diagrama de casos de uso, representando herança entre classes-

mãe e classes-filhas, onde as classes-mãe possuem características comuns e as classes-filhas herdam suas características e possuem suas características específicas, evitando redeclarar atributos e métodos herdados.

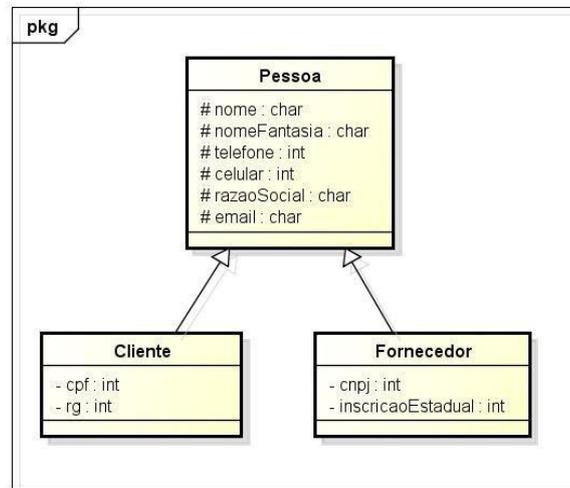


Figura 17 - Representação de Generalização/Especialização

- Classe Associativa: nascem quando ocorre à associação de classes onde ambas as classes possuem multiplicidade muitos (*). “são necessárias quando existem atributos relacionados a associação que não pode ser armazenadas por nenhuma das classes envolvidas” GUEDES (2011, p. 115).

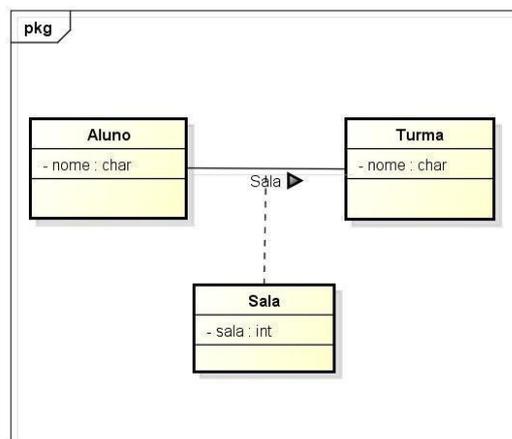


Figura 18 - Representação de classe associativa

- Restrições: podem detalhar ou validar um atributo ou método de uma classe, pode apresentar restrições a determinadas coleções, como ordenação (ordered), permitir ou não a repetição do elemento (bag e

unique), representar uma sequência (sequence ou seq). Uma definição formal de restrição segundo GUEDES:

“Restrições constituem-se em por informações extras que definem condições a serem validadas durante a implementação dos métodos de uma classe, das associações entre as classes ou mesmo de seus atributos. As restrições são representadas por textos limitados por chaves. Restrições podem ser usadas para detalhar requisitos não-funcionais, incluindo regras do negocio” (2011, p. 120).

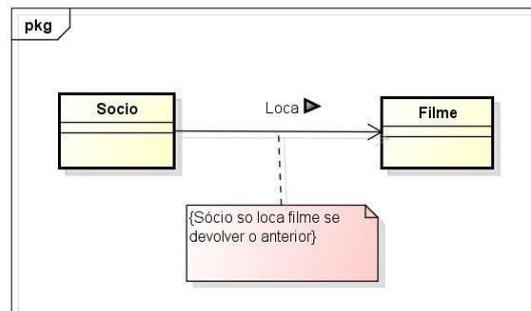


Figura 19 - Representação de restrição

- Estereótipos do Diagrama de Classes: possibilitam a extensibilidade aos componentes, permitindo adquirir características ou funções que os diferenciem, permitindo modelar uma situação diferente. Existem vários estereótipos, mas a linguagem permite mais criações. Em destaques podem ser citados:
 - Estereótipo <<entity>>: explicita que essa classe é uma entidade, ou seja, contém informações recebidas e armazenadas pelo sistema. Prover informações que terão longo período de vida, e que podem ser persistentes (preservadas fisicamente) ou transientes (conservar suas informações somente em memória), se preferir, para melhor clareza pode usar os estereótipos <<transiente>> ou <<persistente>>;
 - Estereótipo <<boundary>>: conhecido como estereótipo de fronteira, servindo de comunicação entre atores externos e o sistema. Torna-se importante quando é necessário definir a existência de uma simples interface para o sistema;
 - Estereótipo <<control>>: identificam classes que servem de intermediário entre classes de fronteira e demais classes, interpretando os eventos ocorridos sobre os objetos de fronteira.

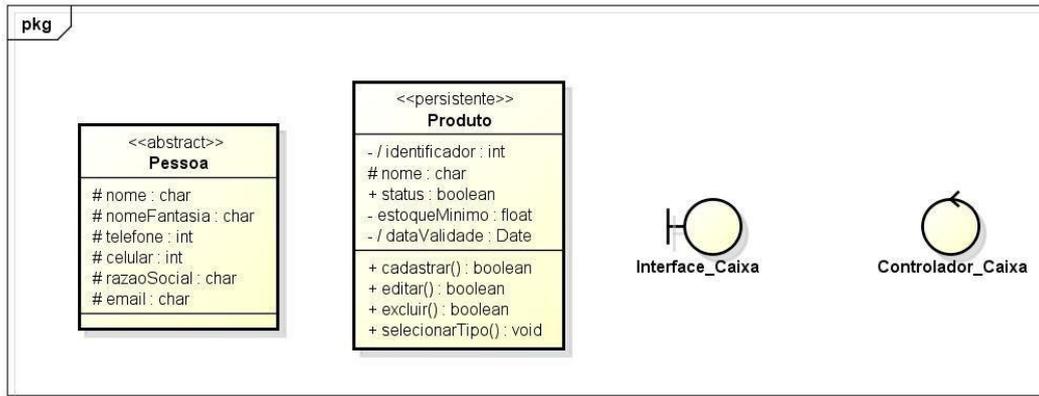


Figura 20 - Representação de estereótipos

2.5.3.3 Diagrama de objetos

Diagrama estrutural, que nada mais é do que uma fotografia do sistema em execução, mostrando os objetos, com os valores de seus atributos e as ligações entre eles.

Definindo formalmente o diagrama de objetos Bezerra esclarece:

“[...] assim como diagramas de classes, diagramas de objetos são estruturas estáticas. Um diagrama de objetos exhibe uma “fotografia” do sistema em certo momento, exibindo as ligações formadas entre os objetos conforme esses interagem e de acordo com os valores dos seus atributos.” (2007, p. 135).

Baseado no diagrama de classe, retratando situações estáticas em que os objetos da classe podem passar, testando seu comportamento, modelam exemplos do domínio do problema. Pode ser utilizado a qualquer momento da especificação.

A esse respeito Pender (2004) relata que: “enquanto o diagrama de Classes modela as definições de recursos, o diagrama de Objetos (Object) modela fatos ou exemplos.” PENDER (2004, pág. 43). Portanto nota-se que há um contraste entre os diagramas de classes e o diagrama de objetos.

A notação é similar ao diagrama de classe, em um retângulo composto pelo nome da classe e o nome do objeto, e os atributos com valores, os métodos não são mostrados. Segue abaixo as características descritas de um diagrama de classe:

- Objeto: um componente similar à classe, mas não apresenta método, somente atributos com seus valores em determinada situação. Podem ser representados de forma mais completa com o nome do objeto, com todas as letras em minúsculo, seguido por dois pontos (:) e o nome da classe com as primeiras letras maiúsculas. Pode ser representado omitindo o nome do objeto, composto pelos dois pontos e o nome da classe. E por fim, pode possuir apenas o nome do objeto.



Figura 21 - Representação Objeto

Os atributos são herdados pelo diagrama de classe, possuindo as mesmas visibilidades e contendo seus valores em uma determinada situação.

- Vínculos: Guedes (2011) retrata vínculos como “[...] instâncias de associações entre as classes representadas no diagrama de classes [...]” (GUEDES, p.184). Possui a mesma representação utilizada no diagrama de classes, porém sem a multiplicidade.



Figura 22 - Representação de Vínculo

2.5.3.4 Diagrama de sequência

Classificado como um diagrama comportamental, tendo como objetivo de mostrar como ocorre a colaboração e as trocas de mensagens entre os objetos ao realizar um cenário de caso de uso e utilizado na identificação dos métodos das classes. Costuma-se usar os mesmos estereótipos do diagrama de classes.

Para se construir um diagrama de sequência seguem alguns passos:

- 1) Escolhe-se um caso de uso;
- 2) Identificar os objetos que fazem parte da interação;
- 3) Identificar o objeto que começa a interação;
- 4) Identificar as mensagens trocadas entre os objetos;
- 5) Identificar a sequência destas mensagens;

Um diagrama de sequência possuem os seguintes elementos:

- Atores: são instâncias dos atores modelados no diagrama de casos de uso e que geram eventos que iniciam processos. São representados por bonecos magros idênticos aos usados no diagrama de casos de uso;
- Lifelines: são retângulos representando um objeto (instância de uma classe) que participa de uma interação, podendo ser criada a qualquer momento ao decorrer da execução. Possui a mesma nomenclatura e regras do diagrama de objetos;
- Linhas de Vida: linha vertical tracejada partindo do objeto (lifeline) representando o seu tempo de vida, sendo interrompida com um “X” quando o objeto é destruído;
- Foco de Controle ou Ativação: barras verticais dentro da linha de vida indicando quando um objeto participou ativamente do processo;
- Mensagens ou Estímulos: demonstram mensagens trocadas entre objetos, contém a assinatura do método e opcionalmente os parâmetros. A mensagem pode acontecer entre atores (não disparando método), entre um ator e um objeto, ou entre objetos. São representadas por linhas cheias ligando os componentes, contendo

setas indicando de onde parte a informação e para onde vai. Contém um texto indicando qual evento ocorreu seguido de dois pontos (:) e o método que foi chamado. Pode acontecer do evento não chamar um método, ou só o método ser descrito. Mensagens podem ser síncronas, ou seja, necessitam de uma mensagem de retorno para dar continuidade no processo, sendo representadas por uma linha e seta cheia, e mensagens assíncronas, ou seja, não necessitam de mensagens de retorno.

- Mensagens de retorno: identifica a resposta de uma mensagem para o objeto ou ator, são representadas por linhas tracejadas com uma seta fina apontando para o objeto que recebe o resultado do método chamado;
- Autochamadas ou Autodelegações: são mensagens enviadas de um objeto para si mesmo;

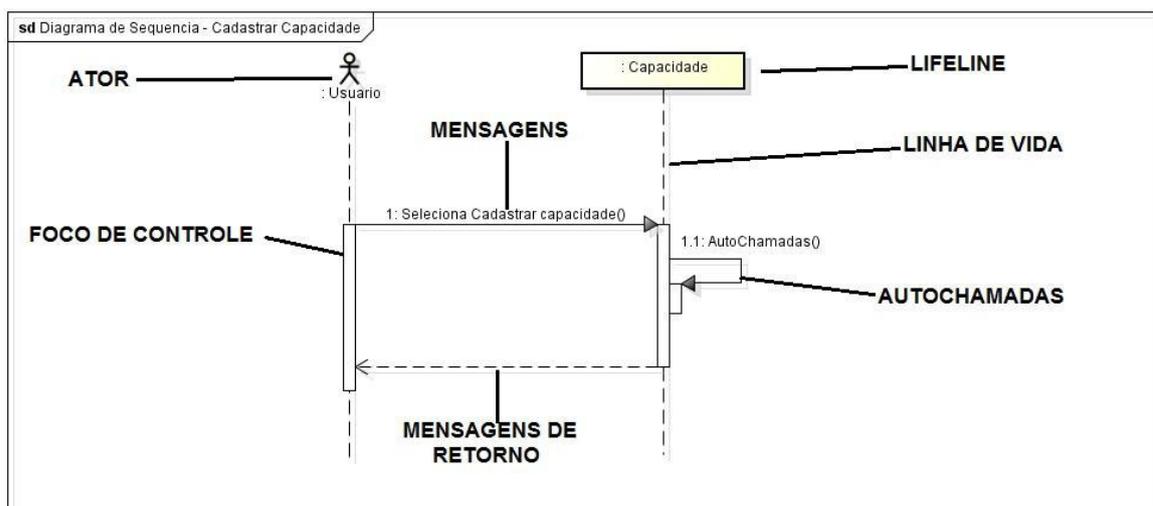


Figura 23 - Elementos do diagrama de sequência

- Uso de Interação: são usados fragmentos de interação como parte de uma interação, no entanto sendo independente. Representado por um retângulo que envolve toda a interação com uma aba no canto superior esquerdo informando a qual diagrama de interação se refere e um texto contendo a descrição da interação que está sendo modelada. Para evitar poluição no diagrama, tem-se a possibilidade de referenciar fragmentos de interação com o operador Ref na aba superior esquerda e ao centro do retângulo se coloca o nome do fragmento o qual é referenciado.

- Fragmentos Combinados e Operadores de Interação: usados para sanar problemas como testes se-senão, laços ou procedimentos paralelos, por meio do uso de fragmentos combinados. Estes são representados por retângulos que determinam a área de abrangência, contendo em sua extremidade superior esquerda o nome do operador de interação. Os operadores mais usados são:
 - ALT (alternatives - alternativa): representando uma escolha entre dois ou mais comportamentos, possuindo uma condição colocada entre colchetes (condição de guarda) para definir qual comportamento será realizado, possuindo uma divisão em linha tracejada dentro do retângulo por cada condição de guarda;
 - OPT (option - opção): representando uma escolha de comportamento, onde pode ou não ser executado;
 - PAR (parallel - paralelo): representando uma execução paralela de dois ou mais comportamento, separado por uma linha tracejada;
 - LOOP (looping - laço): representando um laço, possui uma restrição de interação aplicada ao laço;
 - BREAK: significando quebra, representa a interrupção na execução normal de um processo;
 - CRITICAL REGION: significando região crítica, representando uma interação que não pode ser interrompida por outro processo até ser totalmente concluída.

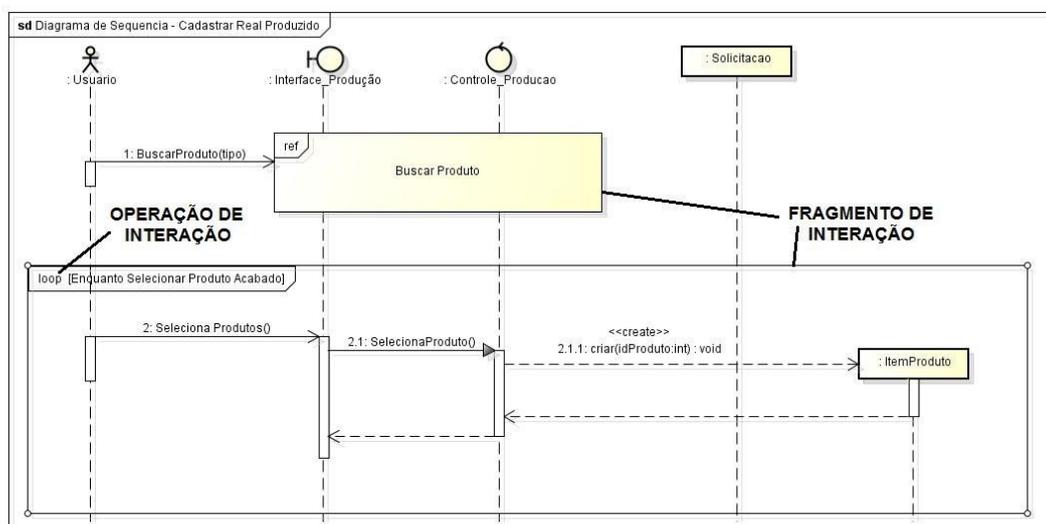


Figura 24 - Fragmento de interação e operador de interação

2.5.3.5 Diagrama de atividades

Sua modelagem enfatiza a sequência e condição para coordenar comportamentos de baixo nível, sendo considerado pela literatura como o diagrama com maior ênfase de algoritmo UML e um dos mais detalhistas (GUEDES, 2011). É o diagrama que mais sofreu mudanças em seu meta-modelo.

Nesse diagrama uma atividade é modelada como uma sequência estruturada de ações, controladas potencialmente por um nó de decisão e sincronismo. É utilizado para modelar um método ou um algoritmo, ou até mesmo um processo, ou seja, uma atividade.

É composto pelos seguintes componentes:

- **Atividade:** coordena a execução de comportamentos subordinados usando um modelo de fluxo de controle de dados. Contém uma sequência de ações que podem ser de vários tipos, como funções aritméticas, início ou fim de comportamento, para comunicação, ou manipulação de objetos e são iniciados por objetos, eventos externos ou outros comportamentos. Uma atividade é representada por um retângulo com bordas arredondadas;
- **Nó de Ação:** são os elementos básicos de uma atividade, representa uma etapa a ser executada numa atividade, não pode ser decomposto. Representado por um pequeno retângulo, similar a atividade, porém em menor proporção;
- **Fluxo de Controle:** conector que liga dois nós, representado por uma linha apontando para o um novo nó, demonstrando a ordem em que as ações são executadas, pode conter uma descrição, condição de guarda ou restrição;
- **Nó Inicial:** utilizado para início de controle de fluxo da atividade, representado por um círculo preenchido;
- **Nó de Final de Atividade:** componente usado para fim do fluxo da atividade, representado com um círculo preenchido dentro de um círculo vazio;

- Nó de Decisão: como o próprio nome diz, é um nó de controle, utilizado na escolha de dois ou mais fluxos, geralmente é acompanhado por uma condição de guarda;
- Nó de Bifurcação/União: é um nó de controle que se divide em dois ou mais nós concorrentes (bifurcação) como pode juntar dois ou mais nós em um só (união), representado por uma barra de onde os fluxos chegam e partem;
- Final de Fluxo: simboliza o encerramento de uma rotina, mas não de toda a atividade, representada por um círculo com um "X";
- Fluxo de Objetos: fluxos que chegam e partem de um nó de objeto, podendo passar dados por ele, modificá-lo, instanciando ou destruindo;
- Nó de Objeto: representa uma instância de uma classe, ou seja, um objeto, representado por um retângulo, seguindo as regras do diagrama de objetos para nomeá-lo. Podem ser considerados como alfinetes (Pins) que são nós de objetos que representam a entrada ou saída para uma ação;

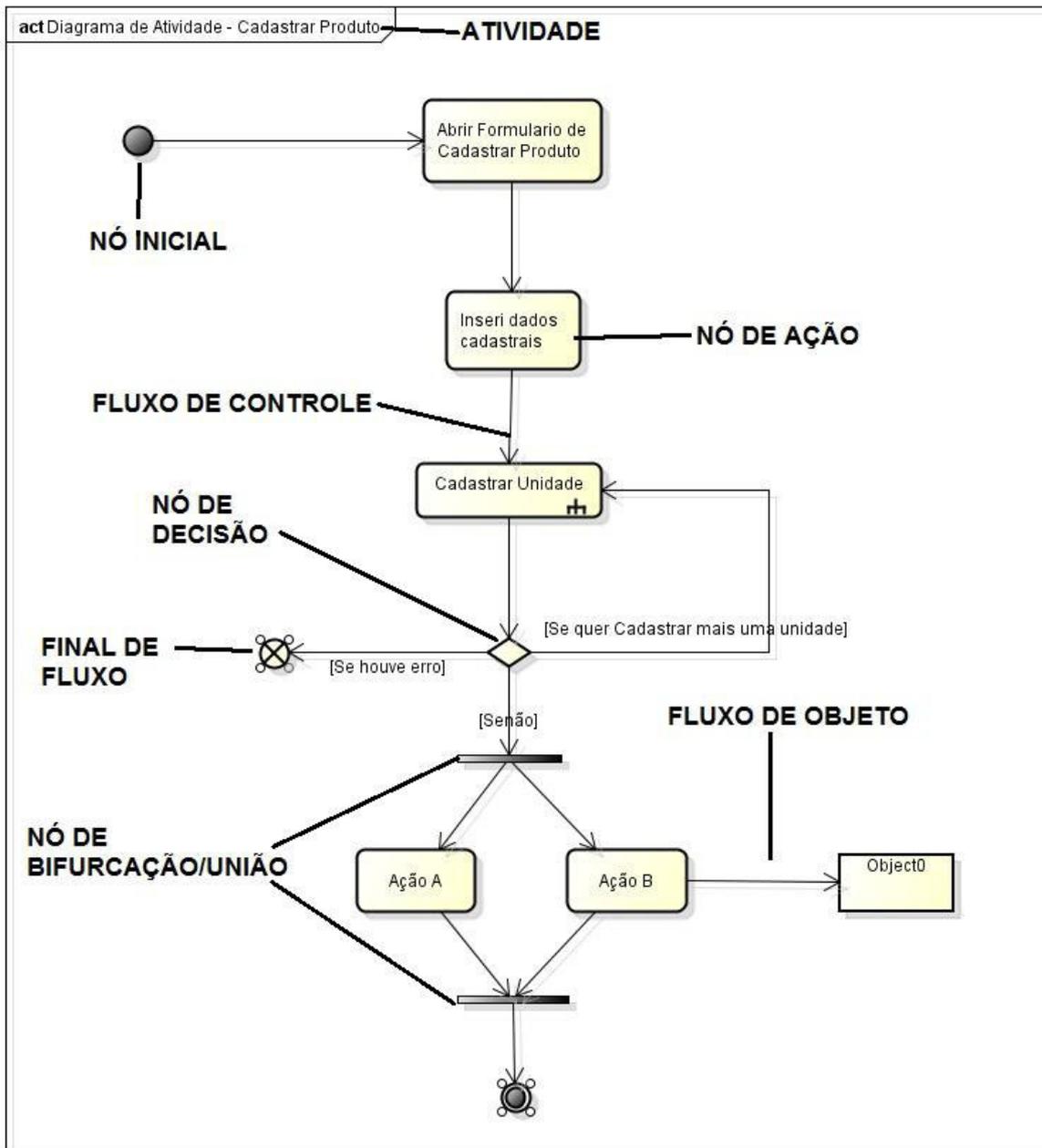


Figura 25 - Componentes do diagrama de atividade

- Nó Parâmetro de Atividade: “Um nó de parâmetro de atividade é um nó de objeto para representar a entrada ou a saída de um fluxo de objetos em uma atividade” (GUEDES, 2011, p. 285);
- Ação de Envio de Sinal: representa um envio de sinal para um objeto ou ação. Representada por um retângulo com uma protuberância no lado direito;
- Ação de Evento de Aceitação: “é uma ação que representa a espera de um evento de acordo com determinadas condições” (GUEDES,

2011, p. 286). Representada por um retângulo com reentrância no lado esquerdo;

- Ação de Evento de Tempo de Aceitação: variação de ação de evento de aceitação que considera o tempo para que o evento seja disparado;
- Nó de Repositório de Dados: usado para armazenar dados ou objetos permanentemente. Os dados que saem são cópias dos dados originais;
- Conectores: atalhos para fluxos;
- Ação de Chamada de Comportamento: ação que invoca um comportamento, podendo este ser uma atividade. Representado com um símbolo de ancinho para baixo em seu canto inferior direito;
- Partição de Atividade: permite a divisão da atividade em setores ou departamentos, ou de um processo manipulado por diversos atores, representa o fluxo entre esses setores. Representados por retângulos na horizontal ou vertical.

2.5.4 UML 2.0

A constante evolução tecnológica causava um desgaste na linguagem UML, pois não tinha como prever determinados comportamentos e ser compatível com tecnologias inovadoras. Esse fato acabou gerando uma necessidade na inovação da linguagem, adaptando-a a uma plataforma ou a um domínio de aplicação específico.

Guedes (2004) afirma que os objetivos que levaram os desenvolvedores da linguagem UML a lançar a versão 2.0, foi reestruturá-la e refiná-la de a fim de torná-la mais fácil de aplicar, implementar e adaptar, melhorando a precisão e capacidade de reutilização. A especificação da linguagem é caracterizada pela utilização de uma abordagem de metamodelagem que se encaixa nas técnicas de especificação formal, fornecendo vantagens intuitivas e pragmáticas.

A infraestrutura visa definir mecanismos que permitam a personificação e à adaptação da UML, criando linguagens para plataformas e domínios específicos, permitindo ser reutilizado em outras arquiteturas de metamodelos.

Segundo Guedes (2011) o desenvolvimento da superestrutura da UML foi motivada por diversos objetivos, tais como:

- Melhorar o suporte ao desenvolvimento com base em componentes;
- Refinar a capacidade de especificação arquitetural;
- Melhorar o suporte para modelagem de processos de negócios;
- Melhorar a escalabilidade e precisão de outras construções comportamentais;
- Aprofundar a precisão de maneira a melhor suportar modelos executáveis.

Com o desenvolvimento da UML passou-se a utilizar com intuito de modelar a estrutura e o comportamento de sistemas, dividindo-se em diversos pacotes que lidam com modelagem estrutural e comportamental, como documentados nos capítulos anteriores pelos diagramas.

2.6 FERRAMENTA ASTAH PROFESSIONAL

Astah é uma ferramenta CASE (Computer-Aided Software Engineering) de responsabilidade da empresa japonesa Change-Vision, que auxilia na atividade em engenharia de software desde análise de requisitos e modelagem até programação e testes. Incorpora características gerais de uma antiga ferramenta descontinuada, conhecida como JUDE (acrônimo de Java and UML Developers Environment – Ambiente para Desenvolvedores UML e Java). Possui versões Community e Professional, sendo a última utilizada no decorrer desse trabalho.

Esta ferramenta é usada para projetar sistemas, possuindo funcionalidades como editor de diagramas UML, que incorpora outros recursos de acordo com a distribuição utilizada, pode ser executado em qualquer sistema operacional rodando sob uma máquina virtual (JVM – Java Virtual Machine), garantindo assim a portabilidade em qualquer plataforma.

O ambiente integrado para desenvolvimento (IDE - Integrated Development Environment), da ferramenta Astah Professional está dividida em quatro visões como mostra na figura 26 :

- Visão de gerenciamento: usado para operações básicas, incluindo funções relacionadas com todo o projeto, e funções usadas com frequência podem ser encontrados na barra de ferramentas;
- Visão do projeto: fornece uma visão geral do projeto pode ser usado para alternar os projetos, visualizar em estrutura de árvores de herança os diagramas, e efetuar buscas ou apelidos usados para os elementos do modelo;
- Visão de propriedades: utilizado para editar as propriedades dos elementos do modelo;
- Editor de diagramas: edita diagramas e modelos. Possibilitando que vários diagramas sejam abertos, representados por meio de abas.

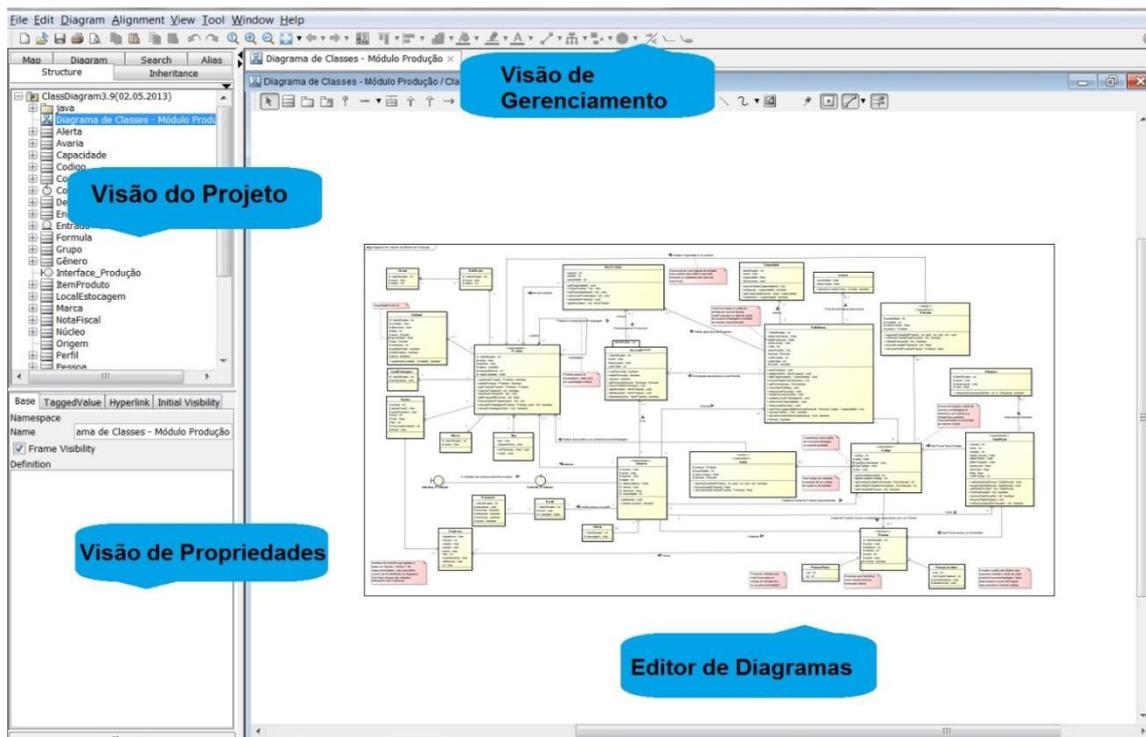


Figura 26 - Ferramenta Astah Professional

A ferramenta oferece suporte a diagramas para UML 2.x: diagrama de Sequência, diagrama de máquina de estado, diagrama de comunicação, diagrama de atividade e diagrama de estrutura composta. Para a UML 1.4, como por exemplo: diagrama de Classes, diagrama de objeto, diagrama de pacotes, diagrama de casos de uso, diagrama de componentes e diagrama de implantação. Oferece também suporte à criação diagramas de processo Erikson-Penker, diagramas de fluxo de dados, fluxogramas, diagramas de entidade e relacionamento, CRUD (acrônimo de Create, Read, Update e Delete em língua Inglesa), para as quatro operações básicas utilizadas em bancos de dados relacionais, diagrama de exigência, tabela de requisitos, diagrama de requisitos, rastreabilidade de mapa e mapa mental, engenharia reversar de bancos de dados.

É importante lembrar que o Astah Professional possui algumas funcionalidades como, simples layout, fácil navegação e alinhamento de modelos, possibilidade de descrição dos casos de uso e criação de um modelo de descrição (Template), a geração do diagrama de classes pode ser feita de forma simples ou detalhada, efetuar impressão em folha simples ou em multiplas páginas, criação de documento em formato HTML e RTF (Rich Text Format), exporta para JPEG, PNG, EMF, SGC, CSV, SQL, exporta e importa XMI (XML –

Extensible Markup Language – Metadata Interchange) permitindo integração com outras ferramentas, exporta relatório de definição de entidades e aceita XML como entrada. Importa Java Source Code e gera esqueleto code especificando a linguagem, podendo assim gerar atributos específicos para cada uma. Suporta linguagens como Java, C# (C-sharp), C++.

Essa ferramenta possui versões para o Windows, Linux e Mac. Fornece a funcionalidade de comparar projetos, efetuar fusões entre projetos, ou usar um projeto para referência somente como leitura, sendo útil para equipe de desenvolvimento para trabalhar em projetos compartilhados.

3. DESCRIÇÃO DO NEGÓCIO

3.1 A empresa

Sabor Laticínios do Nordeste Ltda é uma empresa familiar fundada em 1998, trabalhando inicialmente com uma produção diária de 50 litros de iogurte apenas na capital, situada nos fundos da casa de Antônio Edilson de Melo Filho. Inicialmente o iogurte se chamava Dannyto, mas logo ocorreu uma mudança de nome, pois foi patenteadado pela Danote, passando a se chamar Bonnyto.

A expansão da empresa deu-se em 2002, ampliando seus mercados para o interior do estado do Maranhão a partir de 2003. Atualmente distribui o iogurte Bonnyto para mais de 50 municípios e possui mais de 3000 cliente ativos, além de distribuir para todas as grandes redes de supermercados como Mateus e Maciel.

É uma empresa que atua no setor de laticínio, voltada para produção de iogurte, e se organizando para colocar em foco novas linhas de produção, como light, diet e probiótica, além de novos sabores e distribuição de produtos frios.

A empresa conta com uma frota moderna e constantemente renovada. Contabilizando um total composto por onze veículos, sendo nove próprios e dois terceirizados. Todos os veículos são equipados com isolamento térmico para melhor acondicionamento e controle de temperatura dos produtos. Nas rotas da capital, são usados dois Hyundai Hr, duas Mercedes-Benz Sprinter 413, duas Montanas Combo e um Fiorino Furgão destinados para atendimento do autosserviço e serviços terceirizados. No interior do estado, a frota é composta por um caminhão Mercedes-Benz 13-18, um Volkswagen 8-120 Worker e uma Mercedes-Benz Sprinter 413, todos equipados com isolamento térmico e câmara frigorífica, mantendo assim o produto em ótimo padrão de qualidade, independente de realizarem viagens de longas distâncias. Além disso, a empresa conta com suporte de um Uno Mille para supervisão e suporte nas rotas.

Possui uma produção diária de 5.100 litros geralmente divididos em 25.000 unidades de chupetinha de 110gr, 7.000 unidades de 180gr, 700 unidades de 450gr e 1200 unidades de 850gr nos sabores morango e ameixa. A

chupetinha é considerado o carro chefe da Sabor de Laticínios, sendo fabricadas mais de um milhão por mês.

A Sabor Laticínios acredita que através da modernização continuada de seus processos com foco na melhoria da qualidade de seus produtos, investimento em tecnologia de ponta, capacitação adequada de funcionários, maior integração da empresa academia com a comunidade/empresas, da valorização da pesquisa e da inovação tecnológica, será possível lutar pela sobrevivência da empresa no mercado e contribuir para o fortalecimento do setor com um todo.

No Brasil, a automação na produção de leite, queijo e manteiga é ainda incipiente, fazendo com que as inovações se concentrem em determinados pontos do processo de produção, onde as possíveis falhas humanas têm consequências graves para a qualidade do produto. Por outro lado, o uso de controles automáticos e de processos programados está, atualmente, em quase todas as etapas de produção. Assim, a Sabor Laticínios sentiu necessidade do uso da tecnologia da informação (TI) como uma estratégia de inovação tecnológica e, conseqüentemente, de competitividade em laticínios, modernizando os processos administrativos no âmbito operacional, especificamente nos setores fabril, gerencial e comercial, que mediante otimização subsidiarão a gestão administrativa nos âmbitos estratégicos e táticos, buscando a criação de soluções próprias, personalizadas e adaptadas às necessidades e particularidades da empresa.

3.2 Funcionamento

O funcionamento da empresa está voltado de acordo com a demanda, possuindo atualmente oito tipos diferentes de produtos, com pré-venda para a capital e conta com pronta entrega para os interiores.

Para a capital, existe uma carteira de clientes ativos montada para os vendedores, onde os mesmos devem seguir uma rota preestabelecida diariamente pelo administrativo. Cada carteira possui em média de 80 a 100 clientes. A empresa possui aproximadamente 29 rotas diferentes com carteiras diferentes. Os vendedores são instruídos a seguir 10 regras ao ter contato com o cliente, que por sua vez são chamadas de dez passos do vendedor excelente:

- Ao chegar ao ponto de venda, cumprimentar o cliente com respeito e cordialidade;
- Antes de fazer o pedido, visitar o estoque do cliente detectando suas necessidades dos nossos produtos;
- Realizar rodizio de estoque garantindo que os primeiros produtos que entraram, serão os primeiros a sair;
- Fazer contagem e etiquetagem das trocas a serem realizadas;
- Garantir que os nossos produtos estejam em condições de espaço e posicionamento superiores aos concorrentes nas áreas de vendas;
- Executar o merchandising dos nossos produtos sempre posicionando-os nos melhores espaços que nossos concorrentes;
- Garantir ao consumidor que nossos preços estão competitivos e bem comunicados a eles;
- Formular a proposta e apresentá-la ao cliente contemplando todo o mix adequado ao perfil do mesmo;
- Fechar a venda e sempre informar ao cliente o seu contato, quando será a entrega e a sua próxima visita;
- Despedir-se do cliente com cordialidade.

O pedido é registrado no pocket, assinado e confirmado pelo cliente. O pocket tem função de armazenar os dados para servir de entrada para o sistema e posteriormente para a logística. Cada vendedor recebe bonificação caso consiga uma positivação nas vendas a partir de 70% da carteira.

Ao retorno a empresa, os dados da venda são registrados e gerado baixa em estoque para produção do solicitado, tornando o mercado o termômetro da produção. É gerada a produção da solicitação via papéis, tendo como prazo de 24 horas para término, após o líquido pronto é gerado os produtos industrializados e liberados para a logística. A entrada de dados no sistema só acontece após o término da produção, ou seja, é gerado a solicitação da produção no sistema após a criação de todos os produtos industrializados, deixando assim o sistema atual somente como registro de dados e não como controle da produção.

Existe um controle de produção, onde sempre deve haver em estoque no domingo 180.000 chupetinhas, 18.000 180gr, 6.000 de 450gr, 6.000 de 850gr, todas divididas em ameixa e morango. Para o estoque de quarta feira deverá possuir 50.000 chupetinhas, 10.000 180gr, 3.000 450gr, 3.000 850gr, todas divididas em ameixa e morango. A diferença entre estoque existe devido a saída dos caminhões para as vendas (pronta entrega) nos interiores.

Após término da solicitação, no dia seguinte, acontece à logística, onde saem para entrega do pedido o vendedor, o motorista e um ajudante, levando consigo nota fiscal ou a nota promissória e cópia do pedido. Ambos saem com a finalidade de efetuar a cobrança, arrumar o iogurte bonnyto no freezer de forma com que se destaque dos concorrentes, efetuar o romaneio de carga se necessário (retirar os que venceram e substituir por novos produtos). Ao retorno para a empresa, o vendedor deverá ter em mãos todas as notas e os devidos pagamentos para ser registrado no sistema.

Para realizarem viagens para os interiores, os caminhões são carregados na segunda feira e possuem uma rota de cidades estabelecidas trabalhando com pronta entrega. O abastecimento é feito de acordo com o histórico de clientes para os interiores pedidos e um aditivo extra de produto caso seja necessário. Sendo necessária a prestação de contas e notas ao retornar para a empresa.

3.3 Estrutura organizacional

Como dito anteriormente a empresa é familiar, possui uma estrutura organizacional bem definida sendo composta por três divisões, o conselho, a diretoria executiva e as gerências operacionais.

O conselho possui o maior nível hierárquico, formado por cinco pessoas, dentre elas os donos e alguns sócios. Desempenham papéis de decisões estratégicas, tomada de decisões mediante as metas e evolução da empresa e avaliação dos resultados obtidos.

A diretoria executiva é composta por única pessoa, possui a missão de assegurar a obtenção dos resultados definidos nos planos operacionais, administrativos e estratégicos do conselho de forma concisa e cumprindo os princípios e filosofia de negócio da empresa, por meio de coordenação geral de todas as áreas. Tende a ser o primeiro a entrar e último a sair, visando à elaboração e implementação dos planos estratégicos e operacionais, assegurando o desenvolvimento, crescimento e continuidade. Define a política e objetivos específicos para cada área, coordenando os planos de ação, visando facilitar e integrar o trabalho das equipes, otimizando resultados e esforço para obtenção das metas. Possui uma visão do mercado fora da empresa, almejando identificar novas oportunidades de investimento ou desenvolvimento do negócio.

A gerência operacional se subdivide em quatro áreas (administrativa, venda, logística e produção), cada uma responsável por monitorar, coordenar e acompanhar os processos por meio de coordenadores técnicos. Cada área visa articular, integrar e agrupar a equipe, a fim de se obter um controle dos procedimentos, regulamentar riscos, assegurar qualidade e criar uma atmosfera positiva e eficiente entre direção, funcionários e coordenadores de áreas, tudo em função do planejamento global da empresa. As gerências operacionais são:

- Administrativa: constituída por um supervisor, quatro estagiários e um auditor de conferência. O supervisor desempenha papel de treinar, conferir e demandar rotinas administrativas do corpo administrativo e escritório, se responsabiliza pelos pagamentos e faturamento contábil (entrada e saída de capital), fechamento do caixa e resolução de problemas corpo-administrativas. Possui contato com os gerentes das outras áreas para maior controle contábil da empresa e com seus

estagiários fiscalizando entrada e saída de dados. Os estagiários (treinados pelo supervisor) são responsáveis pela entrada e saída de dados no sistema, bem como guardar e verificar documentos, registrar contas a serem pagas, lançamento de toda a contabilidade feita pelo supervisor ao sistema (posteriormente validado pelo supervisor via sistema), entrada de notas fiscais, registro e controle de todas as notas fiscais para gerenciamento posterior do supervisor, conciliação do caixa ao sistema, registrar e controlar as conciliações bancárias feitas e gerenciadas pelo supervisor, gerar via sistema o fechamento do caixa e faturamento, diariamente, e por fim positivação da produção, gerindo junto ao sistema toda a produção, venda e logística realizada. A auditora desempenha uma atividade independente e objetiva de garantia e consultoria, de forma sistemática e disciplinada para avaliar, fiscalizar e melhorar a efetividade dos processos, verificando existência, suficiência e a aplicação dos controles internos, bem como contribuir para seu aperfeiçoamento, verificar se as normas internas estão sendo seguidas, observar a necessidade de melhoramento das normas internas vigentes, avaliar a necessidade de novas normas internas;

- Venda: composta por um supervisor de vendas e mais nove vendedores. O supervisor é responsável por determinar diariamente as rotas a serem seguidas pelos vendedores, gerenciar os mesmos, gerar dados referentes para positivação de vendedores, rotas, estratégias de custo benefício e entrega dos desses dados para os estagiários registrarem no sistema. Essa área possui três vendedores na capital e sete vendedores motoristas nos interiores e quatro promotores de venda, sendo uma exigência das grandes redes supermercadistas para gerenciar e garantir a qualidade dos produtos. Os vendedores da capital acompanham os motoristas na logística para levar documentos e receber dinheiro, trazendo consigo os documentos para serem entregues ao supervisor, onde os dados serão cadastrados no sistema pelos estagiários. Em se tratando dos vendedores do interior fazem a venda pronta entrega;

- Logística: composta por um supervisor de logística e mais três motoristas, cada um com um auxiliar para descarregamento da carga. O supervisor de logística possui papel de gerir movimentação dos carros, gerenciar as melhores rotas, calculando os melhores índices, tempo médio de entrega e faturamento envolvido em cada entrega, analisando o melhor custo benefício em cada rota traçada. Mantém contato direto com o estoquista e repassa para todos os dados. Os motoristas são responsáveis para receber a rota e entregar toda mercadoria. Por fim os auxiliares colaboram para descarregar as mercadorias;
- Produção: composta por um supervisor de produção, um chefe de produção, um estoquista e 11 auxiliares de produção. O supervisor possui papel estratégico de criação da solicitação, bem como criação de novos sabores e novas fórmulas, gerencia a cadeia de suprimentos e o balanço de estoque de insumos e embalagens, controla a qualidade dos produtos, e mantém contato direto com o estoquista para o controle do estoque de insumos, embalagens e produtos industrializados. O estoquista desempenha papel de passar para os entregadores as rotas, registrar o horário de entrada e saída de cada motorista, registrar a quilometragem inicial e final de cada carro a cada entrega e conferir a entrada e saída dos produtos industrializados para entrega e dos insumos e embalagens para a produção para posteriormente entregar os dados para o supervisor de logística e de produção, além de alimentar todo o estoque. O chefe de produção recebe ordens do supervisor e comanda todos os operadores na produção.

Vale ressaltar que os estagiários não possuem papel de decisão, apenas controlam dados e registros fisicamente e via sistema. A figura abaixo representa o organograma da empresa:

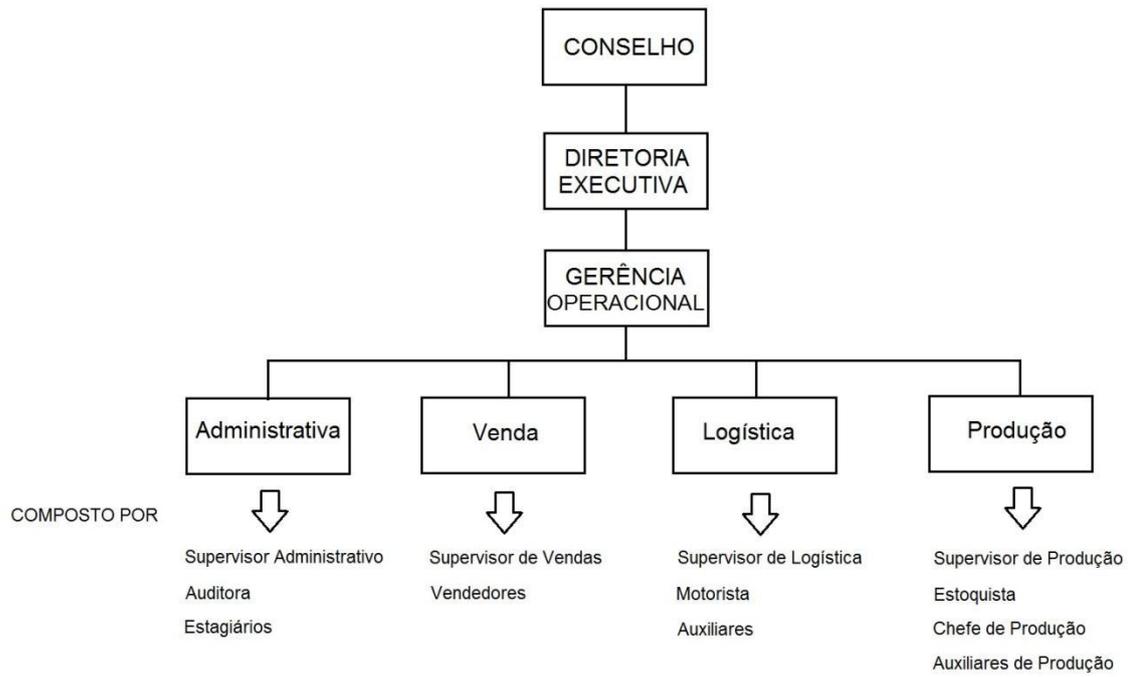


Figura 27 - Organograma

3.4 O processo de produção

O atual processo de produção da Sabor Laticínios é trabalhado com D+1, ou seja, um dia a frente. O administrativo faz a análise dos pedidos feitos pelos clientes da capital, e repassa para a gerência de produção. Vale lembrar que o sistema utilizado é o desktop, realizado em linguagem obsoleta.

A gerência de produção analisa o que deve ser produzido e a quantidade em estoque de insumos e de produtos industrializados, certificando se que o pedido pode ser atendido. Caso o estoque suporte a produção, é tomada a decisão de produzir. Repassa-se para o chefe de produção o que deve ser efetivamente produzido, tendo como função iniciar o processo que demora em média 24 horas para o término da mesma.

Ao término do iogurte gerado, é colocado o líquido em seus respectivos recipientes (referente ao pedido) e logo depois de colocar o rótulo, (em caso de avaria de algum recipiente, é registrado para ser devolvido aos fornecedores) e por fim é realizada a embalagem, passando a ser estocada pronta para ser entregue pela logística.

Depois de todo o pedido já efetuado e realizado, todos os documentos anotados referentes ao mesmo, são passados para o estagiário registrar no sistema. Lembrando que a solicitação é registrada com início no dia anterior e fechamento no dia atual.

A criação da solicitação se dá com o cadastrado do que se deseja produzir (os produtos industrializados com suas respectivas quantidades e fórmula desejada, a Sabor Laticínios atualmente trabalha com duas fórmulas, ameixa e morango) e o sistema calcula a quantidade necessária de insumos para produção. Vale lembrar que o sistema atual considera como insumo todos os insumos básicos para a geração da fórmula como as embalagens e rótulos usados para a criação dos produtos finais. Ao final do cadastro da solicitação, o usuário deve confirmar o que realmente foi produzido e o sistema gera saídas no estoque referentes aos insumos utilizados e entrada no estoque referente aos produtos industrializados. Exemplificando, é feito uma solicitação com 400 chupetinhas, onde o sistema calcula a quantidade de insumos necessários para a produção e depois o estagiário registra para a mesma solicitação que o real produzido foi 380 chupetinhas.

Vale lembrar que para a geração da solicitação, os produtos, o estoque, as fórmulas, os fornecedores e os clientes, devem ser cadastrados previamente no sistema. O estoque é controlado mediante a verificação das entradas com as saídas. Só pode haver entrada de insumos a partir do cadastro dos itens inclusos na nota fiscal.

A tabela abaixo se referente a produção referente ao mês de maio, onde possui 31 dias, porém produção de apenas 19 dias. Podemos visualizar os tipos de produtos (iogurte 110g, 180g, 450g e 850g) com o previsto do mês em volume e o previsto do dia. O previsto do dia é apenas uma previsão acumulativa, ou seja, uma previsão que tende a aumentar a cada dia até o útil dia do mês e igualar ao previsto do mês. Possui também o realizado em volume e o percentual do realizado no mês e no dia (também acumulativo).

| Produtos | Previsto do Mês | Previsto do Dia (Acumulativo) | Realizado | %Mês | %Dia |
|----------|-----------------|-------------------------------|------------|------|------|
| 110G | 106.400,00 | 106.400,00 | 65.255,79 | 61% | 61% |
| 180G | 17.640,00 | 17.640,00 | 12.044,34 | 68% | 68% |
| 450G | 9.090,00 | 9.090,00 | 504,00 | 6% | 6% |
| 850G | 36.380,00 | 36.380,00 | 23.348,65 | 64% | 64% |
| Total | 169.510,00 | 169.510,00 | 101.152,78 | 60% | 60% |

Tabela 1 - Tabela de produção de maio

4. MODELAGEM DO SISTEMA DE PRODUÇÃO

4.1 Sistema de controle de produção (apresentação)

O estudo de caso foi realizado na Empresa Sabor Laticínios do Nordeste Ltda utilizando o auxílio de uma ferramenta para modelagem chamada Astah Professional versão 6.7.0.

A empresa se encontra num contexto dependente de um software desatualizado que pseudo-gerencia e pseudo-controla todo o funcionamento de acordo com seus limites, regras e processos antigos. Foi verificada a necessidade de uma modelagem de todo o funcionamento da empresa para a produção de um software atual que incorpore as novas mudanças e se torne flexível para mudanças futuras, gerando um software de gestão, moderno e de qualidade.

Dentre todos os processos diagnosticados na empresa, foi analisado e modelado o sistema de controle de produção, que representa um dos módulos do sistema como um todo, sendo apresentado de forma individual e autossuficiente visando atender as necessidades da indústria de laticínios.

O sistema deverá ser responsável pela parte do cadastro dos produtos, abertura e fechamento das solicitações, para ter registro do que possui no estoque, servindo de auxílio para o estoquista, que deve saber exatamente onde cada produto foi estocado, para efetuar a retirada para criação da fórmula. Foram encontradas as seguintes funcionalidades para o sistema:

- Devem existir três níveis de usuários: operador administrativo, chefe do almoxarifado e administrador, nessa ordem de hierarquia;
- Apenas usuário administrador pode cadastrar, editar ou excluir (não será excluído do banco de dados, apenas marcado como inativo) outro usuário;
- Para acessar o sistema, todos os usuários devem efetuar o login;
- Todos os usuários podem manter seus próprios dados;

- O sistema pode emitir alertas para os usuários se assim o desejarem (a Sabor coloca frases encorajadoras e motivadoras ao longo da semana);
- Todos os usuários podem manter (cadastrar, editar e/ou excluir) marca, cliente, local de estocagem, núcleo, fornecedor, grupo, subgrupo e endereço.
- Todos os usuários podem manter a capacidade (iogurteira);
- Um grupo pode possuir mais de um subgrupo;
- Cliente e fornecedor devem possuir endereço;
- Todos os usuários podem manter a nota fiscal (o cabeçalho da nota fiscal);
- Nota fiscal pode possuir uma despesa extra, que poderá ser registrada se necessário, por qualquer usuário;
- Os produtos devem ser caracterizados como insumo, embalagem ou produto acabado (industrializados);
- O produto deve possuir um atributo contendo a data de validade, que contém a data de validade do produto que foi dado entrada e está mais perto de vencer, para que seja utilizado esse produto para a próxima solicitação, evitando avaria;
- Todo produto deve possuir no mínimo uma unidade, que pode ser cadastrada por qualquer usuário;
- Produto deve possuir uma marca, um núcleo, um local de estocagem, um grupo e um subgrupo;
- Se o produto for do tipo produto acabado, deverá ser associado um ou mais produtos do tipo embalagem;
- Todos os usuários podem registrar a movimentação de entrada de insumos e embalagens, sendo que as entradas representam os itens registrados na nota fiscal;
- Para o conjunto de itens na nota fiscal (representam as entradas), é gerado um código de entrada, a data de entrada e o tipo da entrada (insumo/embalagem ou produto finalizado);
- Toda entrada de insumo e embalagem deve possuir uma nota fiscal;

- Toda entrada de produto do tipo acabado/industrializado só poderá ser registrado no sistema pelo próprio sistema, mediante abertura de uma solicitação;
- Todo usuário pode cadastrar uma quantidade de estoque mínimo para cada produto cadastrado no sistema;
- Todos os usuários podem visualizar o estoque, buscando por tipo de produto;
- Todos os usuários podem registrar a saída de produto, pode ser para a logística ou como avaria (roubo, perda de produtos, e outros casos);
- Toda saída de produto é associada a um cliente;
- Para cada grupo de saídas de produtos, é gerado um código de saída, contendo a data, o tipo de saída (insumo/embalagem ou produto finalizado) e uma observação;
- A saída pode ser decorrente da solicitação ou gerada saída manualmente por qualquer usuário, referente à saída da produção, roubo, avaria, etc;
- O usuário chefe almoxarifado deve manter a fórmula. Para o cadastro da fórmula, deve possuir produtos cadastrados. A fórmula por padrão é cadastrada em medida de 1000 (mil), ou seja, a quantidade de insumos registrados para a fórmula representa exatamente para criação de mil litros;
- A fórmula é composta por vários produtos do tipo insumo;
- Somente o chefe do almoxarifado e o administrador podem abrir uma solicitação;
- Ao abrir uma solicitação deve ser selecionada a capacidade e a fórmula, que calculará a proporção de insumos utilizados na fórmula e validará a quantidade de insumos no estoque. Se existir a quantidade de insumos no estoque, gera um código de saída e registra a saída de insumos no sistema. E confirma a abertura da solicitação;
- O chefe do almoxarifado pode finalizar as solicitações que ainda estiverem em aberto. Ao finalizar, o usuário cadastra o real produzido, ou seja, já foi produzido pela fábrica a quantidade de líquido registrada na solicitação e já foi feito todos os produtos industrializados

desejados. Logo serão registrados no sistema todos os produtos já produzidos, e se houver alguma avaria do líquido é registrado. Esse requisito foi capturado dessa forma, para se adequar a todas possíveis situações que ocorrem na empresa, evitando assim que o software não se encaixe nas regras;

- Após o usuário inserir todos os produtos realmente produzidos, será calculada a quantidade de embalagens usadas para cada produto cadastrado e registrado, a movimentação de saída de embalagem, associando o mesmo código de saída gerado para a saída de insumo. Sendo que toda saída de embalagem é associada a um cliente;
- A geração manual de saída de produtos acabados pode servir de entradas para outro sistema, ou posteriormente para outros módulos do sistema como um todo, podendo gerar um arquivo em XML.

4.2 Diagrama de caso de uso

O diagrama de caso de uso mostra uma visão centrada na generalização/especialização e todas as funcionalidades do sistema de produção, mostrando os principais atores envolvidos e seus papéis desempenhados na rotina do usuário com o sistema.

No apêndice A mostra o diagrama de casos de uso feito mediante a análise de requisitos capturada com o cliente e relatado no tópico anterior.

Foi selecionado alguns casos de uso, considerados primordiais na visão da equipe de desenvolvimento para ser mostrado nesse trabalho de forma descritiva, mostrando os atores interagindo com a funcionalidade representada por cada caso de uso. Os casos de uso selecionados para ser descritos foram: Selecionar Insumos Líquido (Apêndice B), Selecionar Fórmula (Apêndice C), Selecionar Capacidade (Apêndice D), Registrar Avaria do Líquido (Apêndice E), Recuperar Código de Saída (Apêndice F), Movimentar Saída Produto (Apêndice G), Movimentar Saída Insumo/Embalagem (Apêndice H), Movimentar Entradas (Apêndice I), Manter Unidade (Apêndice J), Manter Produto (Apêndice K), Manter Fórmula (Apêndice L), Manter Estoque Mínimo (Apêndice M), Gerar Código de Saída (Apêndice N), Finalizar Solicitação (Apêndice O), Definir Tipo (Apêndice P), Calcular Embalagem (Apêndice Q), Cadastrar Real Produzido (Apêndice R), Associar Embalagem (Apêndice S), Abrir Solicitação Produção (Apêndice T), Validar Estoque Insumo (Apêndice U). Vale ressaltar que a descrição dos casos de uso foi gerado no Astah Professional e exportados como imagem.

4.3 Diagrama de classes

Para a modelagem do sistema o próximo passo depois de criar os casos de uso é uma análise dos objetos que permitem achar o proprietário de cada uma das tarefas documentadas (MATOS, 2002). Através dessa análise surge o diagrama de classes.

Classificado como diagrama estrutural, o diagrama de classe trata-se de uma estrutura lógica estática. Estão sendo representadas as classes de entidade, negócio e de fronteiras, algumas com seus estereótipos explícitos para auxiliar a equipe de desenvolvimento, através de estereótipos persistentes, uma vez que cada movimentação deve ser registrada e guardada.

Nesse diagrama é mostrado o relacionamento entre as classes, logo pode-se perceber que o produto pode possuir um e somente um núcleo, um local de estocagem, uma marca e possui um único tipo que se caracteriza em ser tipo insumo, embalagem ou produto industrializado. Também possui um subgrupo, este por sua vez está contido em um grupo. Um grupo pode conter mais de um subgrupo, mas um subgrupo só está contido em um único grupo. O produto pode possuir várias unidades. E se o produto for do tipo industrializado deve conter uma ou mais embalagens.

A classe núcleo e pessoa devem possuir um ou mais endereços e a classe pessoa é uma abstração das classes PessoaFisica e PessoaJuridica, sendo representado pelo estereótipo abstract notando-se a herança entre as classes.

Um usuário contém somente um perfil, que por sua vez possui todas as transações para permitir ou negar o acesso do mesmo a determinadas funcionalidades. A classe Usuario também possui uma classe Alerta podendo possuir zero ou muitas instâncias da mesma. A classe Usuario pode criar uma ou várias fórmulas, abrir uma ou varias solicitações, cadastrar um ou vários produtos, assim como notas fiscais e gerar várias entradas ou saídas de produtos (insumo, embalagem ou produto industrializado).

A classe Formula agrega vários itens de produto e o mesmo está ligado a somente um produto. Cada fórmula possui um identificador único explicitado no diagrama através do estereótipo unique, sendo este de valor inteiro.

A classe solicitações se relaciona com as classes capacidades, avaria e código (com cardinalidade 2, gerando para cada solicitação exatamente duas instâncias da classe código, uma para guardar o código de entrada e outra para guardar o código de saída, sendo que código possui uma associação com pessoa representando qual o cliente a saída ou a entrada estão associados), gerando movimentações de entrada e saída de produtos, sendo simbolizado através do relacionamento entre as classes produto, saída e entrada. As classes entrada e saída foram destrinchadas em duas para melhor entendimento da equipe de desenvolvimento da existência de duas movimentações diferentes, utilizando o estereótipo persistente para mostrar que são classes em que seus dados são persistidos no banco de dados em tabela diferentes. Foi realizado dessa forma para melhor migração dos dados do banco anterior para o atual. Vale ressaltar que são classes que contém regras de negócio, sendo colocado nas classes mais importantes o estereótipo entity, noutras classes não foi colocado para evitar sujar o diagrama.

Por fim a classe nota fiscal possui uma associação com pessoa que representa o fornecedor da qual veio, possuindo também uma associação com código para registrar as entradas dos insumos e embalagens. E caso seja necessário existir alguma despesa extra registrada na nota fiscal é associado a classe despesa.

Primeiramente é mostrado o modelo conceitual da modelagem aplicada ao sistema de produção, gerado na fase de análise em que é identificado o domínio do problema, representando as classes no domínio do negócio (Apêndice V).

Depois da criação do modelo conceitual, com auxílio dos diagramas de sequências, foi desenvolvido o modelo de projeto que possui a visão dos objetos para a solução do problema, sendo acrescentados detalhes ao modelo anterior e classes de controle e de fronteira, no entanto para evitar poluição no diagrama a classe de controle foi ligada apenas com uma classe (Apêndice W).

4.4 Diagrama de sequência

O diagrama de sequência tem como finalidade mostrar as principais interações que ocorrem entre os atores e o sistema especificado (LARMAN, 2000).

Esse diagrama apresenta grande simplicidade, por eliminar o excesso de objetos envolvidos no sistema em cada sequência mostrada, e por mostrar o foco de cada objeto nos comandos e mensagens enviadas ao decorrer de sua vida útil. Permitindo visualizar as necessidades que devem ser atendidas na especificação estática das interfaces do sistema com o usuário, partindo de aspectos funcionais.

Caracteriza-se por ser útil na comunicação entre analistas, e estes com seus usuários. As figuras a seguir ilustram as interações, tendo como regra geral um diagrama baseado em um caso de uso.

Para compor esse trabalho, foi escolhido alguns casos de uso e feito o diagrama de sequência do mesmo. O diagrama Fazer Login (Apêndice X) mostra a sequência de mensagens trocadas para que o usuário faça o login no sistema e tenha acesso ao mesmo.

O diagrama de capacidade (Apêndice Y) mostra a sequência de mensagens trocadas para que o usuário cadastre uma capacidade (iogurteira) no sistema. No diagrama de editar capacidade (Apêndice Z) mostra a sequência para editar algum possível dado.

O diagrama de cadastrar produto (Apêndice AA) mostra a sequência de mensagens trocadas, colocando uma condição para que os dados cadastrados sejam validados e posteriormente registrar o tipo do produto, e caso o produto for do tipo embalagem é chamado o diagrama de sequência associar embalagem (Apêndice BB). Após o cadastro de dados básicos a respeito do produto, o usuário pode cadastrar as unidades e posteriormente pode opcionalmente cadastrar o estoque mínimo para esse produto (Apêndice CC). Foi colocado ainda nesse trabalho o diagrama de sequência buscar produto (Apêndice DD), editar produto (Apêndice EE) e excluir produto (Apêndice FF).

Para o cadastro da nota fiscal (Apêndice GG) é colocado condições de validação de dados cadastrais obrigatórios e um componente critical informando que a sequência não pode ser interrompida. Caso o cadastro for efetuado com

sucesso o usuário opcionalmente pode cadastrar despesa (Apêndice HH). O cadastro da despesa pode ser efetuado a qualquer hora, mas tem que ser associado a uma nota fiscal. Também foi registrado o diagrama para editar nota fiscal (Apêndice II), buscar nota fiscal (Apêndice JJ) e excluir nota fiscal (Apêndice KK), todos considerados complementos, e útil para um bom entendimento do sistema.

A movimentação de entrada (Apêndice LL) pode ser efetuada de duas maneiras, ou pelo sistema, automaticamente através da finalização da solicitação ou manualmente inserindo os itens que compõem a nota fiscal. Para a entrada manual é gerado um código de entrada (mostrado no diagrama de sequência Gerar Código de Entrada – Apêndice MM), e após gerar o código o usuário efetuará a busca do produto e quantificará para registrar a movimentação. Para a entrada automática, também é gerado um código de entrada (Apêndice MM) e registrado como entrada o produto com sua quantidade. No diagrama para gerar código de entrada, se for informado à solicitação é necessário informar o cliente. Em casos de entradas manuais não é preciso associar cliente, pois é associado um fornecedor a nota fiscal referente a essas entradas.

A sequência da criação de fórmula (Apêndice NN) é necessário efetuar os cadastros iniciais sobre a fórmula, onde existe essa validação no diagrama e adicionar os itens de produtos necessário, sendo estes apenas do tipo insumo. Seguindo a mesma ideia é mostrado também o diagrama de buscar fórmula (Apêndice OO) e editar fórmula (Apêndice PP).

Para abrir a solicitação (Apêndice QQ) o usuário necessitará buscar a capacidade, buscar a fórmula para o sistema efetuar os cálculos de proporcionalidade e validar os estoques de insumo (Apêndice RR), onde é verificado se existe a quantidade de insumo desejada. Caso exista a quantidade de insumos necessários é gerado um código de saída e para cada produto é registrado uma saída (Apêndice SS) com o código de saída gerado.

Para finalizar uma solicitação (Apêndice TT), acontece uma busca nas solicitações (Apêndice UU), apenas nas abertas, e cadastrado o real produzido (Apêndice VV), posteriormente se necessário cadastra-se alguma avaria do líquido (Apêndice WW), depois é finalizado a solicitação. No cadastro do real produzido é feito a busca dos produtos e colocado sua quantidade, depois é recuperado o mesmo código de saída (Apêndice XX) usado para a saída de

insumos na criação da fórmula e para cada produto industrializado cadastrado é registrado a entrada (Apêndice LL) e para cada embalagens utilizadas é registrado a saída (Apêndice SS).

4.5 Diagrama de atividade

Nessa sessão é apresentado o fluxo de atividade de todas as funcionalidades encontradas no sistema, mostrando o ciclo de vida de objetos em determinada operação. Foram selecionados os principais casos de uso para serem modelados os diagramas de atividade, sendo apresentado com fluxo principal e secundário quando necessário.

No apêndice YY mostra o fluxo de trabalho a ser seguido para cadastrar o produto, onde é solicitado o cadastro do produto e aberto um formulário para o mesmo sendo preenchido e logo em seguida o usuário tem que cadastrar pelo menos uma unidade ao produto, havendo a possibilidade de colocar mais. A atividade cadastrar unidade por ser simples não foi apresentado como diagrama nesse trabalho. O usuário pode ainda, se desejar, registrar um estoque mínimo para o produto em questão, depois irá selecionar o tipo do produto e caso for produto industrializado irá chamar a atividade que associa embalagens e por fim salva os dados preenchidos registrando no objeto e no banco de dados.

No diagrama em relação à associação de embalagem (Apêndice ZZ), deve haver um pré-cadastro de um produto industrializado, ou seja, a atividade cadastrar produto deve ser chamada anteriormente e o tipo selecionado deve ser produto industrializado. Será feito uma busca de produtos do tipo embalagem, selecionar os necessários e confirmar a associação e o produto industrializado escolhido previamente. O usuário também possui fluxos secundários em que é adicionado mais embalagens ou excluindo se necessário.

No diagrama referente à criação de fórmula (Apêndice AAA), o usuário seleciona opção para determinada finalidade, fazendo com que o sistema abra um formulário em que será inserido os dados. Ao prosseguir, é revelado a composição da fórmula, onde o usuário poderá adicionar, remover, editar quaisquer insumo, quantificando os mesmo, e por fim poderá salvar a fórmula. É mostrado o fluxo principal para adicionar itens e salvar fórmula e secundários para excluir e editar itens.

Na atividade relacionada à abertura da solicitação (Apêndice BBB) é mostrado o formulário para cadastro inicial, realizando uma busca e seleção de cliente e associado à solicitação. Posteriormente é selecionada a iogurteira

mostrado em outro diagrama e selecionado a fórmula, também mostrado em outro diagrama. Por fim, o sistema confirma a abertura da solicitação e persiste os dados.

O diagrama possui fluxos secundários (caso ocorra erro ao selecionar a iogurteira e ao selecionar a fórmula), informando ao usuário sobre determinado erro, dando-lhe a opção de cancelar a abertura da solicitação ou escolher outra fórmula.

No diagrama relacionado a selecionar fórmula (Apêndice D2) é mostrado em conjunto com mais dois diagramas, validar estoque insumo e movimentar saída insumo/embalagem.

Para selecionar a fórmula é feito uma busca, podendo mostrar a fórmula detalhadamente se necessário. Após ter selecionado e confirmado a fórmula desejada, o diagrama de atividade irá chamar outro diagrama para validar estoque de insumo, diagrama esse que é carregado os dados da capacidade (iogurteira) escolhida anteriormente e dados da fórmula para obter quais insumos ela contém. O sistema efetuará um cálculo referente a proporção da capacidade da iogurteira para cada insumo e verifica a existência real do insumo no estoque. Caso todos possuam é confirmado a validação do estoque, fazendo com que o sistema gere um código de saída, persistindo no banco os dados do mesmo. É produzida uma lista de insumos com sua quantidade utilizada na fórmula e faz referência ao diagrama de atividade movimentar saída insumo/embalagem para cada item da lista, registrando como saída. Por fim é confirmado todas as saídas, retornando para a atividade de abrir solicitação. É importante lembrar que, caso o estoque não seja validado a atividade é encerrada e retorna para a seleção da fórmula.

Em relação ao diagrama de finalizar solicitação (Apêndice CCC), é selecionada uma solicitação que ainda está em aberta, possibilitando o cadastramento do real produzido mostrado noutra atividade. Caso o cadastro tenha ocorrido com sucesso confirma o fechamento da solicitação. Se necessário, o usuário poderá cadastrar avaria do líquido.

Quando o usuário cadastra o real produzido (Apêndice DDD), será gerado duas atividades paralelas: recuperar código de saída (Apêndice EEE) e gerar código de entrada.

Ao chamar o diagrama para recuperar o código de saída, o mesmo utilizado nessa solicitação para saída de insumos, será utilizado para as embalagens usadas nos produtos, efetuando o cálculo da quantidade de cada embalagem referente a cada produto, registrando no banco através do diagrama movimentar a saída insumo/embalagem mostrado anteriormente.

Em paralelo, ao utilizar o diagrama para gerar código de entrada, o código referente as entradas de produtos junto com data e informações relevantes irá persistir e posteriormente será gerado a movimentação em banco de todos os produtos industrializados gerados, demonstrado no diagrama de atividade movimentar entrada (Apêndice FFF).

No diagrama de atividade movimentar entradas percebe-se os seguintes fluxos: um direcionado para movimentação de entrada produto industrializado, ou seja, oriundos da solicitação, considera da entrada automática e um fluxo direcionado para a entrada manual (insumos e embalagens oriundas da nota fiscal, sendo selecionada a nota e adicionado pelo usuário os insumos e suas respectivas quantidades por fim sendo salvo no banco de dados).

5. CONCLUSÃO

As abordagens da Engenharia de Software utilizadas sobre a UML, com auxílio de ferramentas de modelagem, foram suficientes para o desenvolvimento do Estudo de Caso aplicada a empresa Sabor Laticínios Ltda, situada em tal lugar em cima do módulo do sistema de Produção.

Fica claro a importância da etapa de análise de requisitos e projeto para desenvolver um sistema de qualidade assegurada e que contemple todas as etapas e regras de negócios estabelecidas, podendo visualizar e controlar todo o processo de desenvolvimento de maneira eficaz e detalhada, estipulando e cumprindo prazos.

Por meio da abstração, pode-se perceber as melhorias para a empresa, sendo entendida por qualquer programador da equipe de desenvolvimento, observando a vantagem da orientação a objeto, facilitando quaisquer possíveis mudanças e melhorias ao longo da vida útil do software.

Com a modelagem proposta foi constatado que o sistema atual não atende as expectativas da empresa e nem suporta novas regras de negócio, pois serve apenas como repositório de dados, por exemplo, todas as solicitações são registradas após terem sido efetivamente terminadas, assim não existe mais uma validação efetiva do estoque. Também se verificou que no sistema atual acontece uma mistura dos módulos de produção, de logística, de vendas e de administração, gerando uma mistura das regras. Na figura abaixo mostra o cadastro de produto, onde há uma mistura de dados do produto com dados tributários.

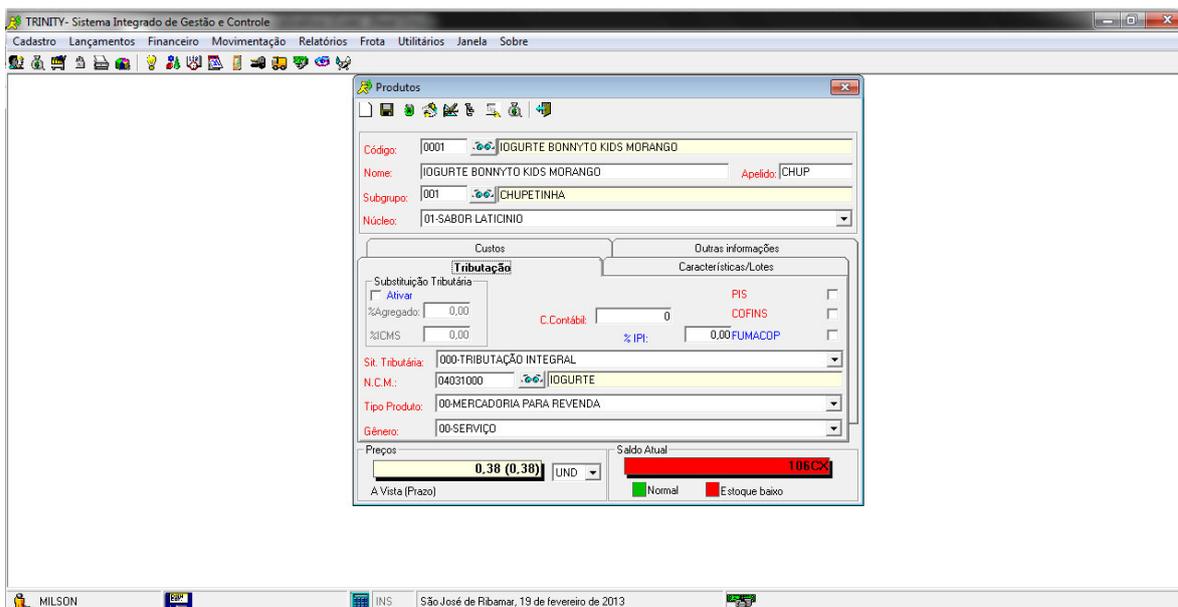


Figura 28 - Sistema desatualizado da Sabor Lactínios

Por ser um sistema web trará mobilidade, podendo ser acessado de qualquer lugar, com qualquer aparelho conectado a internet, diminuindo investimentos em computadores e hardwares, ganhando devido a realização periodicamente de backups, possibilitando acesso dos usuários aos dados com perfis previamente cadastrados, tornando flexível o crescimento, aumentando a capacidade do servidor sem uma aquisição grande. Com o sistema na internet o suporte se torna mais rápido não havendo a necessidade do deslocamento para realizá-lo, o mesmo pode ser dito para atualizações.

Com uma modelagem orientada a objeto, será necessário realizar uma linguagem atual no sistema, aumentando a capacidade de desenvolvimento de novas funcionalidades e de forma mais rápida, gerando assim uma manutenção e atualização mais eficaz e eficiente. Por ser desenvolvido numa linguagem mais atual, apresentará um layout moderno e enxuto, facilitando a usabilidade e melhorando a interface homem computador. Na figura 29 pode-se ver o print do novo sistema

A modelagem para orientação a objeto dará a empresa a possibilidade de evolução para sistemas com geração de gráficos e prospecção. servindo de documentação para manutenções futuras e entendimento da regra de negócio. Na figura 30 pode-se visualizar o novo sistema de produção.

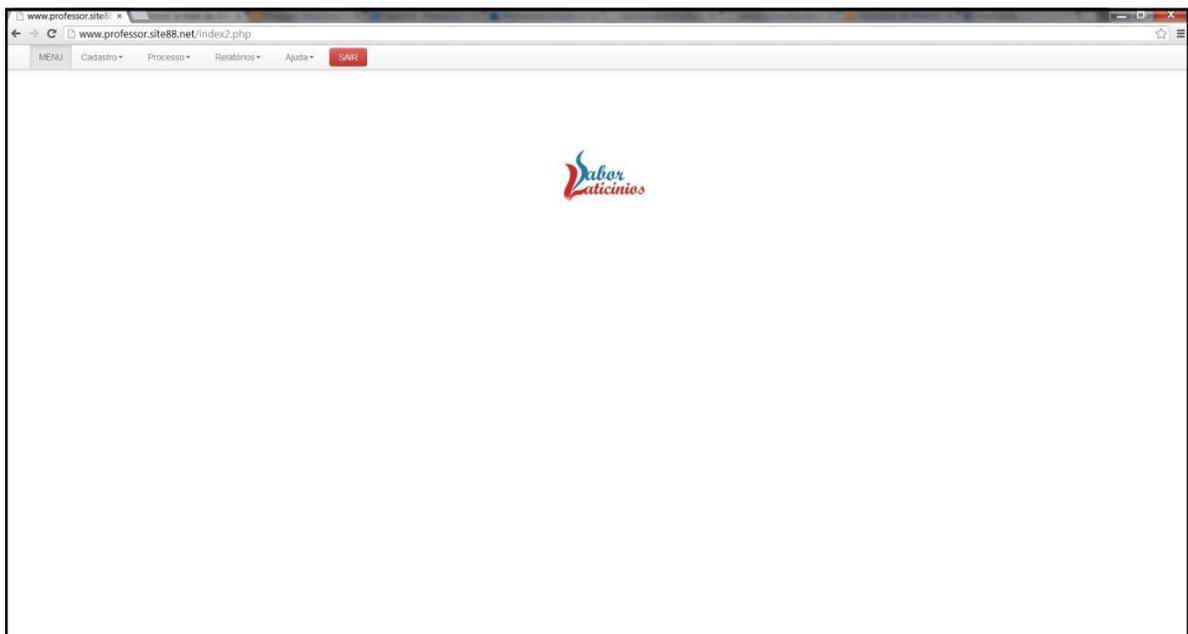


Figura 29 - Solicitação de produção com nova modelagem

O módulo de produção não dependerá de quaisquer dados oriundos de outros módulos, podendo ser aplicado a qualquer empresa que possua uma sistemática de produção de iogurte e com a saída de produtos desse módulo, ou seja, pode alimentar qualquer outro sistema.

A sistemática de produção no sistema foi mudada, adequando-se a necessidade da empresa. Os produtos foram divididos em mais categorias insumo (representam apenas os ingredientes que compõem o líquido), embalagem e produto industrializado. Nesse novo contexto de produção é cadastrado somente o real produzido, pois no sistema atual o procedimento era diferente, sendo informado quais os produtos industrializados desejariam que fosse produzido, no entanto, sempre era produzido uma quantidade taxada na iogurteira, e sempre gerava uma perda do líquido ou gerado produtos diferentes do informado, não equivalendo os dados do estoque com o que está no sistema. Para a nova modelagem, visando evitar desperdício de líquido ao ser gerado e tornando a exceção uma regra, foi feita uma mudança: ao invés de informar apenas quais produtos desejariam produzir, a solicitação seria sempre a quantidade total da iogurteira (que acontecia anteriormente) colocando como observação o que se pretende produzir, mas nada oficial, e para finalizar a mesma seria cadastrado somente o que realmente já foi produzido, não gerando um retrabalho, nem inconsistência de dados entre o sistema e o estoque. A figura

31 mostra o novo sistema em desenvolvimento referente a abertura de solicitação.

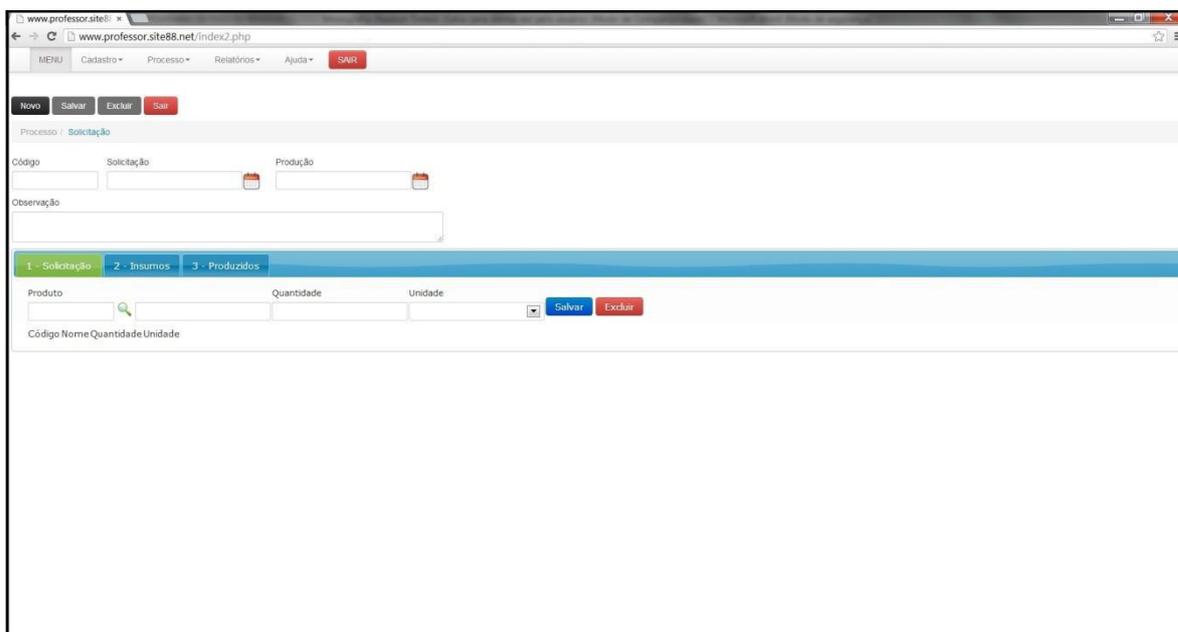


Figura 30 - Abertura de solicitação

O sistema trás nova possibilidade de registrar qualquer avaria do líquido, que antes não possuía. A modelagem proposta serviu de base sólida para o desenvolvimento do software (etapa iniciada), clareando as dúvidas e já satisfazendo seus usuários, que vem acompanhando o processo. Foi observado a importância de ferramenta de modelagem UML tornando a diagramação mais rápida e eficiente. Ressalto ainda que nesse trabalho não foi tratado nenhum processo de desenvolvimento de software, deixando para trabalhos futuros.

O trabalho possibilitou uma maior compreensão do autor sobre a UML e a ferramenta proposta, com potencial da utilização profissional desses conhecimentos, e contribuição para migração de um sistema desatualizado e incompleto para uma visão orientada a objeto, utilizando ferramentas propostas pela UML.

É proposto como sugestão para trabalhos futuros a modelagem dos outros módulos, bem como análise de resultados depois do uso dos novos sistemas e também é sugerido um trabalho sobre um processo de desenvolvimento.

REFERÊNCIAS

ASTAH, tutorial do astah professional

Disponível em: <http://astah.net/tutorials>. Acesso em 23/04/13

BATISTA, E.O. **Sistema de Informação**: o uso consciente da tecnologia para o gerenciamento. São Paulo: Saraiva, 2004.

BERNERS-LEE, T.; CAILLIAU, R.; LUOTONEN, A. et al. **The WorldWide Web. Communications of the ACM**. 1994.

BERNERS-LEE, T.; F, M. **Weaving the Web**: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor. Harper, San Francisco, 1999.

Bezerra, E. **“Princípios de análise e projeto de sistemas com UML”**. Rio de Janeiro: Elsevier, 2007.

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML Guia do Usuário**. Rio de Janeiro: Editora Campus, 2. ed. 2006.

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I.: **The Unified Modeling Language Reference Manual, Second Edition**. Ed. Addison-Wesley, 2004.

BRODIE, M; STONOBRAKER, M. **Migrating Legacy Systems**: gateways, interfaces and the Incremental approach, morgan kaufmann Publishers, USA, 1995.

CONALLEN, J. **Desenvolvendo aplicações web com UML**. Rio de Janeiro: Campus, 2003.

DAVENPORT, T. H; PRUSAK, L. **Conhecimento empresarial**. Tradução de Lenke Peres. Rio de Janeiro: Campus; São Paulo: Publifolha, 1999.

GINIGE, M.; GINIGE, A.; MURUGESAN, S. Web Enguneering: An introduction. **IEEE Multimidia**, v. 8, (Issue 1), Jan/Mar. 2001.

GUEDES, G. **"UML, uma abordagem Prática"**. Editora Novatec. 2004.

GUEDES, G. T. A. **UML 2**: uma abordagem prática. Novatec Editora, 2011.

ISAKOWITZ, T.; BIEBER, M.; VITALI, F. Web Information Systems. **Communications of the ACM**, New York; v. 41, n. 7, p. 78-80, Jul. 1998.

LACOMBE, F.J.M.; HEILBORN, G.L.J. **Administração**: princípios e tendências. São Paulo: Saraiva, 2003.

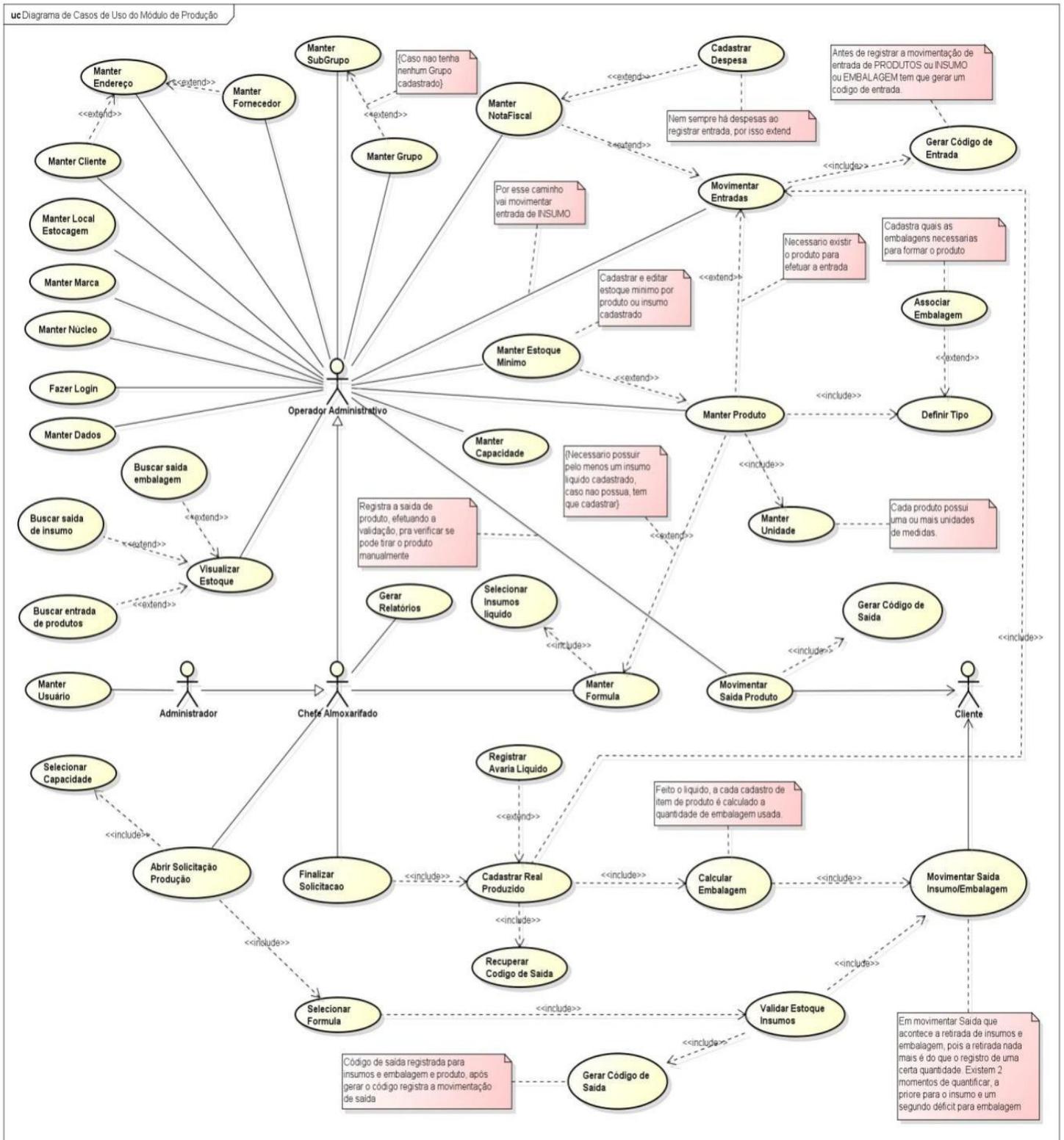
- LARMAN, G. **Utilizando UML e Padroes**: uma introdução à análise e ao projeto orientados a objetos. Porto Alegre: Bookman, 2000.
- LAUDON, K. C.; LAUDON, J. P. **Gerenciamento de sistemas de informação**. 3. ed. LTC: Rio de Janeiro, 2001.
- LAUDON, K. C.; LAUDON, J. P. **Sistemas de informação**. 4. ed. LTC: Rio de Janeiro, 1999.
- MATOS, A. V. **Unified Modeling Language Prático e Descomplicado**. 1. ed. São Paulo: Érica, 2002.
- MAXIMIANO, A. C. A. **Teoria Geral da Administração**: da revolução urbana à revolução digital. São Paulo: Atlas, 2007
- MORAES, A. M. P. **Introdução à Administração**. São Paulo: Prentice Hall, 2004.
- O'BRIEN, J. A. **Sistemas de informação e as decisões gerenciais na era da internet**. Tradução de Cid Knipel Moreira. São Paulo: Saraiva, 2002.
- OLIVEIRA, D. P. R. **Sistemas de informação gerenciais**: estratégias, táticas, operacionais. 8. ed., São Paulo: Atlas, 1992.
- OLIVEIRA, D. P. R. **Sistemas, organização e métodos**: uma abordagem gerencial. 13. ed. São Paulo: Atlas, 2002.
- PADOVEZE, C. L. **Contabilidade gerencial**: um enfoque e sistemas de informação contábil. São Paulo: Atlas, 1997.
- PENDER, T. **UML a Bíblia**. Rio de Janeiro: Editora Campus, 2004.
- PESQUISA Anual CIA - Escola de Administração de Empresas de São Paulo, Fundação Getúlio Vargas – EAESP. 23. ed. 2012.
- PRESSMAN, R. S. **Engenharia de software**. Rio de Janeiro: McGraw-Hill, 2002.
- REZENDE, D. A. **Planejamento de Sistemas de Informação e Informática**. São Paulo: Atlas, 2003.
- SCHAICOSKI, J. C. **A utilização do ROI na análise de projetos de tecnologia da informação**. 2002. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, SC, 2002.
- SOMMERVILLE, I. **Engenharia de Software**. Ed. Addison-Wesley, 2003
- SUWARDY, T.; RATNATUNGA, J.; SOHAL, A. S. et al. Projects: evaluation, outcomes and impediments. **Benchmarking: An International Journal**, v.10, n.4, p. 325-42.2003. Disponível em: <http://www.emeraldinsight.com/1463-5771.htm>. Acesso em 05/03/13.

UMAR, A. **Applications Reengineering – Building Web-Based Applications and Dealing With Legacy**. Prentice Hall, 1997.

UOL Economia. Gartner: Investimento em TI no Brasil deve subir 10% ao ano até 2014. 2011. Disponível em: <http://economia.uol.com.br/ultimas-noticias/valor/2011/10/25/gartner-investimento-em-ti-no-brasil-deve-subir-10-ao-ano-ate-2014.jhtm>.

VON BERTALANFFY, L. **Teoria geral dos sistemas: fundamentos, desenvolvimento e aplicações**. Petrópolis: Vozes. 2009

APÊNDICE A: Diagrama de caso de uso do Sistema de Produção



APÊNDICE B: Descrição do caso de uso Selecionar Insumos Líquido

| ITEM | VALUE |
|---------------------|---|
| Caso de Uso | Selecionar Insumos líquido |
| Resumo | Usuário efetua uma busca nos insumos e seleciona um. |
| Ator(s) | |
| Pre-condição | Chamar caso de uso Manter Fórmula anteriormente. |
| Pós-condição | |
| Fluxo Principal | Usuário efetua uma busca. Sistema retorna os insumos mediante a busca. Usuário seleciona um insumo e sistema volta para o caso de uso anterior retornando o dado selecionado. |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de Nota | |

APÊNDICE C: Descrição do caso de uso Selecionar Fórmula

| ITEM | VALUE |
|-------------------|--|
| Caso de Uso | Selecionar Formula |
| Resumo | Ator seleciona fórmula que será usada na solicitação |
| Ator(s) | |
| Pre-condição | O caso de uso Abrir Solicitação e Selecionar Capacidade devem ser chamados anteriormente |
| Pós-condição | Junta os dados da capacidade e formula e chamará o caso de uso Validar Estoque Insumo |
| Fluxo Principal | Ator busca as fórmulas já cadastradas no sistema. Sistema informa quais as fórmulas já cadastradas mediante a busca. Ator seleciona uma fórmula e manda prosseguir. Caso de uso Validar Estoque Insumos será chamado. |
| Fluxo Alternativo | Ao efetuar a proporção e não foi selecionado a capacidade, sistema emite uma mensagem para o usuário, informando que deve ser selecionado a capacidade primeiramente. |
| Restrições | Existir pelo menos uma formula cadastrada. |
| Inclui Caso de | Validar Estoque Insumos |
| Nota | |

APÊNDICE D: Descrição do caso de uso Selecionar Capacidade

| ITEM | VALUE |
|---------------------|---|
| Caso de Uso | Selecionar Capacidade |
| Resumo | Ator seleciona a capacidade usada para a solicitação |
| Ator(s) | |
| Pre-condição | Caso de uso Abrir Solicitação deve ser chamado anteriormente |
| Pós-condição | Confirma a capacidade escolhida e adiciona na solicitação. |
| Fluxo Principal | Ator busca as capacidades que estão cadastradas no sistema. Sistema informa as capacidades. Ator seleciona uma unica capacidade e manda prosseguir. |
| Fluxo Alternativo | |
| Restrições | Existir pelo menos uma capacidade cadastrada previamente. |
| Inclui Caso de Nota | |

APÊNDICE E: Descrição do caso de uso Registrar Avaria Líquido

| ITEM | VALUE |
|---------------------|---|
| Caso de Uso | Registrar Avaria Líquido |
| Resumo | Registra a quantidade de líquido que foi perdida |
| Ator(s) | |
| Pre-condição | O caso de uso Finalizar Solicitação deve ser chamado anteriormente. |
| Pós-condição | |
| Fluxo Principal | Usuário opta por registrar avaria. Sistema mostra um formulário de cadastro de avaria do líquido. Usuário informa a quantidade, o motivo, e quaisquer outras observações. |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de Nota | |

APÊNDICE F: Descrição do caso de uso Recuperar Código de Saída

| ITEM | VALUE |
|-------------------|--|
| Caso de Uso | Recuperar Código de Saída |
| Resumo | Recupera o código de saída utilizado para saída de insumos na mesma solicitação |
| Ator(s) | |
| Pre-condição | Caso de uso Cadastrar Real Produzido tem que ser chamado anteriormente |
| Pós-condição | |
| Fluxo Principal | Sistema verifica qual solicitação o usuário está modificando e busca o código de saída utilizado para a saída de insumos. Ao achar o código de saída atualiza com alguma observação a mais e retorna para o caso de uso anterior passando o código de saída. |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de | |
| Nota | Geralmente no mesmo dia que é aberto a solicitação é fechado, por isso só uma data no objeto CódigoSaída, mas pode fugir a regra, para isso foi criado o atributo de observação. |

APÊNDICE G: Descrição do caso de uso Movimentar Saída Produto

| ITEM | VALUE |
|-------------------|---|
| Caso de Uso | Movimentar Saída Produto |
| Resumo | Registrar a saída de quaisquer tipos de produtos |
| Ator(s) | Operador Administrativo Cliente |
| Pre-condição | |
| Pós-condição | |
| Fluxo Principal | <p>Usuário seleciona em registrar saída.</p> <p>Sistema abre um grid vazio e usuário busca o(s) produto(s), podendo ser de quaisquer tipo e informa a quantidade por produto e adiciona.</p> <p>Para cada produto adicionado o sistema adiciona o mesmo no grid para melhor visualização do usuário. Usuário associa a saída a um cliente, e pode colocar uma observação.</p> <p>Usuário salva.</p> <p>Sistema solicita confirmação e chama caso de uso Gerar Código de Saída e logo depois registra a saída, sendo cada produto com sua quantidade uma saída, porém todas com o mesmo código de saída.</p> |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de | Gerar Código de Saída |
| Nota | A saída pode ser feita de qualquer produto, por diversos motivos, por exemplo, saída da produção, extraio, avaria, roubo, etc. |

APÊNDICE H: Descrição do caso de uso Movimentar Saída Insumo/Embalagem

| ITEM | VALUE |
|-------------------|---|
| Caso de Uso | Movimentar Saída Insumo/Embalagem |
| Resumo | Registra em banco todas as saídas dos itens |
| Ator(s) | Operador Administrativo Cliente |
| Pre-condição | Se for saída de insumo: O caso de uso Gerar Código de Saída deve ser chamado anteriormente Se for saída de embalagem: O caso de uso Recuperar Código de Saída deve ser chamado anteriormente |
| Pós-condição | Se for saída de insumo: Confirma a abertura da solicitação, permitindo a retirada pelo estoquista dos insumos. Se for saída de embalagem: Confirma a finalização da |
| Fluxo Principal | Se for saída de insumo: Sistema recebe a lista já calculada dos insumos necessários e para cada insumo, efetua a inserção no banco da retirada de cada insumo com sua quantidade. Gerando assim várias saídas, todas com sua chave primária, mas ligados a um único código de saída. |
| Fluxo Alternativo | |
| Restrições | Conexão com o banco não pode ser mantida. Para a saída de embalagem deve ser feito a abertura de uma solicitação. |
| Inclui Caso de | |
| Nota | É usado o mesmo código de saída de embalagem e insumo, contanto que seja a mesma solicitação. |

APÊNDICE I: Descrição do caso de uso Movimentar Entradas

| ITEM | VALUE |
|--------------------|---|
| Caso de Uso | Movimentar Entradas |
| Resumo | Movimentar a entrada de insumos, embalagens ou produtos industrializados |
| Ator(s) | Operador Administrativo |
| Pre-condição | Usuário precisa estar logado. |
| Pós-condição | |
| Fluxo Principal | <p>Se for entrada Manual:</p> <p>Usuário seleciona inserir entradas. Caso não tenha cadastrado nenhuma nota fiscal se faz necessário cadastrar.</p> <p>Usuário seleciona uma nota fiscal para associar as entradas.</p> <p>Usuário informa os insumos e embalagens com suas respectivas quantidades e manda salvar.</p> <p>Sistema solicita confirmação das entradas.</p> <p>Usuário confirma entradas.</p> <p>Chama caso de uso Gerar Código de Entrada.</p> <p>Sistema efetua o registro de todas as entradas, todas com sua chave primária e com o mesmo código de entrada</p> |
| Fluxo Alternativo | |
| Restrições | Para cada entrada, deve ter o produto cadastrado no sistema. |
| Inclui Caso de Uso | <p>Manter Produto</p> <p>Gerar Código de Entrada</p> |
| Nota | <p>A entrada de insumo e embalagens é feita de forma manual, através do cadastro dos itens da nota fiscal, sendo necessário cadastrar primeiramente a nota fiscal.</p> <p>A entrada dos produtos industrializados só é feita pelo sistema, através do fechamento de uma solicitação.</p> |

APÊNDICE J: Descrição do caso de uso Manter Unidade

| ITEM | VALUE |
|-------------------|--|
| Caso de Uso | Manter Unidade |
| Resumo | Cadastra todas as unidades que um determinado produto pode possuir, para ser informado posteriormente no registro das entradas. |
| Ator(s) | Operador Administrativo |
| Pre-condição | Chamar caso de uso Manter Produto anteriormente. |
| Pós-condição | |
| Fluxo Principal | Usuário seleciona adicionar nova unidade. Sistema mostra um formulário de cadastro de unidade. Usuário informa os dados e salva. |
| Fluxo Alternativo | |
| Restrições | Pelo menos uma unidade tem que ser cadastrado por produto. |
| Inclui Caso de | |
| Nota | Pode ser adicionado quantas unidades forem necessárias para cada produto. Exemplos de unidade: unitário, quilo, tonelada, gramas, caixas, etc. |

APÊNDICE K: Descrição do caso de uso Manter Produto

| ITEM | VALUE |
|--------------------|--|
| Caso de Uso | Manter Produto |
| Resumo | Usuário cadastra, edita ou exclui (inativa) todos os produtos no sistema, caracterizando como insumo, embalagem ou produto acabado (industrializado) |
| Ator(s) | Operador Administrativo |
| Pre-condição | Usuário precisa está logado. |
| Pós-condição | |
| Fluxo Principal | Usuário seleciona em cadastrar novo produto. Sistema mostra um formulário para cadastro. Usuário preenche dados do produto e informa o tipo do produto chamando o caso de uso Definir Tipo. Usuário chama caso de uso Manter Unidade. Usuário terminou de preencher todas as informações necessárias e manda salvar. Sistema solicita confirmação e registra dos dados. |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de Uso | Manter Unidade Definir Tipo |
| Nota | |

APÊNDICE L: Descrição do caso de uso Manter Fórmula

| ITEM | VALUE |
|-------------------|--|
| Caso de Uso | Manter Formula |
| Resumo | Usuário pode cadastrar, editar ou excluir (inativar) uma fórmula |
| Ator(s) | Chefe Almoxarifado |
| Pre-condição | Usuário precisa estar logado |
| Pós-condição | |
| Fluxo Principal | <p>Usuário seleciona cadastrar fórmula.</p> <p>Sistema mostra formulário para cadastro.</p> <p>Usuário seleciona os insumos chamando o caso de uso Selecionar Insumo Liquido e informa suas respectivas quantidades e manda salvar.</p> <p>Sistema solicita confirmação.</p> <p>Usuário confirma e sistema registra a fórmula no banco de dados.</p> |
| Fluxo Alternativo | |
| Restrições | Insumos devem ser cadastrados previamente, por isso o extend para o caso de uso Manter Produto para caso não exista produto cadastrado, irá cadastrar primeiramente. |
| Inclui Caso de | Selecionar Insumos liquido |
| Nota | Na fórmula contém os insumos e sua quantidade em proporção de 1000L |

APÊNDICE M: Descrição do caso de uso Manter Estoque Mínimo

| ITEM | VALUE |
|------------------------|---|
| Caso de Uso | Manter Estoque Mínimo |
| Resumo | Registra qual a quantidade mínima o produto deve possuir em estoque, para melhor controle da produção e das entradas. |
| Ator(s) | Operador Administrativo |
| Pre-condição | Produto já deve está cadastrado |
| Pós-condição | |
| Fluxo Principal | Usuário seleciona o produto e insere a quantidade mínima desse produto em estoque. |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de Nota | |

APÊNDICE N: Descrição do caso de uso Gerar Código de Saída

| ITEM | VALUE |
|-------------------|---|
| Caso de Uso | Gerar Código de Saída |
| Resumo | Gera um código de saída para todas os itens de saída do estoque |
| Ator(s) | |
| Pre-condição | O caso de uso Validar Estoque Insumo deve ser chamado anteriormente |
| Pós-condição | |
| Fluxo Principal | Sistema verifica qual o ultimo código de saída gerado no sistema e carregado em memória o proximo sequencial, junto com dados como data da saída, tipo de saída, observação e o cliente responsável pela solicitação e insere no banco. |
| Fluxo Alternativo | Se ocorrer algum problema ao inserir os dados do código de saída no banco, cancela toda a operação. |
| Restrições | Conexão com o banco deve ser mantida o tempo todo. |
| Inclui Caso de | |
| Nota | |

APÊNDICE O: Descrição do caso de uso Finalizar Solicitação

| ITEM | VALUE |
|-------------------|--|
| Caso de Uso | Finalizar Solicitacao |
| Resumo | Finaliza uma solicitação que estava em aberto e a torna fechada, registrando os produtos industrializados que foram produzidos e as embalagens utilizadas. |
| Ator(s) | Chefe Almoxarifado |
| Pre-condição | Deve ser aberto uma solicitação primeiramente, para poder finalizar. |
| Pós-condição | Confirma a geração dos produtos para o sistema e para o usuário. |
| Fluxo Principal | Ator busca uma solicitação que está em aberto. Sistema retorna todas as solicitações em aberto. Ator seleciona uma solicitação. Ator opta por finalizar a solicitação. É chamado o caso de uso Cadastrar Real Produzido. |
| Fluxo Alternativo | Caso nao haja nenhuma solicitação em aberto, sistema informa e usuário nao faz mais nada. |
| Restrições | |
| Inclui Caso de | Cadastrar Real Produzido |
| Nota | |

APÊNDICE P: Descrição do caso de uso Definir Tipo

| ITEM | VALUE |
|-------------------|--|
| Caso de Uso | Definir Tipo |
| Resumo | Responsável para definir o tipo do produto, se é insumo, embalagem ou produto industrializado. |
| Ator(s) | |
| Pre-condição | Chamar caso de uso Manter Produto anteriormente |
| Pós-condição | Caso seja produto industrializado chama o caso de uso Associar Embalagem |
| Fluxo Principal | Usuário seleciona um dos possíveis tipos: Insumo, embalagem ou produto industrializado, podendo ser apenas um. E manda salvar. |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de | |
| Nota | Ao editar o produto, pode-se também editar as embalagens associados ao produto. |

APÊNDICE Q: Descrição do caso de uso Calcular Embalagem

| ITEM | VALUE |
|-------------------|--|
| Caso de Uso | Calcular Embalagem |
| Resumo | Recebe o produto produzido e ja cadastrado em memória pelo usuário e efetua o cálculo da quantidade de embalagens utilizada |
| Ator(s) | |
| Pre-condição | Caso de uso Cadastrar Real Produzido e Recuperar Codigo de Saída deve ser chamado anteriormente |
| Pós-condição | |
| Fluxo Principal | Sistema recebe o produto cadastrado com sua quantidade e efetua o cálculo da quantidade de embalagens e chama o caso de uso Movimentar Saída Insumo/Embalagens |
| Fluxo Alternativo | |
| Restrições | |
| Inclui Caso de | Movimentar Saída Insumo/Embalagem |
| Nota | |

APÊNDICE R: Descrição do caso de uso Cadastrar Real Produzido

| ITEM | VALUE |
|--------------------|--|
| Caso de Uso | Cadastrar Real Produzido |
| Resumo | É cadastrado o que já foi produzido. |
| Ator(s) | |
| Pre-condição | Necessário chamar primeiro o caso de uso Finalizar Solicitação |
| Pós-condição | |
| Fluxo Principal | <p>Ator busca os produtos.</p> <p>Sistema mostra os produtos.</p> <p>Ator seleciona um produto e informa a quantidade que foi produzida e se desejar adiciona mais um produto, efetuando outra busca.</p> <p>Sistema regista cada produto adicionado em memória.</p> <p>Ator manda salvar.</p> <p>Sistema solicita confirmação.</p> <p>Ator confirma e chama o caso de uso Recuperar Código de Saída.</p> <p>Após terminar o caso de uso Recuperar Código de Saída, recebe o código de saída e chama caso de uso Movimentar Entradas e Calcular Embalagem paralelamente.</p> |
| Fluxo Alternativo | Caso haja avaria do líquido, é registrado a avaria do líquido chamando o caso de uso Registrar Avaria Líquido. |
| Restrições | O produto deve ser cadastrado previamente no sistema. |
| Inclui Caso de Uso | <p>Movimentar Entradas</p> <p>Recuperar Código de Saída</p> <p>Calcular Embalagem</p> |
| Nota | Pode acontecer avaria do líquido quando a iogurteira tem uma capacidade de 1000L e todos os produtos produzidos usaram 999L, ou seja, tem 1L que será desperdiçado por não se encaixar em nenhum dos produtos |

APÊNDICE S: Descrição do caso de uso Associar Embalagem

| ITEM | VALUE |
|-------------------|---|
| Caso de Uso | Associar Embalagem |
| Resumo | Associa as embalagens para um determinado produto industrializado selecionado anteriormente. |
| Ator(s) | |
| Pre-condição | Deve ser chamado o caso de uso Definir Tipo anteriormente. |
| Pós-condição | |
| Fluxo Principal | Usuário busca as embalagens. Sistema informa as embalagens mediante a busca. Usuário seleciona as embalagens de acordo com o produto e manda salvar. Sistema registra e volta para o caso de uso que o chamou. |
| Fluxo Alternativo | |
| Restrições | Cadastro de embalagens deve ter sido feita anteriormente. |
| Inclui Caso de | |
| Nota | A quantidade de embalagens será sempre unitária, exemplo selecionando para o produto chupetinha, um recipiente e um rótulo, não sendo necessário especificar a quantidade. |

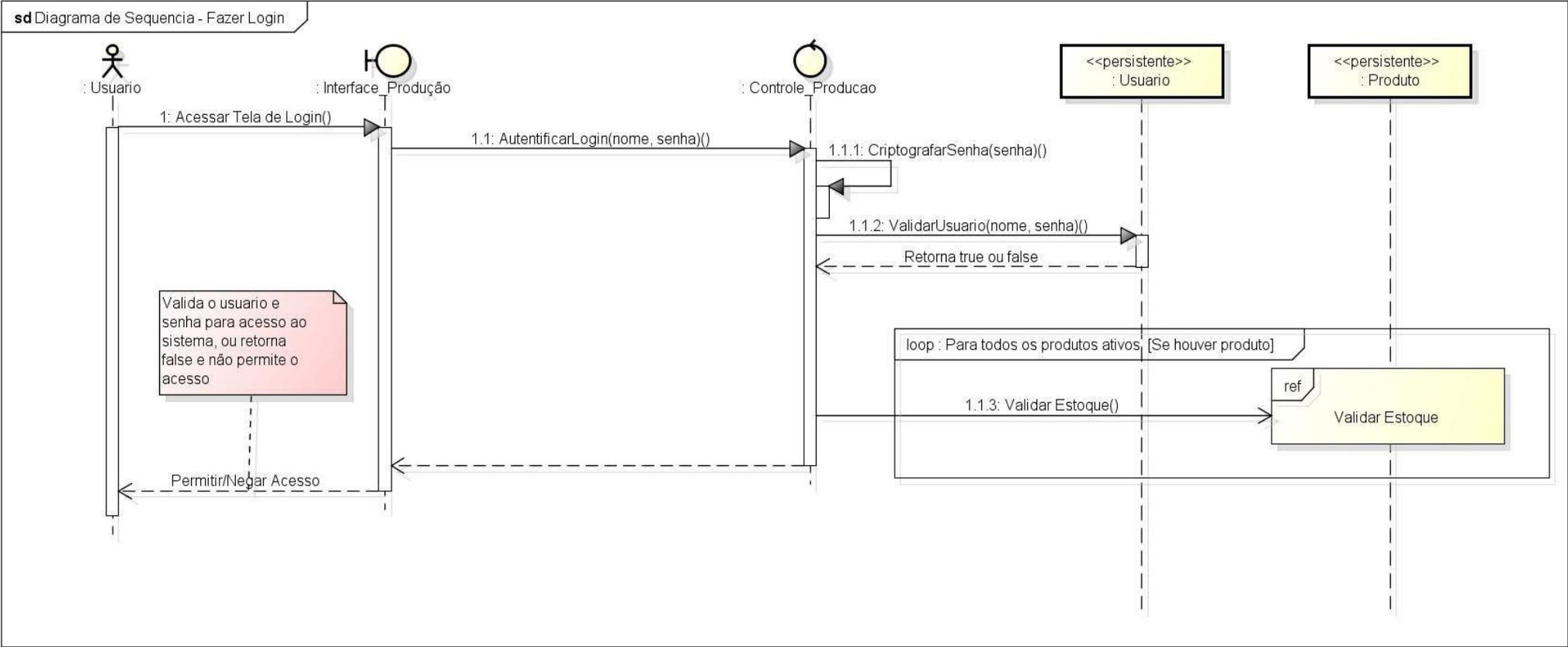
APÊNDICE T: Descrição do caso de uso Abrir Solicitação Produção

| ITEM | VALUE |
|--------------------|---|
| Caso de Uso | Abrir Solicitação Produção |
| Resumo | Ator abre uma solicitação e passa os dados e espera confirmação e autorização do sistema para a produção. |
| Ator(s) | Chefe Almoxarifado |
| Pre-condição | Usuário precisa estar logado. |
| Pós-condição | Confirmação de abertura, autorização para retirada de itens do estoque. |
| Fluxo Principal | Ator seleciona no sistema a opção de abrir solicitação. Sistema abre formulário para cadastro de dados da solicitação, selecionando o cliente. Ator deve executar caso de uso Selecionar Capacidade. Ator deve executar caso de uso Selecionar Formula. |
| Fluxo Alternativo | |
| Restrições | Se não possui capacidade, produto, formula e cliente cadastradas não permite abrir a solicitação. |
| Inclui Caso de Uso | Selecionar Formula Selecionar Capacidade |
| Nota | |

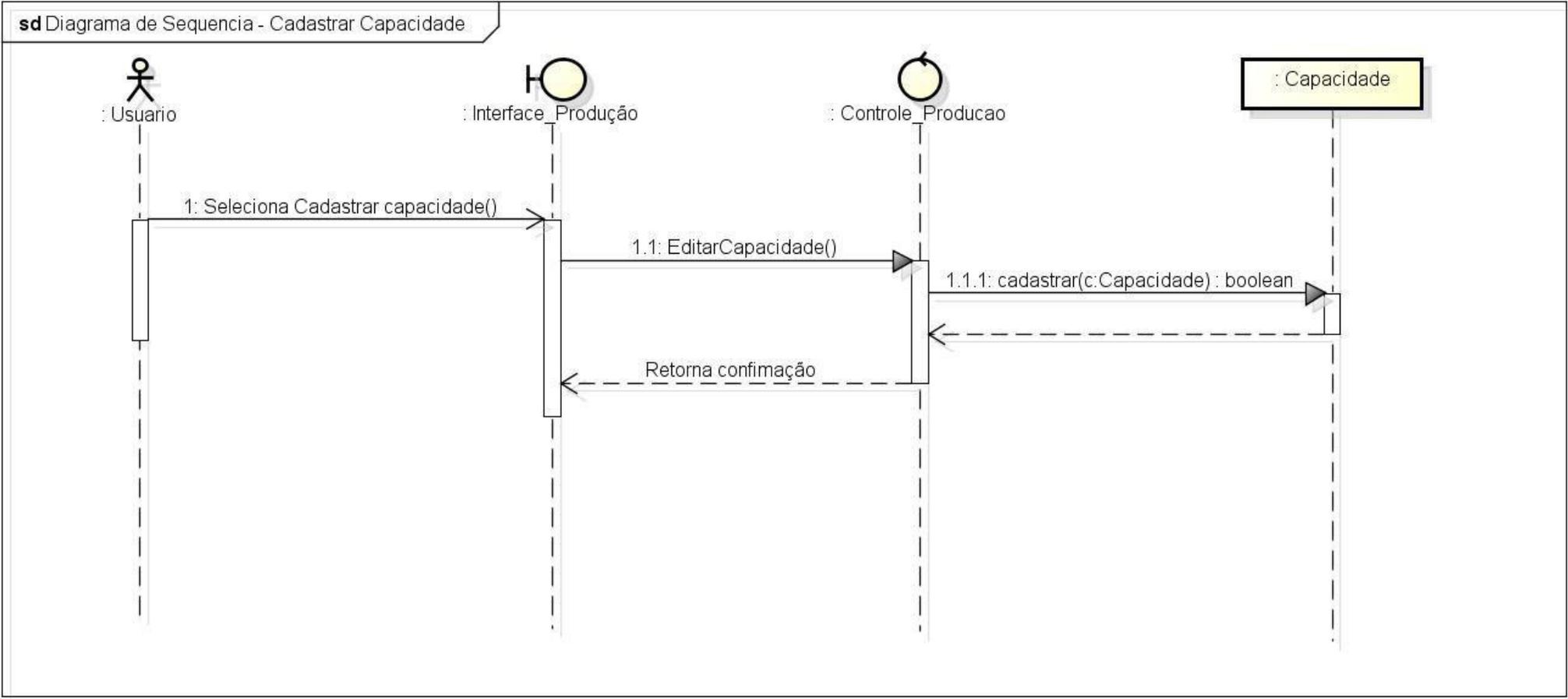
APÊNDICE U: Descrição do caso de uso Validar Estoque Insumos

| ITEM | VALUE |
|--------------------|---|
| Caso de Uso | Validar Estoque Insumos |
| Resumo | Efetuará a proporção dos insumos para a fórmula e autorizará ou não a produção da solicitação |
| Ator(s) | |
| Pre-condição | Caso de uso Selecionar Fórmula deve ser chamado anteriormente. |
| Pós-condição | Confirma a retirada de insumos e autoriza a produção da solicitação |
| Fluxo Principal | <p>Sistema verifica qual a capacidade escolhida.</p> <p>Sistema verifica qual a fórmula escolhida.</p> <p>Sistema busca a quantidade de insumos utilizado para a fórmula selecionada.</p> <p>Sistema efetua o cálculo das quantidades de insumos com a capacidade encontrando a verdadeira quantidade de insumos para utilizar naquela solicitação, gerando uma lista.</p> <p>Após encontrar a quantidade, valida se existe em estoque, se existir todos os insumos permite autorizar a solicitação.</p> <p>Chama o caso de uso Gerar Código de Saída.</p> <p>Após concluir o caso de uso Gerar Código de Saída, chama o caso de uso Movimentar Saída Insumo/Embalagem.</p> |
| Fluxo Alternativo | <p>Caso a quantidade de insumo não seja validada, sistema avisa ao usuário a falta de insumos no estoque e cancela a operação.</p> <p>Se não for gerado um código de saída, não é chamado o caso de uso Movimentar Saída Insumo/Embalagem.</p> |
| Restrições | |
| Inclui Caso de Uso | Gerar Código de Saída Movimentar Saída Insumo/Embalagem |
| Nota | Toda fórmula é feita em proporção de mil litros, e a capacidade (iorguteira) varia nessa proporção, por isso é calculado a proporção para saber a real quantidade de insumos utilizados. |

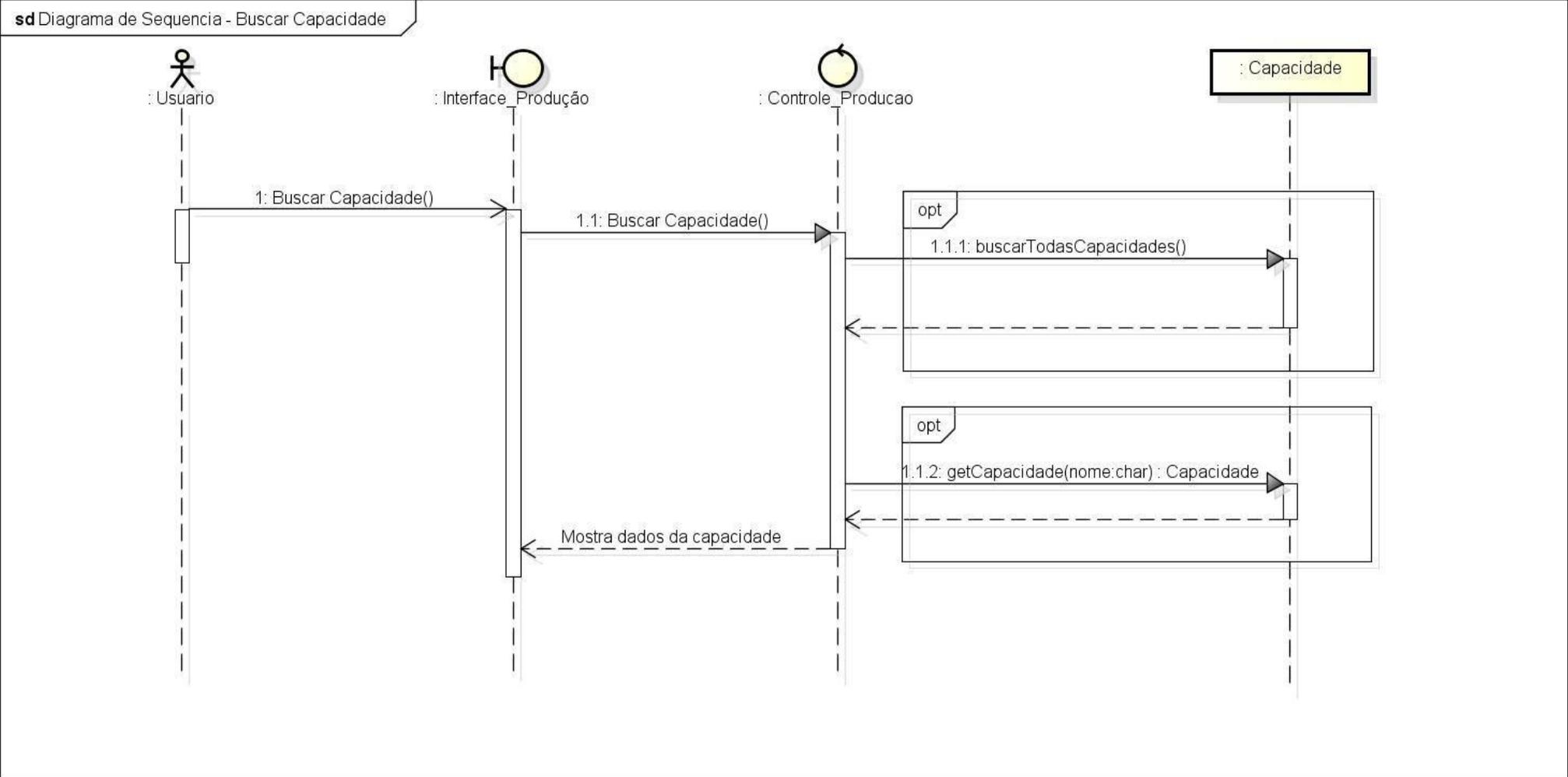
APÊNDICE X: Diagrama de Sequência Fazer Login



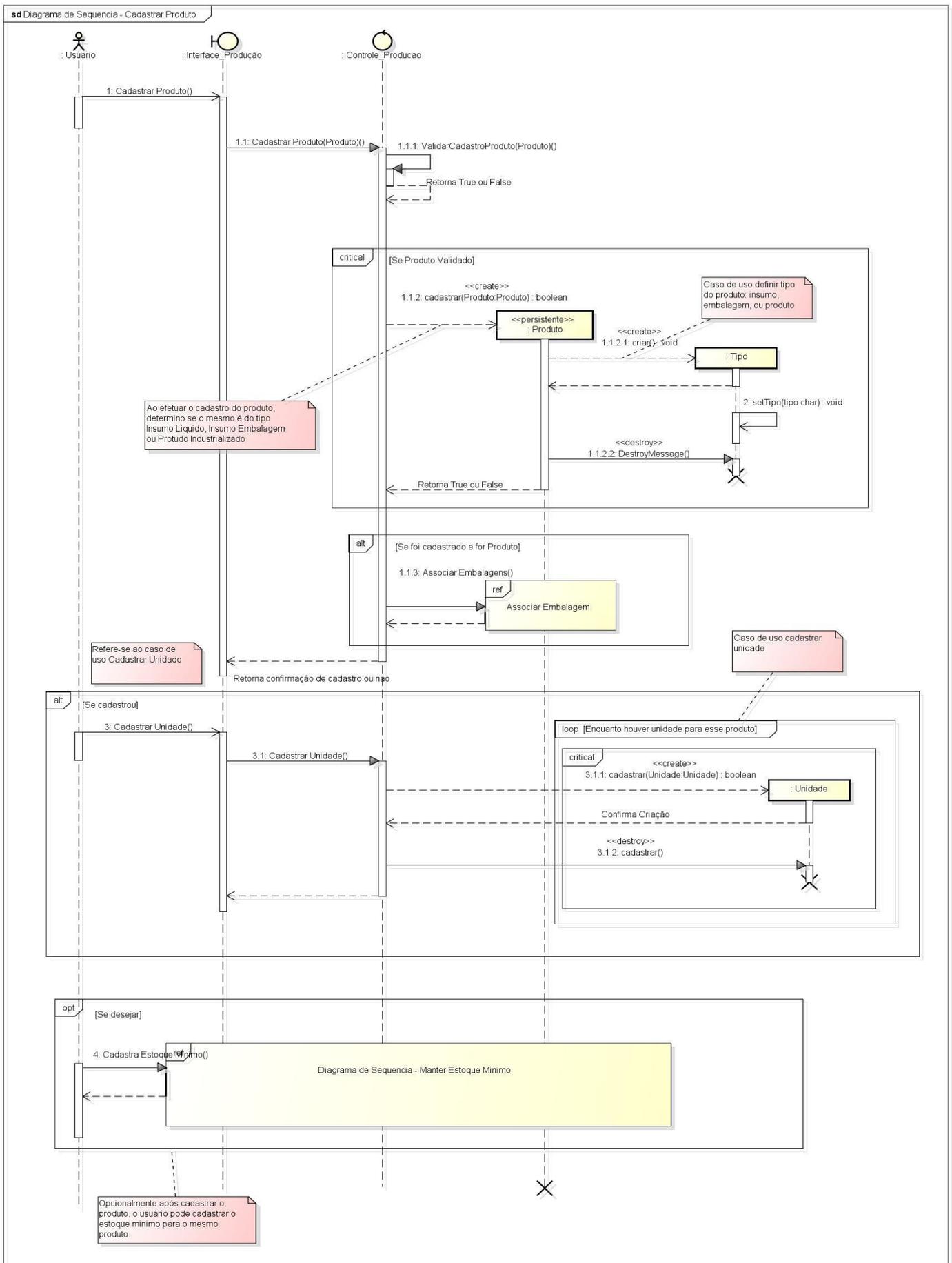
APÊNDICE Y: Diagrama de Sequência Cadastrar Capacidade



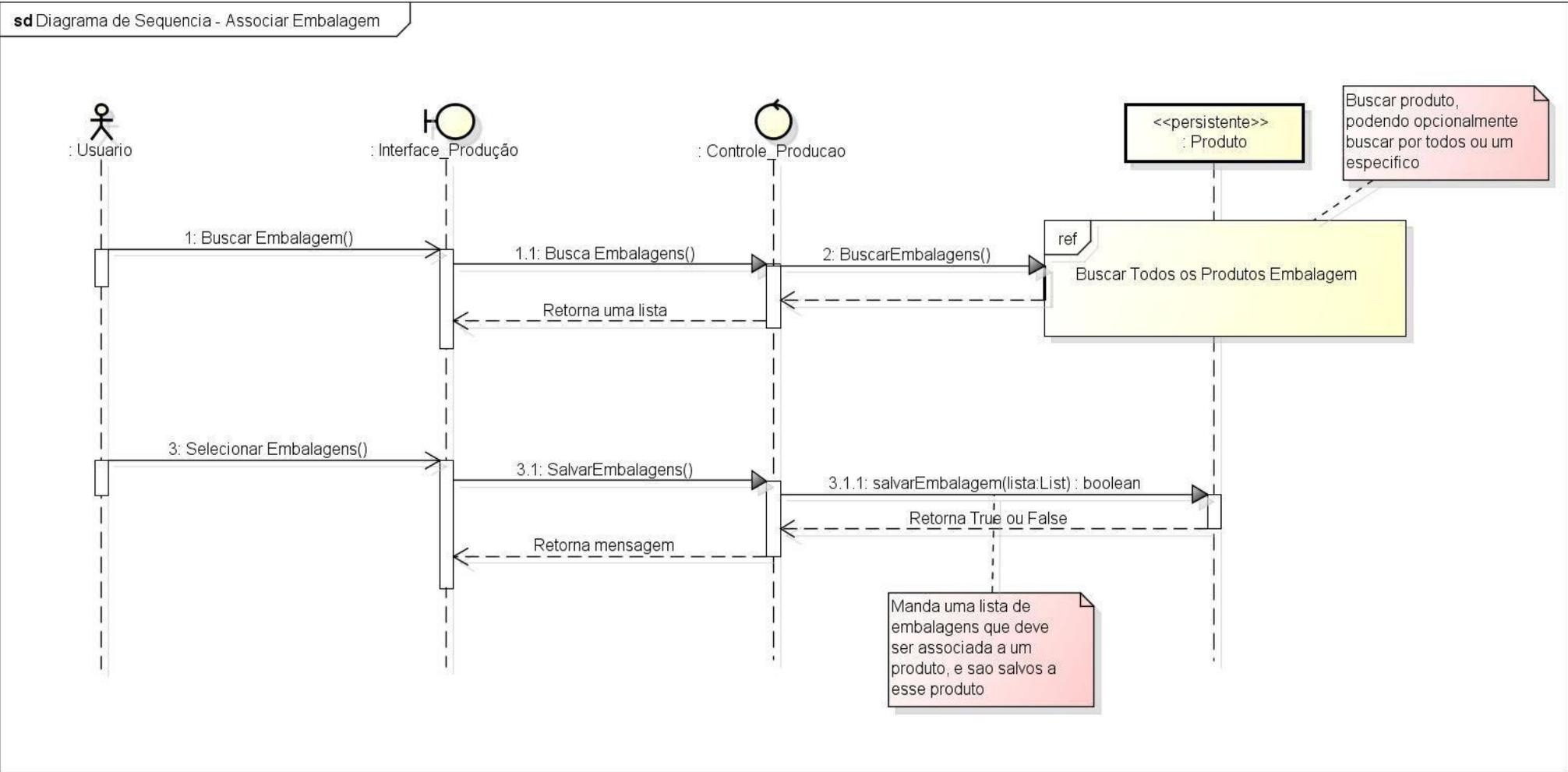
APÊNDICE Z: Diagrama de Sequência Buscar Capacidade



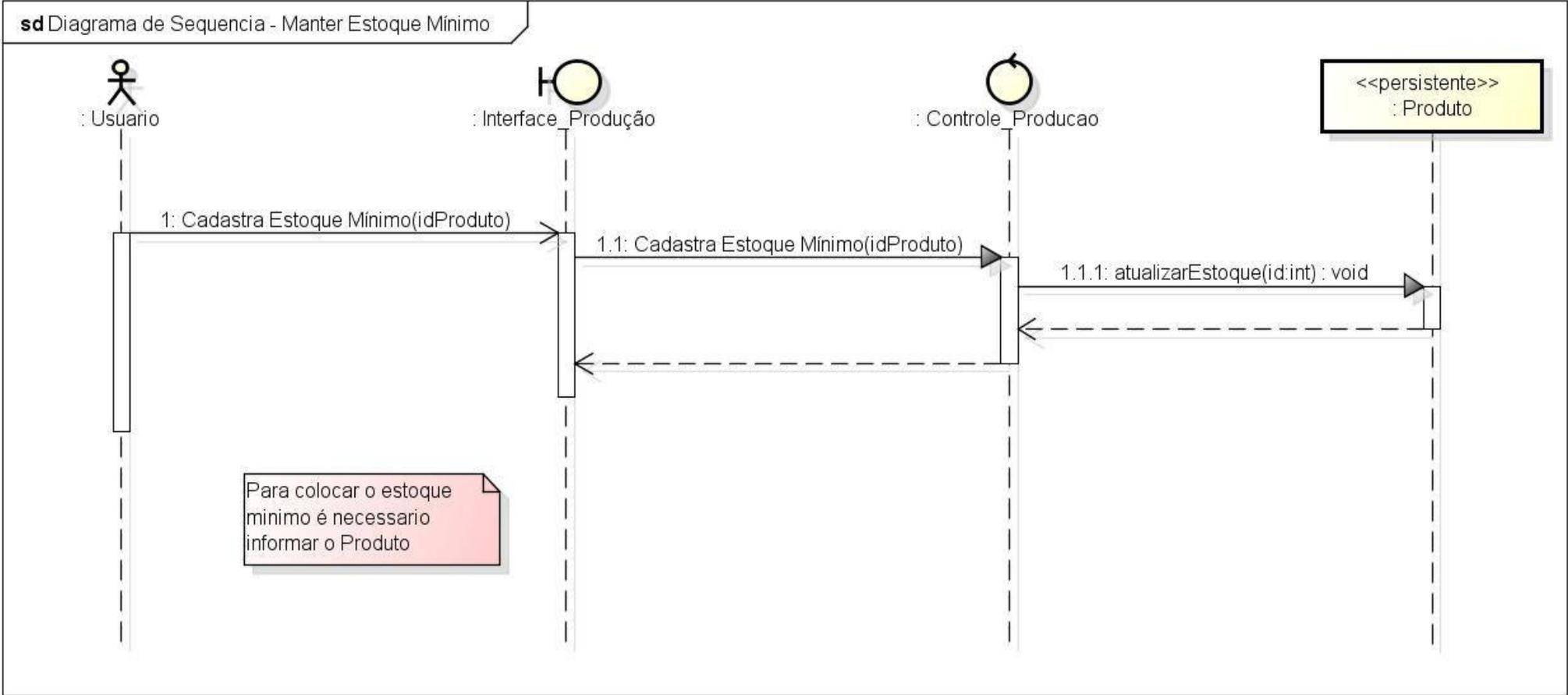
APÊNDICE AA: Diagrama de Sequência Cadastrar Produto



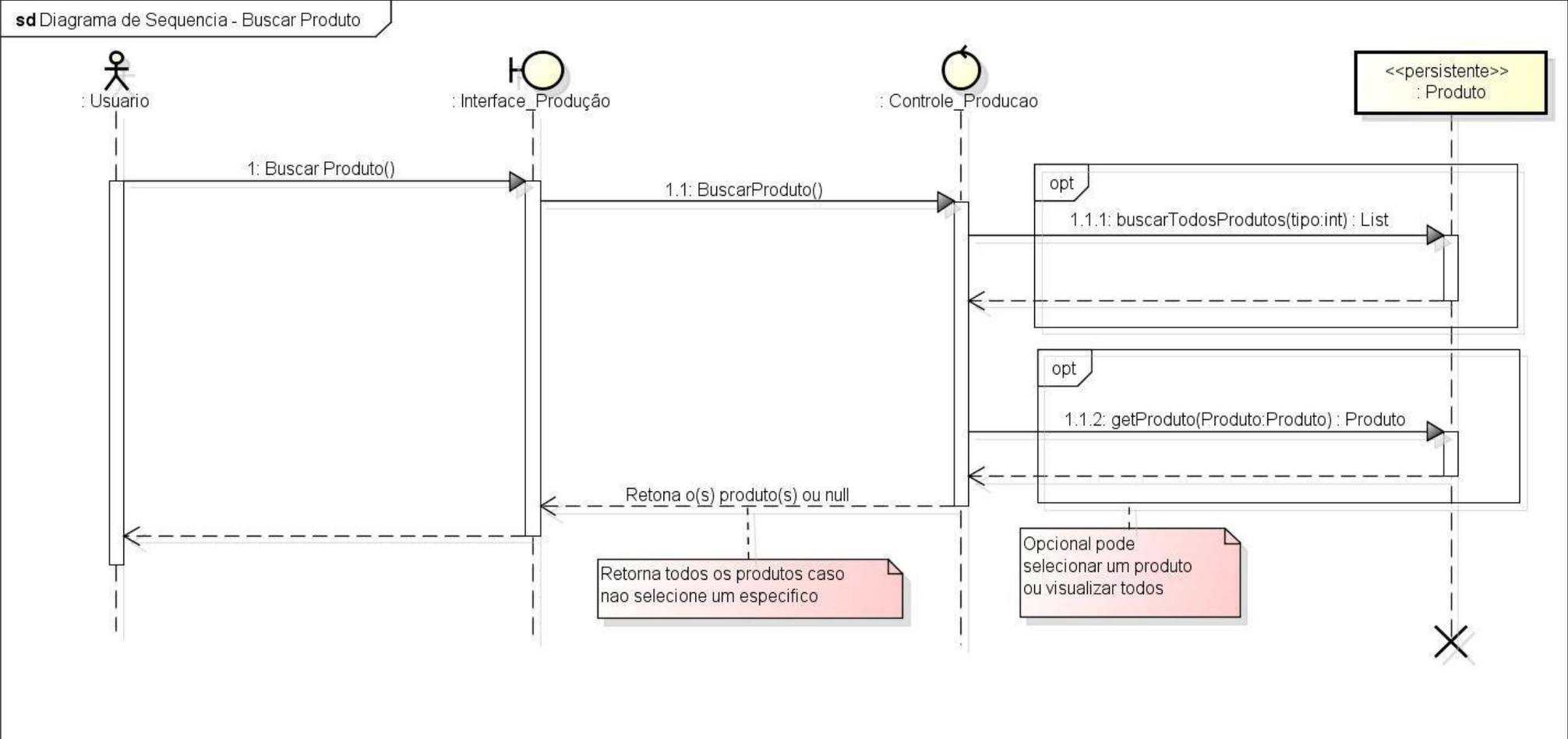
APÊNDICE BB: Diagrama de Sequência Associar Embalagem



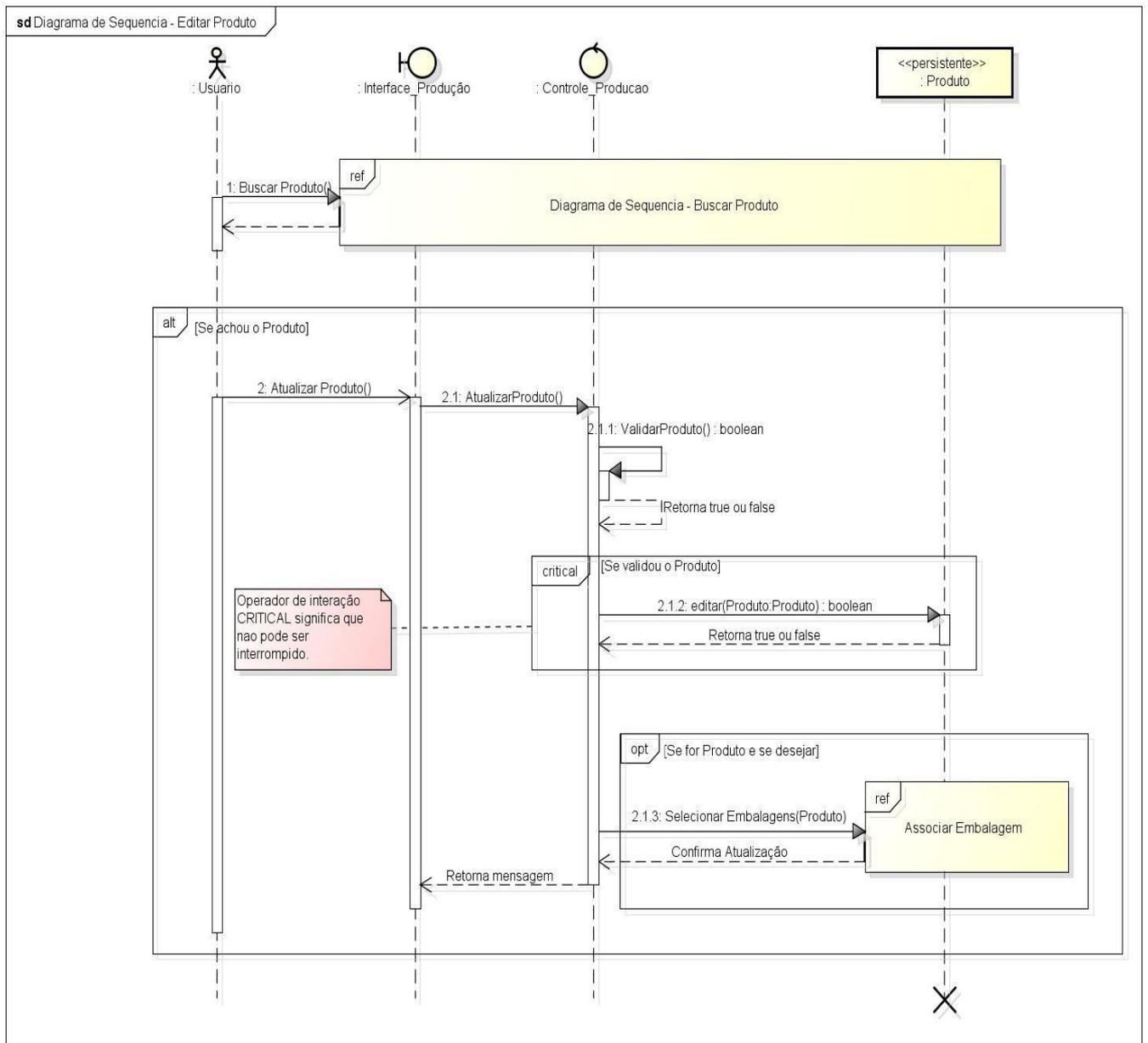
APÊNDICE CC: Diagrama de Sequência Manter Estoque Mínimo



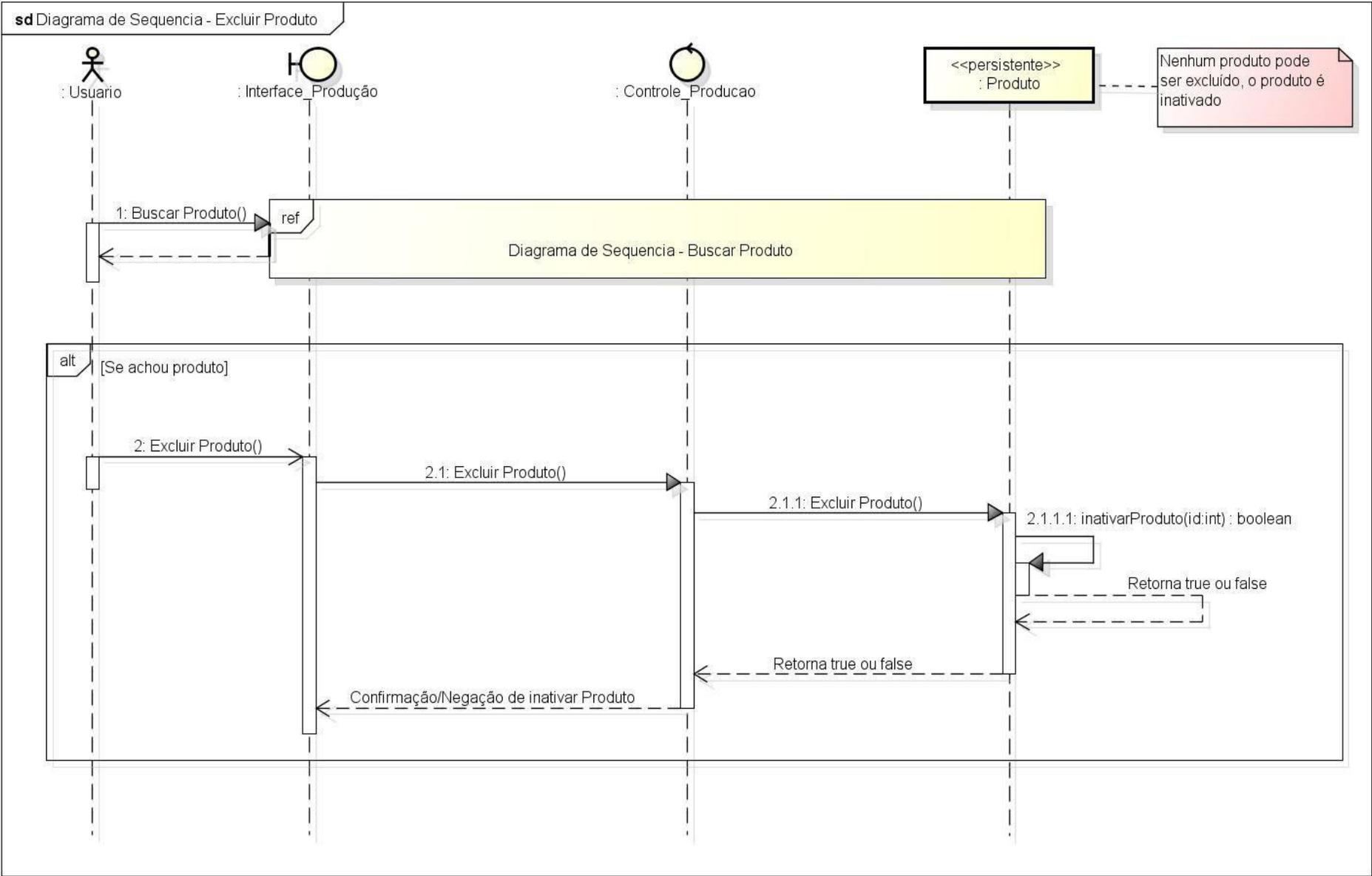
APÊNDICE DD: Diagrama de Sequência Buscar Produto



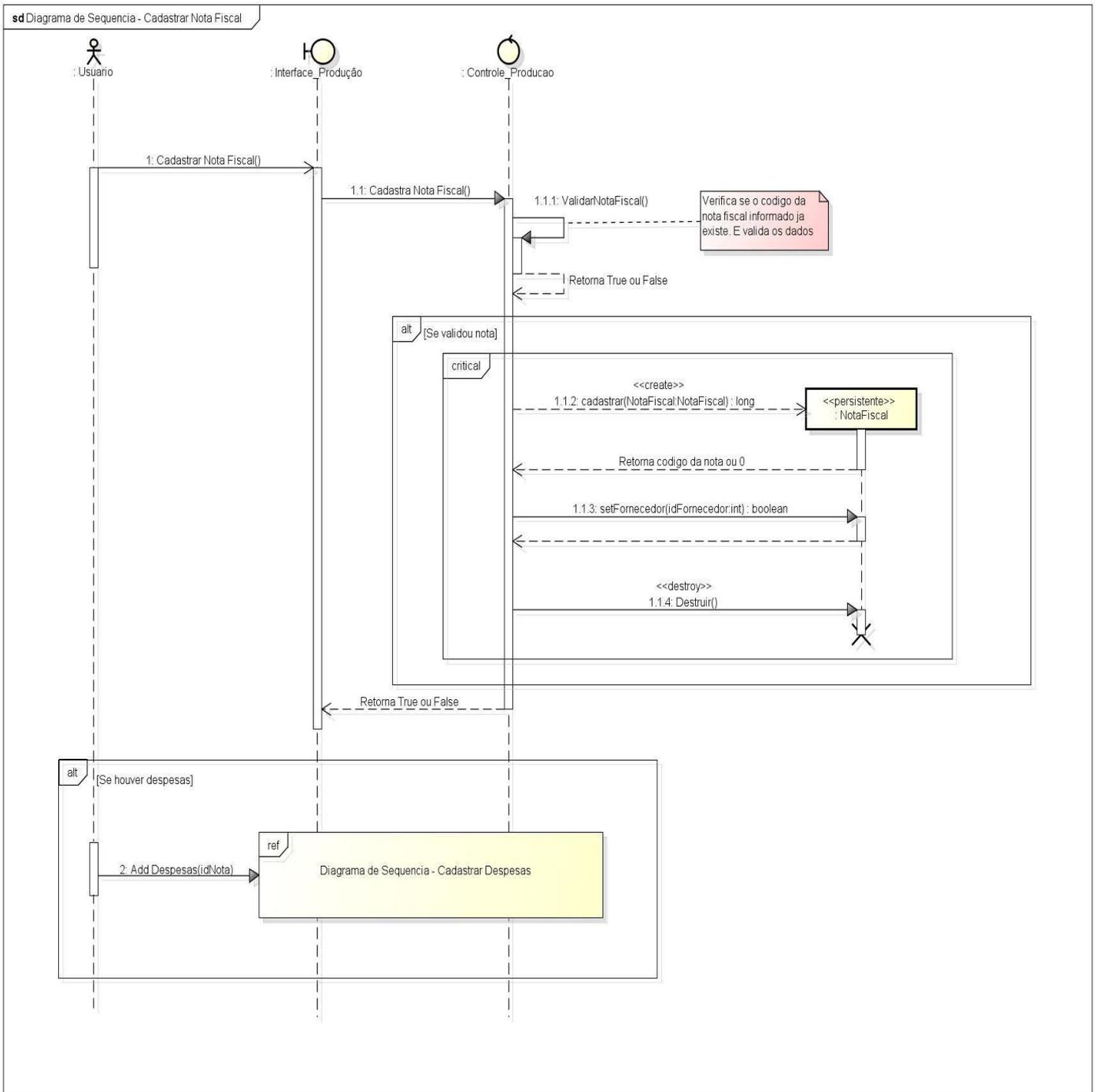
APÊNDICE EE: Diagrama de Sequência Editar Produto



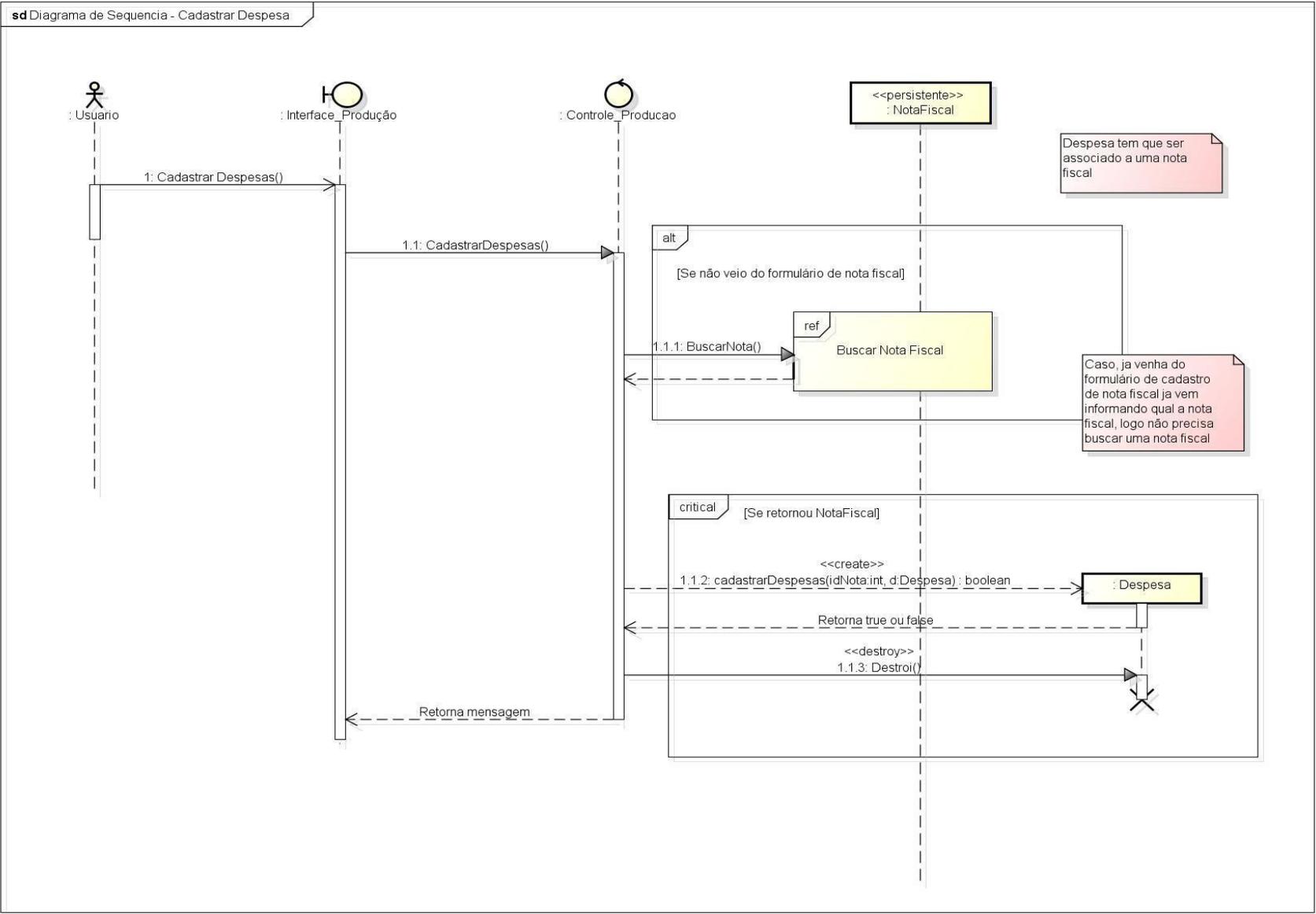
APÊNDICE FF: Diagrama de Sequência Excluir Produto



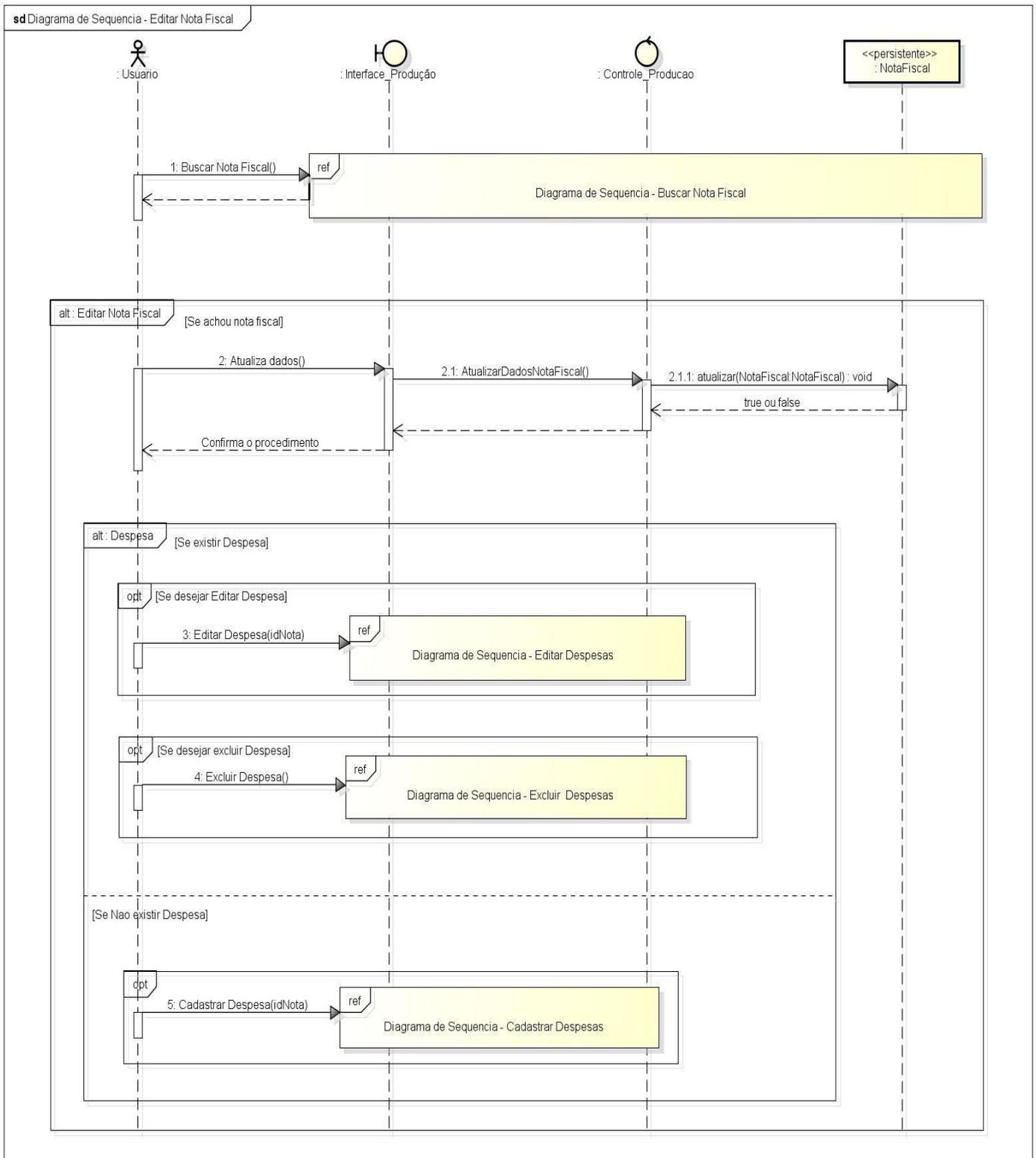
APÊNDICE GG: Diagrama de Sequência Cadastrar Nota Fiscal



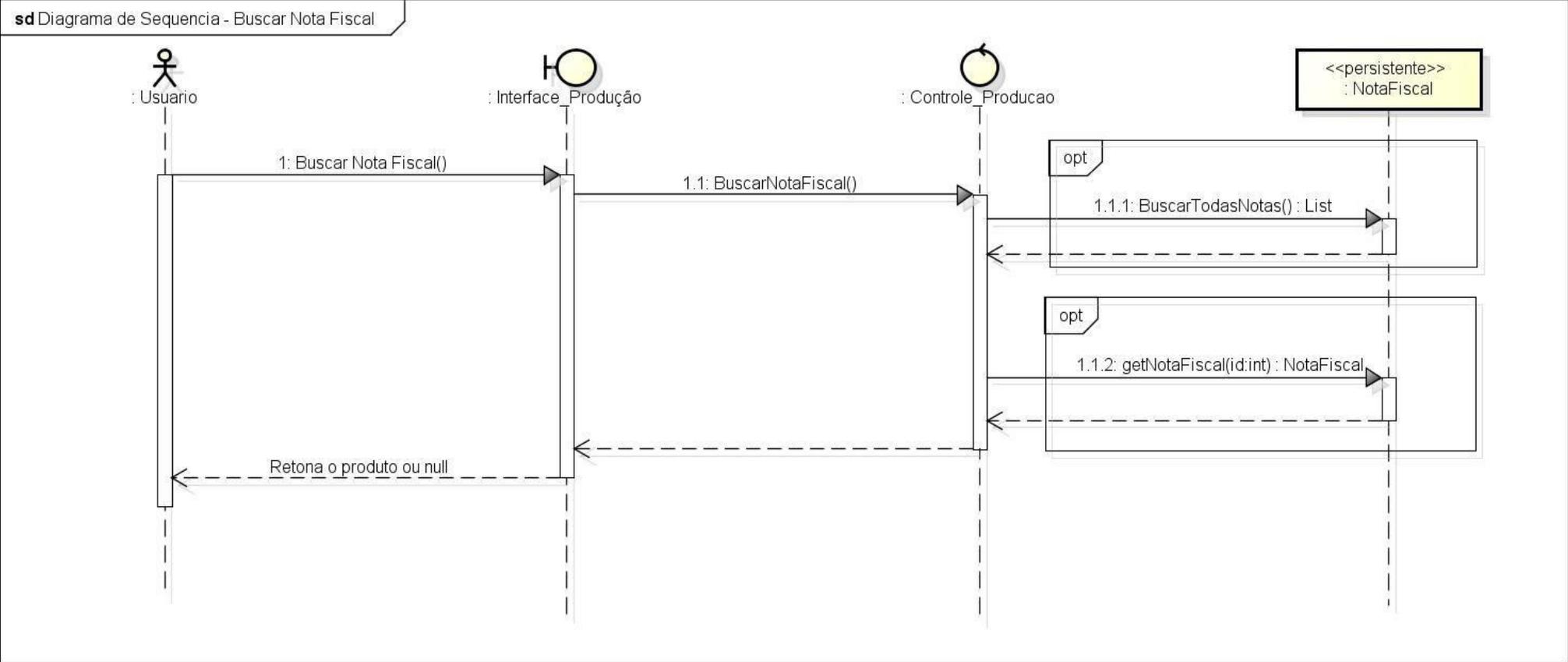
APÊNDICE HH: Diagrama de Sequência Cadastrar Despesa



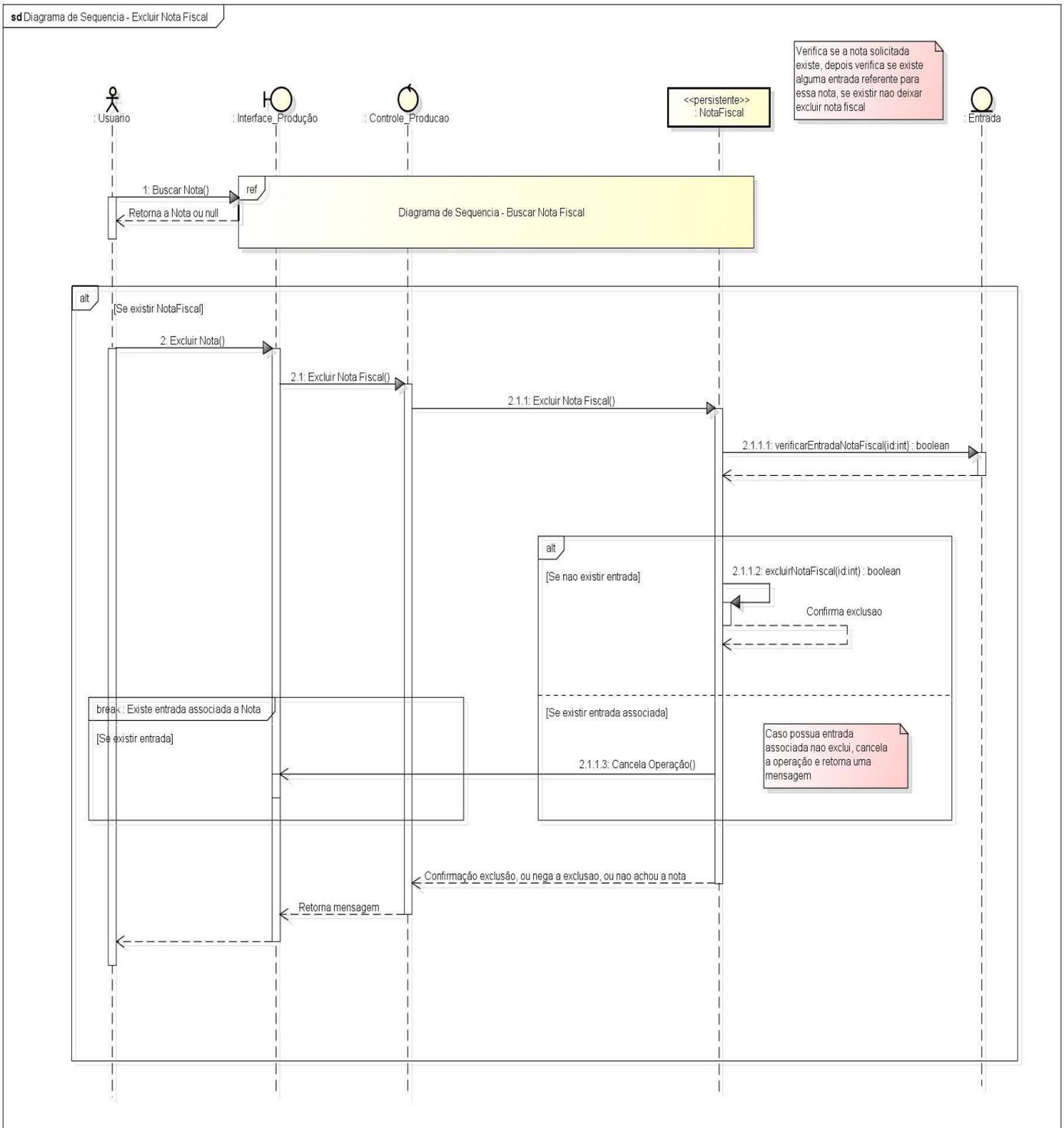
APÊNDICE II: Diagrama de Sequência Editar Nota Fiscal



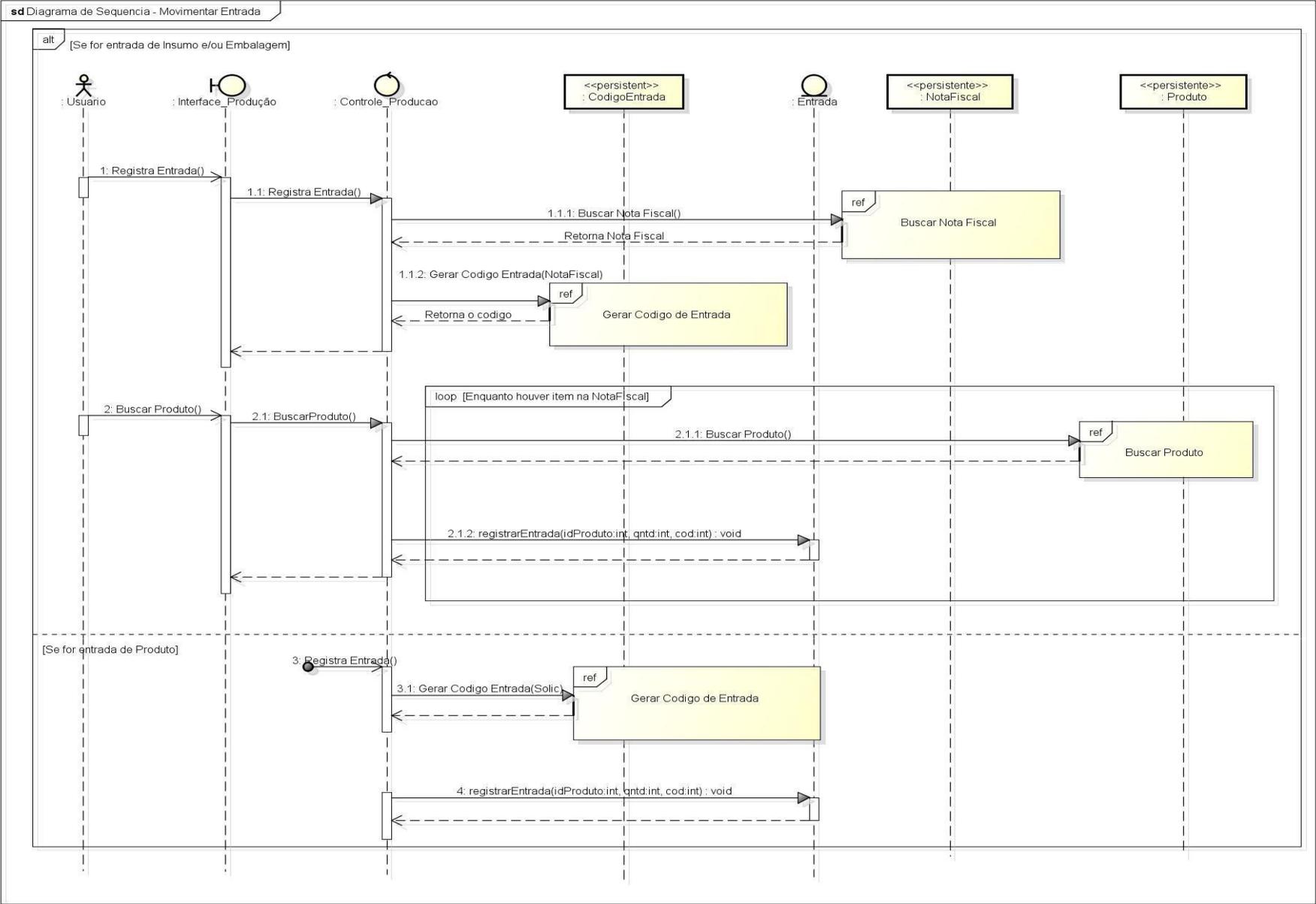
APÊNDICE JJ: Diagrama de Sequência Buscar Nota Fiscal



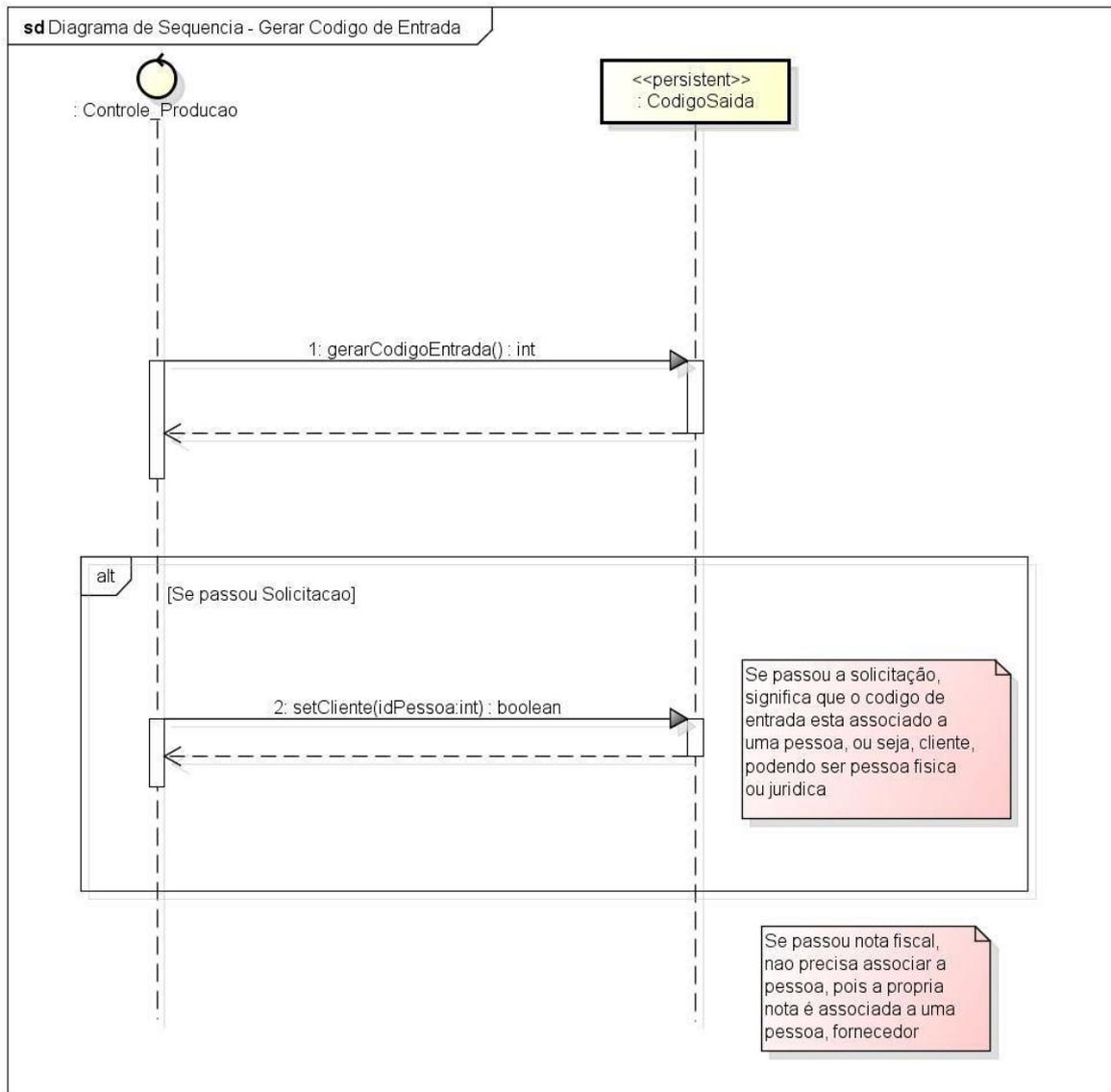
APÊNDICE KK: Diagrama de Sequência Excluir Nota Fiscal



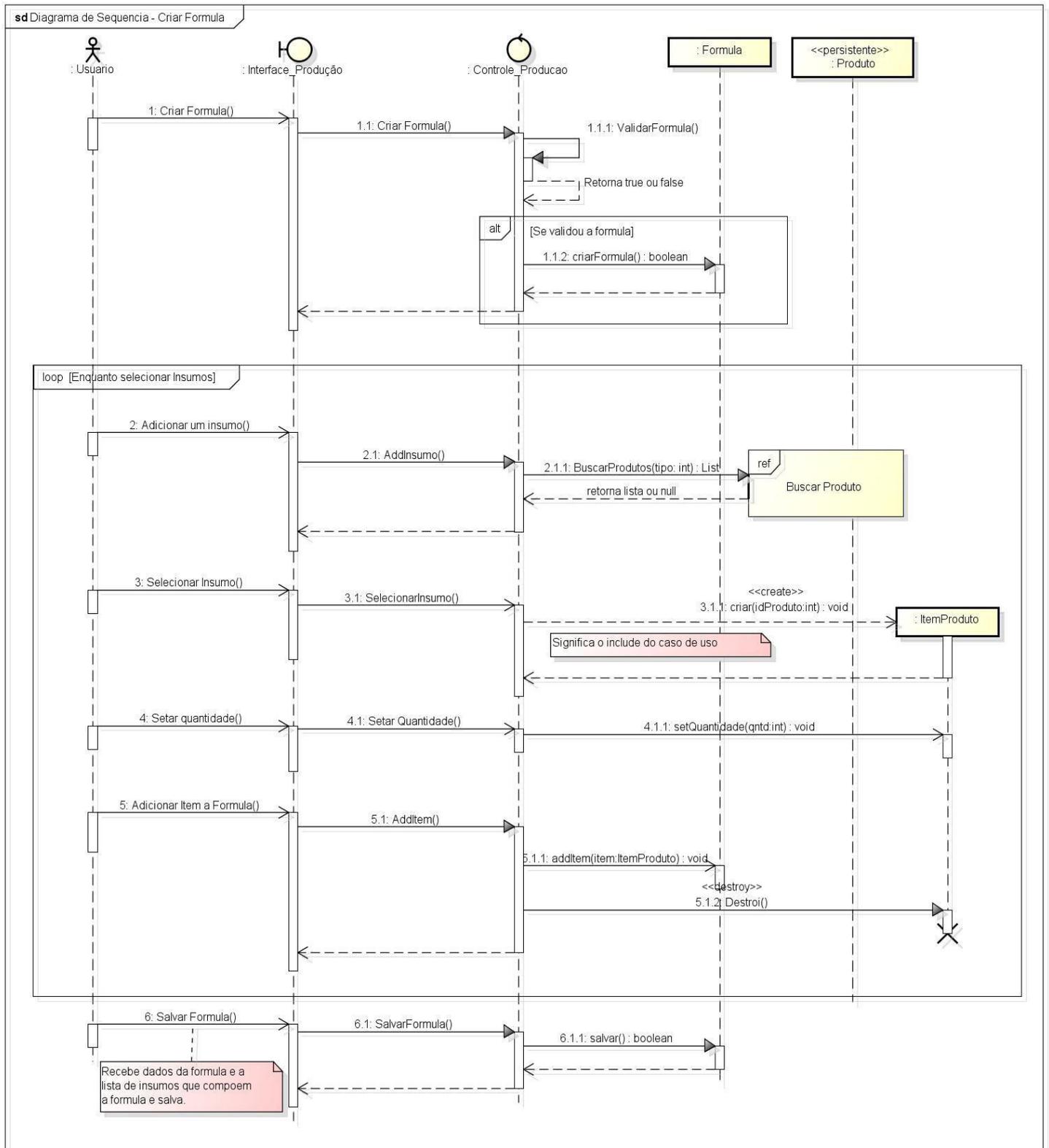
APÊNDICE LL: Diagrama de Sequência Movimentar Entrada



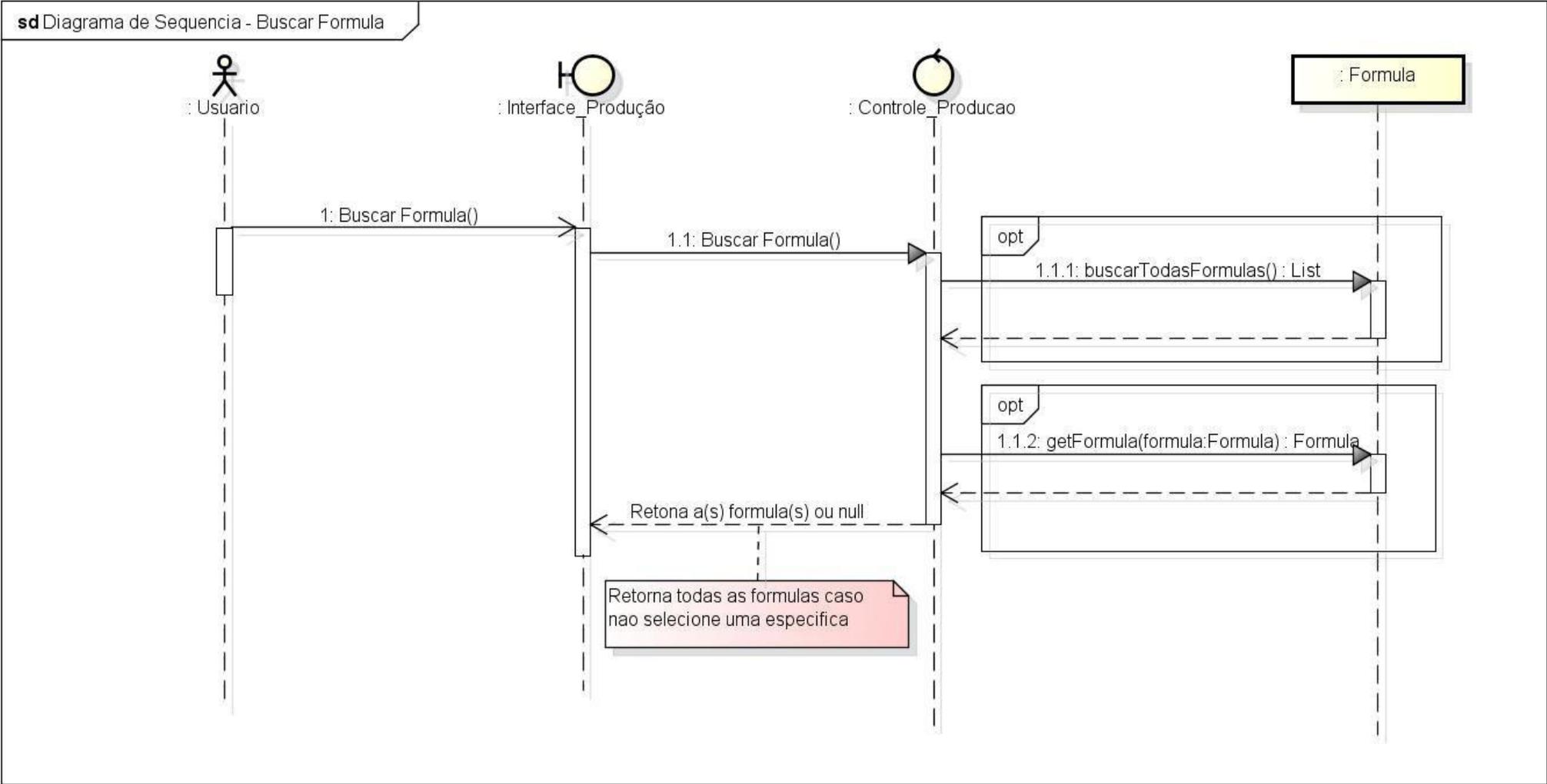
APÊNDICE MM: Diagrama de Sequência Gerar Código de Entrada



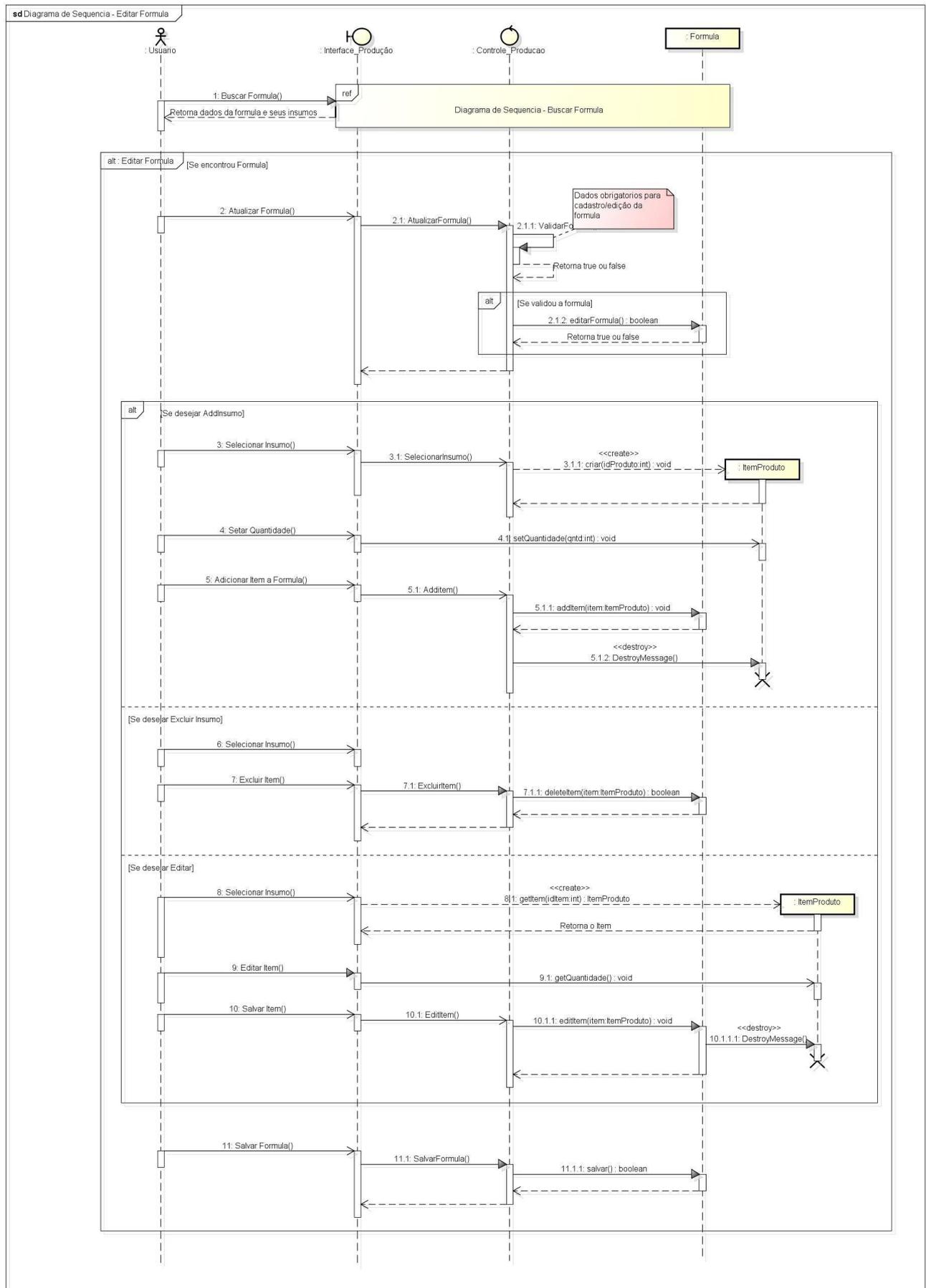
APÊNDICE NN: Diagrama de Sequência Criar Fórmula



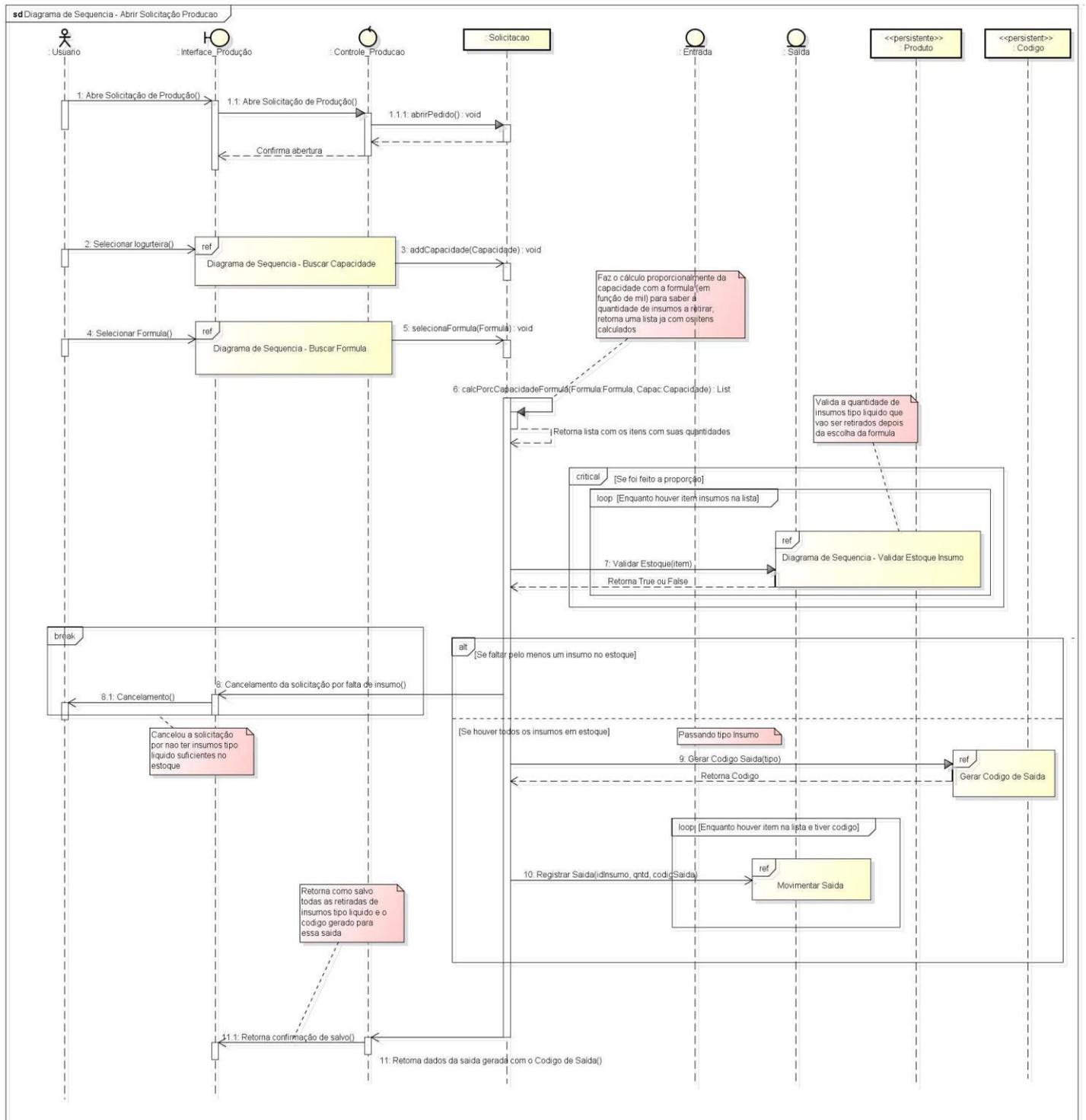
APÊNDICE OO: Diagrama de Sequência Buscar Fórmula



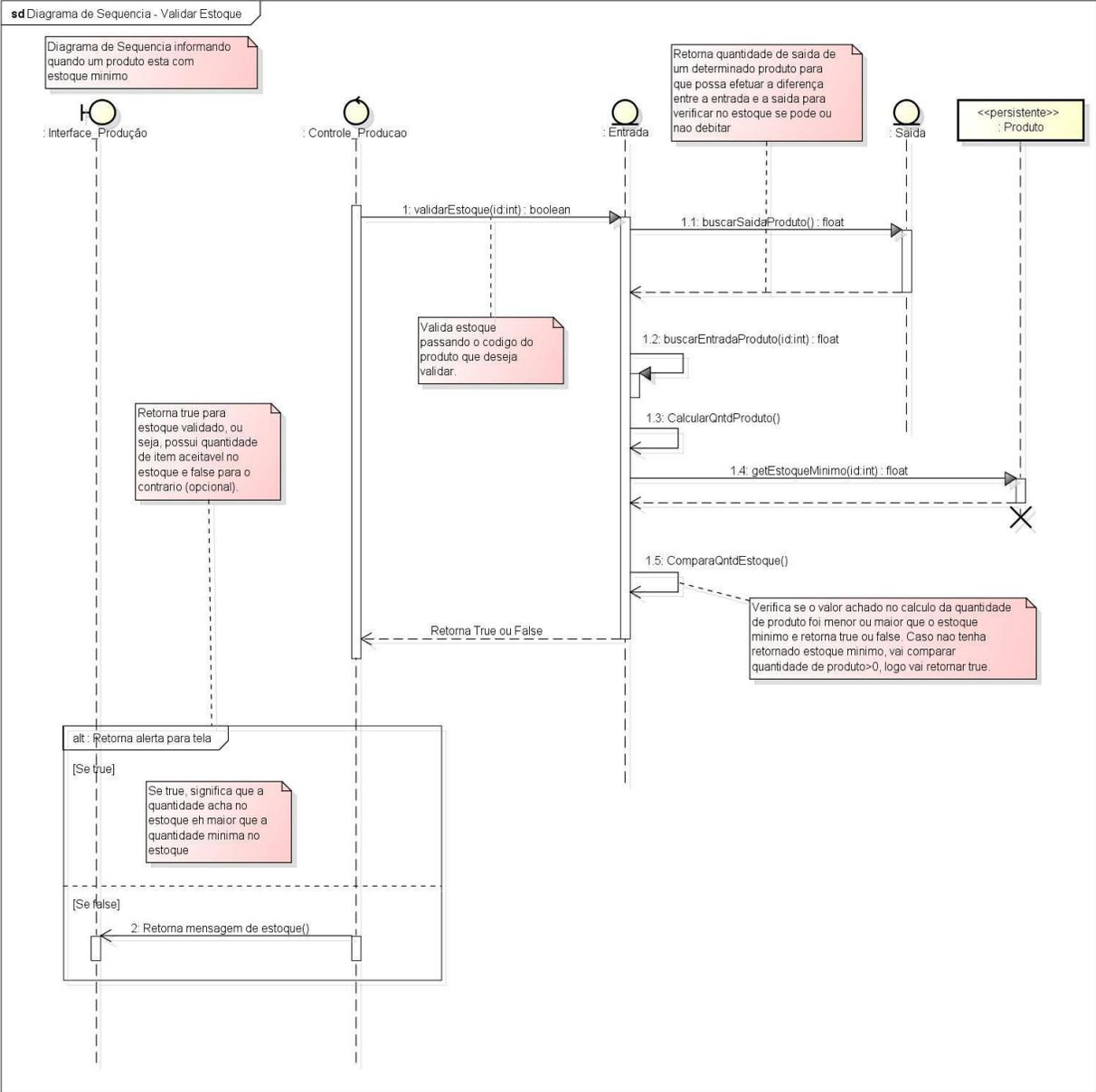
APÊNDICE PP: Diagrama de Sequência Editar Fórmula



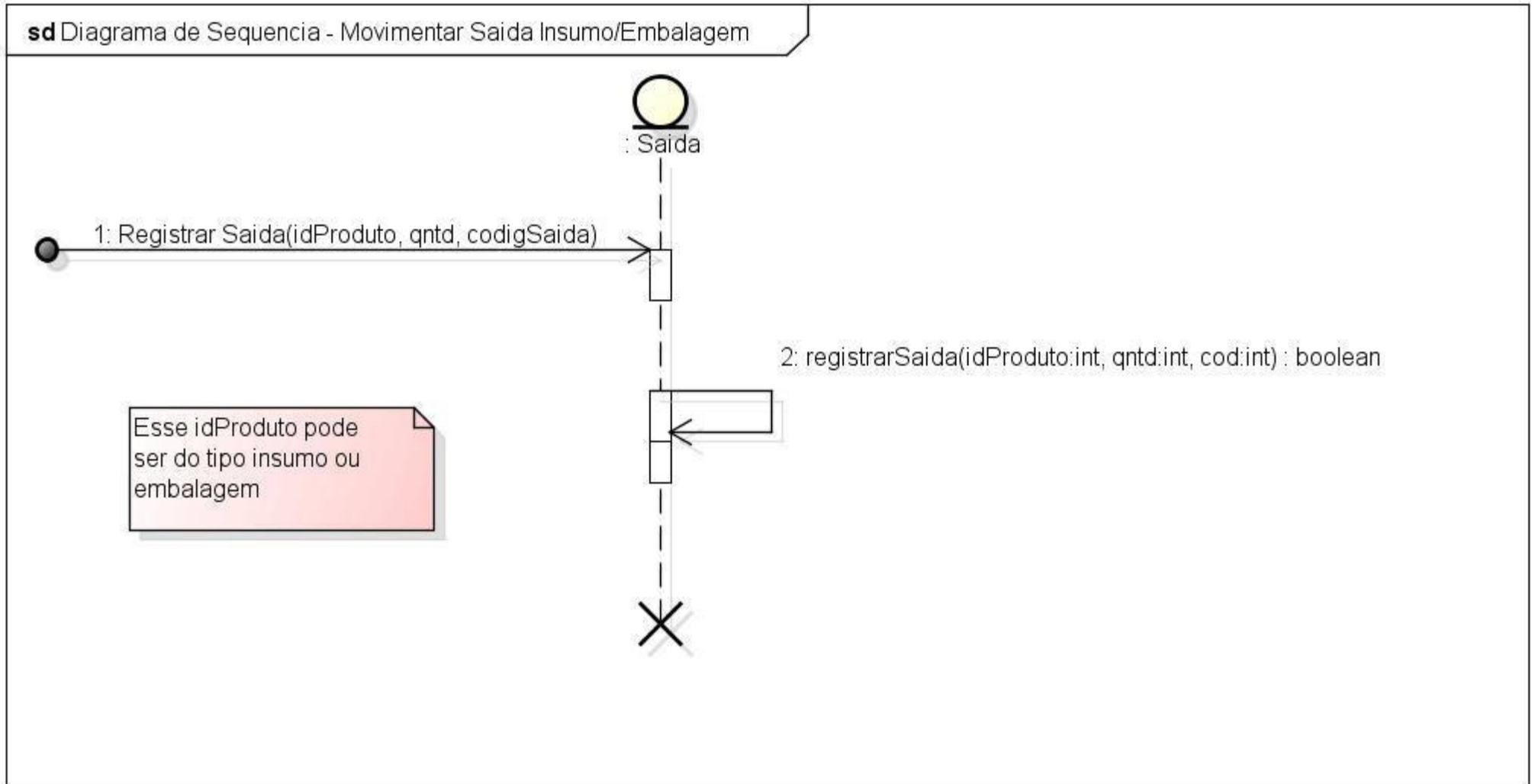
APÊNDICE QQ: Diagrama de Sequência Abrir Solicitação de Produção



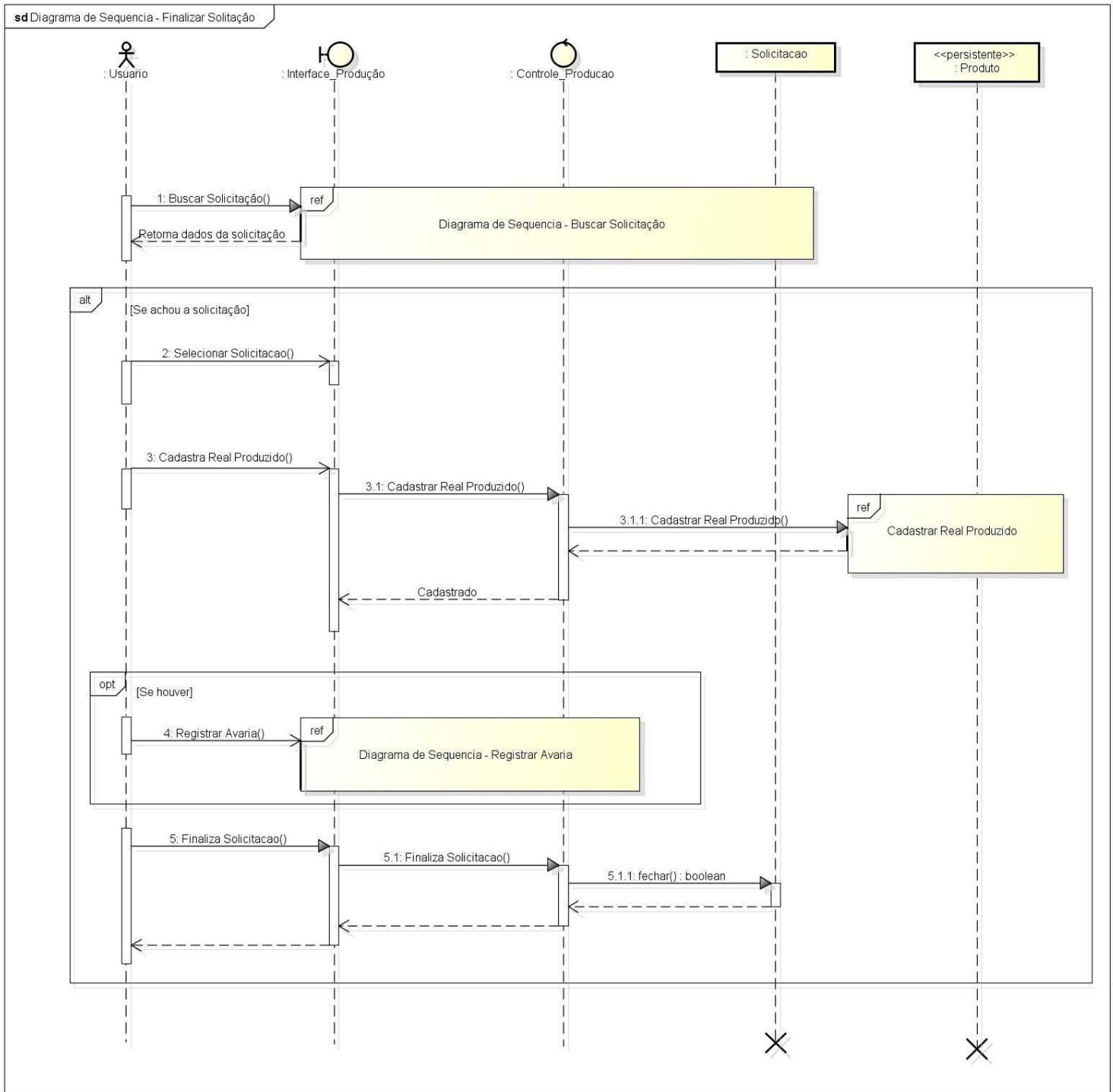
APÊNDICE RR: Diagrama de Sequência Validar Estoque



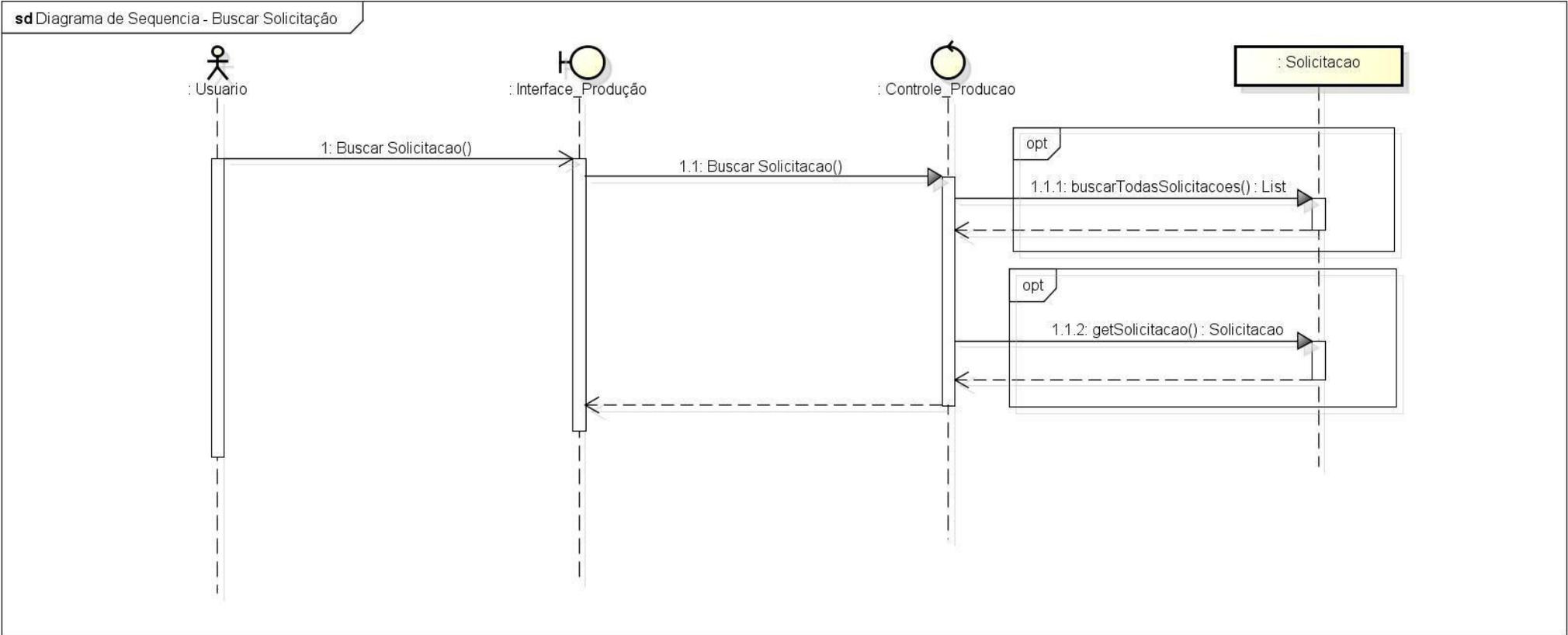
APÊNDICE SS: Diagrama de Sequência Movimentar Saída Insumo/Embalagem



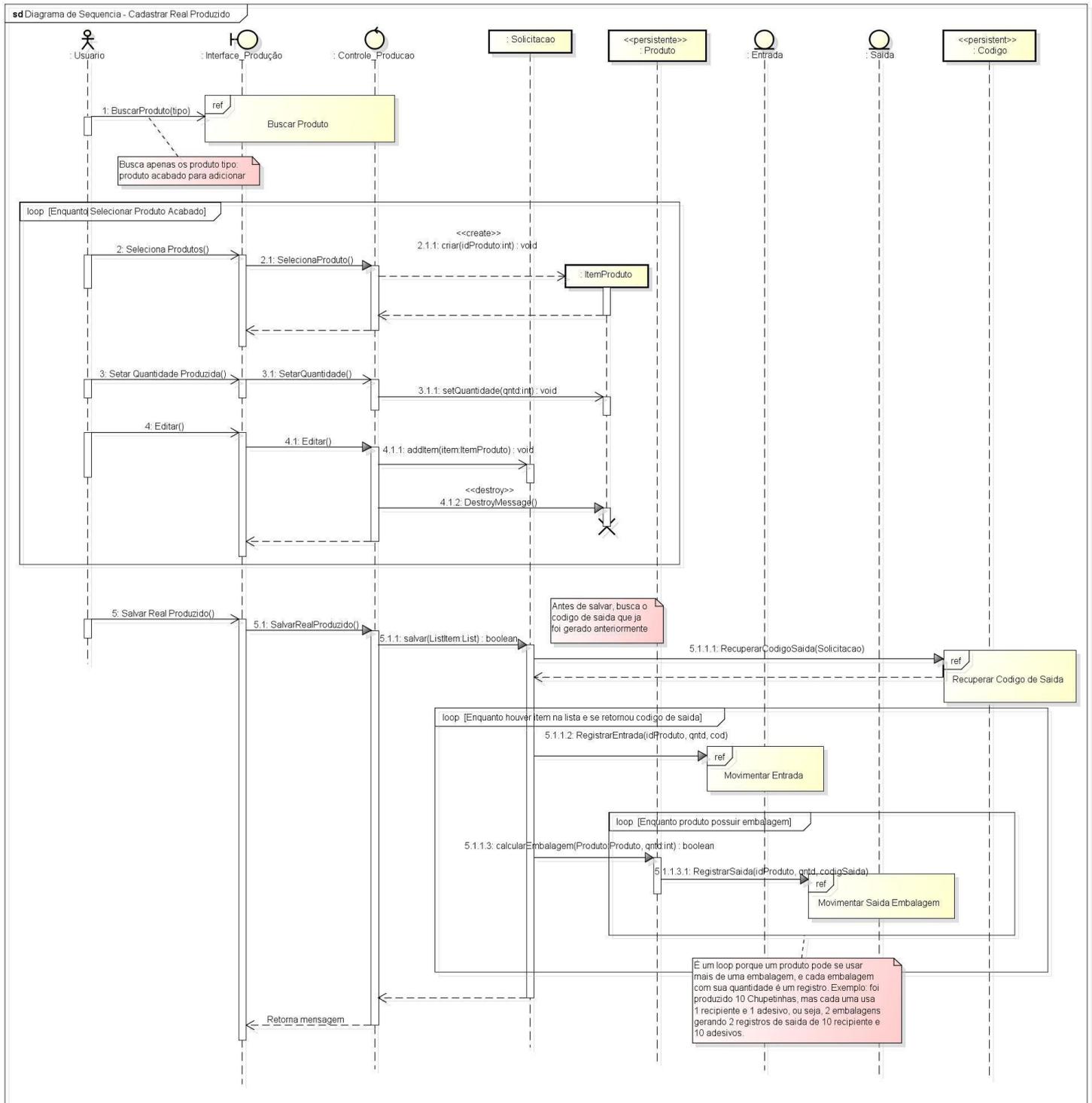
APÊNDICE TT: Diagrama de Sequência Finalizar Solicitação



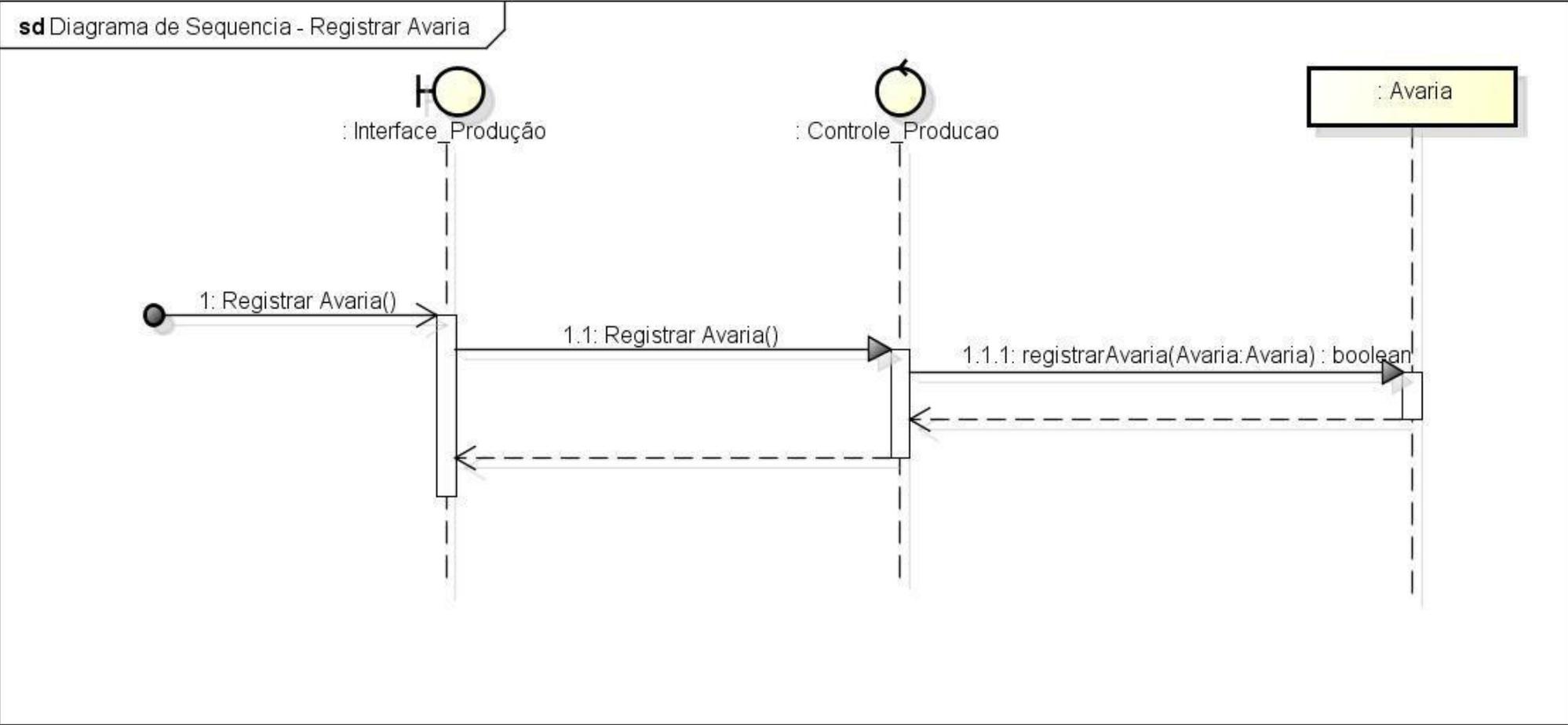
APÊNDICE UU: Diagrama de Sequência Buscar Solicitação



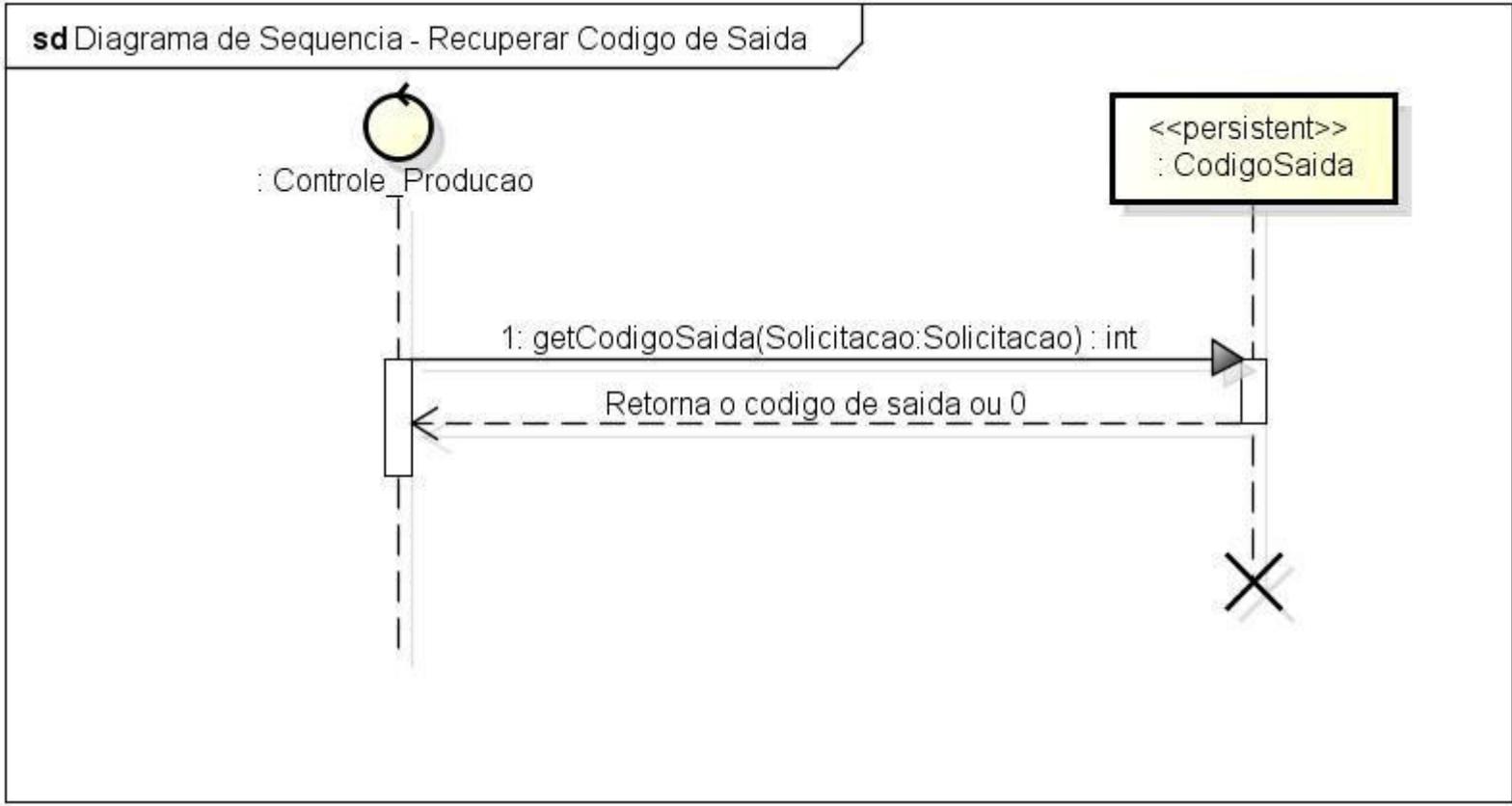
APÊNDICE VV: Diagrama de Sequência Cadastrar Real Produzido



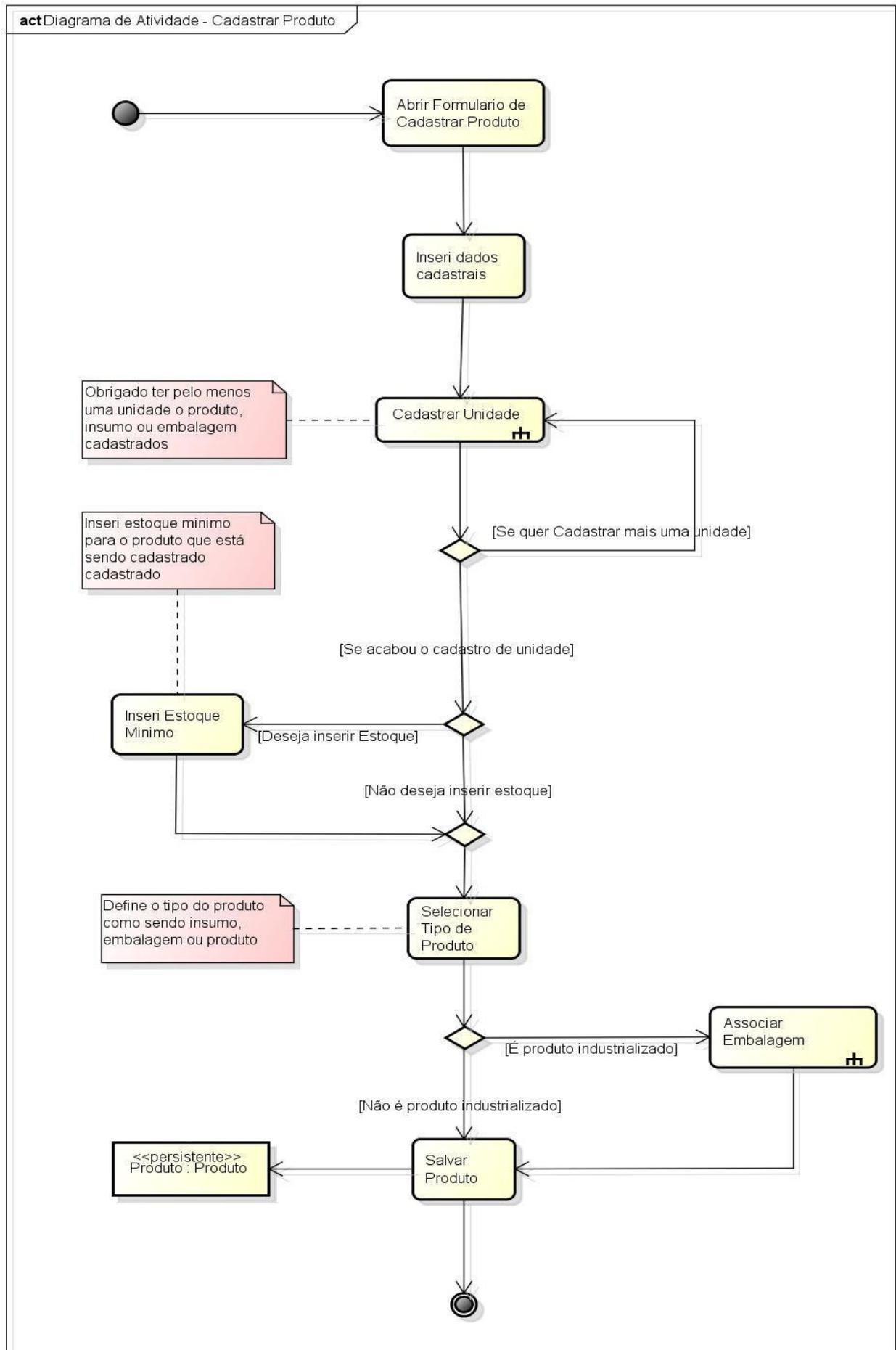
APÊNDICE WW: Diagrama de Sequência Registrar Avaria



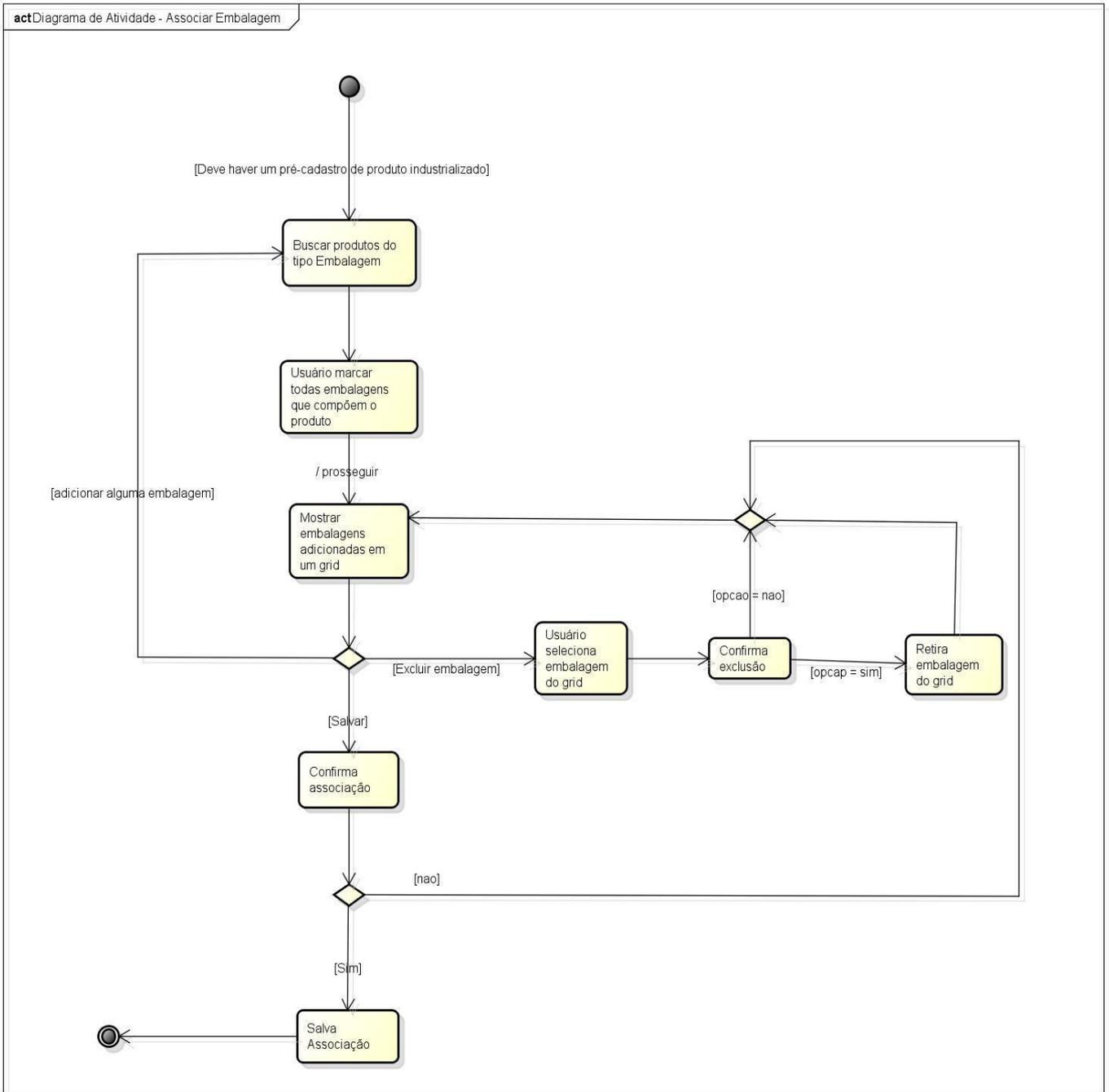
APÊNDICE XX: Diagrama de Sequência Recuperar Código de Saída



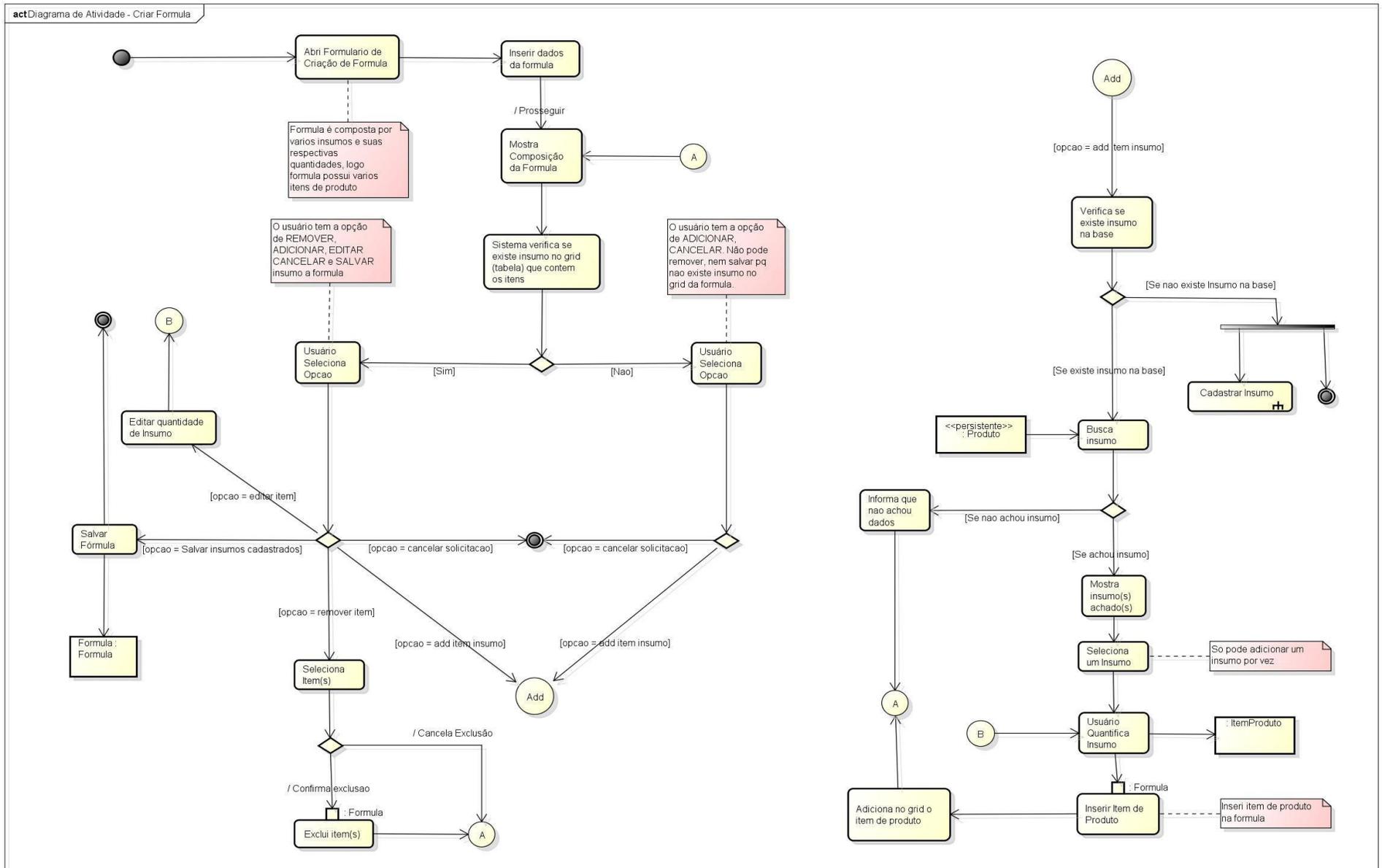
APÊNDICE YY: Diagrama de Atividade Cadastrar Produto



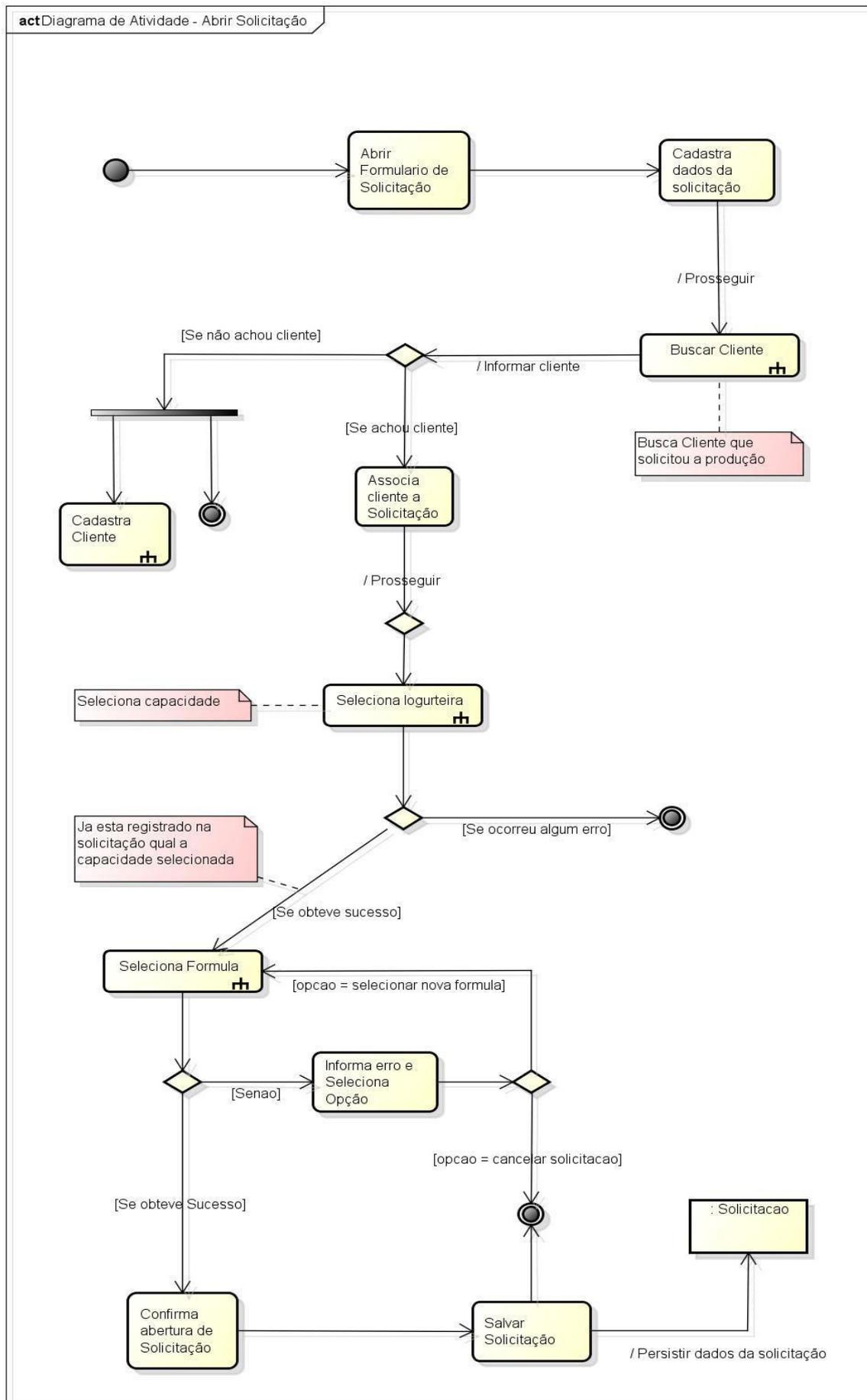
APÊNDICE ZZ: Diagrama de Atividade Associar Embalagem



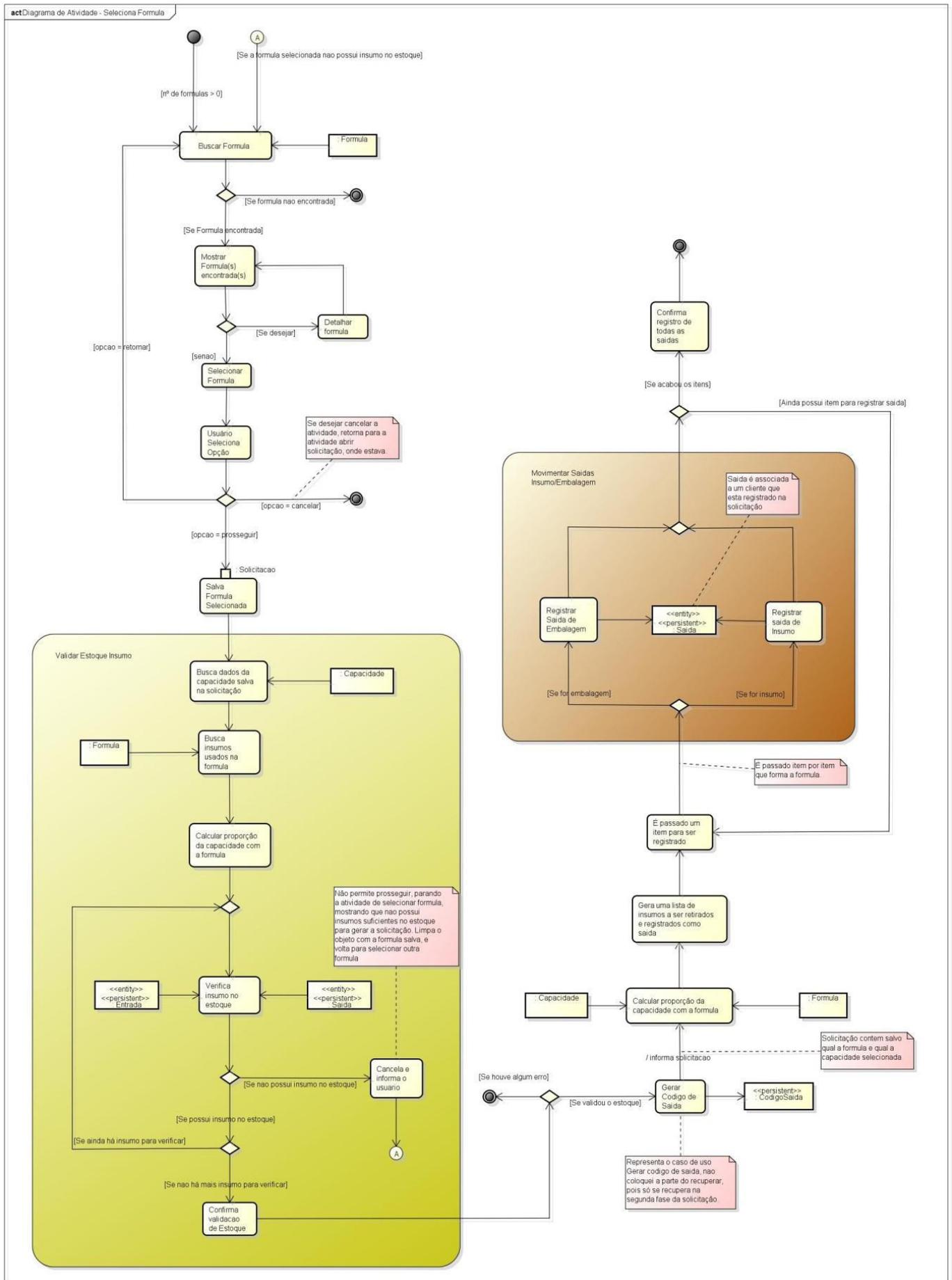
APÊNDICE AAA: Diagrama de Atividade Criar Fórmula



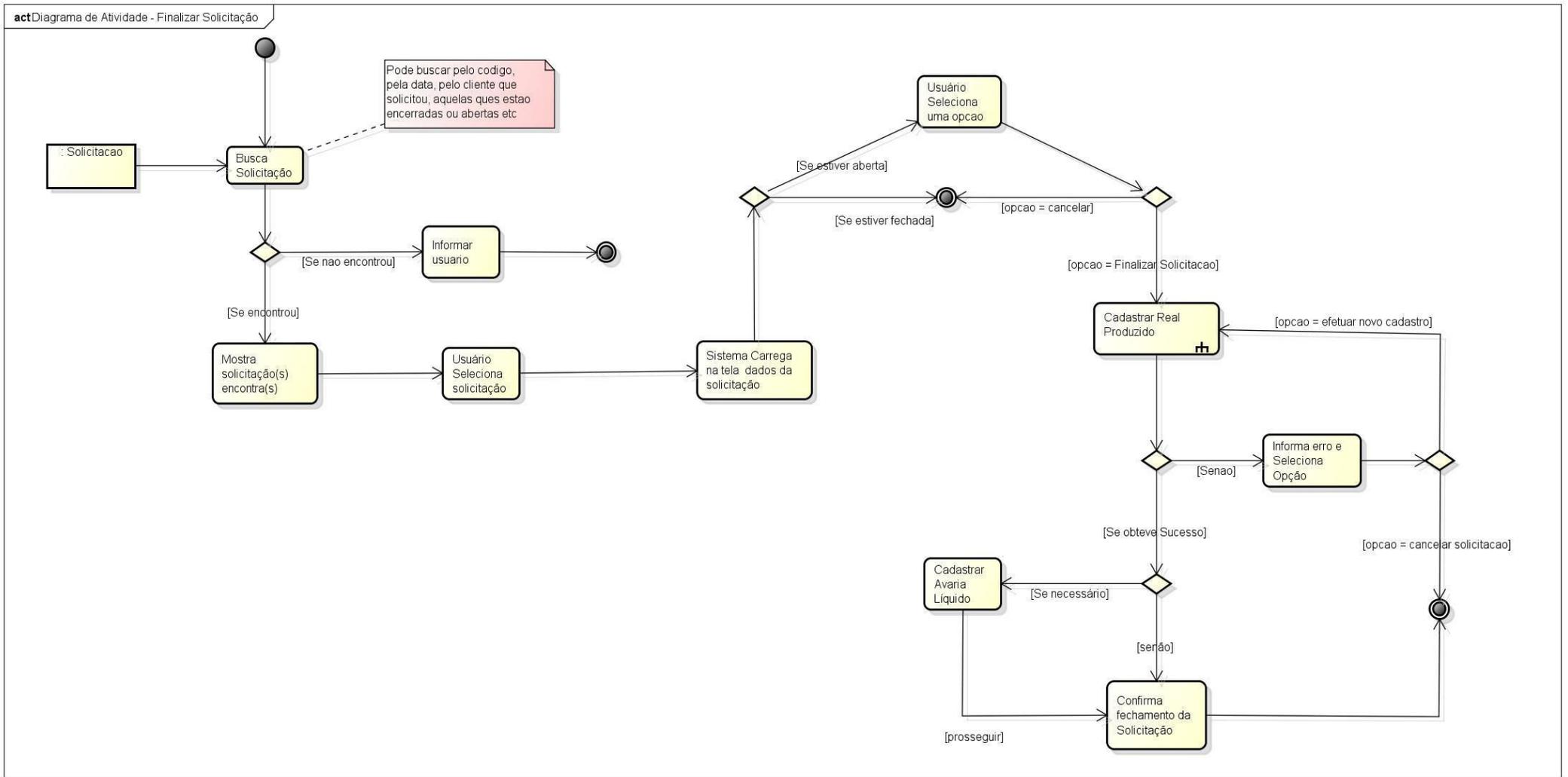
APÊNDICE BBB: Diagrama de Atividade Abrir Solicitação



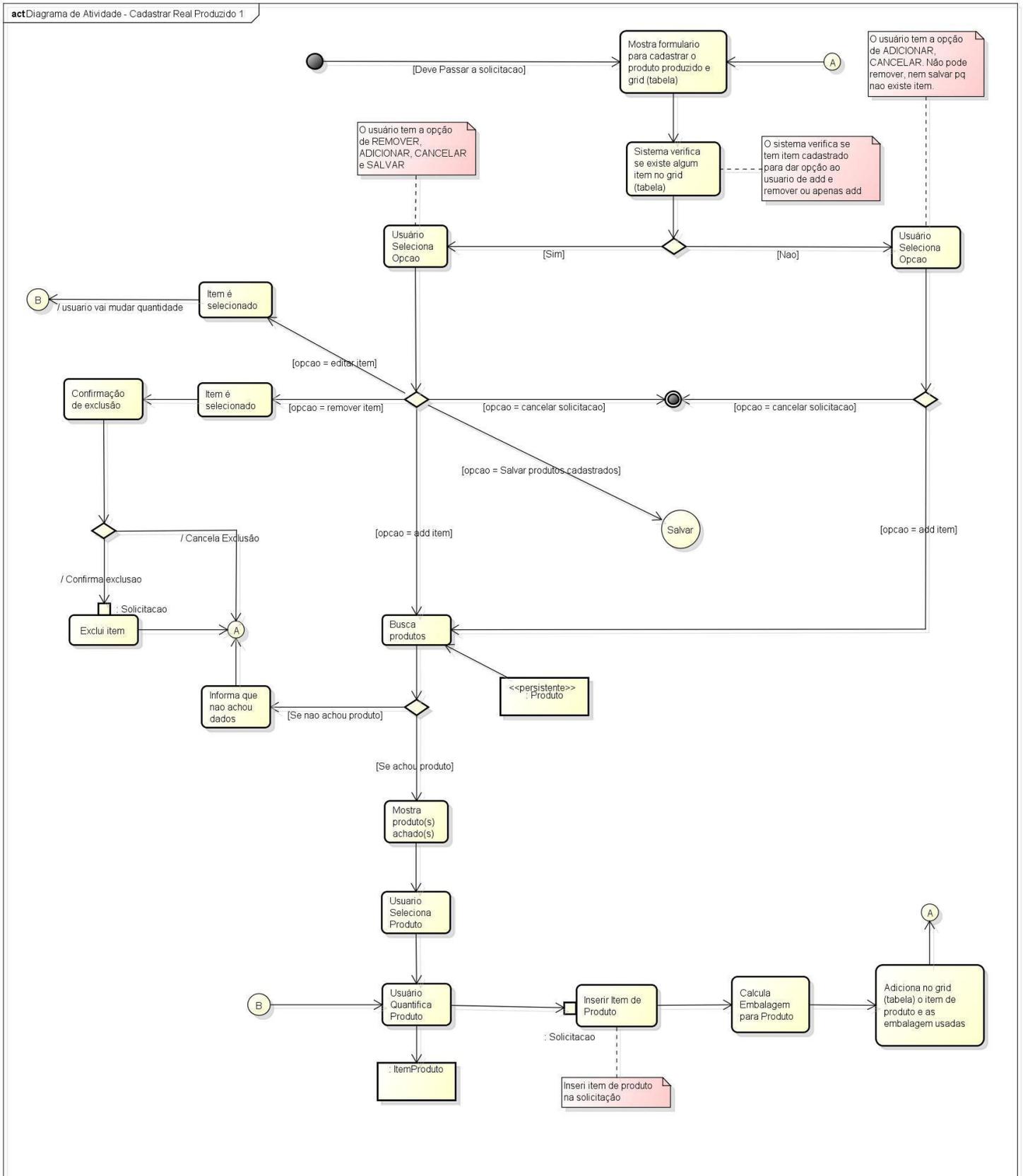
APÊNDICE CCC: Diagrama de Atividade Selecionar Fórmula

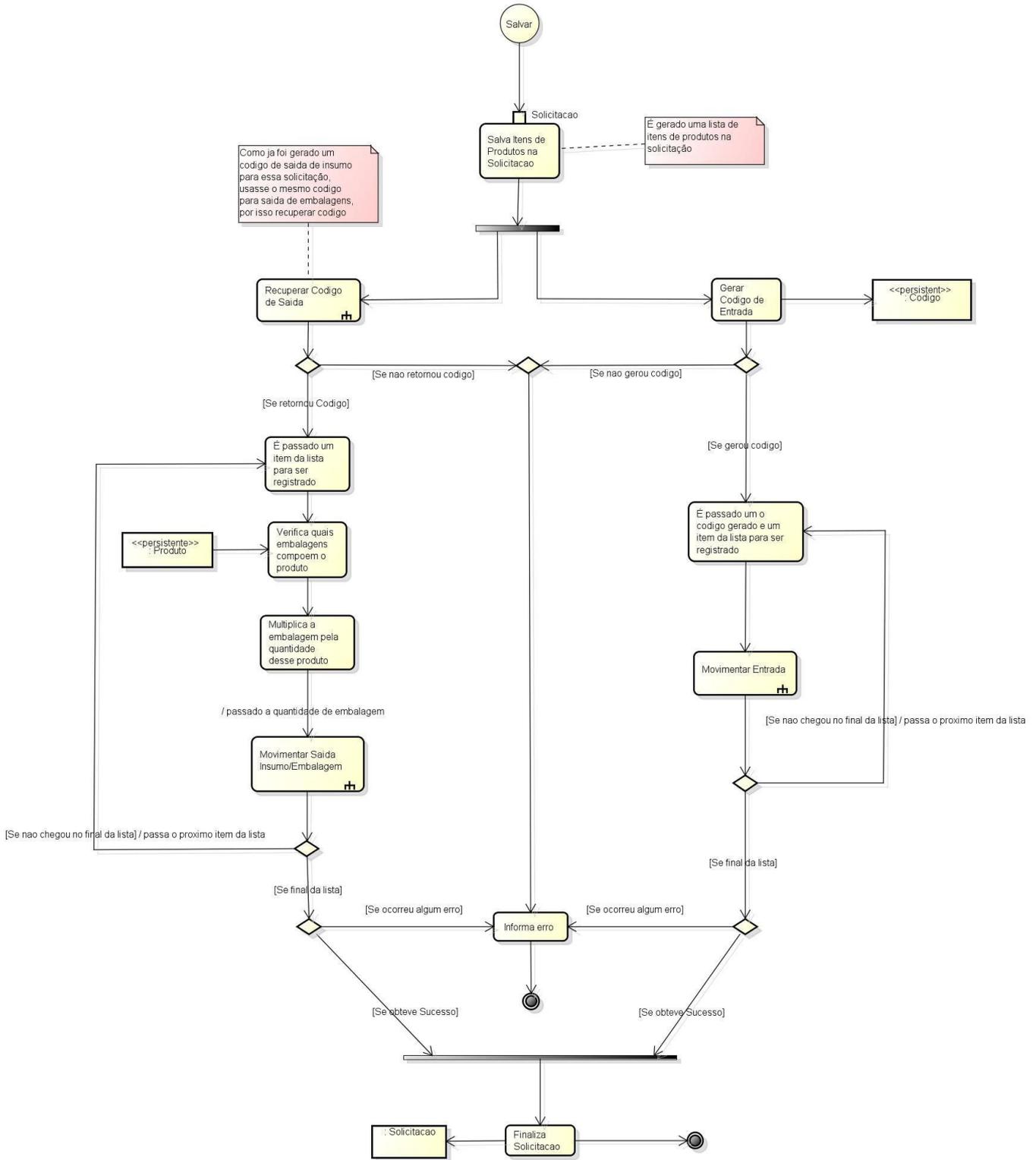


APÊNDICE DDD: Diagrama de Atividade Finalizar Solicitação

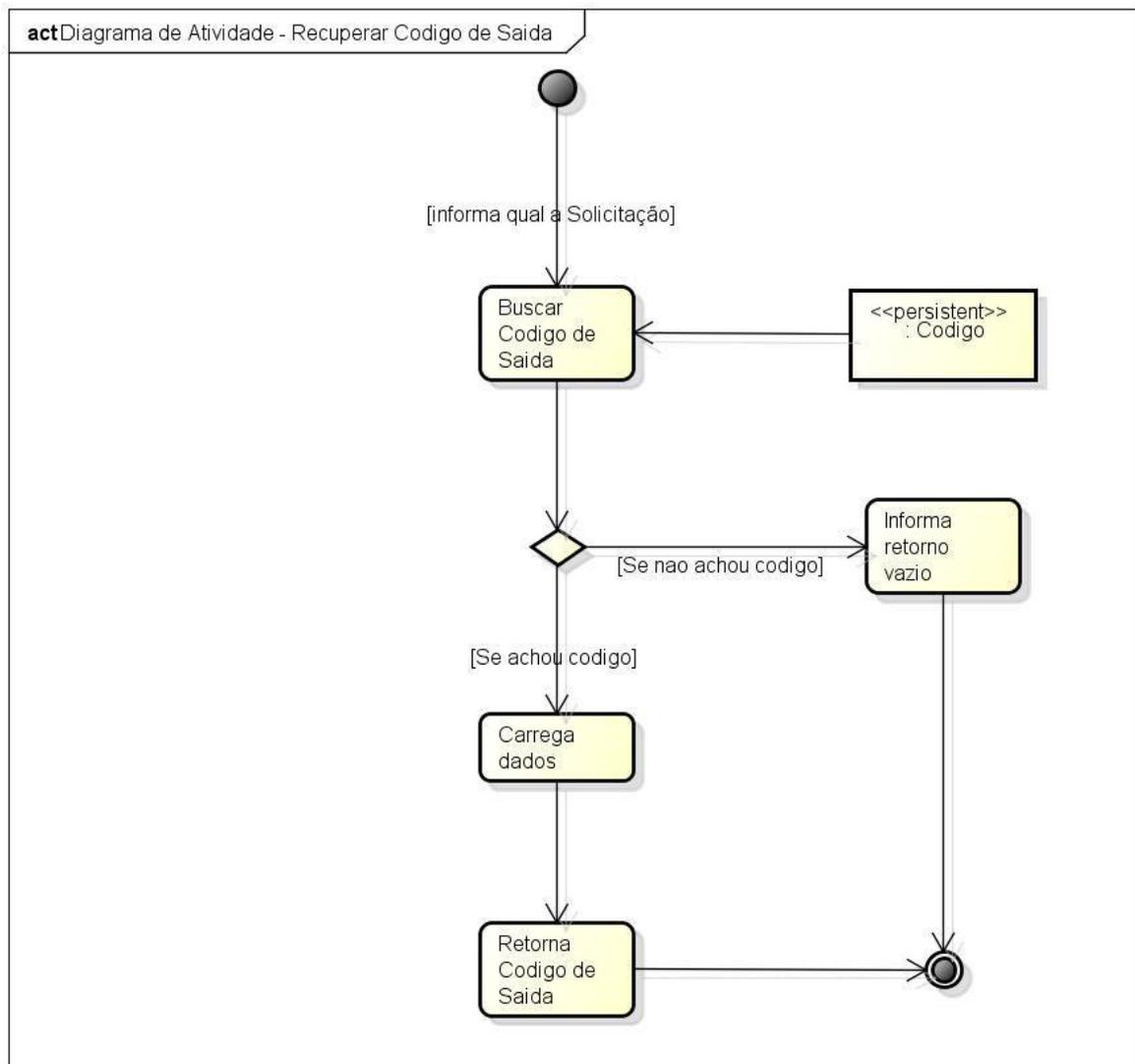


APÊNDICE EEE: Diagrama de Atividade Cadastrar Real Produzido





APÊNDICE FFF: Diagrama de Atividade Recuperar Código de Saída



APÊNDICE GGG: Diagrama de Atividade Movimentar Entrada

