

UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE – UERN
FACULDADE DE CIÊNCIAS EXATAS E NATURAIS – FANAT
DEPARTAMENTO DE INFORMÁTICA – DI

Ermeson Diêgo Vieira da Rocha

Uso da Análise Orientada a Objeto para o desenvolvimento do Sistema Semear

MOSSORÓ - RN

2020

Ermeson Diêgo Vieira da Rocha

Uso da Análise Orientada a Objeto para o desenvolvimento do Sistema Semear

Monografia apresentada à Universidade do Estado do Rio Grande do Norte como um dos pré-requisitos para obtenção do grau de bacharel em Ciência da Computação, sob orientação do Profº. Dr. Maximiliano Araújo da Silva Lopes.

MOSSORÓ - RN

2020

© Todos os direitos estão reservados a Universidade do Estado do Rio Grande do Norte. O conteúdo desta obra é de inteira responsabilidade do(a) autor(a), sendo o mesmo, passível de sanções administrativas ou penais, caso sejam infringidas as leis que regulamentam a Propriedade Intelectual, respectivamente, Patentes: Lei nº 9.279/1996 e Direitos Autorais: Lei nº 9.610/1998. A mesma poderá servir de base literária para novas pesquisas, desde que a obra e seu(a) respectivo(a) autor(a) sejam devidamente citados e mencionados os seus créditos bibliográficos.

Catálogo da Publicação na Fonte.
Universidade do Estado do Rio Grande do Norte.

R672u Rocha, Ermeson Diêgo Vieira da
 Uso da análise orientada a objeto para o desenvolvimento do sistema semear. / Ermeson Diêgo Vieira da Rocha. - Mossoró/RN, 2020.
 55p.

 Orientador(a): Prof. Dr. Maximiliano Araújo da Silva Lopes.

 Monografia (Graduação em Ciência da Computação).
 Universidade do Estado do Rio Grande do Norte.

 1. Sistema Semear. I. Lopes, Maximiliano Araújo da Silva. II. Universidade do Estado do Rio Grande do Norte. III. Título.

Ermeson Diêgo Vieira da Rocha

Uso da Análise Orientada a Objeto para o desenvolvimento do Sistema Semear

Monografia apresentada como pré-requisito para obtenção do título de Bacharel em Ciência da Computação da Universidade do Estado do Rio Grande do Norte – UERN, submetida à aprovação da banca examinadora composta pelos seguintes membros:

Aprovado em: 03/12/2020

Banca Examinadora

Prof. Dr. Sebastião Emídio Alves Filho
Universidade do Estado do Rio Grande do Norte – UERN

Prof. Dr. Isaac de Lima Oliveira Filho
Universidade do Estado do Rio Grande do Norte – UERN

Prof. Dr. Maximiliano Araújo da Silva Lopes
Universidade do Estado do Rio Grande do Norte – UERN

Dedico este trabalho aos meus pais Lourdes e Levi (*in memoriam*), como também a todos os meus familiares que torceram para que meu objetivo fosse alcançado.

AGRADECIMENTOS

Agradeço a minha família por ter me dado apoio e sempre acreditar que seria possível chegar até o final do curso. Também agradeço aos colegas de trabalho da Secretaria de Agricultura do município de Mossoró, que sempre que possível me ajudaram em todo o decorrer do curso.

Quero também agradecer a todos os professores do Departamento de Informática da Universidade do Estado do Rio Grande do Norte, como também aos professores de outros cursos e departamentos, os quais contribuíram e foram todos muito importantes para minha formação acadêmica.

Por fim agradecer a todos os colegas e amigos que torceram para que este objetivo fosse alcançado.

“Se alguma coisa tem a mais remota chance de dar errado, certamente dará”.

Edward A. Murphy

RESUMO

O Programa Semear vinha enfrentando problemas para sua execução por ser executado sem a utilização de um sistema computacional. O Programa tem como objetivo auxiliar os pequenos agricultores da zona rural do município de Mossoró/RN com o corte de terra para plantio no período de inverno. No entanto, todo o processo de cadastro e controle da distribuição de ordens de abastecimento para os tratorista era realizado de forma manual. Com o objetivo de buscar soluções para os problemas enfrentados, foram utilizados os conceitos da Análise Orientada a Objetos para o desenvolvimento de sistemas, como também, da Linguagem de Modelagem Unificada – UML, onde foi possível representar através de diagramas todo o processo para execução do Programa. Para o desenvolvimento do sistema fez-se necessário à utilização de ferramentas bastante importantes e conhecidas no mundo do desenvolvimento de sistemas. As ferramentas utilizadas foram a linguagem de programação Java juntamente com o Sistema Gerenciador de Banco de Dados PostgreSQL, além do IDE NetBeans para acelerar o desenvolvimento da interface do sistema. Ao final os resultados mostraram que o sistema melhorou substancialmente todo o processo de cadastro como também o controle de ordens de abastecimento mesmo este sendo utilizado de forma parcial.

Palavras-chave: Análise Orientada a Objetos, Java, UML.

ABSTRACT

The Semear Program had been facing problems for its execution because it was executed without the use of a computer system. The Program aims to help small farmers in the rural area of the municipality of Mossoró / RN with the cutting of land for planting in the winter period. However, the entire process of registering and controlling the distribution of supply orders to the tractor drivers was carried out manually. In order to seek solutions to the problems faced, the concepts of Object Oriented Analysis were used for the development of systems, as well as the Unified Modeling Language - UML, where it was possible to represent through diagrams the entire process for the execution of the Program. For the development of the system it was necessary to use very important tools and known in the world of systems development. The tools used were the Java programming language together with the PostgreSQL Database Manager System, in addition to the NetBeans IDE to speed up the development of the system interface. In the end, the results showed that the system substantially improved the entire registration process as well as the control of supply orders even though it was partially used.

Keywords: Object Oriented Analysis, Java, UML.

LISTA DE SIGLAS

SEADRU	Secretaria Municipal da Agricultura e Desenvolvimento Rural
UML	Linguagem de Modelagem Unificada
CPF	Cadastro de Pessoas Físicas
HTML	Hyper Text Markup Language
PHP	Personal Home Page
IDE	Integrated Development Environment
GUI	Graphical User Interface
JDK	Java Development Kit
SQL	Structured Query Language
ODBC	Open Database Connectivity
JSP	Java Server Pages
BSD	Berkeley Software Distribution
SGBD	Sistema de Gerenciamento de Banco de Dados
OMG	Object Management Group

LISTA DE FIGURAS

Figura 1: Tela de cadastro do beneficiário (Clipper).....	16
Figura 2: Cadastro de beneficiário (Microsoft Office Excel).....	17
Figura 3: NetBeans IDE 8.2.	24
Figura 4: Interface do PostgreSQL 9.4.....	25
Figura 5: Diagrama de caso de uso – Programa Semear.....	27
Figura 6: Diagrama de classes – Programa Semear.....	29
Figura 7: Diagrama de transição de estados – Consultar histórico.....	30
Figura 8: Tela de cadastro do beneficiário (Java).....	32
Figura 9: Tela de consulta histórico do beneficiário.....	33
Figura 10: Identificação de cadastro já existente.....	33
Figura 11: Verificação de validade do CPF.	34
Figura 12: Impedimento de cadastro do cônjuge como beneficiário.	34
Figura 13: Impedimento de exclusão do tratorista.	35
Figura 14: Tratorista sem beneficiários vinculados.....	35

LISTA DE TABELAS

Tabela 1: Caso de uso cadastrar beneficiário.....	29
--	----

SUMÁRIO

1 INTRODUÇÃO	13
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 PROGRAMA SEMEAR.....	15
2.2 ANÁLISE ORIENTADA A OBJETOS	18
2.2.1 Modelos	20
2.2.2 Classes	20
2.2.3 Atributo.....	20
2.2.4 Métodos	21
2.2.5 Argumentos.....	21
2.2.6 Objeto	21
2.2.7 Herança	22
2.2.8 Interface	22
2.3 FERRAMENTAS UTILIZADAS.....	22
2.3.1 Java	23
2.3.2 IDE NetBeans.....	24
2.3.3 PostgreSQL 9.4	25
3 DESENVOLVIMENTO DO SISTEMA	26
3.1 ENGENHARIA DE REQUISITOS	26
3.1.1 Diagrama de caso de uso	27
3.1.2 Caso de uso	28
3.1.3 Diagrama de classes.....	29
3.1.4 Diagrama de transição de estados.....	30
3.2 IMPLEMENTAÇÃO.....	31
3.3 VALIDAÇÃO	32
3.4 RESULTADOS	35
4 CONCLUSÃO	37
REFERÊNCIAS	38
APÊNDICE A – Casos de uso Sistema Semear	40

1 INTRODUÇÃO

Atualmente, o desenvolvimento de novas tecnologias está acontecendo de forma muito rápida em todo o mundo. Em consequência existe um desafio constante na área de engenharia de software, que é o de melhorar o processo de desenvolvimento.

Cabe ressaltar que para o processo de desenvolvimento de software um dos pontos mais importante é entender o problema existente e encontrar as melhores soluções computacionais para a resolução deste. No entanto, para se chegar as melhores soluções necessita-se da escolha correta referente aos métodos, técnicas e ferramentas que serão utilizados.

Portanto, o uso da análise orientada a objetos é um mecanismo utilizado no mundo do desenvolvimento de software que diante da aplicação dos seus conceitos, permite ao desenvolvedor chegar as melhores soluções computacionais. Aliando a orientação a objetos à Linguagem de Modelagem Unificada – UML, chegou-se a um detalhamento do comportamento de um determinado sistema antes mesmo deste ser desenvolvido em uma linguagem de programação para seu uso final.

Com isto percebe-se que devemos ter certa atenção devido à importância da relação citada anteriormente, tendo em vista que as soluções encontradas com os mecanismos utilizados, poderão ser desenvolvidas posteriormente por qualquer linguagem de programação que use a orientação a objetos.

Utilizando-se dos conceitos citados anteriormente, este trabalho tem como objetivo o desenvolvimento de um sistema para ser usado durante todo o processo de execução do Programa Semear, visando otimizar todo o processo de cadastro e controle da distribuição de ordens de abastecimento.

Em suma, os processos envolvidos para a execução do Programa Semear inicia-se com o cadastramento de representantes ou presidentes de associações e comunidades. Para os passos seguintes outros cadastros são realizados, como por exemplo os beneficiários, cônjuge, tratorista entre outros. Esses processos vinham sendo realizados todos de forma manual fazendo com que o tempo gasto para a execução de todos os procedimentos para o andamento do Programa fossem demorados.

Por sua vez, com a utilização de um sistema computacional o tempo para execução do Programa irá diminuir consideravelmente, tendo em vista que sua utilização otimizará todos os procedimentos para cadastro como também um melhor controle referente a ordens de abastecimento para os tratoristas.

Assim sendo, com o uso do sistema diversos problemas foram solucionados como, por exemplo, a detecção de duplo cadastro para um único beneficiário, que não é permitido conforme critérios do Programa Semear.

Nesse sentido, o desenvolvimento deste sistema passou por uma situação especial que teve que ser tratada com bastante atenção. Como o Programa Semear passou por dois momentos distintos com relação aos cadastros realizados, existiu a necessidade de migração dos dados para o novo sistema, onde no capítulo 3 deste trabalho será colocado como ocorreu este procedimento.

Em linhas gerais, o objetivo do desenvolvimento de um sistema aplicando os conceitos acima, este trabalho está organizado da seguinte forma: No Capítulo 2 é apresentada a fundamentação teórica sobre o Programa Semear e a análise orientada a objetos. No Capítulo 3 está detalhado o processo de execução do Programa Semear, como também, a aplicação da engenharia de requisitos para o desenvolvimento do sistema. Ainda dentro do Capítulo 3 é exposto como ocorreu a implementação diante dos problemas que o Programa Semear apresentava. Ao final o Capítulo 4 apresenta as considerações finais e a perspectiva para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Adentrando nas mudanças no mundo tecnológico e a rapidez das informações o acesso está cada vez mais presente na vida das pessoas. Em contrapartida, podemos perceber que tais informações precisam chegar a quem tem o interesse por elas cada vez mais rápida. No entanto, muitas empresas precisam se adequar para atender as novas exigências do referido avanço e acabam tendo que investir em tecnologias que permitam um melhor atendimento a tais exigências.

Haja vista que, no serviço público não é diferente, e a população que precisa de atendimento em algum órgão, seja ele federal, estadual ou municipal, tem certa exigência para que o atendimento seja realizado de forma rápida.

Diante desse cenário, este capítulo mostrará o que é o Programa Semear e seus principais problemas, como também, a teoria para a busca de soluções computacionais para que o citado Programa através da análise orientada a objetos.

2.1 PROGRAMA SEMEAR

O “Programa Semear” tem por finalidade atender agricultores familiares inseridos em comunidades rurais ou projetos de assentamento, auxiliando-os na execução dos trabalhos de preparo do solo para o plantio na zona rural do município de Mossoró/RN durante o período de inverno na região.

O Semear teve início no final da década de 90 com o objetivo de auxiliar os pequenos agricultores da zona rural do município de Mossoró/RN com o corte de terra para o plantio, principalmente de milho e feijão, para o sustento de suas famílias.

No início do Semear, como era chamado na época, a forma como eram cadastrados os beneficiários atendia, teoricamente, ao objetivo do Semear. Naquele momento não era exigido um documento padrão e portanto, diversos documentos eram apresentados para a realização do cadastro, como por exemplo: carteira de trabalho, título de eleitor, documento de identidade, CPF dentre outros, e em algumas situações nem documento era exigido, constando apenas o nome do beneficiário no cadastro.

Com o decorrer do tempo e diante da necessidade de emissão de comprovante da participação dos beneficiários com o corte de terra, percebeu-se a

necessidade de exigir um documento padrão para identificar o beneficiário. Portanto, no ano de 2005 passou-se a exigir a obrigatoriedade de constar no cadastro o número do CPF deste beneficiário. A partir do ano mencionado foi possível constatar que alguns beneficiários estavam sendo beneficiado mais de uma vez, fora dos critérios do Semear, que permite apenas um cadastro por beneficiário para cada ano de execução.

No entanto, achando que o Semear estava caminhando para que este fosse se aprimorando para evitar possíveis fraudes e desperdício do dinheiro público, no ano de 2010, 01 (um) ano após a mudança de gestores a frente da Prefeitura Municipal de Mossoró e por consequência mudança também na Secretaria de Agricultura do município, responsável por executar o Semear, foi suspensa a utilização do sistema de cadastro, o qual foi desenvolvido na linguagem Clipper (Figura 1).

Figura 1: Tela de cadastro do beneficiário (Clipper).

SEMEAR_exe

GER. EXEC. AGRICULTURA, ABAST. E REC. HIDRICOS 28/10/2020 [12:33:57 hs]
 Sistema de Controle de Semeação Usuário: AGRICULTURA

1. BENEFICIÁRIOS

Registro | CPF: |
 Nome.....
 Apelido... RG: Tipo:

Inclui F3, F4, F9 Conf

A N O	Deseja Corte?	Horas	Trator	Banco Semen tes?	*** ÁREAS PLANTADAS EM HECTARES ***						
					AT	M	F	A	S	O	Cód./ Localidade

Leg. → AT= Área Total | M= Milho | F= Feijão | A= Algodão | S= Sorgo | O= Outros

Fonte: O autor (2020).

Esta suspensão tinha como objetivo de melhor se adequar aos requisitos do Semear. No entanto, com a suspensão da utilização do referido sistema de cadastro, os funcionários passaram a realizar o cadastro em folha de papel e posteriormente passado para planilhas do Microsoft Office Excel (Figura 2).

Figura 2: Cadastro de beneficiário (Microsoft Office Excel).

	A	B	C	D
1		P.A PAULO FREIRE 2020		
2		NOME	CPF	
3		AIRTON FERNANDES	182.955.764-53	
4		ALCIDES DE OLIVEIRA PEREIRA	182.517.304-49	
5		ANA MARIA	029.753.634-60	
6		ANTONIA ADAIRES DA SILVA	025.418.384-09	
7		ANTONIA FRANCISCA CASSIANO	011.967.034-85	
8		ANTONIA GILVANIA PEREIRA LOPES	012.278.624-66	
9		ANTONIO MARCOS DAS CHAGAS	007.852.544-63	
10		ANTONIO PEREIRA DE LIMA NETO	655.523.504-72	
11		EDILSON NOGUEIRA	673.079.734-72	
12		EDIMUNDO ARAUJO MONTEIRO	423.472.974-49	
13		ELIQUESONIA PEREIRA DA SILVA	074.817.874-09	
14		EVALDO CARLOS BATISTA	392.980.434-49	
15		FELIX SOARES DA SILVA	785.435.174-00	
16		FRANCILEIDE LUCIMAR DE OLIVEIRA FERREIRA	010.388.884-50	
17		FRANCISCA INES MARTINS DE SOUZA	336.109.594-87	
18		FRANCISCA IONEIDE DA SILVA	565.218.634-87	
19		FRANCISCO BEZERRA DE BRITO	653.398.354-78	
20		FRANCISCO CAETANO DE ANDRADE	316.497.184-20	
21		FRANCISCO CHAGAS DOS SANTOS	638.403.084-53	
22		FRANCISCO RUFINO DA SILVA	027.000.094-10	
23		GILMAR JOAQUIM MONTEIRO	875.883.074-04	
24		GILVANETE NONATA DA SILVA	001.476.503-99	
25		GIVALDO MEDEIROS NOGUEIRA	048.187.754-10	

Fonte: O autor (2020).

Outro ponto bastante importante é com relação ao controle da distribuição do combustível óleo diesel, onde este sempre foi usado como forma de pagamento ao serviço executado pelos tratoristas cadastrados no Semear. Todo controle do óleo diesel sempre foi feito utilizando planilhas do Microsoft Office Excel, tendo que ser atualizada manualmente a cada alteração na quantidade de beneficiários que cada tratorista teria que atender com o serviço de corte de terra, se tornando numa situação muito propícia para o erro na liberação de ordem de abastecimento para os tratoristas, podendo ocasionar pagamento inferior ou superior ao total do que se teria direito.

Diante da não utilização de um sistema adequado durante o período de 2010 à 2020, ficou praticamente impossível identificar qualquer irregularidade durante a realização do cadastro dos beneficiários, como também ter um controle a respeito da liberação das ordens de abastecimento para os tratoristas.

Regulamentado através da Lei Nº 3.301, de 21 de agosto de 2015, o “Programa Semear” tem caráter contínuo e é executado no primeiro semestre de cada ano, observado o período de chuvas no município de Mossoró/RN. (MOSSORÓ, 2015, p. 1).

O Programa disponibiliza ao beneficiário 01 (uma) hora de corte de terra, por família beneficiada para fins exclusivos de auxiliar no corte de terra, onde o tratorista recebe por cada hora a quantidade de 22 (vinte e dois) litros de óleo diesel como forma de pagamento pelo serviço executado. (MOSSORÓ, 2015, p. 1).

De acordo com a Lei citada anteriormente a Secretaria de Agricultura do município de Mossoró/RN deverá apresentar plano de trabalho em até 60 (sessenta) dias antes de ser iniciada a execução das ações integrantes do programa.

Para o cumprimento do plano elaborado, no mês de dezembro do ano anterior a execução do Programa Semear, os presidentes de associações ou representantes dos assentamentos e comunidades rurais são convocados à sede da Secretaria de Agricultura do município de Mossoró/RN para preenchimento de um formulário disponibilizado pela referida secretaria. Após o preenchimento deste formulário os presidentes/representantes recebem outro formulário para relacionarem quais serão os beneficiários com o corte de terra. Os beneficiários do corte de terra de cada assentamento/comunidade escolhem quem será o tratorista responsável por realizar o trabalho do corte de terra, o qual deverá assinar um contrato para a execução do serviço. Depois de devolvida a relação dos beneficiários e indicação do tratorista, a Secretaria irá realizar o cadastramento de todos os envolvidos no Programa Semear (tratoristas, representantes de assentamentos/comunidades e beneficiários).

Após a realização de todos os cadastros os tratoristas começam a execução do serviço de corte de terra para os beneficiários.

Na medida em que o serviço do corte de terra vai sendo executado o tratorista irá recebendo o pagamento através de Ordem de Abastecimento (combustível óleo diesel) como forma de pagamento pelo serviço.

Com o objetivo de otimizar todos os processos de cadastro e controle da distribuição de ordens de abastecimento, o propósito deste trabalho é demonstrar o desenvolvimento de um sistema para solucionar os problemas apresentados, como também identificar irregularidades que não obedeçam aos critérios e objetivos do Programa Semear.

2.2 ANÁLISE ORIENTADA A OBJETOS

Para garantir uma boa solução aos problemas, conforme colocado acima, vamos partir dos conceitos referente à análise orientada a objetos, por esta ter extrema importância durante o processo de desenvolvimento de sistemas.

O rápido crescimento da capacidade computacional das máquinas resultou na demanda por sistemas de software cada vez mais complexos, que tirassem proveito de tal capacidade. Por sua vez, o surgimento desses sistemas mais complexos resultou na necessidade de reavaliação da forma de se desenvolver sistemas. Conseqüentemente, desde o aparecimento do primeiro computador até os dias de hoje, as técnicas utilizadas para a construção de sistemas computacionais tem evoluído de forma impressionante, notavelmente no que tange à modelagem de sistemas. (BEZERRA, 2007, p. 11).

No que se refere às técnicas para modelagem de sistemas, no início da década de 90 surge o paradigma de modelagem, a análise orientada a objetos. No fim da década de 90 o paradigma da orientação a objetos atinge sua maturidade e nesse período surge a Linguagem de Modelagem Unificada – UML. Com isso, entre os anos de 1990 e 1996 surgem várias propostas de técnicas de modelagem orientada a objetos.

No ano de 1997, a UML foi aprovada como padrão pelo Object Management Group – OMG a partir da unificação das muitas linguagens gráficas de modelagem orientada a objetos, onde foram aproveitadas as melhores características de cada uma das linguagens já existentes.

A UML é uma linguagem visual para modelar sistemas orientados a objetos. Isso quer dizer que a UML é uma linguagem constituída de elementos gráficos (visuais) utilizados na modelagem que permitem representar os conceitos do paradigma de orientação a objetos. Através dos elementos gráficos definidos nesta linguagem pode-se construir diagramas que representam diversas perspectivas de um sistema. (BEZERRA, 2007, p. 14).

Segundo Pressman (2011, p. 727), de acordo com o padrão atual, que é a UML 2.0, esta fornece 13 diferentes diagramas para uso na modelagem software.

No capítulo 3 serão mostrados alguns desses diagramas para comprovar a importância do uso desses mecanismos para o desenvolvimento de sistemas.

Para o processo de desenvolvimento de sistemas alguns passos devem ser seguidos, e um deles é a análise de requisitos. De acordo com Gonçalves e Cortés (2015, p. 36), na fase de análise, as especificações de requisitos são estudadas para que seja feito o detalhamento e refinamento destas através de modelos, realizando assim uma aproximação em direção à solução final.

De acordo com Rubira e Brito (2009, p. 64) a análise orientada a objetos “Modela o mundo real de tal modo que ele possa ser compreendido. Durante a análise, a ênfase está em encontrar e descrever as entidades do domínio do problema que sejam relevantes para o sistema que se pretende construir.”

Um dos desafios de projetar em orientação a objetos é representar o sistema a partir de uma perspectiva global (Objetivo da Análise). Podemos dizer que o que se faz na fase de análise é uma espécie de tradução das especificações dos requisitos, que se encontram na linguagem do cliente, para uma representação que use uma linguagem do desenvolvedor. (GONÇALVES; CORTÉS, 2015, p. 36).

“O Paradigma Orientado a Objeto (POO) tem como principal característica uma melhor e maior expressividade das necessidades do nosso dia a dia.” (CARVALHO, 2016, p. 5).

Diante do exposto serão apresentados a seguir conceitos fundamentais para uma melhor compreensão a respeito da análise orientada a objetos.

2.2.1 Modelos

A representação de algo que serve para ser reproduzido é o conceito mais básico referente a modelos e que nos leva ao seguinte conceito a partir do que diz Santos (2001, p. 3) em que: Modelos são representações simplificadas de objetos, pessoas, itens, tarefas, processos, conceitos, ideias, etc., usados comumente por pessoas no seu dia-a-dia, independente do uso de computadores.

2.2.2 Classes

Podemos entender classes como sendo a forma mais básica de definir uma única vez como os objetos serão criados a partir desta classe.

Classes são estruturas das linguagens de programação orientadas a objetos para conter, para determinado modelo, os dados que devem ser representados e as operações que devem ser efetuadas com estes dados. Cada classe deve ter um nome que seja facilmente associável ao modelo que a classe representa. (SANTOS, 2001, p. 15).

Uma classe é um modelo que define as variáveis e os métodos comuns a todos os objetos do mesmo tipo. Percebe-se com isso que se deve ter um cuidado todo especial para que erros não se propagem a partir de uma elaboração inadequada desses modelos.

2.2.3 Atributo

Atributo pode ser definido como as variáveis/campos que guardam valores conforme características que os objetos podem possuir.

“Atributo é o elemento de uma classe, responsável por definir sua estrutura de dados. O conjunto destes será responsável por representar suas características e farão parte dos objetos criados a partir da classe.” (CARVALHO, 2016, p. 37).

Essa definição deixa bem claro que os atributos devem ser definidos dentro da classe. Devido a isso, são responsáveis por definir sua estrutura de dados. É a partir do uso de atributos que será possível caracterizar (detalhar) as classes, sendo possível representar fielmente uma entidade do mundo real. (CARVALHO, 2016, p. 37).

2.2.4 Métodos

Os métodos são responsáveis por eventos realizados que possibilitam ao objeto realizar ações. São definidos na declaração de uma classe e definem o comportamento (operações) dos objetos daquela classe. Métodos são geralmente chamados ou executados explicitamente a partir de outros trechos de código na classe que o contém ou a partir de outras classes.

Método é uma porção de código (sub-rotina) que é disponibilizada pela classe. Este é executado quando é feita uma requisição a ele. Um método serve para identificar quais serviços, ações, que a classe oferece. Eles são responsáveis por definir e realizar um determinado comportamento. (CARVALHO, 2016, p. 41)

2.2.5 Argumentos

Os argumentos são variáveis especificadas entre parênteses, ou seja, são informações adicionais para a execução de um método quando acontece uma chamada a este.

A partir da criação de métodos, estes de acordo com Santos (2001, p. 15), podem opcionalmente receber argumentos para métodos na forma de valores de tipos nativos de dados ou referências à instâncias de classes.

2.2.6 Objeto

Os objetos são características que foram definidas pelas classes. Neles é permitido instanciar objetos da classe para inicializar os atributos e invocar os métodos.

Na Orientação a Objetos, os objetos do mundo real são modelados e representados no mundo computacional, ou seja, dentro do sistema, por meio de objetos de software.

Um objeto ou instância é uma materialização da classe, e assim pode ser usado para representar dados e executar operações. Para que os objetos ou instâncias possam ser manipulados, é necessária a criação de referências a estes objetos, que são basicamente variáveis do "tipo" da classe. (SANTOS, 2001, p. 15)

2.2.7 Herança

Podemos entender herança como sendo a criação de novas classes a partir de outras previamente criadas. Essas novas classes são chamadas de subclasses e as classes já existentes, que deram origem às subclasses, são chamadas de superclasses.

“O conceito de herança nada mais é do que uma possibilidade de representar algo que já existe no mundo real.” (CARVALHO, 2016, p. 80).

De forma mais simples, Horstmann e Cornell (2010, p. 81) diz que a ideia por trás de herança é que você pode criar novas classes com base nas classes existentes. Ao herdar a partir de uma classe existente, reutiliza (ou herda) os métodos e campos e adiciona novos métodos e campos para adaptar sua nova classe a novas situações.

2.2.8 Interface

Uma interface é similar a um contrato, através dela podemos especificar quais métodos as classes que implementam esta interface são obrigados a implementar.

“Interface define um contrato que deve ser seguido pela classe que a implementa. Quando uma classe implementa uma interface, ela se compromete a realizar todos os comportamentos que a interface disponibiliza.” (CARVALHO, 2016, p. 117).

Outra definição de acordo com Horstmann e Cornell (2010, p. 81) é que interface é um modo de descrever o que as classes devem fazer, sem especificar como devem fazer.

Diante dos conceitos apresentados, percebemos que para darmos início um projeto orientado a objetos é entender bem o que se deseja desenvolver. O bom entendimento levará a uma melhor qualidade e evita estar sempre corrigindo possíveis erros devido ao mau entendimento do problema, podendo comprometer a qualidade do trabalho desenvolvido.

2.3 FERRAMENTAS UTILIZADAS

Para a elaboração deste trabalho fez-se necessário à utilização de ferramentas para desenvolvimento de sistemas.

A seguir serão mostrados os motivos para a escolha das ferramentas utilizadas para o desenvolvimento do sistema em questão.

2.3.1 Java

Java é uma linguagem de programação bastante conhecida na área de desenvolvimento de sistemas, desde os iniciantes até os profissionais mais experientes.

“O Java é a base para praticamente todos os tipos de aplicações em rede e é o padrão global para o desenvolvimento e distribuição de aplicações móveis e incorporadas, jogos, conteúdo baseado na web e softwares corporativos.” (JAVA, 2020).

“De laptops a datacenters, consoles de games a supercomputadores científicos, telefones celulares à Internet, o Java está em todos os lugares!” (JAVA, 2020).

O Java foi projetado para permitir o desenvolvimento de aplicações portáteis de alto desempenho para a mais ampla variedade possível de plataformas de computação. Ao disponibilizar aplicações entre ambientes heterogêneos, as empresas podem fornecer mais serviços e aumentar a produtividade, a comunicação e a colaboração do usuário final.

Algumas razões da escolha da linguagem Java:

Java é obrigatoriamente orientada a objetos. Algumas linguagens permitem que objetos e variáveis existam em diversos pontos de um programa, como se estivessem desatreladas de qualquer estrutura.

Java é simples. A estrutura de programas e classes em Java segue a organização de linguagens tradicionais como C e C++, mas sem elementos que tornam programas e programação mais complexos.

Java é portátil. O código-fonte de um programa ou classe em Java pode ser compilado em qualquer computador, usando qualquer sistema operacional, contanto que este tenha uma máquina virtual Java adequada.

Java é gratuita. A máquina virtual Java, mencionada acima, está à disposição para cópia no site da Sun e em vários outros. Compiladores simples, de linha de comando (sem interfaces visuais elaboradas) fazem parte do JDK, o ambiente de desenvolvimento gratuito de Java.

Java é robusta. Administração de memória (alocação e liberação) e o uso de ponteiros, duas das fontes de erros e bugs mais frequentes em programas em C e C++, são administrados internamente na linguagem, de forma transparente para o programador.

Java tem bibliotecas prontas para diversas aplicações. As bibliotecas de classes de Java contém várias classes que implementam diversos mecanismos de entrada e saída, acesso à Internet, manipulação de Strings em alto nível, poderosas estruturas de dados, utilitários diversos e um conjunto completo de classes para implementação de interfaces gráficas. (SANTOS, 2001, p. vi-vii).

2.3.2 IDE NetBeans

Para o desenvolvimento do sistema foi utilizado o NetBeans IDE 8.2 no qual permite desenvolver de forma rápida e fácil aplicativos para desktop, dispositivos móveis e web com Java, JavaScript, HTML5, PHP, C / C ++ e muito mais.

Figura 3: NetBeans IDE 8.2.



Fonte: O autor (2020).

Para facilitar o desenvolvimento de vários tipos de aplicativos, o NetBeans IDE fornece integração com algumas ferramentas e serviços externos, como por exemplo, o uso do driver de conexão **postgresql-42.2.5.jar**, no qual foi utilizado no desenvolvimento do sistema para possibilitar a conexão com o banco de dados PostgreSQL, como também o uso do driver **poi-3.10.1.jar** que é utilizado para fazer a leitura e escrita de dados em documentos Microsoft Office, que no caso deste trabalho foi utilizado para a leitura de dados que estavam no Microsoft Office Excel e possibilitar a migração para o PostgreSQL.

O NetBeans permite ao programador Java desenvolver de forma rápida a interface de qualquer sistema disponibilizando componentes visuais, os quais podem fazer parte da aplicação apenas arrastando com o mouse o componente desejado. Podemos citar como exemplo tabela, caixa de texto, botões, barra de progresso dentre muitos outros elementos.

O depurador do NetBeans permite colocar pontos de interrupção em seu código-fonte, adicionar observações de campo, percorrer seu código, executar métodos, obter instantâneos e monitorar a execução conforme ela ocorre. Você também pode anexar o depurador a um processo já em execução. (NETBEANS, c2017-2020).

2.3.3 PostgreSQL 9.4

PostgreSQL é um poderoso sistema de banco de dados relacional de objeto de código aberto que usa e estende a linguagem SQL combinada com muitos recursos que armazenam e escalam com segurança as cargas de trabalho de dados mais complicadas. (POSTGRESQL, c1996-2020).

Para os programadores e desenvolvedores há bibliotecas e drivers de conexão para o PostgreSQL para as principais plataformas e linguagens utilizadas, podendo-se citar as seguintes: C/C++, Java/JSP, PHP, ASP, .NET, Perl, Python, Ruby, Tcl, e driver ODBC, entre outros. (MILANI, 2008, p. 28)

Em versões mais atuais, o PostgreSQL pode ser encontrado para instalação em diversos sistemas operacionais, como por exemplo: Linux, MacOS, Windows, Solaris e BSD.

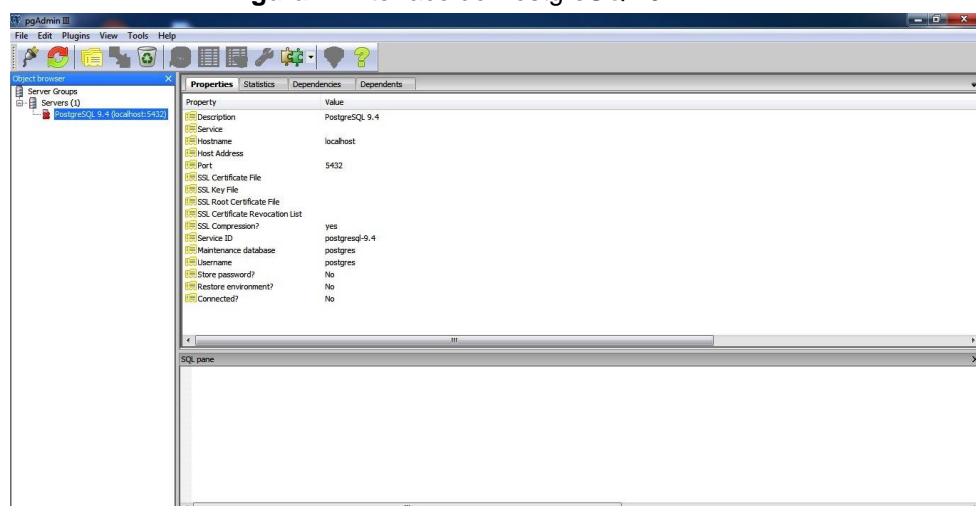
Com mais de 15 anos de desenvolvimento ativo e uma arquitetura que comprovadamente ganhou forte reputação de confiabilidade, integridade de dados e conformidade a padrões. (CARVALHO, 2017, p. 4)

Ainda segundo (Vinícius Carvalho, 2017) o PostgreSQL é fácil de usar, seguro, poderoso, confiável e rápido.

De acordo com (Vinícius Carvalho, 2017) o qual realizou uma pesquisa entre desenvolvedores que usam o PostgreSQL no dia a dia, 90% dos pesquisados se mostraram satisfeitos com o desempenho.

Portanto, podemos perceber que existem diversos motivos para se escolher o PostgreSQL (Figura 4) como o SGBD para o desenvolvimento do Sistema Semear no que se refere à persistência dos dados.

Figura 4: Interface do PostgreSQL 9.4.



Fonte: O autor (2020).

3 DESENVOLVIMENTO DO SISTEMA

Este capítulo irá tratar de forma detalhada como aconteceu o desenvolvimento do sistema, sua implementação, validação e resultados.

3.1 ENGENHARIA DE REQUISITOS

O resultado do levantamento de requisitos são documentos que contém as descrições do que deve ser implementado descrito em alto nível de modo a permitir o entendimento e aceitação entre os futuros usuários e os analistas.

Para a elicitação dos requisitos foi observado um sistema similar antes já utilizado, além da utilização da técnica de entrevista aberta com o objetivo de compreender melhor as reais necessidades para o desenvolvimento do sistema.

“Para que um projeto de desenvolvimento de software seja considerado de sucesso, uma das premissas é que o produto gerado atenda o que o cliente deseja.” (GONÇALVES; CORTÉS, 2015, p. 23).

“Durante a análise orientada a objetos, há uma ênfase em encontrar e descrever os objetos – ou conceitos – no domínio do problema.” (LARMAN, 2005, p. 35).

De acordo com Fowler (2005, p. 47), “a atividade de análise de requisitos procura descobrir o que os usuários e clientes de um produto de software querem que o sistema faça.”

O primeiro estágio de qualquer processo de projeto de software é o desenvolvimento de uma compreensão dos relacionamentos entre o software que está sendo projetado e seu ambiente externo. Isso é essencial para decidir como oferecer a funcionalidade requerida para o sistema e como estruturar o sistema para se comunicar com seu ambiente. A compreensão do contexto também permite estabelecer os limites do sistema. (SOMMERVILLE, 2011, p.126).

“Quando você modela as interações de um sistema com seu ambiente, deve usar uma abordagem abstrata sem muitos detalhes. Uma maneira de fazer isso é usar um modelo de caso de uso.” (SOMMERVILLE, 2011, p.126).

Conforme mencionado no Capítulo 2, a seguir serão utilizadas algumas técnicas de UML e aplicar os conceitos para o desenvolvimento do Sistema Semear.

“Casos de uso e o diagrama de caso de uso ajudam a determinar a funcionalidade e as características do software sob o ponto de vista do usuário.” (PRESSMAN, 2011, p. 731).

3.1.1 Diagrama de caso de uso

O diagrama de caso de uso é utilizado para apresentar uma visão externa geral das funções e serviços que um sistema possui. Nesse instante é importante saber o que o sistema faz não se preocupando com os detalhes de como o sistema faz. Os diagramas de casos são compostos por três principais partes:

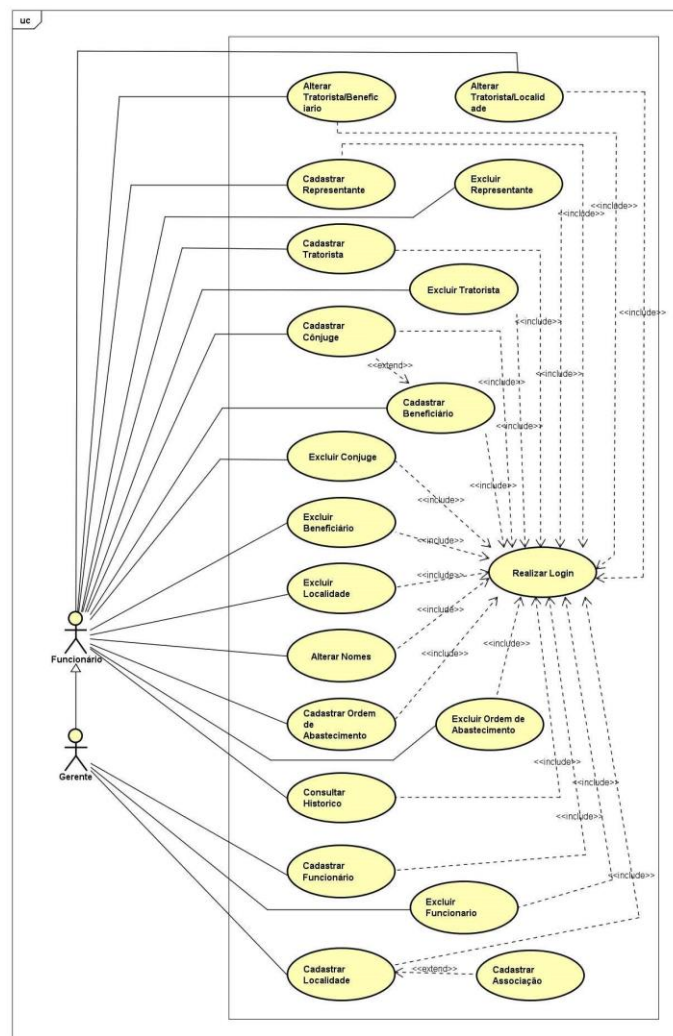
Atores: Quem executa a funcionalidade;

Casos de uso: Qual é a funcionalidade;

Relacionamentos: Como atores e casos de uso se relacionam.

“Um diagrama UML de caso de uso é uma visão geral de todos os casos de uso e como estão relacionados. Fornece uma visão geral da funcionalidade do sistema.” (PRESSMAN, 2011, p. 732).

Figura 5: Diagrama de caso de uso – Programa Semear.



Fonte: O autor (2020).

3.1.2 Caso de uso

“Um caso de uso descreve como um usuário interage com o sistema definindo os passos necessários para atingir um objetivo específico.” (PRESSMAN, 2011, p. 732).

A seguir será detalhado, como exemplo, o caso de uso para realizar o cadastro de um beneficiário no sistema. Os demais casos de uso constam nos apêndices deste trabalho.

Tabela 1: Caso de uso Cadastrar Beneficiário

Nome do Caso de Uso	Cadastrar Beneficiário
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para cadastrar um Beneficiário no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Cadastro do Beneficiário realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para cadastro do Beneficiário.	
	2- Exibe a tela para cadastro.
3- Seleciona a localidade.	
4- Informa o número do CPF do Tratorista.	
	5- Valida CPF.
	6- Localiza Tratorista.
7- Informa o número do CPF do Beneficiário.	
	8- Valida CPF.
	9- Verifica existência de cadastro no ano.
10- Insere nome do Beneficiário.	
11- Informa o número do CPF do Cônjuge.	
	12- Valida CPF.
	13- Verifica existência de cadastro no ano.
14- Insere nome do Cônjuge.	
15- Informa na tela a finalização do cadastro.	
	16- Exibe mensagem se o cadastro foi realizado com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- Tratorista deverá estar cadastrado.
	3- Beneficiário não poderá ter mais de 1 (um) cadastro no mesmo ano.
	4- Beneficiário não possui Cônjuge.
	5- Cônjuge não pode ter cadastro no ano.

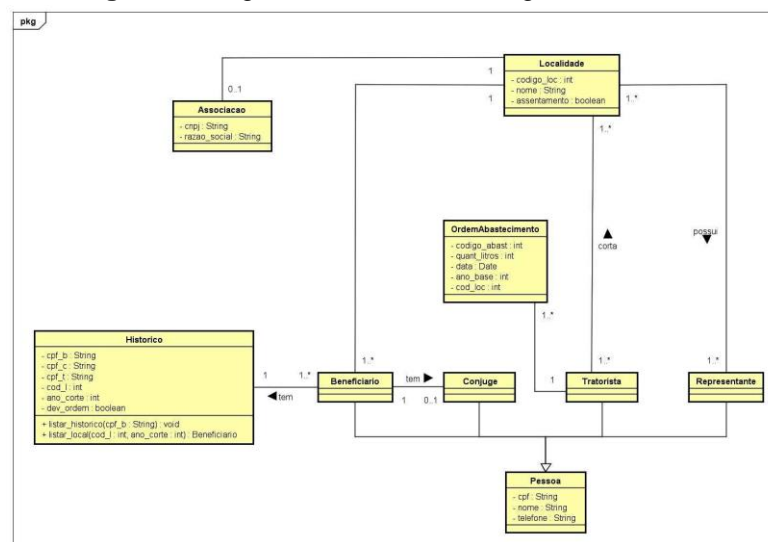
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – Tratorista não possui cadastrado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Tratorista não possui cadastro.
Fluxo de Exceção III – Beneficiário já possui cadastro no ano	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o beneficiário não pode ser cadastrado.
Fluxo de Exceção IV – Beneficiário não possui Cônjuge	
Ações do Ator	Ações do Sistema
1- Segue para a ação 15 (Finalizar).	
Fluxo de Exceção V – Cônjuge já possui cadastro	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o beneficiário não pode ser cadastrado.

3.1.3 Diagrama de classes

De acordo com Fowler (2005, p. 52), um diagrama de classes descreve os tipos de objetos presentes no sistema e os vários tipos de relacionamentos estáticos existentes entre eles. Os diagramas de classes também mostram as propriedades e as operações de uma classe e as restrições que se aplicam à maneira como estão conectados.

Através da Figura 6 serão mostradas as classes obtidas a partir dos levantamentos de requisitos do Programa Semear e como estas se relacionam.

Figura 6: Diagrama de classes – Programa Semear.



Fonte: O autor (2020).

3.1.4 Diagrama de transição de estados

A seguir serão citados conceitos para este diagrama tratado com nomenclaturas diferentes para alguns autores, mas se tratando do mesmo diagrama e com o mesmo objetivo dentro da engenharia de requisitos.

O propósito do uso de um diagrama de transição de estado é demonstrar o comportamento de um elemento por meio de um conjunto de transições de estado.

“Os diagramas de máquina de estados são uma técnica conhecida para descrever o comportamento de um sistema.” (FOWLER, 2005, p. 110).

“Um diagrama de estado modela os estados de um objeto, as ações executadas dependendo daqueles estados e as transições entre os estados do objeto.” (PRESSMAN, 2011, p. 739).

Um diagrama de transição de estados possui alguns elementos básicos. Os principais são:

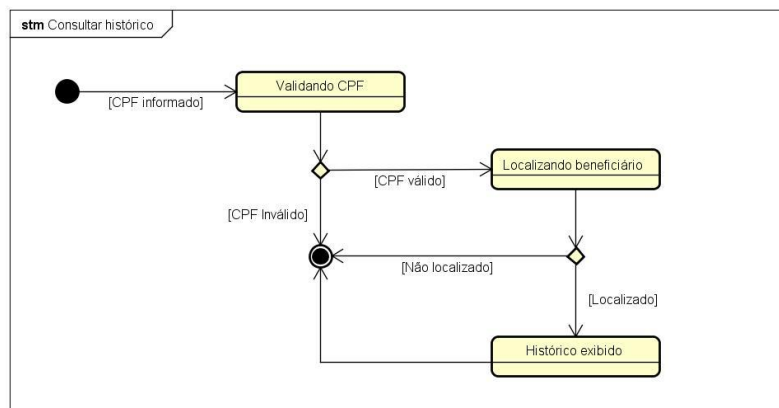
Estados: É a condição em que se encontra um objeto;

Transições: Relação entre estados. Indica qual o estado posterior de um objeto após a ocorrência de um evento;

Evento: É um acontecimento significativo que geralmente promove uma mudança de estado.

A figura 7 mostra exemplo do referido diagrama para consultar o histórico do beneficiário.

Figura 7: Diagrama de transição de estados – Consultar histórico.



Fonte: O autor (2020).

3.2 IMPLEMENTAÇÃO

De todo o processo para o desenvolvimento do sistema, a implementação foi uma das partes mais sensíveis. Quando se trata de dados devemos ter um cuidado todo especial para não comprometer todo o trabalho devido a alguma manipulação equivocada desses dados.

Um dos pontos mais importante para o desenvolvimento do sistema foi a migração de dados, os quais estavam no sistema desenvolvido na linguagem Clipper. Para possibilitar essa migração foi utilizada a ferramenta Microsoft Office Excel, já que esta possibilita a leitura de arquivos usados pela linguagem Clipper para a persistência de dados (arquivos com extensão **.DBF**) e fazer a conversão para arquivos Microsoft Office Excel (arquivos com extensão **.xls**). Esse sistema foi utilizado durante o período do ano 2005 à 2009. A partir do ano de 2010 até o ano de 2020, todos os dados estavam incluídos em planilhas do Microsoft Office Excel. No entanto, esse período foi o mais complicado para a migração dos dados, tendo em vista que muitas pessoas estavam envolvidas no processo de cadastro e inclusão desses dados nas planilhas. Esse processo fez com que fossem usados domínios diferentes para um mesmo dado e quando esse tipo de erro era detectado foi preciso proceder com tratamento para o domínio correto e dar prosseguimento ao processo de migração dos dados.

Com o uso do sistema gerenciador de banco de dados PostgreSQL, foi possível, através da análise de requisitos e o uso dos conceitos de orientação a objetos, desenvolver todo o banco de dados para receber os dados dos períodos anteriores e posteriores ao desenvolvimento do sistema proposto neste trabalho.

A linguagem de programação Java foi escolhida para o desenvolvimento de toda a parte lógica do sistema e teve fundamental importância para o manuseio dos dados citados anteriormente. Diante do poder da linguagem em questão e como uso de bibliotecas específicas, foi possível migrar todos os dados das planilhas do Microsoft Office Excel para o banco de dados desenvolvido através do SGBD PostgreSQL.

Diante da necessidade de migração de dados, conforme mencionado acima, com o auxílio da biblioteca **postgresql-42.2.5.jar** desenvolvida para ser usada com a linguagem Java, foi possível fazer a conexão para interação da linguagem e o banco de dados. Outra biblioteca utilizada foi a **poi-3.10.1.jar**, onde com ela foi

possível fazer a leitura de todos os dados das planilhas do Microsoft Office Excel e possibilitar a gravação no banco de dados do sistema atual.

Sanados os problemas através de soluções implementadas de acordo com os parágrafos acima, iniciou-se o desenvolvimento de toda parte lógica com o uso da linguagem de programação Java.

O IDE NetBeans foi utilizado para acelerar o desenvolvimento da interface do sistema, o qual tem um bom desempenho neste quesito. Na Figura 8 a tela para cadastro do beneficiário é mostrada como exemplo.

Figura 8: Tela de cadastro do beneficiário (Java).

A imagem mostra uma janela de software intitulada "Inserir Beneficiário". O formulário é dividido em seções:

- Localidade:** Um menu suspenso com o texto "Selecione".
- Ano:** Um menu suspenso com o valor "2020".
- CPF Tratorista:** Um campo de texto com máscara ". . ." e um botão "Localizar".
- Titular:** Um campo "CPF:" com máscara ". . ." e botão "Localizar", seguido de um campo "Nome:".
- Cônjuge:** Radio buttons para "Sim" e "Não", um campo "CPF:" com máscara ". . ." e botão "Localizar", seguido de um campo "Nome:".

 No rodapé da janela, há dois botões: "Limpar" e "Inserir". À direita do formulário, há uma área vazia rotulada "Nome".

Fonte: O autor (2020).

3.3 VALIDAÇÃO

Para a validação das funcionalidades do sistema, foram utilizados dois computadores desktops, um com sistema operacional Windows 10 Home, 64 bits, 4G de memória RAM e outro com Windows 7 Ultimate, 32 bits, 2G de memória RAM. No primeiro caso com todas as possíveis funcionalidades desenvolvidas e no segundo para consulta do histórico do beneficiário (Figura 9). Em ambos, o sistema atendeu ao propósito do desenvolvimento deste trabalho, conforme as ilustrações a seguir.

Figura 9: Tela de consulta histórico do beneficiário.

The screenshot shows a window titled "Histórico Beneficiário". At the top, there is a text input field for "CPF:" containing "355.165.854-49" and a "Localizar" button. Below this, the name "JOSE EXPEDITO DA SILVA" is displayed. A table with two columns, "Localidade" and "Anos", lists the following data:

Localidade	Anos
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2005
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2006
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2007
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2008
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2009
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2010
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2011
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2012
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2015
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2017
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2018
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2019
P.A. NOVA ESPERANÇA (FAZ. SAO JOAO)	2020

At the bottom of the window is a "Nova Consulta" button.

Fonte: O autor (2020).

A consulta ao histórico do beneficiário é realizada com bastante frequência para a emissão de declaração para comprovação da participação no Programa Semear com o propósito de obtenção de natalidade, aposentadoria, atualização de cadastro junto ao Sindicato dos Trabalhadores na Lavoura de Mossoró, dentre outros.

Como mostrado na Figura 10, é possível perceber que o sistema verifica a tentativa de inclusão para um beneficiário que já possui cadastro no ano informado. Neste caso, uma mensagem de alerta é mostrada na tela informando da existência do cadastro, impedindo assim um duplo cadastro no mesmo ano, evitando o recebimento de mais de 1 (uma) hora por beneficiário.

Figura 10: Identificação de cadastro já existente.

The screenshot shows a window titled "Inserir Beneficiário". It contains several input fields: "Localidade:" (ALAGOINHA), "Ano:" (2020), "CPF Tratorista:" (009.688.154-21), "CPF Titular:" (701.965.164-04), and "Cônjuge:" (radio buttons for Sim and Não). A "Mensagem" dialog box is overlaid on the window, displaying an information icon and the text "Beneficiário já cadastrado em 2020!" with an "OK" button. At the bottom of the window are "Limpar" and "Inserir" buttons.

Fonte: O autor (2020).

O sistema também atende ao requisito de não aceitar CPFs inválidos, como mostrado na Figura 11. Esta funcionalidade é utilizada para qualquer momento em tenha que usar o CPF para executar qualquer função que necessite da inclusão deste.

Figura 11: Verificação de validade do CPF.



Fonte: O autor (2020).

De acordo com critérios do Programa Semear, caso o beneficiário cadastrado possua um cônjuge, este não poderá realizar cadastro para o mesmo propósito. Portanto, a Figura 12 mostra também uma mensagem de alerta informando a situação.

Figura 12: Impedimento de cadastro do cônjuge como beneficiário.



Fonte: O autor (2020).

Outra funcionalidade bastante importante que o sistema possui é evitar que o tratorista seja excluído quando o mesmo tiver beneficiários vinculados ao seu cadastro. A situação mencionada não poderá ocorrer, tendo em vista, que o cadastro do beneficiário deverá obrigatoriamente ter um tratorista. Neste caso uma

substituição deverá ocorrer. Portanto, a Figura 13 identifica que o tratorista não poderá ser excluído.

Figura 13: Impedimento de exclusão do tratorista.



Fonte: O autor (2020).

O sistema ainda dispõe do controle para cadastro de ordens de abastecimento (Figura 14), em que só será permitida a inclusão de ordens se o tratorista tiver beneficiários vinculados ao seu cadastro. Esta funcionalidade evita o cadastro de ordens para tratorista que tenha cadastro, mas ainda não tenha pelo menos 1 (um) beneficiário vinculado.

Figura 14: Tratorista sem beneficiários vinculados.



Fonte: O autor (2020).

Diante das situações mostradas acima, o sistema atende aos principais requisitos para que este resolva os principais problemas apresentados pelo Programa Semear.

3.4 RESULTADOS

Após os procedimentos de validação, puderam-se tirar alguns resultados referente ao sistema desenvolvido. O primeiro deles é com relação ao procedimento

para consulta do histórico de beneficiários e a emissão de comprovante de participação no Programa Semear que antes levava pelo menos 20 (vinte) minutos e que agora diminuiu para menos de 5 (cinco) minutos. Esse tempo é levado em consideração desde a chegada do beneficiário até a assinatura pelo funcionário responsável e posterior entrega do documento. Todo o processo para consulta está concentrado em um único sistema, diferente de como acontecia antes, o qual foi desenvolvido para possibilitar consultas mais rápidas, proporcionando uma redução considerável no tempo para o atendimento à solicitação.

Outro ponto importante é a detecção da tentativa de duplo cadastro por parte de um mesmo beneficiário e o impedimento do cônjuge do beneficiário de ser beneficiado com o mesmo propósito no mesmo período, dando a oportunidade ao atendimento de outros beneficiários, beneficiando assim um número maior de famílias com o corte de terra.

Outro resultado relevante é referente ao controle da distribuição de ordem de abastecimento. Como antes o controle era feito em planilhas do Microsoft Office Excel, esta teria que ser atualizada manualmente a cada momento em que um novo beneficiário era cadastrado no Programa Semear. Com este novo modelo de trabalho as atualizações ocorrem automaticamente, evitando qualquer tipo de erro em consequência de esquecimento por parte do funcionário.

Em mais um resultado com certa relevância é a forma como o relatório do beneficiário é emitido. No modelo anterior o beneficiário teria que informar em qual localidade o mesmo foi beneficiado, onde em alguns momentos este não tinha precisão em qual localidade estaria o seu cadastro. Em muitos casos isso ocorre quando o beneficiário reside em extremos entre uma localidade e outra. Com o atual procedimento de consulta é informado apenas o número do CPF, e o sistema já localiza todas as localidades e os anos em que este participou do Programa.

De acordo com o que foi analisado esses são os resultados mais relevantes em relação ao sistema desenvolvido, pois o mesmo atende ao que foi proposto durante a análise dos requisitos.

4 CONCLUSÃO

Diante dos requisitos levantados no decorrer deste trabalho e com a aplicação dos conceitos de orientação a objetos, foi possível desenvolver as principais funcionalidades para um sistema que atenda e resolva os problemas apresentados pelo Programa Semear desde o seu início.

Dessa forma, com o uso de conceituadas ferramentas para desenvolvimento de sistemas, chegou-se a soluções bastante satisfatórias referente à usabilidade do sistema. Facilitando assim, um aprendizado mais rápido para os usuários do sistema.

O sistema foi desenvolvido para ser usado em ambientes com computadores desktop, mas levando em consideração o uso da orientação a objetos para o desenvolvimento do mesmo, este poderá futuramente ser desenvolvido e usado em outros ambientes, como por exemplo, web e mobile, possibilitando que alguns serviços sejam oferecidos sem que tenha a necessidade de comparecimento do beneficiário a Secretaria de Agricultura.

Em trabalhos futuros será implementada a inserção de dependentes no cadastro do beneficiário e a possibilidade de informar qual o tipo do plantio será realizado, como por exemplo, milho, feijão etc., traçando assim um perfil do beneficiário. Outra funcionalidade prevista para ser implementada é a consulta e emissão de comprovante da participação no Programa Semear via internet, evitando o deslocamento de beneficiários até a Secretaria de Agricultura.

Compreendemos que, o desenvolvido o sistema, irá contribuir para um melhor atendimento a todos os envolvidos no Programa Semear, tendo em vista que a informatização acelera a busca por informações e por consequência proporciona um menor tempo para atender as mais variadas solicitações.

Diante de tudo que foi exposto, a feitura deste trabalho atendeu ao seu objetivo, que era desenvolver um sistema para otimizar todos os procedimentos necessários para a execução do Programa Semear.

REFERÊNCIAS

- About. **postgresql.org**, c1996-2020. Disponível em: <<https://www.postgresql.org/about/>>. Acesso em: 30 out. 2020.
- Apache NetBeans. **netbeans.org**, c2017-2020. Disponível em: <<https://netbeans.org/>>. Acesso em: 23 out. 2020.
- BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. Rio de Janeiro: Campus/Elsevier, 2007.
- CARVALHO, Thiago Leite e. **Orientação a objetos: Aprenda seus conceitos e suas aplicabilidades de forma efetiva**. São Paulo: Casa do Código, 2016.
- CARVALHO, Vinícius. **PostgreSQL: Banco de dados para aplicações web modernas**. São Paulo: Casa do Código, 2017.
- FOWLER, Martin. **UML essencial: um breve guia para linguagem padrão de modelagem de objetos**. Porto Alegre: bookman, 2005.
- GONÇALVES, Enyo José Tavares; CORTÉS, Mariela Inés. **Análise e projeto de sistema**. Fortaleza: EdUECE, 2015.
- HORSTMANN, Cay S.; CORNELL, Gary. **Core java, volume 1: fundamentos**. São Paulo: Pearson, 2010.
- Integração com Ferramentas e Serviços Externos. **netbeans.org**, c2017-2020. Disponível em: <<https://netbeans.org/kb/trails/tools.html>>. Acesso em: 23 out. 2020.
- LARMAN, Craig. **Utilizando uml e padrões: Uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. Porto Alegre: bookman, 2005.
- MOSSORÓ. **Lei n. 3.301, de 21 agosto de 2015**. Jornal Oficial de Mossoró, Mossoró-RN, 21 de agosto de 2015. Disponível em: <<http://jom.prefeiturademossoro.com.br/2015/08/21/jom-n-318/>>. Acesso em: 17 nov. 2020.
- NetBeans IDE Features. **netbeans.org**, c2017-2020. Disponível em: <<https://netbeans.org/features/java/debugger.html>>. Acesso em: 23 out. 2020.
- Obtenha Informações sobre a Tecnologia Java. **Java™**. Disponível em: <<https://www.java.com/pt-BR/about/>>. Acesso em: 03 nov. 2020.
- PostgreSQL 9.4.26 Documentation. **postgresql.org**, c1996-2020. Disponível em: <<https://www.postgresql.org/files/documentation/pdf/9.4/postgresql-9.4-A4.pdf>>. Acesso em: 30 out. 2020.
- PRESSMAN, Roger S.. **Engenharia de software: Uma abordagem profissional**. Porto Alegre: bookman, 2011.

RUBIRA, Cecília Mary Fischer; BRITO, Patrick Henrique da Silva. **Introdução à análise orientada a objetos e projeto arquitetural**. Campinas-SP: Instituto de computação, 2009.

SANTOS, Rafael. **Introdução à programação orientada a objetos usando java**, 2001. Disponível em: <<https://ramonrdm.files.wordpress.com/2011/09/java-orientado-a-objetos.pdf>>. Acesso em: 30 out. 2020.

SOMMERVILLE, Ian. **Engenharia de software**. São Paulo: Pearson, 2011.

APÊNDICE A – Casos de uso Sistema Semear

Nome do Caso de Uso	Cadastrar Cônjuge
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para cadastrar um Cônjuge no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Cadastro do Cônjuge realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para cadastro do Cônjuge.	
	2- Exibe a tela para cadastro.
3- Informa o número do CPF do Beneficiário.	
	4- Valida CPF.
	5- Localiza Beneficiário.
6- Informa o número do CPF do Cônjuge.	
	7- Valida CPF.
8- Insere nome do Cônjuge.	
9- Informa na tela a finalização do cadastro.	
	10- Exibe mensagem se o cadastro foi realizado com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O Beneficiário deverá estar cadastrado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O Beneficiário não possui cadastro	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Beneficiário não foi localizado.

Nome do Caso de Uso	Cadastrar Tratorista
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para cadastrar um Tratorista no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Cadastro do Tratorista realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para cadastro do Tratorista.	
	2- Exibe a tela para cadastro.
3- Informa o número do CPF do Tratorista.	
	4- Valida CPF.
5- Insere nome do tratorista.	
6- Insere o número do telefone do Tratorista.	
7- Informa na tela a finalização do cadastro.	
	8- Exibe mensagem se o cadastro foi realizado com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O número do telefone deverá ser informado
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O número do telefone não foi informado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o número do telefone é obrigatório.

Nome do Caso de Uso	Cadastrar Representante
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para cadastrar um Representante no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Cadastro do Representante realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para cadastro do Representante.	
	2- Exibe a tela para cadastro.
3- Seleciona a localidade.	
4- Seleciona o ano.	
5- Informa o número do CPF do Representante.	
	6- Valida CPF.
7- Insere nome do Representante.	
8- Insere o número do telefone do Representante.	
9- Informa na tela a finalização do cadastro.	
	10- Exibe mensagem se o cadastro foi realizado com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O número do telefone deverá ser informado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O número do telefone não foi informado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o número do telefone é obrigatório.

Nome do Caso de Uso	Cadastrar Localidade
Ator Principal	Gerente.
Resumo	Descreve as etapas e condições para cadastrar uma Localidade no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Cadastro da Localidade realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para cadastro da Localidade.	
	2- Exibe a tela para cadastro.
3- Insere nome da localidade.	
4- Seleciona o tipo da localidade (Assentamento ou Comunidade).	
5- Informa na tela a finalização do cadastro.	
	6- Verifica nome da localidade.
	7- Exibe mensagem se o cadastro foi realizado com êxito.
Restrições/Validações	1- O nome da localidade não pode já existir.
	2- O tipo da localidade deverá ser informado.
Fluxo de Exceção I – O nome da localidade já existe	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o nome informado já existe.
Fluxo de Exceção II – Tipo da localidade não informado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o tipo da localidade não foi informado.

Nome do Caso de Uso	Cadastrar Associação
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para cadastrar uma Associação no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Cadastro da Associação realizada.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para cadastro da Associação.	
	2- Exibe a tela para cadastro.
3- Seleciona a localidade.	
4- Informa o número do CNPJ da Associação.	
	5- Valida CNPJ.
6- Insere nome da Associação.	
7- Informa na tela a finalização do cadastro.	
	8- Exibe mensagem se o cadastro foi realizado com êxito.
Restrições/Validações	1- O número do CNPJ informado deverá ser válido.
Fluxo de Exceção I – CNPJ inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CNPJ não está correto.

Nome do Caso de Uso	Cadastrar Ordem de Abastecimento
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para cadastrar uma Ordem de Abastecimento no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Cadastro da Ordem realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para cadastro da Ordem de abastecimento.	
	2- Exibe a tela para cadastro.
3- Informa o número do CPF do Tratorista.	
	4- Valida CPF.
	5- Localiza Tratorista.
	6- Localiza Localidade(s).
7- Seleciona localidade.	
8- Informa código da ordem.	
9- Informa data da ordem.	
10- Informa quantidade do combustível.	
11- Informa na tela a finalização do cadastro.	
	12- Exibe mensagem se a ordem foi cadastrada com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O Tratorista não foi localizado.
	3- Localidade(s) não foi localizada.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O Tratorista não foi localizado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Tratorista não foi localizado.
Fluxo de Exceção III – Nenhuma localidade foi encontrada	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Tratorista ainda não possui beneficiário(s) vinculado.

Nome do Caso de Uso	Excluir Beneficiário
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para excluir um Beneficiário no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Exclusão do Beneficiário realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para exclusão do Beneficiário.	
	2- Exibe a tela para exclusão.
3- Seleciona o ano para exclusão.	
4- Informa o número do CPF do Beneficiário.	
	5- Valida CPF.
	6- Localiza Beneficiário.
7- Clica no botão excluir.	
	8- Exibe mensagem se o Beneficiário foi excluído com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O Beneficiário não foi localizado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O Beneficiário não foi localizado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Beneficiário não foi localizado no ano informado.

Nome do Caso de Uso	Excluir Cônjuge
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para excluir um Beneficiário no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Exclusão do Cônjuge realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para exclusão do Cônjuge.	
	2- Exibe a tela para exclusão.
3- Seleciona o ano para exclusão.	
4- Informa o número do CPF do Cônjuge.	
	5- Valida CPF.
	6- Localiza Cônjuge.
7- Clica no botão excluir.	
	8- Exibe mensagem se o Cônjuge foi excluído com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O Cônjuge não foi localizado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O Cônjuge não foi localizado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Cônjuge não foi localizado no ano informado.

Nome do Caso de Uso	Excluir Tratorista
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para excluir um Tratorista no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Exclusão do Tratorista realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para exclusão do Tratorista.	
	2- Exibe a tela para exclusão.
3- Informa o número do CPF do Tratorista.	
	4- Valida CPF.
	5- Localiza Tratorista.
	6- Verifica a existência de Beneficiários vinculados ao Tratorista.
7- Clica no botão excluir.	
	8- Exibe mensagem se o Tratorista foi excluído com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O Tratorista não foi localizado.
	3- Existem Beneficiários vinculados.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O Tratorista não foi localizado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Tratorista não foi localizado.
Fluxo de Exceção III – Existem Beneficiários vinculados ao Tratorista.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Tratorista não pode ser excluído.

Nome do Caso de Uso	Excluir Representante
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para excluir um Representante no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Exclusão do Representante realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para exclusão do Representante.	
	2- Exibe a tela para exclusão.
3- Seleciona a localidade.	
4- Seleciona o ano para exclusão.	
5- Informa o número do CPF do Representante.	
	6- Valida CPF.
	7- Localiza Representante.
8- Clica no botão excluir.	
	9- Exibe mensagem se o Representante foi excluído com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- O Representante não foi localizado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – O Representante não foi localizado	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que o Representante não foi localizado.

Nome do Caso de Uso	Excluir Localidade
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para excluir uma Localidade no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Exclusão da Localidade realizada.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para exclusão da Localidade.	
	2- Exibe a tela para exclusão.
3- Seleciona Localidade.	
4- Seleciona excluir.	
	5- Verifica a existência de Beneficiários vinculados a Localidade.
	8- Exibe mensagem se a Localidade foi excluída com êxito.
Restrições/Validações	1- Existem Beneficiários vinculados.
Fluxo de Exceção I – Existem Beneficiários vinculados a Localidade.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem que a Localidade não pode ser excluída.

Nome do Caso de Uso	Excluir Associação
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para excluir uma Associação no sistema.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Exclusão da Associação realizada.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para exclusão da Associação.	
	2- Exibe a tela para exclusão.
3- Informa o número do CNPJ da Associação.	
	4- Valida CNPJ.
	5- Localiza Associação.
6- Seleciona excluir.	
	7- Exibe mensagem se a Associação foi excluída com êxito.
Restrições/Validações	1- O número do CNPJ informado deverá ser válido.
	2- Associação não foi localizada.
Fluxo de Exceção I – CNPJ inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CNPJ não está correto.
Fluxo de Exceção II – Associação não foi localizada.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que a Associação não foi localizada.

Nome do Caso de Uso	Excluir Ordem de Abastecimento
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para excluir uma Ordem de Abastecimento.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Exclusão da Ordem realizada.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para exclusão da Ordem de abastecimento.	
	2- Exibe a tela para exclusão.
3- Seleciona o ano base da ordem.	
4- Informa o número do CPF do tratorista.	
	5- Valida CPF.
	6- Localiza Tratorista.
	7- Localiza ordens.
6- Seleciona o código da ordem para excluir.	
Seleciona excluir.	
	7- Exibe mensagem se a Ordem foi excluída com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- Tratorista não localizado.
	3- Tratorista não possui ordem cadastrada.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – Tratorista não foi localizada.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o tratorista não foi localizado.
Fluxo de Exceção III – Tratorista não possui ordem cadastrada.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que nenhuma ordem foi localizada.

Nome do Caso de Uso	Alterar Tratorista/Beneficiário
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para alterar o tratorista de um beneficiário.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Alteração de tratorista realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para alteração do tratorista.	
	2- Exibe a tela para alteração.
3- Seleciona o ano para alteração.	
4- Informa o número do CPF do beneficiário.	
	5- Valida CPF.
	6- Localiza beneficiário.
	7- Exibe dados.
8- Informa o número do CPF do tratorista substituto.	
	9- Valida CPF.
	10- Localiza tratorista.
11- Seleciona substituir.	
	12- Exibe mensagem se a alteração foi executada com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- Beneficiário não localizado.
	3- Tratorista não localizado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – Beneficiário não foi localizado.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o beneficiário não foi localizado.
Fluxo de Exceção III – Tratorista não foi localizada.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o tratorista não foi localizado.

Nome do Caso de Uso	Alterar Tratorista/Localidade
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para alterar o tratorista de uma localidade.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Alteração de tratorista realizado.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para alteração do tratorista.	
	2- Exibe a tela para alteração.
3- Informa o número do CPF do tratorista atual.	
	4- Valida CPF.
	5- Localiza tratorista.
	6- Exibe dados.
7- Seleciona localidade.	
8- Seleciona ano.	
9- Informa o número do CPF do tratorista atual.	
	10- Valida CPF.
	11- Localiza tratorista.
12- Seleciona substituir.	
	13- Exibe mensagem se a alteração foi executada com êxito.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- Tratorista não localizado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – Tratorista não foi localizada.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o tratorista não foi localizado.

Nome do Caso de Uso	Consultar histórico
Ator Principal	Funcionário.
Resumo	Descreve as etapas e condições para realizar a consulta do histórico de um beneficiário.
Pré-Condições	Estar logado no sistema.
Pós-Condições	Histórico exibido.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1- Seleciona o menu para consulta do histórico.	
	2- Exibe a tela para consulta.
3- Informa o número do CPF do beneficiário.	
	4- Valida CPF.
	5- Localiza beneficiário.
	6- Exibe histórico.
Restrições/Validações	1- O número do CPF informado deverá ser válido.
	2- Beneficiário não localizado.
Fluxo de Exceção I – CPF inválido	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o número do CPF não está correto.
Fluxo de Exceção II – Beneficiário não foi localizado.	
Ações do Ator	Ações do Sistema
	1- Exibir mensagem informando que o beneficiário não foi localizado.