

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SÃO PAULO
CÂMPUS BIRIGUI

ERICK SOUSA
HIGOR GRIGÓRIO
IGOR AUGUSTO
LEONARDO RENERES
RAUL DANTAS

PROJETO NATURART::

Quão bonito é seu jardim

BIRIGUI

2022

ERICK SOUSA
HIGOR GRIGÓRIO
IGOR AUGUSTO
LEONARDO RENERES
RAUL DANTAS

Projeto Naturart::
Quão bonito é seu jardim

Trabalho de Conclusão de Semestre
apresentado ao Instituto Federal de
Educação, Ciência e Tecnologia de
São Paulo, como requisito parcial para
conclusão do curso de Engenharia da
Computação.

Área de Concentração: Área de
Tecnologia

Orientador: Profa. Dra. Helen de Freitas
Santos

BIRIGUI

2022

1 ELICITAÇÃO DE REQUISITOS E ANÁLISE

Parte principal do texto, que contém a exposição ordenada e pormenorizada do assunto. Divide-se em seções e subseções, que variam em função da abordagem do tema e do método.

1.1 Requisitos do Usuário

Naturart é uma iniciativa que busca a criação de um produto que consiga aproximar as pessoas do meio ambiente, seja este o seu alqueire de terra ou o seu jardim no fundo de casa. Este projeto busca atingir este objetivo com a criação de um sistema e um dispositivo, os quais juntos, conseguem proporcionar ao usuário a experiência de possuir uma peça artística(imagem) à qual baseia-se nas informações que o seu jardim lhe proporciona: temperatura, ph da terra, parte por milhão e umidade relativa deste jardim, experiência que pode ser aberta para maiores extensões com base na quantidade de sensores utilizados.

Para que este objetivo seja concluído são necessárias as seguintes ações dos usuários:

- Poder comprar um ou mais sensores.
- Com o(s) sensor(es) poder cadastrá-lo(s) em um aplicativo mobile, para controlá-lo(s).
- Poder pedir para gerar a imagem com base nos dados coletados pelo(s) sensor(es), podendo especificar quais dados e em qual período de tempo.
- Poder fazer o download das imagens as quais o cliente fez o pedido.

1.2 Requisitos do Sistema

Fazer uma pequena definição de requisitos do sistema para introduzir a seção (informar a fonte). Assim, esta seção apresenta os requisitos identificados nesse trabalho.

Fazer uma definição de requisitos do sistema. Desta forma, essa seção apresenta os requisitos identificados para este sistema, levando em consideração o sistema web, o sistema mobile e o sistema do hardware.

1.2.1 Requisitos Funcionais

Os requisitos funcionais são os requisitos que estão diretamente relacionados às funcionalidades do sistema. E os requisitos não funcionais são os requisitos que não estão diretamente relacionados às funcionalidades do sistema.

Tabela 1 – Requisitos Funcionais do Sistema Web

Código	Descrição	Ator
RFW01	<p>O sistema deve permitir o cadastro, a alteração e a remoção de um estado. Tendo como atributos:</p> <ul style="list-style-type: none"> • nome (String) • Sigla (String) <p>Todos os atributos são obrigatórios, e não pode registrar dois Estados com a mesma sigla e com o mesmo nome</p>	Sistema
RFW02	<p>O sistema deve permitir o cadastro, a alteração e a remoção de uma cidade. Tendo como atributos:</p> <ul style="list-style-type: none"> • Nome (String) • Estado (String - Cadastrado previamente); <p>Todos os atributos são obrigatórios, e não pode registrar duas cidades com o mesmo nome associado no mesmo Estado.</p>	Sistema
RFW03	<p>O sistema deve permitir O cadastro, a alteração e a remoção de um bairro. Tendo como atributos:</p> <ul style="list-style-type: none"> • Nome (String) <p>Todos os atributos são obrigatórios.</p>	Sistema
RFW04	<p>O sistema deve permitir o cadastro, a alteração e a remoção de um logradouro. Tendo como atributos:</p> <ul style="list-style-type: none"> • Nome (String) • Tipo de logradouro (já cadastro) <p>Todos os atributos são obrigatórios.</p>	Sistema
RFW05	<p>O sistema deve permitir o cadastro, a alteração e a remoção de um tipo de logradouro. Tendo como atributos:</p> <ul style="list-style-type: none"> • Tipo de logradouro (String - EX: Rua, Avenida, Travessa, etc.). <p>Todos os atributos são obrigatórios.</p>	Sistema
RFW06	<p>O Sistema deve permitir o cadastro, a alteração e a remoção de um endereço. Tendo como atributos:</p> <ul style="list-style-type: none"> • CEP (String); • bairro (Bairro - Cadastrado Previamente); • Tipo de logradouro (TipoLogradouro- Cadastrado previamente); • Logradouro (Logradouro- Cadastrado previamente); • Cidade (Cidade- Cadastrado previamente); • Estado (Estado- cadastrado previamente); <p>Todos os atributos são obrigatórios. Caso Haja CEP único o usuário terá que inserir os dados manualmente nos sistemas.</p>	

RFW07	<p>O programa precisa permitir um cadastro de clientes com os seguintes dados:</p> <ul style="list-style-type: none">• Nome(Obrigatório) (50 char)• CPF (Obrigatório) (Único) (11 char)• Data de Nascimento (Obrigatório)• Telefone (Obrigatório)• E-mail(Obrigatório) (50 char)• Senha(Obrigatório) (50 char)• Recuperação de Senha (Obrigatório) (50 char)• Opção de login com redes sociais (Desejável)• Identificador do Endereço(Endereço - Cadastrado)	Cliente
RFW08	<p>O sistema deve permitir ao usuário o cadastro dos seus telefones, Guardando:</p> <ul style="list-style-type: none">• DDD• Número• Código• Cliente ao qual está associado	Cliente
RFW09	<p>O sistema deve permitir o cadastro, a alteração e a remoção do carrinho de compra do cliente Guardando:</p> <ul style="list-style-type: none">• Data de criação do carrinho• Quantidade de kits	Cliente
RFW10	<p>O sistema permite cadastrar no Website os produtos disponíveis, ou seja, os Kits de Sensores disponíveis para comprar. Armazenando:</p> <ul style="list-style-type: none">• Identificador• Descrição• Modelo• Latitude• Longitude• Carrinho ao qual está associado (pode ser nulo)• Cliente ao qual está associado(pode ser null)	Sistema

RFW11	O sistema deve permitir o cadastro de tipo de sensor, guardando: <ul style="list-style-type: none">• Tipo Sensor	Sistema
RFW12	O Sistema deve permitir o cadastro de sensores, guardando: <ul style="list-style-type: none">• Id do sensor• Kit ao qual está associado• tipo do sensor (já cadastrado)	Sistema
RFW13	O sistema deve permitir o cadastro, a alteração e a remoção dos cartões de compra do cliente, guardando: <ul style="list-style-type: none">• Tipo do cartão• número de cartão• agência do cartão	Cliente
RFW14	O sistema deve permitir e guardar informações de compra do produto em questão ao cliente, permitindo-o pagar por cartão de crédito, débito, boleto bancário ou PIX. Guardando: <ul style="list-style-type: none">• Quantidade• Data da compra• Valor• Id do cliente• cartão do cliente utilizado(Já cadastrado, pode ser nulo)• Carrinho que ela está associada	Cliente
RFW15	O sistema deve permitir guardar informações dos últimos 6 meses de medições dos sensores, guardando: <ul style="list-style-type: none">• Data da medição• Valor medido• Sensor específico(Já cadastrado)	Sistema
RFW16	O sistema deve permitir ao cliente fazer reclamações, guardando: <ul style="list-style-type: none">• Compra que está se referindo• Descrição da reclamação• Data da reclamação• Hora da reclamação	Cliente

RFW17	<p>O sistema deve permitir ao cliente fazer trocas/devoluções, guardando:</p> <ul style="list-style-type: none">• Compra que está se referindo• Descrição da troca/devolução• Hora da troca/Devolução• Tipo de troca(dinheiro ou produto)• Data da troca/devolução	Cliente
RFW18	<p>O programa precisa permitir que, uma vez cadastrado o cliente, este possa fazer login no site ou aplicativo. Pedindo as seguintes informações:</p> <ul style="list-style-type: none">• E-mail(Obrigatório) (50 char)• Senha(Obrigatório) (50 char)• Opção de login com redes sociais (Desejável)	Cliente
RFW19	<p>O sistema deve: permitir ao administrador a visualização e edição das informações dos clientes, kits e sensores em tempo.</p> <ul style="list-style-type: none">• Quantidade de clientes cadastrados• Quantidade de clientes online• Kits Online• Quantidade de kits• Quantidade de sensores cadastrados	Administração
RFW20	<p>O sistema deve permitir ao usuário a consulta dos dados gerados pelo(s) sensor(es). Os dados disponíveis:</p> <ul style="list-style-type: none">• Umidade• PH• PPM(Parte por milhão)• Temperatura(°C)• Data da coleta(horário/dia/mês/ano)• Id do sensor(id gerado pelo aplicativo) <p>Na aba de pesquisa deve ser possível consultar os dados dos meses no período de 6 meses. O cliente não pode alterar ou remover dados do sistema, apenas consultar</p>	Cliente

Fonte: Elaborada pelo autor

RFW21	<p>O sistema deve conseguir receber um pedido para geração de imagem, recebendo o seguintes dados:</p> <ul style="list-style-type: none"> • Data da petição • Hora da petição • Id do cliente • Conjunto de dados do cliente • PH • Id do sensor • Presente ou ausente • Data da coleta • Hora da coleta • Umidade • Id do sensor • Presente ou ausente • Data da coleta • Hora da coleta • Temperatura(°C) • Id do sensor • Presente ou ausente • Data da coleta • Hora da coleta • Parte por Milhão(PPM) • Id do sensor • Presente ou ausente • Data da coleta • Hora da coleta • Resolução desejada da imagem • Latitude do Kit Especificado • Longitude do Kit Especificado Podendo também selecionar mais de um sensor. 	Cliente
RFW22	<p>O sistema deve gerar imagens de sentido artístico a partir de um conjunto de dados coletados a partir do conjunto de sensores do cliente e escolhidos pelo próprio.</p>	Sistema

RFW23	<p>Uma vez geradas as imagens do cliente, o sistema deve disponibilizar as imagens geradas, para que o cliente possa realizar o download em:</p> <ul style="list-style-type: none">• PDF• JPEG• PNG <p>Também deve ser possível compartilhar a imagem nas redes sociais, diretamente do site ou app, onde o cliente pode fazer a propaganda da marca NaturArt com uma legenda gerada automaticamente.</p>	Sistema
RFW24	<p>Uma vez que o cliente faça o pedido da imagem e está sendo gerada, o cliente deve poder fazer o download da imagem nos formatos, na qualidade que desejar</p> <ul style="list-style-type: none">• PDF• PNG• JPEG	Cliente

Tabela 2 – Requisitos Funcionais do Sistema Mobile

Código	Descrição	Ator
RFM01	<p>O sistema programa precisa permitir um cadastro de clientes com os seguintes dados:</p> <ul style="list-style-type: none"> • nome (Char); • CPF (String); • Endereço Principal (String); • Data de Nascimento (String); • Telefone (String); • E-mail (String); • Senha (String); • Recuperação de Senha (String); • Opção de login com as redes sociais. 	Cliente
RFM02	<p>O programa precisa permitir que, uma vez cadastrado o cliente, este possa fazer login no site ou aplicativo. Pedindo as seguintes informações:</p> <ul style="list-style-type: none"> • E-mail (String); • Senha (String); • Opção de login com as redes sociais. 	Cliente
RFM03	O sistema disponibiliza opções de compra do produto em questão ao cliente, permitindo-o pagar por cartão de crédito, débito, boleto bancário ou Pix.	Cliente
RFM04	O cliente pode cadastrar um ou mais sensores quando estiver logado no aplicativo mobile. Obrigatoriamente, a primeira conexão do sensor na rede Wifi será efetuada via USB. O cliente deve estar logado na sua rede Wifi, assim quando requisitado no aplicativo, o mesmo deverá efetuar a conexão do sensor na mesma rede logada. Também é possível registrar mais de uma rede Wifi. No momento do cadastro do sensor, é gravado no próprio sensor, o id do cliente para identificá-lo. Desta maneira, a comunicação entre a obtenção de dados dos sensores e o banco de dados será facilitada, pois o id do cliente e o n° de cadastro do sensor serão fornecidos conjuntamente. O cliente pode realizar alterações das propriedades dos sensores, por exemplo, nome do sensor (apelido).	Cliente
RFM05	O sistema permite ao usuário a visualização das informações coletadas pelo sensores, seja em um dia, semana ou espaço de tempo desejado. Apresentando estes de forma didática.	Cliente
RFM06	O cliente pode visualizar as informações coletadas pelos sensores, seja em um dia, semana ou espaço de tempo desejado. Apresentando estes de forma didática.	Cliente

RFM07	<p>O sistema deve conseguir receber um pedido para geração de imagem, recebendo o seguintes dados:</p> <ul style="list-style-type: none">• Data da petição (dia/mês/ano);• Senha (hora/minutos);• Identificador do cliente(id do cliente);• Conjunto de dados do cliente(estes podem ser desde dados de um dia até dados de um mês);• Resolução da imagem desejada.	Cliente
RFM08	<p>Uma vez que o cliente faça o pedido da imagem e está sendo gerada, o cliente deve poder fazer o download da imagem nos formatos:</p> <ul style="list-style-type: none">• PDF;• JPEG;• PNG.	Cliente
RFM09	<p>O sistema deve permitir ao usuário, uma vez que este esteja conectado a um sensor, configurar o sensor conectado, especificando a frequência com que o sensor deve enviar informações.</p>	Cliente
RFM10	<p>O sistema permite ao cliente adicionar produtos a um carrinho para compra-los, e guarda informações da compra:</p> <ul style="list-style-type: none">• Data adição ao carrinho• Código dos produtos	Cliente

Fonte: Elaborada pelo autor

Tabela 3 – Requisitos Funcionais do Sistema do Hardware

Código	Descrição	Ator
RFH01	O sistema deve, por meio de um sistema cabeado, conectar-se pela primeira vez ao sistema mobile e deve permitir ser configurado a partir do aplicativo mobile	Sistema do Sensor
RFH02	O sistema deve, por meio da configuração do cliente, permitir a conexão com a internet por meio do Wifi.	Sistema do Sensor
RFW03	O sistema deve enviar as informações de acordo com a frequência pedida pelo cliente por meio da internet: <ul style="list-style-type: none">• PH;• Umidade;• PPM(Parte por milhão);• Temperatura;• Identificador do sensor;	Sistema do Sensor
RFW04	O sistema deve permitir a customização do tempo de envio de informações individuais para cada sensor.	Sistema do Sensor

Fonte: Elaborada pelo autor

1.2.2 Requisitos Não-Funcionais

Tabela 4 – Requisitos Não Funcionais

Código	Descrição	Nome	Notabilidade
RNF01	As informações dos usuários não devem ser vazadas e somente os usuários com poder de administrador podem ver.	Integridade	Essencial
RNF02	O servidor, o site e o aplicativo devem ficar disponíveis sempre.	Disponibilidade	Essencial
RNF03	O aplicativo mobile deve ter as mesmas funcionalidades que o site Web, logo deve ser possível adquirir os produtos através do mesmo. O cliente deve poder utilizar facilmente o gerenciamento de seus dados para obter imagens.	Usabilidade	Essencial
RNF04	O sistema deve ser possível de utilizar em qualquer sistema operacional e mobile também.	Portabilidade	Essencial
RNF05	Todos os códigos e alterações devem ser versionados através de um software de controle de versão, com a finalidade de prevenir possíveis erros que percam dados atualizados, deixando o software de vendas, fora de uso, para manutenção, no mínimo 12 horas fora do ar.	Manutenibilidade	Essencial

RNF06	O sistema deve ter uma versão de histórico das escolhas do usuário disponíveis para uso em até 15 minutos após sair do site, caso ocorra algum erro de software/hardware por parte do usuário. E salvar a versão mais recente, por tempo indeterminado caso o sistema caia.	Robustez	Essencial
RNF07	No momento da compra, é necessário gerar a nota fiscal eletrônica e enviar diretamente para o e-mail do cliente responsável pela compra.	Nota Fiscal Eletrônica	Essencial

1.2.3 Requisitos Adiados

Ao longo do projeto, não houve nenhum requisito adiado.

1.3 Casos de Uso

O caso de uso tem como objetivo descrever como será a funcionalidade de um sistema. Um caso de uso consiste principalmente em uma especificação textual (chamada de Especificação de Caso de Uso) que contém uma descrição do fluxo de eventos que descreve a interação entre os atores e o sistema. A especificação também contém normalmente outras informações, como pré-condições, pós-condições, requisitos especiais e cenários principais. O caso de uso também pode ser representado visualmente em UML para mostrar o relacionamento com outros casos de uso e atores.

A UML (Unified Modeling Language ou “Linguagem de Modelagem Unificada”) possui uma série de diagramas – cada um com uma finalidade específica com suas respectivas regras, premissas e restrições – que podem ser utilizados na especificação de sistemas (tanto sistemas de software quanto em sistemas de hardware, por exemplo).

Estes diagramas são divididos em dois grandes grupos: diagramas estruturais e diagramas comportamentais.

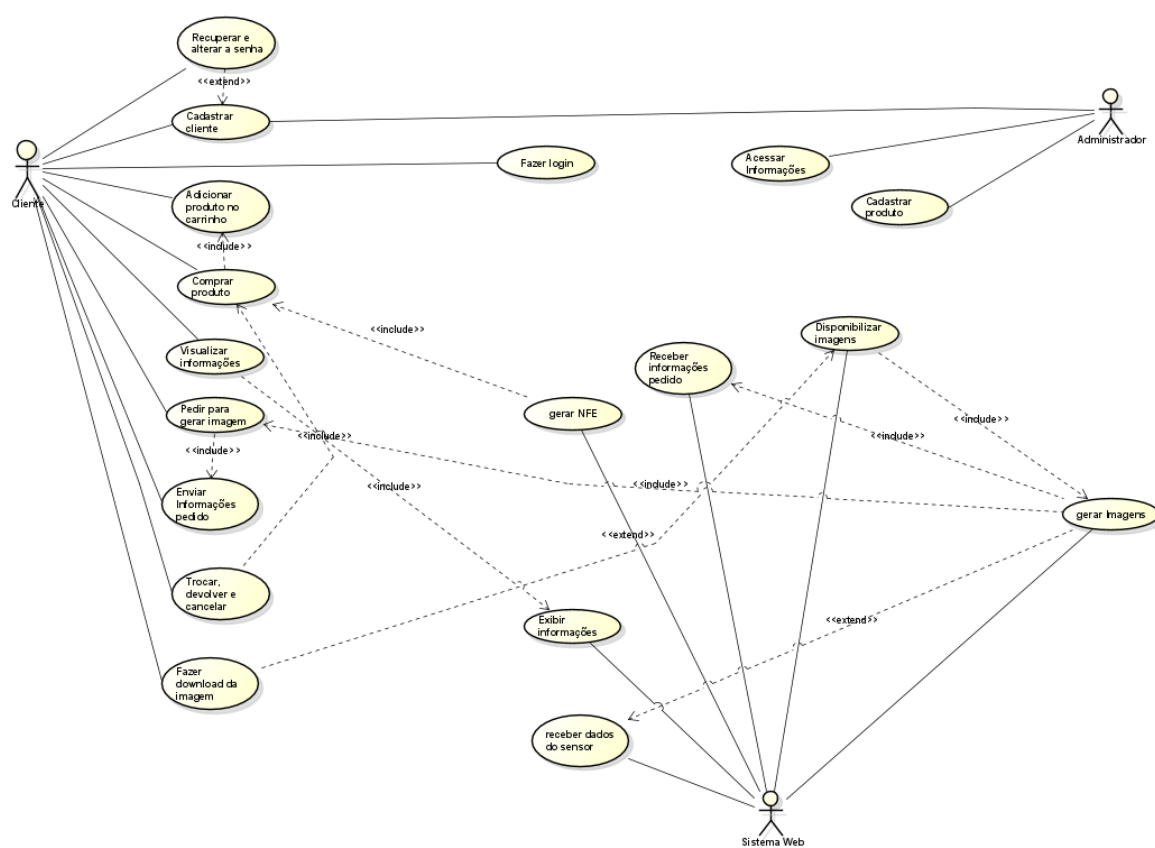
Os diagramas estruturais são utilizados para se especificar a estrutura do sistema, a parte estática. Os diagramas comportamentais são utilizados para especificar o comportamento do sistema, a parte dinâmica do sistema. Exemplo é o Diagrama de Sequência, que demonstra como algum pedaço do sistema mais dinâmico (ou algoritmo ou uma funcionalidade, por exemplo) se comportará num contexto específico, Diagrama de Caso de Uso (demonstra como uma funcionalidade é utilizada, como ela se comportará diante de eventos, inputs, exceções etc.).

1.3.1 Diagrama de Casos de Uso

O diagrama de caso de uso resume os detalhes dos usuários do seu sistema (também conhecidos como atores) e as interações deles com o sistema.

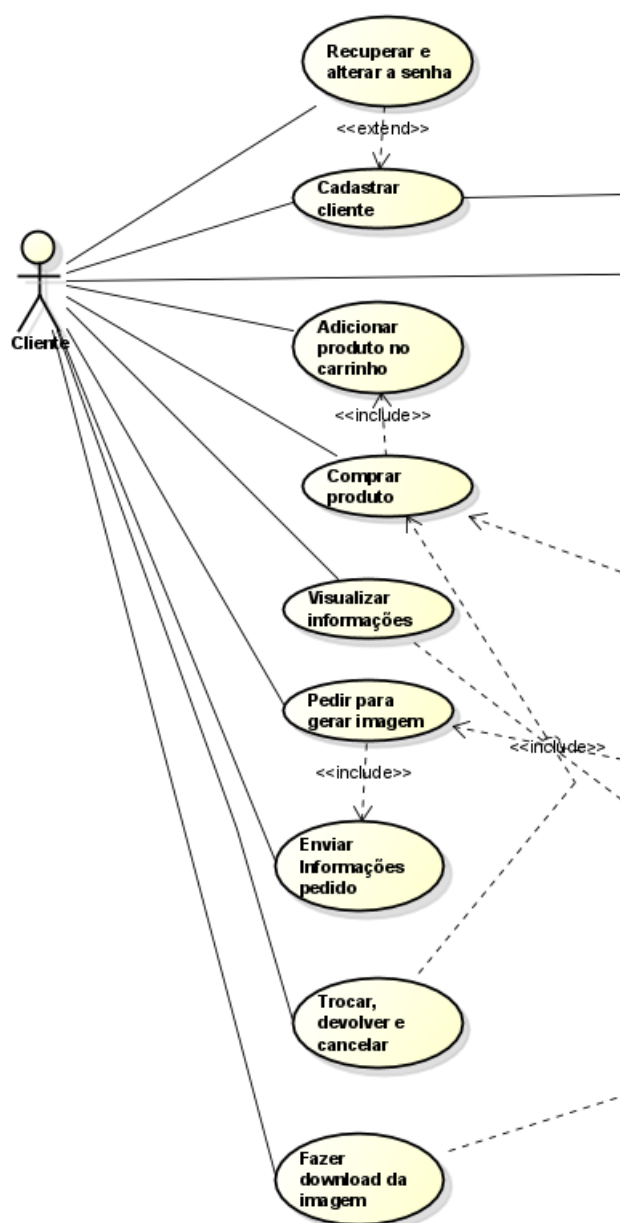
Esse diagrama documenta o que o sistema faz do ponto de vista do usuário. Em outras palavras, ele descreve as principais funcionalidades do sistema e a interação dessas funcionalidades com os usuários do mesmo sistema. Nesse diagrama não nos aprofundamos em detalhes técnicos que dizem como o sistema faz.

Figura 1 – Diagrama de casos de uso - Sistema web (visão geral)



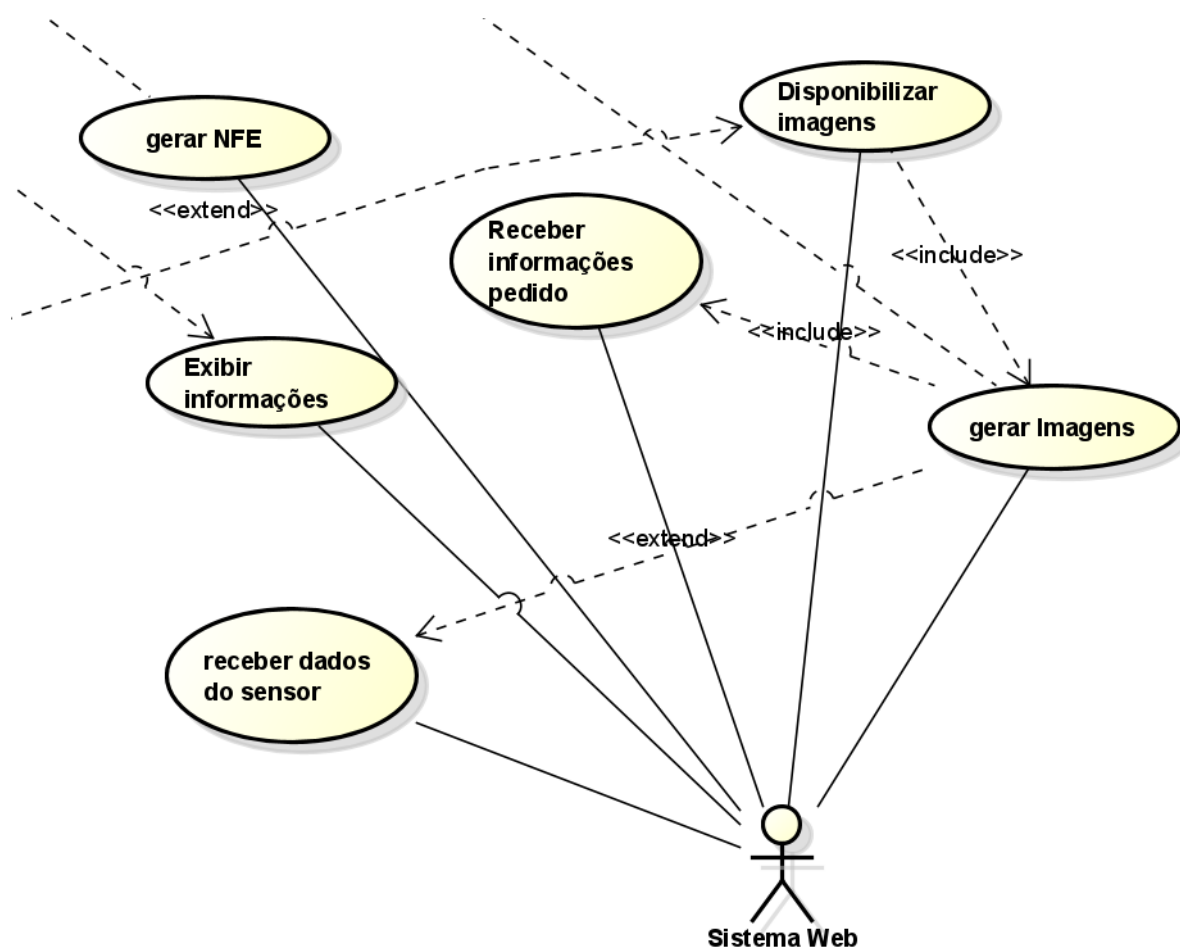
Fonte: Elaborada pelo autor

Figura 2 – Diagrama de casos de uso - Sistema Web (cliente)



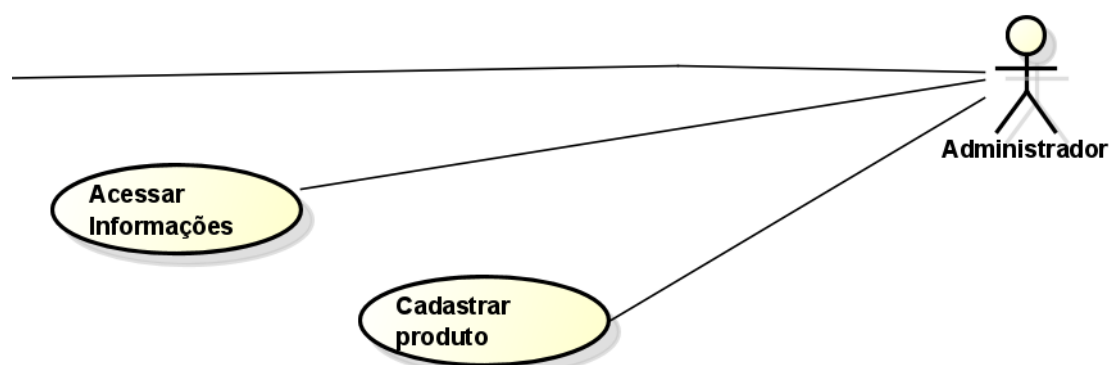
Fonte: Elaborada pelo autor

Figura 3 – Diagrama de casos de uso - Sistema Web (web)



Fonte: Elaborada pelo autor

Figura 4 – Diagrama de casos de uso - Sistema Web (administrador)



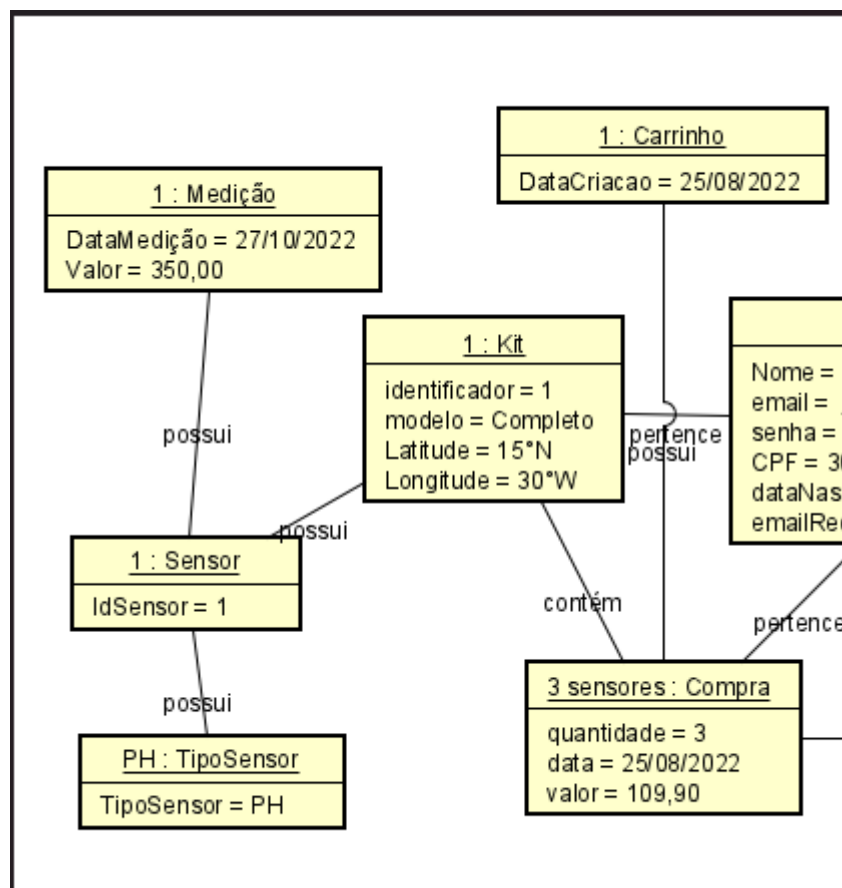
Fonte: Elaborada pelo autor

1.4 Diagrama de Objetos

O diagrama de objetos é uma variação do diagrama de classes e utiliza quase a mesma notação. A diferença é que o diagrama de objetos mostra os objetos que foram

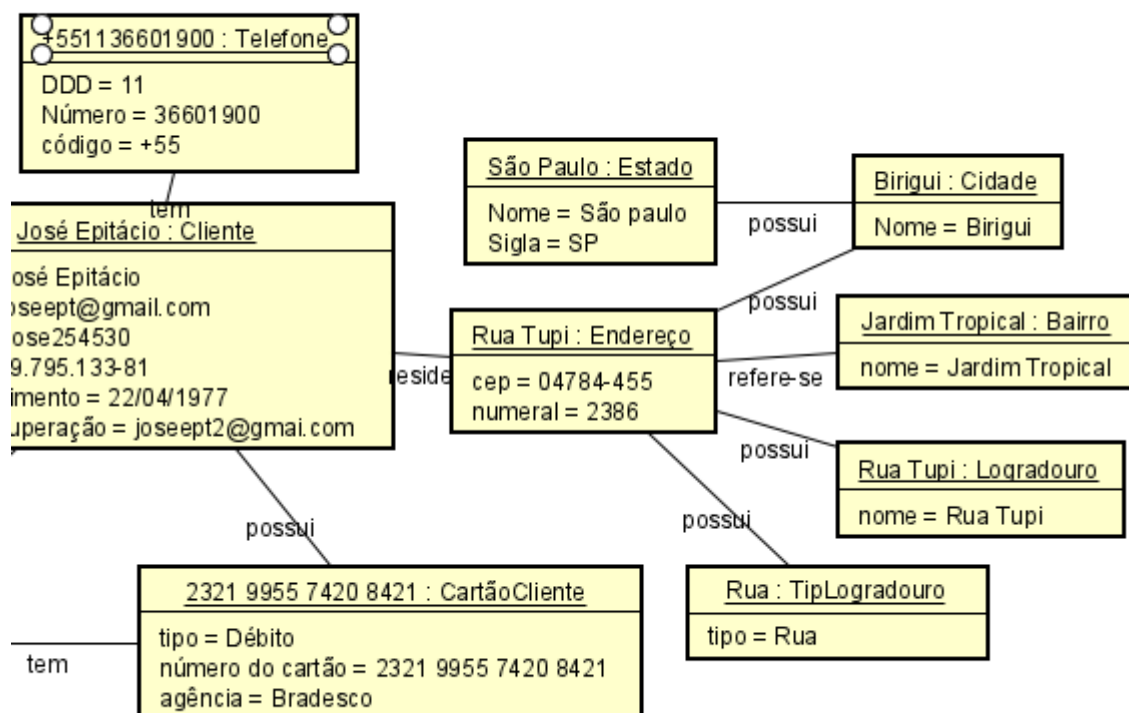
instanciados das classes. O diagrama de objetos é como se fosse o perfil do sistema em um certo momento de sua execução.

Figura 5 – Diagrama de Objetos - 1 parte



Fonte: Elaborada pelo autor

Figura 6 – Diagrama de Objetos - 2 parte



Fonte: Elaborada pelo autor

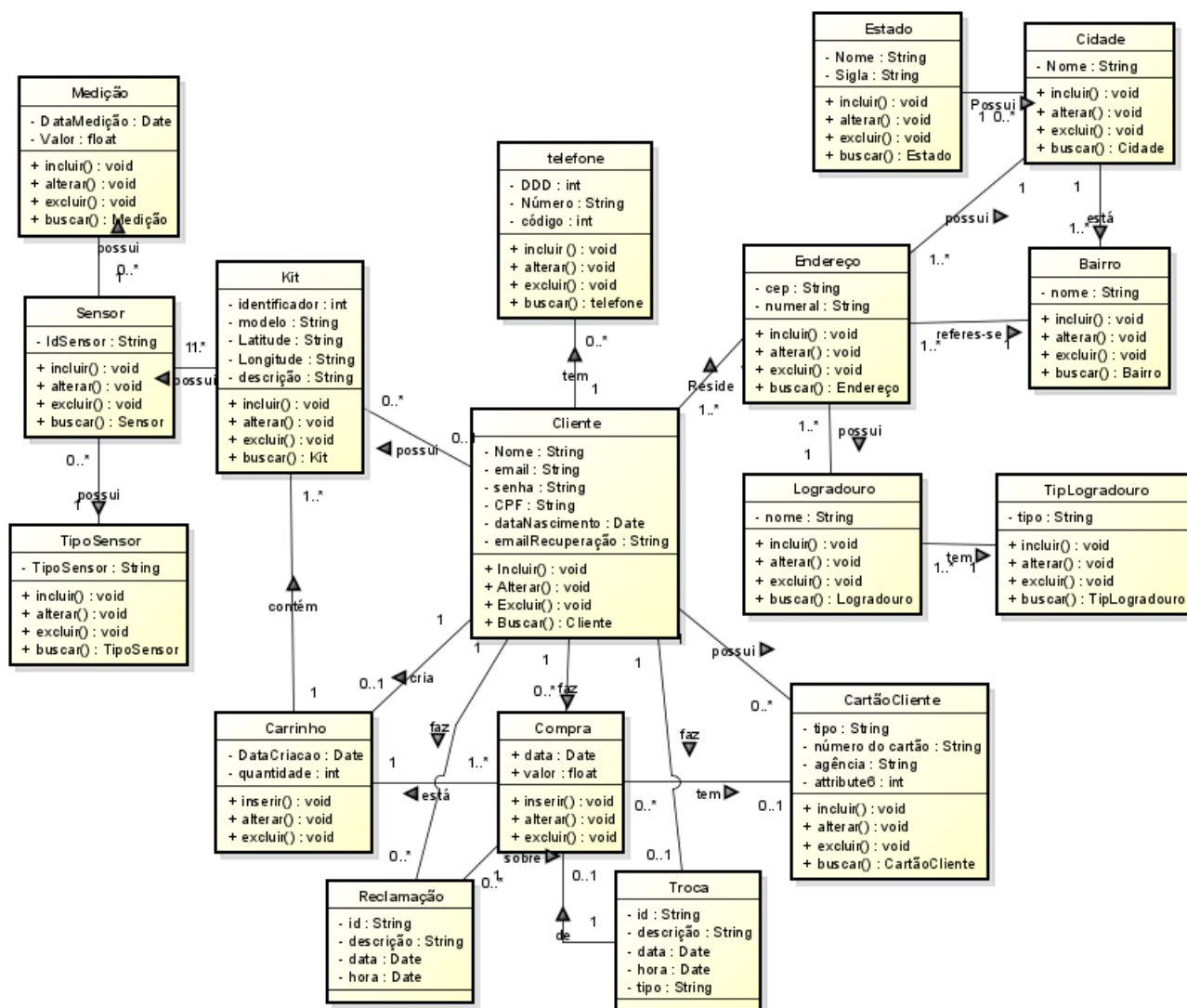
1.5 Diagrama de Classes de Análise

Em UML, diagramas de classes são um de seis tipos de diagrama estrutural. Os diagramas de classe são fundamentais para o processo de modelagem de objetos e modelam a estrutura estática de um sistema. Dependendo da complexidade de um sistema, é possível utilizar um único diagrama de classe para modelar um sistema inteiro ou vários diagramas de classe para modelar os componentes de um sistema. Os diagramas de classe são as cópias do sistema ou subsistema. Pode-se utilizar os diagramas de classe para modelar os objetos que compõem o sistema, para exibir os relacionamentos entre os objetos e para descrever o que esses objetos fazem e os serviços que eles fornecem.

Os diagramas de classe são úteis em muitos estágios do design do sistema. No estágio de análise, um diagrama de classe pode ajudá-lo a compreender os requisitos do domínio do problema e a identificar seus componentes. Em um projeto de software orientado a objetos, os diagramas de classe criados durante os estágios iniciais do projeto contêm classes que normalmente são convertidas em classes e objetos de software reais quando você grava o código. Posteriormente, é possível refinar a análise e os modelos conceituais anteriores em diagramas de classe que mostrem as partes específicas do sistema, interfaces com o usuário, implementações lógicas e assim por diante. Os diagramas de

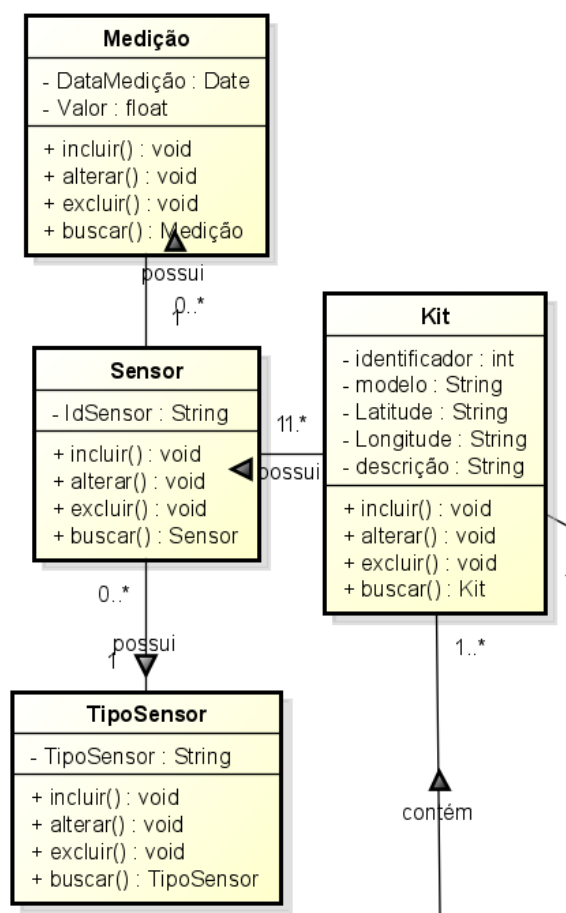
classe tornam-se, então, uma captura instantânea que descreve exatamente como o sistema funciona, os relacionamentos entre os componentes do sistema em vários níveis e como planeja implementar esses componentes.

Figura 7 – Diagrama de Classes - Visão geral)



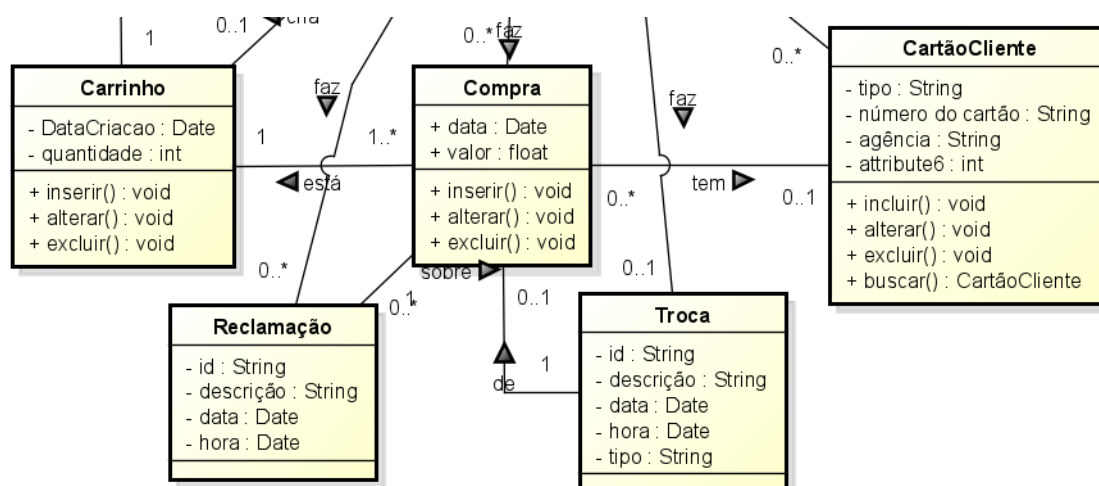
Fonte: Elaborada pelo autor

Figura 8 – Diagrama de Classes - Kit



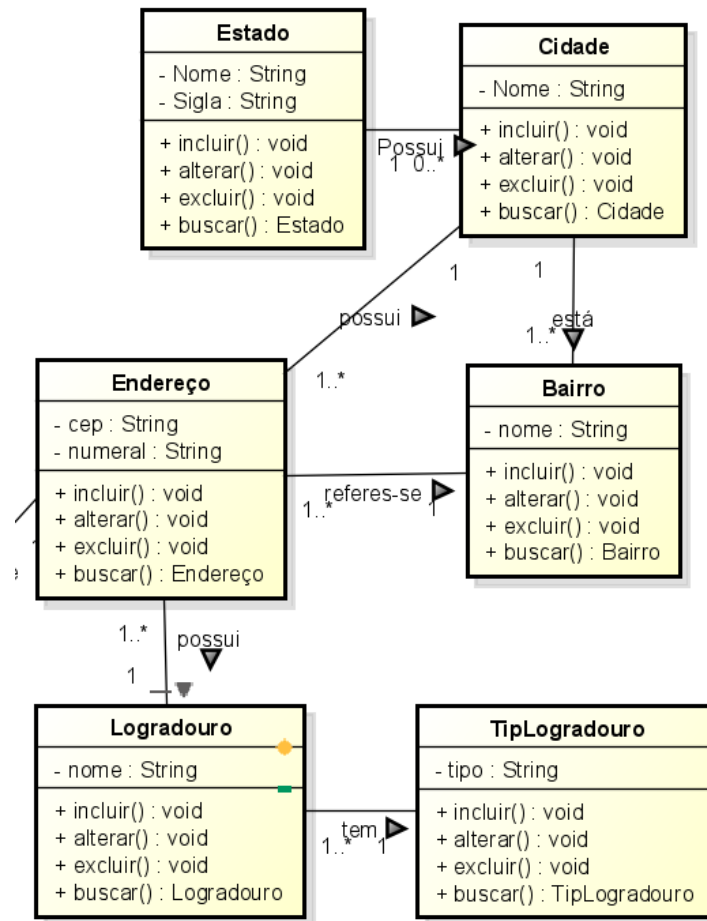
Fonte: Elaborada pelo autor

Figura 9 – Diagrama de Classes - Compra



Fonte: Elaborada pelo autor

Figura 10 – Diagrama de Classes - Endereço



Fonte: Elaborada pelo autor

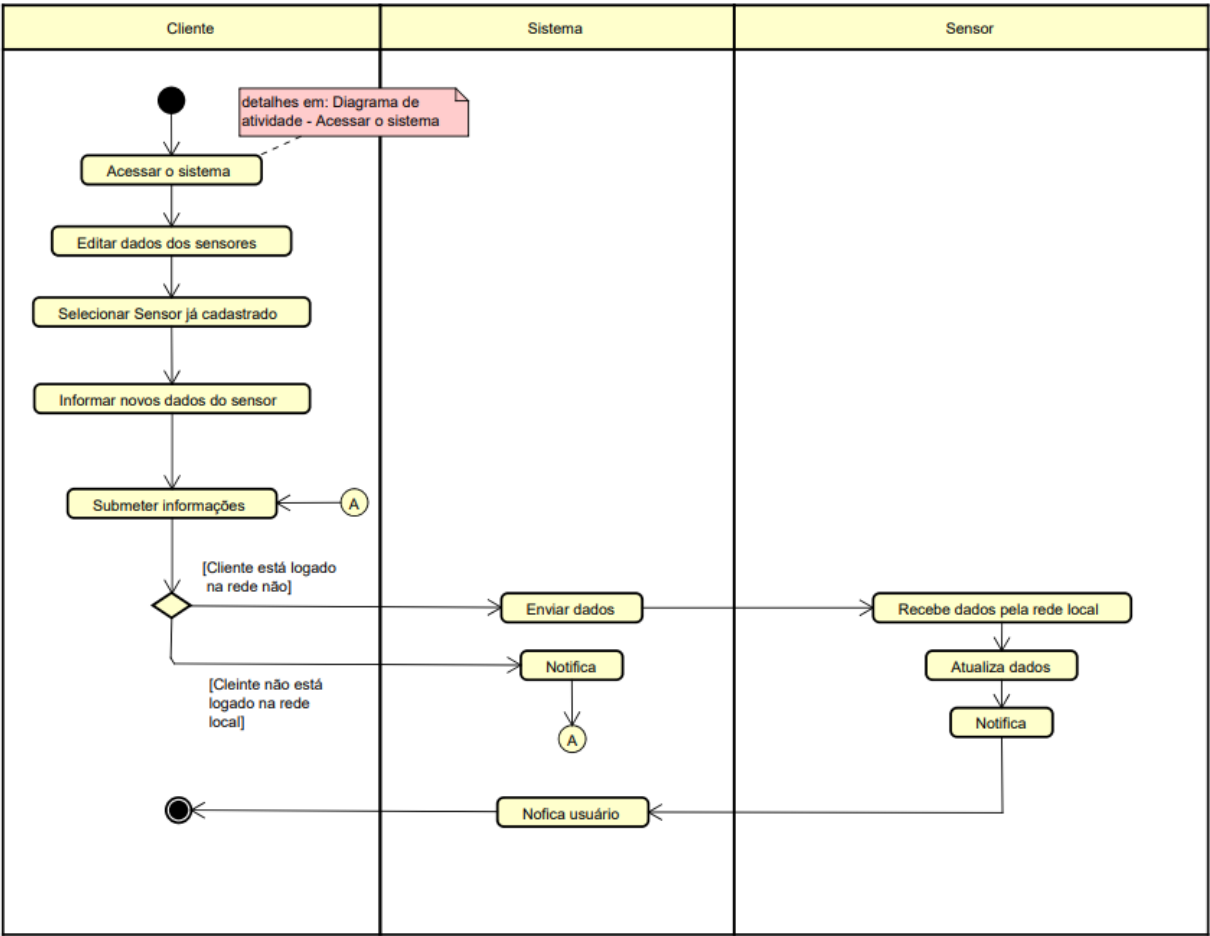
1.6 Diagrama de Atividades

Um diagrama de atividade ilustra a natureza dinâmica de um sistema pela modelagem do fluxo de controle de atividade à atividade. Uma atividade representa uma operação em alguma classe no sistema que resulta em uma mudança no estado do sistema.

Tipicamente, diagramas de atividades são usados para modelar fluxos de processos, processos de negócios ou operações internas. o diagrama de atividades é similar a uma máquina de estados, mas tem um propósito diferente, o qual envolve capturar ações e seus resultados em termos de mudanças do estado do objeto.

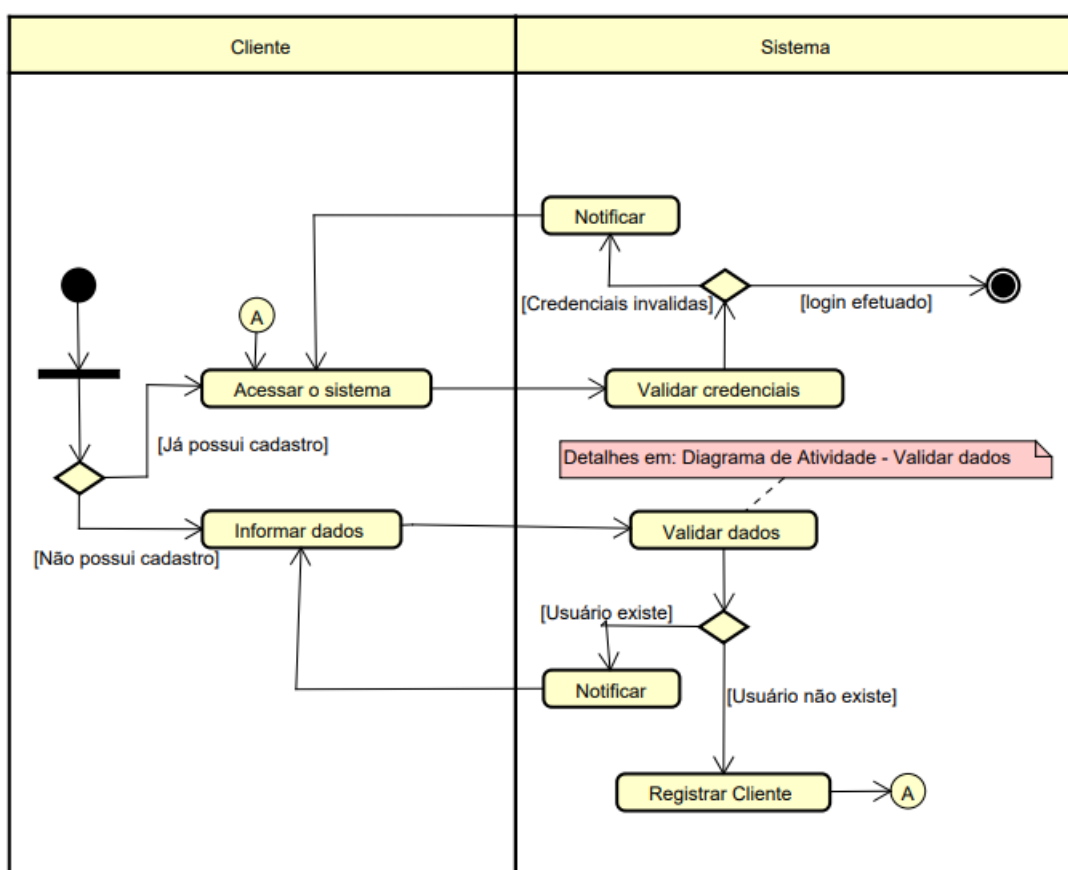
A seguir estão representados os principais diagramas de atividade do projeto.

Figura 11 – Editar dados dos sensores



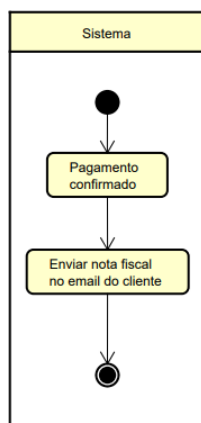
Fonte: Elaborada pelo autor

Figura 12 – Acessar o sistema



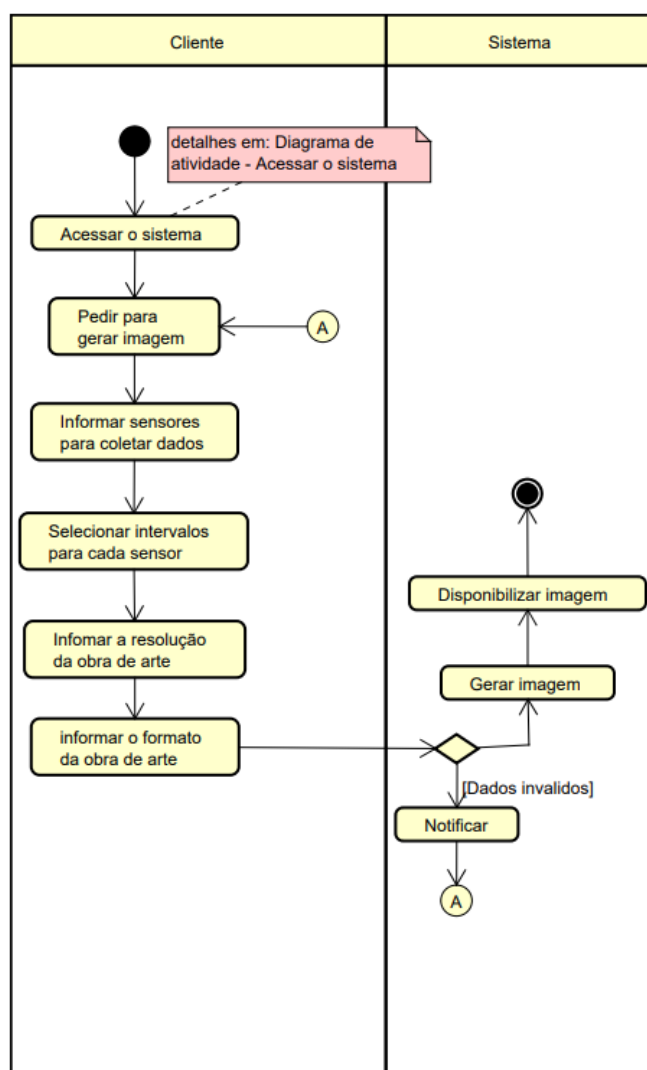
Fonte: Elaborada pelo autor

Figura 13 – Enviar Nota Fiscal



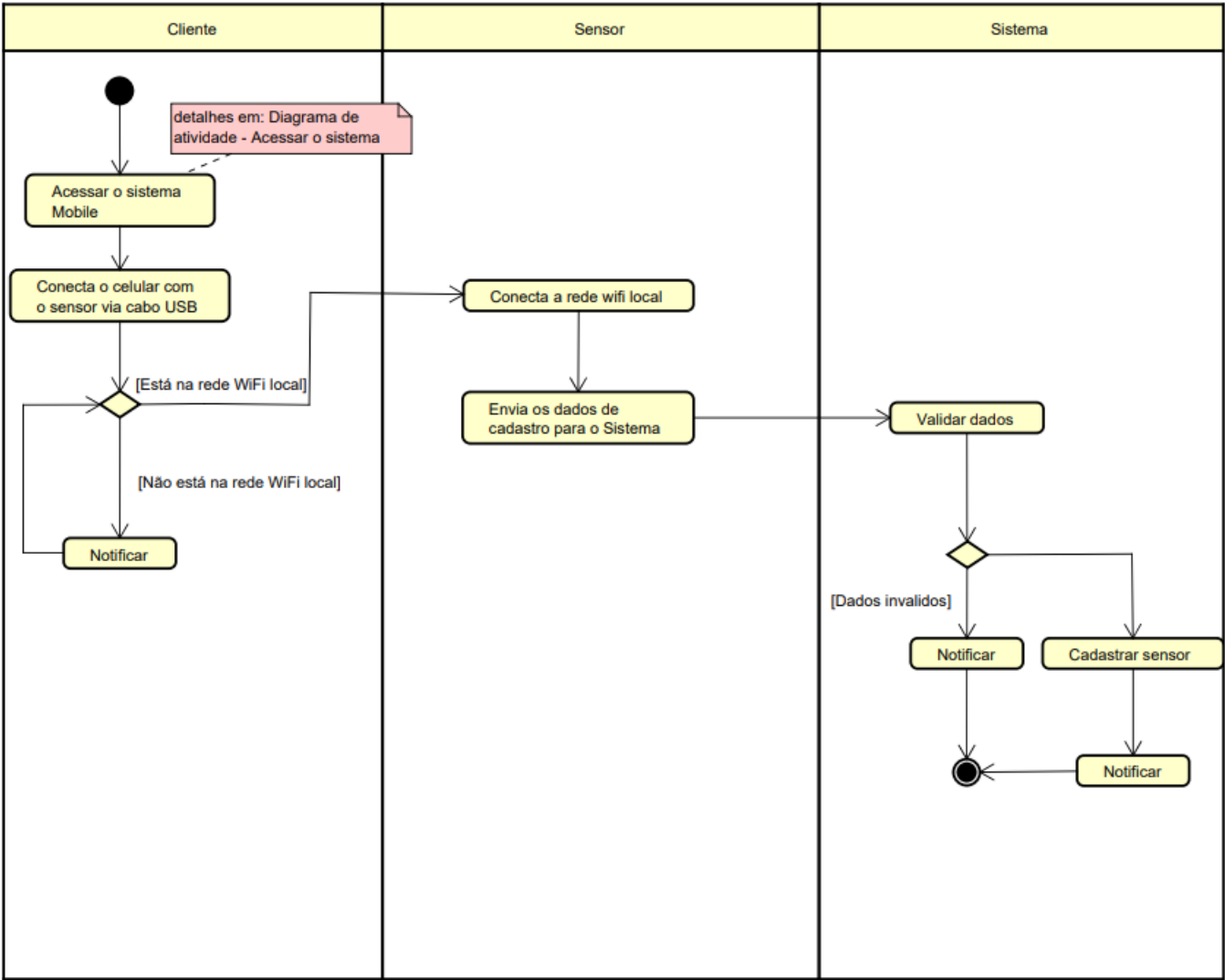
Fonte: Elaborada pelo autor

Figura 14 – Pedir para gerar imagem



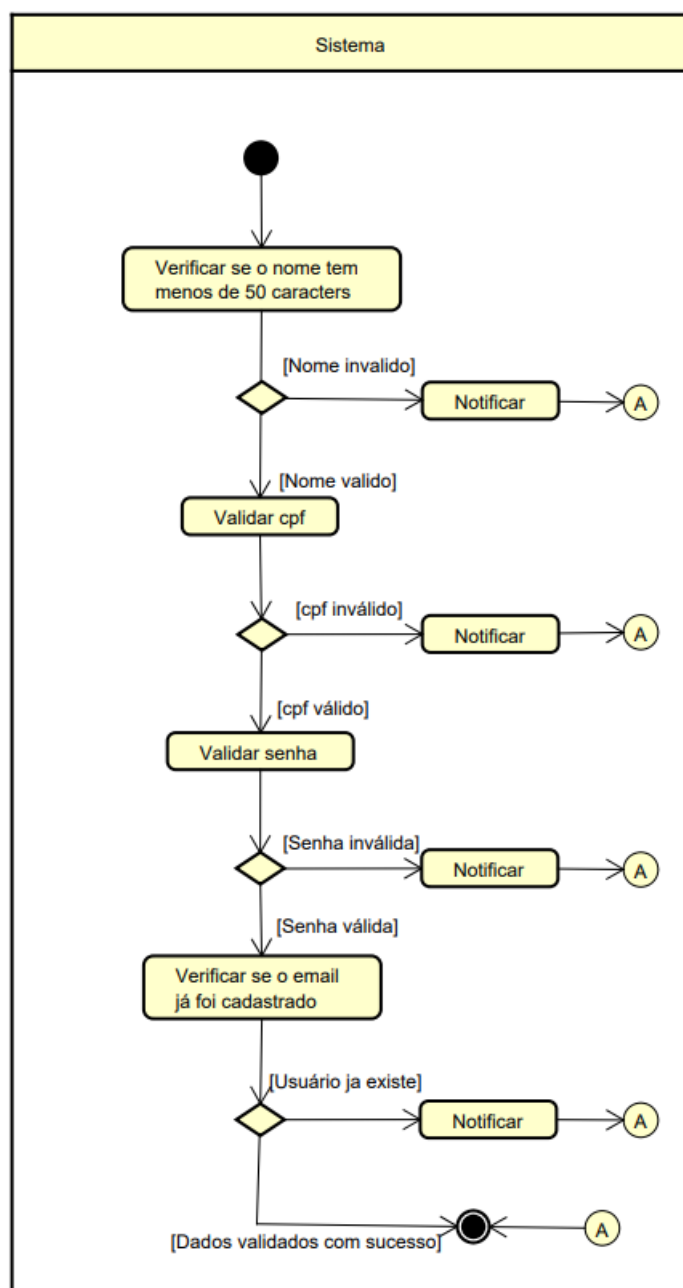
Fonte: Elaborada pelo autor

Figura 15 – Primeira conexão com o sensor



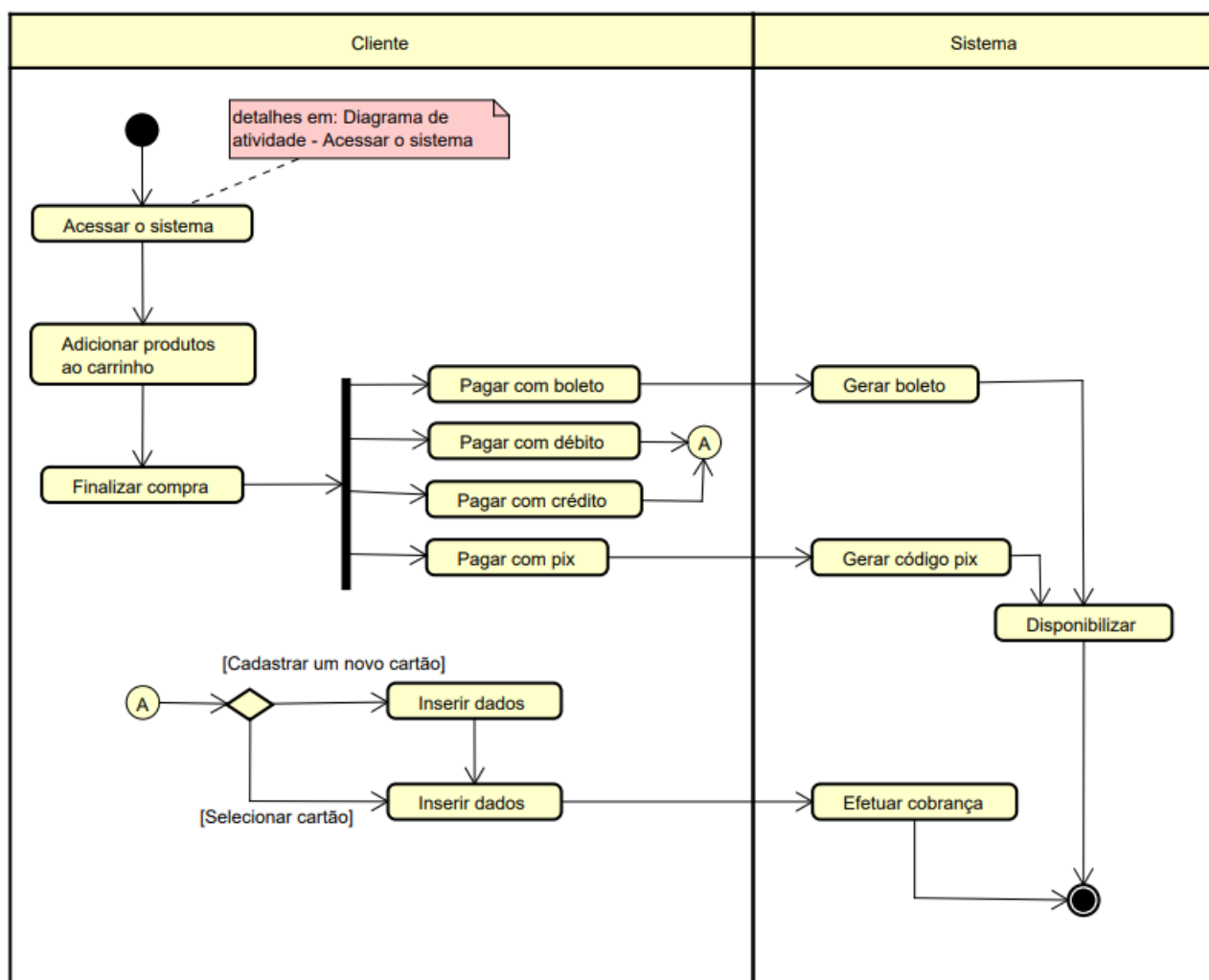
Fonte: Elaborada pelo autor

Figura 16 – Validar dados



Fonte: Elaborada pelo autor

Figura 17 – Comprar produto

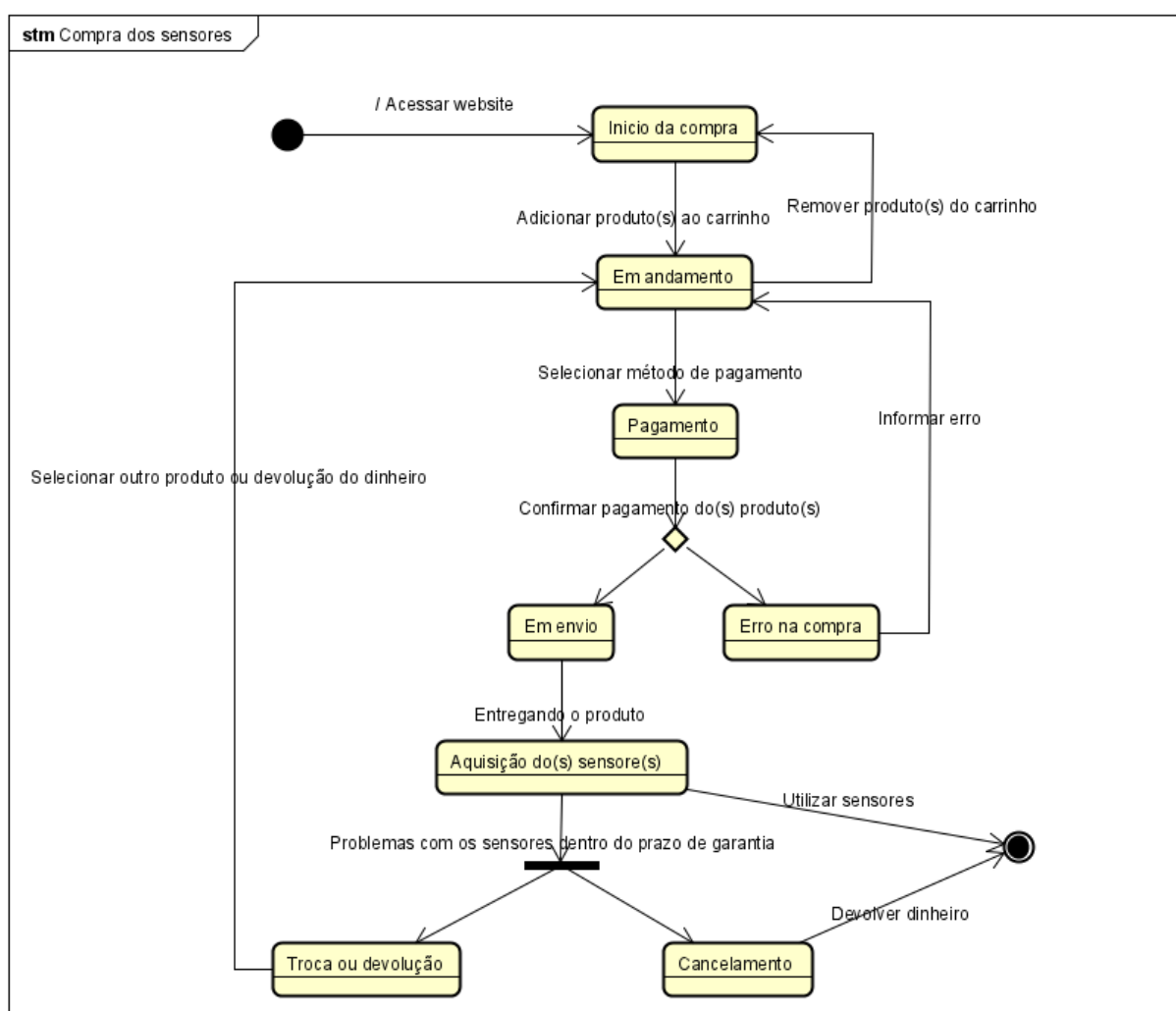


Fonte: Elaborada pelo autor

1.7 Diagrama de Estados

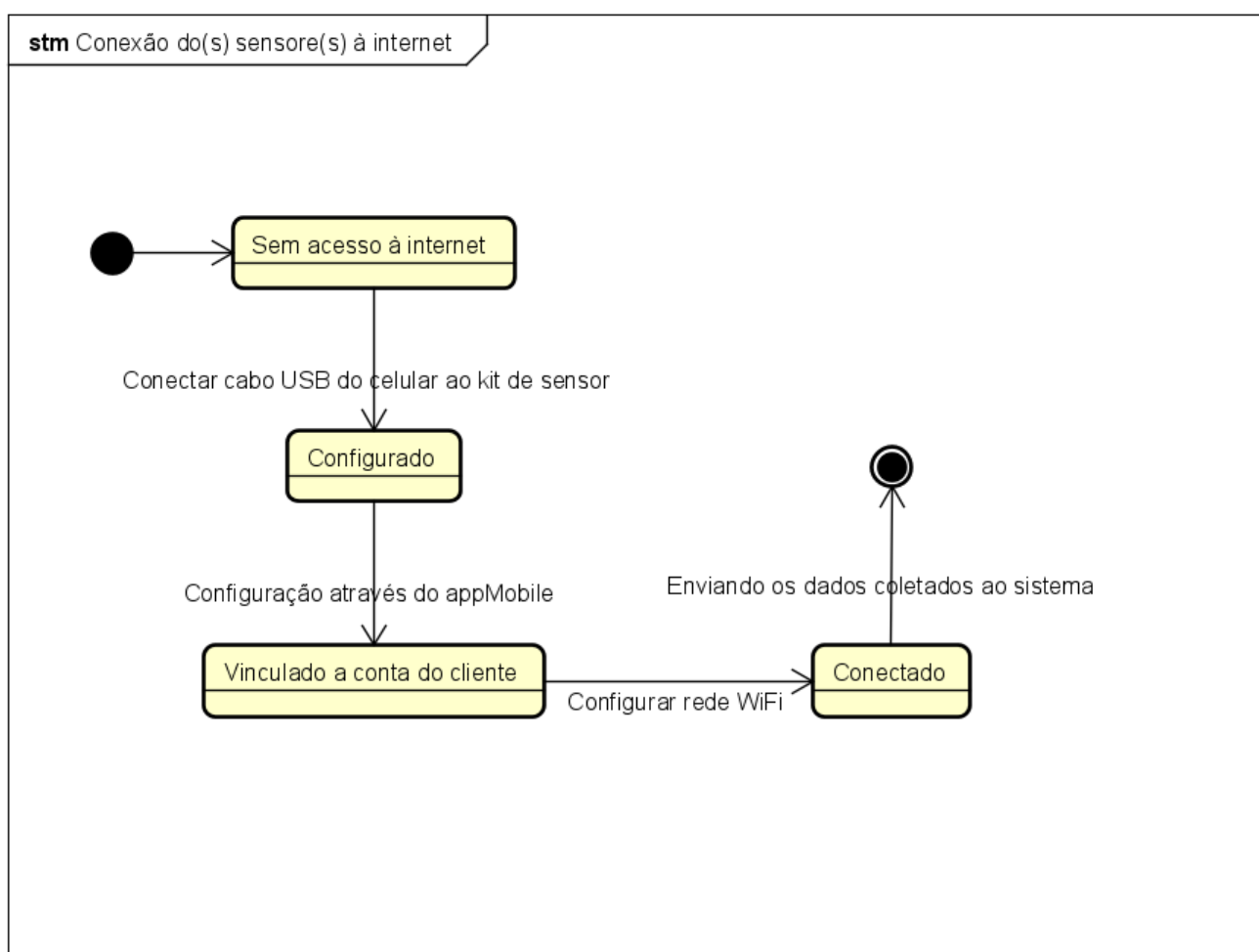
Um diagrama de estados detalha os possíveis estados pelos quais pode passar um objeto e as transações responsáveis pelas suas mudanças de estado, tomando por base a execução de um processo dentro do sistema que se está considerando. A seguir estão os principais processos de execução do projeto, representados por diagramas de estados:

Figura 18 – Diagrama de Estados sobre a compra de um sensor



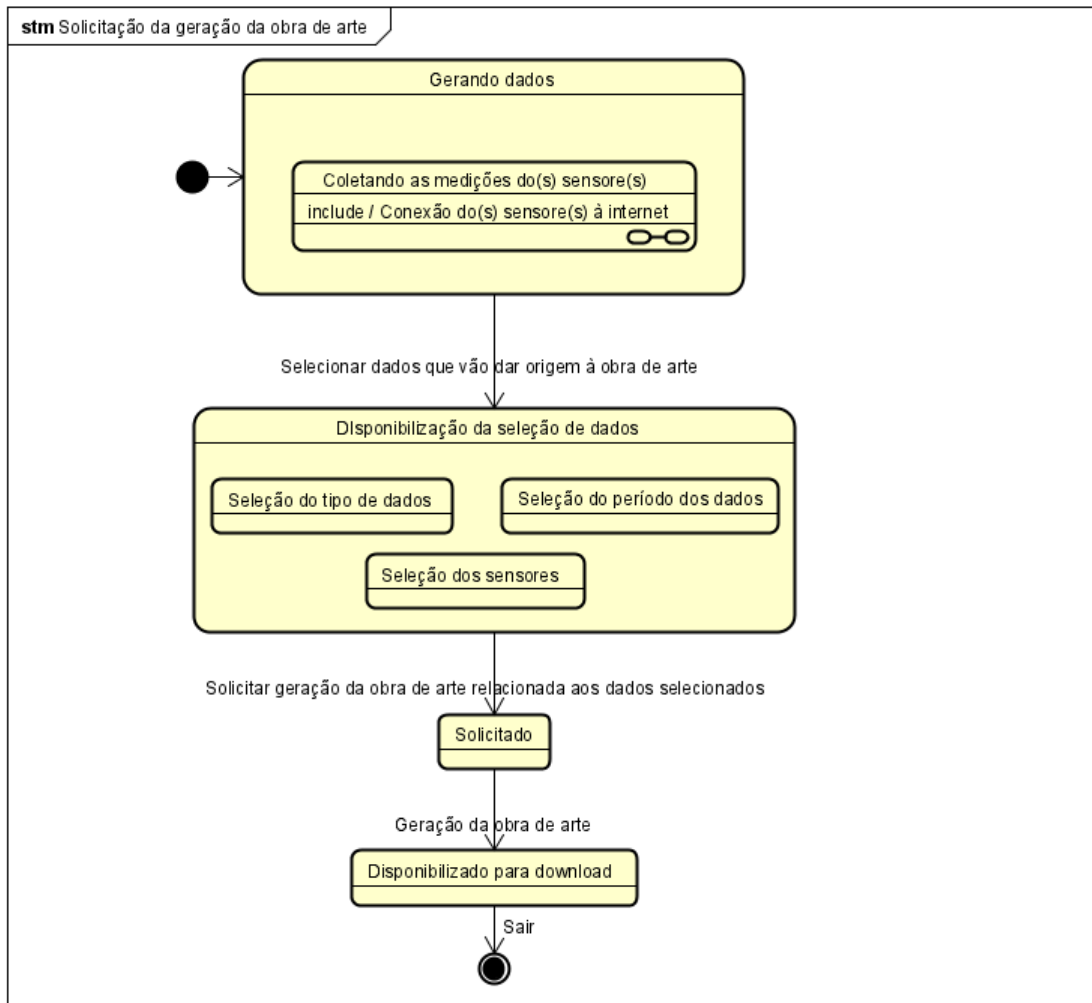
Fonte: Elaborada pelo autor

Figura 19 – Diagrama de Estados sobre a conexão do(s) sensore(s) à internet



Fonte: Elaborada pelo autor

Figura 20 – Diagrama de Estados sobre a solicitação da geração da obra de arte

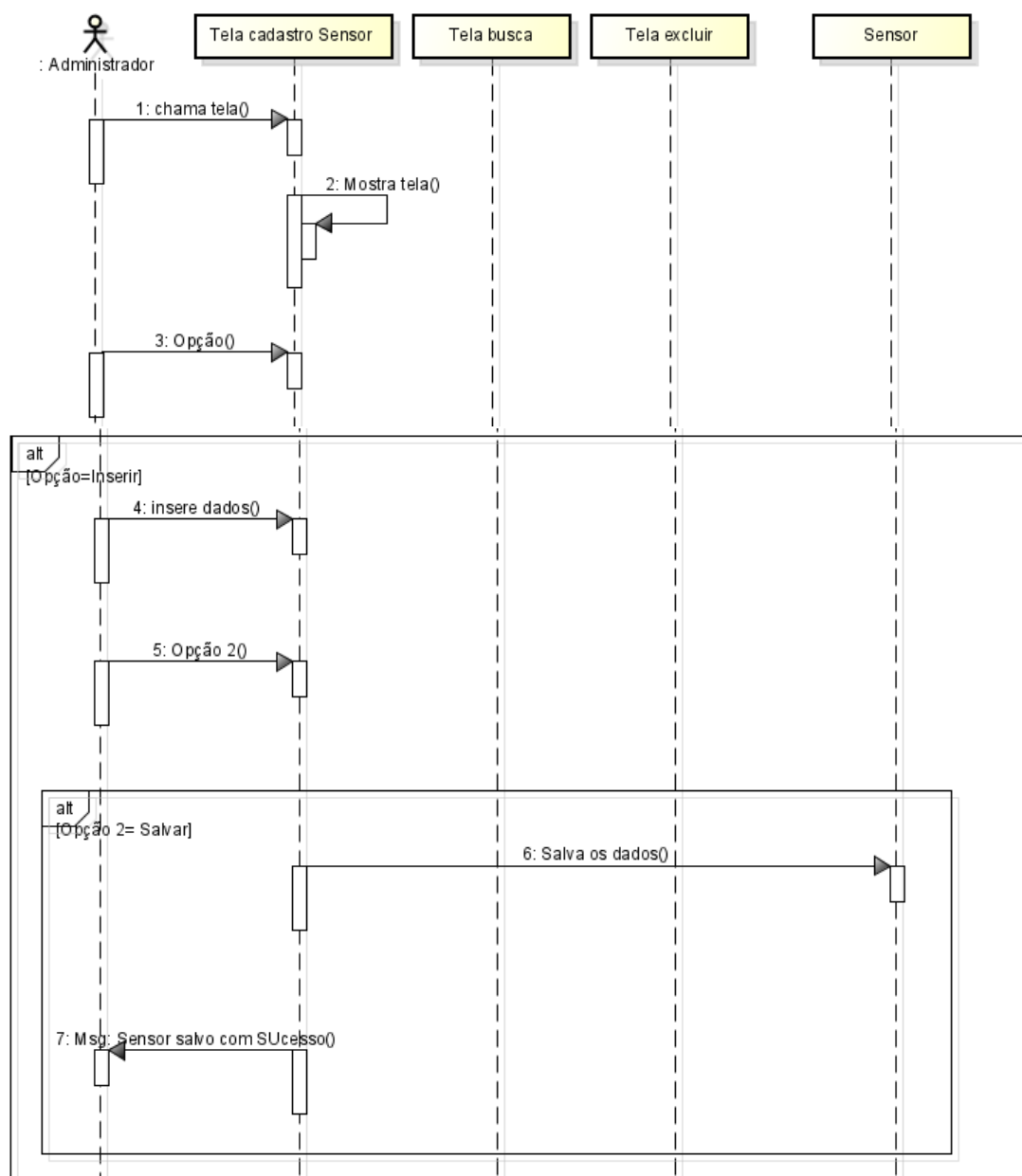


Fonte: Elaborada pelo autor

1.8 Diagrama de Sequências

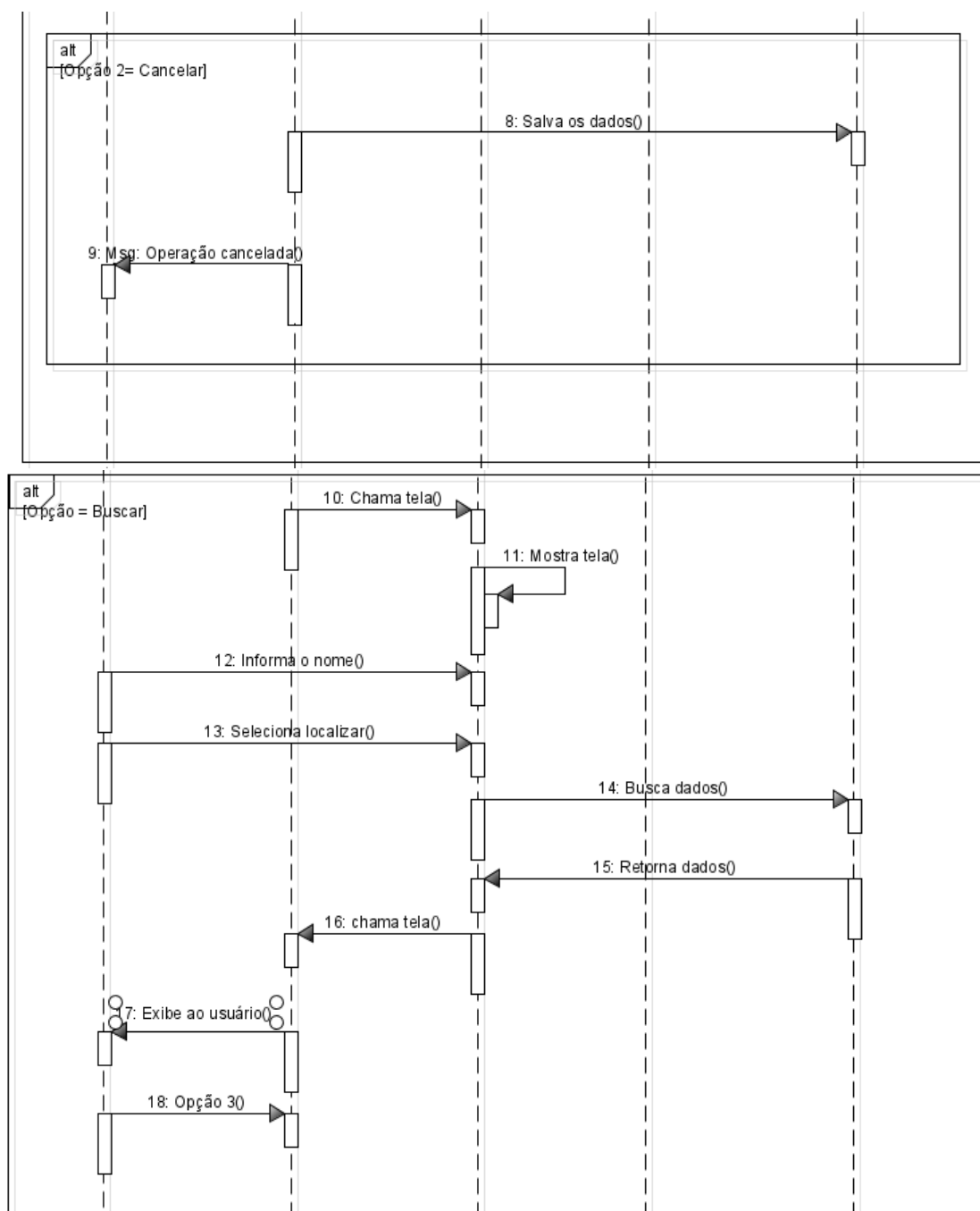
Um diagrama de sequência é o que dá seguimento para as ações do sistema, ou que representa, uma sequência de processos que existem em um computador. Usando o método de 'linhas de vida', o diagrama de sequências, também é conhecido como diagrama de interações ou de eventos, já que ele pode descrever como, e em qual ordem, um grupo de objetos trabalham em conjunto. A seguir, estão os procedimentos para a criação e execução do projeto, representado por diagramas.

Figura 21 – Diagrama de Sequências sobre o Cadastro do sensor



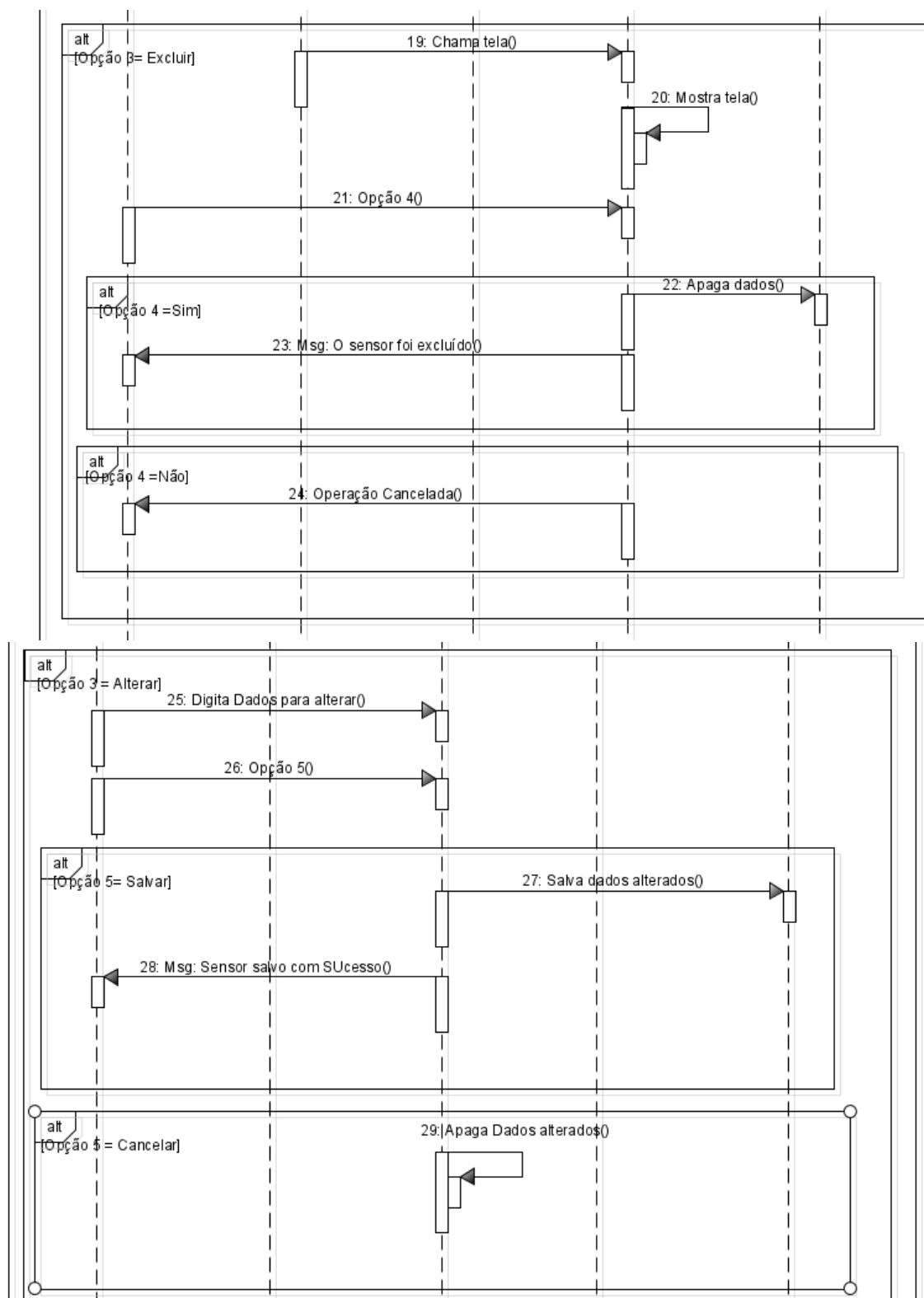
Fonte: Elaborada pelo autor

Figura 22 – Diagrama de Sequências sobre o Cadastro do sensor



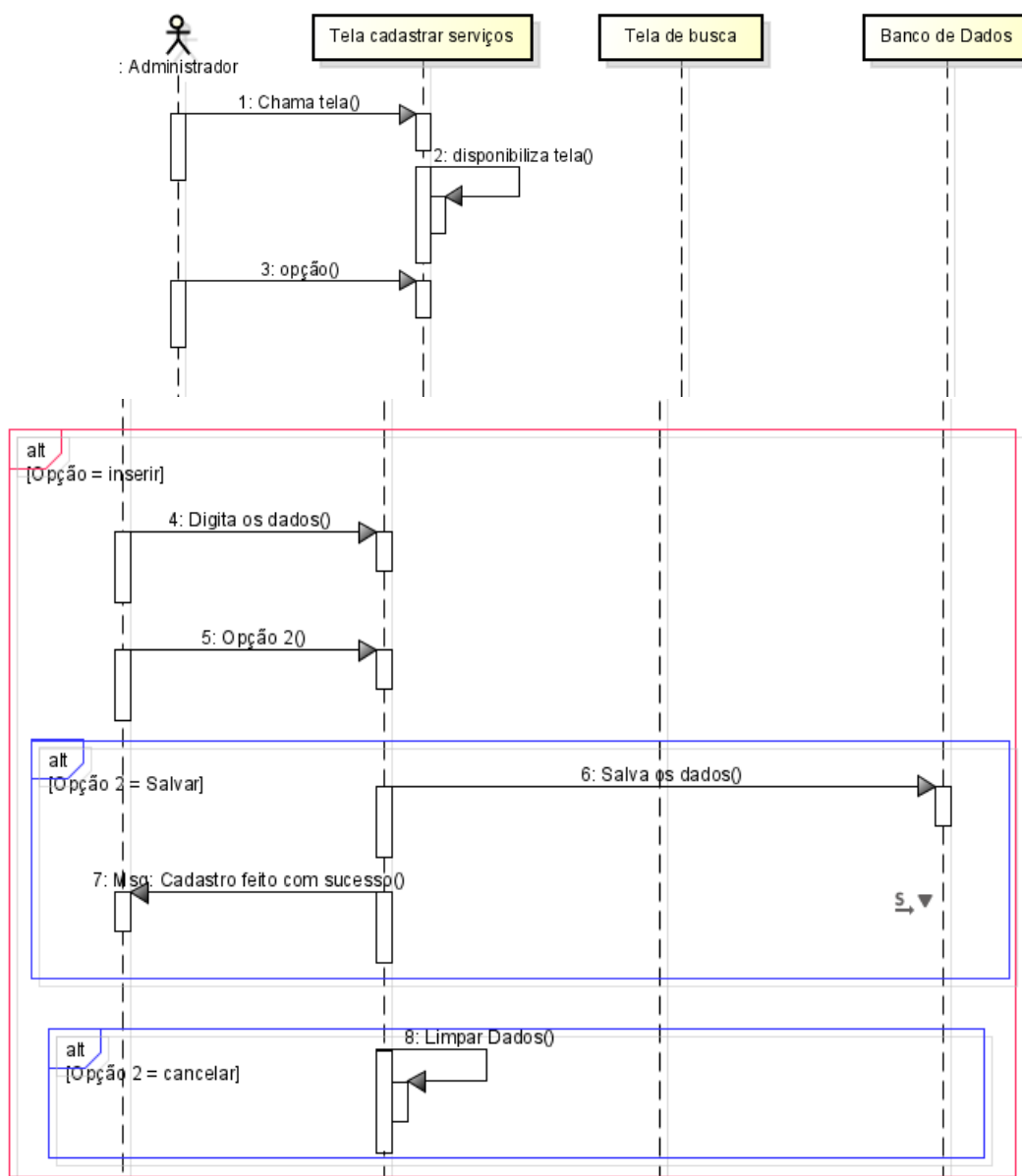
Fonte: Elaborada pelo autor

Figura 23 – Diagrama de Sequências sobre o Cadastro do sensor



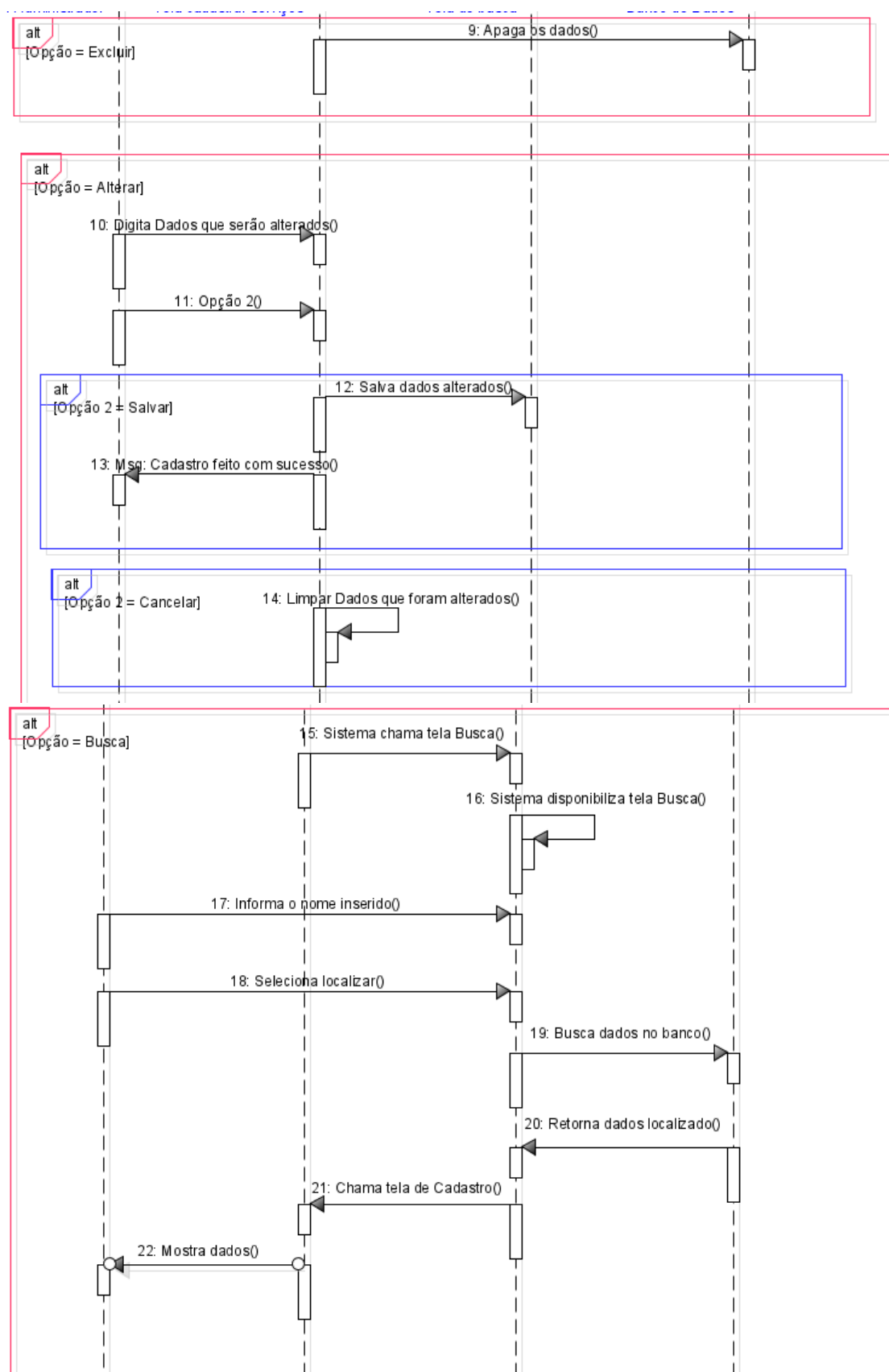
Fonte: Elaborada pelo autor

Figura 24 – Diagrama de Sequências sobre o Cadastro do Cliente



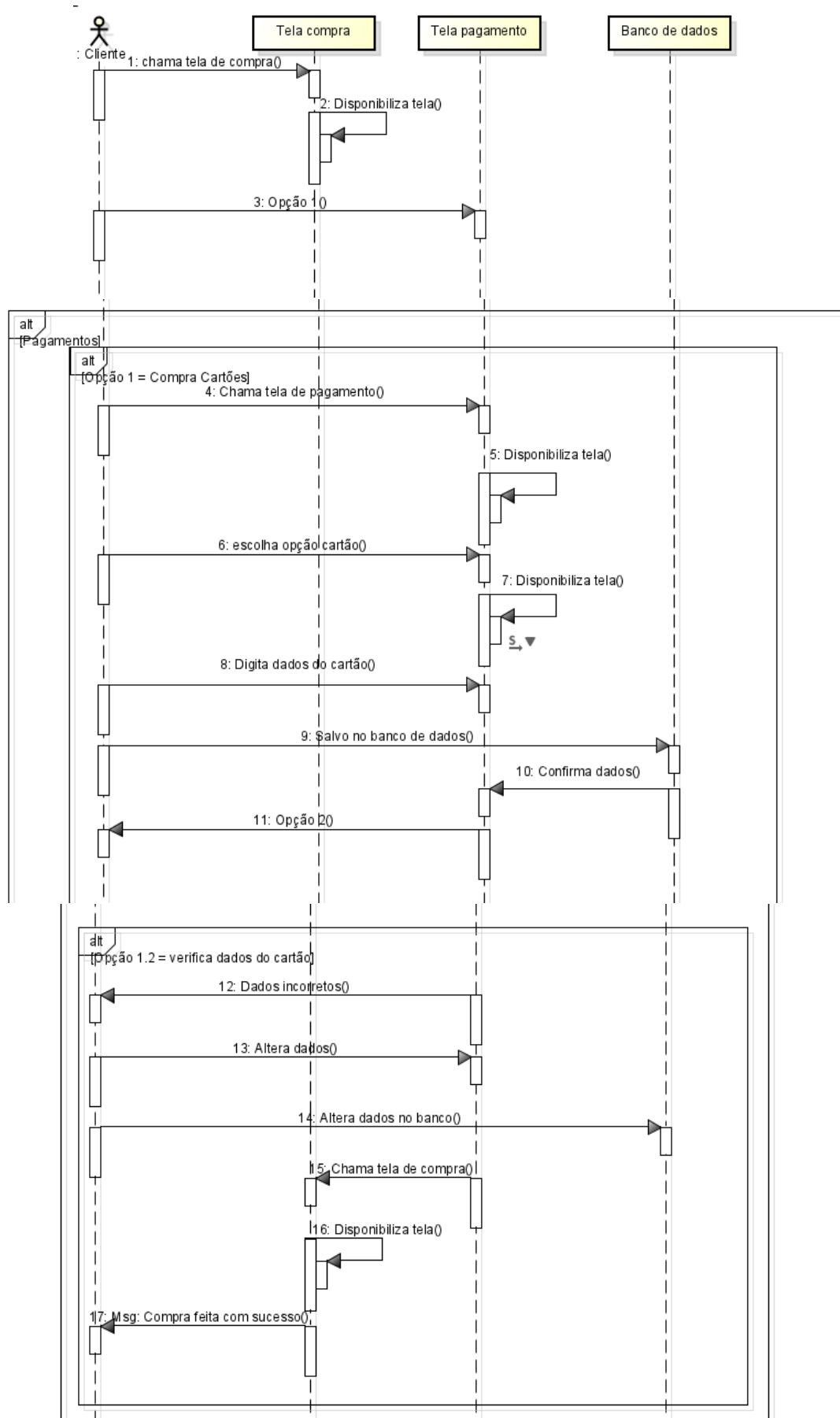
Fonte: Elaborada pelo autor

Figura 25 – Diagrama de Sequências sobre o Cadastro do Cliente



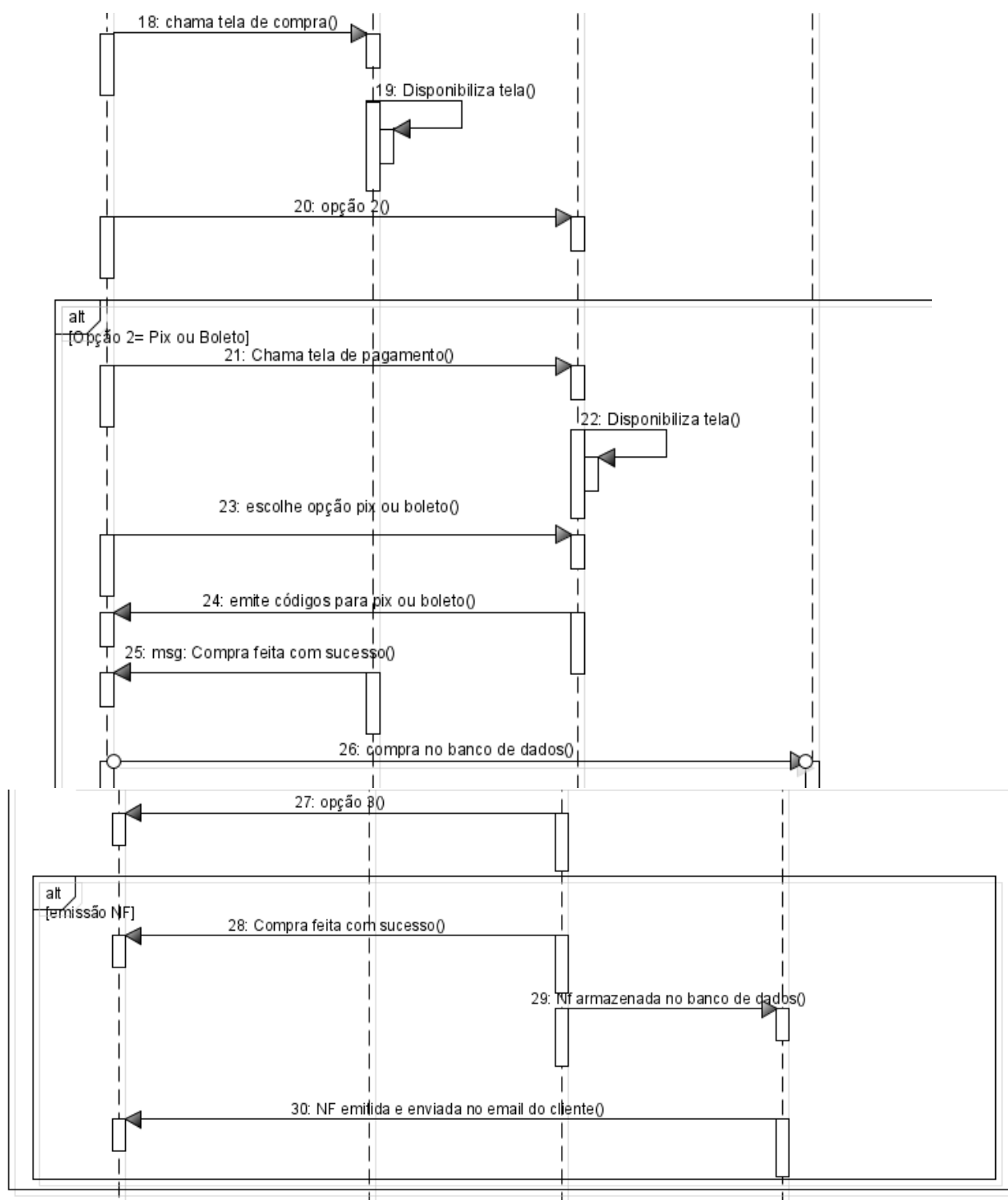
Fonte: Elaborada pelo autor

Figura 26 – Diagrama de Sequências sobre a Realização da compra do produto



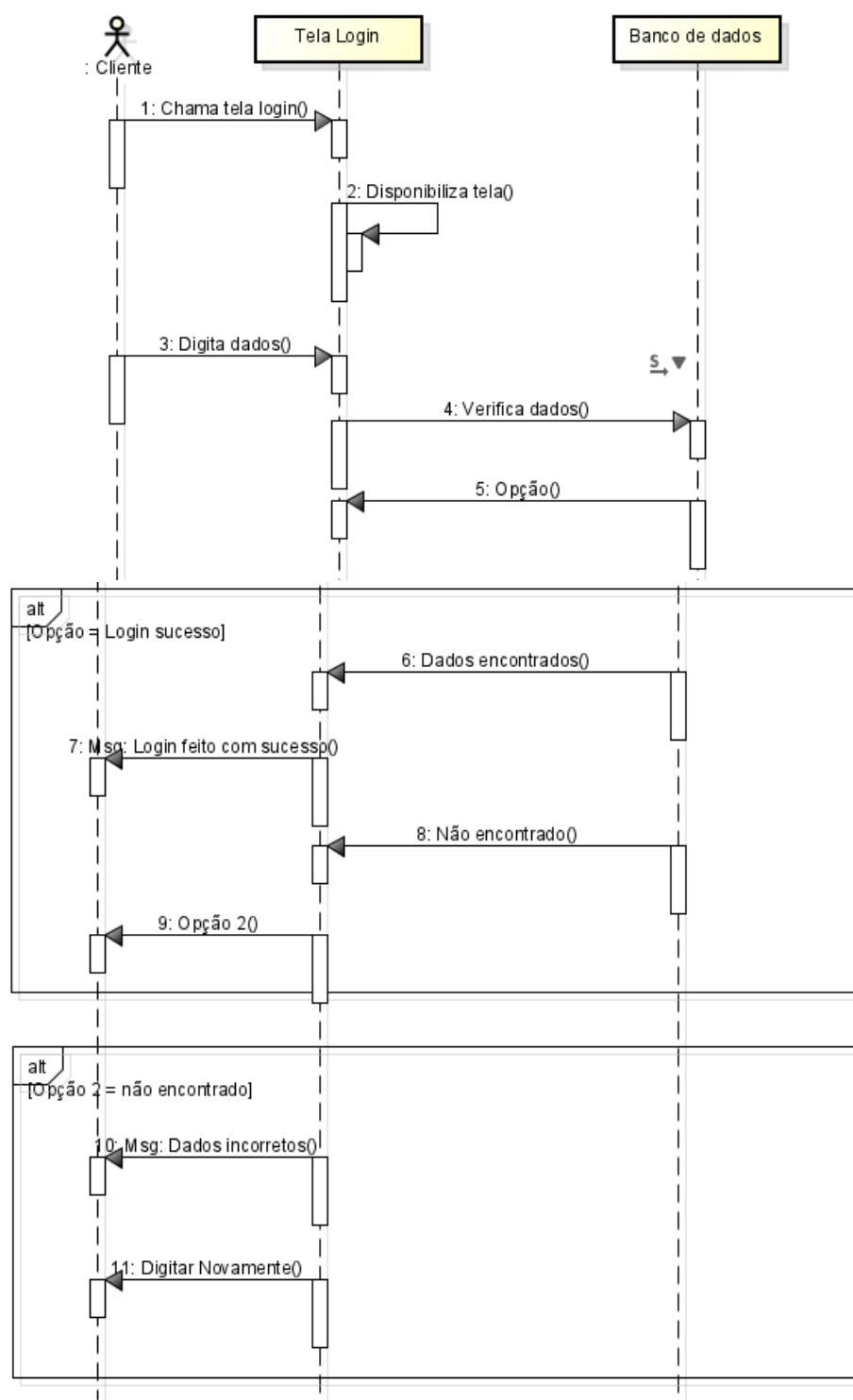
Fonte: Elaborada pelo autor

Figura 27 – Diagrama de Sequências sobre a Realização da compra do produto



Fonte: Elaborada pelo autor

Figura 28 – Diagrama de Sequências sobre a Realização da opção de Login



Fonte: Elaborada pelo autor

2 PROJETO DE SOFTWARE

Este capítulo aborda a exposição ordenada e pormenorizada do projeto NaturArt. Nele estão prescritos os modelos de banco de dados, a estrutura das tabelas no banco de dados e o projeto de interface.

2.1 Projeto de Interface

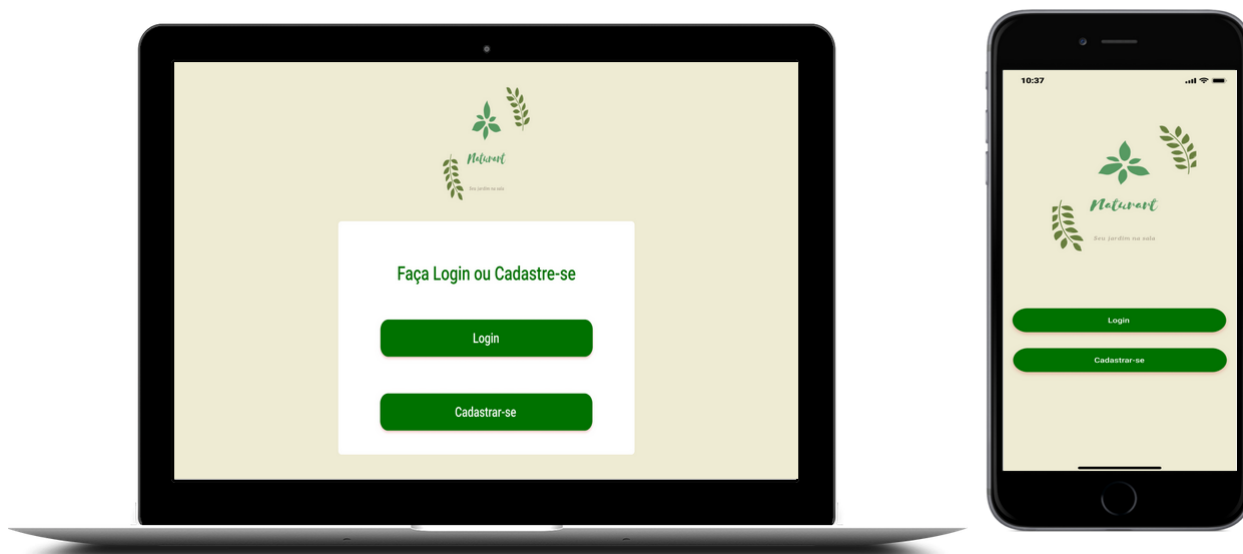
Esta sessão apresenta o protótipo das interfaces do sistema tanto desktop quanto mobile. Foram construídos utilizando a ferramenta de prototipação chamada Figma.

Figura 29 – Protótipo da tela de inicialização



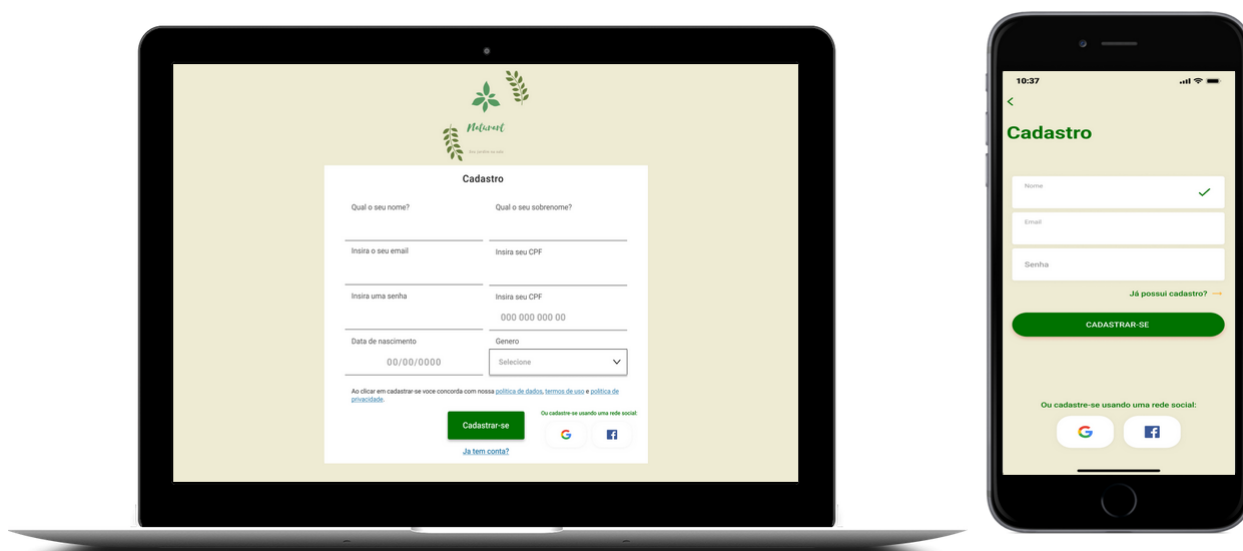
Fonte: Elaborada pelo autor

Figura 30 – Protótipo da tela de cadastro/login



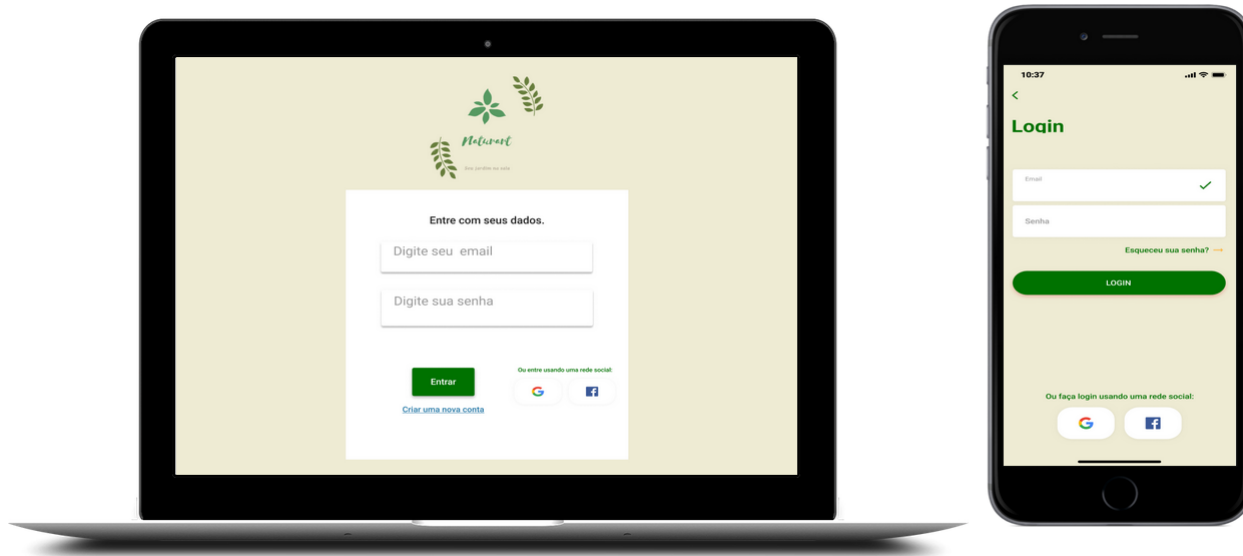
Fonte: Elaborada pelo autor

Figura 31 – Protótipo da tela de cadastro



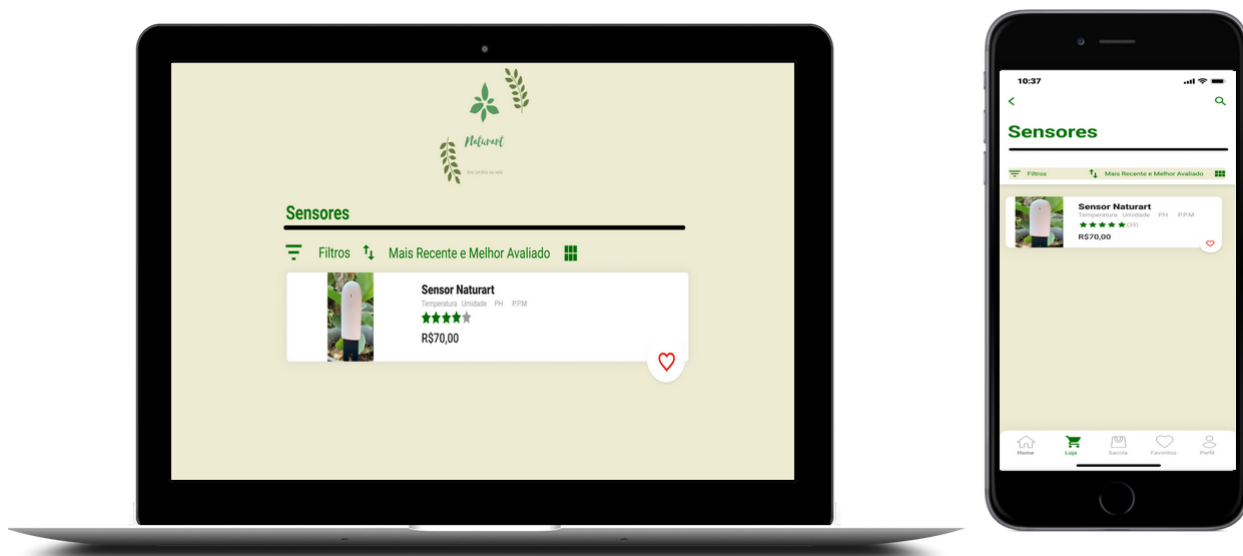
Fonte: Elaborada pelo autor

Figura 32 – Protótipo da tela de login



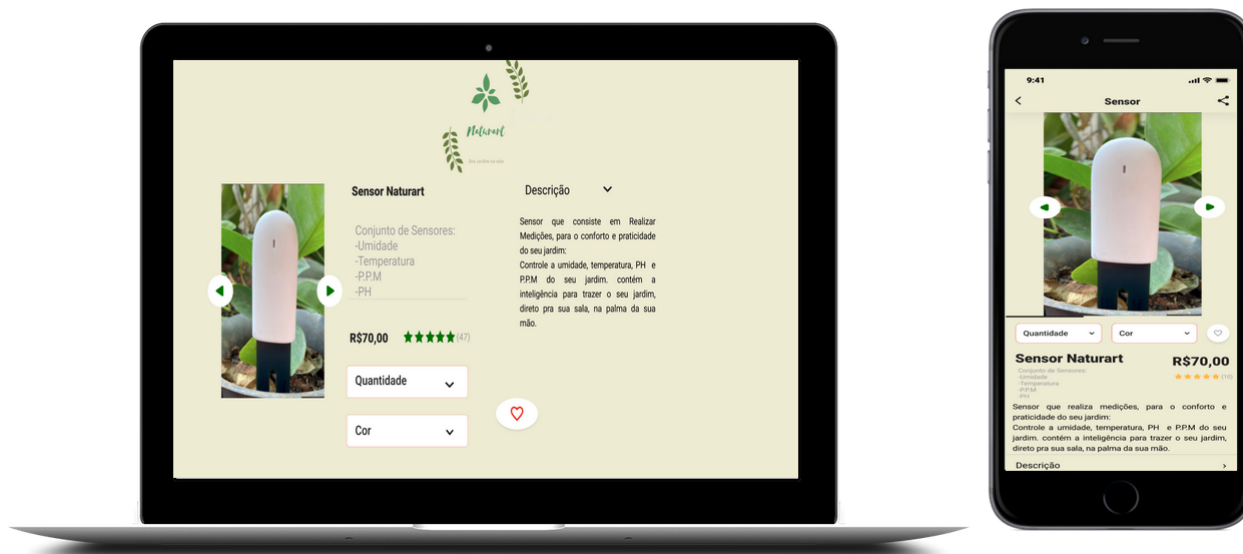
Fonte: Elaborada pelo autor

Figura 33 – Protótipo da tela de compra



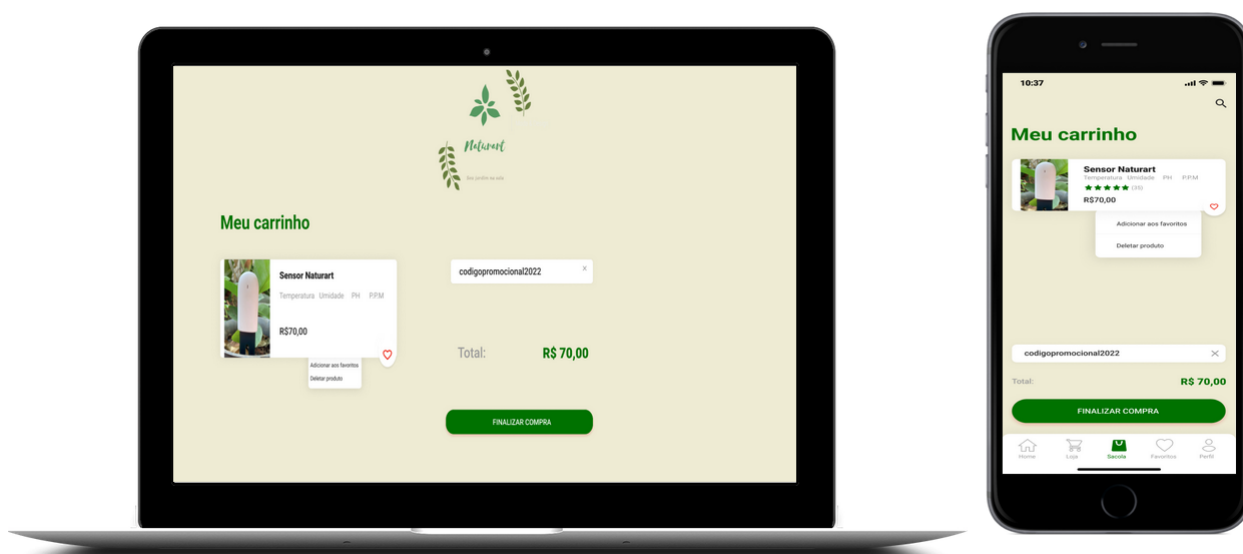
Fonte: Elaborada pelo autor

Figura 34 – Protótipo da tela do sensor



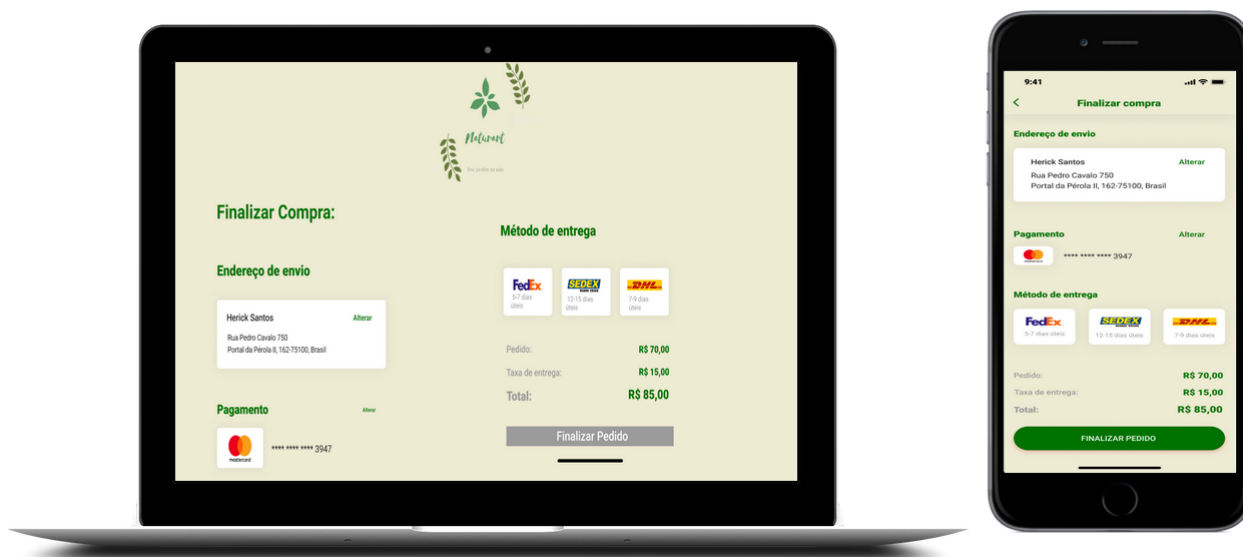
Fonte: Elaborada pelo autor

Figura 35 – Protótipo da tela de carrinho



Fonte: Elaborada pelo autor

Figura 36 – Protótipo da tela de finalização da compra



Fonte: Elaborada pelo autor

Figura 37 – Protótipo da tela de método de pagamento



Fonte: Elaborada pelo autor

Figura 38 – Protótipo da tela de visualização dos dados



Fonte: Elaborada pelo autor

Figura 39 – Protótipo da tela de obra de arte



Fonte: Elaborada pelo autor

2.2 Projeto de Dados

Para este projeto foi convencionado utilizar o MySQL como estrutura de banco de dados, com auxílio da IDE MySQL Workbench.

2.2.1 Estrutura das Tabelas no Banco de Dados

Foi convencionado que o nome dos objetos devem obedecer o que está definido na Tabela 5.

Tabela 5 – Convenção para Nome dos Objetos no Banco de Dados

Objeto	Padrão Adotado
Chave Primária	NomeDaTabela_PK
Chave Estrangeira	NomeDaTabela_NomeDaTabelaEstrangeira_FK_nn, onde nn é a sequência de ocorrência do par NomeDaTabela e NomeDaTabelaEstrangeira
Check	NomeDaTabela_CK_nn, onde nn é a sequência de checks da tabela
Chave Única	NomeDaTabela_UK_nn, onde nn é a sequência de chave única da tabela

Fonte: Elaborada pelo autor

Tabela 6 – Medicao

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
id	int	X	X		
dataMedicao	date	X			
valor	float	X			
idSensor	int	X		Sensor	id

Fonte: Elaborada pelo autor

Tabela 7 – Sensor

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
id	int	X	X		
tipo	varchar(50)	X			

Fonte: Elaborada pelo autor

Tabela 8 – Kit

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
idCliente	int	X		Cliente	id
idCarrinho	int	X		Carrinho	id

Fonte: Elaborada pelo autor

Tabela 9 – Carrinho

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
id	int	X	X		
idCliente	int	X		Cliente	id
dataCriacao	date	X			
quantidade	int	X			

Fonte: Elaborada pelo autor

Tabela 10 – Compra

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
id	int	X	X		
data	date	X			
valor	float	X			
numero	int	X		Cartao	numero
idCarrinho	int	X		Carrinho	id
idCliente	int	X		Cliente	id

Fonte: Elaborada pelo autor

Tabela 11 – Telefone

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
numero	int	X	X		
ddd	int	X			
codigo	int	X			
idCliente	int	X		Cliente	id

Fonte: Elaborada pelo autor

Tabela 12 – Cartao

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
numero	varchar(16)	X	X		
tipo	int	X			
agencia	varchar(50)	X			
idCliente	int	X		Cliente	id

Fonte: Elaborada pelo autor

Tabela 13 – Cliente

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
id	int	X	X		
nome	varchar(50)	X			
email	varchar(50)	X			
senha	varchar(50)	X			
cpf	varchar(11)	X			
dataNasc	date	X			
emailSec	varchar(50)	X			
cep	varchar(8)	X		Endereco	cep

Fonte: Elaborada pelo autor

Tabela 14 – Reclamacao

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
id	int	X	X		
data	date	X			
descricao	varchar(255)	X			
hora	date	X			
idCliente	int	X		Cliente	id
idCompra	int	X		Compra	id

Fonte: Elaborada pelo autor

Tabela 15 – CompraTroca

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
idCompra	int	X		Cliente	id
idTroca	int	X		Compra	id

Fonte: Elaborada pelo autor

Tabela 16 – Troca

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
id	int	X	X		
data	date	X			
descricao	varchar(255)	X			
hora	date	X			
idCliente	int	X		Cliente	id
idCompra	int	X		Compra	id

Fonte: Elaborada pelo autor

Tabela 17 – Endereco

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
cep	varchar(8)	X	X		
numero	int	X			
idCidade	varchar(50)	X		Cidade	nome
idBairro	varchar(50)	X		Bairro	nome
idLogradouro	int	X		Logradouro	nome

Fonte: Elaborada pelo autor

Tabela 18 – Cidade

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
nome	varchar(50)	X	X		
idEstado	int	X		Estado	nome

Fonte: Elaborada pelo autor

Tabela 19 – Estado

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
sigla	varchar(2)	X	X		
nome	varchar(50)	X			

Fonte: Elaborada pelo autor

Tabela 20 – Cidade

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
nome	varchar(50)	X	X		
idTipo	varchar(50)	X		Logradouro	tipo

Fonte: Elaborada pelo autor

Tabela 21 – Cidade

Campo	Tipo de Dado	Obrigatório?	Chave Primária?	Tabela	Campo
tipo	varchar(50)	X	X		

Fonte: Elaborada pelo autor