

Centro Federal de Educação Tecnológica Celso Suckow da Fonseca

Departamento de Ensino Superior

Departamento de Informática

Curso Superior de Tecnologia em Sistemas para Internet

Trumah – sistema de doações de bens

Daniel de Souza Brum

Rodrigo Bezerra

Raphael Quintanilha

**Rio de Janeiro
2013**

Trumah – sistema de doações de bens

Daniel de Souza Brum

Rodrigo Bezerra

Raphael Quintanilha

Projeto final apresentado ao Curso Superior de Tecnologia em Sistemas para Internet da CEFET/RJ, como requisito para a obtenção do título de grau de Tecnólogo em Sistemas para Internet.

Orientador: Rafael Castaneda

Doutor

**Rio de Janeiro
2013**

Daniel de Souza Brum

Rodrigo Bezerra

Raphael Quintanilha

Trumah – sistema de doações de bens

Projeto final apresentado ao Curso Superior de Tecnologia em Sistemas para Internet da CEFET/RJ, como requisito para a obtenção do título de grau de Tecnólogo em Sistemas para Internet.

Aprovado em agosto de 2013

BANCA EXAMINADORA

Rafael Castaneda

Doutor

Eduardo Bezerra

Doutor

Renato Mauro

Mestre

**Rio de Janeiro
2013**

**Rio de Janeiro
2013**

Resumo

Realizar uma doação de forma efetiva e organizada em um momento emergencial não é muito comum, o desenvolvimento desse sistema é baseado na necessidade de organizar as coletas e doações, pois muitas doações são trocadas ou desnecessárias para aquele beneficiário no momento em que se encontra. Faremos uma explanação sobre o sistema de doações, sobre a arquitetura do sistema, as técnicas utilizadas para a criação do mesmo e como ele atuará nessa para suprir essas necessidades.

Palavras chaves: Doação, Controle de Bens materiais, Benefício

Abstract

Making a donation in an effective and organized way is not very common in an emergency situation. This system was developed to help the society, by organizing all the donations and redirect them for the right place, because sometimes cities that do not need donations, at that time, receives the donation in place of the city that really needs the donations. We will make an explanation about the donation system, on the system architecture, the techniques used for its creation and how it will work to solve that kind of problem.

Keywords: Donation, System, Benefit

Agradecimentos

Daniel Brum

Agradeço primeiramente a Deus pela inspiração que nos foi dada para a criação e o desenvolvimento do projeto, ao meu pai (Alonso de Oliveira Brum) a minha mãe (Elidia de Souza Brum), a minha irmã (Beatriz de Souza Brum) pela força e por acreditar a todo momento que essa etapa da vida seria vencida.

Agradeço a minha noiva (Luana Nobrega Dantas), por acreditar, por motivar e acima de tudo compreender os momentos de ausência.

Agradeço ao Professor Rafael Castaneda, pelas orientações e pela paciência e aos meus companheiros de desenvolvimento Rodrigo e Raphael, pela determinação e persistência essencial para vencermos as adversidades encontradas no projeto.

Raphael Quintanilha

Agradeço a Deus, ao meu pai (Ivan de Mello Quintanilha), a minha mãe (Elizabeth Jorge Quintanilha) e aos meus amigos de infância que sempre me passaram confiança nesse caminho percorrido.

Agradeço ao Cefet por ter me ajudado a ganhar uma bolsa para estudar nos Estados Unidos, agradeço também aos meus amigos Rodrigo e Daniel, e ao nosso orientador Rafael Castaneda, por toda ajuda durante esse processo.

Rodrigo Bezerra

Primeiramente agradecer à Deus, a minha Mãe (Rosa Maria Bezerra da Silva) e a meu Pai (João Américo dos Santos Maia), por todo apoio dado.

A minha esposa (Poliana Marques Maia) por toda à compreensão, paciência e dedicação.

Ao meu filho (Pedro Henrique Marques Maia) que me dá forças todos os dias para continuar, isso tudo é por você e para você.

E finalmente aos meus amigos Raphael e Daniel, pela dedicação no desenvolvimento do projeto, e ao nosso orientador Rafael Castaneda, sem ele esse projeto não seria possível

Sumário

Lista de Figuras.....	10
Lista de Tabelas.....	11
Lista de Siglas.....	12
Capítulo I.....	1
Introdução.....	1
1.1 <i>Motivação</i>	1
1.2 <i>Justificativa</i>	2
1.3 <i>Objetivos</i>	2
1.4 <i>Metodologia</i>	3
1.4.1 <i>Estudo de Caso</i>	4
1.4.2 <i>Especificação de modelo</i>	4
1.4.3 <i>Processo de desenvolvimento do sistema</i>	4
1.5 <i>Organização do trabalho</i>	5
Capítulo II.....	6
Projeto do sistema.....	6
2.1 <i>Contextualização</i>	6
2.2 <i>Metodologia de desenvolvimento</i>	6
2.3 <i>Especificação, Análise e Projeto do Sistema</i>	7
2.3.1 <i>Especificação</i>	7
2.3.2 <i>Especificação dos requisitos funcionais do sistema</i>	8
2.3.3 <i>Especificação das regras de negócio</i>	8
2.4 <i>Modelo de Casos de Uso</i>	9
2.4.1 <i>Diagrama de Casos de Uso</i>	9
2.4.2 <i>Descrições textuais dos atores</i>	10
2.5 <i>Descrição textuais dos Casos de Uso</i>	11
2.6 <i>Modelo de Classes</i>	15
2.6.1 <i>Diagrama de Classes</i>	15

2.7	<i>Visões de Classes Participantes</i>	16
	A figura 5 demonstra as classes envolvidas no caso de uso Captar Doações	18
	18
2.8	<i>Modelo de Interações</i>	22
2.9	<i>Diagramas de sequência de cada cenário relevante dos casos de uso</i>	22
2.10	<i>Projeto de Banco de Dados</i>	27
2.11	<i>Projeto Lógico do Banco de Dados</i>	27
2.12	<i>Projeto Físico do Banco de Dados</i>	36
2.13	<i>Projeto da Interface Gráfica</i>	37
Capítulo III		39
Aspectos de Implementação		39
3.1	<i>Visão geral da arquitetura da aplicação</i>	39
3.2	<i>Estruturação da camada de apresentação</i>	43
3.3	<i>Estruturação da camada de Negócio</i>	47
3.4	<i>Estruturação da camada de Persistência</i>	47
Capítulo IV		49
Conclusão		49
4.1	<i>Análise Retrospectiva</i>	49
4.2	<i>Trabalhos Futuros</i>	49
Referências		51
Apêndice		52

Lista de Figuras.

Figura 1: Diagrama de Casos de Uso	10
Figura 2: DDC – Diagrama de Classe Conceitual	16
Figura 3: VCP – Agendar Coletas e Distribuições	17
Figura 4: VCP – Cadastrar Beneficiário.....	17
Figura 5: VCP - Captar Doações.....	18
Figura 6: VPC - Conceder Benefício	19
Figura 7: VCP - Criar Rota de Coleta e Distribuição.....	20
Figura 8: VCP - Gerar Gráficos de Doações	20
Figura 9: VCP Gerar Gráficos de Itens doados	21
Figura 10: VCP - Visualizar Mapa de Doações	22
Figura 11: SD - Agendar Coletas e Distribuições	23
Figura 12: SD - Cadastrar Beneficiário.....	23
Figura 13: SD - Captar Doações	24
Figura 14: SD - Incluir Item Doação	24
Figura 15: SD - Conceder Benefício	25
Figura 16: SD - Criar Rota de Coleta e Distribuição	25
Figura 17: SD - Gerar Gráficos de Doações	26
Figura 18: SD - Gerar Gráficos de Itens Doados	26
Figura 19: SD - Visualizar Mapa de Doações	27
Figura 20: PF - Projeto Físico do Banco de Dados	37
Figura 21: SS - Gerar Ranking Itens Doados	38
Figura 22: FC - Estrutura MVC	39
Figura 23: FX - Fluxo JSF	40
Figura 24: CVH - Ciclo de Vida Estado Hibernate	41
Figura 25: SS - Google Directions	42
Figura 26: SS - Google Geocoding	43
Figura 27: SS - Exemplo de Uso Prime Faces.....	44
Figura 28: Primefaces Módulos.....	44
Figura 29: SS - Criar Rota	45
Figura 30: Tela de Verificação de Rota Criada.....	46
Figura 31: Tela de Visualização de coordenadas de Doadores x Beneficiários.....	46
Figura 32: SS - Persistence XML	47
Figura 33: SS – DoacaoDAOImpl.....	48

Lista de Tabelas.

Tabela 1: PL - Tabela assistente_logistica	28
Tabela 2: PL - Tabela assistente_social	28
Tabela 3: PL - Tabela assistente_social_beneficiario	28
Tabela 4: PL - Tabela assistente_social_doacao.....	28
Tabela 5: Tabela atendente.....	29
Tabela 6: PL - Tabela beneficiario	29
Tabela 7: Tabela beneficio	29
Tabela 8: PL - Tabela cidade	30
Tabela 9: PL - Tabela coordenadas.....	30
Tabela 10: PL - Tabela dependente.....	30
Tabela 11: PL - Tabela doacao.....	31
Tabela 12: Tabela doacao_itens.....	31
Tabela 13: Tabela doador.....	31
Tabela 14: PL - Tabela doador_fisico.....	32
Tabela 15: PL - Tabela doador_juridico.....	32
Tabela 16: PL - Tabela endereco	32
Tabela 17: PL - Tabela estado.....	33
Tabela 18: PL - Tabela funcionario	33
Tabela 19: PL - Tabela item	33
Tabela 20: PL - Tabela itemdoado.....	33
Tabela 21: PL -Tabela menu	34
Tabela 22: PL - Tabela menu_perfil.....	34
Tabela 23: PL - Tabela pais	34
Tabela 24: PL - Tabela perfil	34
Tabela 25:PL - Tabela perfil_usuario	35
Tabela 26: PL - Tabela pessoa	35
Tabela 27: PL - Tabela rota.....	35
Tabela 28: PL - Tabela rota_enderecos	36
Tabela 29: PL - Tabela tipomaterial.....	36
Tabela 30: PL - Tabela usuario.....	36

Lista de Siglas

JSF: JavaServer Faces

MVC: Model-view-controller

SGBD: Sistema de Gerenciamento de Banco de Dados

JPA: Java Persistence API

SQL: Structured Query Language

OO: Orientado ao Objeto

UML: Unified Modeling Language

RUP: Rational Unified Process

UI: User Interface

IDE: Interface Development Environment

WAR: Web application Archive

ORM: Object Relational Model

API: Application Programming interface

HTTP: Hypertext Transfer Protocol

XML: eXtensible Markup Language

**Rio de Janeiro
2013**

Capítulo I

Introdução

Devido ao incidente de causas naturais ocorridos no estado do Rio de Janeiro em meados de janeiro do ano de 2011 que causou um grande estrago na região serrana do estado, essa enchente de impacto devastador foi noticiada pela mídia e logo obtiveram repercussão internacional, em contrapartida, o sentimento sobre o desastre natural foi revertido de desespero e angústia em solidariedade e boas ações, conseqüentemente várias doações chegaram dos mais variados cantos do Brasil e do mundo.

Tivemos a oportunidade de conhecer um colaborador que gentilmente cedeu férias para agir em prol das vítimas, esse homem discursou sobre as situações que defrontou, sobre a falta de estrutura, sobre a organização e sobre o controle adequado do que se recebia como doações, e no momento da homilia já que o homem mencionado nas linhas anteriores é um padre, acometidos pela onda contagiante de solidariedade chegamos à conclusão que a forma que poderíamos ajudar era com o conhecimento que possuímos, logo o surgiu a ideia que conceber um sistema que os auxiliasse a gerenciar o recebimento dos bens e doações como também otimizasse e gerenciasse as rotas das doações desde o processo de solicitação até a chegada ao destino final.

1.1 *Motivação*

“Ao vivenciarmos os desígnios de nossa condição humana, estamos sempre tendo motivos para se fazer presente o arrependimento por atos falhos, atos até mais ousados, os quais nos levarão simplesmente à percepção de nossa fragilidade.”

Ivan Teorilang

Difícilmente nos preocupamos com grandes problemas até que aconteça, acreditamos que todo o mundo refletiu após os ocorridos na tragédia da região serrana em 2011, após o incidente houve outras ocorrências em outros pontos do estado registrados, mas recentes em janeiro de 2012, ou seja o que era um incidente tornou-se recorrente, todos de alguma forma seja por orações ou doações materiais se manifestaram solidariamente, após o relato do padre, pensamos em uma forma de organizar, uma forma de ajudar as pessoas com o que tínhamos, no nosso caso o conhecimento de informática. Segundo Karl Marx (1818~1883) "Uma ideia torna-se uma força material quando ganha as massas organizadas.", tomando essa ideia como norte, começamos a buscar pessoas de diferentes habilidades que tornasse possível o objetivo de produzir um sistema que ajudasse a organizar as doações, traçar as

rotas para coletas e doações e que produzisse dados para que houvesse um planejamento ao qual amenizasse o impacto de uma nova tragédia.

1.2 Justificativa

“Aquele que tenta reformar o seu próximo não deve cultivar nunca uma grande amizade com ele antes do objetivo ter sido atingido.
Mahatma Gandhi

Com a marca de quase 1000 (mil) vítimas entre vítimas fatais e desabrigados a situação caótica propendia a falta de organização e antes que ocorra a reorganização muitos auxílios se perdem ou demoram a chegar as regiões afetadas, o intuito do sistema é catalogar os doadores e beneficiários, regiões afetadas e regiões beneficentes traçando rotas da melhor forma para a chegada do auxílio, não somente em condições de emergência mas no cotidiano.

Com base nos dados coletados é possível trabalhar atuando de forma inicialmente reativa sob as carências regionais e pró-ativamente nas regiões com tendências a ser afetadas, com base nos dados socioeconômicos coletados.

1.3 Objetivos

Com os dados históricos das regiões afetadas pretende-se atuar pró-ativamente nas regiões de mais necessidades canalizando as doações de forma eficiente através dos princípios básicos da logística:

"Logística é a parte do Gerenciamento da Cadeia de Abastecimento que planeja, implementa e controla o fluxo e armazenamento eficiente e econômico de matérias-primas, materiais semiacabados e produtos acabados, bem como as informações a eles relativas, desde o ponto de origem até o ponto de consumo, com o propósito de atender as exigências dos clientes" (Carvalho, 2002).[3]

Dentro dos objetivos do Sistema seria a melhoria na administração das doações, com os dados coletados será possível personalizar a necessidade de cada beneficiário, e alinhar com os itens doados.

Além disso é possível determinar e divulgar os bens de maior necessidade. Será possível traçar rotas e levar a ajuda diretamente ao beneficiário, descentralizando as doações e desta forma minimizando fraudes e desperdícios.

Gerar relatórios, que podem antecipar às necessidades, minimizando assim os impactos de futuras tragédias

1.4 Metodologia

Para que o sistema possua dados históricos para análise é necessário o cadastro pessoal dos beneficiários e doadores, sendo um beneficiário o cadastro do mesmo é realizado por um assistente social que identifica as reais necessidades do mesmo, caso seja um doador poderá ser acessado via telemarketing ou de forma ativa, seja ligando para o telemarketing receptivo, ou cadastrando sua doação através do site.

O sistema será desenvolvido em Java e será responsável por processar relativos as regiões tal como traçar rotas, apenas o escopo de funcionários terá acesso as informações que são de cunho sigiloso. Devido a propensão em atuar em grandes distancias entre o servidor os pontos de acesso ao sistema será acessado pela web.

O intuito do sistema é processar as informações sobre regiões, gerar dados com o máximo de concisão possível, tais como identificar carências e generosidades das regiões e mapear as rotas mais eficientes para a coleta e entrega das doações.

Como informado anteriormente o sistema será desenvolvido em Java, será utilizado JSF [17], que utiliza o padrão MVC [1], será utilizado SGBD e servidor web, a arquitetura será desenvolvida em camadas e segundo o conceito será desenvolvido em 3 camadas:

- Camada de interface.
- Camada de negócios ou domínio.
- Camada de dados.

Camada de interface tratará da interação do usuário com o sistema (parte gráfica), a camada de negócios ou domínio tratará das regras de negócio e a camada de dados tratará da persistência dos dados a pretensão de uso do Hibernate.

Utilizaremos o ambiente Eclipse para o desenvolvimento da aplicação em Java/JSF, o banco de dados utilizado será o MySQL e o servidor de aplicação será o TomCat, o sistema operacional será o Windows 7© para o servidor hospedar a aplicação, todavia as configurações do cliente necessitará apenas de um web browser.

Escolha decorrente ao equipamento que possuímos atualmente dentro das capacidades de hardware para demonstração da aplicação.

1.4.1 Estudo de Caso

O intuito é realizar a modelagem de um sistema que auxilie a montagem de rotas para aperfeiçoar o tempo de entrega e coleta de doações, e futuramente dados históricos para antecipação a eventuais catástrofes.

1.4.2 Especificação de modelo

Realizamos toda a modelagem do sistema com a ferramenta UML, toda a parte de desenvolvimento foi realizado em OO, o que permitiu que o sistema pudesse ser desenvolvido de forma segmentada, ou seja, cada membro desenvolveu uma parte do sistema de forma incremental e interativamente.

1.4.3 Processo de desenvolvimento do sistema

Conforme o descrito anteriormente o desenvolvimento do sistema ocorreu embasado no Modelo de Ciclo de Vida Iterativo e Incremental, padrão de desenvolvimento que de forma intrínseca uniu o RUP e o desenvolvimento ágil de software. Para a concepção de um software conforme o modelo eleito utilizou-se dos seguintes itens:

1. Levantamento de requisitos: Foram realizadas entrevistas com analistas de logística [2] e assistentes sociais para obter informações essenciais para a concepção do sistema;
2. Análise de requisitos: Foi realizada a confecção da diagramação UML [7] através dos modelos de dados obtido conforme do levantamento de requisito e o uso da diagramação UML se fez útil tanto para a orientação do desenvolvimento quanto para um entendimento facilitado do escopo do sistema;
3. Projeto: A escolha da linguagem a ser implementada, o banco de dados a ser utilizado e o acesso liberado a internet se faz estritamente necessário para que o sistema seja desenvolvido e funcione em sua plenitude tanto para testes quanto em produção;
4. Desenvolvimento: A concepção do aplicativo gerada a partir da linguagem de programação previamente alinhada, a definição de uma UI, o armazenamento das informações no SGBD[4] e o acesso à internet para o uso de Webservice;
5. Teste e correções: Todas as funções do sistema foram testadas.
6. Implementação: O sistema será colocado em produção para uso de instituições filantrópicas.

1.5 **Organização do trabalho**

A estrutura do trabalho está definida conforme os seis capítulos listados abaixo:

Capítulo I - Introdução: Apresentará uma colocação sobre a forma de realizar doações de bens materiais assim como a logística dos mesmos, desde o momento da captura assim como a entrega, tratará também sobre a motivação, e os pontos que serão evidenciados os pontos fundamentais para o desenvolvimento do sistema e a metodologia que será empregada no mesmo;

Capítulo II - Projeto do sistema: Demonstrará detalhes da arquitetura aplicada, a estrutura de dados, componentes e interfaces de funções vitais para o desenvolvimento do sistema, será ressaltado o uso do padrão de arquitetura MVC, o uso de frameworks e o uso fundamental de banco de dados;

Capítulo III- A implementação: Demonstrará as condições básicas para que o sistema flua de forma regular e o modo como o foi realizado a preparação para o uso do sistema Thrumah;

Capítulo IV - Conclusão: Finaliza o estudo aqui referenciado, demonstrando argumentos terminativos sobre o projeto, abordando novas características com a visão de futuras possibilidades de evolução do projeto e uma explanação sobre o conhecimento adquirido para alcançar as metas de entrega do projeto.

Capítulo II

Projeto do sistema

Esse Capítulo descreve o projeto do software desenvolvido bem como os aspectos técnicos envolvidos no processo.

A Seção 2.1 apresenta uma breve explicação sobre as decisões referentes a implementação do processo escolhido para o funcionamento do sistema de doações. A metodologia de desenvolvimento é apresentada na Seção 2.2 A Seção 2.3 descreverá a especificação dos requisitos funcionais, análise e projeto do sistema. A modelagem dos casos de uso é apresentada na Seção 2.4. As descrições dos casos de uso do sistema estão presentes na Seção 2.4.1. O modelo de classes conceituais é apresentado na Seção 2.4.2. A Seção 2.4.3 denota as visões de classes participantes dos casos de uso. O modelo de iterações é representado na Seção 2.8. A Seção 2.9 exibe os diagramas de sequência para cada cenário relevante dos casos de uso. A seção 1.10 descreve o projeto de banco de dados. O projeto lógico de banco de dados é apresentado na Seção 2.11. A Seção 2.12 apresenta o projeto físico do banco de dados. E por fim a Seção 2.13 descreve o projeto da interface gráfica.

2.1 Contextualização

Após a definição da base do sistema, foi necessário consultar analistas da área de logística e assistência social para uma interação dos tramites de coletas e entregas e sobre como atender as necessidades de forma mais objetiva, como forma de auxiliar as formas de criações de rotas, chegamos à conclusão de que visualizar um mapa com os pontos de doações e coletas deixam as dimensões mais palpáveis, ou seja, fica mais visível para quem define a rota o caminho que feito da origem até o destino.

As marcações de doações e beneficiários no mapa dão a noção do impacto, como por exemplo, que região realiza mais doações ou que região necessita de mais de doações esse estímulo visual torna mais participativo e perceptível para as partes envolvidas que podem tomar medidas para auxiliar na resolução do problema.

As próximas seções iram elucidar o processo de desenvolvimento do sistema de doação e as tecnologias empregadas para que ocorresse a criação do sistema.

2.2 Metodologia de desenvolvimento

Atualmente o paradigma de programação mais utilizado é o Orientado Objeto e por causa do seu uso mais comum, mais fácil é de encontrarmos referencias para solucionarmos eventuais problemas, por esse motivo escolhemos esse paradigma para o desenvolvimento do sistema.

Escolhemos a linguagem JAVA[5], por ser uma linguagem utilizada no nosso cotidiano e domínio relativamente básico, para a parte de interface gráfica utilizamos o PrimeFaces 2.2 como framework que possui bastante referência e interfaces gráficas amigáveis, para persistência, escolhemos usar o framework Hibernate 3.5 [10] por sua melhor integração com o Java e sua vasta base de referências, para o sistema de gerenciamento de banco de dados o escolhido foi MySql [18] por ter uma relação performance robustez compatível com a estrutura do sistema, escolhemos a IDE Eclipse para realizarmos o desenvolvimento do sistema tanto da parte Java quanto da parte gráfica, para a modelagem do projeto físico e logico do sistema usamos WorkBench 5.2 [19] e para realizar toda a parte de diagramas utilizamos o Astah [15].

2.3 Especificação, Análise e Projeto do Sistema

Neste capítulo será abordado as características da especificação, análise e projeto de sistema. Será descrito detalhadamente e especificadamente os requisitos funcionais e não funcionais presentes na seção 2.3.1 A Seção 2.3.2 possui a representação do modelo de casos de uso, logo é onde são especificados o diagrama de casos de uso, a descrição dos atores e o detalhamento dos casos de uso. O modelo de classes em conjunto com o diagrama de classes de negócio, dicionário das classes e a visão das classes participantes são apresentados na Seção 2.7. Na Seção 2.9 denotará o modelo de interações contendo e conterá a descrição de e diagramas de sequência. A Seção 2.10 estará descrito o projeto lógico e físico de banco de dados

do sistema. Por fim, a Seção 2.13, apresenta o projeto de interface gráfica que compreenderá o padrão das telas.

2.3.1 Especificação

Os requisitos funcionais que fundamentaram a construção do sistema:

- O atendente poderá cadastrar, alterar e consultar o Doador
- O atendente poderá cadastrar, alterar e consultar itens na doação
- O atendente poderá agendar coletas e distribuições
- O analista de logística poderá cadastrar, consultar, excluir e editar rotas de coletas e distribuição.
- O analista de logística poderá consultar mapa de doações
- O assistente social poderá cadastrar, consultar, excluir e editar beneficiário.
- O assistente social poderá cadastrar, consultar, excluir e editar benefício.
- A administração poderá gerar relatórios de doações.
- A administração poderá gerar relatórios de itens doados.

2.3.2 Especificação dos requisitos funcionais do sistema

A interface gráfica deve ser intuitiva para o usuário conforme as características da usabilidade.

A aplicação foi desenvolvida para acesso web, logo poderá ser acessada através de qualquer meio que possua acesso à internet.

2.3.3 Especificação das regras de negócio

As regras de negócio são políticas, normas, orientações ou restrições que uma organização possui no seu funcionamento. Relaciona a forma que o sistema opera independentemente da existência de um sistema informatizado.

Doação:

RN01 – O atendente pode cadastrar doadores de que estão na lista ou que ligaram par a central de atendimento.

RN02 – O atendente pode incluir itens de doações para o doador que estão cadastrados no sistema.

RN03 – O atendente somente poderá agendar coleta ou distribuição dos itens que estão no cadastro de doações

Manuseio dos itens doados:

RN04 – O analista de logística somente poderá alterar, incluir ou editar as rotas que englobam o doador e beneficiário que estão envolvidos na doação cadastrada no sistema.

RN05 – O analista de logística será o único perfil a ter acesso a consultar os mapas de doações.

Assistência ao doador e ao beneficiário:

RN06 – O assistente social será o único perfil que a ter acesso a excluir doador e a cadastrar, editar e remover um beneficiário.

RN07 – O assistente social será o único perfil a ter acesso a concessão ou a exclusão de benefício.

Administração das informações estatísticas geradas pelo sistema:

RN08 – O administrador é o único perfil que terá acesso a gerar relatórios das doações realizadas.

RN09 – O administrador é o único perfil que terá acesso a gerar relatórios dos itens doados.

2.4 Modelo de Casos de Uso

Segundo Ana Melo, “Um caso de uso descreve uma sequência de ações que representa o cenário principal e cenários alternativos, com o objetivo de demonstrar o comportamento de um sistema através de interações com atores” [7].

Após o levantamento dos requisitos com os usuários é necessário que seja documentados os casos de uso para o entendimento e a validação tanto da parte do desenvolvedor quando da parte do usuário. Esse procedimento evita que informações importantes fiquem sem referências. Para a modelagem do sistema, separamos as funcionalidades do sistema e agrupamos as ações que havia objetivos claros. O fluxo principal ou cenário principal são as ações rotineiras sem problemas em sua execução, ou seja, um cenário perfeito, já o fluxo alternativo ou cenário alternativo

representa as exceções e os Atores são a representação de uma interação externa com o caso de uso, seja um ator uma pessoa ou outro sistema.

2.4.1 Diagrama de Casos de Uso

O diagrama de caso de uso foi criado para expressar os limites do sistema e diagramar os requisitos definidos, a sua existência faz com que ocorra uma visão mais praticável entre os casos de usos e os atores.

Devido a sua simplicidade, os seu uso é de fácil entendimento para os usuários ou requisitantes do sistema. Os atores serão representados por uma figura chamada "Stickman" ou no português claro boneco palito, os casos de usos serão representados por uma figura elíptica e dentro dela contém o nome do caso de uso que representa.

A figura 2.1 demonstra o Diagrama de Caso de Uso relacionado ao Sistema de Doações.

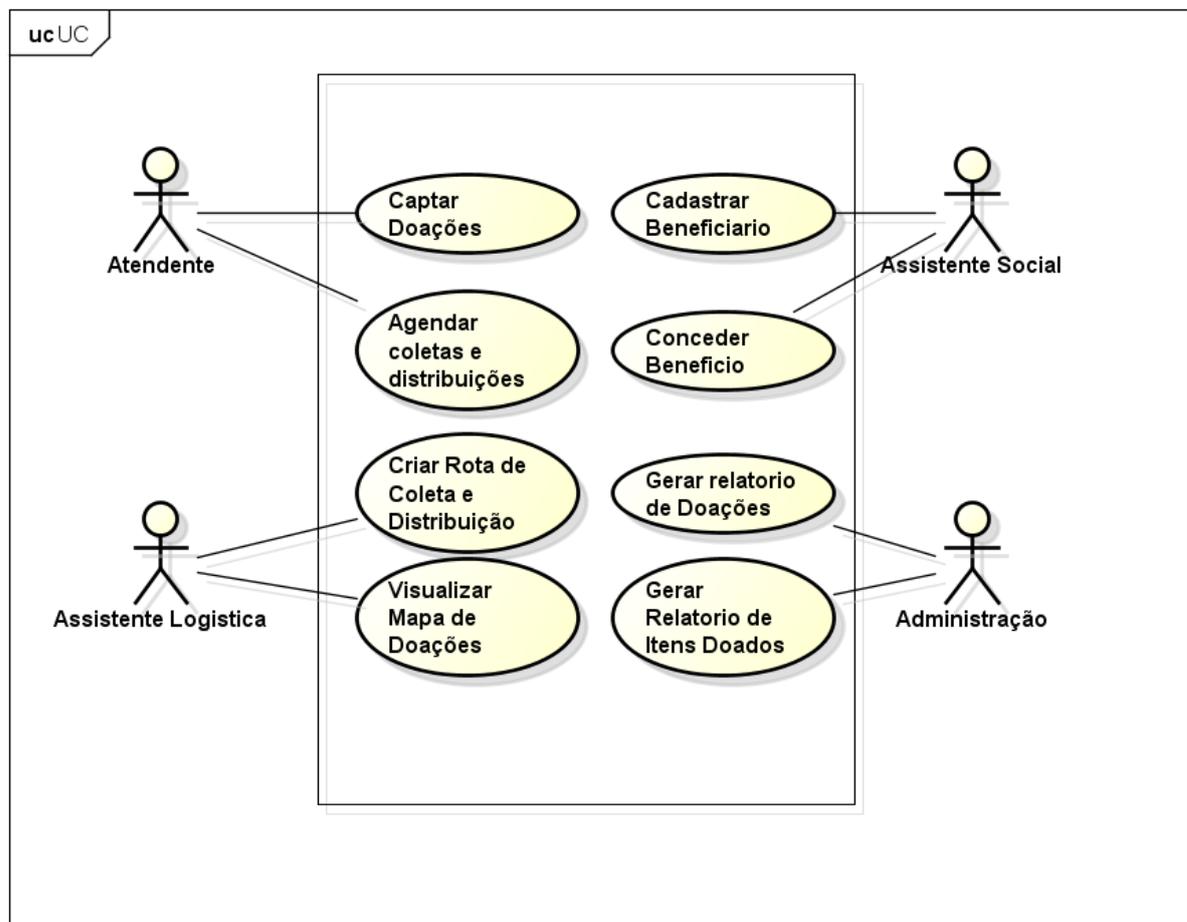


Figura 1: Diagrama de Casos de Uso

2.4.2 Descrições textuais dos atores

As descrições dos atores que fazem participam do diagrama de caso de uso do sistema será detalhada nesta seção:

Atendente: é responsável por cadastrar, incluir itens em doações, agendar recebimento e coleta de doações

Analista de Logística: é responsável por Manter rota de coletas e distribuições e visualizar mapas de doações x beneficiários.

Assistente Social: é responsável por Manter os beneficiários, Manter Doadores e conceder benefícios.

Administração: é responsável por gerar Gráficos de Doações e gerar gráficos de itens doados.

2.5 Descrição textuais dos Casos de Uso

Nesta seção serão descritos os detalhes dos casos de uso que fundamentam o sistema:

CSU 1 – Captar Doações

Sumário: Esse caso de uso permite que o Atendente cadastre doador e inclua itens doação, essa operação gera inclusão de Doador, inclusão itens doados e inclusão de doação.

Ator Principal: Atendente

Fluxo Principal:

1. O Atendente seleciona opção “cadastrar doador” no menu.
2. O sistema direciona o Atendente para a tela de cadastro de doador.
3. O Atendente preenche os dados do Doador e clicar em “Realizar Doação”.
4. O sistema direciona o Atendente para a tela “Cadastrar doação”.
5. O Atendente escolhe o tipo de material que deseja doar.
6. O sistema retorna a lista de itens cadastrados.
7. O Atendente seleciona item cadastrado.
8. O Atendente define a quantidade de itens que será doado.
9. O Atendente seleciona a opção “Incluir item”. O sistema atualiza lista de itens.
10. O Atendente repete os passos 5, 6, 7, 8 e 9 enquanto desejar incluir itens distintos.

11. O Atendente seleciona a opção “Concluir Doação”. O sistema atualiza o cadastro de doadores.

Fluxo Alternativo: (Item 2) O Atendente seleciona a opção “Salvar”

a) O sistema atualiza o cadastro de doadores.

Fluxo Alternativo (Item 5) O tipo de material que o doador deseja doar não está cadastrado no sistema.

- a) O Atendente seleciona a opção “Cadastrar Tipo de Material” no menu.
- b) O Atendente preenche a descrição do tipo de material no campo “Grupo de Material”.
- c) O Atendente seleciona a opção Salvar.
- d) O Atendente seleciona a opção Visualizar Doadores no menu.
- e) O Atendente escolhe o doador na lista de doadores exibida no sistema.
- f) O Atendente escolhe a opção editar, seleciona a opção “Realizar Doação”.
- g) O Atendente repete os passos 10 e 11.

Fluxo Alternativo: (Item 6) O item que o doador deseja doar não está cadastrado no sistema.

- a) O Atendente seleciona a opção “Cadastrar Item” no menu.
- b) O Atendente escolhe o tipo de material que deseja categorizar o item.
- c) O Atendente inclui o nome do item que deseja cadastrar.
- d) O Atendente seleciona a opção Salvar.
- e) O Atendente seleciona a opção “Visualizar Doadores” no menu.
- f) O Atendente seleciona o doador na lista de doadores.
- g) O Atendente escolhe a opção editar, clicar em “Realizar Doação”.
- h) O Atendente repete os passos 10 e 11.

Fluxo Alternativo: (Item 9) O Atendente não escolhe o tipo de material e seleciona a opção “Incluir item”.

a) O sistema exibe mensagem “Tipo de material não selecionado”.

Fluxo Alternativo: (Item 9) O Atendente não escolhe o item e seleciona a opção “Incluir item”.

a) O sistema exibe mensagem “Item Não Incluído”.

Fluxo Alternativo: (Item 9) O Atendente não inclui a quantidade de itens e seleciona a opção “Incluir item”.

a) O sistema exibe a mensagem “Quantidade Inválida”.

Fluxo Alternativo: (Item 11) O Atendente exclui o item da lista de itens doados.

1. O sistema atualiza Lista de Itens Doados.

CSU 2 - Agendar coletas e distribuições

Sumário: Esse caso de uso permite que o atendente agende a data de coleta da doação.

Ator Primário: Atendente

Fluxo principal:

1. O Atendente seleciona a opção “Visualizar Doações” no menu.
2. O Atendente deverá selecionar a Doação
3. O Atendente deverá selecionar a opção Editar.
4. O Atendente deverá Incluir a data de Coleta.
5. O Atendente deverá selecionar a opção Salvar.

Fluxo alternativo:(Item 4) O Atendente escolhe data de coleta para a doação em dia não útil.

- a) Ao selecionar a opção salvar o sistema exibirá mensagem “Data Invalida”.

CSU 3 - Criar rota de coleta e distribuição

Sumário: Esse caso de uso permite que o Analista de logística crie rota entre doações e beneficiários.

Ator Primário: Analista de logística

Fluxo principal:

1. O Analista de logística seleciona a opção “Criar rotas”.
2. O sistema exibe uma lista contendo os endereços de doadores e beneficiários.
3. O Analista de logística seleciona os endereços que deseja criar rota.
4. O Analista de logística seleciona a opção “Verificar rota”.
5. O sistema retorna a visualização do mapa contendo os endereços interligados.
6. O sistema exibe uma lista com os endereços selecionados.
7. O Analista de logística define um nome para a rota.
8. O Analista de logística seleciona a opção “Criar Rota”.
9. O sistema exibe mensagem de confirmação.
10. O Analista de logística seleciona a opção “sim”.
11. O Sistema atualiza o cadastro de rotas

Fluxo alternativo:(Item 4) O Analista de logística seleciona opção “Verificar rota” sem selecionar os endereços.

- a) O sistema exibe mensagem de erro e bloqueia o acesso a opção “Verificar Rota” sem que pelo menos um endereço seja selecionado.

Fluxo alternativo:(Item 8) O Analista de logística seleciona a opção “Criar Rota” sem que o nome da rota seja definido.

- a) O sistema exibe mensagem de erro e bloqueia o acesso a opção “Criar Rota” sem que o nome da rota seja preenchido.

Fluxo alternativo:(Item 10) O Analista de logística seleciona a opção “não”.

- a) O sistema cancela a inclusão da rota.
- b) O sistema retorna a tela “Verificar rota”.

CSU 4 - Visualizar mapa de Doações

Sumário: Esse caso de uso permite que o Analista de logística visualize os mapas com as coordenadas de doadores e beneficiários definidas na tela.

Ator Primário: Analista de logística

Fluxo principal:

1. O Analista de logística seleciona a opção “Visualizar Coordenadas” no menu.
2. O sistema retorna a tela com o mapa contendo os doadores e beneficiários.

CSU 5 - Cadastrar Beneficiário

Sumário: Esse caso de uso permite que o Assistente social inclua um beneficiário.

Ator Primário: Assistente social

Fluxo principal:

1. O Assistente social seleciona a opção “Cadastrar Beneficiário” no menu.
2. O Sistema exibe a tela de cadastro de beneficiários.
3. O Assistente social envia os dados do beneficiário para o sistema.
4. O sistema valida as informações digitadas pelo Assistente Social.
5. O Sistema atualiza o cadastro de beneficiários.

Fluxo alternativo: As informações digitadas pelo Assistente Social não atendem ao padrão esperado pelo sistema.

- a) O sistema exibe mensagem com o campo fora do padrão.
- b) O sistema aguarda a correção das informações.

CSU 6 - Conceder benefício

Sumário: Esse caso de uso permite que o Assistente Social Conceda uma Doação (benefício) a um beneficiário.

Ator Primário: Assistente social

Fluxo principal:

1. O Assistente Social seleciona a opção “Conceder benefício” no menu.
2. O Sistema retorna a tela “Conceder benefício”.
3. O Sistema retorna uma lista de beneficiários e doadores na tela.
4. O Assistente Social seleciona o doador e o beneficiário.
5. O Assistente Social seleciona a opção “Conceder benefício”.
6. O sistema exibe mensagem “Deseja conceder o benefício?”.

7. O Assistente Social seleciona a opção “sim”.
8. O sistema atualiza o cadastro de beneficiários.

Fluxo alternativo: O Assistente Social seleciona a opção “Conceder benefício” sem selecionar o doador e o beneficiário.

- a) O sistema exibe mensagem de que é necessário selecionar um doador e um beneficiário.
- b) O sistema aguarda a seleção do beneficiário e do Doador.

CSU 7 - Gerar relatório de Doações

Sumário: Esse caso de uso permite ao administrador gerar gráficos das doações realizadas por datas.

Ator Primário: Administrador

Fluxo principal:

1. O Administrador seleciona a opção “Gerar Gráficos” no menu.
2. O sistema exibe a tela “Gerar Gráficos”.
3. O Administrador seleciona a data início e a data fim do período de doações.
4. O Administrador seleciona a opção “Gerar”
5. O Sistema exibe o gráfico de doações realizadas por período.

Fluxo alternativo: O Administrador seleciona a opção “Gerar” sem selecionar as datas início e fim.

- a) O sistema exibe mensagem de que o campo data início e fim é necessária.
- b) O sistema aguarda a seleção da data início e fim.

CSU 8 - Gerar relatório de itens Doados

Sumário: Esse caso de uso permite ao administrador gerar gráficos das doações realizadas por datas.

Ator Primário: Administrador

Fluxo principal:

1. O Administrador seleciona a opção “Gerar Gráficos” no menu.
2. O sistema exibe a tela “Gerar Gráficos”.
3. O Administrador seleciona a opção “Gerar Ranking”
4. O sistema exibe o gráfico com o ranking de itens doados.

Fluxo alternativo: O atendente seleciona a opção “Salvar”.

- a) O sistema atualiza o cadastro de doadores.

2.6 Modelo de Classes

O modelo de classe é a representação dos objetos, tendo em vista que o paradigma de desenvolvimento escolhido é o Orientado Objeto, se torna essencial o uso dessa ferramenta.

Após o levantamento de requisitos verificamos os objetos que são pertinentes à aplicação, realizado uma classificação e modelagem, todavia o cerne do sistema não está tão somente nas classes, mas nos relacionamentos gerados através dessa modelagem.

2.6.1 Diagrama de Classes

O diagrama de classes é o resultado do modelo de classes e torna mais fácil a compreensão do cerne do sistema, dos relacionamentos entre os objetos, dos atributos e das operações, tais informações possuem são necessárias para a projeção do sistema.

A Figura 2 demonstra a aplicação do conceito da modelagem de classe em forma de diagrama.

A figura 5 demonstra as classes envolvidas no caso de uso Captar Doações

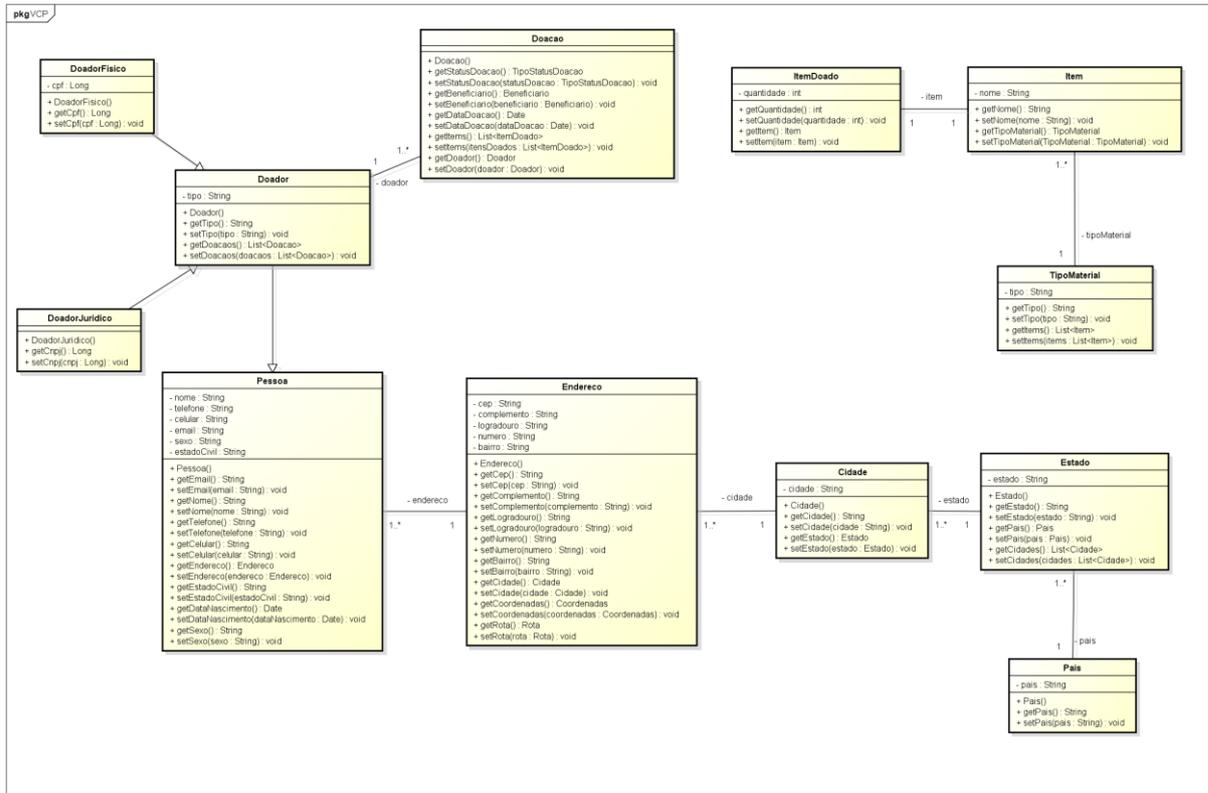


Figura 5: VCP - Captar Doações

A Figura 6 demonstra as classes envolvidas no caso de uso Conceder Benefício

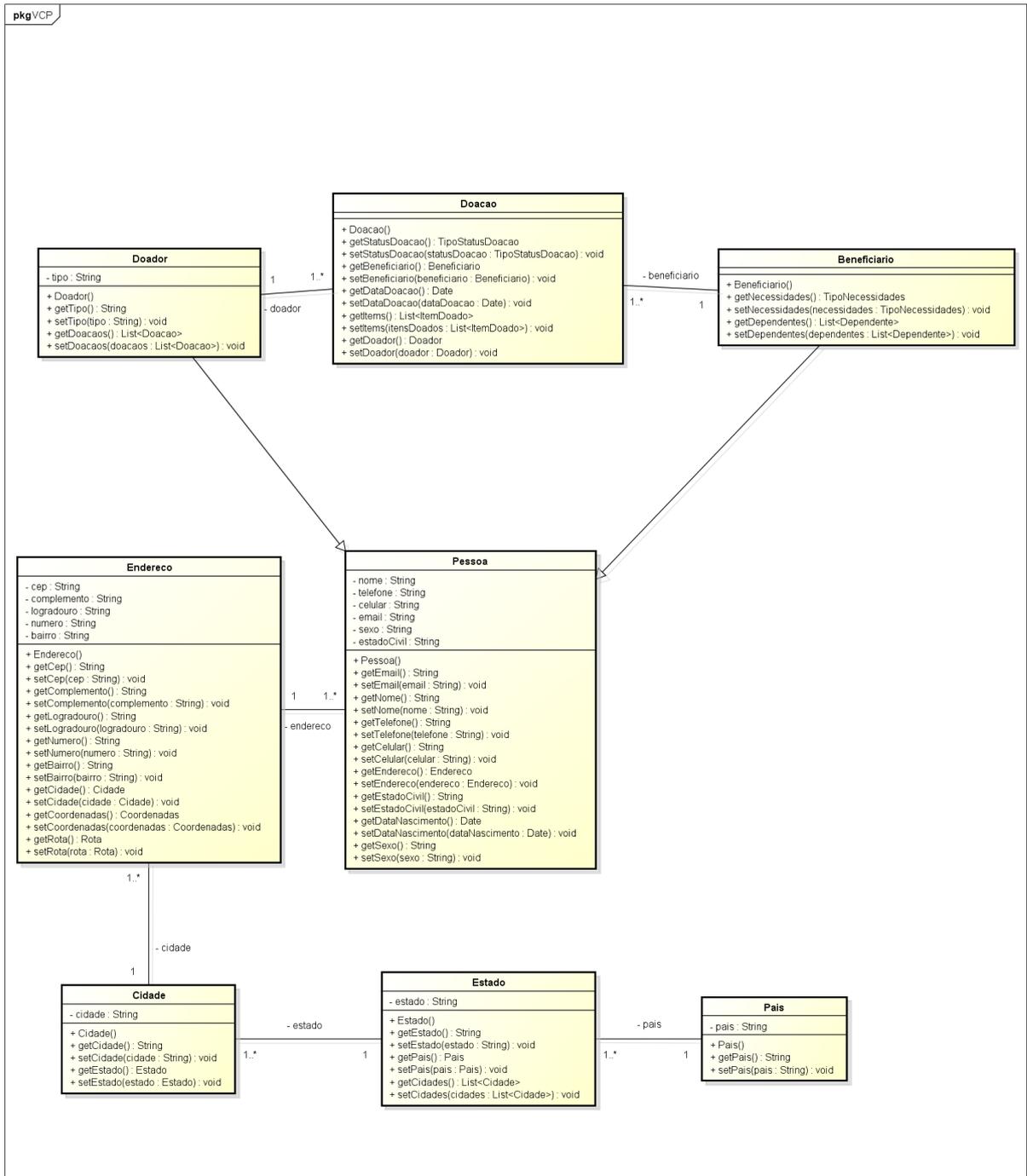


Figura 6: VPC - Conceder Benefício

A figura 7 demonstra as classes envolvidas no caso de uso Criar rota de coleta e distribuição

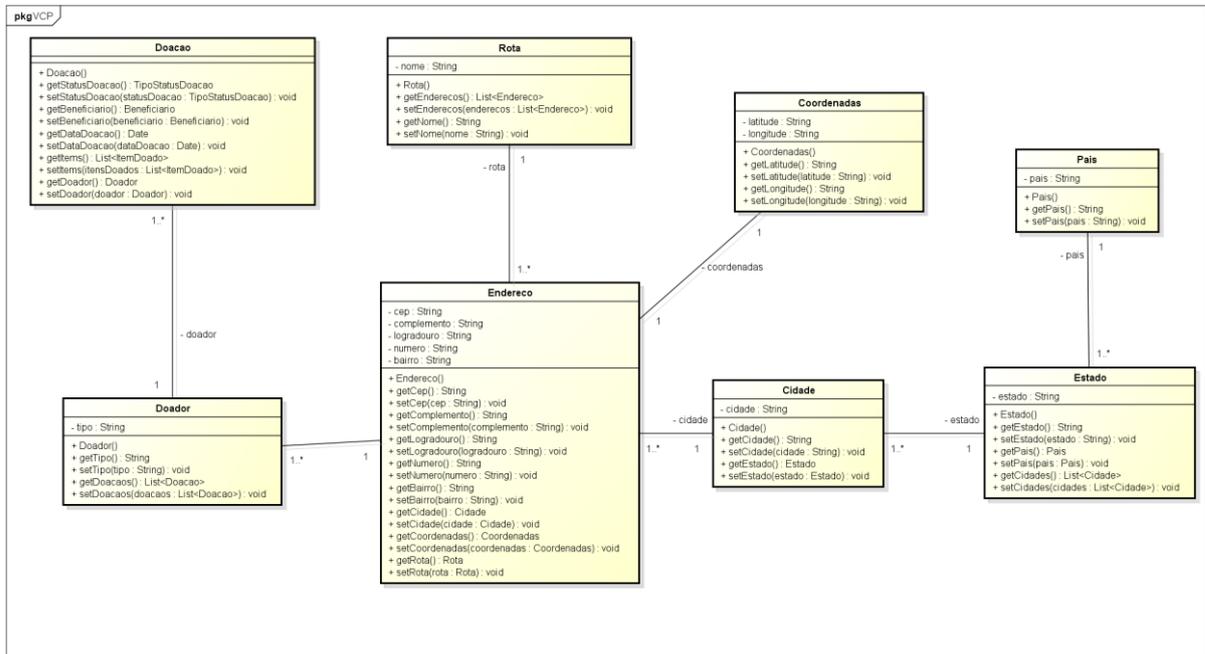


Figura 7: VCP - Criar Rota de Coleta e Distribuição

A figura 8 demonstra as classes envolvidas no caso de uso Gerar Gráficos de doações

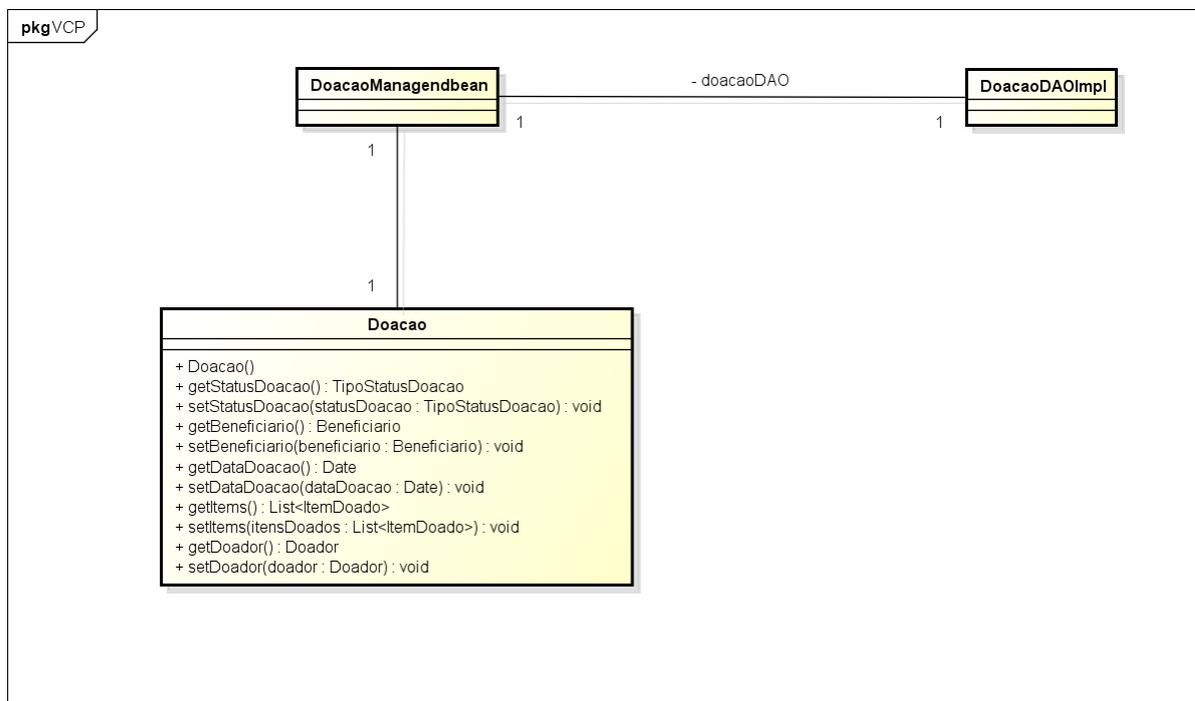


Figura 8: VCP - Gerar Gráficos de Doações

A figura 9 demonstra as classes envolvidas no caso de uso Gerar Gráficos de Itens dados

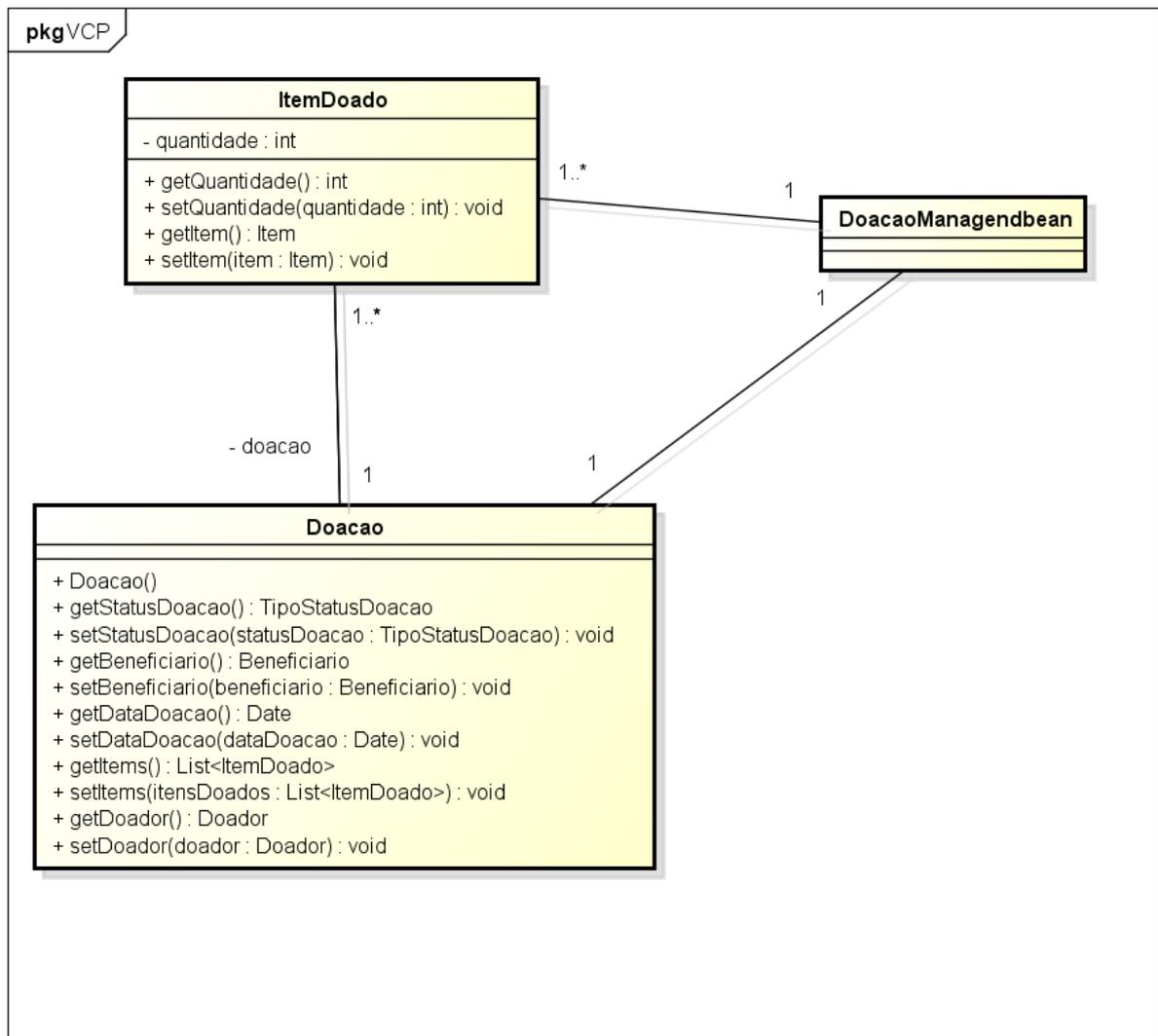


Figura 9: VCP Gerar Gráficos de Itens doados

A figura 10 demonstra as classes envolvidas no caso de uso Visualizar Mapa de Doações

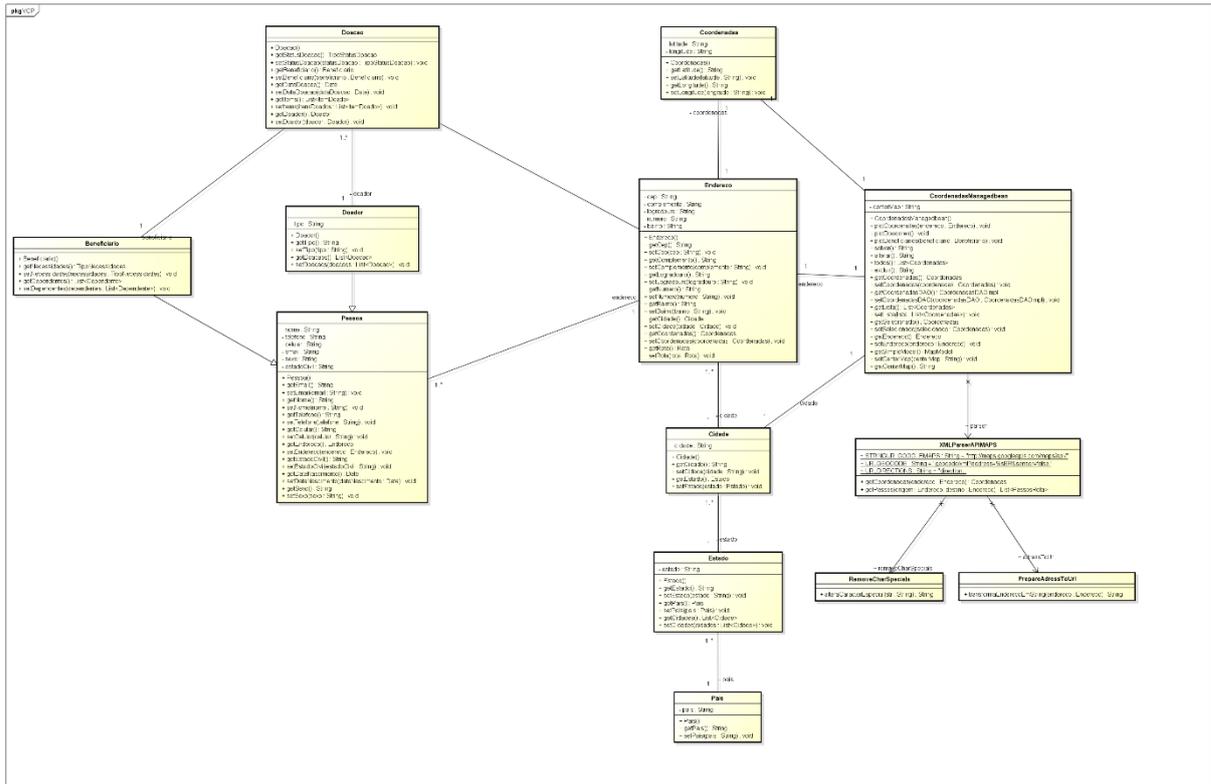


Figura 10: VCP - Visualizar Mapa de Doações

2.8 Modelo de Interações

O modelo de interações tem como funcionalidade demonstrar a interações entre os atores e os objetos que compõe o sistema durante a execução de um caso de uso, tem como orientação duas dimensões, onde a dimensão vertical é o tempo de vida da interação de um objeto e a dimensão horizontal são as interações entre os objetos, ou seja requisições entre os objetos. O diagrama que representa o modelo de interações em UML é o diagrama de sequência.

2.9 Diagramas de sequência de cada cenário relevante dos casos de uso

A ilustração do diagrama se dá pela representação das trocas de mensagens entre objetos na realização de cada cenário de um caso de uso.

A Figura 11 representa o diagrama de sequência referente ao caso de uso Agendar Coletas e Distribuições.

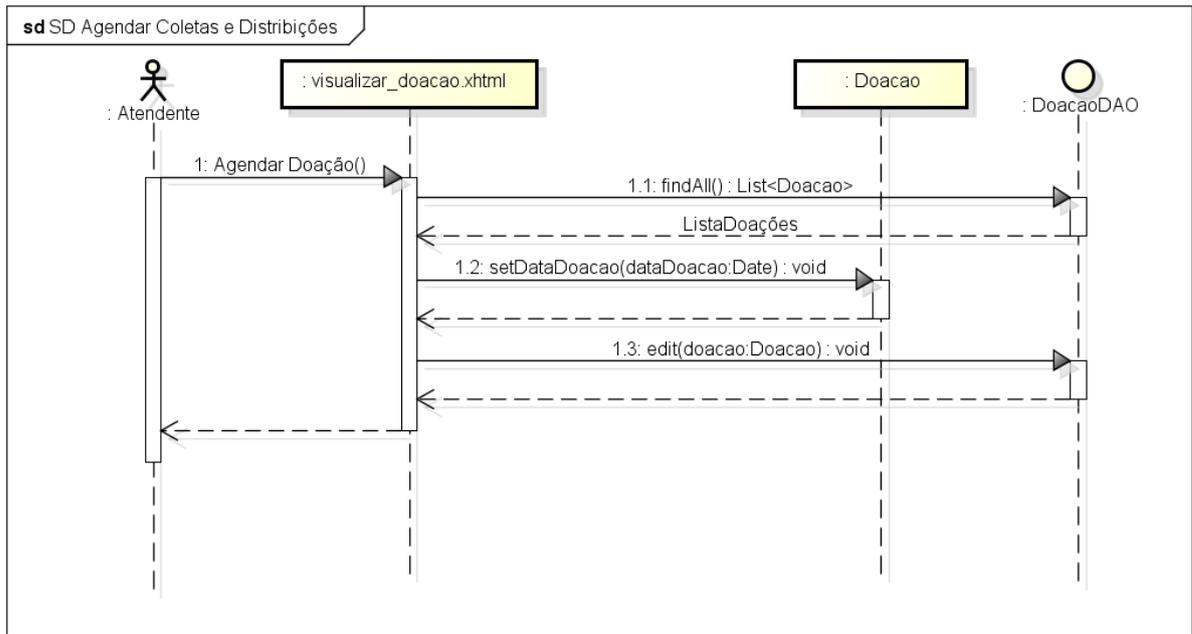


Figura 11: SD - Agendar Coletas e Distribuições

A figura 12 representa o diagrama de seqüência referente ao caso de uso Cadastrar Beneficiários.

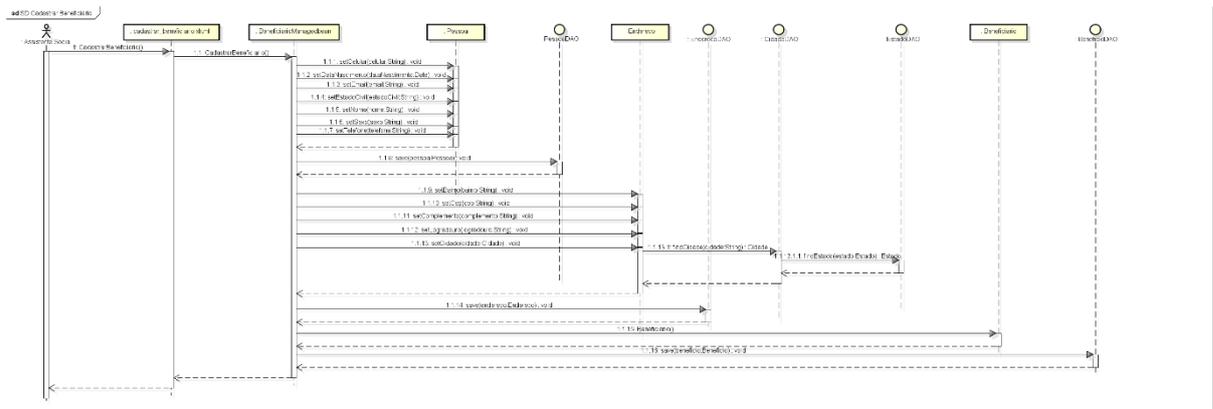


Figura 12: SD - Cadastrar Beneficiário

A figura 13 representa o diagrama de seqüência referente ao caso de uso Captar Doações

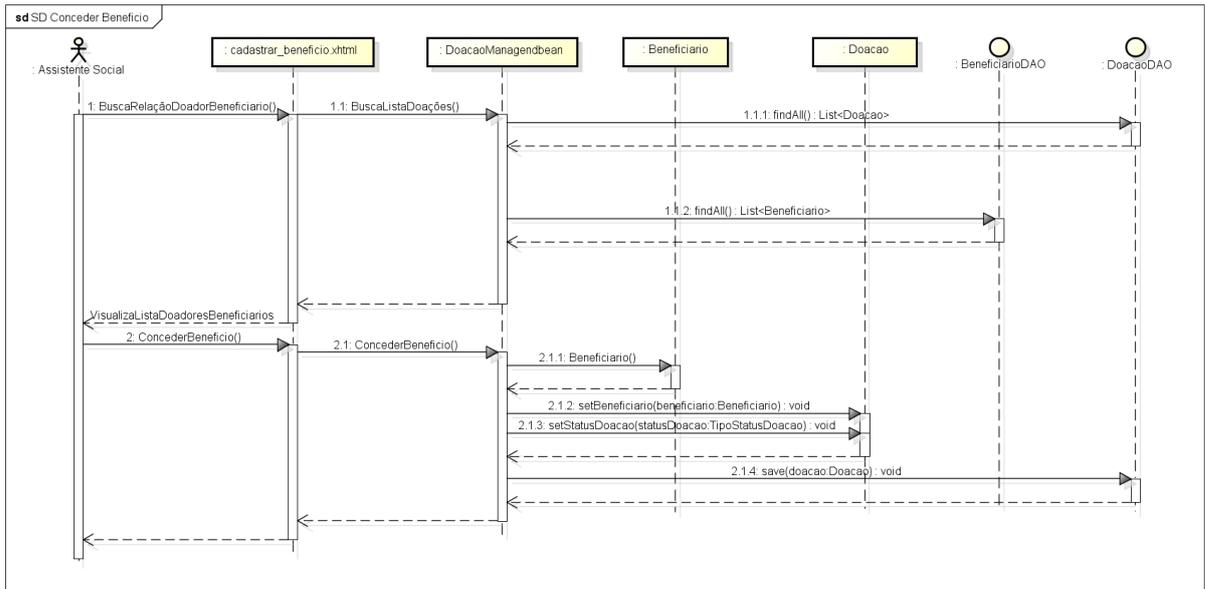


Figura 15: SD - Conceder Benefício

A figura 16 representa o diagrama de seqüência referente ao caso de uso Criar Rota de Coleta e Distribuição

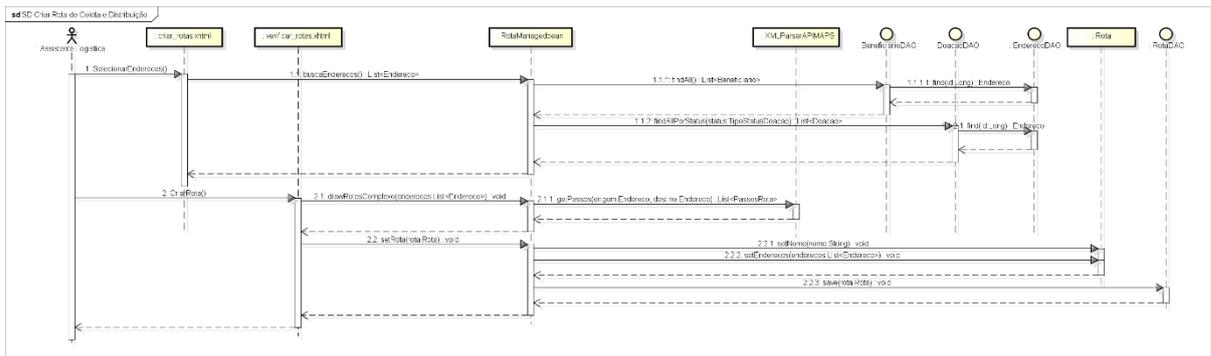


Figura 16: SD - Criar Rota de Coleta e Distribuição

A figura 17 representa o diagrama de seqüência referente ao caso de uso Gerar Gráficos de Doações

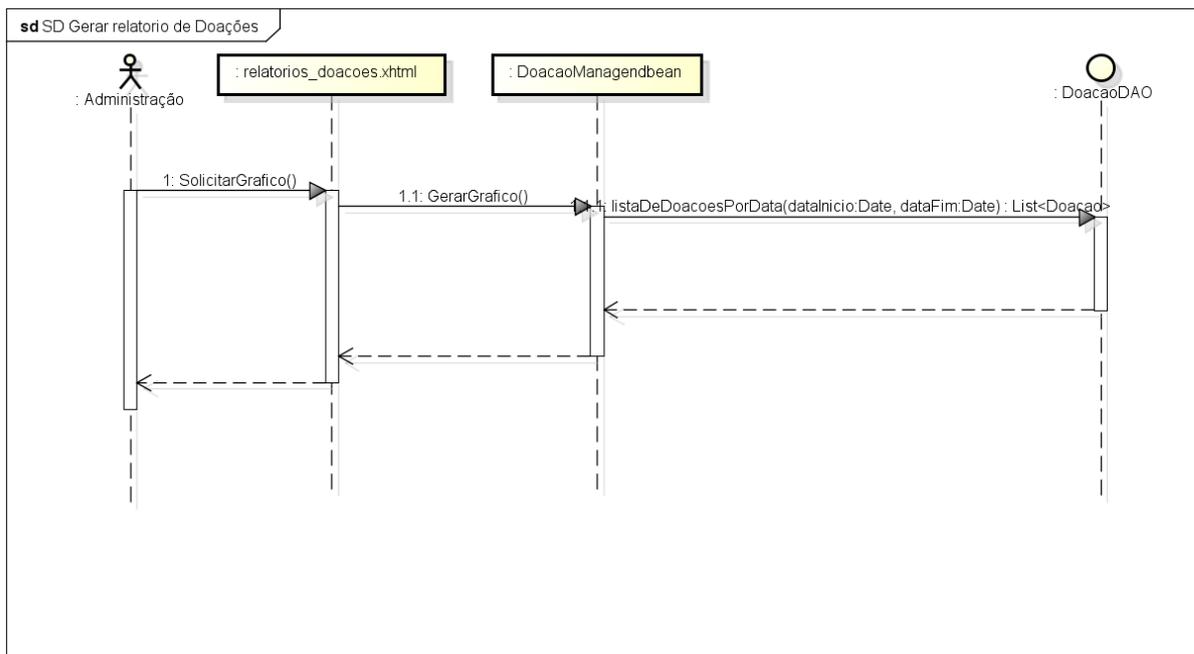


Figura 17: SD - Gerar Gráficos de Doações

A figura 18 representa o diagrama de sequência referente ao caso de uso Gerar Gráficos de Itens Doados.

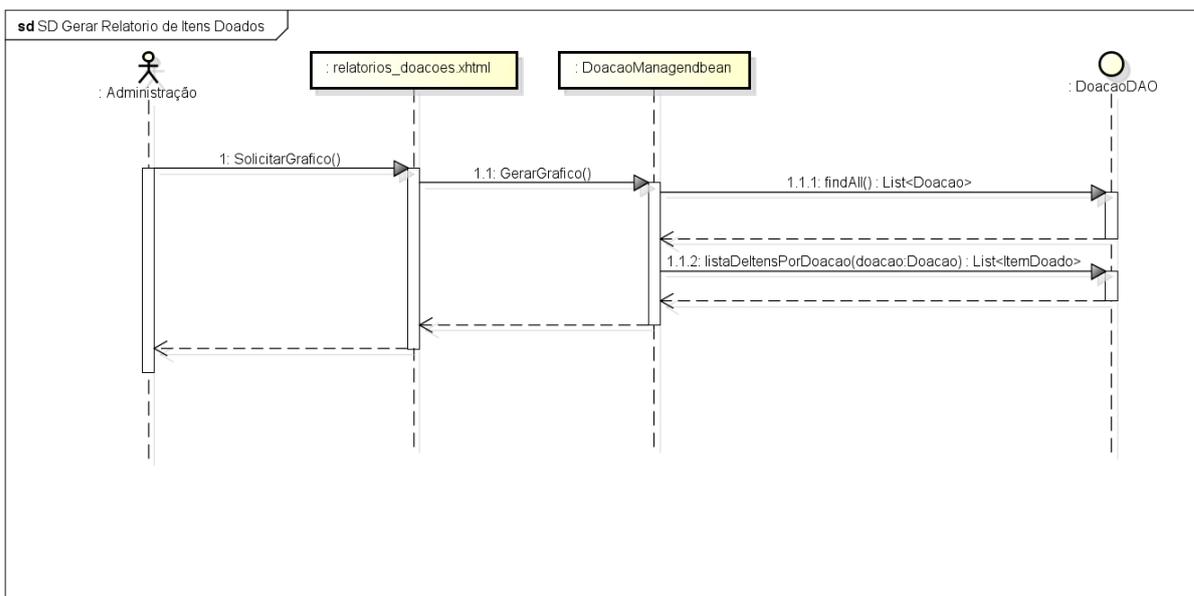


Figura 18: SD - Gerar Gráficos de Itens Doados

A figura 19 representa o diagrama de sequência referente ai caso de uso Visualizar Mapa de Doações

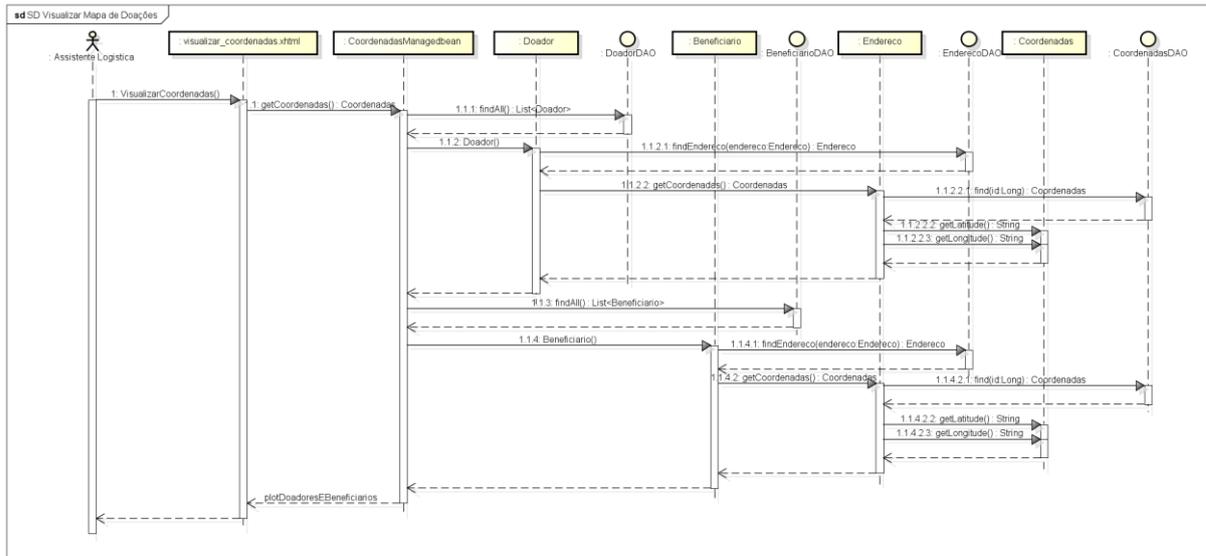


Figura 19: SD - Visualizar Mapa de Doações

2.10 Projeto de Banco de Dados

O projeto de banco de dados é uma atividade de modelagem que pode ser definida pelos níveis conceitual, lógico e físico.

2.11 Projeto Lógico do Banco de Dados

A criação do projeto lógico de um banco de dados é equivalente ao design de um esquema de banco relacional baseado no diagrama de classes conceitual na Figura 2.

A relação entre as classes se reflete em um conjunto de atributos, dos elementos que compõem esse conjunto são criadas chaves primárias, chaves estrangeiras ou a combinação das duas. Para destacar tais informações na listagem de tabelas que virá a seguir, definimos que as chaves primárias estarão em **negrito** e as chaves estrangeiras estarão sublinhado tracejado.

Tabela 1: PL - Tabela assistente_logistica

Tabela assistente_logistica		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES funcionario (id)

Tabela 2: PL - Tabela assistente_social

Tabela assistente_social		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES funcionario (id)

Tabela 3: PL - Tabela assistente_social_beneficiario

Tabela assistente_social_beneficiario		
Atributo	Tipo	Referência
<u>Assistente_Social_id</u>	bigint(20) NOT NULL	FOREIGN KEY (Assistente_Social_id) REFERENCES assistente_social (id)
<u>beneficiários_id</u>	bigint(20) NOT NULL	FOREIGN KEY (beneficiaries_id) REFERENCES }[]beneficiario(id)

Tabela 4: PL - Tabela assistente_social_doacao

Tabela assistente_social_doacao		
Atributo	Tipo	Referência
<u>Assistente_Social_id</u>	bigint(20) NOT NULL	FOREIGN KEY (Assistente_Social_id) REFERENCES assistente_social (id)
<u>doacao_id</u>	bigint(20) NOT NULL	FOREIGN KEY (beneficiaries_id) REFERENCES doacao(id)

Tabela 5: Tabela atendente

Tabela Atendente		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES funcionariol (id)

Tabela 6: PL - Tabela beneficiario

Tabela beneficiário		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES pessoa ;(id) Ç
necessidades	varchar(255) DEFAULT NULL	

Tabela 7: Tabela beneficio

Tabela beneficio		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES pessoa (id)
descricao	Varchar(255) DEFAULT NULL	
databeneficio	datetime DEFAULT NULL	
<u>itemDoacao_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY(itemDoacao_id) REFERENCES tipomaterial(id)
<u>TipoMaterial_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (TipoMaterial_id) REFERENCES tipomaterial (id)

Tabela 8: PL - Tabela cidade

Tabela cidade		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
Cidade	varchar(255) DEFAULT NULL	
<u>estado_id</u>	varchar(20) DEFAULT NULL	FOREIGN KEY (estado_id) REFERENCES estado (id)

Tabela 9: PL - Tabela coordenadas

Tabela coordenadas		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
latitude	varchar(20) DEFAULT NULL	
longitude	varchar(20) DEFAULT NULL	

Tabela 10: PL - Tabela dependente

Tabela dependente		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
nome	varchar(255) DEFAULT NULL	
dataNascimento	date DEFAULT NULL	
sexo	varchar (20) DEFAULT NULL	
grau_parentesco	varchar(20) DEFAULT NULL	
<u>beneficiario_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (beneficiario_id) REFERENCES beneficiario (id)

Tabela 11: PL - Tabela doacao

Tabela doação		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
status_doacao	varchar(255) DEFAULT NULL	
<u>beneficiario_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (beneficiario_id) REFERENCES beneficiario (id)
<u>beneficio_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (beneficio_id) REFERENCES beneficio (id)
dataDoacao	datetime DEFAULT NULL	
<u>doador_id</u>	bigint (20) DEFAULT NULL	FOREIGN KEY (doador_id) REFERENCES doador (id)

Tabela 12: Tabela doacao_itens

Tabela doação_itens		
Atributo	Tipo	Referência
<u>doação_id</u>	bigint(20) NOT NULL	FOREIGN KEY (doacao_id) REFERENCES doacao (id)
<u>item_id</u>	bigint(20) NOT NULL	FOREIGN KEY (item_id) REFERENCES item (id)

Tabela 13: Tabela doador

Tabela doador		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES pessoa (id)
tipo	varchar(255) DEFAULT NULL	

Tabela 14: PL - Tabela doador_fisico

Tabela doador_fisico		
Atributo	Tipo	Referência
id	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES doador (id)
cpf	bigint(20) DEFAUT NULL	

Tabela 15: PL - Tabela doador_juridico

Tabela doador_juridico		
Atributo	Tipo	Referência
id	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES doador (id)
cnpj	bigint(20) DEFAULT NULL	

Tabela 16: PL - Tabela endereco

Tabela endereço		
Atributo	Tipo	Referência
id	bigint(20) NOT NULL AUTO_INCREMENT	
logradouro	varchar(255) DEFAULT NULL	
numero	varchar(255) DEFAULT NULL	
bairro	varchar(255) DEFAULT NULL	
cidade_id	bigint(20) DEFAULT NULL	FOREIGN KEY (cidade_id) REFERENCES cidade (id)
cep	bigint(20) DEFAULT NULL	
complemento	varchar(255) DEFAULT NULL	
coordenadas_id	bigint(20) DEFAULT NULL	FOREIGN KEY (coordenadas_id) REFERENCES coordenadas (id)

Tabela 17: PL - Tabela estado

Tabela estado		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
estado	varchar(255) DEFAULT NULL	
<u>pais_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (pais_id) REFERENCES pais (id)

Tabela 18: PL - Tabela funcionario

Tabela funcionário		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL	FOREIGN KEY (id) REFERENCES pessoa (id)
matricula	varchar (255) DEFAULT NULL	
<u>usuário_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (usuario_id) REFERENCES usuario (id)

Tabela 19: PL - Tabela item

Tabela item		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
nome	varchar(255) DEFAULT NULL	
<u>tipoMaterial_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (tipoMaterial_id) REFERENCES item (id)

Tabela 20: PL - Tabela itemdoado

Tabela itemdoado		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
quantidade	bigint(20) DEFAULT NULL	
<u>item_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (item_id) REFERENCES item (id)

Tabela 21: PL -Tabela menu

Tabela menu		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
descrição	varchar(255) DEFAULT NULL	
view	varchar(255) DEFAULT NULL	
<u>id_sbmenu</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (id_sbmenu) REFERENCES menu (id)

Tabela 22: PL - Tabela menu_perfil

Tabela menu_perfil		
Atributo	Tipo	Referência
<u>perfil_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (perfil_id) REFERENCES perfil (id)
<u>menu_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (menu_id) REFERENCES menu (id)

Tabela 23: PL - Tabela pais

Tabela pais		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
pais	varchar(255) DEFAULT NULL	

Tabela 24: PL - Tabela perfil

Tabela perfil		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
descrição	varchar(255) DEFAULT NULL	

Tabela 25: PL - Tabela perfil_usuario

Tabela perfil_usuario		
Atributo	Tipo	Referência
<u>usuários_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (usuarios_id) REFERENCES usuario (id)
<u>perfil_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (perfil_id) REFERENCES perfil (id)

Tabela 26: PL - Tabela pessoa

Tabela pessoa		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
celular	varchar(255) DEFAULT NULL	
dataNascimento	date DEFAULT NULL	
email	varchar(255) DEFAULT NULL	
nome	varchar(255) DEFAULT NULL	
sexo	varchar(255) DEFAULT NULL	
telefone	varchar(255) DEFAULT NULL	
<u>endereço_id</u>	bigint(20) DEFAULT NULL	FOREIGN KEY (endereço_id) REFERENCES endereço (id)
estadoCivil	varchar(255) DEFAULT NULL	

Tabela 27: PL - Tabela rota

Tabela rota		
Atributo	Tipo	Referência
<u>id</u>	bigint(20) NOT NULL AUTO_INCREMENT	
nome	varchar(255) DEFAULT NULL	

Tabela 28: PL - Tabela rota_enderecos

Tabela rota_enderecos		
Atributo	Tipo	Referência
rota_id	bigint(20) DEFAULT NULL	FOREIGN KEY (rota_id) REFERENCES rota (id)
endereco_id	bigint(20) DEFAULT NULL	FOREIGN KEY (endereco_id) REFERENCES endereco (id)

Tabela 29: PL - Tabela tipomaterial

Tabela tipomaterial		
Atributo	Tipo	Referência
id	bigint(20) NOT NULL AUTO_INCREMENT	
tipo	varchar(255) DEFAULT NULL	

Tabela 30: PL - Tabela usuario

Tabela usuário		
Atributo	Tipo	Referência
id	bigint(20) NOT NULL AUTO_INCREMENT	
login	varchar(255) DEFAULT NULL	UNIQUE KEY
senha	varchar(255) DEFAULT NULL	
perfil_id	bigint(20) DEFAULT NULL	FOREIGN KEY (perfil_id) REFERENCES perfil (id)

2.12 Projeto Físico do Banco de Dados

Conforme o definido no projeto lógico definido na Seção 2.1.11 diagrama abaixo representa a implementação do projeto que visa garantir a integridade do banco, bom desempenho e manutenibilidade do mesmo.

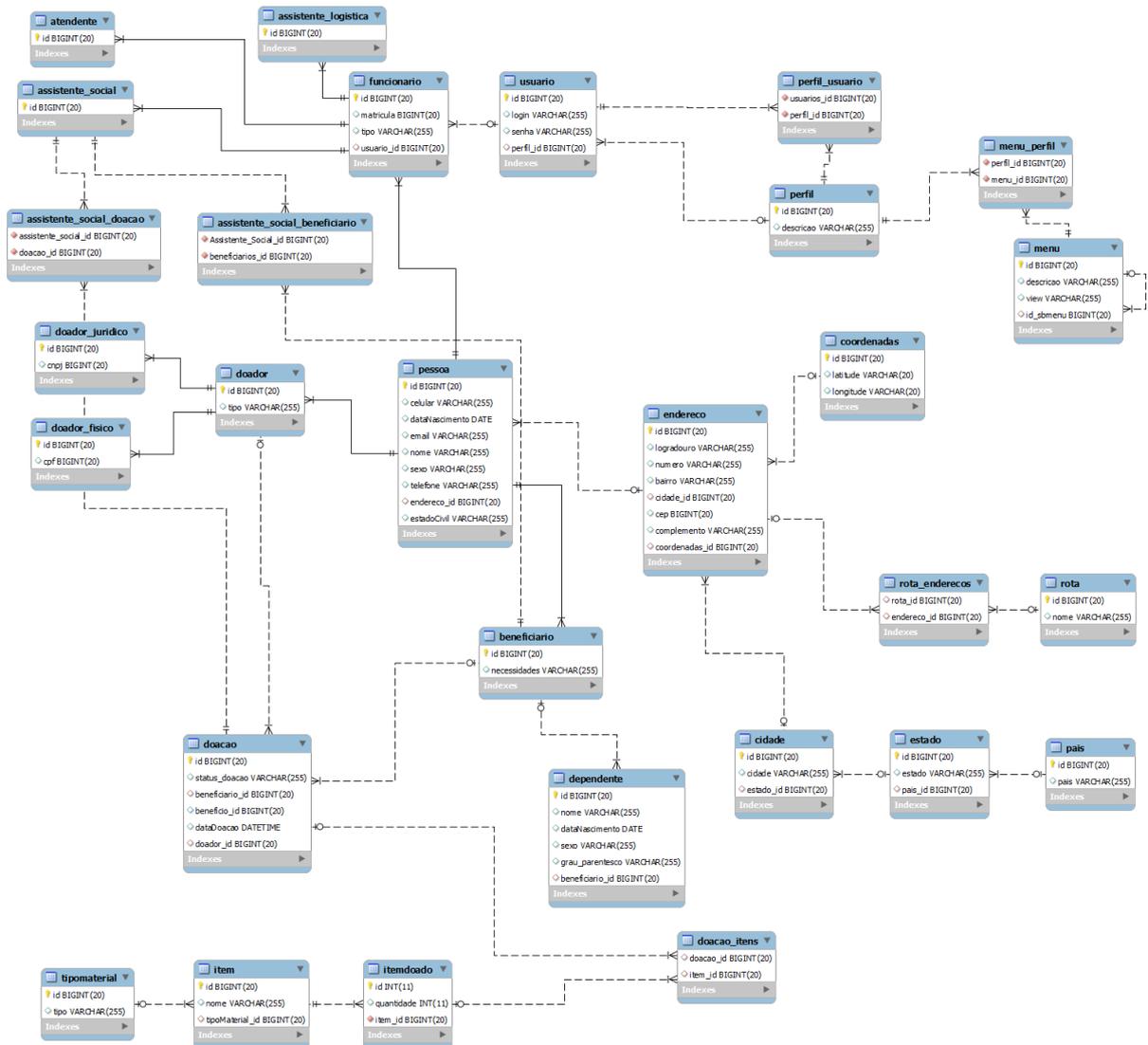


Figura 20: PF - Projeto Físico do Banco de Dados

2.13 Projeto da Interface Gráfica

A interface gráfica é de suma importância para um software tão quando a sua qualidade, para que o sistema possua uma interface amigável e intuitiva, utilizamos de técnicas de usabilidade para que essa interação homem-máquina, sem esquecer é claro que os elementos que compõem essa interface devem atender à necessidade dos usuários.

Após o início da utilização do PrimeFaces para o desenvolvimento do Sistema de doações Trumah, o padrão de interface gráfica foi recomposto adaptando-o ao modelo e métodos de composição de interface gráfica do framework, além de alterações em termos de programação visual. A Figura 21 mostra a tela gerar ranking

de itens doados do Sistema de doações Trumah resultante da interface gráfica com o uso do framework PrimeFaces.

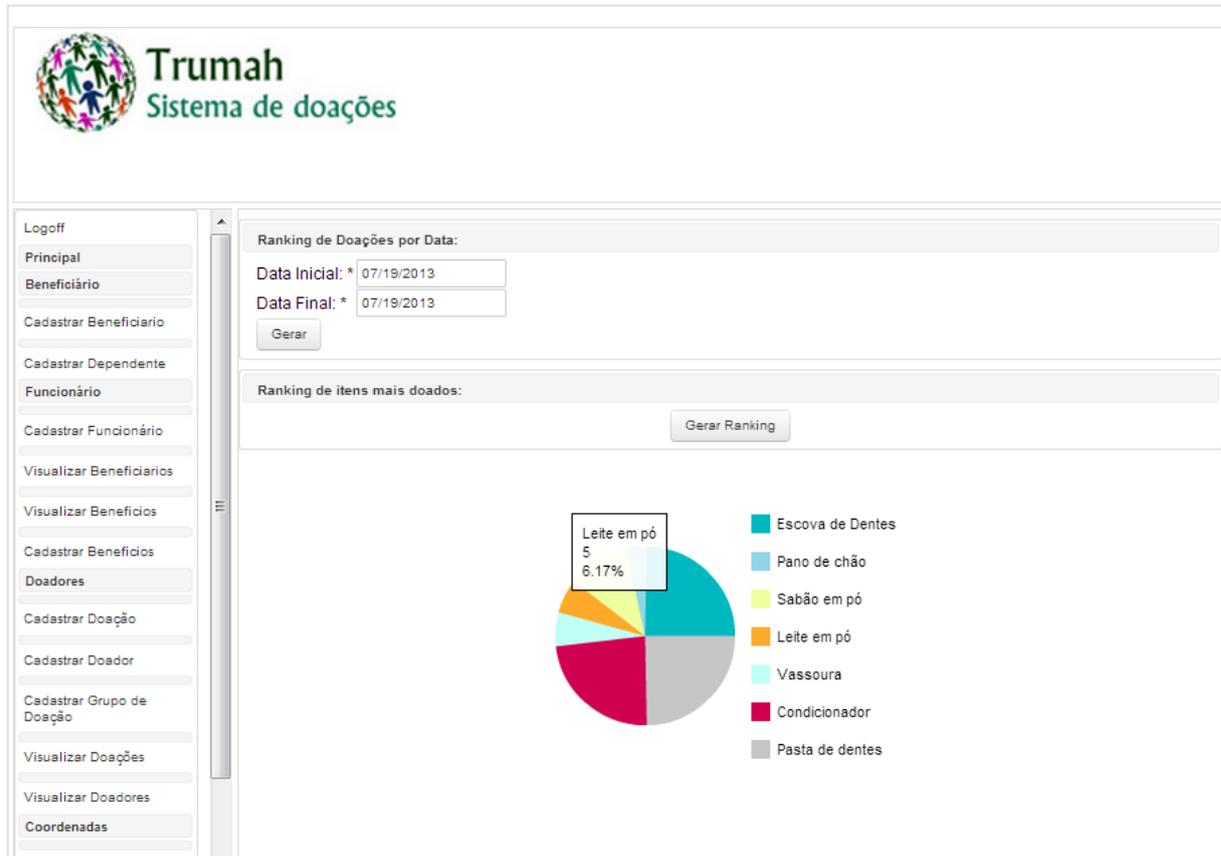


Figura 21: SS - Gerar Ranking Itens Doados

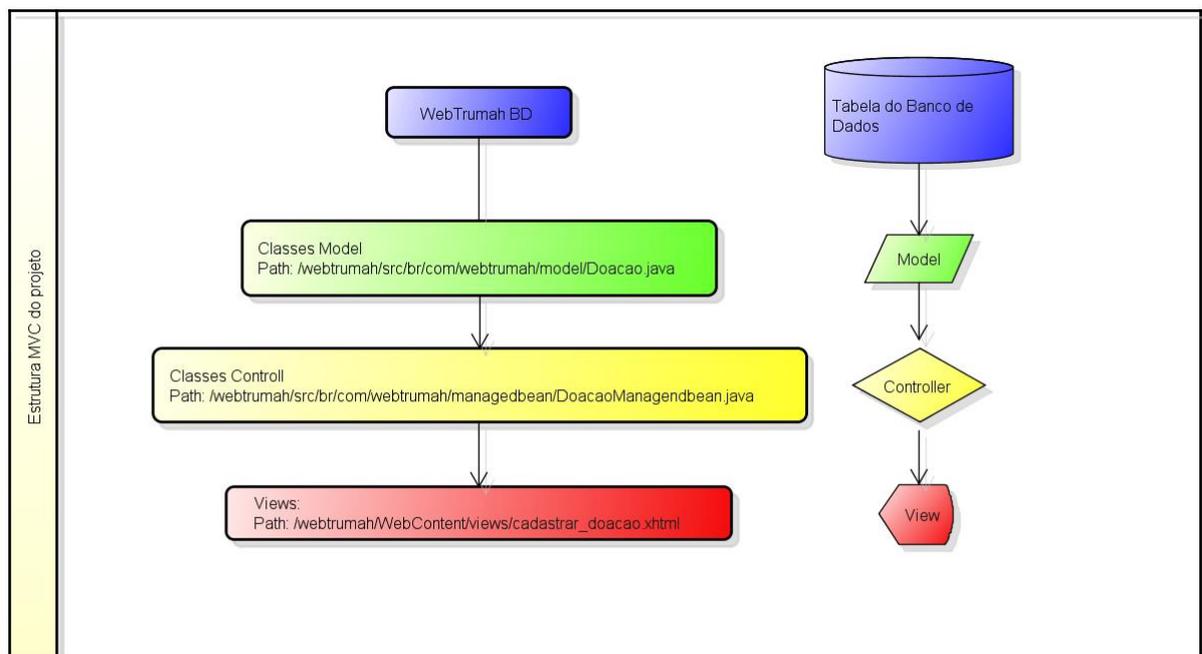
Capítulo III

Aspectos de Implementação

A última etapa do processo de instalação consiste em acessar a estrutura do servidor de aplicação realizar a transferência do arquivo .WAR. Após ser realizada a transferência do arquivo o Tomcat verificará a existência de um novo arquivo web, realizará a descompactação do arquivo ao termino do processo o sistema estará acessível através do acesso `http://<dominio>/webtrumah/`

3.1 Visão geral da arquitetura da aplicação

O sistema foi desenvolvido com o auxílio do framework primefaces, é baseado no modelo MVC, tal modelo é composto por três componentes: Model, View e Controller. O modelo é muito utilizado na atualidade é considerado um padrão de projeto e a interação dos componentes é dividida da seguinte forma:



powered by Astah

Figura 22: FC - Estrutura MVC

Esse modelo de desenvolvimento possui vantagens que evidenciam ainda mais o motivo pelo qual é mais utilizado na atualidade:

- Possui um maior aproveitamento de código.

- É possível uma melhor integração com a equipe.
- O padrão tem sido adotado em diversas tecnologias.
- Manutenibilidade do código evolutiva e corretiva.
- Um código mais limpo.

Junto ao uso do primefaces utilizamos JSF, um outro framework baseado em Java para o desenvolvimento de aplicações web, atualmente faz parte da plataforma Java, e baseia-se em um modelo de desenho de interface de usuário constituída por componentes, obtendo a estrutura de visão de arquivos XML, no caso desse projeto, utiliza-se de um arquivo Xhtml e os pedidos são interpretados através do FacesServlet que por sua vez, traz modelo de visão compatível com o requisitado monta a árvore de componentes, executa os eventos e apresenta a resposta ao cliente.

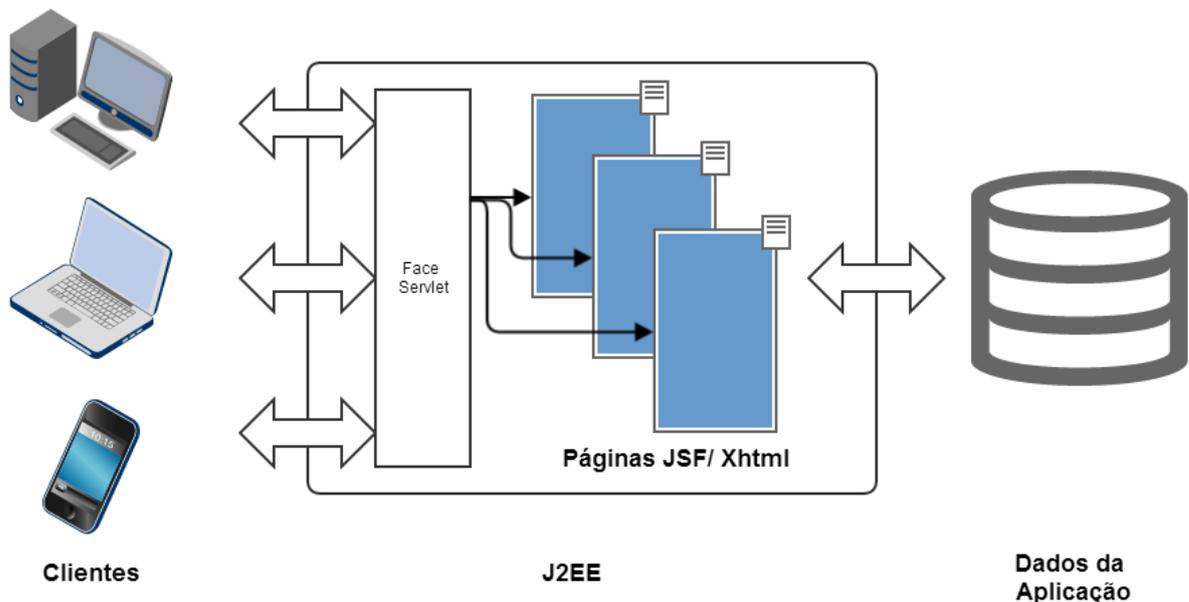


Figura 23: FX - Fluxo JSF

Para o mapeamento objeto relacional e persistência dos dados utilizamos o hibernate, para reduzir o tempo de desenvolvimento e gerenciamento de estado dos objetos manipulados.o hibernate possui três estados, segue a definição abaixo:

Transiente (Transient): é o objeto que foi instanciado através do uso do operador new e não há associação a alguma session no hibernate, esse objeto não existe no banco e não há nenhum identificador, objetos da instância transientes são destruídos pelo garbage se não manter a sua referência.

Persistente (Persistent): O objeto no estado persistente é o objeto que possui representação em banco e um identificador, se encontra no session do hibernate e qualquer mudança que ocorra no escopo do objeto o hibernate realizará uma sincronização do seu estado com o banco de dados.

Destacado (Detached): é o objeto que já está presente no banco de dados e a sua session foi encerrada, porém, a sua referência é válida e a instância fica disponível para usos futuros e outras novas sessions.

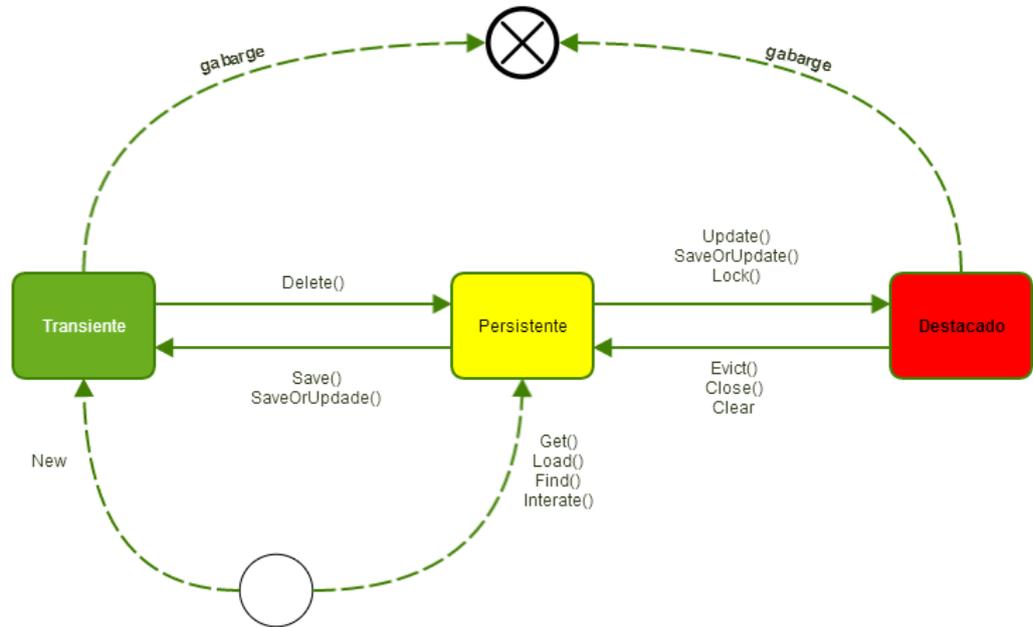


Figura 24: CVH - Ciclo de Vida Estado Hibernate

O uso da api e o webservice da Google se faz necessário para mapeamento de coordenadas, definições de rota e visualização de mapas, pontos cruciais para um sistema de criações de rotas para isso foram utilizadas duas API's, Google Directions API e Google Geocoding API, abaixo a descrição das API's:

Google Directions: é o serviço responsável pelo cálculo das rotas através de requisições HTTP e as rotas podem ser definidas por modo de transporte, ou seja, transportes públicos, ciclovias ou até mesmo caminhos para andar a pé. Para que ocorra a consulta é necessário especificar a origem e o destino, o processo de cálculo de rota consome bastante recurso e por consequência é um processo de resposta relativamente demorada e a saída escolhida para consumo da informação foi XML. A Figura 25 demonstra um trecho da resposta XML

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

▼<DirectionsResponse>
  <status>OK</status>
  ▼<route>
    <summary>ON-401 E</summary>
    ▼<leg>
      ▼<step>
        <travel_mode>DRIVING</travel_mode>
        ▼<start_location>
          <lat>43.6533103</lat>
          <lng>-79.3827675</lng>
        </start_location>
        ▼<end_location>
          <lat>43.6557259</lat>
          <lng>-79.3837332</lng>
        </end_location>
        ▼<polyline>
          <points>e`miGhmocNgBf@cBb@KBkGnB</points>
        </polyline>
        ▼<duration>
          <value>28</value>
          <text>1 min</text>
        </duration>
        ▼<html_instructions>
          Siga na direção <b>norte</b> na <b>Bay St</b> em direção à <b>Hagerman St</b>
        </html_instructions>
        ▼<distance>
          <value>280</value>
          <text>0,3 km</text>
        </distance>
      </step>
      ▼<step>
        <travel_mode>DRIVING</travel_mode>
        ▼<start_location>

```

Figura 25: SS - Google Directions

Google Geocoding: Realiza o processo de conversão de endereço em coordenadas contendo latitude e longitude, o processo inverso (transformar coordenada em endereço) também ocorre, tal processo tem o nome de geocodificação reversa. Da mesma forma que a API Directions o processo consome bastante recurso e a saída escolhida também foi XML.

A Figura 26 demonstra um trecho da resposta XML.

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

▼<GeocodeResponse>
  <status>OK</status>
  ▼<result>
    <type>sublocality</type>
    <type>political</type>
    ▼<formatted_address>
      Porto da Pedra, São Gonçalo - Rio de Janeiro, República Federativa do Brasil
    </formatted_address>
    ▶<address_component>...</address_component>
    ▶<address_component>...</address_component>
    ▶<address_component>...</address_component>
    ▶<address_component>...</address_component>
    ▼<geometry>
      ▼<location>
        <lat>-22.8212888</lat>
        <lng>-43.0644088</lng>
      </location>
      <location_type>APPROXIMATE</location_type>
      ▼<viewport>
        ▼<southwest>
          <lat>-22.8249604</lat>
          <lng>-43.0798196</lng>
        </southwest>
        ▼<northeast>
          <lat>-22.8158693</lat>
          <lng>-43.0569944</lng>
        </northeast>
      </viewport>
      ▼<bounds>
        ▼<southwest>
          <lat>-22.8249604</lat>
          <lng>-43.0798196</lng>
        </southwest>
      </bounds>
    </geometry>
  </result>
</GeocodeResponse>

```

Figura 26: SS - Google Geocoding

3.2 Estruturação da camada de apresentação

Para melhorar a produtividade, minimizar os riscos de erros, e tornar o desenvolvimento do Sistema de Doações Trumah mais agradável foi utilizado o framework primefaces pelo simples fato do mesmo seguir a especificação JSF e otimizar a construção da camada de apresentação com características acentuadas, onde tem como cerne uso da interface para como uso intermediário entre o sistema e usuário. Há vários frameworks para interface voltados para mapeamento com o banco e entre outros, contudo as características marcantes do framework o deixou em extrema importância para o sistema em questão, são elas:

1. Suíte open source de componente para integração JSF 2.0 mais de 100 componente completos de fácil implementação[12];
2. Componentes utilizam ajax nativo;
3. Atualização intensa do JQuery embutido nos diversos themes;
4. Tamanho da bibliografia reduzida;
5. Criação de interfaces amigáveis;
6. Maior nível de abstração e compatibilidade entre aplicações;

7. Documentação detalhada, exemplos com código e bem elaborado no site oficial www.primefaces.org.

Para utilizar o primeface basta utilizar chamada no código como descrito no exemplo abaixo:

A Figura 27 demonstra como é realizado uma chamada do framework PrimeFaces em uma página xhtml.

```

1 <html xmlns="http://www.w3.org/1999/xhtml"
2   xmlns:h="http://java.sun.com/jsf/html"
3   xmlns:ui="http://java.sun.com/jsf/facelets"
4   xmlns:p="http://primefaces.prime.com.tr/ui"
5   xmlns:f="http://java.sun.com/jsf/core">
6
7 <ui:decorate template="/views/home.xhtml">
8   <ui:define name="centro">
9     <h:form prependId="false">
10      <div align="center">
11        <h:panelGrid>
12          <p:spacer height="20"/>
13          <p:dataGrid id="ListaDoacoes"
14            rows="10"
15            value="#{doacaoManagedbean.ListaDoacoesDataModel}"
16            columns="1"
17            var="currentRow"
18            paginator="true"
19            paginatorPosition="bottom"
20            paginatorTemplate="{CurrentPageReport}
21              {FirstPageLink}
22              {PreviousPageLink}
23              {PageLinks}
24              {NextPageLink}
25              {LastPageLink}
26              {RowsPerPageDropdown}"
27            rowsPerPageTemplate="10,15,20">
28          <p:panel header="Doacao: #{currentRow.id} - #{currentRow.doador.nome}" style="text-align:left">
29            <p:dataTable value="#{currentRow.itensDoados}"
30              var="gridItensDoados">
31              <p:column id="itGridcolumn2" headerText="Item Doado">
32                <div align="center"><h:outputText id="outIGriditNomeDtl" value="#{gridItensDoados.item.nome}" /></div>
33              </p:column>
34              <p:column id="itGridcolumn3" headerText="Quantidade">
35                <div align="center"><h:outputText id="outGridIttdDtl" value="#{gridItensDoados.quantidade}" /></div>
36              </p:column>
37            </p:dataTable>
38          </p:panel>
39        </h:panelGrid>
40      </div>
41    </h:form>
42  </ui:define>
43 </ui:decorate>

```

Figura 27: SS - Exemplo de Uso Prime Faces

Primefaces é constituído de 3 módulos:

- UI
- Components
- Optimus e Faces Trace

A Figura 28 representa a estrutura do primeface

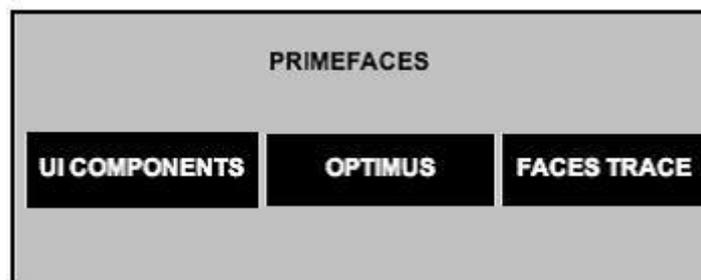
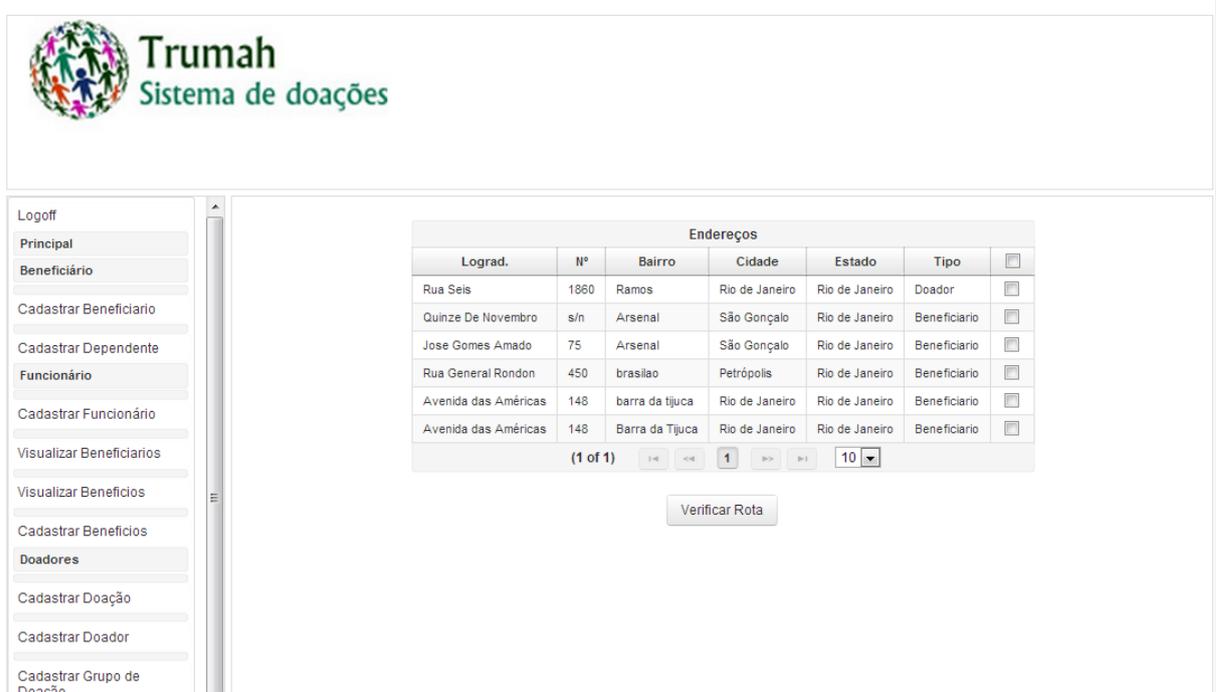


Figura 28: Primefaces Módulos

1. UI Components: São os componentes que contêm as funcionalidades embutidas de Ajax, Javascript e gráficos animados.
2. Optimus: Oferece soluções para integrações no processo de desenvolvimento com JSF.

3. FacesTrace: São funções relacionadas a performance da aplicação.

A figura 29 representa a estrutura do Primeface



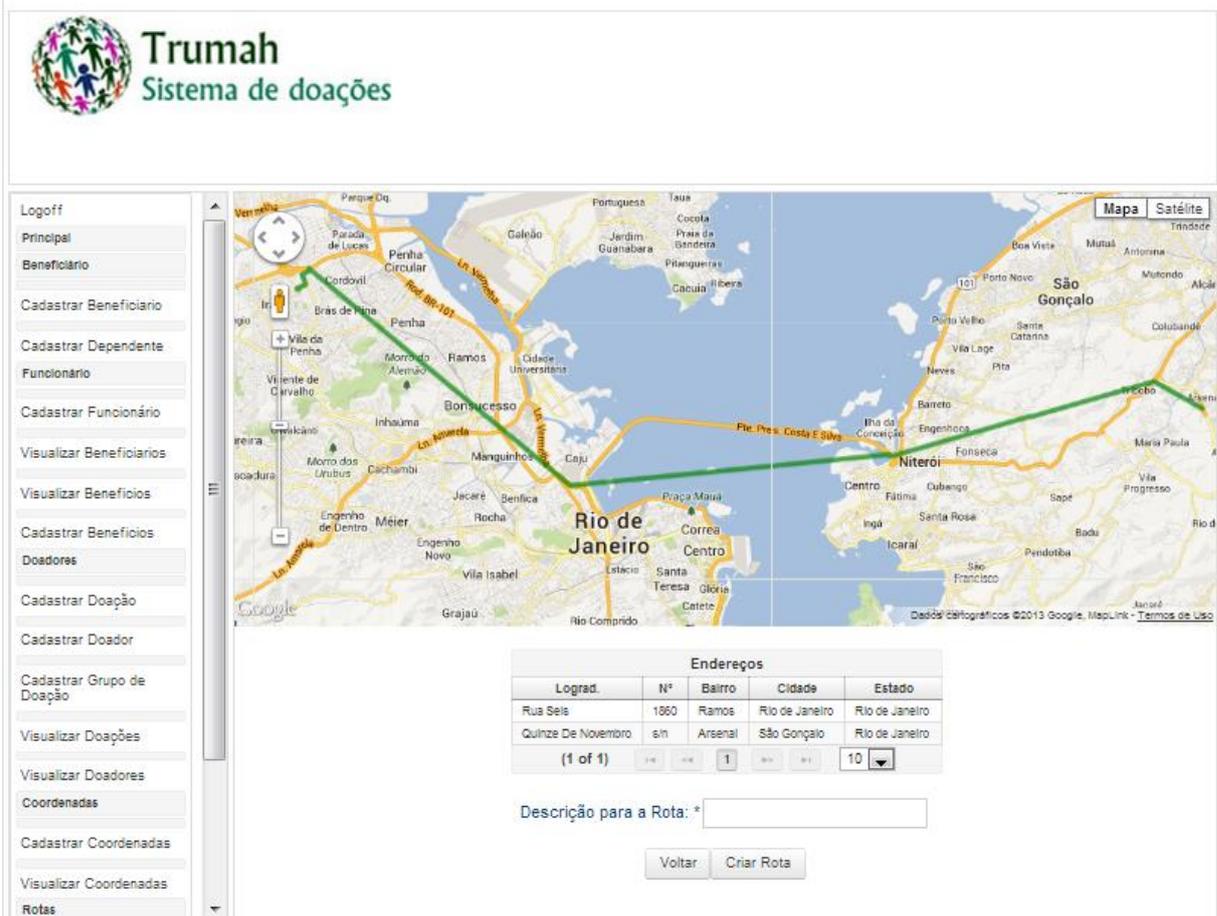
The screenshot displays the Trumah Sistema de doações interface. On the left is a navigation menu with options like 'Logoff', 'Principal', 'Beneficiário', 'Cadastrar Beneficiário', 'Cadastrar Dependente', 'Funcionário', 'Cadastrar Funcionário', 'Visualizar Beneficiários', 'Visualizar Benefícios', 'Cadastrar Benefícios', 'Doadores', 'Cadastrar Doação', 'Cadastrar Doador', and 'Cadastrar Grupo de Doação'. The main content area features a table titled 'Endereços' with the following data:

Lograd.	Nº	Bairro	Cidade	Estado	Tipo	<input type="checkbox"/>
Rua Seis	1860	Ramos	Rio de Janeiro	Rio de Janeiro	Doador	<input type="checkbox"/>
Quinze De Novembro	s/n	Arsenal	São Gonçalo	Rio de Janeiro	Beneficiário	<input type="checkbox"/>
Jose Gomes Amado	75	Arsenal	São Gonçalo	Rio de Janeiro	Beneficiário	<input type="checkbox"/>
Rua General Rondon	450	brasiao	Petrópolis	Rio de Janeiro	Beneficiário	<input type="checkbox"/>
Avenida das Américas	148	barra da tjuca	Rio de Janeiro	Rio de Janeiro	Beneficiário	<input type="checkbox"/>
Avenida das Américas	148	Barra da Tjuca	Rio de Janeiro	Rio de Janeiro	Beneficiário	<input type="checkbox"/>

Below the table is a pagination control showing '(1 of 1)', navigation arrows, a page number '1', and a dropdown menu set to '10'. A 'Verificar Rota' button is located below the pagination controls.

Figura 29: SS - Criar Rota

A figura 30 representa a estrutura do Primeface



Trumah
Sistema de doações

Logoff
Principal
Beneficiário
Cadastrar Beneficiário
Cadastrar Dependente
Funcionário
Cadastrar Funcionário
Visualizar Beneficiários
Visualizar Benefícios
Cadastrar Benefícios
Doadores
Cadastrar Doação
Cadastrar Doador
Cadastrar Grupo de Doação
Visualizar Doações
Visualizar Doadores
Coordenadas
Cadastrar Coordenadas
Visualizar Coordenadas
Rotas

Endereços				
Lograd.	Nº	Bairro	Cidade	Estado
Rua Sels	1860	Ramos	Rio de Janeiro	Rio de Janeiro
Quilze De Novembro	s/n	Arsenal	São Gonçalo	Rio de Janeiro

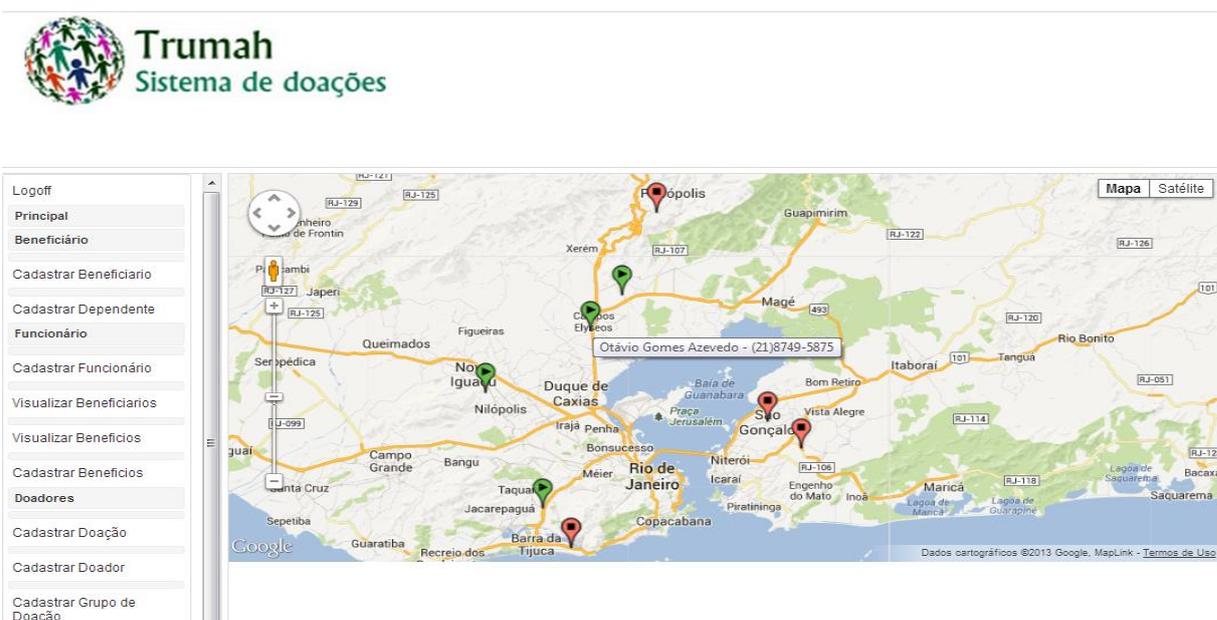
(1 of 1) 1 10

Descrição para a Rota: *

Voltar Criar Rota

Figura 30: Tela de Verificação de Rota Criada

A figura 31 representa a estrutura do Primeface



Trumah
Sistema de doações

Logoff
Principal
Beneficiário
Cadastrar Beneficiário
Cadastrar Dependente
Funcionário
Cadastrar Funcionário
Visualizar Beneficiários
Visualizar Benefícios
Cadastrar Benefícios
Doadores
Cadastrar Doação
Cadastrar Doador
Cadastrar Grupo de Doação

Mapa Satélite

Otávio Gomes Azevedo - (21)8749-5875

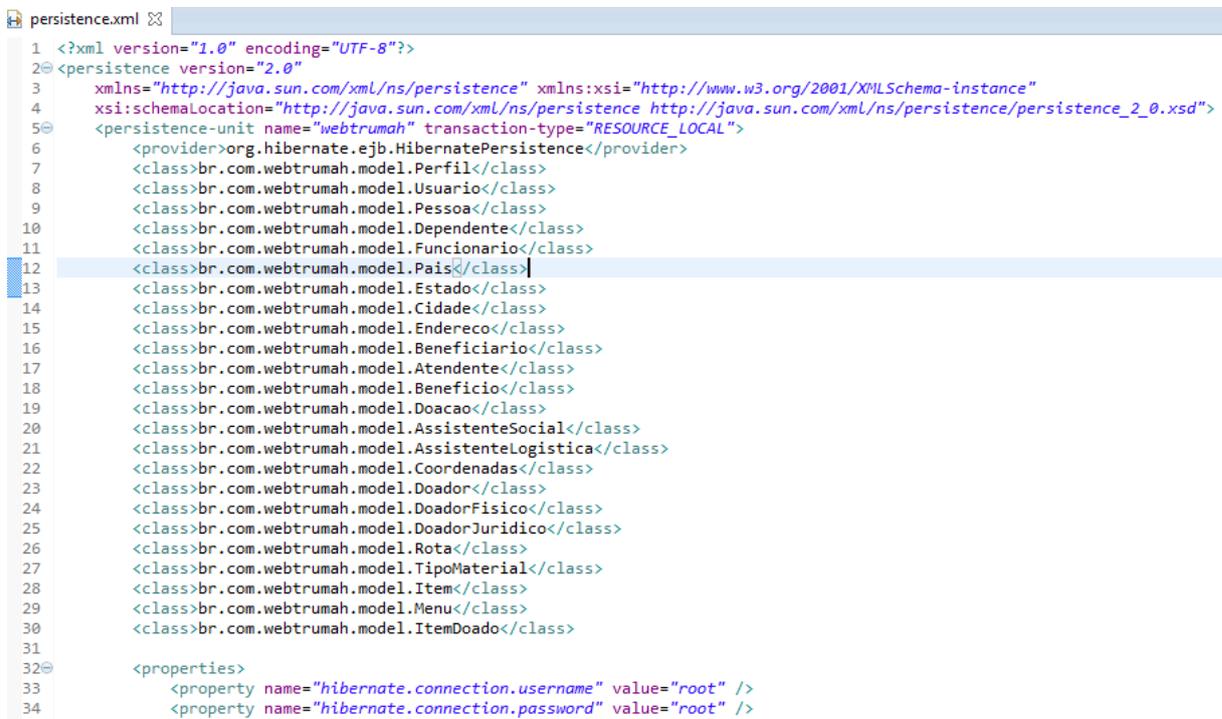
Figura 31: Tela de Visualização de coordenadas de Doadores x Beneficiários

3.3 Estruturação da camada de Negócio

A lógica da camada de negócio está contida nas classes de modelo, que por sua vez compõe o padrão MVC. Esses modelos são manipulados pelo framework Hibernate que é um framework ORM, que tem por função realizar o mapeamento entre o modelo orientado objeto e modelo relacional, pode ser feito através de um arquivo xml ou anotações, o Sistema de Doações Trumah usa anotações para tal mapeamento.

Para configurar o framework Hibernate é necessário que crie um arquivo chamado persistence.xml

A figura 32 demonstra o uso do arquivo de configuração do Hibernate



```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <persistence version="2.0"
3     xmlns="http://java.sun.com/xml/ns/persistence" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:schemaLocation="http://java.sun.com/xml/ns/persistence http://java.sun.com/xml/ns/persistence/persistence_2_0.xsd">
5   <persistence-unit name="webtrumah" transaction-type="RESOURCE_LOCAL">
6     <provider>org.hibernate.ejb.HibernatePersistence</provider>
7     <class>br.com.webtrumah.modelo.Perfil</class>
8     <class>br.com.webtrumah.modelo.Usuario</class>
9     <class>br.com.webtrumah.modelo.Pessoa</class>
10    <class>br.com.webtrumah.modelo.Dependente</class>
11    <class>br.com.webtrumah.modelo.Funcionario</class>
12    <class>br.com.webtrumah.modelo.Pais</class>
13    <class>br.com.webtrumah.modelo.Estado</class>
14    <class>br.com.webtrumah.modelo.Cidade</class>
15    <class>br.com.webtrumah.modelo.Endereco</class>
16    <class>br.com.webtrumah.modelo.Beneficiario</class>
17    <class>br.com.webtrumah.modelo.Atendente</class>
18    <class>br.com.webtrumah.modelo.Beneficio</class>
19    <class>br.com.webtrumah.modelo.Doacao</class>
20    <class>br.com.webtrumah.modelo.AssistenteSocial</class>
21    <class>br.com.webtrumah.modelo.Assistentelogistica</class>
22    <class>br.com.webtrumah.modelo.Coordenadas</class>
23    <class>br.com.webtrumah.modelo.Doador</class>
24    <class>br.com.webtrumah.modelo.DoadorFisico</class>
25    <class>br.com.webtrumah.modelo.DoadorJuridico</class>
26    <class>br.com.webtrumah.modelo.Rota</class>
27    <class>br.com.webtrumah.modelo.TipoMaterial</class>
28    <class>br.com.webtrumah.modelo.Item</class>
29    <class>br.com.webtrumah.modelo.Menu</class>
30    <class>br.com.webtrumah.modelo.ItemDoado</class>
31
32    <properties>
33      <property name="hibernate.connection.username" value="root" />
34      <property name="hibernate.connection.password" value="root" />

```

Figura 32: SS - Persistence XML

3.4 Estruturação da camada de Persistência

Para guardar o valor dos atributos proveniente dos objetos que integram o sistema, assim como o controle de transações e suas operações de criação, leitura, atualização e remoção dos dados, o Sistema de Doação de Bens utiliza um mecanismo de armazenamento para armazenar dados de um SGBD (Sistema de Gerência de Banco de Dados), devido ao uso da tecnologia não somente no meio corporativo como também para fins acadêmicos, além de estar no grupo de

ferramentas opensource, é respeitado como uma das tecnologias que ganha força no mercado com performance aceitável. Todavia a tecnologia de orientação a objetos e a tecnologia de banco de dados relacional, seguem lado a lado com grau de impedância entre as duas vias, ferramentas computacionais que contornam este descasamento de informações foram criadas com a finalidade de facilitar o processo de desenvolvimento em lidar direto com o SQL manipulando modelo relacional. O Sistema de Doações de Bens usa JPA [16] versão 2.0 e hibernate versão 3.5

A figura 33 demonstra um trecho da estrutura da classe DoacaoDAOImpl, responsável pela manipulação e persistência da classe Doador



```
1 package br.com.webtrumah.dao.impl;
2
3 import java.util.Date;
14
15 public class DoacaoDAOImpl implements DoacaoDAO {
16
17     private EntityManager em;
18     private EntityTransaction trans;
19
20     public DoacaoDAOImpl() {
21         em = Persistence.createEntityManagerFactory("webtrumah").createEntityManager();
22         trans = em.getTransaction();
23     }
24
25     public void initialTransation(){
26         trans.begin();
27     }
28
29     public void commitTransation(){
30         trans.commit();
31     }
32
33     @Override
34     public void save(Doacao doacao) {
35         initialTransation();
36         em.persist(doacao);
37         commitTransation();
38     }
39
40     @Override
41     public Doacao find(Long id) {
42         initialTransation();
43         Doacao p = (Doacao) em.find(Doacao.class, id);
44         commitTransation();
45         return p;
46     }
}
```

Figura 33: SS – DoacaoDAOImpl

Capítulo IV

Conclusão

As conclusões sobre o desenvolvimento do sistema de doações de bens são referenciadas nesse capítulo. As considerações sobre o trabalho desenvolvido é apresentada na Seção 4.1 Os trabalhos futuros a serem desenvolvidos a partir desta versão do sistema são apresentados na Seção 4.2

4.1 *Análise Retrospectiva*

Ao decorrer do desenvolvimento do Sistema de Doações, obtivemos uma grande oportunidade para adquirirmos novos conhecimentos, seja na área de exatas ou na área de humanas. Na área de exatas, nossa área, como exemplo tivemos que buscar conhecimento em Xpath [20] e uma funcionalidade Java pouco divulgada no meio acadêmico assim como a transação de dados com webservices, o processo de levantamento de requisitos também foi outra característica que agregou conhecimento, não que não tenhamos esse conhecimento acadêmico, muito pelo contrário, fomos bem instruídos com relação a isso, porém o conhecimento adquirido pela rotina de trabalho e essa interação com usuário real, mostra que sempre temos algo novo a aprender, temos como exemplo sobre o levantamento de requisitos o momento de interação com um analista de logística, os fundamentos para gerar rotas, como ocorre o processo de coletas e entregas é tudo completamente diferente do que havíamos especulado.

Na área de humanas as entrevistas com a assistente social, aprendemos sobre as necessidades básicas do homem, a ser solidários e solícitos com quem realmente precisa. Coisas que não costumamos valorizar no nosso dia a dia, coisas simples como um copo de água, uma cama ou uma roupa limpa para vestir, simplesmente usamos e não nos damos conta de que faria muita diferença de acesso a essas coisas nos fossem tirados ao acaso. Em resumo, o desenvolvimento do projeto nos trouxe tanto conhecimentos técnicos quanto experiência de vida e uma motivação para juntar tecnologia a vontade de ajudar ao próximo.

4.2 *Trabalhos Futuros*

Tendo como base os diagramas de classes, fica claro que a aplicação possui uma espera para um porte maior do que está sendo apresentado, por exemplo, realizar associações diretas de endereço no momento em que a doação é associada a um beneficiário.

O sistema pode enviar um e-mail para o Doador no momento que a doação for entregue ao beneficiário.

Referências

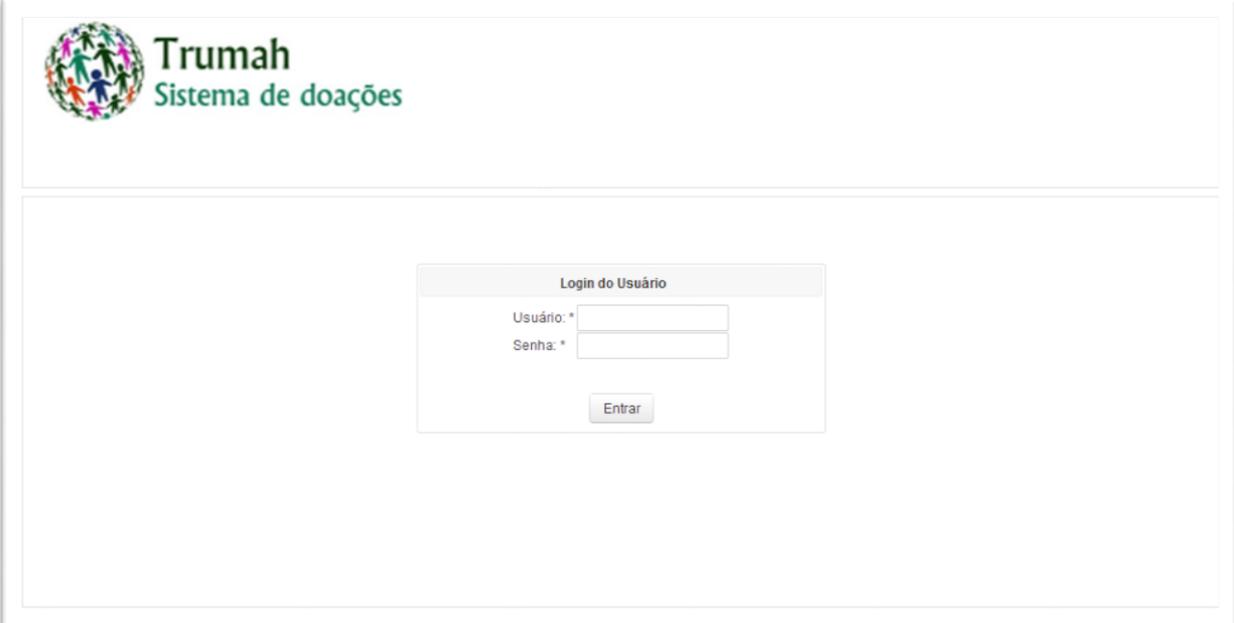
- [1] ZAGARI, Nicola. Padrões-05 - Padrões Arquiteturais - MVC. [S.l: s.n.], 2010. Disponível em: <<http://www.slideshare.net/zagari/padres05-padres-arquiteturais-mvc-3591612>>. Acesso em: 25 maio 2013.
- [2] BALLOU, Ronald H. Gerenciamento da cadeia de suprimentos/logística empresarial. Porto Alegre: Bookman, 2006.
- [3] CARVALHO, José Mexia Crespo De. Logística. Lisboa: Sílabo, 2002.
- [4] DATE, C. J; VIEIRA, André. Introdução a sistemas de bancos de dados. Rio de Janeiro: Campus, 2004.
- [5] GUPTA, Arun. Java EE 6 pocket guide. Sebastopol, CA: O'Reilly Media, 2012.
- [6] INSTITUTO GEOLÓGICO (SÃO PAULO, Brazil). Desastres naturais: conhecer para prevenir. 1a. ed ed. São Paulo: Instituto Geológico, 2009.
- [7] MELO, A.N.A.C. Desenvolvendo Aplicações com UML 2.2. [S.l.]: BRASPORT, [S.d.]. Disponível em: <<http://books.google.com.br/books?id=BPVHsG17bAYC>>.
- [8] PRESSMAN, R.S. Engenharia de software. [S.l.]: McGraw-Hill, 2006. Disponível em: <<http://books.google.com.br/books?id=MNM6AgAACAJ>>.
- [9] JENDROCK, Eric et al. The Java EE 5 Tutorial. [S.l: s.n.], 2010. Disponível em: <<http://docs.oracle.com/javaee/5/tutorial/doc/index.html>>. Acesso em: 17 abril. 2013.
- [10] HIBERNATE - Persistência Relacional para Java Idiomático. [S.l: s.n.], [S.d.]. Disponível em: <<http://docs.jboss.org/hibernate/core/3.5/reference/pt-BR/html/>>. Acesso em: 19 maio 2013.
- [11] Documentação Prime Faces. [S.l: s.n.], [S.d.]. Disponível em: <<http://www.primefaces.org/documentation.html>>. Acesso em: 17 maio 2013.
- [12] VARAKSIN, Oleg. Show Case Prime Faces. [S.l: s.n.], [S.d.]. Disponível em: <<http://www.primefaces.org/showcase/ui/home.jsf>>. Acesso em: 17 maio 2013.
- [13] A Google Directions API. [S.l: s.n.], 2013. Disponível em: <<https://developers.google.com/maps/documentation/directions/?hl=pt-BR>>. Acesso em: 17 abril. 2013.
- [14] A Google Geocoding API. [S.l: s.n.], 2013. Disponível em: <<https://developers.google.com/maps/documentation/geocoding/?hl=pt-BR>>. Acesso em: 17 abril. 2013.
- [15] Astah Community. [S.l: s.n.], [S.d.]. Disponível em: <<http://astah.net/company>>. Acesso em: 12 março. 2013.
- [16] Java Persistence API. [S.l: s.n.], [S.d.]. Disponível em: <<http://www.oracle.com/technetwork/java/javaee/tech/persistence-jsp-140049.html>>. Acesso em: 17 abril. 2013.
- [17] JavaServer Faces Technology - Documentation. [S.l: s.n.], [S.d.]. Disponível em: <<http://www.oracle.com/technetwork/java/javaee/documentation/index-137726.html>>. Acesso em: 17 abril. 2013.
- [18] MySQL Database. [S.l: s.n.], [S.d.]. Disponível em: <<http://www.mysql.com/products/enterprise/database/>>. Acesso em: 15 abril. 2013.
- [19] MySQL Workbench. [S.l: s.n.], [S.d.]. Disponível em: <<http://www.mysql.com/products/workbench/>>. Acesso em: 15 abril. 2013.
- [20] XPATH - Java. [S.l: s.n.], [S.d.]. Disponível em: <<http://www.w3schools.com/xpath/>>. Acesso em: 19 maio 2013.

Apêndice

Anexo A – Manual do Usuário

Tela de acesso

Para acessar o sistema basta digitar o usuário e a senha cadastrada e clicar em entrar para ter acesso ao menu com as funções permitidas para o perfil



A imagem mostra a tela de acesso do sistema. No topo esquerdo, há o logotipo do sistema, que consiste em um círculo com várias figuras humanas em cores diferentes, e ao lado dele o texto "Trumah Sistema de doações". Abaixo do logotipo, há um formulário de login centralizado. O formulário tem o título "Login do Usuário" e contém dois campos de entrada: "Usuário: *" e "Senha: *". Abaixo dos campos, há um botão "Entrar".

Caso digite um usuário e senha inválida o sistema exibirá uma mensagem "Usuário ou senha inválido"

Menu de acesso as telas

O menu abaixo lista todos as telas de acesso liberado de acordo com o perfil do usuário:



Basta selecionar a tela que deseja acessar para que o sistema exiba.

Tela de cadastro de beneficiário

A tela de cadastro de beneficiário possui os campos obrigatórios destacados por um *.

Beneficiário | Confirmação

Dados do Beneficiário

Nome: * Data de Nascimento: *

Sexo: * Masculino Feminino Estado Civil: -- Selecione --

Telefone: * Celular:

Email: * Logradouro: *

Número: * Complemento:

Estado: * -- Selecione -- Cidade: * -- Selecione --

Cep: * Bairro: *

* Campo(s) Obrigatório(s)

→ Avançar

Caso os campos não sejam preenchidos conforme o esperado, o sistema exibe mensagem ressaltando os campos esperados:

Trumah
Sistema de doações

Logout

Principal

Beneficiário

Cadastrar Beneficiário

Conceder Benefício

Visualizar Beneficiários

Visualizar Benefícios

Doadores

Cadastrar Doação

Cadastrar Doador

Cadastrar Grupo de Doação

Cadastrar Item para Doação

Visualizar Doações

Beneficiário | Confirmação

Dados do Beneficiário

Nome: Erro de validação: o valor é necessário.
Data Nascimento: Erro de validação: o valor é necessário.
Sexo: Erro de validação: o valor é necessário.
Estado Civil: Erro de validação: o valor é necessário.
Telefone: Erro de validação: o valor é necessário.
Email: Erro de validação: o valor é necessário.
Logradouro: Erro de validação: o valor é necessário.
Número: Erro de validação: o valor é necessário.
Estado: Erro de validação: o valor é necessário.
Cidade: Erro de validação: o valor é necessário.
Cep: Erro de validação: o valor é necessário.
Bairro: Erro de validação: o valor é necessário.

Nome: * Data de Nascimento: *

Sexo: * Masculino Feminino Estado Civil: -- Selecione --

Telefone: * Celular:

Email: * Logradouro: *

Número: * Complemento:

Estado: * -- Selecione -- Cidade: * -- Selecione --

Cep: * Bairro: *

Tela de Cadastro de Doador

A tela de cadastro de doador possui os campos obrigatórios destacados por um *.

Dados do Doador

Nome: *	Esther W. Williams	Data de Nascimento: *	15/09/1960
Sexo: *	<input checked="" type="radio"/> Masculino <input type="radio"/> Feminino	Estado Civil:	-- Selecione --
Telefone: *	(21)9886-2869	Celular:	(21)9886-2869
Email: *	EstherWWilliams@arm	Pessoa: *	<input checked="" type="radio"/> Física <input type="radio"/> Jurídica
CPF: *		Logradouro: *	Travessa Nova
Número: *	815	Complemento:	
Estado: *	Rio de Janeiro	Cidade: *	Rio de Janeiro
Bairro: *	Maré	CEP: *	21922250

* Campo(s) Obrigatório(s)

[← Voltar](#) [→ Avançar](#)

Caso os campos não sejam preenchidos conforme o esperado, o sistema exibe mensagem ressaltando os campos esperados.

Confirmação de cadastro de Doador

Após o preenchimento dos campos corretamente, a próxima tela é a tela de confirmação das informações do doador.

Confirmação

Nome:	Esther W. Williams	Data de Nascimento:	Thu Sep 15 00:00:00 BRT 1960
Sexo:	Feminino	Estado Civil:	
Telefone:	(21)9886-2869	Celular:	(21)9886-2869
Email:	EstherWWilliams@armyspy.com	Logradouro:	Travessa Nova
Número:	815	Complemento:	
Estado:	Rio de Janeiro	Cidade:	Rio de Janeiro
Bairro:	Maré	CEP:	21922250

[← Voltar](#) [Salvar](#) [Realizar Doação](#)

Caso o doador apenas queira constar no banco de doadores para uma futura doação basta clicar em salvar, caso o doador deseje realizar uma doação basta clicar no botão “Realizar Doação”.

Tela de Realizar Doação

Após o doador ter optado por realizar uma doação o atendente preencherá a quantidade de itens que estão pré cadastrados no sistema .

Doador: Esther W. Williams

Tipo de material: * Material de higiene e limpeza

Item: * Pano de chão

Quantidade: 4

Data de Doação:

Incluir Item

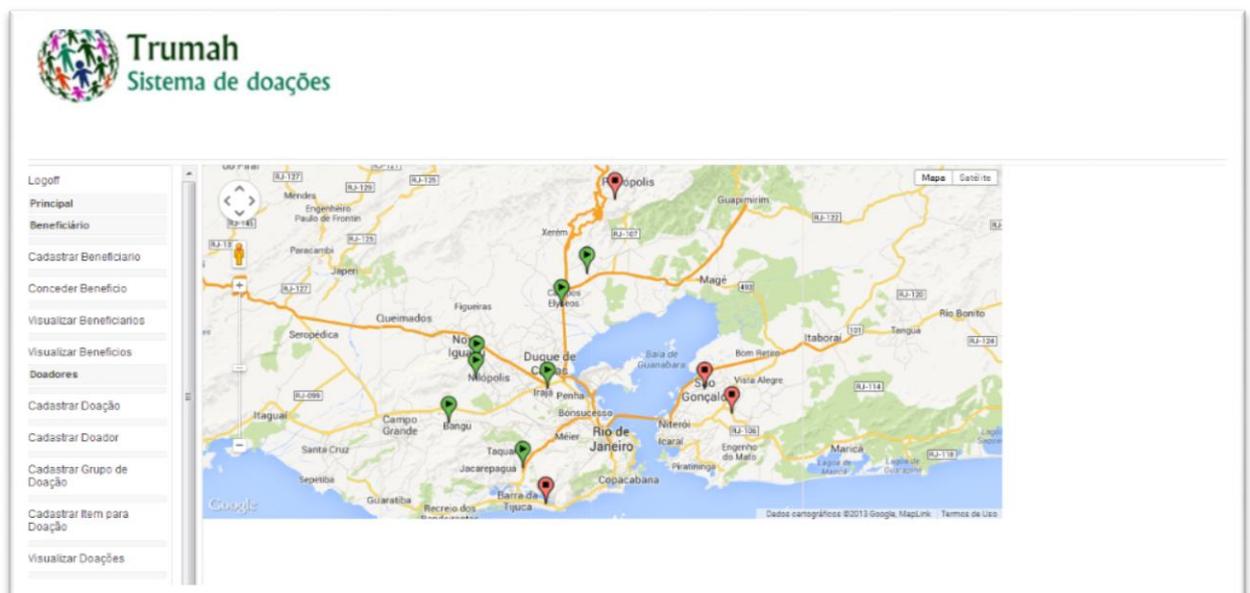
Nome	Qtd
Vassoura	3

Concluir Doação

Após a inclusão dos itens e a conclusão da doação o doador fica disponível para ser associado a um beneficiário.

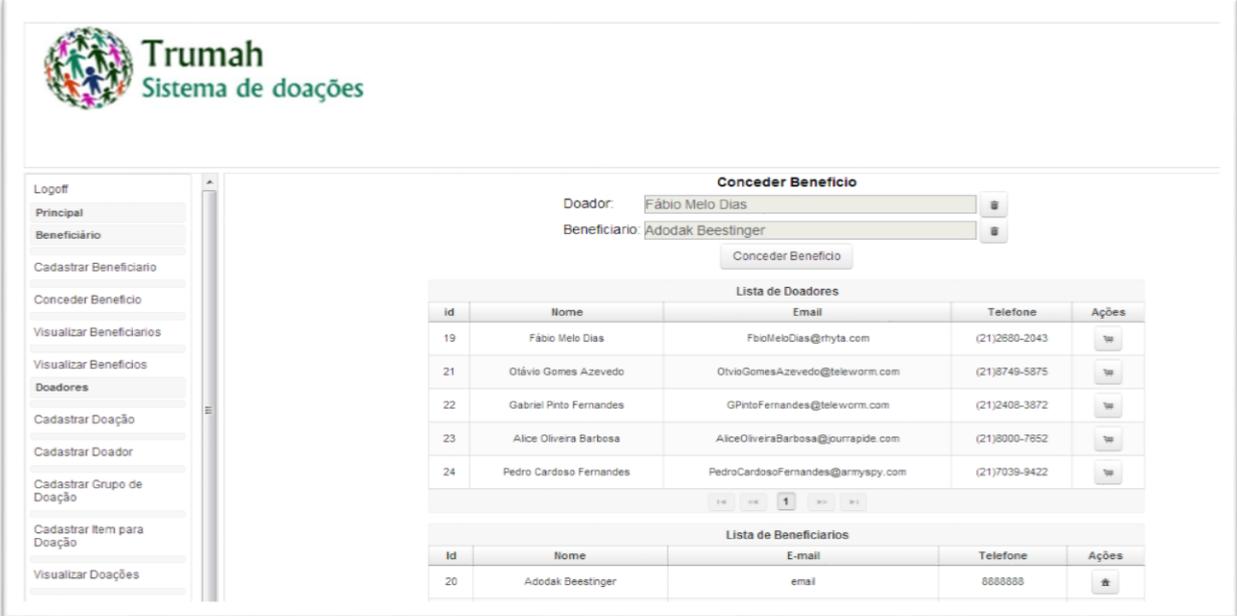
Tela de Visualização de coordenadas

Essa ferramenta permite uma visualização das coordenadas dos doadores e beneficiários tornando visível os pontos mais carentes e os pontos mais diviçosos.



Tela de Conceder Benefício

A partir do momento que o doador disponibiliza uma doação, fica disponível para ser associado a um beneficiário:



Trumah
Sistema de doações

Conceder Benefício

Doador: Fábio Melo Dias

Beneficiário: Adodak Beestinger

Conceder Benefício

Lista de Doadores

Id	Nome	Email	Telefone	Ações
19	Fábio Melo Dias	FbioMeloDias@rmyta.com	(21)2880-2043	
21	Otávio Gomes Azevedo	OtvioGomesAzevedo@teleworm.com	(21)8749-5875	
22	Gabriel Pinto Fernandes	GPintoFernandes@teleworm.com	(21)2408-3872	
23	Alice Oliveira Barbosa	AliceOliveiraBarbosa@purrapide.com	(21)8000-7852	
24	Pedro Cardoso Fernandes	PedroCardosoFernandes@armyspy.com	(21)7039-9422	

Lista de Beneficiários

Id	Nome	E-mail	Telefone	Ações
20	Adodak Beestinger	email	8888888	

Para que o doador seja incluso na concessão do benefício, o assistente social deverá clicar no botão correspondente a linha do beneficiário da grade exibida:



Para que o beneficiário seja incluso na concessão do benefício, o assistente social deverá clicar no botão correspondente a linha do beneficiário da grade exibida:

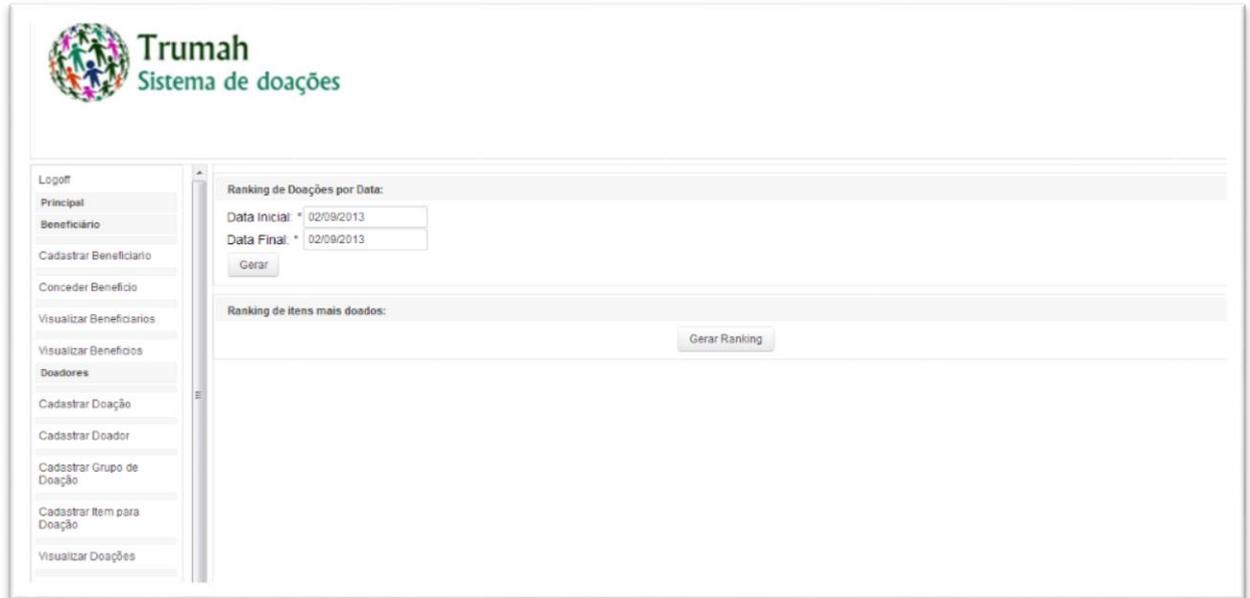


Caso o assistente social deseje realizar a troca do doador ou beneficiário, basta clicar no botão abaixo relacionado a concessão do benefício:

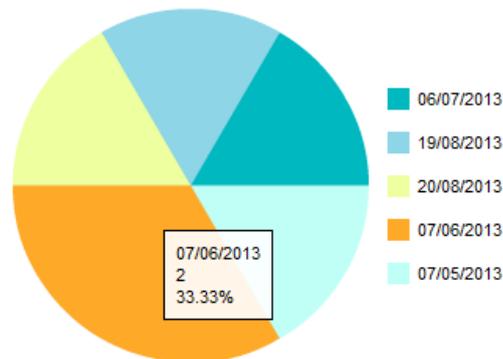


Tela Gerar Gráficos

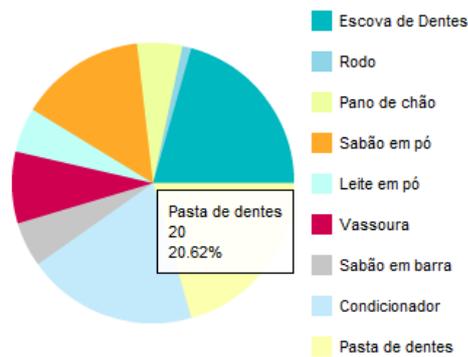
Basta filtrar as datas e clicar no botão “Gerar”:



Nesse momento aparecerá um gráfico semelhante a ilustração abaixo, contendo as quantidades de doações realizadas por data



Caso tenha escolhido a opção “Gerar Ranking” será gerado uma ilustração como a demonstrada abaixo com o total de itens doados:



Tela de visualização de rota

Após a rota criada o sistema permite visualizar as rotas que foram criadas entre doadores e beneficiários, basta clicar no botão “Visualizar Mapa”



Logoff
Principal
Beneficiário
Cadastrar Beneficiário
Conceder Benefício
Visualizar Beneficiários
Visualizar Benefícios
Doadores
Cadastrar Doação
Cadastrar Doador
Cadastrar Grupo de Doação
Cadastrar Item para Doação
Visualizar Doações
Visualizar Doadores

Rota: 21

Lograd.	Nº	Bairro	Cidade	Estado
Rua Seis	1860	Ramos	Rio de Janeiro	Rio de Janeiro
Jose Gomes Amado	75	Arsenal	São Gonçalo	Rio de Janeiro

Visualizar mapa

Rota: 22

Lograd.	Nº	Bairro	Cidade	Estado
Rua Seis	1860	Ramos	Rio de Janeiro	Rio de Janeiro
Avenida das Américas	148	barra da tijuca	Rio de Janeiro	Rio de Janeiro

Visualizar mapa

(1 of 1) 1 10

O sistema mostrará a rota entre os endereços que constam na grade escolhida:



The screenshot displays the Trumah Sistema de doações web application. On the left, a sidebar menu contains the following items: Logoff, Principal, Beneficiário, Cadastrar Beneficiário, Conceder Benefício, Visualizar Beneficiários, Visualizar Benefícios, Doadores, Cadastrar Doação, Cadastrar Doador, Cadastrar Grupo de Doação, Cadastrar Item para Doação, Visualizar Doações, and Visualizar Doadores. The main area shows a Google Map of Rio de Janeiro with a green route connecting several points. A 'Voltar' button is located below the map. The map includes standard navigation controls and a copyright notice for Google Maps.