

PROJETO INTEGRADO EM INTERFACE COM O USUÁRIO: UM SOFTWARE DE
DOAÇÃO QUE REALIZA INTERMEDIÇÃO VOLTADA PARA AGRUPAMENTO DE
USUÁRIOS DE DOAÇÕES.

EDWARD JUN HO CHOI	RA: 111646
EMERSON DE SOUZA NOGUEIRA	RA: 113574
IGOR FELIPY LOPES DE FRANÇA	RA: 114137
LUCCAS NEVES PALHARES	RA: 031717
TIAGO MARINHO SOUSA	RA: 113946

São Paulo - SP

2019

EDWARD JUN HO CHOI

EMERSON DE SOUZA NOGUEIRA

IGOR FELIPY LOPES DE FRANÇA

LUCCAS NEVES PALHARES

TIAGO MARINHO SOUZA

PROJETO INTEGRADO EM INTERFACE COM O USUÁRIO: UM SOFTWARE DE
DOAÇÃO QUE REALIZA INTERMEDIÇÃO VOLTADA PARA AGRUPAMENTO DE
USUÁRIOS DE DOAÇÕES.

Projeto apresentado à banca do
Centro Universitário das Américas como
exigência parcial para a aprovação se-
mestral sob orientação do prof. Dr. Wag-
ner Varalda.

São Paulo - SP

2019

EDWARD JUN HO CHOI
EMERSON DE SOUZA NOGUEIRA
IGOR FELIPY LOPES DE FRANÇA
LUCCAS NEVES PALHARES
TIAGO MARINHO SOUZA

PROJETO INTEGRADO EM INTERFACE COM O USUÁRIO: UM SOFTWARE DE
DOAÇÃO QUE REALIZA INTERMEDIÇÃO VOLTADA PARA AGRUPAMENTO DE
USUÁRIOS DE DOAÇÕES.

Projeto apresentado à banca do
Centro Universitário das Américas como
exigência parcial para a aprovação se-
mestral sob orientação do prof. Dr. Wag-
ner Varalda.

Data de aprovação:	/ /
--------------------	-----

	Assinaturas
Edward Jun Ho Choi	
Emerson de Souza Nogueira	
Igor Felipy Lopes de França	
Luccas Neves Palhares	
Tiago Marinho Souza	

RESUMO

O objetivo do Projeto Integrado (PI) em Interfaces com o Usuário é integrar as disciplinas: Engenharia de Software, Orientação a Objetos e Análise e Projeto de Sistemas do curso de Bacharelado da Ciência da Computação do 3º semestre, como resultado destas integrações, desenvolveu-se o P.I., onde a programação Orientação a Objetos é a parte de construção de software usando o Java para assimilar a interface que interage com o usuário e Oracle para a construção do Banco de Dados do aplicativo, em Análise será feita toda a parte de Projeto e Construção do Sistema. Este software visa unir todos os meios de doações e doadores, agrupados e sincronizados em um único banco de dados, facilitando o cadastro, criando um acesso prático para usuários e doadores verem tabelas semanais, mensais e anuais de cada processo, acesso de histórico de transação, facilidade de cadastro para futuros clientes, e assim, alcançando cada vez mais pessoas que possam contribuir para aqueles que necessitam de doações para uma vida melhor.

Palavras-chave: Sistema de doações, doações coletivas, dispositivos móveis, aplicativo para doações.

ABSTRACT

The objective of the Integrated Project (PI) in User Interfaces is to integrate the disciplines: Software Engineering, Object Orientation and Analysis and Systems Design of the Bachelor's Degree in Computer Science of the 3rd semester, as a result of these integrations, the subject P.I. was developed, where Object Oriented programming is the part of Software Engineering using Java to assimilate the interface that interacts with the user and Oracle to build the Application Database, in Analysis will be done every part of the Project and Construction of the System. This software aims to unite all means of donations and donors, grouped and synchronized in a single database, facilitating the registration, creating a practical access for users and donors to see weekly, monthly and annual tables of each process, access of transaction history, ease of registration for future clients, and thus reaching more and more people who can contribute to those who need donations for a better life.

Keywords: Donation system, collective donation, mobile phones, donation application.

SUMÁRIO

1 Introdução.....	7
1.1 Objetivo Geral.....	7
1.2 Objetivos Específicos.....	7
1.3 Justificativa.....	7
2 Fundamentação Teórica.....	8
2.1 Engenharia de Software.....	8
2.2 Interface com o Usuário.....	28
2.3 Análise e Desenvolvimento de Sistemas.....	29
2.4 Programação Orientada a Objetos.....	34
3 Especificação do Projeto.....	35
4 Modelagem de Sistemas.....	36
5 Implementação da Aplicação.....	40
6 Conclusão.....	46
Referências.....	47

1 Introdução.

Este projeto teve como objetivo realizar estudos sobre, conteúdos interdisciplinares dos cursos de Bacharelado de Ciências da Computação, Desenvolvimento de Sistemas, Tecnologia de Análise, nos quais estão combinadas com as matérias de Engenharia de Software, Programação Orientada a Objetos, Análise e Projeto de Sistema, Projeto Integrado em Interfaces de Usuário ao terceiro semestre. Com base nos estudos realizados, pode-se observar que há uma necessidade de informação sobre métodos de doação e arrecadação do mesmo. A partir da necessidade identificada, nosso grupo formou um projeto para criação de um software específico no ramo de doações permitindo a aplicação dos conceitos teóricos desenvolvidos através das aulas práticas. O foco é expressar o aprendizado dos acadêmicos envolvidos, com base em informações fornecidas pelos docentes.

1.1 Objetivo Geral.

O objetivo geral deste projeto é singularizar o método de doações já existentes, desmitificando os meios para quem quer doar e para quem precisa da doação. O software suporta cadastro de usuários em níveis diferentes, histórico de transação, cadastro e visualização de perfil/usuário, pessoas e campanhas. O que torna este software a ter um impacto diretamente naqueles que o utilizarem.

1.2 Objetivos Específicos.

A necessidade da singularidade de um software unir todos os meios de doações e doadores, onde todos os que fossem usuários beneficiários como: instituições de caridade, refugiados, ONG's, afetados por desastres naturais, voluntários, usuários como influenciadores digitais, e usuários que optam arrecadar dinheiro na função “vaquinha” agrupados e sincronizados em um único banco de dados.

1.3 Justificativa.

Este projeto foi criado com o destino de ser um software que facilita um indivíduo a encontrar uma lista de instituições de caridade, ONG's e afins, independente de ser usuário doador ou beneficiário, gerando um movimento maior entre usuários dispostos a interagir com informação direta de seu criador, sendo causas e notícias locais (pode expandir para outros), pode expandir o mapa com pesquisa e lista de campanhas ativas e desativadas, além da interação de visualização de perfil (pode expandir para ganhar conquistas), o que torna bem singular a experiência sobre a visão do usuário em relação a doações. Visando que falta este tipo de vinculação no mercado, até a data de entrega deste projeto.

2 Fundamentação Teórica.

A fundamentação teórica deste Projeto Integrado (P.I.) é baseada nas aulas e explicações dos docentes das matérias de Engenharia de Software, Análise e Desenvolvimento de Sistemas e Programação Orientada a Objetos para a realização e concretização do software de doações Help Us, cujo objetivo é unificar os meios já existentes de doações e das instituições e vertentes em um único software.

2.1 Engenharia de Software.

Nesta matéria foram instruídos: Elementos Essenciais de Engenharia de Software; Definição de Processo de Desenvolvimento (visão geral e didática); Desenvolvimento de Software; Ciclo de Vida; Prototipação; Modelo Espiral; Modelo V; RUP (Rational Unified Process) Framework; Conceitos Chave, Elementos Essenciais do Processo e Princípios e Filosofias. Artefatos do RUP: Avaliação de Iteração e Status, Caso de Desenvolvimento, Caso de Negócios, Documento de Arquitetura de Software, Especificação Complementar e de Caso de Uso, Lista de Riscos, Plano de Desenvolvimento de Software, Plano de Implementação, Plano de Iteração, Plano de Teste e Visão. IBM Rational Method Composer (IBM RMC): Tríade; Desenvolvimento Orientado a Negócios, SOA. RMC Visão Geral da Terminologia e Conceitos-Chave. Desenvolvimento Ágil de Software. Desenvolvimento Ágil, Tríade do Desenvolvimento Ágil, Atributos de Qualidade, Preceitos-Chave e Dificuldades Práticas. Scrum: Ciclo de Vida, Planejamento Geral e Projeto de Arquitetura, Ciclo Sprint, Encerramento de Projeto, Gerenciamento, Vantagens, Desvantagem. Arquitetura de Software: Metáforas (Esquema, Linguagem, Decisão e Literatura), Considerações Decisivas, Arquitetura Centrada em Dados, Fluxo de Dados, Programa Principal/Subprograma, Camadas.

Declaração do Problema.

O problema de	Falta de interatividade entre doadores e beneficiários
Afeta	Instituições de caridade, criadores de conteúdo, ONG's, usuários em geral.
O impacto é o seguinte	Dificulta a vida das pessoas que necessitam de doações
Uma solução bem sucedida seria	Ter um meio dessas pessoas conseguirem suas doações de forma prática.

Especificação do Projeto

Recursos do suporte	Benefícios ao usuário
O suporte do aplicativo Help Us identifica os usuários necessitados de doações, fazendo uma ponte entre os usuários para que possam realizar as doações.	Juntar dinheiro das pessoas em prol de um determinada causa pessoal.
Os problemas são relacionados por nível de importância, desde catástrofes a ajuda á instituições de caridade, desastres naturais. Notificando-o usuário de qual causa necessita de ajuda o mais rápido possível.	Campanha de arrecadação junta os doadores de diversos lugares afim de ajudar ás instituições de caridades , ONG's etc...
Tendências do mercado de entretenimento listado por "ranking" de mais populares entres os diversos criadores de conteúdo.	Essa funcionalidade permite com que os seguidores possam doar de forma fácil e simples para seus criadores preferidos.
Bancos de dados das contas bancárias de todas as instituições, pessoas, influenciadores, ONG's, que usam o recurso do aplicativo.	Facilita a transferência de valores paras as contas de seus respectivos bancos, dando uma comodidade maior ao usuário.

Declaração da Posição do Produto.

Para	Criadores de conteúdo.
Que	Recebe doações de seus seguidores.
O (Help Us)	É uma solução de arrecadação monetária.
Oferece	Integração financeira entre criadores de conteúdo e seus seguidores.

Para	ONG's
Que	Precisam de uma forma de ajuda para manter as suas ações.
O (Help Us)	É uma solução para a apresentação de projetos para o público.
Oferece	Uma forma de arrecadação de dinheiro.

Para	Público em geral.
Que	Necessitam de formas de vaquinha.
O (Help Us)	É um software de vaquinha.
Oferece	Uma forma de vaquinha entre grupos específicos.

Para	Instituições de caridade.
Que	Necessitam de doações do público para que suas ações sociais sejam concretizadas.
O (Help Us)	É um software de doações.
Oferece	Uma forma de arrecadação de dinheiro.

Recursos do Produto

5.1 Criar Perfil (Doador ou Beneficiário).

Função: Cria perfil para usuário doador ou beneficente.

5.2 Criar Campanha.

Função: Cria Campanha para efetivar metas.

5.3 Receber Doação.

Função: Recebe doação de usuário doador.

5.4 Pesquisar Campanha e Usuário.

Função: Pesquisa campanha ativa e desativada e usuários.

5.5 Visualizar Usuário e Campanha.

Função: Visualiza Usuário e Campanha pesquisada.

5.6 Visualizar Histórico de Doação.

Função: Exibe um histórico de doação feito à partir da data de cadastro.

5.7 FAQ.

Função: Frequent Asked Questions (Perguntas Frequentes).

5.8 SAC.

Função: Sistema de Atendimento ao Cliente.

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.1 Criar Perfil (Doador ou Beneficiário).	Data: 27/05/2019
<Identificador de documento>	

Índice

- 1. Breve Descrição.**
- 2. Fluxo Básico de Eventos.**
- 3. Fluxos Alternativos.**
 - 3.1 Nome de cadastro não preenchido.**
 - 3.2 CNPJ não permitido.**

Especificação de Caso de Uso: Criar Perfil (Doador ou Beneficiário).

1. Breve Descrição.

Vem do 5.1. Essa função cadastra um novo usuário cuja função é doar ou arrecadar dinheiro de um usuário doador pré-cadastrado.

2. Fluxo Básico de Eventos.

1. Usuário informa Nome.
2. Usuário informa Idade.
3. Usuário informa Estado.
4. Usuário informa Cidade.
5. Usuário informa Telefone.
6. Usuário informa E-mail.
7. Usuário informa (Registro Geral) RG.
8. Usuário informa (Cadastro de Pessoa Física) CPF.
9. Usuário informa (Cadastro Nacional de Pessoa Jurídica) CNPJ.
10. Sistema cria usuário.

3. Fluxos Alternativos.

3.1 Nome de cadastro não preenchido.

No passo 1 do Fluxo Básico, o sistema verifica que o Nome de usuário não foi preenchido. Sistema informa: “Obrigatório informar Nome de usuário”. Retorna ao passo 1 do Fluxo Básico.

3.2 CNPJ não permitido.

No passo 9 do Fluxo Básico o sistema verifica que o CNPJ não está no padrão. Retorna ao passo 9 do Fluxo Básico.

HelpUs

Especificação de Caso de Uso: Criar Perfil (Doador ou Beneficiário)

Versão <1.0>

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.2 Criar Campanha	Data: 27/05/2019
<Identificador de documento>	

Índice

- 1. Breve Descrição.**
- 2. Fluxo Básico de Eventos.**
- 3. Fluxos Alternativos.**
 - 3.1 Nome de Campanha não preenchido.**
 - 3.2 Tempo não permitido.**

Especificação de Caso de Uso: Criar Campanha

1. Breve Descrição.

Vem do 5.2. Essa função cria os recurso de recebimento e coleta de dinheiro, conforme a necessidade do usuário.

2. Fluxo Básico de Eventos.

1. Usuário informa Tipo de Campanha.
2. Usuário informa Nome da Campanha.
3. Usuário informa Meta da Campanha.
4. Usuário informa Tempo Disponível para doação.
5. Usuário confirma criação da campanha.
6. Sistema cria campanha.

3. Fluxos Alternativos

3.1 Nome de Campanha não preenchido.

No passo 2 do Fluxo Básico, o sistema verifica que o Nome da Campanha não foi preenchido.

Sistema informa: “Obrigatório informar Nome da Campanha”.

Retorna ao passo 2 do Fluxo Básico.

3.2 Tempo não permitido.

No passo 4 do Fluxo Básico o sistema verifica que o Tempo Disponível não está no padrão.

Retorna ao passo 4 do Fluxo Básico.

HelpUs

Especificação de Caso de Uso: Criar Campanha

Versão <1.0>

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.3 Receber Doação	Data: 27/05/2019
<Identificador de documento>	

Índice

1. Breve Descrição.

2. Fluxo Básico de Eventos.

3. Fluxos Alternativos.

3.1 Usuário não confirmou número token.

3.2 Token não permitido.

3.3 Excedendo limite.

Especificação de Caso de Uso: Receber Doação.

1. Breve Descrição.

Vem do 5.3. Essa função transfere o dinheiro arrecadado para os meios de recebimento suportados pelo sistema.

2. Fluxo Básico de Eventos.

1. Usuário confirma dados para transferência.
2. Uma mensagem de token é enviada para o telefone móvel.
3. Uma mensagem de token é enviada para o e-mail.
4. Usuário digita token recebido para confirmar a identidade do usuário.

3. Fluxos Alternativos.

3.1 Usuário não confirmou número token.

No passo 4 do Fluxo Básico, o sistema verifica que o token não foi preenchido.

Sistema informa: “Obrigatório informar token”.

Retorna ao passo 4 do Fluxo Básico.

3.2 Token não permitido.

No passo 4 do Fluxo Básico o sistema verifica que o token não está no padrão.

Retorna ao passo 4 do Fluxo Básico.

3.3 Excedendo limite.

No passo 4 do Fluxo Básico o sistema verifica que o token foi digitado mais de 5 vezes erroneamente.

Sistema informa: “Encerrar transferências por 12 horas”.

Retorna ao passo 4 do Fluxo Básico com limite de 12 horas para preencher o token novamente.

HelpUs

Especificação de Caso de Uso: Receber Doação

Versão <1.0>

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.4 Pesquisar Campanha e Usuário.	Data: 27/05/2019
<Identificador de documento>	

Índice

1. Breve Descrição.

2. Fluxo Básico de Eventos.

3. Fluxos Alternativos.

3.1 Nome não preenchido corretamente.

3.2 Nome da campanha não encontrado.

Especificação de Caso de Uso: Pesquisar Campanha e Usuário.

1. Breve Descrição.

Vem do 5.4. Essa função permite o usuário pesquisar campanhas e/ou usuários cadastrados, conforme sua necessidade.

2. Fluxo Básico de Eventos.

1. Usuário pesquisa Campanha ou Usuário.
2. Usuário pesquisa por Nome.
3. Usuário pesquisa por CPF.
4. Usuário pesquisa por CNPJ.
5. Usuário pesquisa por Tipo de Campanha.

3. Fluxos Alternativos.

3.1 Nome não preenchido corretamente.

No passo 2 do Fluxo Básico, o sistema verifica que o Nome para pesquisa não foi preenchido corretamente.

Sistema informa: “Não há nada com esta busca”.

Retorna ao passo 2 do Fluxo Básico.

3.2 Nome da campanha não encontrado.

No passo 5 do Fluxo Básico o sistema verifica que o Nome de Campanha não está no padrão.

Sistema informa: “Não há nada com esta busca”.

Retorna ao passo 5 do Fluxo Básico.

HelpUs

Especificação de Caso de Uso: Receber Doação

Versão <1.0>

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.5 Visualizar Usuário e Campanha	Data: 27/05/2019
<Identificador de documento>	

Índice

- 1. Breve Descrição.**
- 2. Fluxo Básico de Eventos.**
- 3. Fluxos Alternativos.**
 - 3.1 Tipo de Campanha não preenchido.**
 - 3.2 CNPJ não permitido.**

Especificação de Caso de Uso: Visualizar Usuário e Campanha

1. Breve Descrição.

Vem do 5.5. Essa função permite o usuário visualizar campanhas e/ou usuários cadastrados, conforme sua necessidade.

2. Fluxo Básico de Eventos.

1. Usuário informa Tipo de Campanha.
2. Usuário informa Nome da Campanha.
3. Usuário informa Meta da Campanha.
4. Usuário informa CPF.
5. Usuário informa CNPJ.
6. Usuário confirma opção de visualização.
7. Sistema apresenta tipo escolhido.

3. Fluxos Alternativos.

3.1 Tipo de Campanha não preenchido.

No passo 1 do Fluxo Básico, o sistema verifica que o Tipo da Campanha não foi preenchido.

Sistema informa: “Obrigatório informar Tipo da Campanha”.

Retorna ao passo 2 do Fluxo Básico.

3.2 CNPJ não permitido.

No passo 4 do Fluxo Básico o sistema verifica que o CNPJ não está no padrão.

Sistema informa: “Não é possível visualizar este tipo de CNPJ”.

Retorna ao passo 4 do Fluxo Básico.

HelpUs

Especificação de Caso de Uso: Visualizar Campanha e Usuário.

Versão <1.0>

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.6 Visualizar Histórico de Doação.	Data: 27/05/2019
<Identificador de documento>	

Índice

1. Breve Descrição.

2. Fluxo Básico de Eventos.

3. Fluxos Alternativos.

3.1 Usuário não salva dados de doação.

3.2 Usuário não efetua doação.

Especificação de Caso de Uso: Visualizar Histórico de Doação.

1. Breve Descrição.

Vem do 5.6. Essa função exibe o histórico de doação desde a data do cadastro de usuário.

2. Fluxo Básico de Eventos.

1. Usuário efetua doação.
2. Sistema registra Data.
3. Sistema registra Hora.
4. Sistema registra Valor.
5. Sistema registra Favorecido.
6. Usuário confirma salvar informações para consulta.
7. Sistema apresenta Histórico de Doação.

3. Fluxos Alternativos.

3.1 Usuário não salva dados de doação.

No passo 6 do Fluxo Básico, o sistema verifica que o usuário não salvou os dados da doação.

Sistema informa: “Nenhum favorecido cadastrado”.

Retorna ao passo 6 do Fluxo Básico.

3.2 Usuário não efetua doação.

No passo 1 do Fluxo Básico o sistema verifica que o usuário não registrou os dados de doação.

Sistema informa: “Nenhum valor cadastrado”.

Retorna ao passo 1 do Fluxo Básico.

HelpUs

Especificação de Caso de Uso: Visualizar Histórico de Doação.

Versão <1.0>

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.7 FAQ (Frequent Asked Questions: Perguntas Frequentes).	Data: 27/05/2019
<Identificador de documento>	

Índice

1. Breve Descrição.

2. Fluxo Básico de Eventos.

3. Fluxos Alternativos.

3.1 Usuário não encontra dúvida no FAQ.

3.2 Usuário pesquisa dúvida não listada.

Especificação de Caso de Uso: FAQ (Frequent Asked Questions: Perguntas Frequentes).

1. Breve Descrição.

Vem do 5.7. Essa função apresenta as perguntas que a maioria dos usuários têm em relação ao aplicativo.

2. Fluxo Básico de Eventos.

1. Usuário pesquisa dúvida sobre a funcionalidade o software.
2. Sistema apresenta FAQ.
3. Usuário verifica se no FAQ contém a pergunta correspondente.
4. Usuário envia e-mail aos desenvolvedores.

3. Fluxos Alternativos.

3.1 Usuário não encontra dúvida no FAQ.

No passo 1 do Fluxo Básico, o usuário pesquisa a dúvida sobre a funcionalidade do software.

Sistema informa: “Nenhuma informação relacionada foi encontrada”.

Retorna ao passo 1 do Fluxo Básico.

3.2 Usuário pesquisa dúvida não listada.

No passo 1 do Fluxo Básico o usuário verifica que nenhuma pergunta do FAQ corresponde com a do usuário.

Sistema informa: “Envie uma mensagem aos desenvolvedores”.

Retorna ao passo 4 do Fluxo Básico.

HelpUs

Especificação de Caso de Uso: FAQ (Frequent Asked Questions: Perguntas Frequentes).

Versão <1.0>

Help Us	Versão <1.0>
Especificação de Caso de Uso: 5.8 SAC (Sistema de Atendimento ao Cliente).	Data: 27/05/2019
<Identificador de documento>	

Índice

1. Breve Descrição.

2. Fluxo Básico de Eventos.

3. Fluxos Alternativos.

3.1 Usuário envia sugestão ao SAC.

3.2 Usuário escreve no Fluxo Básico não correspondente.

Especificação de Caso de Uso: SAC (Sistema de Atendimento ao Cliente).

1. Breve Descrição.

Vem do 5.8. Essa função cria um meio de comunicação entre o usuário e a empresa.

2. Fluxo Básico de Eventos.

1. Usuário escreve uma sugestão.
2. Usuário escreve um elogio.
3. Usuário escreve uma dúvida em relação ao software.
4. Usuário escreve reclamação.
5. Usuário envia mensagem correspondente ao Fluxo Básico para o SAC.
6. Equipe do SAQ responde em curto prazo.

3. Fluxos Alternativos.

3.1 Usuário envia sugestão ao SAC.

No passo 1 do Fluxo Básico, o usuário envia uma sugestão.

Sistema informa: “Informe a sua sugestão”.

Retorna ao passo 5 do Fluxo Básico.

3.2 Usuário escreve no Fluxo Básico não correspondente.

No passo 2 do Fluxo Básico o usuário escreve uma dúvida em relação ao software.

Sistema informa: “Opção inválida”.

Retorna ao passo 2 do Fluxo Básico.

HelpUs

Especificação de Caso de Uso: SAC (Sistema de Atendimento ao Cliente).

Versão <1.0>

Aqui permanece a importância de construir um layout, onde o usuário possui a facilidade de compreensão e fluidez da interface gráfica, utilizada para interagir com o software. Na Visão Geral estabelece um meio de comunicação efetivo entre o ser humano e o computador. Seguindo o princípio de projeto de interfaces, identifica-se objetos e ações de interface, cria um layout de tela que forma a base para o protótipo de interface com o usuário com a identificação dos usuários, das tarefas e dos requisitos do ambiente. São usadas ferramentas para criar a implementação do modelo de projeto, o resultado é avaliado em termos de qualidade. A Usabilidade é indicada pelos seguintes atributos: facilidade de compreensão, facilidade de aprendizagem e operabilidade. Na Estética estabelece elementos agradáveis e de fácil de assimilação pelo usuário. No Protótipo, valida-se a arquitetura e resolução de riscos que garantem ao cliente um software que será sempre aprimorado, buscando a melhor experiência. Os usuários testam a interface e o feedback desses testes é usado para a próxima modificação iterativa do protótipo. Nos Princípios para Projetos de Interface o usuário tem controle total para definir modos de interação, evitando realizar ações desnecessárias ou indesejadas, proporcionando flexibilidade onde o usuário pode personalizar a interação e projetar com objetos que aparecem direto na tela. Em Análise e Projeto de Interface Gráfica abordamos os temas: Análise e Projeto de Interfaces, Modelos de Análise e Projeto de Interfaces, Processo de Análise e Projeto para Interfaces do Usuário, Análise e Modelagem de Interface, Construção de Interface e Validação de Interface. Em Análise de Usuários um dos princípios fundamentais ao se criar uma interface é conhecer o perfil do usuário que irá usar a ferramenta, entender o seu problema e desenvolver uma solução.

Nesta matéria foram instruídos pelo docente os seguintes Diagramas para compor este projeto: Diagramas de Classe, Diagrama de Objeto, Diagrama de Caso de Uso, Diagrama de Pacotes, Diagrama de Sequência, Diagrama de Atividade e Diagrama de Tempo, com a exceção de alguns diagramas não mostrados e/ou detalhados neste documento, por não terem sido utilizados no processo do mesmo.

Diagrama de Caso de Uso

1. Usuário pode iniciar campanha.
2. Usuário pode fazer doação
3. Usuário pode iniciar vaquinha.
4. Usuário pode iniciar campanha de apoio.
5. Usuário pode ver perfis.
6. Usuário pode pesquisar perfis.
7. Usuário pode pesquisar campanhas.
8. Usuário pode criar perfil.
9. Usuário pode finalizar campanha.

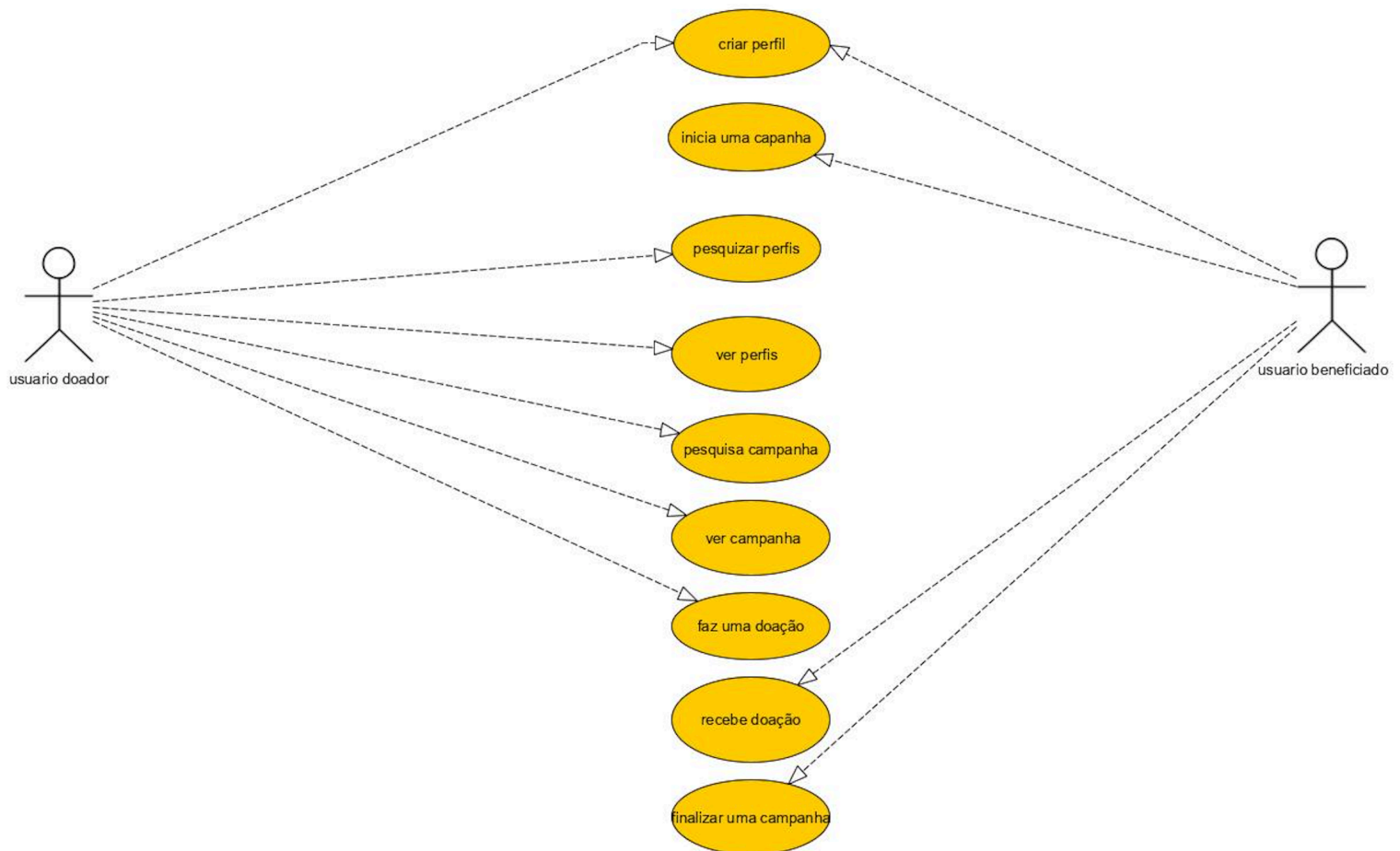


Diagrama de Atividades

1. Usuário pode iniciar campanha.
2. Usuário pode fazer doação.
3. Usuário pode iniciar campanha de apoio.
4. Usuário pode ver perfis.
5. Usuário pode pesquisar perfis.
6. Usuário pode pesquisar campanhas.
7. Usuário pode criar perfil.
8. Usuário pode finalizar campanha.

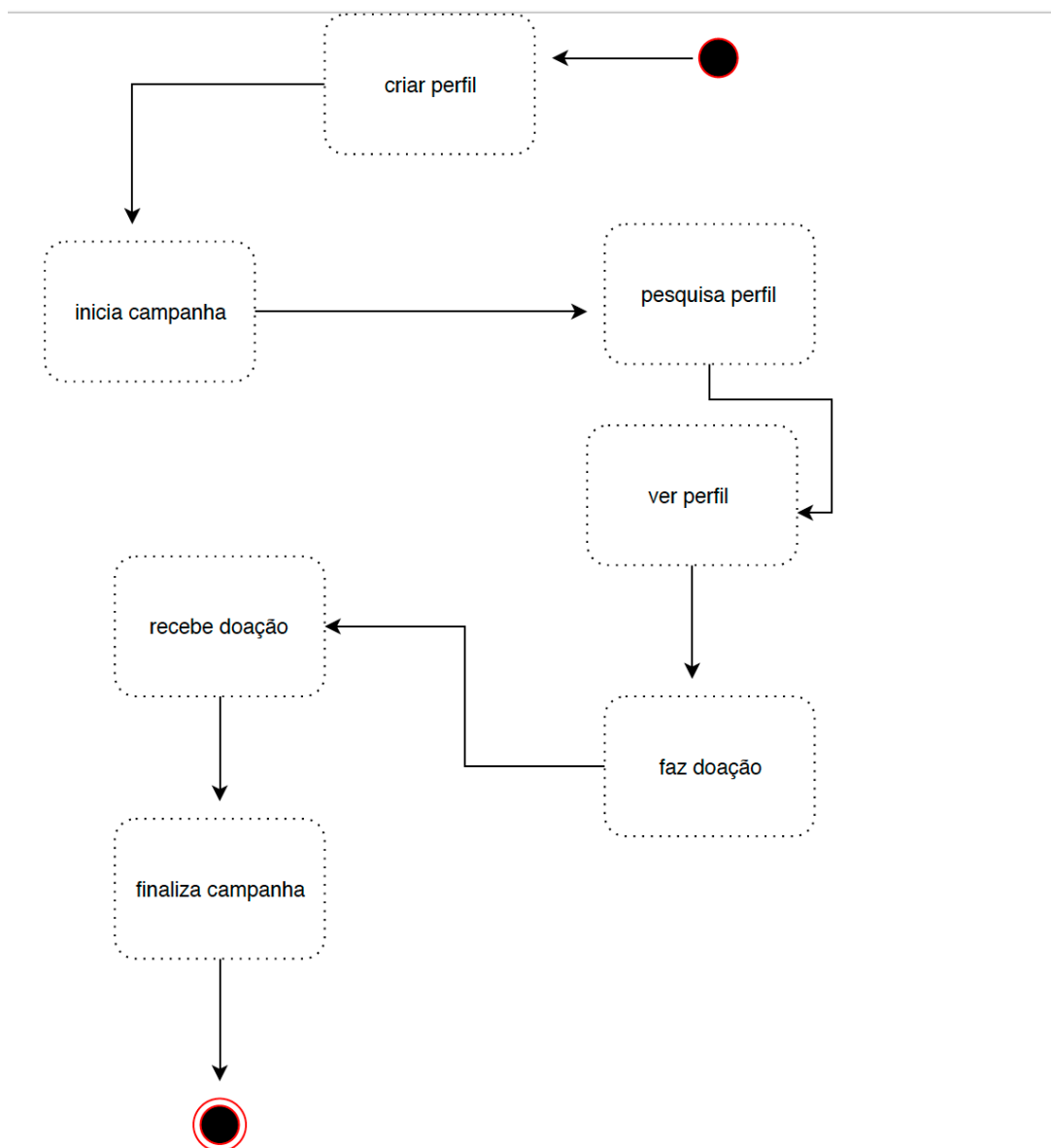


Diagrama de Classes

1. Usuário pode iniciar campanha.
2. Usuário pode fazer doação.
3. Usuário pode iniciar vaquinha.
4. Usuário pode iniciar campanha de apoio.
5. Usuário pode ver perfis.
6. Usuário pode pesquisar perfis.
7. Usuário pode pesquisar campanhas.
8. Usuário pode criar perfil.
9. Usuário pode finalizar campanha.

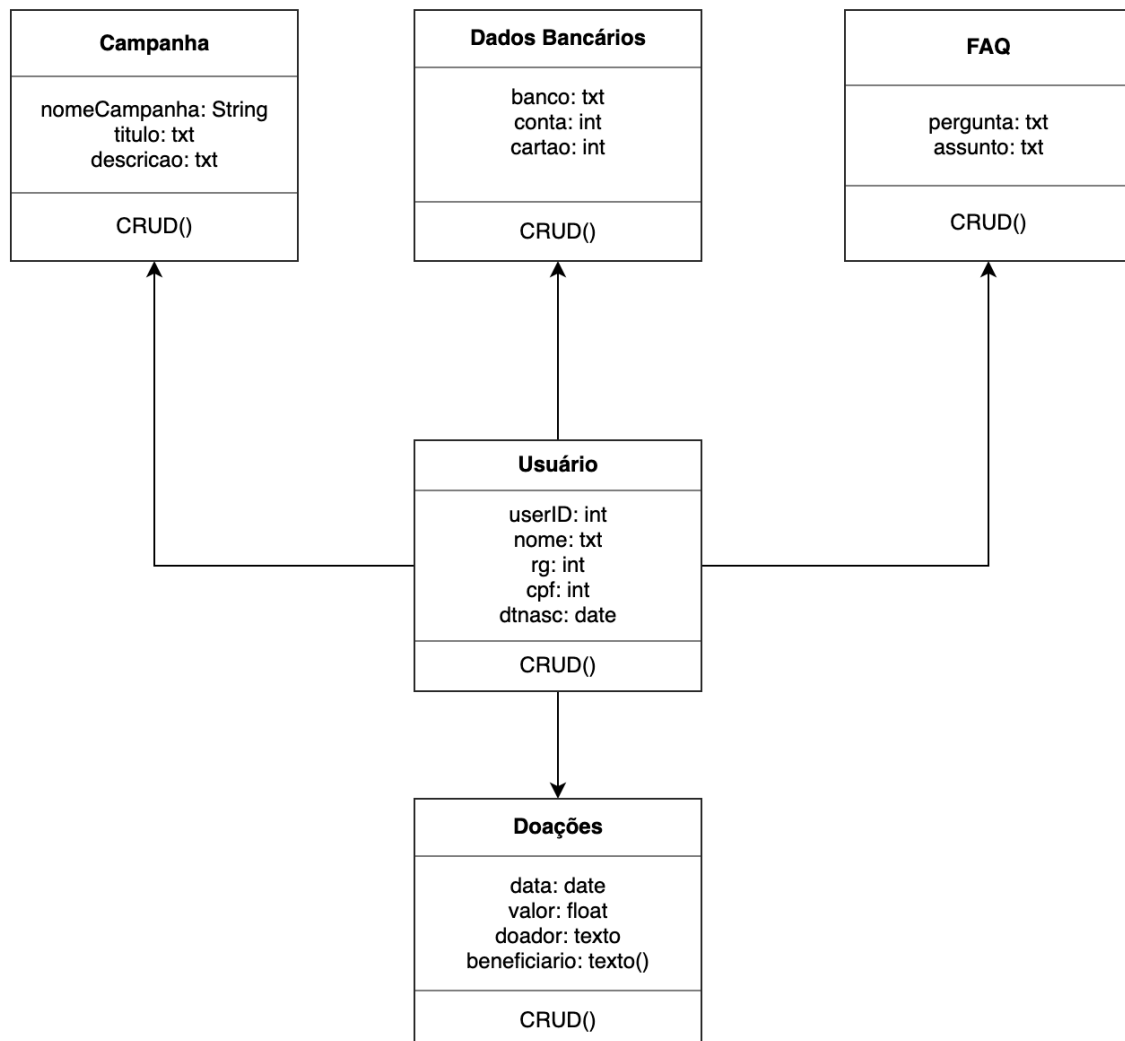
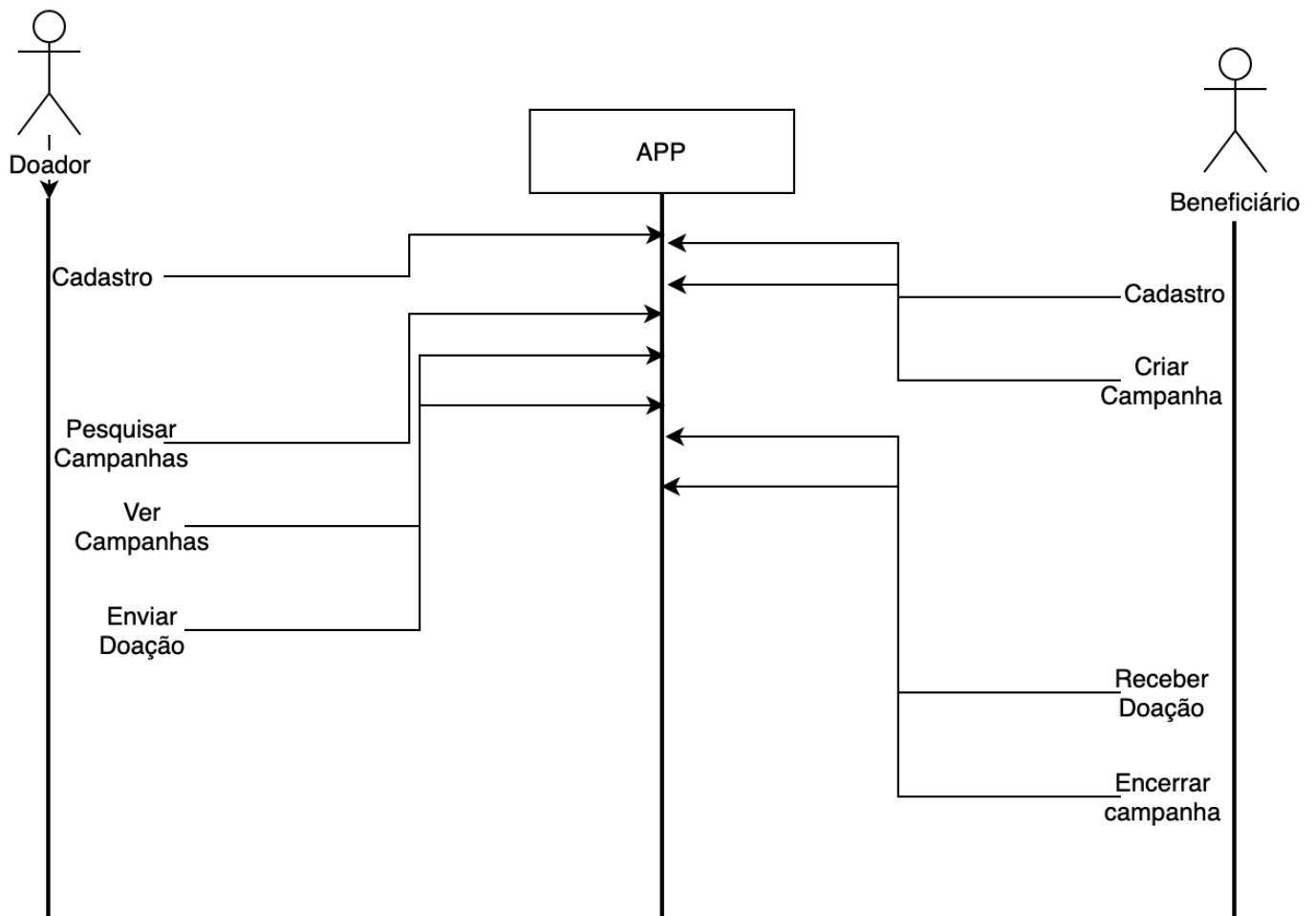


Diagrama de Sequência

1. Usuário pode iniciar campanha.
2. Usuário pode fazer doação.
3. Usuário pode iniciar vaquinha.
4. Usuário pode iniciar campanha de apoio.
5. Usuário pode ver perfis.
6. Usuário pode pesquisar perfis.
7. Usuário pode pesquisar campanhas.
8. Usuário pode criar perfil.
9. Usuário pode finalizar campanha.



Nesta matéria foram instruídos: A Origem de Java, Evolução das Linguagens de Programação (Visão Geral), Heurísticas, Origem da Orientação a Objetos, Conceitos de Orientação a Objetos, Abstração em Orientação a Objetos, Encapsulamento (Proteção de Dados), Polimorfismo (Prevenção de Erros), Interface Gráfica, Componente JOptionPane, showMessageDialog(), showInputDialog(), Componente JFrame, Componente JLabel, Pacotes comumente utilizados, Declaração da Classe e dos Rótulos, Adição de um JLabel simples, Adição de um JLabel com um ícone, Adição de um JLabel em detalhes, Programa Principal, Componente JButton, Declaração dos JButtons, JButton com ícone, Associação dos Eventos dos JButtons, Interface dos Eventos dos JButtons, Execução, Herança, Classes Abstratas, Interface Gráfica com o Usuário, Java Swing, Java Application, Distribuição dos Paradigmas, .

3 Especificação do Projeto.

Todos os usuários que dependem do meio de doação para monetizar suas atividades são relativamente os mais frágeis, é comum a repercussão de dúvidas sobre onde podemos doar, como encontrar informações sobre quem recebe, como será o meio de utilização, talvez até a falta de comprometimento com informações sobre se a atividade foi realmente realizada quando atingida a meta. Para estes usuários, projetamos o Help Us, que ressalva a importância da singularidade dessa informação de usuários em um único banco de dados. Gerando muita facilidade e agilidade de interação entre usuários ajudarem com doações em campanhas que poderiam nunca atingir sua meta.

Diagrama de Atividade

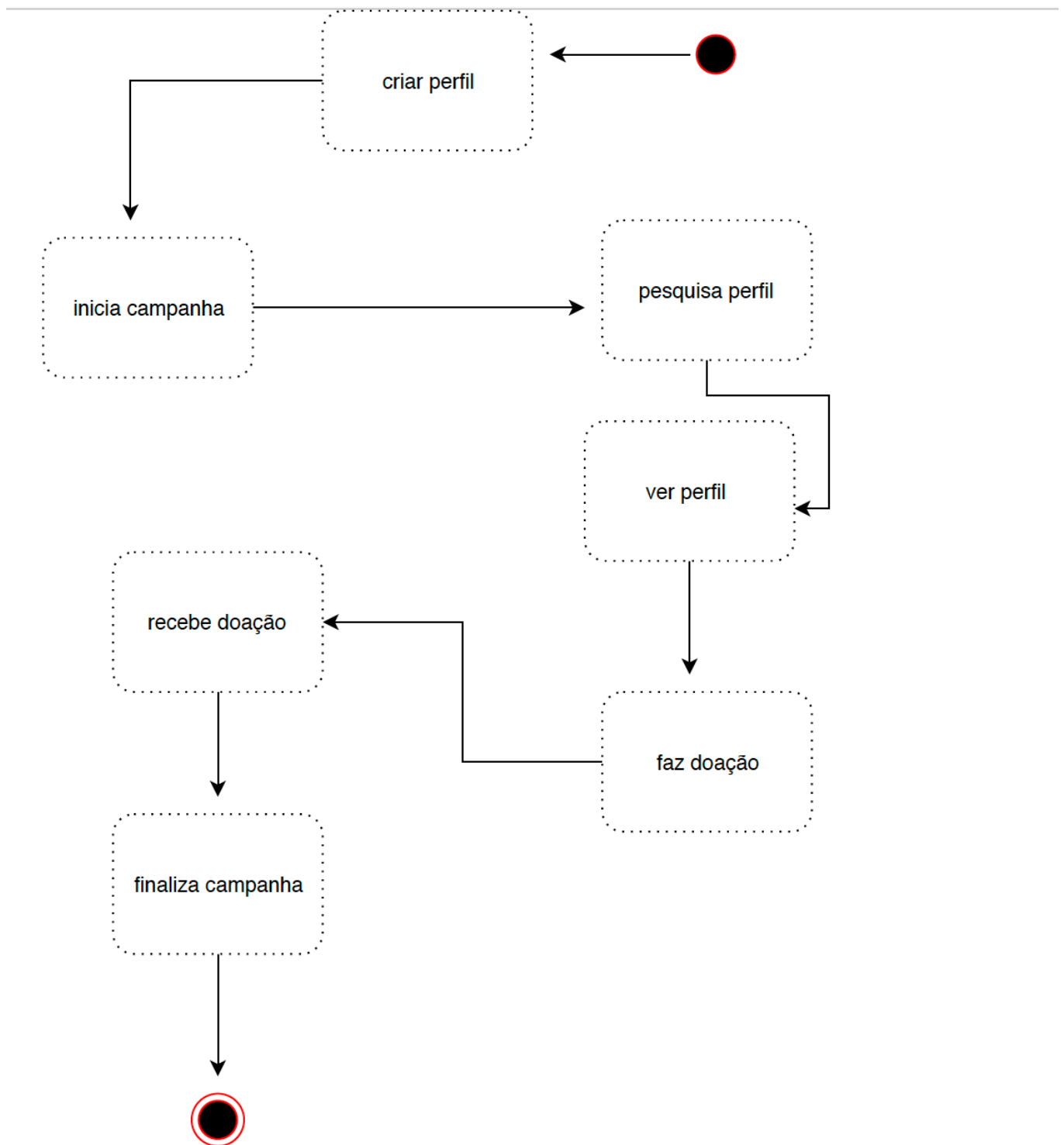


Diagrama de Sequência

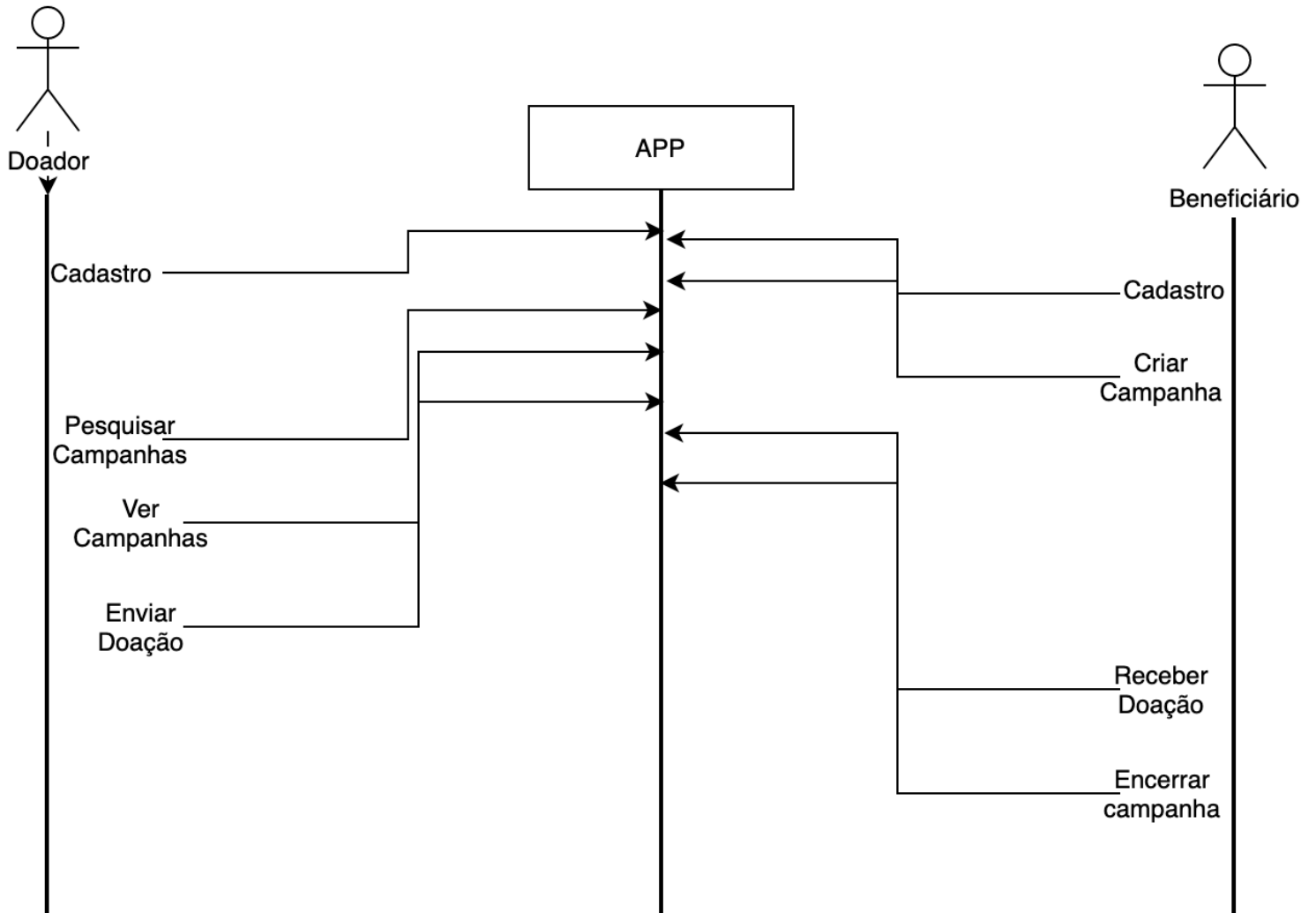


Diagrama de Classe

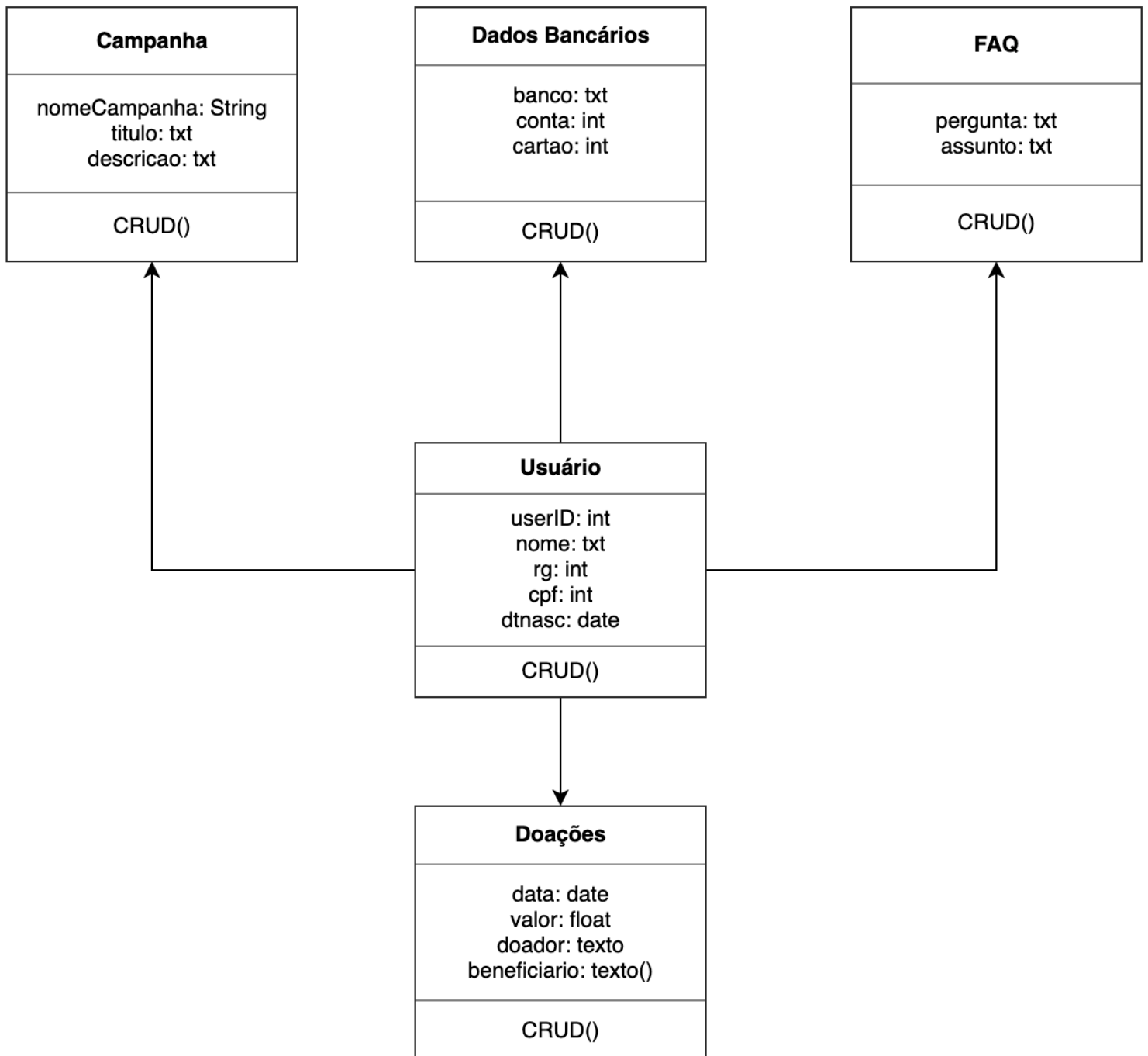
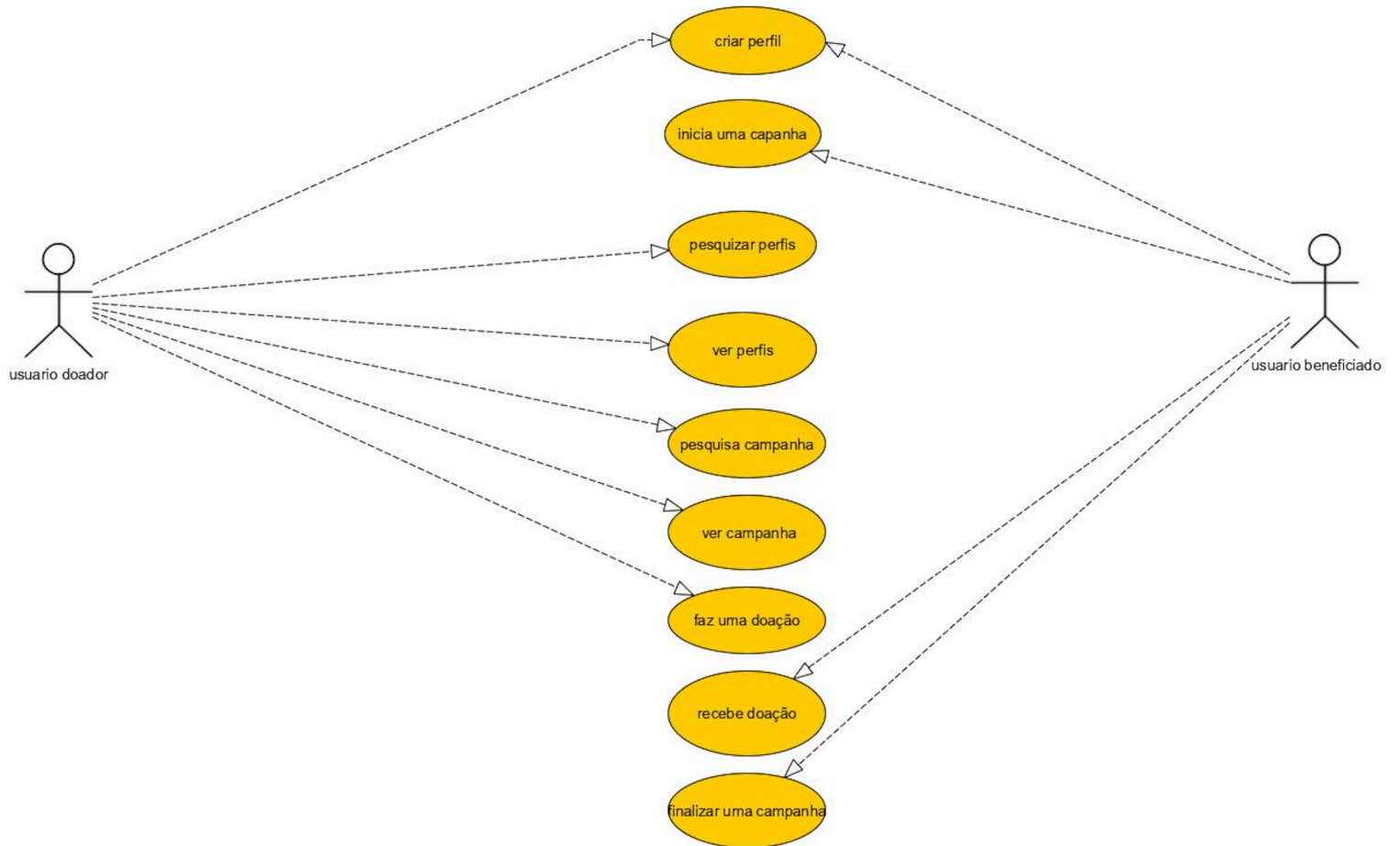


Diagrama de Caso de Uso



5 Implementação da Aplicação

Tela de Login



The screenshot shows a login window titled 'Hus' in the top right corner. The window has a light gray background and a pink border. It contains two input fields: 'Login' and 'Senha' (Password). The 'Login' field is a simple white rectangle with a blue border. The 'Senha' field is a white rectangle with a gray border and a mouse cursor pointing at its right end. Below the 'Senha' field is a button labeled 'Entrar' (Enter). At the bottom of the window, there is a link that says 'Se voce não tiver uma conta clique em cadastrar' (If you don't have an account, click on register) and a button labeled 'Cadastrar' (Register).

Hus

Login

Senha

Se voce não tiver uma conta clique em cadastrar

Tela de Cadastro

Hus

Nome

CPF

data de nascimento

Endereço

Complemento

CEP

Telefone

Celular

Email

Usuario

Senha

CÓDIGO FONTE

```
package apphelpus;

import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.logging.Level;
import java.util.logging.Logger;

public class Clientes {
    private String Nome;
    private String CPF;
    private String DataNascimento;
    private String Endereço;
    private String complemento;
    private String CEP;
    private String Telefone;
    private String Celular;
    private String Email;
    private String Usuario;
    private String Senha;

    public String getNome() {
        return Nome;
    }

    public void setNome(String Nome) {
        this.Nome = Nome;
    }

    public String getCPF() {
        return CPF;
    }

    public void setCPF(String CPF) {
        this.CPF = CPF;
    }

    public String getDataNascimento() {
        return DataNascimento;
    }
}
```

```
public void setDataNascimento(String DataNascimento) {  
    this.DataNascimento = DataNascimento;  
}
```

```
public String getEndereço() {  
    return Endereço;  
}
```

```
public void setEndereço(String Endereço) {  
    this.Endereço = Endereço;  
}
```

```
public String getComplemento() {  
    return complemento;  
}
```

```
public void setComplemento(String complemento) {  
    this.complemento = complemento;  
}
```

```
public String getCEP() {  
    return CEP;  
}
```

```
public void setCEP(String CEP) {  
    this.CEP = CEP;  
}
```

```
public String getTelefone() {  
    return Telefone;  
}
```

```
public void setTelefone(String Telefone) {  
    this.Telefone = Telefone;  
}
```

```
public String getCelular() {  
    return Celular;  
}
```

```
public void setCelular(String Celular) {  
    this.Celular = Celular;  
}
```

```
public String getEmail() {  
    return Email;  
}
```

```
public void setEmail(String Email) {  
    this.Email = Email;  
}
```

```
public String getUsuario() {  
    return Usuario;  
}
```

```
public void setUsuario(String Usuario) {  
    this.Usuario = Usuario;  
}
```

```
public String getSenha() {  
    return Senha;  
}
```

```
public void setSenha(String Senha) {  
    this.Senha = Senha;  
}
```

```
public String salvar(){  
  
    try {  
        FileWriter fw = new FileWriter("Clientes.txt",true);  
        PrintWriter pw = new PrintWriter(fw);  
        pw.println("Nome: " +this.Nome);  
        pw.println("CPF: " +this.CPF);  
        pw.println("Data de nascimento: " +this.DataNascimento);  
        pw.println("Endereço: " +this.Endereço);  
        pw.println("Complemento: " +this.complemento);  
        pw.println("CEP: " +this.CEP);  
        pw.println("Telefone: " +this.Telefone);  
        pw.println("Celular: " +this.Celular);  
        pw.println("Email: " +this.Email);  
        pw.println("Usuario: " +this.Usuario);  
        pw.println("Senha: " +this.Senha);  
        pw.println("");  
        pw.println("");  
        pw.flush();  
        pw.close();  
        fw.close();  
    }  
}
```

```
    } catch (IOException ex) {  
        Logger.getLogger(Clientes.class.getName()).log(Level.SEVERE, null, ex);  
    }  
  
    return "Cadastrado com sucesso";  
}  
  
}
```

6 Conclusão.

Este Projeto Integrado está sendo entregue com base nas informações fornecidas pelos docentes das matérias de Engenharia de Software, Análise e Desenvolvimento de Sistemas e Programação Orientada a Objetos, cuja orientação foi indispensável para o desenvolvimento do software Help Us, que foi criado para cobrir as necessidades mercado, que até a data de entrega, não fornecia um banco de dados exclusivamente voltado para intermediação de doações cujo doadores e beneficiários encontravam-se desagregados de um único meio, surgindo novos softwares para cada ocasião especial, onde a maioria está divulgada e citada nas referências no final deste documento. Todos os acadêmicos envolvidos obtiveram cooperação dos docentes em questão deste projeto, cujo objetivo é demonstrar o aprendizado de cada matéria, em relação aos que participaram para a construção deste software para a apresentação do terceiro semestre como exigência parcial para a aprovação semestral.

Referências

ABACASHI. “A vaquinha online mais barata do mercado.”

Disponível em: <https://abacashi.com/>

Acesso em: 08/03/2019

BREAD, Partake the. “Sinta o poder da gratidão, doe!”

Disponível em: <https://partakethebread.com/>

Acesso em: 08/03/2019

COFRIN. “Aplicativo desenvolvido especialmente para facilitar o acompanhamento e prestação de contas de projetos sociais.”

Disponível em: <https://noslivres.org.br>

Acesso em: 08//2019

PATREON. “Best way for artists and creators to get sustainable income.”

Disponível em: <https://www.patreon.com/>

Acesso em: 08/03/2019

PATEL, Ricken. “Avaaz.ORG - O mundo em ação. (Associação Civil Sem Fins Lucrativos)”

Disponível em: <https://secure.avaaz.org/page/po/>

Acesso em: 08/03/2019

RIBON. “Ajude a salvar vidas todos os dias.”

Disponível em: <https://home.ribon.io>

Acesso em: 08/03/2019

VAKINHA. “O maior site de vaquinhas no Brasil!”

Disponível em: <https://vakinha.com.br>

Acesso em: 08/03/2019