Daniel Akio Okawa
Diego Iida Giraldes
Wellington Felipe Calligaris

Sistema de Agendamento de Transferência Bancária por celular

São Paulo
2015

Daniel Akio Okawa
Diego Iida Giraldes
Wellington Felipe Calligaris

Sistema de Agendamento de Transferência Bancária por celular

Trabalho de conclusão de curso de graduação apresentado à Escola Politécnica da Universidade de São Paulo.

São Paulo
2015

Daniel Akio Okawa
Diego Iida Giraldes
Wellington Felipe Calligaris

Sistema de Agendamento de Transferência Bancária por celular

Trabalho de conclusão de curso de graduação apresentado à Escola Politécnica da Universidade de São Paulo.

Área de concentração:
Engenharia Elétrica - ênfase Computação

Orientador:
Professor Doutor Paulo S. L. M. Barreto
Co-orientador:
Professor Doutor Marcos Antônio Simplício Junior

São Paulo
2015

Ficha catalográfica

Okawa, Daniel
Sistema de Agendamento de Transferência Bancária por celular / D. Okawa, D. Giraldes, W. Calligaris -- São Paulo, 2015.
51 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1.Desenvolvimento de tecnologia 2.Desenvolvimento de software I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II.t. III.Giraldes, Diego IV.Calligaris, Wellington

RESUMO

O presente trabalho visa criar um sistema de agendamento de transferências bancárias por celular, garantido acessibilidade e segurança neste processo ao evitar que informações confidenciais, como a identificação da conta bancária, seja armazenada ou trocada em meios inseguros. Para tanto, foram empregados os conhecimentos adquiridos durante o curso de engenharia elétrica com ênfase em computação, principalmente os relacionados com desenvolvimento de software, gerenciamento de projetos e criptografia, principalmente no que diz respeito a curvas elípticas.

Palavras-chave: Engenharia de software, aplicativo celular, segurança, curvas elípticas, criptografia.

ABSTRACT

This project has as objective create a system of funds transfer schedule by mobile, ensuring accessibility and security in this process to prevent sensitive information, bank account identification, to be stored and exchanged in unsafe ways. Therefore, it contains knowledge acquired during the following course: Electrical Engineering - option Computation, mainly related to software development, project management and cryptography with focus on elliptic curves.

Keywords: Software engineering, mobile app, security, elliptic curves, cryptography.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIAÇÕES

SPB	Sistema de Pagamentos Brasileiro
RSFN	Rede do Sistema Financeiro Nacional
ECC	Curvas Elípticas

1 Sumário

2 Introdução

2.1 Objetivo

O objetivo do projeto é fornecer uma plataforma que possibilite duas pessoas realizarem transferências bancárias utilizando seus aparelhos celulares e a agenda de contatos ao invés das informações bancárias do destinatário utilizando o algoritmo de criptografia de curvas elípticas na transferência da informação envolvida como alternativa mais vantajosa em relação ao que é usado atualmente, proporcionando principalmente maior usabilidade e segurança neste processo.

2.2 Motivação do projeto

Atualmente as transações bancárias envolvem a comunicação entre as partes envolvidas e troca de informação que não é frequentemente utilizada.

Uma transferência bancária segue as seguintes etapas:

- As partes envolvidas devem entrar em acordo com a quantidade a ser transferida.
- Após o acordo, cada parte deve decidir em qual conta o montante deve ser depositado e informar a identificação desta para o depositante.
- De posse desta informação, o depositante realiza o depósito por meio eletrônico ou em uma agência bancária.
- Por sua vez, quem receberá o depósito precisa saber de qual conta este será proveniente para que possa identificar a realização do mesmo.

Em resumo, nosso projeto teve como principal motivação a possibilidade de evitar a necessidade de troca de informação que não é frequentemente utilizada pelas partes envolvidas no depósito: a identificação das contas envolvidas. Notamos que somente quando há necessidade de transferir dinheiro para alguém ou confirmar uma compra na

internet no crédito é que esses dados são utilizados. Portanto, visando evitar um armazenamento inseguro desta informação, seja ele um pedaço de papel ou até mesmo arquivo de computador, e promover acessibilidade e facilidade no processo de transferência bancária por meio de uma tecnologia cada vez mais presente na rotina de todo brasileiro, o smartphone, um sistema foi criado.

Cabe ressaltar que um requisito para que o sistema fosse criado é a existência de uma arquitetura segura que armazene os dados das partes envolvidas, algo que posteriormente será apresentado com mais detalhe neste relatório.

A fim de reconhecer oportunidades e ameaças no mercado, além dos pontos fortes e fracos do projeto, segue uma análise SWOT, cujas siglas possuem os seguintes significados: S de *Strenghts*, W de *Weaknesses*, O de *Opportunities* e T de *Threats*.

2.2.1 Strengths

- Uso de algoritmos de criptografia modernos e leves;
- Fácil de determinar o destinatário das transferências, pois utiliza da agenda de contatos para tal;
- Possibilidade de realizar transferências entre diversos bancos em tempo real;
- Acessibilidade, permitindo que transferências sejam realizadas em qualquer horário e lugar;
- Pouco consumo de banda, bateria e hardware;
- A oportunidade em adquirir competências no que diz respeito a sistemas que envolvem segurança de informação e aplicativos celulares, contribuindo para o futuro profissional dos envolvidos no projeto.

2.2.2 Weaknesses

- Pouca visibilidade em termos de marketing para a divulgação do produto quando comparado a grandes nomes como bancos, *Facebook*, *Google*, entre outros;
- Demora em manutenção e atualização por ser uma equipe pequena de desenvolvedores.

2.2.3 Opportunities

- Brasil é o quinto país no ranking mundial em número de internautas, população e área, e quarto lugar no ranking mundial de número de usuários de celular com acesso à Internet;
- Possui previsão de expansão para os próximos anos no mercado do tipo *e-commerce* de 70% anual, um crescimento estimado de 40 para 180 milhões de consumidores. Sendo *Mobile Payment* a chave para tal crescimento;
- Brasil é número um no ranking entre os países latinos de desenvolvimento de aplicativos;
- Avanços na tecnologia de celulares e algoritmos criptográficos permitem a diversificação na criação de aplicativos, garantindo segurança em transações;
- O projeto seria um novo produto, podendo até mesmo ser caracterizado como a integração de algumas ideias já aplicadas (Whatsapp de banco). Ou seja, inovação e entrada em um mercado ainda emergente.

2.2.4 Threats

- Existência de aplicativos similares, como o do Banco do Brasil e Itaú, que possui funcionalidades similares às de seus respectivos serviços de *internet banking*. Além da *Google Wallet* e *PayPal*;

- Aceitação por parte dos usuários, visto que o Brasil se destaca no índice de fraudes em cartões de crédito e débito, o que influencia num aumento de desconfiança;
- Dependência da autorização e adesão de bancos para permitir uma transferência;
- Dependência de bateria e internet, além da possibilidade de casos de risco como roubos e perdas do aparelho.

2.3 Organização

Este documento é composto de x tópicos que formam a estrutura do relatório final, organizado da seguinte maneira:

Introdução: consiste em três subtópicos apresentados a seguir: Objetivo, Motivação e Organização.

Objetivo: apresenta de forma precisa e concisa o objetivo do projeto.

Motivação: apresenta a motivação e justificativa para a realização do trabalho.

Organização: apresenta a organização do documento: o que cada capítulo, anexo e apêndice aborda.

Aspectos conceituais: apresenta tecnologias e conceitos empregados, além de revisão da literatura.

Especificação do projeto: apresenta as características funcionais, especificação de requisitos e da arquitetura.

Considerações finais: apresenta os resultados atingidos e não atingidos com as respectivas justificativas, assim como contribuições, perspectivas de continuidade e comentário individual de cada um dos membros.

Bibliografia: apresenta as referências utilizadas para a elaboração deste documento.

3 Aspectos conceituais

Serão apresentadas as tecnologias utilizadas no projeto e aspectos conceituais do cenário da Rede do Sistema Financeiro Nacional no que diz respeito à criptografia empregada seguindo para um aprofundamento dos algoritmos utilizados e do algoritmo visto como alternativa pelo grupo de modo a compará-los para justificar a escolha da solução.

Antes de definirmos o cenário do Sistema de Pagamentos Brasileiro (SPB) devemos apresentar brevemente o que é tal sistema e para que existe.

3.1 Sistema de Pagamentos Brasileiro (SPB)

De acordo com o Banco Central do Brasil, órgão cujo papel é o de promover o funcionamento normal e contínuo, além de aprimoramentos de tal sistema, o SPB compreende as entidades, os sistemas e os procedimentos relacionados com o processamento e a liquidação de operações de transferência de fundos, de operações com moeda estrangeira ou com ativos financeiros e valores imobiliários. Ou seja, seu funcionamento adequado é essencial para a estabilidade financeira do país. Além disso, apresenta alto grau de automação, com crescente utilização de meios eletrônicos para transferência de fundos e liquidação de obrigações, representando um pilar central de sustentação da estabilidade financeira, sendo essencial que funcionem de forma segura e eficiente.

Diante disso, cabe definir o cenário descrito conforme o Manual de Segurança da Rede do Sistema Financeiro Nacional (RSFN):

3.2 Cenário da Rede do Sistema Financeiro Nacional

- Apenas transações bancárias acima de 5 mil reais (Transferência Eletrônica Disponível) são protegidas por algoritmos criptográficos;
- Necessário trocar informações bancárias com outra pessoa para poder realizar um depósito bancário (número de conta, agência, entre outras);
- Transações financeiras acontecem 24 horas por dia durante a semana, ou seja, alto volume de requisição;
- Transações abaixo de 5 mil reais pode ter sua integridade afetada, caso sejam interceptadas;
- Utilização de algoritmos criptográficos antigos e pouco eficientes em termos computacionais (RSA).

3.3 Segurança

Segurança é um requisito não funcional, mas neste projeto sua garantia é essencial. Em outras palavras, entende-se que é desejado que os dados sejam mantidos seguros de acessos não autorizados, tendo as cinco características a seguir asseguradas por ele.

- Confidencialidade, que é a capacidade de prevenir vazamento de informações;
- Integridade, que é garantir que um dado não seja modificado sem autorização;
- Disponibilidade, que é a capacidade da informação estar disponível;
- Autenticação, que é capacidade de estabelecer ou confirmar se algo ou alguém é autêntico;
- Não repúdio, que é capacidade de garantir que um usuário ou sistema realmente realizou uma operação.

Tendo isso em vista, exploremos agora tópicos a respeito de criptografia, dando um enfoque para curvas elípticas.

3.4 Conceitos de criptografia

Um dos algoritmos de criptografia mais antigos e simples é o algoritmo simétrico, em que é usado o conceito de chave.

A mensagem a ser enviada usa uma chave associada a uma função, compartilhada entre origem e destino, para criptografar e descriptografar. Por exemplo:

c = E(m,k)➧Sendo “c” a mensagem criptografada, “m” a mensagem original, “k” a chave e E(,) a função. Sendo E(,) uma função que usa as variáveis para montar uma exponencial, “m” igual a 20 e “k” igual a 3, teríamos como valor de “c” ➧ $c = 20^3 = 8000$.

As vantagens desse algoritmo estão na facilidade em implementá-lo e as desvantagens: no problema de escalabilidade, em que o número de chaves depende da quantidade de participantes na troca de mensagens; se a chave for encontrada, a mensagem fica vulnerável; não pode ser usado em assinatura digital.

Diante de tal cenário, o conceito de chave pública (Diffie Hellman) foi desenvolvido. Basicamente a chave era dividida em dois tipos: pública e privada. As etapas para geração das chaves serão descritas a seguir, assim como a lógica de troca de informações.

- Participantes Alice e Bob escolhem dois grandes números primos “a” e “b”; Alice escolhe outro número grande “x” e calcula R = $b^x mod\ a$;
- Bob escolhe outro número grande “y” e calcula S = $b^y mod\ a$;
- Alice recebe “S” do Bob e gera sua chave privada K1 = $S^x mod\ a$;
- Bob age de maneira similar, K2 = $R^y mod\ a$, o resultado é que K1 = K2;

Mensagem a ser enviada é criptografada usando a chave privada do emissor e chave pública do destinatário. E será descriptografada de maneira similar à apresentada anteriormente, usando-se as chaves envolvidas.

As vantagens relacionadas a esse conceito são de que o problema de escalabilidade apresentado pelo algoritmo anterior é retirado, é possível usá-lo nas assinaturas digitais. Já as desvantagens, é mais demorado que o algoritmo simétrico e o tamanho da mensagem criptografada costuma ser muito maior que a mensagem original.

Foquemos agora no algoritmo citado e utilizado na Rede do Sistema Financeiro Nacional, RSA.

RSA foi o primeiro algoritmo de criptografia de dados a possibilitar assinatura digital e uma das grandes inovações em criptografia de chave pública, também chamada por criptografia assimétrica. Apenas para recapitular: na criptografia assimétrica existe um par de chaves, pública e privada. A primeira é distribuída livremente para todos e a segunda, é conhecida apenas pelo seu dono. A mensagem cifrada com a chave pública pode somente ser decifrada pela sua chave privada correspondente.

No RSA as chaves são geradas da seguinte maneira:

- Escolhe de forma aleatória dois números primos grandes “p” e “q”, da ordem de 10 elevado a 100 no mínimo;
- Calcula n = p*q;
- Calcula a função totiente em n: phi(n) = (p-1)(q-1);
- Escolhe um inteiro “e”, tal que 1 < e, < phi(n), de forma que “e” e phi(n), sejam primos entre si.
- Calcula “d” de forma que d*e equivale a 1 * mod(phi(n)), ou seja, “d” seja o inverso multiplicativo de “e” em mod(phi(n)).

No final, a chave pública será definida pelo par (n,e) e a privada pela tripla (p,q,d).

O processo de criptografar a mensagem (m) ocorre da seguinte maneira: mensagem cifrada (c), $c = m^e mod\ n$. E o de descriptografar, $m = c^d mod\ n$.

A escolha de números primeiros “p” e “q” de ordem grande garante que métodos de tentativa e erro para descobrir a chave privada do destinatário se torne computacionalmente inviável.

Sendo assim, concluímos que o RSA aumenta a segurança em comparação com os métodos anteriores, no entanto, possui baixa velocidade no processo de criptografar mensagem.

Antes de passarmos para o algoritmo de curvas elípticas cabe apresentarmos alguns conceitos matemáticos importantes para seu entendimento. Lembrando que, para maiores detalhes, [17] pode ser usado como material de estudo.

3.5 Curvas elípticas

Curvas elípticas são definidas em dois tipos de campos finitos: os de característica ímpar F_p e par F_{2^m}.

Primeiramente, um campo finito é a abstração de um conjunto numérico abeliano, que consiste em um conjunto de elementos que satisfazem as seguintes propriedades:

- Associatividade: (X • Y) • Z = X • (Y • Z)
- Existência da identidade: ∃O ∈ G : X • O = X
- Existência do inverso: ∃~X ∈ G : X • ~X = O
- Comutatividade: X • Y = Y • X

No conjunto F_p a aritmética utilizada é a modular, finita sobre a ordem do conjunto, em outras palavras, pelo número total de elementos.

Já F_{2^m} pode ser representado na forma de um polinômio, sendo os coeficientes representados de maneira binária ao fazer parte do conjunto 0 e 1. Verificamos a garantia das propriedades grupo abeliano fazendo operações aritméticas similares à binária.

Exemplo:

$\mathbb{F}_{2^4}$

0	z^2	z^3	z^3+z^2
1	z^2+1	z^3+1	z^3+z^2+1
z	z^2+z	z^3+z	z^3+z^2+z
$z+1$	z^2+z+1	z^3+z+1	z^3+z^2+z+1.

The following are some examples of arithmetic operations in $\mathbb{F}_{2^4}$ with reduction polynomial $f(z)=z^4+z+1$.

(i) Addition: $(z^3+z^2+1)+(z^2+z+1)=z^3+z$.

(ii) Subtraction: $(z^3+z^2+1)-(z^2+z+1)=z^3+z$. (Note that since $-1=1$ in $\mathbb{F}_2$, we have $-a=a$ for all $a\in\mathbb{F}_{2^m}$.)

(iii) Multiplication: $(z^3+z^2+1)\cdot(z^2+z+1)=z^2+1$ since

$$(z^3+z^2+1)\cdot(z^2+z+1)=z^5+z+1$$

and

$$(z^5+z+1) \bmod (z^4+z+1)=z^2+1.$$

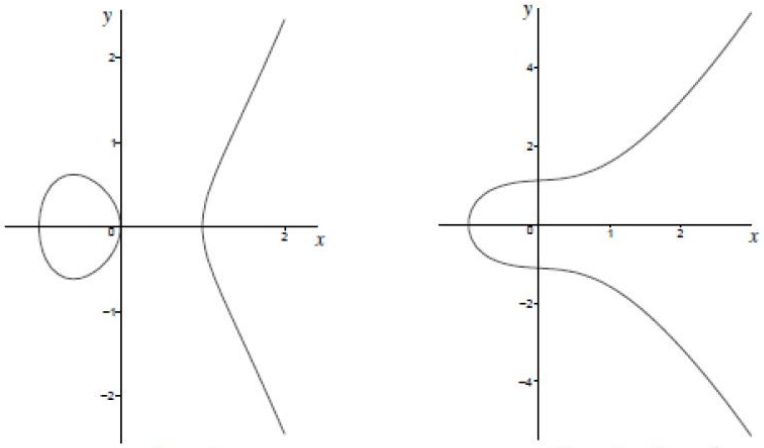
(iv) Inversion: $(z^3+z^2+1)^{-1}=z^2$ since $(z^3+z^2+1)\cdot z^2 \bmod (z^4+z+1)=1$.

Figura 1: Exemplo de operações aritméticas em campo finito par.

Visto isso, prossigamos para a definição de curvas elípticas. Estas curvas são equações que definem um conjunto de pontos em um plano bidimensional, possuindo a seguinte forma:

$$y^2 + axy + by = x^3 + cx^2 + dx + e$$

Os coeficientes precisam respeitar uma condição matemática que impede que a curva tenha uma tangente em qualquer ponto, traduzida pela condição do discriminante ter valor diferente de zero.



(a) $E_1: y^2=x^3-x$ (b) $E_2: y^2=x^3+\frac{1}{4}x+\frac{5}{4}$

Figura 2: Exemplo de curvas elípticas.

Dada uma equação de curvas elípticas e aplicando as condições estabelecidas pelo grupo abeliano, conseguimos reunir um conjunto de pontos junto ao ponto identidade da curva definido como O.

Sendo assim, podemos agora apresentar o processo de criptografar mensagens usando o algoritmo de curvas elípticas.

Primeiro um inteiro grande é selecionado, ou um primo ou um inteiro da forma 2^m e os coeficientes da equação da curva. Dessa maneira o grupo elíptico de pontos é definido. Em seguida um ponto G cuja ordem seja um valor muito grande “n” é escolhido.

O emissor seleciona um inteiro $n_A$menor que “n” como chave privada e gera sua chave pública $P_A = n_A * G$. Para cifrar a mensagem “M”, mapeada em um ponto P_M, um inteiro aleatório é escolhido de modo que o texto criptografado seja o par de pontos $C_M = \{ kG, P_M + kP_B\}$, sendo P_B a chave pública do receptor criada de maneira similar à do emissor.

O processo de descriptografar ocorre da seguinte maneira: o receptor multiplica o primeiro ponto pela sua chave privada e subtrai do segundo ponto.

$$P_M + kP_B - n_B(kG) = P_M$$

A dificuldade em desvendar a mensagem criptografada sem a chave privada está no fato de ser computacionalmente inviável calcular “k” a partir de “kG”, sendo este problema conhecido como o Problema do Logaritmo Discreto Elíptico, que é de complexidade NP.

Vimos que os algoritmos de RSA usam produtos de números primos para criptografar a mensagem, no entanto, com o progresso em fatorar, as chaves públicas de RSA devem possuir longos bits para fornecer a segurança adequada. Entretanto, em um grande grupo finito, encontrar soluções para equações que envolvem números reais ou complexos é completamente difícil e conhecido como o problema citado anteriormente. Conhecendo a equação característica de curva elíptica e o associando a este problema que ainda não teve publicada nenhuma resolução, vemos que as chaves escolhidas podem ser muito mais curtas para um nível comparável da segurança, o que implica também em um tempo de processamento menor para os envolvidos na troca de mensagens.

Tabela 1. Comparação de tamanho da chave (em bits)

Algoritmos simétricos	Curvas Elípticas	RSA
80	163	1024
112	233	2240
128	283	3072
192	409	7680
256	571	15360

Tabela 2. Performance na geração de chave

Tamanho da Chave (bits)		Tempo (s)	
RSA	Curvas Elípticas	RSA	Curvas Elípticas
1024	163	0,16	0,08
2240	233	7,47	0,18
3072	283	9,80	0,27
7680	409	133,90	0,64
15360	571	679,06	1,44

Tabela 3. Comparação completa entre algoritmos RSA e curvas elípticas.

Parameters	ECC	RSA
Computational Overheads	Roughly 10 times than that of RSA can be saved	More than ECC
Key Sizes	System parameters and key pairs are shorter for the ECC.	System parameters and key pairs are larger for the RSA.
Bandwidth saving	ECC offers considerable bandwidth savings over RSA	Much less bandwidth saving than ECC
Key Generation	Faster	Slower
Encryption	Much Faster than RSA	At good speed but slower than ECC
Decryption	Slower than RSA	Faster than ECC
Small Devices efficiency	Much more efficient	Less efficient than ECC

A vantagem está no aumento de velocidade de processamento, menos uso de memória e menor tamanho de chave. Já desvantagem está na pouca exploração de tal técnica.

3.6 Tecnologias empregadas

O grupo optou pela plataforma Android para desenvolvimento do aplicativo celular, utilizando da ferramenta Android Studio versão Windows.

Já o framework para aplicação web foi o Spark, que não segue o padrão MVC, mas que é de rápida implementação.

Sendo assim, a linguagem de programação utilizada foi Java.

4 Especificações do projeto

Este tópico apresentará todos os requisitos funcionais e não funcionais do sistema, além de sua arquitetura, casos de uso, diagramas de classe e de sequência e protótipos.

4.1 Requisitos funcionais

- Cadastrar usuário por meio de um aplicativo;
- Enviar código de confirmação de cadastro por SMS Associar informações bancárias do usuário;
- Baseado na sua lista de contatos, importar lista de contatos cadastrados no sistema Cadastro de PIN *number* para realização de operações no sistema;
- Enviar solicitação de depósito para contato;
- Solicitação de PIN *number* para confirmar intenção de depósito;
- Receber notificação de solicitação de depósito;
- Aprovar notificação de depósito;
- Reprovar notificação de depósito;
- Notificar transações pendentes aos bancos envolvidos;
- API de comunicação entre banco e sistema;
- Interface WEB para o usuário cancelar depósitos agendados;
- Visualização de histórico de transações feitas no sistema.

4.2 Requisitos Não-Funcionais

- Disponibilidade de serviço de 99% em um mês;
- Funcionalidades devem ser realizadas com até 5 toques;
- Informação criptografada na comunicação;

- Informação criptografada no banco de dados;
- Mensagens enviadas devem possuir menos de 10KB;
- Criptografar uma mensagem deve consumir menos de 100mAH;
- Não ocupar mais de 150MB quando instalado em disco.

4.3 Arquitetura

Figura 3: Modelo de arquitetura do sistema.

A arquitetura proposta possui dois servidores: um para o banco e outro para hospedar o processamento de dados da aplicação. Já o usuário possui duas interfaces para acessar os recursos providos pela aplicação.

O servidor do banco é capaz de receber notificações de transações a serem consolidadas e notificar a aplicação de que uma ou um lote de transações foi consolidada. Além disso, o servidor também tem a funcionalidade de prover um *token* quando o conjunto de dados fornecidos corresponde a um usuário cadastrado no seu banco de dados, permitindo que a aplicação não armazene os dados bancários do usuário no banco de dados, e sim o *token* correspondente.

O servidor da aplicação atua com intermediador da comunicação entre usuário e banco. Implementação das rotinas de confirmação de cadastro, agendamento de transações bancárias, visualização de histórico de transações e cancelamento de agendamento sãs as funções presentes nesse servidor.

Já as interfaces de usuário são responsáveis por realizar chamadas às funções implementadas no servidor de aplicação.

4.4 Funcionamento do sistema

4.4.1 Instalação e cadastro

O usuário precisa instalar uma aplicação no seu aparelho celular. Após a instalação, o usuário preenche um cadastro com seus dados pessoais e dados de identificação de suas contas bancárias. Os dados necessários para o cadastro são:

1. Dados pessoais
    I. Nome
    II. CPF
    III. Data de nascimento
    IV. Número do celular
2. Dados bancários
    I. Banco
    II. Número de agência e de conta corrente

Após o cadastro do usuário, uma mensagem de texto é enviada com um código de confirmação. Após a confirmação do usuário, é necessário cadastrar um código de verificação que será solicitado sempre que o usuário for realizar uma transação no aplicativo e outro código para suspensão de transações. Isso serve para adicionar uma camada de segurança no aplicativo, assim tenta-se evitar que o aplicativo seja utilizado por pessoas não autorizadas. Após o cadastro do código de verificação, sua agenda de contatos é fornecida ao aplicativo para buscar quais contatos que o usuário

possui na agenda utilizam o aplicativo. O aplicativo busca todos os contatos no banco de dados do sistema e atualiza a lista de contatos do usuário no aplicativo.

4.4.2 Realizando transferências

Após realizar o cadastro, o usuário está pronto para realizar transferências entre seus contatos. Para tanto, deve selecionar o contato para quem deseja transferir o dinheiro, selecionar de qual conta bancária o dinheiro deve sair e enviar a solicitação de transferência para o destinatário. Caso o usuário possua somente uma conta cadastrada, a seleção de conta não é realizada. Uma solicitação de transferência é criada e marcada como pendente.

O destinatário recebe uma notificação de que há uma transferência pendente no momento e mais detalhes de transferência são mostrados. Assim ele pode escolher entre aprovar ou reprovar essa transferência.

O resultado será notificado ao usuário remetente. Se o destinatário reprovar a transferência, a solicitação de transferência é marcada como reprovada e mais nenhuma outra ação acontece. Caso contrário, ele deve selecionar em qual conta bancária o montante transferido deve ser depositado. Caso o usuário possua somente uma conta cadastrada, a seleção de conta não é realizada.

Após a seleção, a solicitação é marcada como confirmada e duas transações são criadas. Uma para retirar dinheiro da conta origem e outra para depositar o mesmo montante na conta destino. Ambas sendo marcadas como confirmadas.

4.4.3 Cancelar uma transação

Como o aparelho celular é um equipamento muito vulnerável a extravios, existe uma ferramenta que permite ao usuário cancelar transações que ainda não foram

efetuadas. O usuário acessa um website e fornece o código de suspensão e pode marcar as transações suspeitas como suspensas.

4.4.4 Consolidação de transações

Ao final do dia, o sistema envia todas transações marcadas como confirmadas e criadas há mais de 24 horas são enviadas aos bancos responsáveis e marca-as como efetuadas. Depois desse ponto, toda o controle sobre o andamento das transações é dos bancos responsáveis.

Visualização de histórico de transações.

O usuário pode ver o histórico de todas as transações realizadas utilizando o aplicativo. Há a possibilidade de aplicar filtros para ver somente transações que aconteceram durante um período de tempo, para um destinatário específico e/ou por estado de transação (confirmada/efetuada/cancelada).

4.5 Casos de Uso

1. Cadastrar usuário na plataforma via aplicativo

1.1 Atores

1.1.1 [ATR01] Usuário do aplicativo

Usuário padrão do aplicativo.

1.2 Pré-condições

1.2.1 Usuário sem cadastro na plataforma

O usuário não pode possuir cadastro na plataforma.

1.2.2 Usuário com aplicativo instalado no celular

É necessário que o aplicativo esteja instalado no aparelho do usuário.

1.3 Pós-Condições

1.3.1 Usuário com perfil temporário na plataforma

O usuário vai ter um perfil temporário, que aguarda a confirmação de cadastro para criação de um perfil definitivo.

1.4 Fluxos de Evento

1.4.1 Fluxo Básico

- Usuário provem informações pessoais obrigatórias – Nome, CPF, Data de nascimento e Foto (opcional)
- Usuário envia informações para o servidor
- Servidor armazena informações no banco de dados
- Servidor cria perfil temporário para o usuário
- Mensagem de sucesso é exibida ao usuário.

1.4.2 Fluxo de exceção

- Usuário provem alguma das informações obrigatórias em branco
- Usuário enviar informações para o servidor
- Servidor analisa informações enviadas
- Servidor detecta falta de informações obrigatórias
- Servidor reenvia informações exibidas
- Aplicativo exibe tela de cadastro novamente com as informações enviadas
- Aplicativo exibe mensagem de erro nos campos que estão com erro.

2. Enviar código de confirmação via SMS

2.1 Atores

2.1.1 [ATR01] Usuário

2.2 Pré-condições

2.2.1 Usuário com cadastro temporário na plataforma

O usuário já enviou as informações obrigatórias para criar o perfil na plataforma de agendamento de transações bancárias.

2.3 Pós-Condições

2.3.1 Usuário com posse do código de verificação

O usuário ainda possui seu perfil temporário, porém agora está de posse do código que faz com que seu perfil passe de temporário para consolidado.

2.4 Fluxos de Evento

2.4.1 Fluxo Básico

- Servidor cria perfil temporário do usuário.
- Servidor gera código para verificar perfil do usuário.
- Servidor envia código de verificação para o usuário.

2.4.2 Fluxo de exceção

2.4.3 Erro na criação do perfil temporário verificação

- Servidor não consegue criar perfil temporário.
- Mensagem de erro é enviada para o usuário.

2.4.4 Erro na geração do código de verificação

- Servidor cria perfil temporário.
- Servidor não consegue gerar código de verificação de perfil do usuário.
- Mensagem de erro é enviada para o usuário.

2.4.5 Erro no envio do código de verificação

- Servidor cria perfil temporário.
- Servidor gera código de verificação de perfil do usuário.
- Servidor não consegue enviar código de verificação para o usuário.
- Mensagem de erro é enviada para o usuário.

3. Associar informações bancárias ao perfil do usuário

3.1 Atores

3.1.1 [ATR01] Usuário

3.1.2 [ATR02] Banco

3.2 Pré-condições

3.2.1 Usuário com perfil na plataforma

Usuário possui um perfil na plataforma, que pode ou não ter informações bancárias associadas a esse perfil.

3.3 Pós-Condições

3.3.1 Usuário com novo token associado ao seu perfil

Informações enviadas pelo usuário são processadas pela plataforma e o usuário pode utilizá-las para agendar transferências bancárias.

3.4 Fluxos de Evento

3.4.1 Fluxo Básico

- Usuário envia informações bancárias para o servidor (Banco, número da agência bancária, número da conta corrente).
- Servidor repassa informações para o banco.
- Banco verifica se informações são válidas.
- Banco gera token para essa conta bancária.
- Banco envia token para o servidor.
- Servidor associa token enviado pelo banco com o perfil do usuário.
- Servidor envia mensagem de sucesso para o usuário.

3.4.2 Fluxo de exceção

3.4.3 Usuário não envia todas informações obrigatórias

- Usuário envia informações para o servidor.
- Servidor verifica que existem informações obrigatórias que não foram enviadas.
- Servidor reenvia informações para o usuário.
- Informações enviadas são exibidas para o usuário.
- Campos com erros são marcados e mensagens de erro apropriadas são

exibidas.

3.4.4 Usuário envia todas informações inválidas

- Usuário envia informações para o servidor.
- Servidor repassa informações para o banco.
- Banco verifica que informações não conferem com as presentes no seu banco de dados.
- Banco envia mensagem de erro para o servidor.
- Servidor reenvia informações enviadas pelo usuário para o usuário.
- Servidor envia mensagem de erro de veracidade de informações.
- Informações são exibidas para o usuário.
- Mensagem de erro de veracidade é exibida para o usuário.

3.4.5 Comunicação com banco indisponível

- Usuário envia informações para o servidor.
- Servidor não consegue repassar informações para o banco.
- Servidor gera mensagem de erro de comunicação com o banco.
- Mensagem de erro de comunicação é enviada para o usuário.

4. Importar contatos de usuário

4.1 Atores

4.1.1 [ATR01] Usuário

4.2 Pré-condições

4.2.1 Usuário com perfil na plataforma

Usuário possui um perfil na plataforma, que pode ou não ter uma lista de contatos disponíveis para agendar transferências bancárias.

4.3 Pós-Condições

4.3.1 Usuário com lista de contatos que podem agendar transferências

A lista de contatos que podem receber transferências bancárias será criada no caso de não existir anteriormente ou atualizada no caso de existir.

4.4 Fluxos de Evento

4.4.1 Fluxo Básico

- Usuário envia lista de contatos existentes no seu telefone.
- Servidor busca pelo perfil de todos os números fornecidos pelo usuário.
- Servidor atualiza lista de contatos para os quais o usuário pode agendar transferências.
- Servidor reenvia lista de contatos disponíveis para agendamentos para o usuário.
- Usuário visualiza lista de contatos disponíveis para agendamentos.

4.4.2 Fluxo de exceção

Não há.

5. Cadastrar PIN number para realizar agendamentos na plataforma

5.1 Atores

5.1.1 [ATR01] Usuário

5.2 Pré-condições

5.2.1 Usuário com perfil na plataforma

Usuário possui um perfil na plataforma.

5.3 Pós-Condições

5.3.1 Usuário com PIN number atualizado

PIN number (mínimo de 4 dígitos) é cadastrado localmente no aplicativo.

5.4 Fluxos de Evento

5.4.1 Fluxo Básico

- Usuário digita seu PIN number no aplicativo.
- Aplicativo registra seu PIN number.
- Mensagem de sucesso é exibida para o usuário.

5.4.2 Fluxo de exceção

5.4.3 PIN number possui menos de 4 dígitos

- Usuário digita seu PIN number no aplicativo.
- Aplicativo detecta que há menos de 4 dígitos.

- Mensagem de erro é exibida para o usuário.

6. Enviar solicitação de depósito para contato

6.1 Atores

6.1.1 [ATR01] Usuário do aplicativo

Usuário padrão do aplicativo.

6.2 Pré-condições

6.2.1 Usuário com cadastro e conectado na plataforma

O usuário deve possuir cadastro na plataforma e estar conectado na plataforma, já tendo inserido o PIN de confirmação de intenção de depósito.

6.2.2 Usuário possuir lista de contatos

É necessário que o usuário já possua lista de contatos.

6.3 Pós-Condições

6.3.1 Solicitação de depósito é criada e marcada como pendente

O usuário que fez a solicitação é notificado a respeito do depósito.

6.4 Fluxos de Evento

6.4.1 Fluxo Básico

- [PRT01] Tela que habilita seleção dos parâmetros necessários para a criação da solicitação de depósito.
- Usuário seleciona contato, quantia e conta da qual será retirada a quantia.
- Sistema verifica o valor da quantia.
- Usuário envia solicitação.
- Servidor recebe informação.
- Servidor cria notificação de depósito para quem solicitou.

6.4.2 Fluxo de exceção

- Usuário provê alguma das informações obrigatórias em branco
- Usuário enviar informações para o servidor
- Servidor analisa informações enviadas
- Servidor detecta falta de informações obrigatórias
- Servidor reenvia informações exibidas

- Aplicativo exibe tela de criação de depósito novamente com as informações enviadas
- Aplicativo exibe mensagem de erro nos campos que estão com erro.

7. Receber notificação de solicitação de depósito

7.1 Atores

7.1.1 [ATR01] Usuário

7.2 Pré-condições

7.2.1 Usuário com cadastro e conectado na plataforma

O usuário deve possuir cadastro na plataforma e estar conectado na plataforma.

7.2.2 Um outro usuário efetua solicitação de depósito

Um outro usuário efetuou solicitação de depósito.

7.3 Pós-Condições

7.3.1 Usuário recebe notificação de solicitação de depósito

O usuário recebe a notificação de solicitação de depósito.

7.4 Fluxos de Evento

7.4.1 Fluxo Básico

- Servidor gera notificação de depósito para quem receberá após ela ter sido criada.

7.4.2 Fluxo de exceção

7.4.3 Erro no acesso ao servidor

- Sem comunicação, não há recebimento de notificação.
- Mensagem de erro é enviada para o usuário.

8. Solicitação de PIN para confirmar intenção de depósito

8.1 Atores

8.1.1 [ATR01] Usuário

8.1.2 [ATR02] Banco

8.2 Pré-condições

8.2.1 Usuário com perfil na plataforma e com lista de contatos

O usuário deve possuir cadastro na plataforma, estar conectado na plataforma e possuir lista de contatos.

8.3 Pós-Condições

8.3.1 Usuário é direcionado para a tela de depósito

Informações enviadas pelo usuário são processadas pela plataforma e o usuário recebe acesso para a criação de depósito.

8.4 Fluxos de Evento

8.4.1 Fluxo Básico

- Usuário envia informações do PIN.
- Servidor faz validação.
- Banco verifica se informação é válida.
- Sistema exibe tela para criação de notificação de depósito.

8.4.2 Fluxo de exceção

8.4.3 Usuário não envia informação correta

- Usuário envia PIN errado para o servidor.
- Servidor verifica que informação é inválida.
- Sistema exibe mensagem de erro para o usuário e número de tentativas é incrementado.

8.4.4 Usuário ultrapassa o número de tentativas de inserção do PIN

- Usuário efetua três tentativas.
- Servidor verifica informação.
- Sistema bloqueia uso do aplicativo.

8.4.5 Comunicação com banco indisponível

- Usuário envia PIN para o servidor.
- Servidor não consegue verificar no banco.
- Servidor gera mensagem de erro de comunicação com o banco.
- Mensagem de erro de comunicação é enviada para o usuário.

9. Aprovar notificação de depósito

9.1 Atores

9.1.1 [ATR01] Usuário

9.2 Pré-condições

9.2.1 Usuário conectado na plataforma e com notificação pendente

Usuário possui um perfil na plataforma, está conectado e possui notificação pendente de depósito.

9.3 Pós-Condições

9.3.1 Criação da solicitação de depósito

Solicitação de depósito é criada para que ocorra no término do dia.

9.4 Fluxos de Evento

9.4.1 Fluxo Básico

- Usuário visualiza notificação pendente e seleciona conta bancária que deseja receber o depósito, aprovando.
- Servidor recebe informações.
- Solicitação de depósito é criada.

9.4.2 Fluxo de exceção

9.4.2.1 Comunicação com banco indisponível

- Servidor não consegue verificar banco.
- Mensagem de erro é exibida para o usuário.

10. Reprovar notificação de depósito

10.1 Atores

10.1.1 [ATR01] Usuário

10.2 Pré-condições

10.2.1 Usuário conectado e com notificação pendente

Usuário possui um perfil na plataforma, está conectado e possui notificação pendente de depósito.

10.3 Pós-Condições

10.3.1 Notificação reprovada

Notificação é reprovada e tem status alterado, impossibilitando futuras interações.

10.4 Fluxos de Evento

10.4.1 Fluxo Básico

- Usuário visualiza notificação pendente e reprova.
- Servidor recebe informações e muda status da notificação.

10.4.2 Fluxo de exceção

10.4.2.1 Comunicação com banco indisponível

- Servidor não consegue verificar banco.
- Mensagem de erro é exibida para o usuário.

11. Cancelar Depósito

11.1 Atores

11.1.1 [ATR01] Usuário do aplicativo

Usuário padrão do aplicativo.

11.2 Pré-condições

11.2.1 Usuário com aplicativo instalado no celular

É necessário que o aplicativo esteja instalado no aparelho do usuário.

11.2.2 Usuário cadastrado no sistema

É necessário que o usuário esteja cadastrado no sistema.

11.2.3 Depósito pendente

Existência de depósito pendente no sistema e não notificado ao banco.

11.2.4 Usuário logado no sistema via WEB

O usuário deve acessar interface web para cancelamento.

11.3 Pós-Condições

11.3.1 Depósito cancelado

O depósito pendente selecionado é cancelado e não será notificado ao banco.

11.4 Fluxos de Evento

11.4.1 Fluxo Básico

- Usuário seleciona depósito a ser cancelado
- Usuário confirma o cancelamento do depósito
- Servidor cancela o depósito
- Mensagem de sucesso é exibida ao usuário.

11.4.2 Fluxo de exceção

- Usuário não possui depósito pendente
- Sistema envia mensagem
- Usuário mensagem é exibida ao usuário

12. Visualizar Transações

12.1 Atores

12.1.1 [ATR01] Usuário do aplicativo

Usuário padrão do aplicativo

12.2 Pré-condições

12.2.1 Usuário com aplicativo instalado no celular

É necessário que o aplicativo esteja instalado no aparelho do usuário.

12.2.2 Usuário cadastrado no sistema

É necessário que o usuário esteja cadastrado no sistema.

12.2.3 Usuário logado no sistema

O usuário deve estar logado no sistema para visualização.

12.3 Pós-Condições

12.3.1 Usuário visualiza seu histórico

O usuário visualiza seu histórico de transações

12.4 Fluxos de Evento

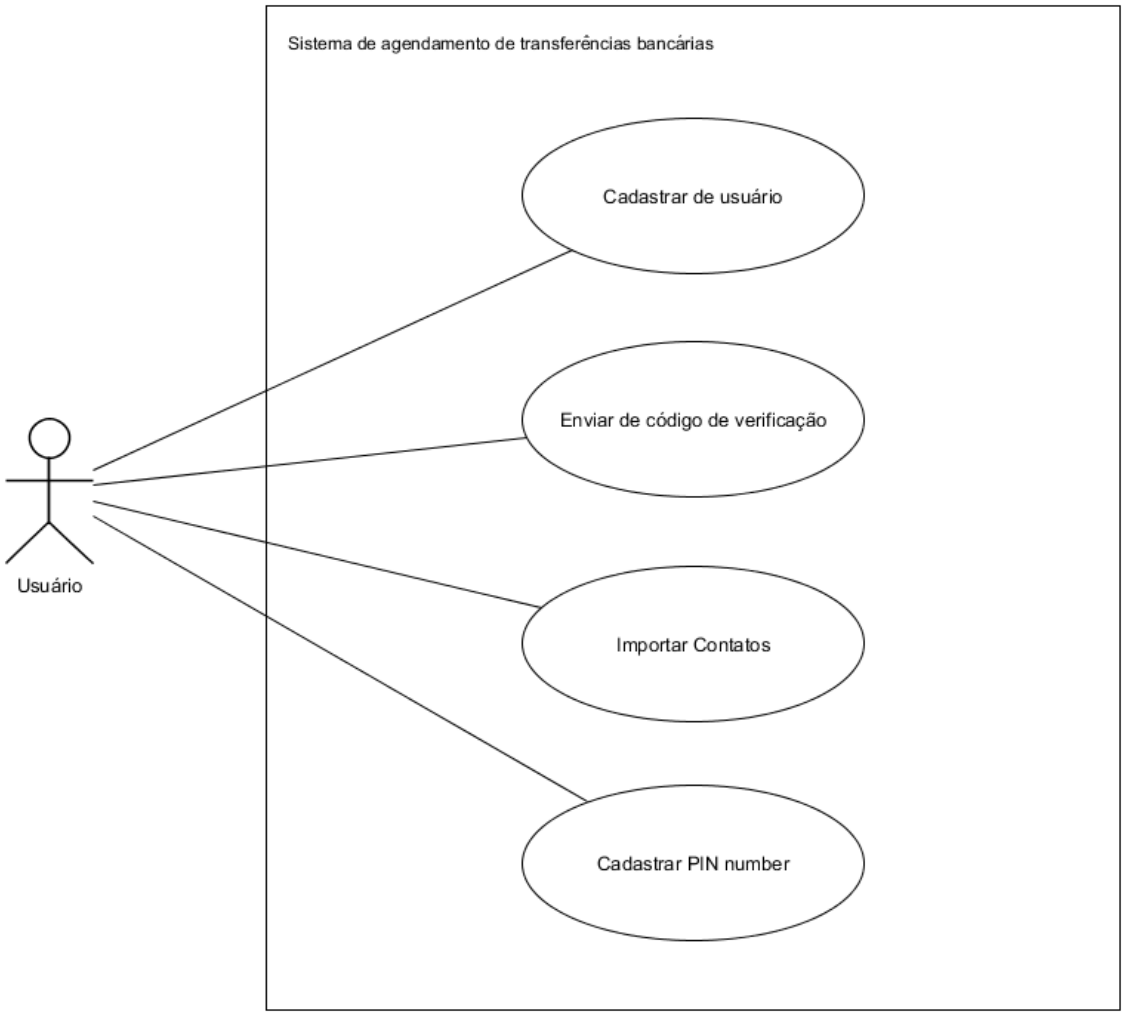
12.4.1 Fluxo Básico

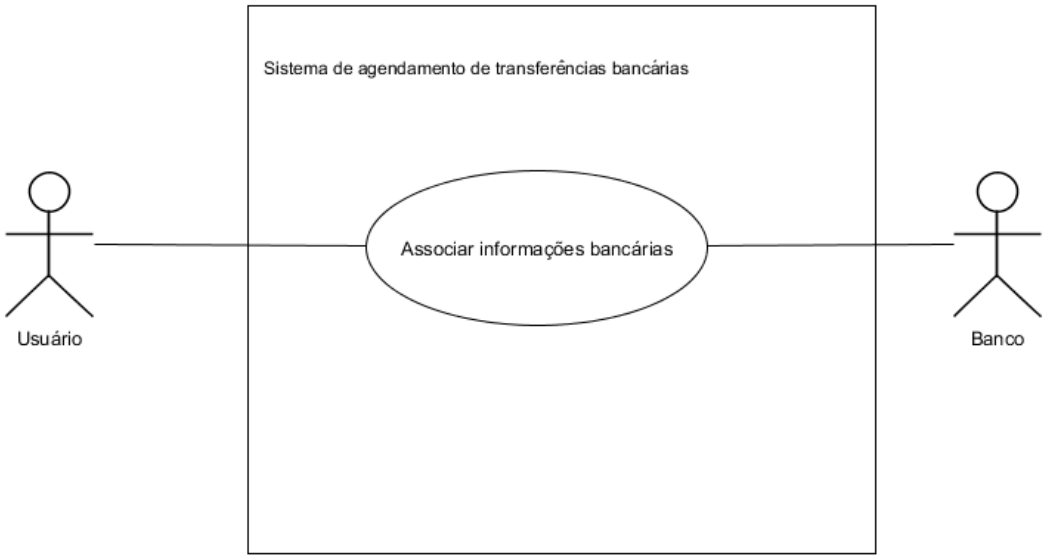
- Usuário faz uma requisição de visualização de histórico.
- Sistema envia histórico ao usuário.
- Usuário visualiza seu histórico.

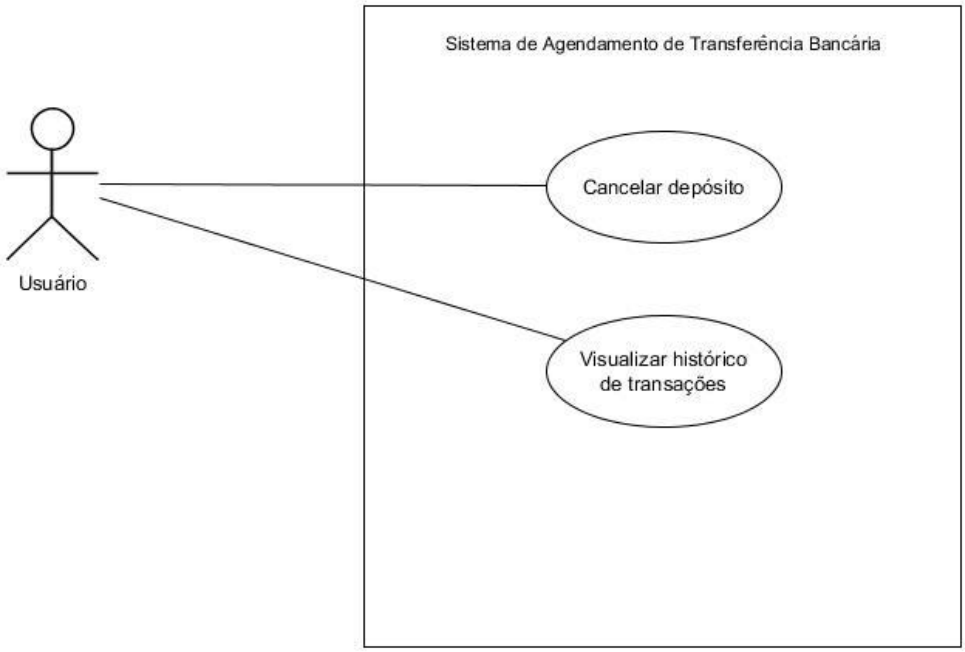
12.4.2 Fluxo de exceção

Não há.

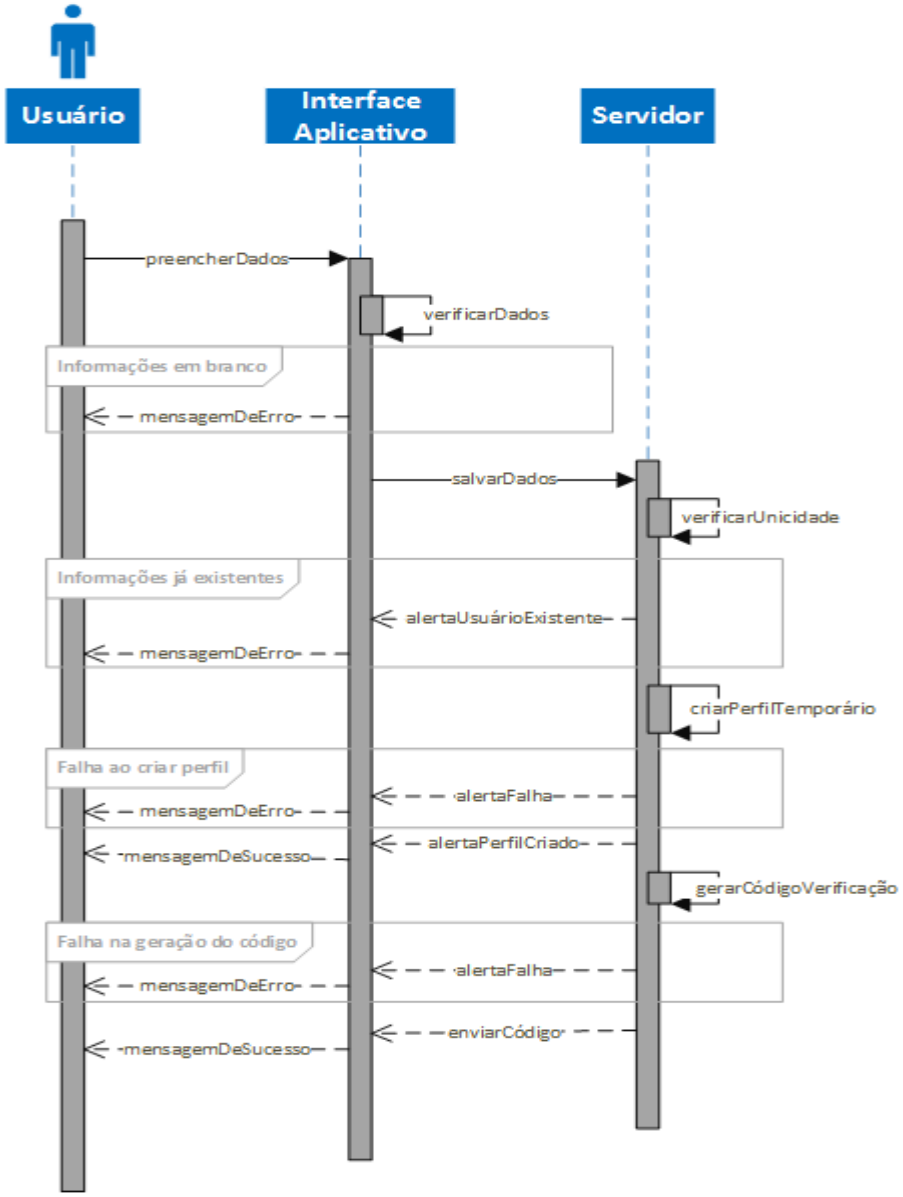
4.6 Diagramas de Caso de Uso

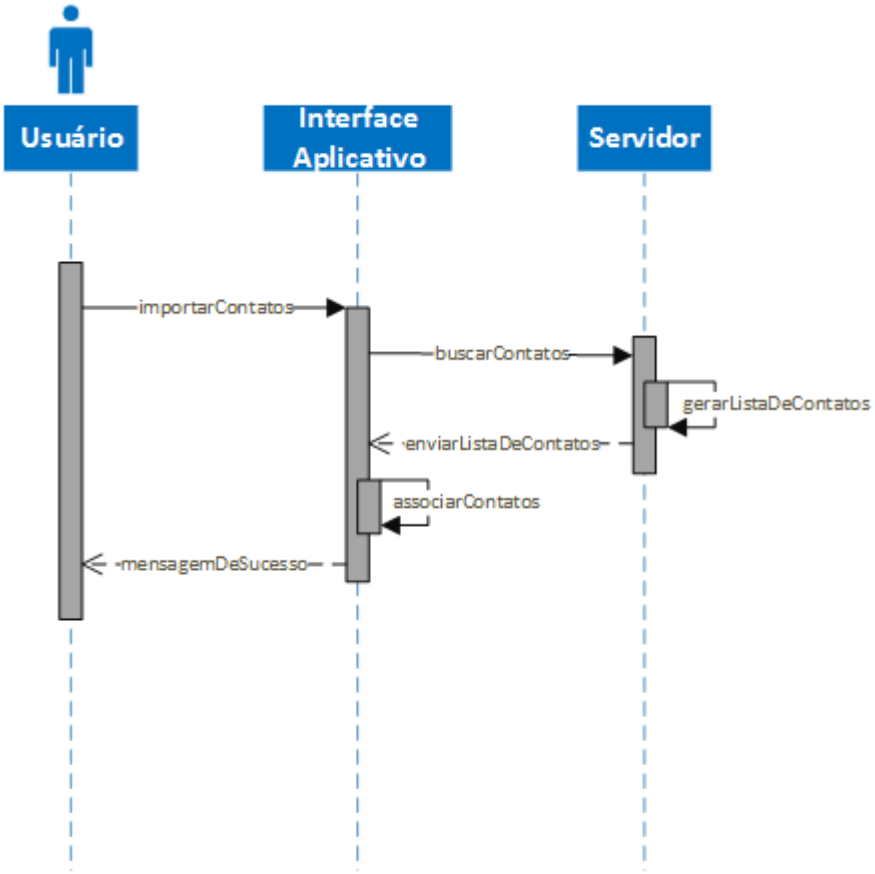


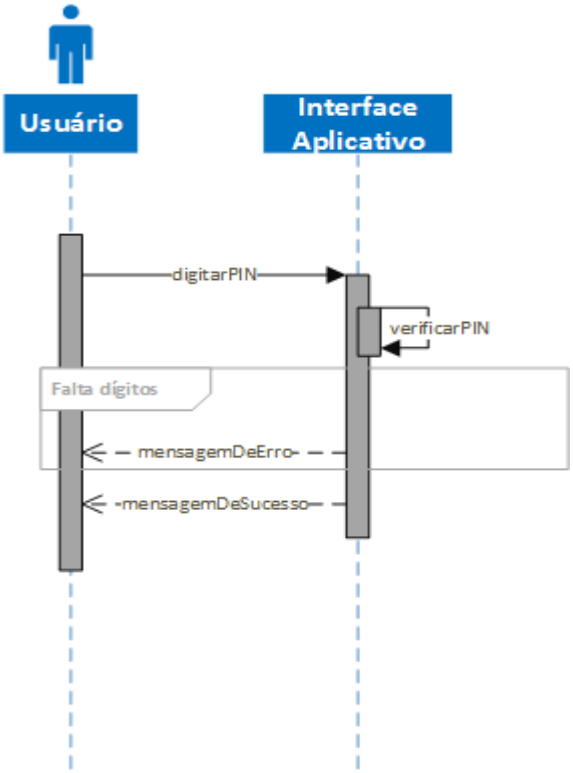

Sistema de agendamento de transferências bancárias
Associar informações bancárias
Usuário
Banco

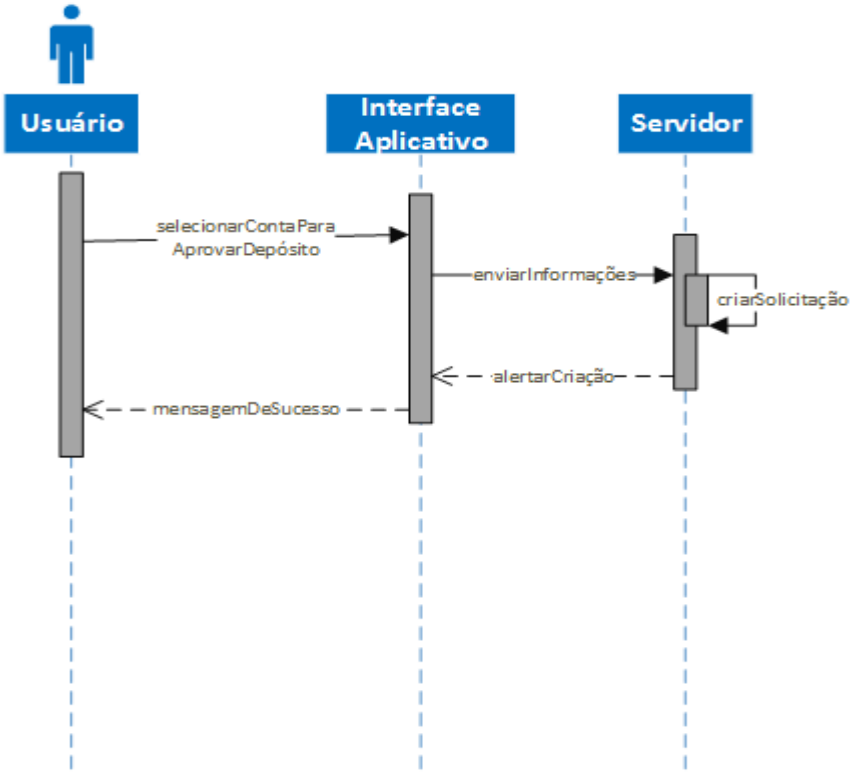

Sistema de Agendamento de Transferência Bancária
Cancelar depósito
Visualizar histórico
de transações
Usuário

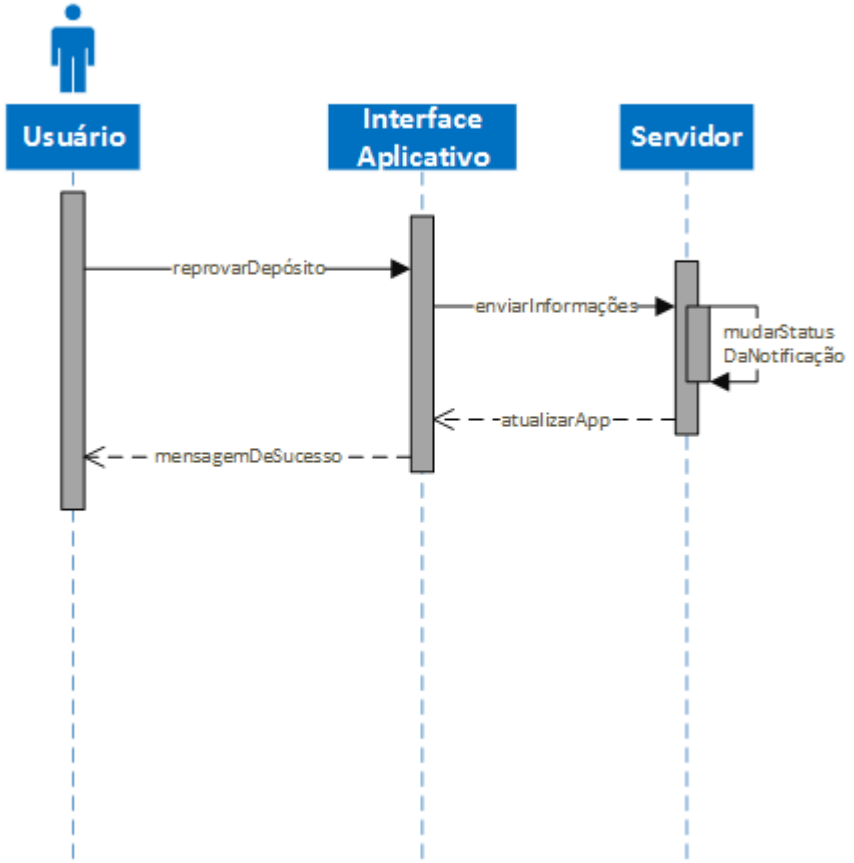
4.7 Diagramas de sequência

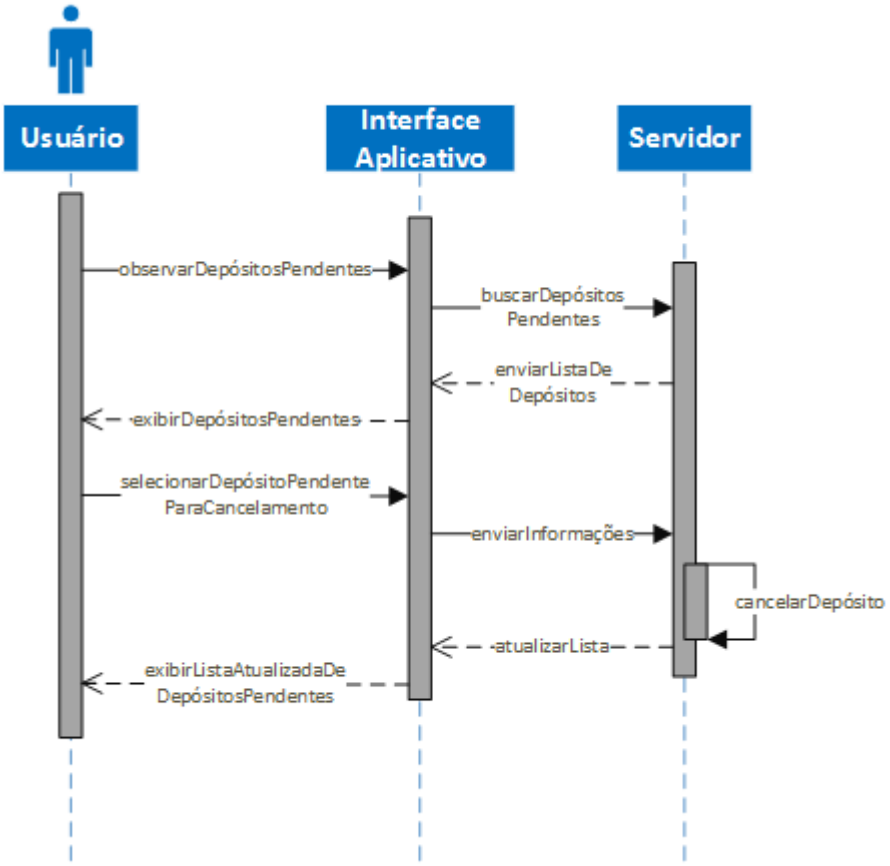



Usuário
Interface Aplicativo
Servidor
importarContatos
buscarContatos
gerarListaDeContatos
enviarListaDeContatos
associarContatos
mensagemDeSucesso

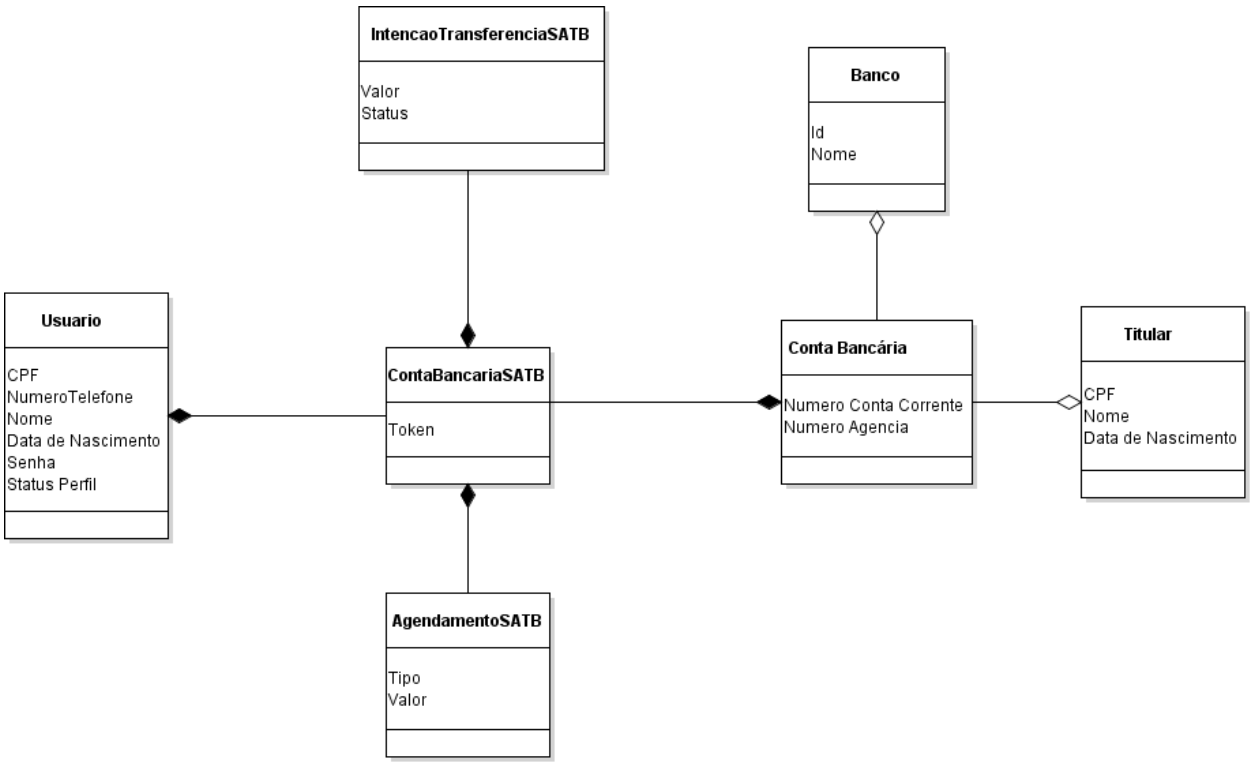

Usuário
Interface Aplicativo
digitarPIN
verificarPIN
Falta dígitos
mensagemDeErro
mensagemDeSucesso


Usuário
Interface Aplicativo
Servidor
selecionarContaPara AprovarDepósito
enviarInformações
criarSolicitação
alertarCriação
mensagemDeSucesso


Usuário
Interface Aplicativo
Servidor
reprovarDepósito
enviarInformações
mudarStatus DaNotificação
atualizarApp
mensagemDeSucesso

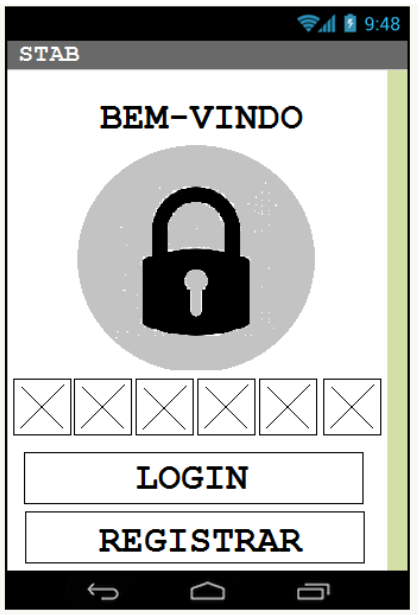

Usuário
Interface Aplicativo
Servidor
observarDepósitosPendentes
buscarDepósitos Pendentes
enviarListaDe Depósitos
exibirDepósitosPendentes
selecionarDepósitoPendente ParaCancelamento
enviarInformações
cancelarDepósito
atualizarLista
exibirListaAtualizadaDe DepósitosPendentes

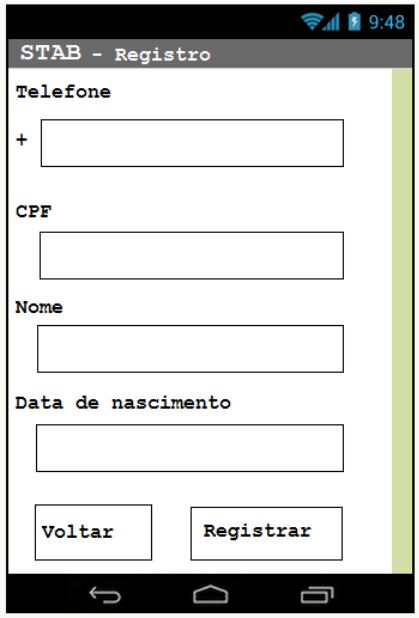
4.8 Diagrama de classes



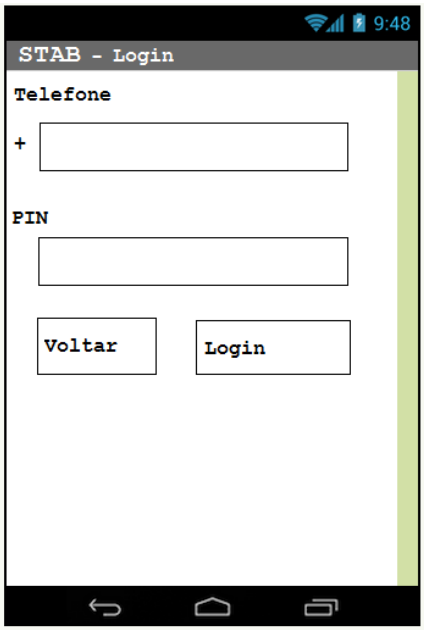
4.9 Protótipo

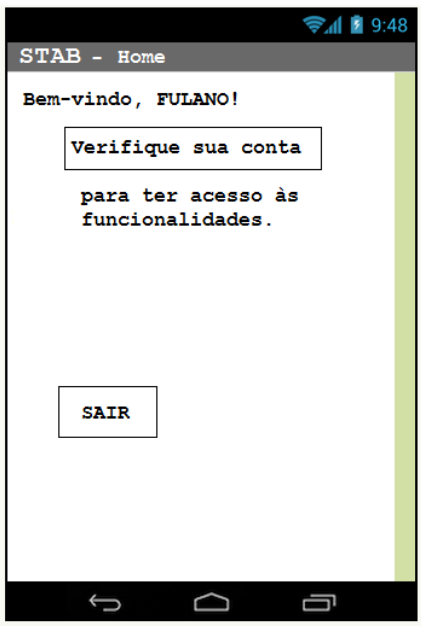
A navegação no protótipo é simples. Cada tela é identificada com um número após o título da tela correspondente (SATB - <Nome da tela>). Cada botão leva com um número correspondente leva às telas indicadas pelos números presentes no botão. Quando houver mais de um número, significa que ao clicar no botão pode acontecer um fluxo de exceção.

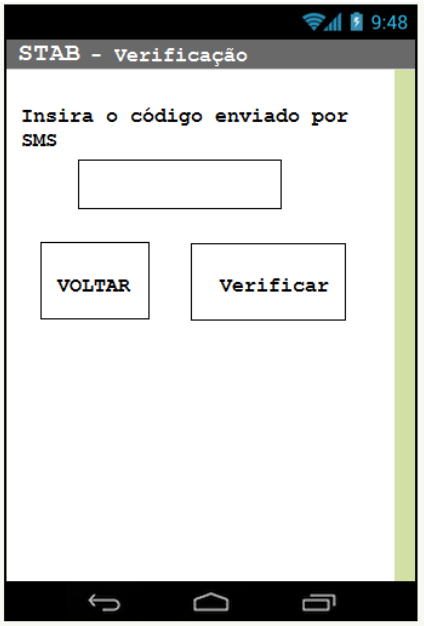

9:48
STAB
BEM-VINDO
LOGIN
REGISTRAR

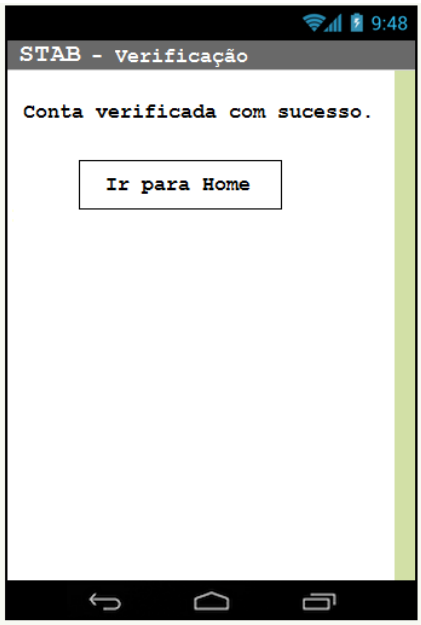

9:48
STAB - Registro
Telefone
+
CPF
Nome
Data de nascimento
Voltar
Registrar

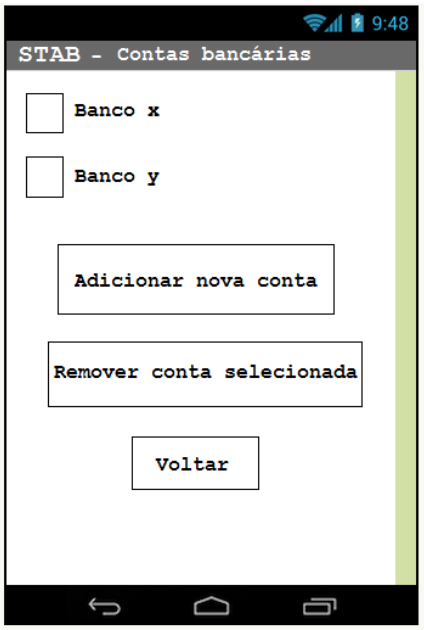

9:48
STAB
BEM-VINDO
Transações
Contas
Histórico
Contatos

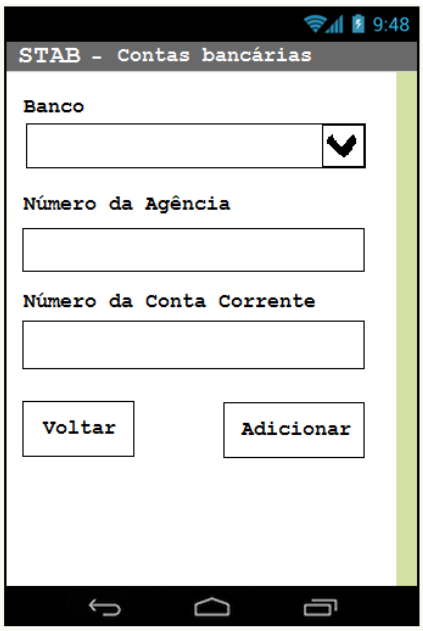

9:48
STAB - Login
Telefone
+
PIN
Voltar
Login

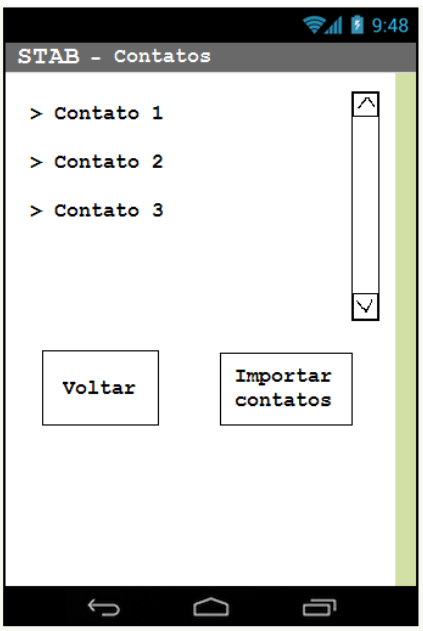

9:48
STAB - Home
Bem-vindo, FULANO!
Verifique sua conta
para ter acesso às funcionalidades.
SAIR

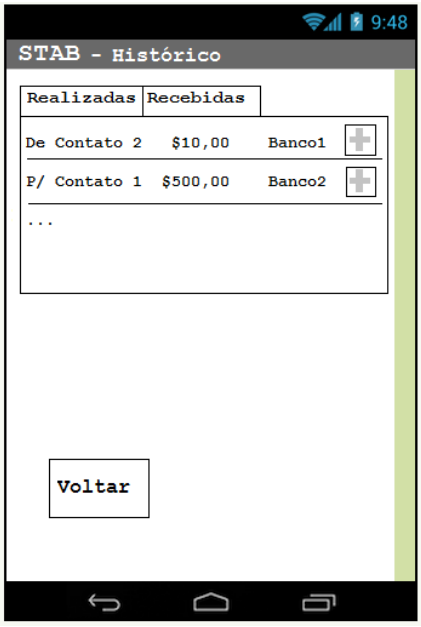

9:48
STAB - Verificação
Insira o código enviado por SMS
VOLTAR
Verificar

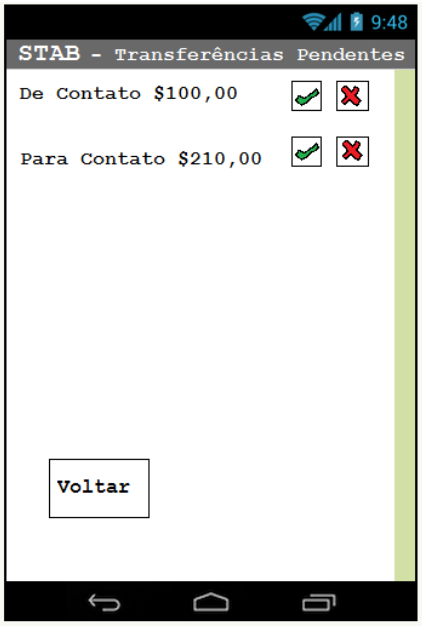

9:48
STAB - Verificação
Conta verificada com sucesso.
Ir para Home

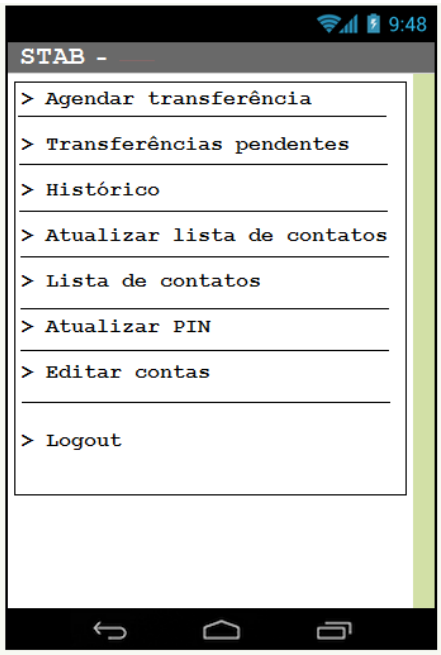

9:48
STAB - Contas bancárias
Banco x
Banco y
Adicionar nova conta
Remover conta selecionada
Voltar

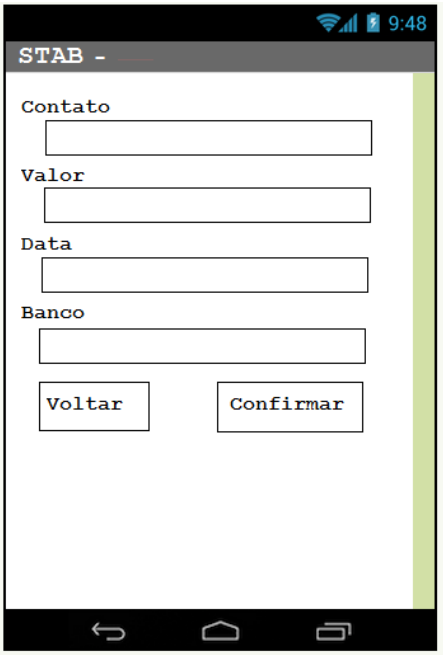

9:48
STAB - Contas bancárias
Banco
Número da Agência
Número da Conta Corrente
Voltar
Adicionar


9:48
STAB - Contatos
> Contato 1
> Contato 2
> Contato 3
Voltar
Importar contatos


9:48
STAB - Histórico
Realizadas
Recebidas
De Contato 2 $10,00 Banco1
P/ Contato 1 $500,00 Banco2
...
Voltar


9:48
STAB - Transferências Pendentes
De Contato $100,00
Para Contato $210,00
Voltar


9:48
STAB -
> Agendar transferência
> Transferências pendentes
> Histórico
> Atualizar lista de contatos
> Lista de contatos
> Atualizar PIN
> Editar contas
> Logout


9:48
STAB -
Contato
Valor
Data
Banco
Voltar
Confirmar

5 Considerações Finais

O projeto do Sistema de Agendamento de Transferências Bancárias desenvolveu um aplicativo na plataforma Android visando garantir acessibilidade e segurança ao facilitar o processo de transferências bancárias e evitar a troca de informações pouco utilizadas, mas sigilosas como a identificação da conta bancária.

Considera-se ter alcançado parcialmente os objetivos traçados, com exceção dos requisitos não funcionais do projeto no que diz respeito ao consumo de banda e energia, sendo menos robusto do que o planejado inicialmente.

Surpreendidos pela complexidade além da prevista e dificuldades de implementação pelo fato de utilizarmos tecnologias até então novas, o grupo optou por recorrer a simplificações para o término do projeto.

Tarefas foram realizadas conforme o planejado. Entretanto, o planejamento ocorreu em momentos próximos aos prazos de entrega de modo que alguns componentes não tenham ficado tão eficientes como o planejado. Dessa maneira, tomamos como aprendizado que uma melhor gerência nos prazos e no tempo disponível seria útil e teria grande impacto na obtenção de maior qualidade.

De todo modo, atividades foram bem divididas e gerenciadas no momento em que eram colocadas em prática. Cabe ressaltar o trabalho de forma harmoniosa do grupo e resolução de conflito de opiniões com maturidade.

Além do que foi apresentado nos aspectos conceituais em termos de criptografia e desenvolvimento de aplicativo celular, o grupo adquiriu um grande aprendizado a respeito da gerência de um projeto ao exercê-la na fase final.

A perspectiva para o sistema é a de continuidade para que possa vir a facilitar e garantir segurança em uma prática comum, integrando ainda mais a tecnologia na rotina do brasileiro.

6 Bibliografia

[1] A. Kahate,” Cryptography and Network Security”, TMH-2003.

[2] P. Zimmermann,“Introduction to Cryptography”, 2000.

[3] N. Jansma and B. Arrendondo,”Performance Comparison of Elliptic Curve and RSA Digital Signatures”,2004.

[4]The Study Material
http://www.thestudymaterial.com/presentationseminar/electronics-presentation/248-ellip.html?start=3

[5] V.B. Kute et al,”A software comparison of RSA & ECC”, IJCSA Vol 2, No.1, April/May 2009.

[6] S. Vanstone ,” Handbook of Applied Cryptography”, CRC Press, 1996.

[7] W. Stallings,” Cryptography and Network Security”, 2006.

[8] Wikipedia.orghttp://en.wikipedia.org/wiki/Cryptography

[9] Wikipedia.orghttp://en.wikipedia.org/wiki/Elliptic_curve

[10] Wikipedia.org- http://en.wikipedia.org/wiki/RSA

[11] R. Zuccherato,”Elliptic Curve Cryptography Support in Entrust”, 2000.

[12] RSA Laboratories TR201,”High-Speed RSA Implementation”,Technical report, November 1994.

[13] B.A. Forouuzan,”Data Communication and Networking”, (4/e, Tata McGraw-Hill).

[14] W.Stallings,”Network Security Essentials, Applications and Standards”, (2/e , Pearson Education).

[15] R.L. Rivest et al,”A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”, 1983.

[16] B. Kaliski,”The Mathematics of the RSA Public-Key Cryptosystem”, 1989.

[17] BROWN, Daniel R L. Standard for Efficient Cryptography 1: Elliptic Curve Cryptography. Certicom Corp, 2009