FACULDADE INTEGRADO DE CAMPO MOURÃO - PARANÁ



TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

JAMERSON CHIMINACIO PEREIRA

SISTEMA OPERACIONAL PARA CAIXA ELETRÔNICO

CAMPO MOURÃO

2016

FACULDADE INTEGRADO DE CAMPO MOURÃO – PARANÁ

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

JAMERSON CHIMINACIO PEREIRA

SISTEMA OPERACIONAL PARA CAIXA ELETRÔNICO

TRABALHO INTEGRADOR

TRABALHO APRESENTADO AO CURSO DE TECNOLOGIA EM ANÁLISE DE DESENVOLVIMENTO DE SISTEMAS DA FACULDADE INTEGRADO DE CAMPO MOURÃO - PR COMO REQUISITO PARCIAL PARA AVALIAÇAO NO PROJETO INTEGRADOR.

PROFESSOR – ROSELY SCHEFFER

CAMPO MOURÃO

2016.

**SISTEMA OPERACIONAL PARA CAIXA ELETRÔNICO**

**RESUMO: O presente artigo busca o desenvolvimento de um sistema de caixa eletrônico e a elaboração da documentação exigida para a criação do mesmo. As informações foram obtidas por meio de pesquisa em caixas eletrônicos regionais, livros e busca online.**

**1 – INTRODUÇÃO**

Entendemos por caixa eletrônico como uma máquina capaz de realizar várias operações bancarias, que veio para facilitar e diminuir nosso tempo de espera por atendimento nas agências bancárias, este equipamento está em constante evolução seja ela no hardware e principalmente no software que gerencia todos os recursos garantindo segurança e performance nas operações. Seu sistema operacional é desenvolvido usando todos os recursos tecnológicos e de segurança disponíveis no mercado, inicialmente são levantados todos os requisitos de utilização, desses são elaborados vários diagramas e documentos para que ajude na compreensão e desenvolvimento, e, por fim, são efetuados todos os testes necessários e implantação nos terminais.

**2 – REVISÃO BIBLIOGRÁFICA**

**2.1 – LEVANTAMENTO DOS REQUISITOS**

Requisito é a característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir seus objetivos.

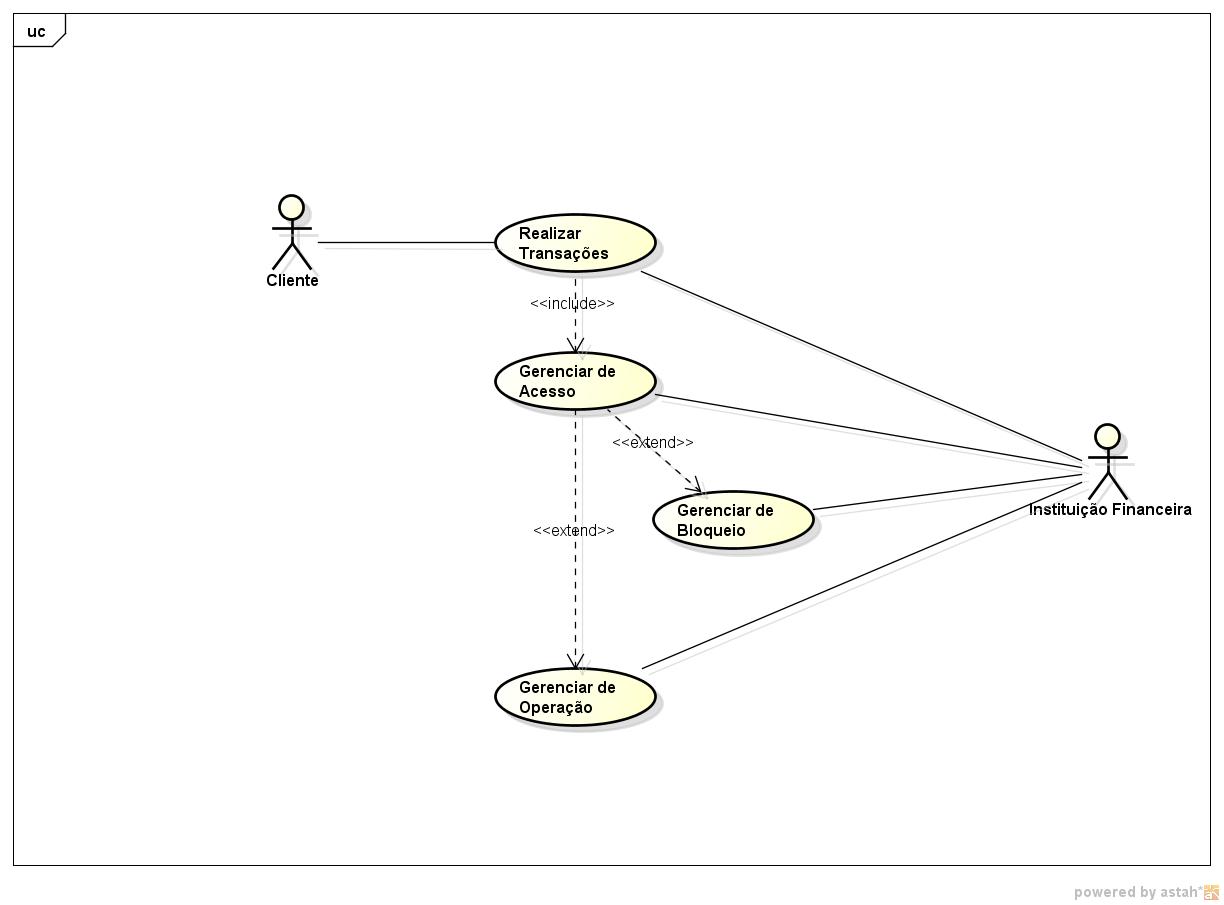
Segundo Pfleeger (2004), “Um requisito é uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir os seus objetivos.”

Na ocasião foi levando inicialmente os seguintes requisitos para o funcionamento de um caixa eletrônico em um banco.

O banco possui vários caixas eletrônicos que estão geograficamente distribuídos e conectados via rede a um servidor central, cada caixa eletrônico tem monitor, leitora de cartão, teclado e impressora, nos equipamentos o cliente pode retirar dinheiro da conta-corrente ou poupança, consultar saldo, extrato e efetuar transferência de dinheiro entre contas.

A transação é iniciada quando o cliente insere o cartão na leitora de cartões ou toca a tela, quando o cliente apenas toca a tela, é direcionada para a tela de operações sem cartão que terá apenas a opção de depósito em conta corrente ou poupança, já quando é inserido o cartão o sistema valida se o cartão está ativo e solicita a senha para o usuário, se digitada a senha três vezes errada o cartão é bloqueado, caso contrário é liberada a tela de operações que é composta por saldo, extrato, pagamento e transferência.

**2.2. – DIAGRAMA DE CASO DE USO COM DESCRIÇÃO.**

Segundo Booch (2005) “Um caso de uso é uma descrição de um conjunto de sequências e ações, que um sistema executa para produzir um resultado de valor observável por um ator. Este representa um conjunto de papéis que os usuários de casos de uso desempenham quando interagem com esses casos de uso. Um caso de uso descreve o que um sistema faz”.

|  |  |
| --- | --- |
| **Caso de uso** | Realizar transação |
| **Atores** | Cliente, Instituição Financeira |
| **Descrição** | Cliente insere o cartão no leitor para iniciar a transação |
| **Pré-condição** | Possuir o cartão e conta na agência bancária |
| **Pós-condição** | Cliente realizará a transação ou receberá notificação de bloqueio |
| **Fluxo de eventos:** | |
| -Cliente: Insere cartão no leitor  -Cliente: Insere senha  -Instituição Financeira: Automaticamente verificará se a senha é válida, ou se o cartão é válido | |

|  |  |
| --- | --- |
| **Caso de uso** | Gerenciar Acesso |
| **Atores** | Cliente, Instituição Financeira |
| **Descrição** | Cliente faz três tentavas de digitar senha, se o cartão foi perdido ou roubado o mesmo e confiscado |
| **Pré-condição** | Possuir o cartão, conta na agência bancária e senha |
| **Pós-condição** | Cliente realizará a transação ou receberá notificação de bloqueio |
| **Fluxo de eventos:** | |
| -Instituição Financeira: Verifica se a senha está correta  -Instituição Financeira: Verifica se o cartão está com alerta de perda ou furto  -Instituição Financeira: Caso a senha esteja correta e não haja nenhum alerta de perda ou furto libera o acesso para o cliente  -Instituição Financeira: Caso a senha esteja incorreta ou haja perda ou furto, bloqueia o cartão | |

|  |  |
| --- | --- |
| **Caso de uso** | Gerenciar Bloqueio |
| **Atores** | Instituição Financeira |
| **Descrição** | Caso senha esteja incorreta e foram realizadas três tentativas o cartão será bloqueado e confiscado, o mesmo acontece caso o cartão tenha sido furtado ou perdido |
| **Pré-condição** | Senha incorreta por mais de 3 tentativas ou cartão furtado ou perdido |
| **Pós-condição** | Bloqueio e confiscação do cartão |
| **Fluxo de eventos:** | |
| -Instituição Financeira: Bloqueia o cartão  -Instituição Financeira: Confisca o cartão | |

|  |  |
| --- | --- |
| **Caso de uso** | Gerenciar operações |
| **Atores** | Cliente, Instituição Financeira |
| **Descrição** | Cliente poderá sacar, consultar saldo ou transferir dinheiro para outra conta |
| **Pré-condição** | Cliente possuir senha correta e cartão válido |
| **Pós-condição** | Cliente sacar, consultar saldo ou transferir dinheiro |
| **Fluxo de eventos:** | |
| -Cliente: Pode checar saldo  -Cliente: Pode efetuar um saque  -Cliente: Pode transferir dinheiro para outra conta  - Instituição Financeira: Validar operação solicitada  - Instituição Financeira: Imprimir extrato para o cliente, caso “Consultar saldo” seja a operação desejada.  - Instituição Financeira: Entregar dinheiro para o cliente, caso “Sacar dinheiro” seja a operação solicitada, e haja saldo  - Instituição Financeira: Imprimir comprovante de saque  - Instituição Financeira: Realizar o envio do d Instituição Financeira dinheiro da conta do cliente para outra conta caso “Transferir dinheiro” seja a operação desejada  - Instituição Financeira: Imprimir comprovante de transferência | |

**2.2.1 – DIAGRAMA DE CLASSES**

Segundo Booch e Jacobson (2005) “Um diagrama de classes mostra um conjunto de classes, interfaces, colaborações e seus relacionamentos, permitindo a modelagem da visão estática do sistema. O diagrama de classes contém os itens classes, interfaces, relacionamentos de dependência, generalização e associação.”

-

endereco : int

-

numero : int

**Agencia**

-

tipo : int

limite : boolean

-

saldo : double

-

numero : int

-

**Conta**

numero : String

-

**Cartao**

numero : int

-

**Banco**

-

nome : String

**Operadora**

gerar(operacao : Operacao) : void

+

**Comprovante**

-

CPF : int

nome : String

-

**Cliente**

efetuaOperacao(de : Conta, para : Conta, valor : double) : void

+

**Operacao**

ContaCliente

**ContaCliente**

1..\*

1

1

1..\*

1

1..\*

numero : int

-

**CaixaEletronico**

1

1..\*

**Transferencia**

**Saque**

**Deposito**

0..1

0..\*

**Termica**

**Email**

**SMS**

1

1

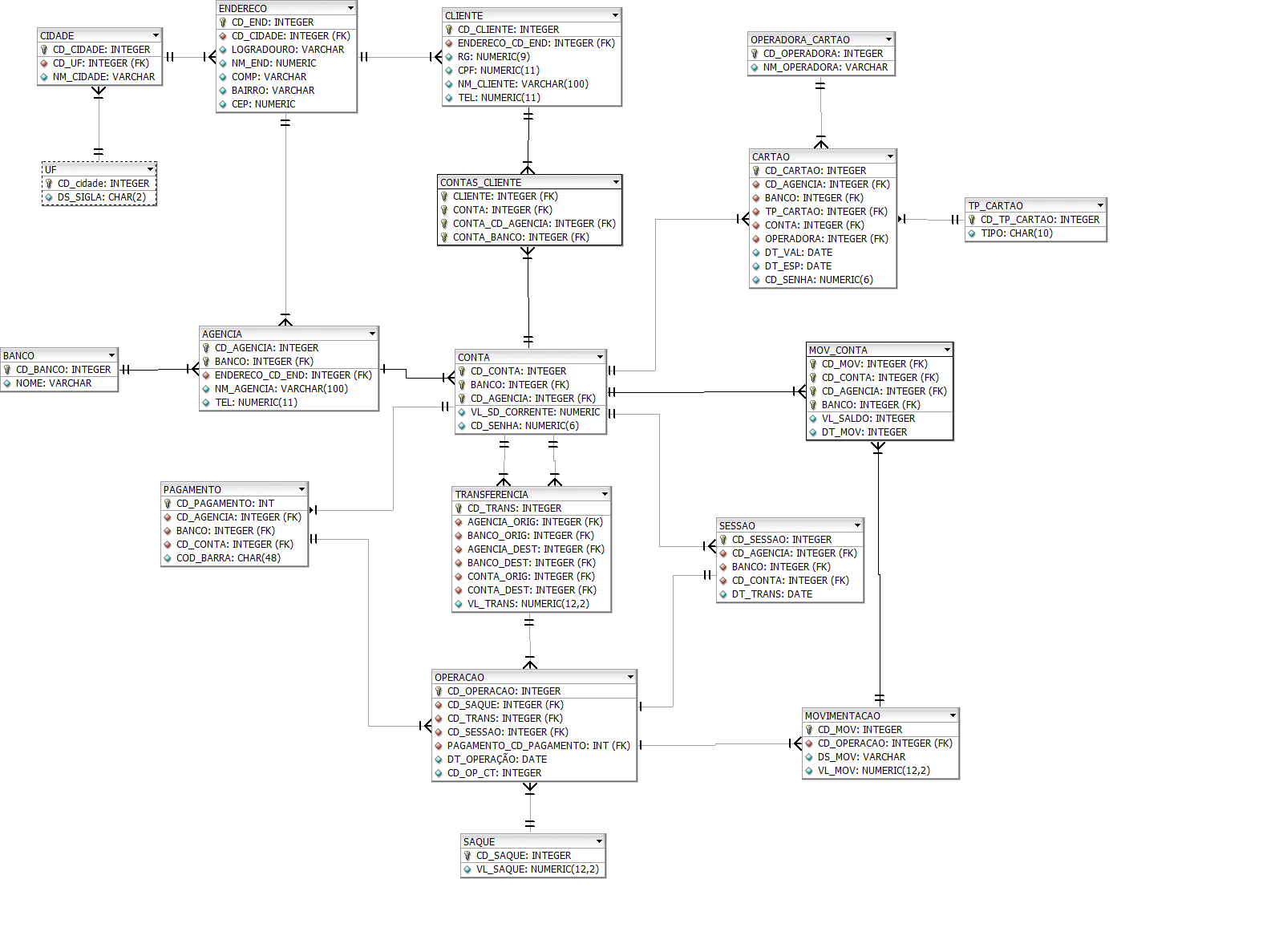
1

0..\*

**2.2.2 – MER – MODELO ENTIDADE RELACIONAL**

Modelo é um grupo de objetos que representa o mundo real com relacionamentos, entidades e atributos, geralmente e usado para implementar o banco de dados.

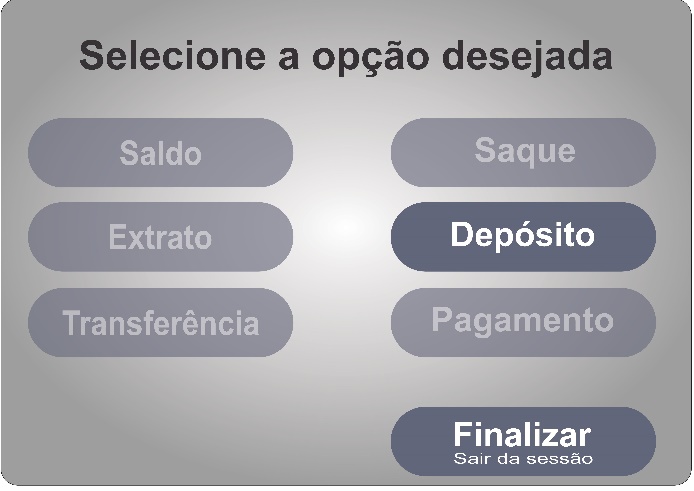
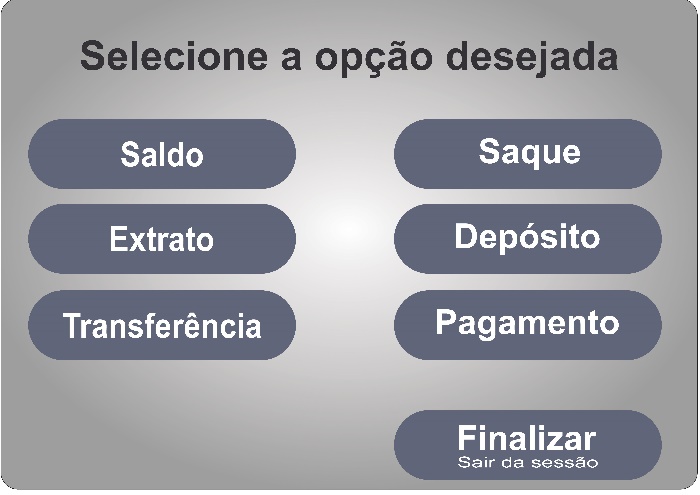
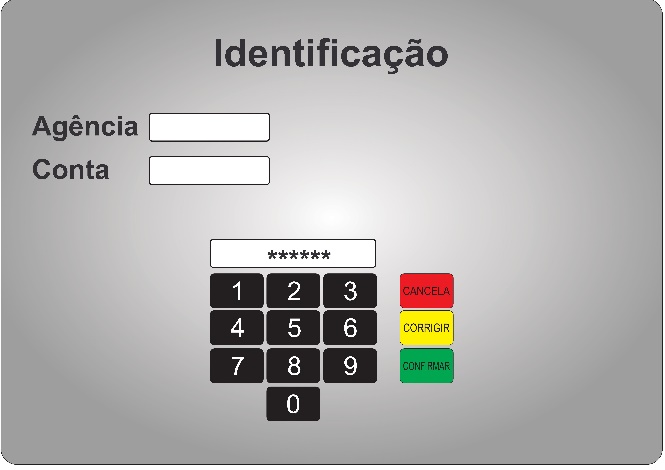
Chen (1976) afirma que o modelo Entidade-Relacionamento adota uma visão mais natural do mundo real, ou seja, dos negócios analisados. Portanto este é o modelo mais recomendado para projetar bancos de dados.

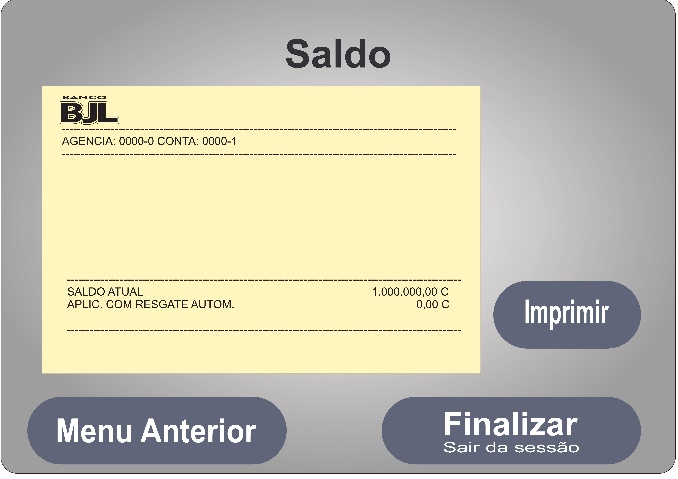
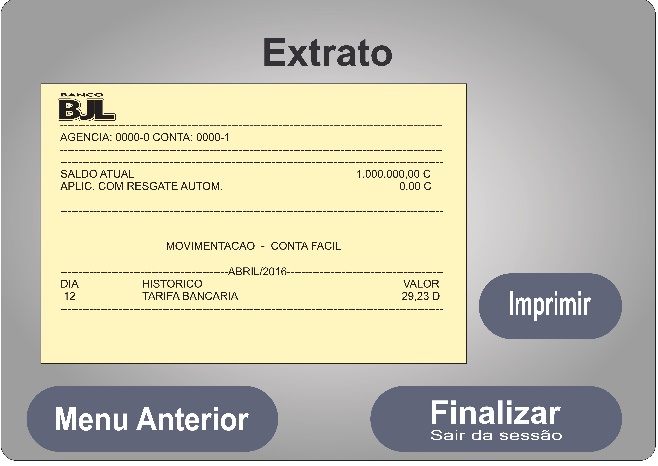
**2.2.3 – PROTOTIPAGEM**

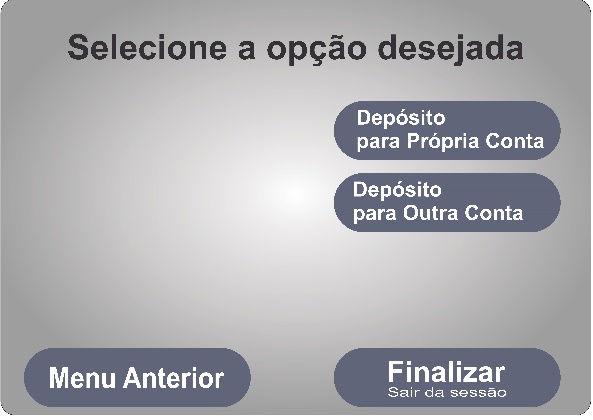
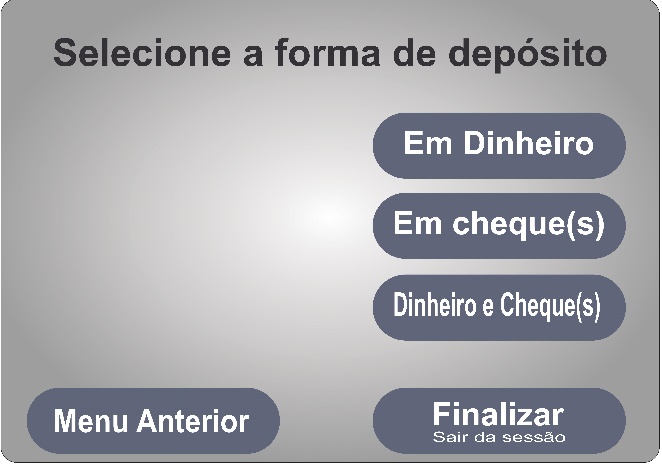
Prototipagem tem como objetivo de mostrar para o usuário como deverá ficar inicialmente as telas de seu sistema, mostrando toda a estrutura operacional do software, tem como vantagem redução de custo e tempo, facilita na fase de levantamento de requisitos, considerada um dos melhores instrumentos de comunicação como o usuário.

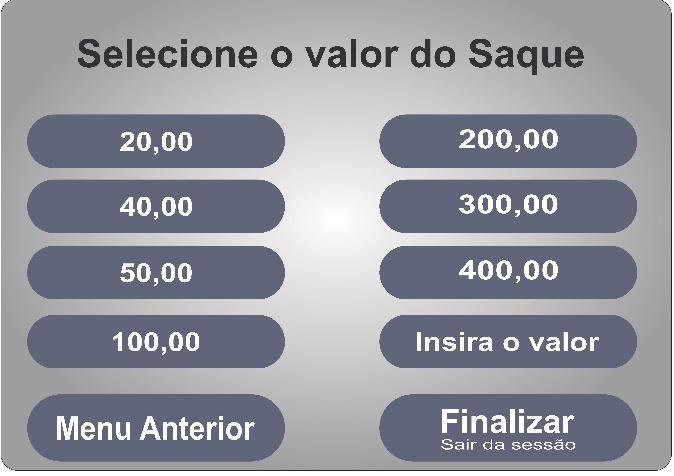
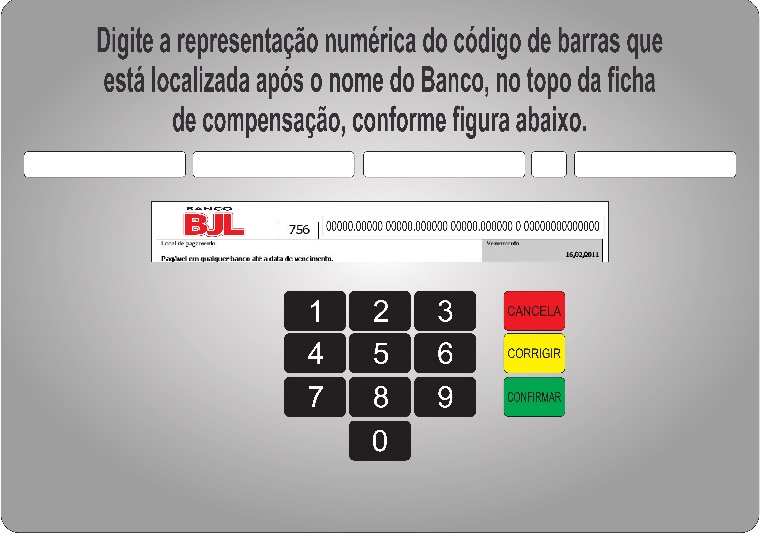
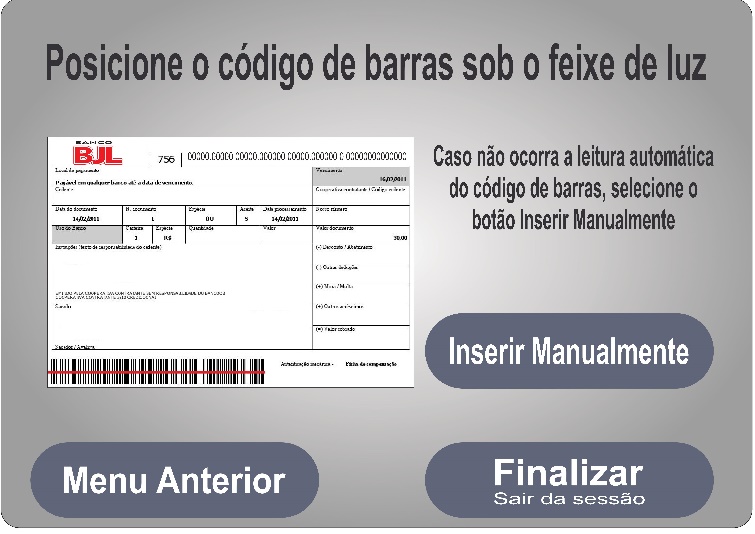
Segundo Sommerville, Sawyer 1997 “Protótipo é um sistema de demonstração que se apresenta aos utilizadores e Stakeholders (Os Stakeholders são as pessoas ou organizações que são de alguma forma afetadas pelo sistema e/ou que tem diretamente ou indiretamente influência nos requisitos do sistema) de forma a validar os requisitos conhecidos ou obtê-los quando os requisitos conhecidos são vagos ou indefinidos. Um protótipo pode ser usado como meio de comunicação entre os diversos membros da equipe de desenvolvimento ou mesmo como meio de nós mesmos testarmos as nossas ideias.”

Segundo Kotonya, Sommerville 1998 “A prototipagem “Throw-away” consiste no desenvolvimento de um protótipo com o objetivo de aumentar a qualidade do documento de requisitos. O desenvolvimento do protótipo tem por base os requisitos que não estão bem definidos. Os requisitos bem definidos poderão nunca ser implementados no protótipo.”









**3 – CONCLUSÃO**

Ao final deste trabalho podemos concluir que para a construção de um sistema operacional que será usado no caixa eletrônico não é tão simplório como sua interface sugere. Devido a diversos métodos usados na tentativa de corromper o sistema, seu desenvolvimento tem que seguir métodos e técnicas disponíveis no mercado para garantir sua confiabilidade e disponibilidade ao usuários, o sistema fornece vários dispositivos de segurança, mas o usuário tem que seguir recomendações de segurança fornecidas pela instituição financeira para evitar transtornos.

Do ponto de vista acadêmico fica totalmente visível a necessidade de criar a documentação do sistema, pois ela torna a compreensão mais completa e deixa mais claros pontos que por vezes não foram abstraídos em um primeiro levantamento de requisitos. A documentação consome tempo, planejamento e pesquisa, mas é completamente viável por tornar o sistema mais completo e confiável.

**REFERÊNCIAS**

PFLEEGER, S. L. **Engenharia de software:** teoria e pratica. 2. ed. São Paulo, SP: Prentice Hall, 2004.

BOOCH, G., RUMBAUGH, J., JACOBSON, I. **UML:** guia do usuário. Rio de Janeiro: Elsevier, 2005.

SOMMERVILLE, Ian. **Engenharia de software**, 8 ed. Pearson Education, 2007.

CHEN P, **O modelo entidade relacionamento**: em direção a uma visão unificada de dados. ACM Transactions em Banco de Dados de Sistemas de 1976 .

PRESSMAN, R. S. **Engenharia de software**. 7. ed. Porto Alegre: McGraw-Hill, 2011.

PRINCE, J. **Oracle Database 11g SQL: domine SQL e PL/SQL no banco de dados Oracle**. Porto Alegre: Bookman, 2009.

MICHAEL, M. **Oracle Database 11g SQL: PL/SQL Programação**. Porto Alegre: Alta Books, 2009.

DEITEL, HARVEY M. **Java como programar**, 8ª. Ed / Harvey M. Deitel & Paul J. Deitel: Tradução Edson Furmankiewicz: revisão técnica Fabio Luis Picelli Lucchini. São Paulo: Person Prentice Hall, 2010.