

UNIVERSIDADE ESTADUAL DE GOIÁS
UNIDADE UNIVERSITÁRIA DE ITABERAÍ
GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

Joabe Perillo Moreira Santos

GERENCIADOR FINANCEIRO MÓVEL – GFM

Itaberaí

12/2011

Joabe Perillo Moreira Santos

GERENCIADOR FINANCEIRO MÓVEL – GFM

Trabalho apresentado ao curso de Graduação em Sistemas de Informação, da Universidade Estadual de Goiás, como pré-requisito para obtenção do grau de Bacharel em Sistemas de Informação.

orientação: Prof^ª. Esp.. Rogério Alexandre Leite Pereira

Itaberaí

2011

Joabe Perillo Moreira Santos

GERENCIADOR FINANCEIRO MÓVEL – GFM

Aprovado em ___/___/_____

Banca examinadora:

Prof.Esp Rogério Alexandre Leite Pereira

Prof.Esp. Odney Candido Jacomini

A todos aqueles que me apoiara e estiveram comigo neste desafio, em especial a minha mãe ELENI MOREIRA BRANDÃO, que sempre me incentivou a capacitação.

RESUMO

Observando a velocidade de acontecimentos atuais, o dinamismo do mundo dos negócios, faz-se necessário que cada pessoa tenha seus ativos e passivos em plena organização e controle. Tendo em vista que atualmente existem muitos gerenciadores financeiros – na maioria das vezes complexos de serem utilizados – para desktops, faz-se necessário, para aqueles que não têm tempo para (ou até recursos) estar em frente a um computador, a existência de um software que esteja sempre à mão, seja simples de utilização e não tenha custo elevado tanto de software como de hardware. O fato de existir um sistema para controle de finanças no bolso(a), torna o domínio do “fluxo de caixa ” muito mais fácil, sendo que a nova entrada de ganho ou gasto pode ser inserida no momento do negócio, sem despesas

Palavras-Chaves: Gerenciador Financeiro, Android, UML, Orientação a Objetos, Ativos, Passivos, Matemática Financeira, Juros.

ABSTRACT

Observing the speed of current events, the dynamism of the business world, it is necessary that each person has their assets and liabilities in full organization and control. Since there are now many financial managers – most often complex to use – for desktops, it is necessary for those who do not have time to (or even resources) to be in front of a computer, the existence of a software that is always at hand, is simple to use and not costly both software and hardware. The fact that there is a system for control of finances in your pocket makes the domain of “cash flow” much easier, and the new input gain or expense may be inserted at the time of the business free of charge.

Key Words: Financial Manager, Android, UML, Object Orientation, Assets, Liabilities, Financial Mathematics, Interest.

LISTA DE ABREVIATURAS E SIGLAS

GFM – Gerenciador Financeiro Móvel.

OHA – Open Handset Alliance.

ADT – Android Development Toolkit.

UML – Unified Modeling Language

OO – Orientação à Objetos

MER – Modelo de Entidade e Relacionamento

SUMÁRIO

1 Introdução	10
1.1 Objetivos	10
2. Apresentação	11
2.1 Gerenciador Financeiro Móvel	12
2.2 Dificuldades	12
3 Fundamentos Teóricos	13
3.1 Orientação a Objetos	13
3.2 UML	13
3.2.1 Diagramas da UML	14
3.3 Modelo de Entidade e Relacionamento	18
3.3.1 Entidades	18
3.3.2 Atributos	18
3.3.3 Relacionamentos	19
3.3.4 Cardinalidade	19
4 Estudo de Caso: Gerenciador Financeiro Móvel	21
4.1 Identificação dos Requisitos	21
4.2 Descrição dos Processos	21
4.3 Requisitos Funcionais	22
4.4 Requisitos Não Funcionais	23
4.5. Diagramas de Caso de Uso	23
4.6. Descrição dos casos de uso	24
Cadastrar Proventos	24
Cadastrar Despesas	24
Cadastrar Investimentos	25
Cadastrar Objetivos	25
Alterar Proventos	26
Alterar Despesas	26
Alterar Investimentos	26
Alterar Objetivos	27
Excluir Proventos	27
Excluir Despesas	28
Excluir Investimentos	28

Excluir Objetivos	28
Contabilizar	29
4.7 Glossário de Mensagens	29
5 considerações finais	32
6 Trabalhos futuros	33
7 Referência Bibliográfica	34
Apêndice A – Documentação do Sistema	35
APÊNDICE A.1-MER	36
APÊNDICE A.2 -DER	37
APÊNDICE A.3- DIAGRAMA DE ATIVIDADE – ATUALIZAR PROVENTOS	38
APÊNDICE A.4- DIAGRAMA DE ATIVIDADE – ATUALIZAR DESPESAS	39
APÊNDICE A.5 -DIAGRAMA DE ATIVIDADE – ATUALIZAR INVESTIMENTOS	40
APÊNDICE A.6-DIAGRAMA DE ATIVIDADE – ATUALIZAR OBJETIVOS	41
APÊNDICE A.7-DIAGRAMA DE ATIVIDADE – CONTABILIZAR	42
APÊNDICE A.8-DIAGRAMA DE CLASSES	43
Apêndice B – Protótipos do Software	44
Apêndice B.1-Protótipo Tela Principal	45
Apêndice B.2-Protótipo Incluir Proventos	46
Apêndice B.3-Protótipo Incluir Despesas	47
Apêndice B.4-Protótipo Incluir Investimentos	48
Apêndice B.5-Protótipo Incluir Objetivos	49
Apêndice B.6-Protótipo Origem	50
Apêndice B.7-Alterar Proventos	51
Apêndice B.8-Alterar Despesas	52
Apêndice B.9-Alterar Investimentos	53
Apêndice B.10-Alterar Objetivos	54
Apêndice B.11-Alterar Origem	55

1 INTRODUÇÃO

Na “correria” do dia-a-dia quase não existe tempo nem para refeições, lazer ou qualquer outra atividade que não seja trabalhar ou estudar, então como ter sob controle algo que para a maioria das pessoas é complicado, que envolve matemática financeira e contabilidade?

A resposta está no novo sistema GFM, que vem para atender as necessidades daqueles que não aceitam perder um único centavo de vista, ou seja, tem ou desejam ter o perfeito domínio de suas transações financeiras, sabem onde estão colocando seu dinheiro, bem como de onde ele está vindo, são essencialmente perfeccionistas e já estão cansadas de gastar tempo com o papel, a caneta e a calculadora, tentando se lembrar onde está a diferença entre o que está na “carteira” e o que deveria estar segundo as próprias contas.

O GFM chega para acabar com toda essa dor de cabeça, com interface simples, intuitiva, não exige grandes conhecimentos na vasta ciência dos números, ele mesmo faz todas as contas para que o usuário possua pleno conhecimento do que está acontecendo no próprio bolso e ainda o auxilia na tomada de decisões.

Para a construção e documentação deste projeto foram usados Orientação a Objetos, UML, e como ferramenta de programação foi escolhido o ambiente Eclipse, juntamente com o plugin Android Development Toolkit.

1.1 Objetivos

Este projeto tem como objetivo criar uma solução prática para o controle de finanças pessoais, de qualquer indivíduo que possua um aparelho com o Sistema Operacional Android, então eis aqui a solução em contabilidade pessoal: GFM.

2. APRESENTAÇÃO



Figura 1 - Logotipo Android

Inicialmente vamos entender de onde veio o Android, segundo Alves:

O Android é uma plataforma desenvolvida pela Google voltada para dispositivos móveis. Em 5 de novembro de 2007, a empresa tornou pública a primeira plataforma Open Source de desenvolvimento para dispositivos móveis baseada na plataforma Java com sistema operacional Linux, na qual foi chamada de Android. Essa plataforma é mantida pela OHA (Open Handset Alliance), um grupo formado por mais de 40 empresas as quais se uniram para inovar e acelerar o desenvolvimento de aplicações, serviços, trazendo aos consumidores uma experiência mais rica em termos de recursos, menos dispendiosa em termos financeiros para o mercado móvel. Pode-se dizer que a plataforma Android é a primeira plataforma móvel completa, aberta e livre (2011, p. 6).

Como está escrito na citação acima, a Google não é a única empresa responsável pela criação do Android, juntamente com ela estão dezenas de empresas, dentre as principais estão HTC, Intel, Samsung, Motorola, etc. Juntas elas formam a Open Handset Alliance.

O objetivo deste grande time é buscar um padrão de Sistema Operacional o qual atenda as necessidades dos dias atuais e como uma das linguagens mais utilizadas de programação é o Java, a OHA possui o mundo trabalhando em seu produto.

A plataforma Android possui diversos atrativos, dentre os principais estão:

- Desenvolvimento de softwares usando a linguagem Java;
- SQLite (gratuito) como banco de dados nativo;
- Todas as ferramentas necessárias são gratuitas;

- Código fonte livre;
- Flexibilidade.

2.1 Gerenciador Financeiro Móvel

O GFM tem como principal objetivo apresentar ao usuário uma interface simples, na qual ele poderá usar o software com toda a facilidade e sem complicações.

Além disso, o sistema tem em suma dois grandes objetivos funcionais, são eles:

- Fazer a contabilização entre os ativos e passivos cadastrados pelo usuário, mostrando-lhe o histórico dos dados por ele inseridos, apresentando-lhe também o capital disponível, permitindo a inteira autonomia para que o indivíduo apresente ao sistema os próprios dados e fluxos conforme suas necessidades;
- Calcular de forma íntegra o capital (seja de crédito ou débito) aplicado (ou apenas uma simples simulação) em juros, sejam eles compostos ou simples, também permitindo ao usuário que este adapte a forma em que os dados são inseridos e mantidos de acordo com as necessidades.

2.2 Dificuldades

Atualmente, os grandes problemas no controle de finanças pessoais são:

- para aqueles que não possuem recursos para adquirir um software deste tipo, torna-se demasiadamente moroso registrar de forma organizada o que entra e o que sai no orçamento, e ainda passar muito tempo fazendo contas e mais contas, as quais exigem que seja empregado um esforço mental normalmente maior do que em atividades normais, pois a grande maioria das pessoas tem dificuldade em matemática, o que se dirá de matemática financeira;
- Para aqueles que possuem recursos para obter o desktop e ainda o sistema, é inviável passar algum tempo parado em frente ao computador registrando ativos e passivos.

3 FUNDAMENTOS TEÓRICOS

Aqui são expostos ao leitor os fundamentos e técnicas utilizados para a confecção deste projeto.

3.1 Orientação a Objetos

Para entender o que é orientação a objetos, primeiramente se faz necessário entender objetos.

Segundo LIMA, (2008): “Ao escrever um programa de computador em uma linguagem orientada a objetos, você criará, em seu computador, um modelo de alguma parte do mundo. As partes das quais o modelo é construído são os objetos que aparecem no domínio do problema.”

A OO apresenta grandes benefícios, entre eles:

- Reutilização de código;
- Concentração de funções nos lugares pertencentes a cada objeto;
- Fácil manutenção e alteração;

3.2 UML

Quando a Orientação a Objetos surgiu existia um grande problema: não havia padronização na documentação dos requisitos, da análise e do design dos sistemas que atendesse às necessidades dos mais variados tipos de aplicações. Isso gerava desordem entre aqueles que trabalhavam na área, implicando no difícil entendimento do sistema e conseqüentemente em sua manutenção.

A UML além de vir padronizar a documentação de um novo software, ela também está intrínseca na forma Orientada a Objetos.

A necessidade dos Sistemas de Informação comparam-se as necessidades da construção civil, na qual quanto maior a complexidade da obra, maior será a necessidade de boas maquetes e plantas.

Como escrito na citação acima, a UML possui vários recursos disponíveis aos desenvolvedores, os quais chamamos de diagramas, mas abordaremos aqui apenas os necessários à documentação deste projeto.

3.2.1 Diagramas da UML

Na definição de Lima:

Todo diagrama é constituído de elementos, cada um com um propósito, regras e notação diferentes para definir uma situação do processo de desenho. Embora a maioria dos diagramas sejam gráficos, apresentando **nós** (símbolos ou construções gráficas) relacionados entre si por uma linha, demonstrando uma conexão, o relacionamento pode ser expresso na forma de recipiente, em que um símbolo está contido em outro, ou de anexo, em que um símbolo aparece próximo a outro, geralmente ligado por uma linha pontilhada indicando uma nota explicativa ou comentário (2008).

3.2.1.1 Diagrama de Caso de Uso

Primeiramente, faz-se necessário entender o que é visão de caso de uso, novamente segundo Lima:

...a visão de caso de uso mostra conceitualmente o conjunto de funções que o sistema deve executar para atender aos requisitos do cliente, servindo como um contrato entre o cliente e o desenvolvedor. O sistema é visto sob a perspectiva do usuário, por isso a visão do caso de uso ocupa uma posição central – a base para as demais visões, essencial para análise, desenho, implementação, testes e plano do desenvolvimento do sistema (2008).

O caso de uso é representado graficamente por uma elipse conectada a um símbolo de ator. No diagrama pode ser incluído opcionalmente um retângulo, o qual mostra as atividades desenvolvidas dentro do software. Os atores ficam do lado de fora do retângulo, mostrando o que estão fazendo no programa. Os relacionamentos entre atores e casos de uso são apresentados por linhas entre eles.

Não apenas os atores podem interagir com casos de uso, mas também os casos de uso podem interagir entre si, através de *includes* ou *extends*, veja o exemplo abaixo de abertura de conta bancária:

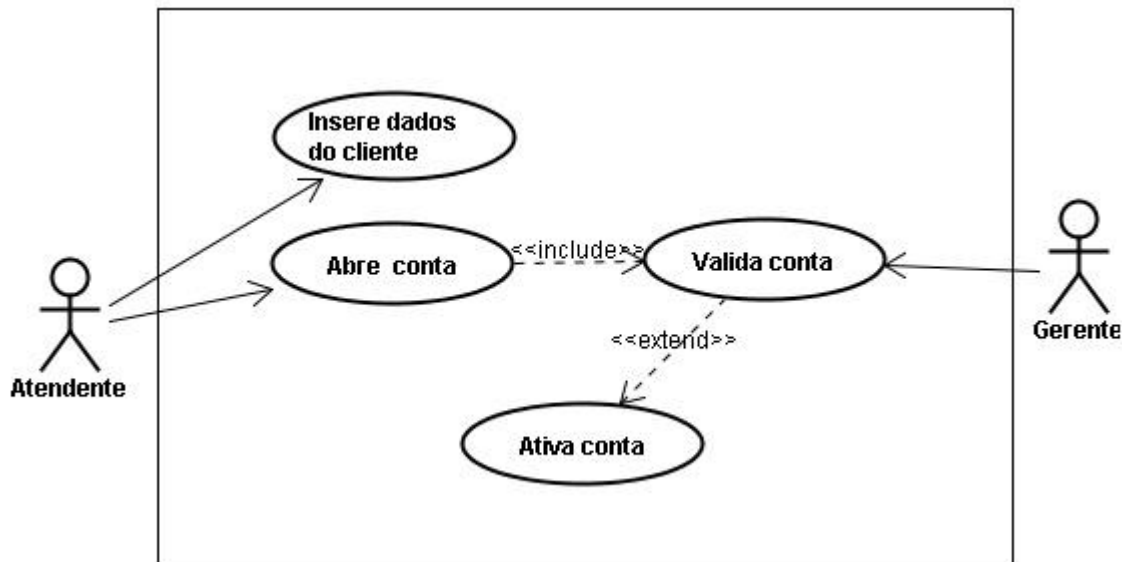


Figura 2 – Exemplo Diagrama de Caso de Uso

3.2.1.2 Diagrama de Atividade

Os três amigos, , Grady Booch, James Rumbaugh e Ivar Jacobson entraram no consenso de que:

O diagrama de atividade mostra o fluxo sequencial das atividades, é normalmente utilizado para demonstrar as atividades executadas por uma operação específica do sistema. Consistem em estados de ação, que contém a especificação de uma atividade a ser desempenhada por uma operação do sistema. Decisões e condições, como execução paralela, também podem ser mostrados no diagrama de atividade. O diagrama também pode conter especificações de mensagens enviadas e recebidas como partes de ações executadas (2011, p. 29)

O diagrama de atividades tem foco principal no que acontece dentro do sistema para que algo seja realizado, ou seja, quais “caminhos” o software percorre em uma determinada atividade. Conforme descreve Bames:

Segundo OMG, um diagrama de atividade, ou simplesmente uma atividade, especifica a coordenação de execuções de comportamentos, usando um modelo de fluxo de controle e de dados. O fluxo de execução é modelado como nós de atividade conectados por extremidades. Um nó (**nodo** ou **node**) pode ser a execução de um comportamento subordinado, como um cálculo computacional, uma chamada para uma operação ou manipulação de conteúdos de objetos. Nós de atividade também incluem construções de fluxo de controle, como sincronização, decisão e controle de concorrência. Atividades podem invocar hierarquias de outras atividades, resolvidas em ações individuais (2009).

Veja um exemplo de diagrama de atividades:

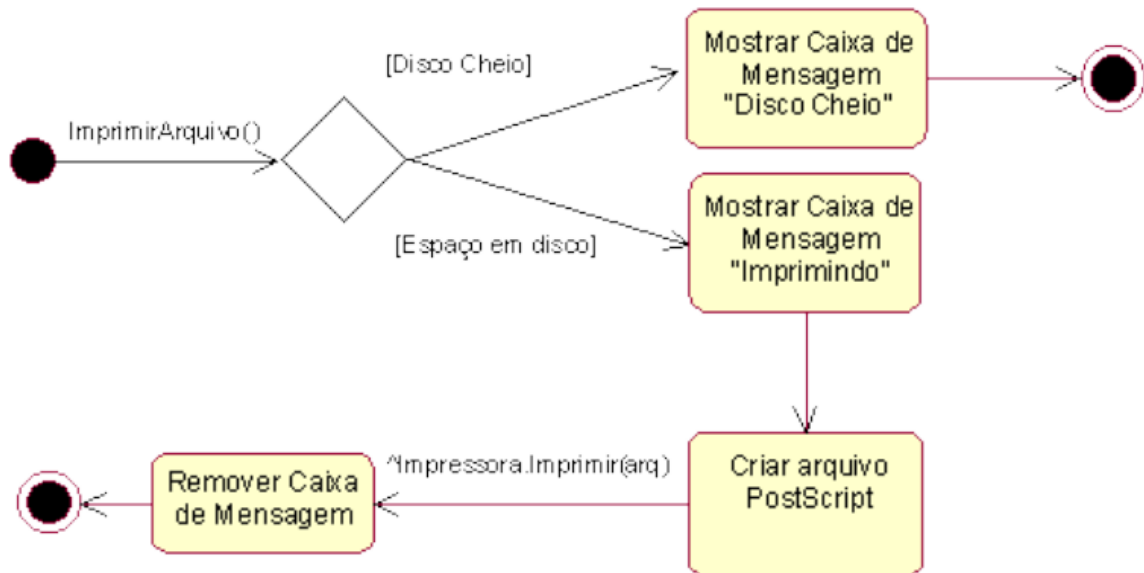


Figura 3 – Exemplo Diagrama de Atividades

Na figura acima vemos o estado inicial representado por um círculo negro preenchido, com o nome da atividade a ser executada. Logo após está o losango que representa a decisão (o losango pode conter mais de quatro saídas). Os retângulos com bordas curvas indicam a resposta do sistema conforme o decorrer da atividade. E finalmente temos o símbolo apresentando o final da atividade, o qual é um círculo menor preenchido dentro de outro não preenchido, veja que este pode ocorrer mais de uma vez.

3.2.1.3 Diagrama de Classes

Este diagrama, nas palavras de Bames:

O diagrama de classes representa a estrutura do sistema, recorrendo ao conceito de classe e suas relações. O modelo de classes resulta de um processo de abstração onde são identificados os objetos relevantes do sistema em estudo. Um objeto é uma ocorrência que tem interesse para o sistema em estudo e que se pretende descrever no seu ambiente, contendo identidade e comportamento. O comportamento de um objeto define o modo como ele age e reage a estímulos externos e a identidade de um objeto é um atributo que o distingue de todos os demais, sendo preservada quando o seu estado muda. Um objeto não é mais do que uma instância da classe (2009).

O grande objetivo deste diagrama é mostrar como as classes do software se relacionam, então eis a pergunta: o que é uma classe?

Segundo a documentação da UML pelo OMG, “uma classe descreve um conjunto de objetos que compartilham as mesmas especificações de atributos, operações, restrições e semântica. A finalidade de uma classe é classificar objetos e especificar os recursos que caracterizam a estrutura e o comportamento desses objetos”.

Um atributo mostra uma característica de um objeto que é comum aos demais da classe, exemplo pratico, existem vários livro em uma biblioteca, embora todos sejam livros, cada um deles possui um titulo, autor e editora diferentes

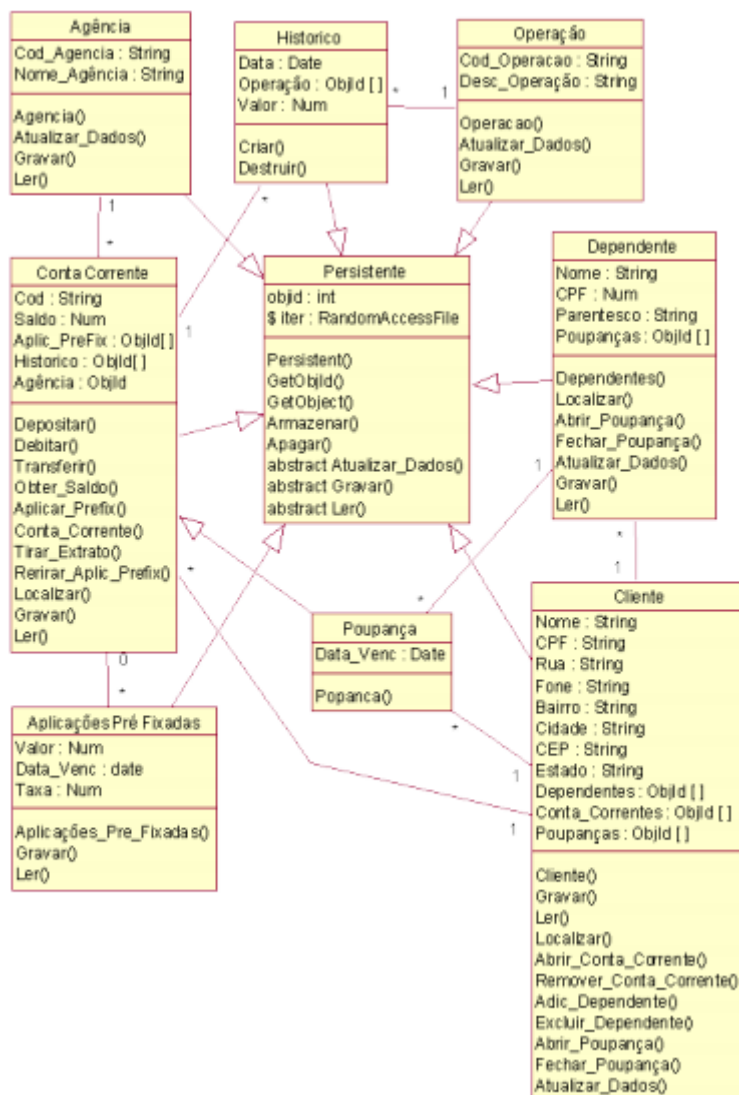


Figura 4 – Exemplo Diagrama de Classes

Como pode ser visto acima, em cada retângulo temos no topo o nome da classe, abaixo seus atributos e por último as funções que cada uma delas pode executar.

3.3 Modelo de Entidade e Relacionamento

O Modelo de Entidade e Relacionamento tem a função de apresentar em forma de imagens e símbolos a estrutura do banco de dados de um sistema de informação, buscando aproximar-se o máximo possível com o mundo real, onde os objetos e classes se relacionam.

Basicamente o MER é formado por: entidades, atributos, relacionamentos e cardinalidades.

3.3.1 Entidades

Tem a função de representar tabelas específicas do banco de dados, as quais cada uma guarda informações sobre um grupo de objetos semelhantes.

As entidades são representadas por retângulos.

3.3.2 Atributos

São os aspectos normalmente comuns a todos objetos da entidade, mas nem por isso os objetos são iguais. Por exemplo, na entidade Pessoas temos o atributo Altura, embora todos os objetos desta entidade sejam pessoas, elas possuem diferentes alturas.

Os atributos são representados por linhas seguidas de um círculo. Eles podem se diferenciar na forma gráfica dependendo do tipo do atributo.

3.3.3 Relacionamentos

Os relacionamentos tem a função de apresentar as ligações entre as tabelas no Banco de Dados.

São representados por um losango ligado de uma entidade a outra por linhas.

3.3.4 Cardinalidade

Tem a função de mostrar a possibilidade de vezes que uma classe pode interagir com a outra na qual está conectada.

São representados pelos símbolos “ 1 ”, “ : ” e “ n ”.

Exemplo:

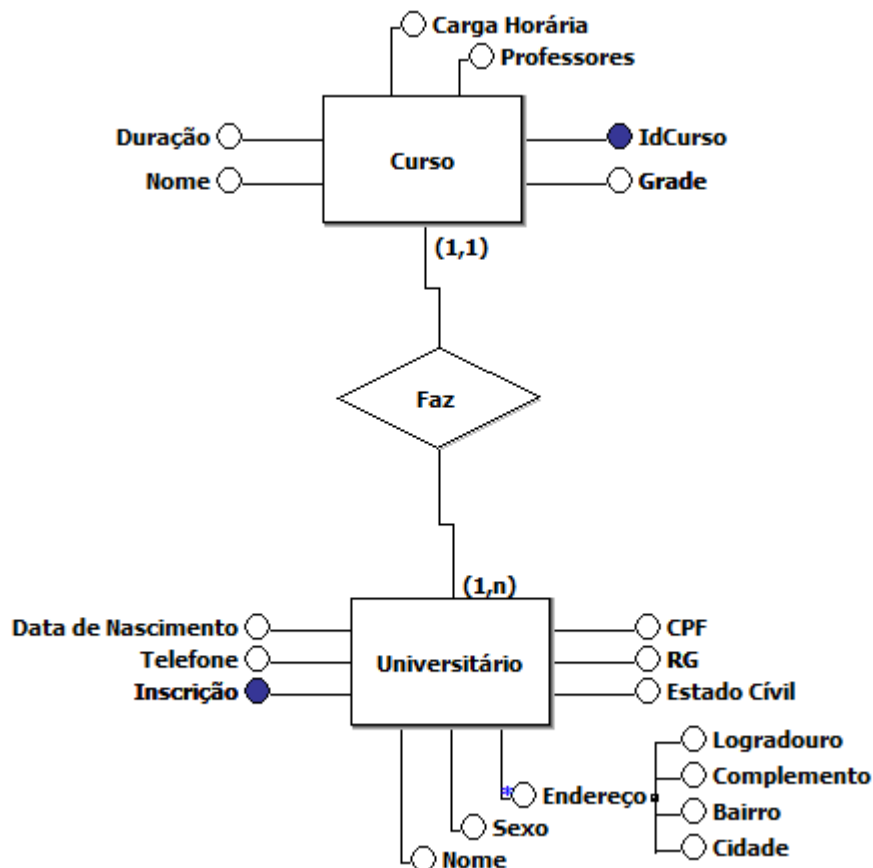


Figura 5 – Exemplo MER

A figura acima exemplifica todos os conceitos de MER vistos, onde Universitário e Curso são entidades, as quais possuem atributos, sendo eles chaves primárias (representados por um círculo preenchido), atributos comuns (representados por círculos não preenchidos) e compostos (atributos ligados a outros atributos).

O nome do relacionamento do losango é bem sugestivo, mostrando a razão do mesmo. As cardinalidades expressam que um aluno pode fazer apenas um curso (1:1), mas em um curso pode haver um ou mais alunos (1:n).

4 ESTUDO DE CASO: GERENCIADOR FINANCEIRO MÓVEL

O objetivo do GFM é trazer ao usuário mais agilidade e precisão na hora de controlar suas finanças.

Este software se faz necessário devido primeiramente a grande necessidade das pessoas em manter em dia suas contas.

4.1 Identificação dos Requisitos

Para se chegar ao levantamento correto de requisitos houve extensas conversas entre eu e o orientador, nas quais nos colocamos na condição de pessoas comuns que desejam facilitar o controle de finanças de forma prática e ágil.

4.2 Descrição dos Processos

O usuário deverá incluir na tabela Proventos do sistema o que recebe, e também seus gastos na tabela Despesas. O usuário poderá fazer a inclusão tanto de despesas quanto de recebíveis à quantidade que for necessária e em qualquer momento, isto é, o usuário poderá ter mais de um provento e mais de uma despesa.

Após cadastrados dados nas tabelas Proventos e Despesas o sistema fará a contabilização de determinado tempo, e mostrará na tela principal do software a quantia disponível ou devedora daquele período.

Além disso, o usuário também poderá manter atualizadas seus investimentos, de forma que o mesmo fornecerá os dados na tabela Investimentos, informando o valor, tempo, e taxa de rendimento, então o GFM calculará na forma de juros compostos os rendimentos e os adicionará ao principal.

O usuário também poderá simular metas a serem alcançadas, informando os dados na tabela Objetivos, então o sistema fará a soma entre o que o usuário possui em seus investimentos, quanto ele recebe por mês e quanto ele gasta, então mostrará ao usuário a quantia necessária que ele deve guardar no tempo desejado para que se tenha êxito no objetivo ou mostrará que não é possível alcançá-lo nas presentes condições.

4.3 Requisitos Funcionais

Esta tabela informa as funcionalidades do sistema, sem contudo, grandes especificações detalhistas.

Requisitos	Descrição
1 – Atualizar Proventos	Aqui devem estar as funções de incluir, alterar, excluir e consultar os proventos do usuário.
2 – Atualizar Despesas	Aqui devem estar as funções de incluir, alterar, excluir e consultar as despesas do usuário.
3 – Atualizar Investimentos	Aqui devem estar as funções de incluir, alterar, excluir e consultar os investimentos do usuário. Além disso, o sistema deverá calcular os rendimentos declarados pelo usuário, sobre juros compostos.
4 – Atualizar Objetivos	Aqui devem estar as funções de incluir, alterar, excluir e consultar os objetivos do usuário. O sistema deverá fazer as contas entre os Proventos, Despesas e Investimentos.
5 – Contabilizar	O software fará o balanço entre o que o usuário recebe (Proventos) e o que ele gasta (Despesas).

4.4 Requisitos Não Funcionais

Requisitos
O GFM funcionará somente na plataforma Android.
O sistema deverá ser de fácil usabilidade e intuitivo.

4.5. Diagramas de Caso de Uso

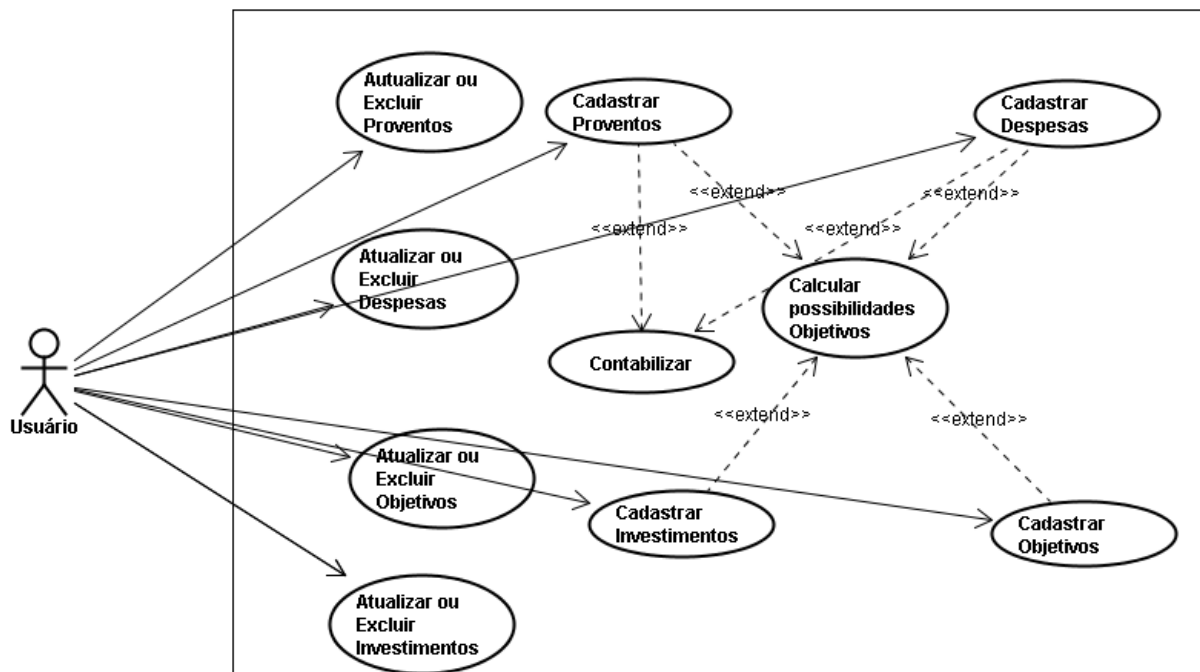


Figura 6 - Diagrama de Caso de Uso

4.6.Descrição dos casos de uso

Cadastrar Proventos

O usuário informa dados dos proventos que recebe.

Sistema: Usuário

Curso básico

- 1- O usuário aperta o botão “Incluir”
- 2- O sistema disponibiliza a interface com campos a serem preenchidos.
- 3- O usuário insere dados de seus proventos.
- 4- O usuário aperta o botão “Origem”.
- 5- O sistema disponibiliza a tela com os campos da origem, a serem preenchidos.
- 6- O usuário aperta o botão “Confirmar”.
- 7- O sistema retorna a tela de inserção de proventos.
- 8- O usuário aperta o botão “Salvar”.
- 9- MSG01: “Provento salvo”

Cadastrar Despesas

O usuário informa dados das despesas.

Usuário: Usuário.

Curso básico

- 1- O usuário aperta o botão “Incluir”
- 2- O sistema disponibiliza a interface com campos a serem preenchidos.
- 3- O usuário insere dados de suas despesas.
- 4- O usuário aperta o botão “Origem”.
- 5- O sistema disponibiliza a tela com os campos da origem, a serem preenchidos.
- 6- O usuário aperta o botão “Confirmar”.
- 7- O sistema retorna a tela de inserção de despesas.
- 8- O usuário aperta o botão “Salvar”
- 9- MSG02: “Despesa salva”

Cadastrar Investimentos

O usuário informa dados de seus investimentos.

Usuário: Usuário.

Curso básico

- 1- O usuário aperta o botão “Incluir”
- 2- O sistema disponibiliza a interface com campos a serem preenchidos.
- 3- O usuário insere dados de seus investimentos.
- 4- O usuário aperta o botão “Origem”.
- 5- O sistema disponibiliza a tela com os campos da origem, a serem preenchidos.
- 6- O usuário aperta o botão “Confirmar”.
- 7- O sistema retorna a tela de inserção de investimentos.
- 8- O usuário aperta o botão “Salvar”.
- 9- MSG03: “Investimento salvo”.

Cadastrar Objetivos

O usuário informa dados de seus objetivos.

Usuário: Usuário.

Curso básico

- 1- O usuário aperta o botão “Incluir”
- 2- O sistema disponibiliza a interface com campos a serem preenchidos.
- 3- O usuário insere dados de seus objetivos.
- 4- O usuário aperta o botão “Salvar”
- 5- Caso não haja recursos suficientes para atingir o objetivo; MSG04: “Confrontação de Objetivos e Despesas: Objetivo impossível de ser alcançado”.
- 6- Caso seja possível, o sistema mostrará a quantidade a ser disponibilizado o tempo para atingimento e a quantia que deve ser guardada.
- 7- MSG05: “Objetivo salvo”.

Alterar Proventos

O usuário irá alterar o provento desejado.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará provento já existente.
- 2- O sistema apresentará o provento com o botão “Alterar” ativo
- 3- O sistema disponibilizará os campos já preenchidos para alteração
- 4- O usuário fará as alterações e apertará o botão “Salvar”
- 5- MSG06: “Alterações salvas”

Alterar Despesas

O usuário irá alterar a despesa desejada.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará despesa já existente.
- 2- O sistema apresentará a despesa com o botão “Alterar” ativo
- 3- O sistema disponibilizará os campos já preenchidos para alteração
- 4- O usuário fará as alterações e apertará o botão “Salvar”
- 5- MSG07: “Alterações salvas”

Alterar Investimentos

O usuário irá alterar o investimento desejada.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará investimento já existente.
- 2- O sistema apresentará o investimento com o botão “Alterar” ativo
- 3- O sistema disponibilizará os campos já preenchidos para alteração
- 4- O usuário fará as alterações e apertará o botão “Salvar”

- 5- MSG08: “Alterações salvas”

Alterar Objetivos

O usuário irá alterar o objetivo desejado.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará o objetivo já existente.
- 2- O sistema apresentará o objetivo com o botão “Alterar” ativo
- 3- O sistema disponibilizará os campos já preenchidos para alteração
- 4- O usuário fará as alterações e apertará o botão “Salvar”
- 5- MSG09: “Confrontação de Objetivos e Despesas: Objetivo impossível de ser alcançado”.
- 6- MSG10: “Alterações salvas”

Excluir Proventos

O usuário irá excluir o provento desejado.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará provento já existente.
- 2- O sistema apresentará o provento com o botão “Excluir” ativo
- 3- O usuário apertará o botão “Excluir”
- 4- MSG11: “Executar exclusão?”
- 5- O usuário apertará o botão “Sim”
- 6- MSG12: “Provento excluído”

Excluir Despesas

O usuário irá excluir a despesa desejada.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará despesa já existente.
- 2- O sistema apresentará a despesa com o botão “Excluir” ativo
- 3- O usuário apertará o botão “Excluir”
- 4- MSG13: “Executar exclusão?”
- 5- O usuário apertará o botão “Sim”
- 6- MSG14: “Despesa excluída”

Excluir Investimentos

O usuário irá excluir o investimento desejado.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará o investimento já existente.
- 2- O sistema apresentará o investimento com o botão “Excluir” ativo
- 3- O usuário apertará o botão “Excluir”
- 4- MSG15: “Executar exclusão?”
- 5- O usuário apertará o botão “Sim”
- 6- MSG16: “Investimento excluído”

Excluir Objetivos

O usuário irá excluir o objetivo desejado.

Usuário: Usuário.

Curso básico

- 1- O usuário selecionará objetivo já existente.
- 2- O sistema apresentará o investimento com o botão “Excluir” ativo

- 3- O usuário apertará o botão “Excluir”
- 4- MSG17: “Executar exclusão?”
- 5- O usuário apertará o botão “Sim”
- 6- MSG18: “Objetivo excluído”

Contabilizar

O sistema irá fazer a contabilização entre Proventos e Despesas.

Sistema: Sistema

Curso básico

- 1- O sistema fará a soma entre despesas e proventos.
- 2- Será emitido o resultado na tela inicial

4.7 Glossário de Mensagens

A descrição dos termos é organizada em uma tabela onde cada linha representa uma rastreabilidade. As colunas da tabela representam respectivamente: Código de Rastreabilidade, Descrição de Mensagens e o caso de uso que utiliza a mensagem.

Código de Rastreabilidade	Descrição da Mensagem	Caso de uso
MSG01	Provento salvo	Cadastrar Proventos
MSG02	Despesa salva	Cadastrar Despesa
MSG03	Investimento salvo	Cadastrar Investimento

MSG04	Confrontação de Objetivos e Despesas: Objetivo impossível de ser alcançado	Cadastrar Objetivos
MSG05	Objetivo salvo	Cadastrar Objetivos
MSG06	Alterações salvas	Atualizar ou Excluir Proventos
MSG07	Alterações salvas	Atualizar ou Excluir Despesas
MSG08	Alterações salvas	Atualizar ou Excluir Investimentos
MSG09	Confrontação de Objetivos e Despesas: Objetivo impossível de ser alcançado	Atualizar ou Excluir Objetivos
MSG10	Alterações salvas	Atualizar ou Excluir Objetivos
MSG11	Executar exclusão?	Alterar ou Excluir Proventos
MSG12	Provento excluído	Alterar ou Excluir Proventos
MSG13	Executar exclusão?	Alterar ou Excluir Despesas
MSG14	Despesa excluída	Alterar ou Excluir Despesas
MSG15	Executar exclusão?	Alterar ou Excluir Investimentos
MSG16	Investimento excluída	Alterar ou Excluir Investimento
MSG17	Executar exclusão?	Alterar ou Excluir Investimentos

MSG18	Objetivo excluído	Alterar ou Excluir Objetivos
-------	-------------------	---------------------------------

5 CONSIDERAÇÕES FINAIS

O desenvolvimento deste projeto teve como base principal a Orientação à Objetos, na qual torna-se possível expor ao usuário/cliente as diversas características do sistema de forma compreensível a qualquer pessoas, mesmo que sem grandes conhecimentos desta ciência. Para a construção do GFM foi utilizada como ferramenta principal o Eclipse, no qual foram utilizadas as linguagens Java, XML e para o banco de dados SQLite. Esta importante ferramenta proporcionou

Tendo como principal linguagem de edificação do sistema o Java, que possui características que o tornam ideal para o desenvolvimento orientado à objetos.

6 TRABALHOS FUTUROS

Pretende-se aprimorar o software com funcionalidades extras, como cálculo de parcelas de créditos a serem adquiridos, como também a serem emprestados, inserir mais campos em cada tabela de forma a deixar ainda mais completas as informações prestadas pelo sistema. E com o aperfeiçoamento do mesmo disponibilizá-lo no ANDROID MARKET.

7 REFERÊNCIA BIBLIOGRÁFICA

- GUEDES, Gilleanas T.A..**UML: Uma Abordagem Prática**. terceira edição, São Paulo, 2008. Novatec
- JÚNIOR. RODOLFO M. S.. **Análise e Projeto Orientado a Objetos Usando UML e o Processo Unificado**. Dissertação de Conclusão de Mestrado – Universidade Federal do Pará.,2001.
- ALVES. Luciano. Apostila Android – Programando Passo a Passo 4ª Edição PDF.Acessado em 20 de outubro de 2011.
- LIMA, Adilson da Silva, **UML 2.0: do requisito à solução**. 3 ed. São Paulo: Ed. Érica, 2008.
- BOOCH, Grady et al **Unified Modeling Language. Linguagem de Modelagem Unificada (Em português)**, disponível em <http://www.oocities.org/br/analiset43/Material_UML.pdf> Acessado em 20 de outubro de 2011.
- BOOCH, Grady, **UML: guia do usuário / Grady Booch, James Rumbaugh, Ivar Jacobson**; tradução de Fabio Freitas da Silva e Cristina de Amorim Machado. –5 ed. Rio de Janeiro: Elsevier, 2005
- Barnes, David J.& Michael Kölling; **Programação Orientada a Objetos com Java**. Tradução Edson Furmankiewicz. 4 ed. São Paulo:Pearson Prentice Hall, 2009.

APÊNDICE A – DOCUMENTAÇÃO DO SISTEMA

APÊNDICE A.1-MER

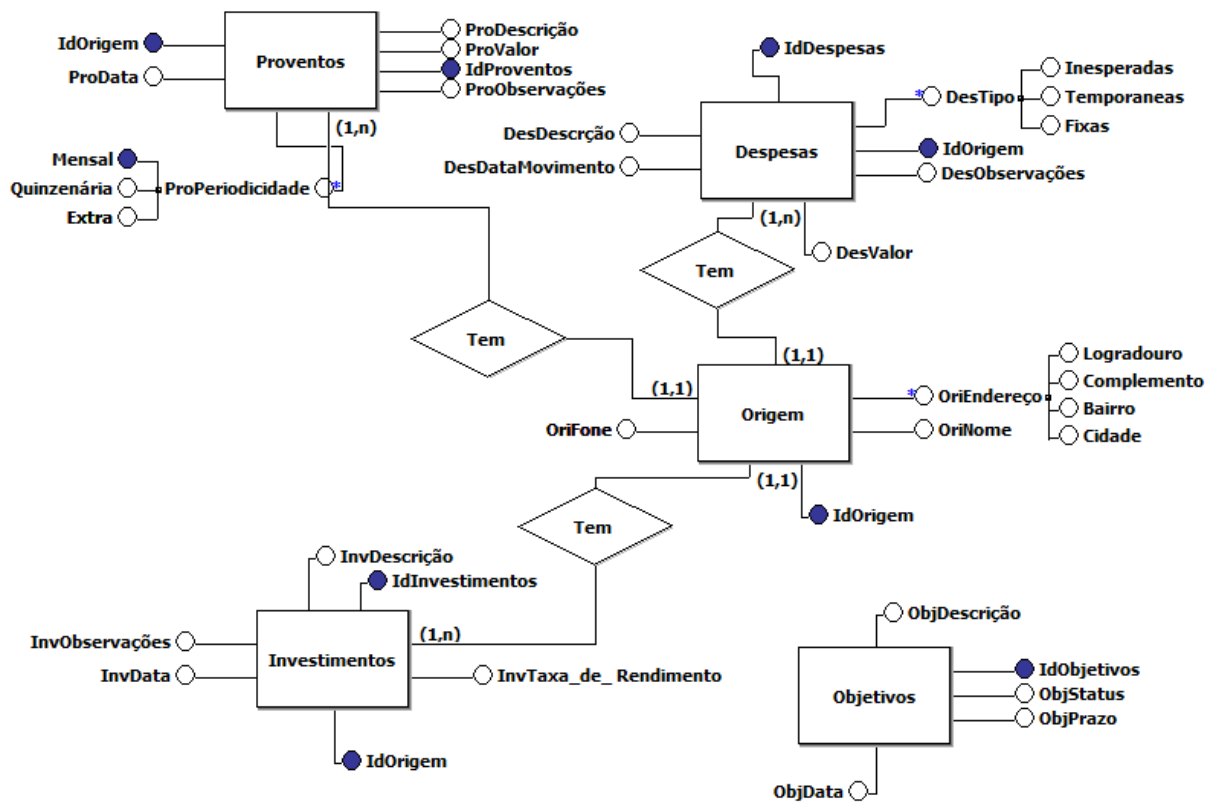


Figura 7- MER

APÊNDICE A.2 -DER

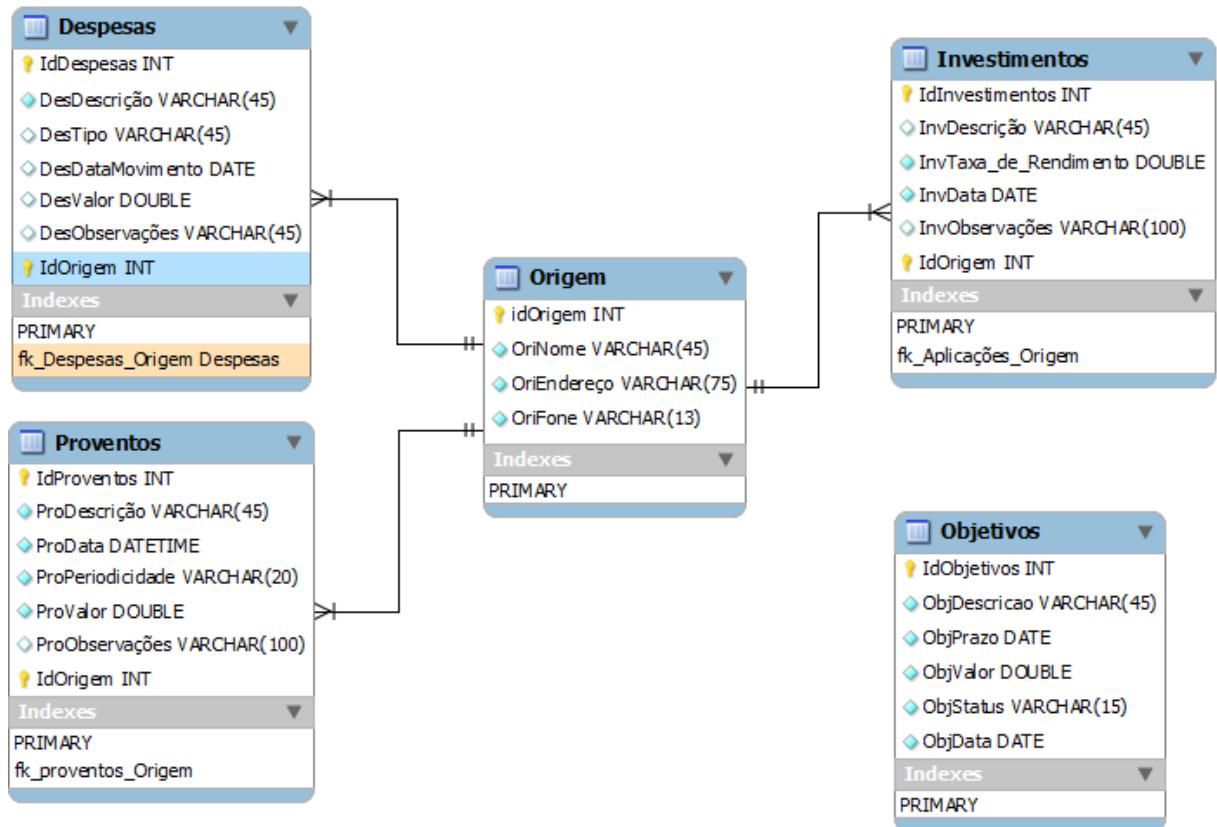
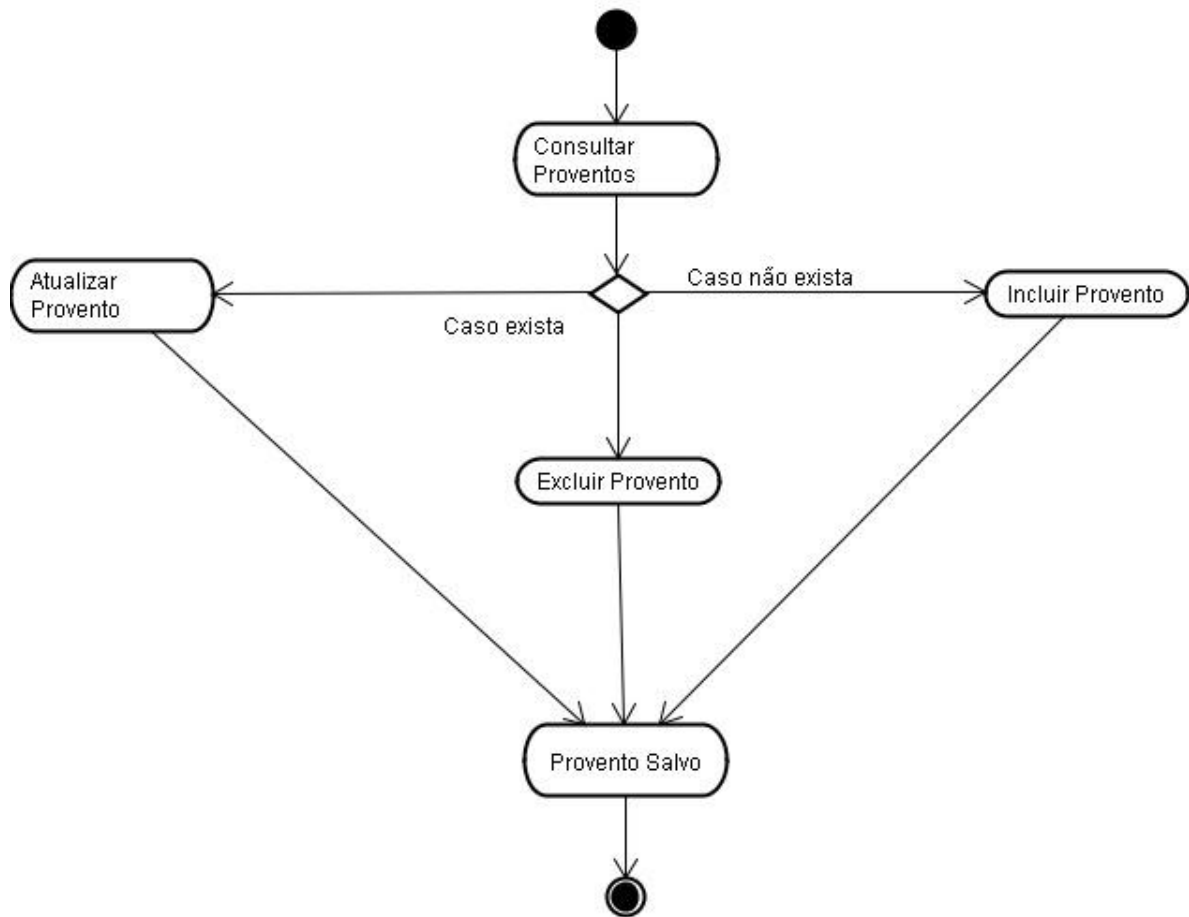
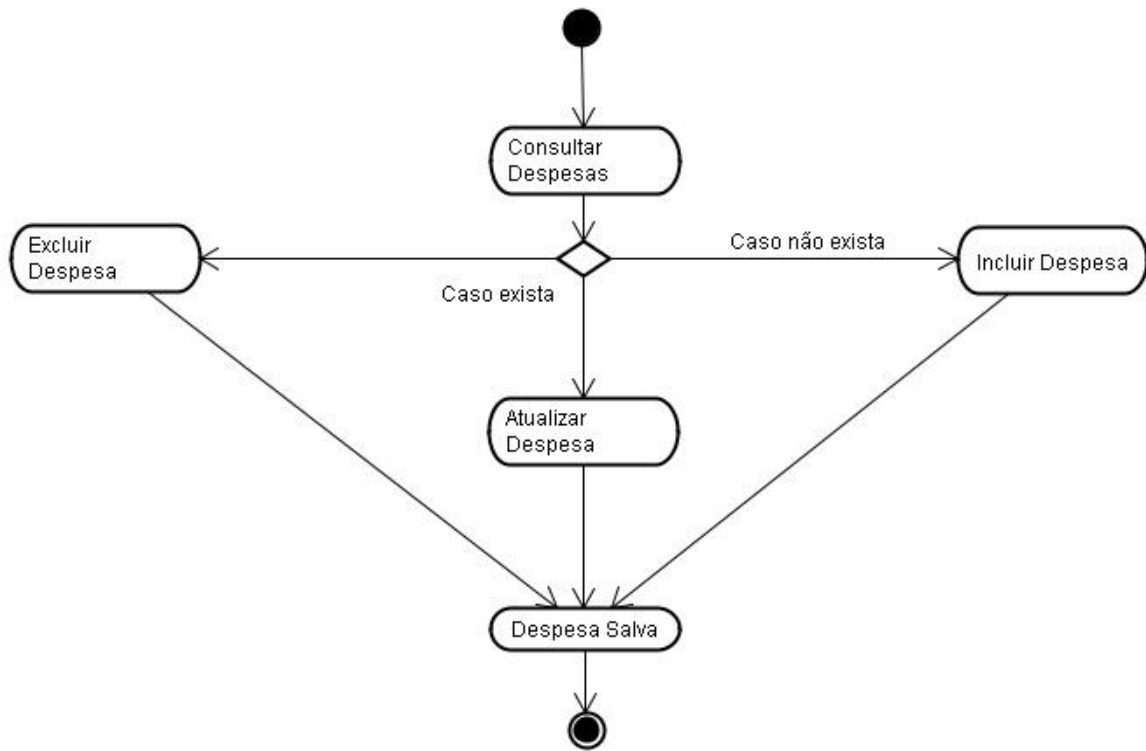
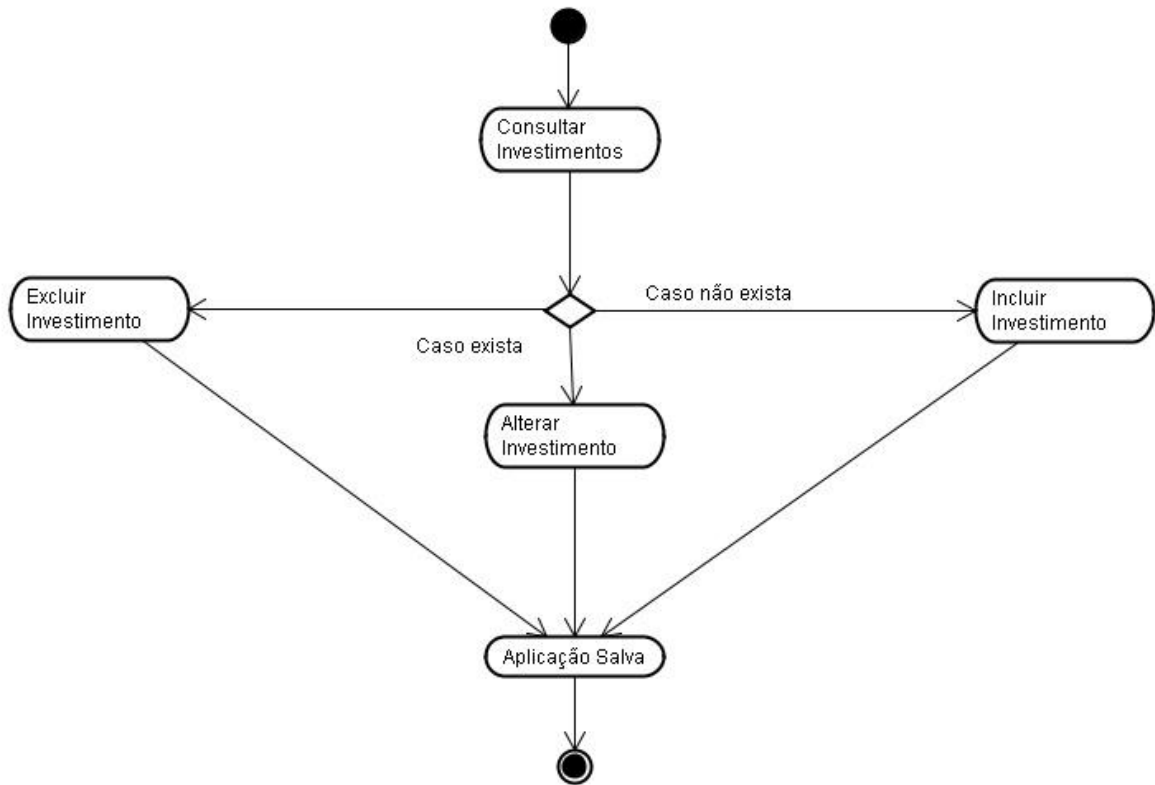
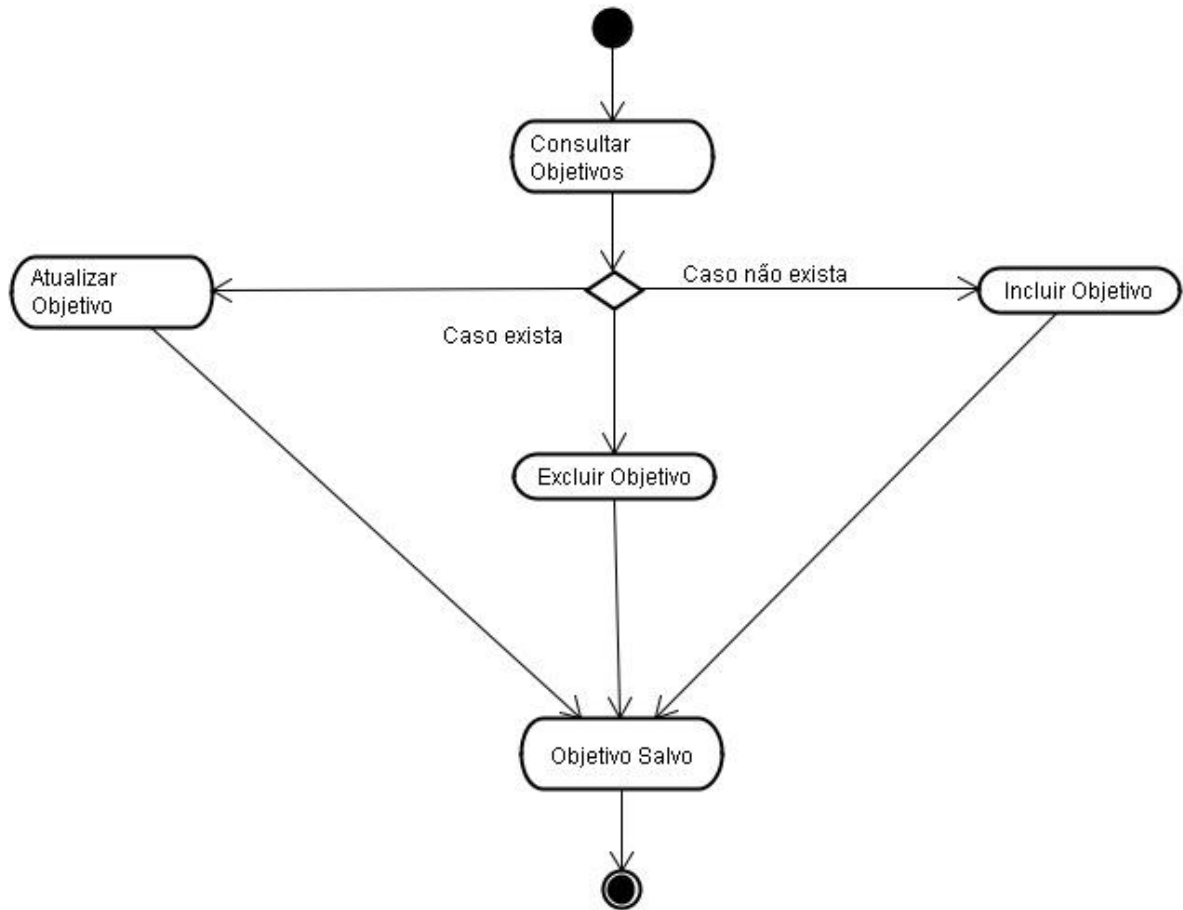


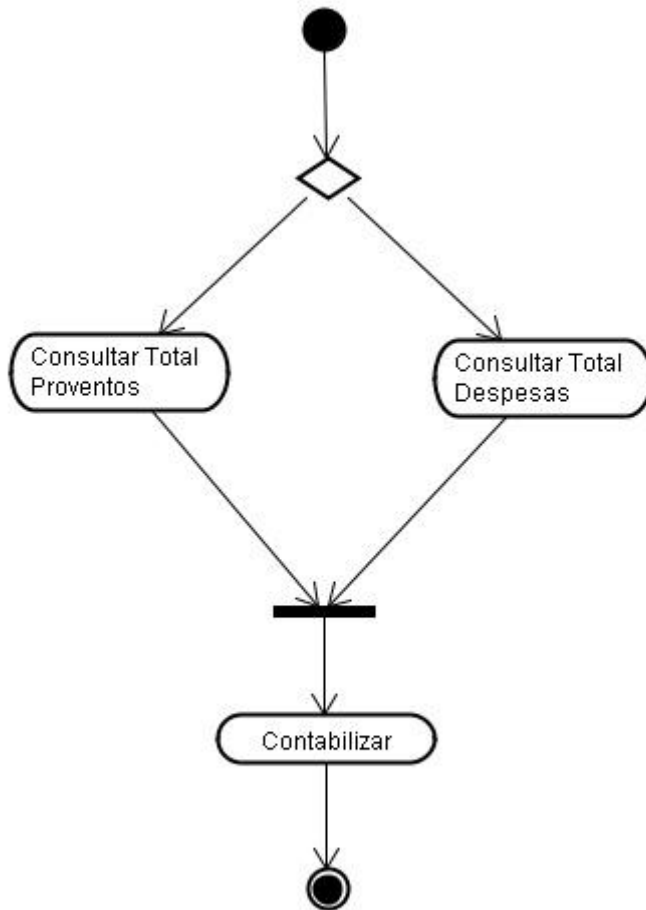
Figura 8 - DER

APÊNDICE A.3- Diagrama de Atividade – Atualizar Proventos**Figura 9 - Diagrama de Atividade Atualizar Proventos**

APÊNDICE A.4- Diagrama de Atividade – Atualizar Despesas**Figura 10 – Diagrama de Atividade Atualizar Despesas**

APÊNDICE A.5 -Diagrama de Atividade – Atualizar Investimentos**Figura 11 - Diagrama de Atividade Atualizar Investimentos**

APÊNDICE A.6-Diagrama de Atividade – Atualizar Objetivos**Figura 12 - Diagrama de Atividade Atualizar Objetivos**

APÊNDICE A.7-Diagrama de Atividade – Contabilizar**Figura 13 - Diagrama de Atividade Contabilizar**

APÊNDICE A.8 –Diagrama de Classes

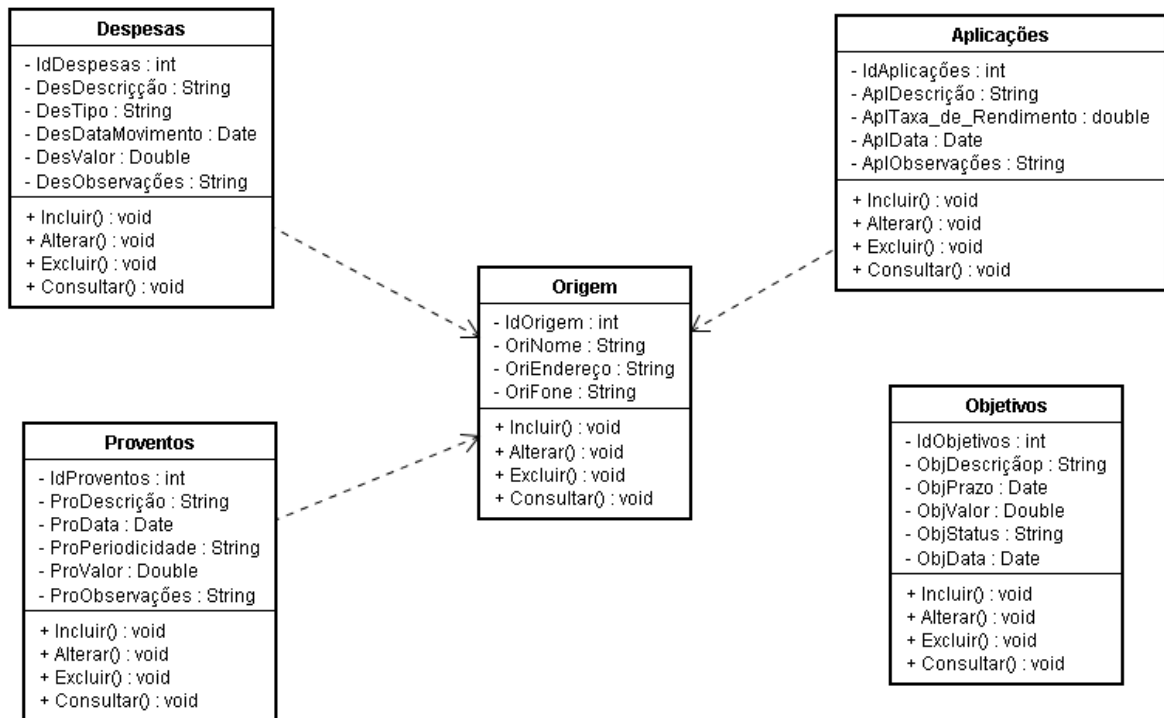


Figura 14 - Diagrama de Classes

APÊNDICE B – PROTÓTIPOS DO SOFTWARE

APÊNDICE B.1-Protótipo Tela Principal

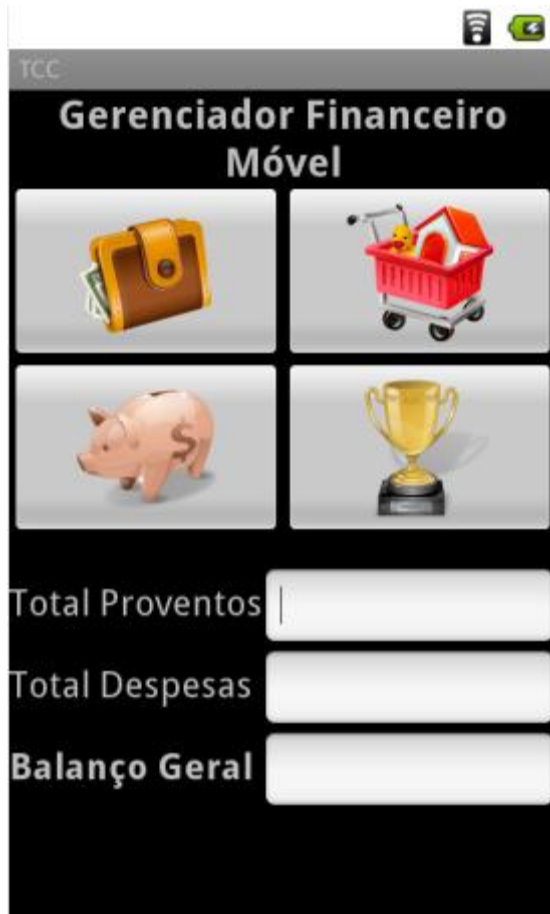


Figura 15 – Protótipo Tela Principal

APÊNDICE B.2-Protótipo Incluir Proventos



TCC

Incluir Proventos

Descrição

Data

Period.

Valor

Origem

Observações

Total

Salvar

Detailed description: This is a mobile application prototype for adding income. The screen has a dark background with white text and input fields. At the top, there's a status bar with 'TCC' and icons for signal, Wi-Fi, and battery. Below that is the title 'Incluir Proventos'. The form consists of several rows: 'Descrição' with a text input field; 'Data' with a date input field; 'Period.' with a dropdown menu; 'Valor' with a text input field; 'Origem' with a text input field; 'Observações' with a text input field; and 'Total' with a text input field. At the bottom left, there is a 'Salvar' button.

Figura 16 – Protótipo Incluir Proventos

APÊNDICE B.3-Protótipo Incluir Despesas



O protótipo da tela 'Incluir Despesas' apresenta o seguinte layout:

- Barra superior: 'TCC' e ícones de Wi-Fi e bateria.
- Título: 'Incluir Despesas'.
- Campos de entrada: 'Descrição', 'Tipo' (menu suspenso), 'Data', 'Valor', 'Origem' (botão) e 'Observações'.
- Botão 'Salvar' na base da tela.
- Elemento 'Total' com um campo de entrada associado.

Figura 17 – Protótipo Incluir Despesas

APÊNDICE B.4 – Protótipo Incluir Investimentos



O protótipo de tela, intitulado 'Incluir Investimentos', apresenta um layout vertical de campos de entrada. No topo, há uma barra de status com o texto 'TCC' e ícones de conexão. O formulário contém os seguintes elementos:

- Um campo de texto para 'Descrição'.
- Um campo de texto para 'Taxa Rend.'.
- Um campo de texto para 'Valor'.
- Um campo de texto para 'Origem'.
- Um campo de texto para 'Observações'.
- Um campo de texto para 'Total'.
- Um botão 'Salvar' localizado na base da tela.

Figura 18 – Protótipo Incluir Investimentos

APÊNDICE B.5 – Protótipo Incluir Objetivos



Um protótipo de tela de um aplicativo móvel com o título "TCC" no topo. O formulário principal é intitulado "Incluir Objetivos" e contém os seguintes campos:

- Descrição: Campo de texto com uma barra de cursor.
- Prazo: Campo de texto.
- Valor: Campo de texto.
- Status: Campo de seleção com uma seta para baixo.
- Data: Campo de texto.
- Total: Campo de texto.

Um botão "Salvar" está localizado na parte inferior esquerda da tela.

Figura 19 – Protótipo Incluir Objetivos

APÊNDICE B.6 – Protótipo Origem

TCC

Origem

Nome

Endereço

Logradouro

Complemento

Bairro

Cidade

Fone

Salvar

Figura 20 – Protótipo Origem

Apêndice B.7-Alterar Proventos



TCC

Alterar Proventos

Descrição

Data

Period.

Valor

Origem

Observações

Total

Salvar

The image shows a mobile application screen with a dark background. At the top, there are status icons for Wi-Fi and battery. Below them is a header bar with the text 'TCC' on the left and 'Alterar Proventos' in the center. The main content area contains several input fields: 'Descrição', 'Data', 'Period.' (with a dropdown arrow), 'Valor', 'Origem' (with a button-like appearance), 'Observações', and 'Total'. At the bottom left, there is a 'Salvar' button.

Figura 21 - Protótipo Alterar Proventos

APÊNDICE B.8 – Alterar Despesas



O protótipo da tela 'Alterar Despesas' apresenta os seguintes elementos:

- Barra superior: 'TCC' e ícones de Wi-Fi e bateria.
- Título: 'Alterar Despesas'.
- Campos de entrada: 'Descrição', 'Tipo' (menu suspenso), 'Data', 'Valor', 'Origem' (botão) e 'Observações'.
- Botão 'Total' e campo de entrada.
- Botão 'Salvar'.

Figura 22 - Protótipo Alterar Despesas

APÊNDICE B.9 – Alterar Investimentos



TCC

Alterar Investimentos

Descrição

Taxa Rend.

Valor

Origem

Observações

Total

Salvar

Detailed description: This is a mobile application prototype for editing investments. The screen has a black background with white text and input fields. At the top, there is a status bar with 'TCC' and icons for signal, Wi-Fi, and battery. Below that is a title bar with 'Alterar Investimentos'. The form consists of several rows: 'Descrição' with a text input field; 'Taxa Rend.' with a text input field; 'Valor' with a text input field; 'Origem' with a text input field; 'Observações' with a text input field; and 'Total' with a text input field. At the bottom left, there is a 'Salvar' button.

Figura 23 – Protótipo Alterar Investimentos

APÊNDICE B.10 – Alterar Objetivos



O protótipo da tela 'Alterar Objetivos' apresenta o seguinte layout:

- Barra superior: 'TCC' e ícones de Wi-Fi e bateria.
- Título: 'Alterar Objetivos'.
- Campos de entrada: 'Descrição', 'Prazo', 'Valor', 'Data' e 'Total'.
- Menu suspenso: 'Status' com uma seta para baixo.
- Botão: 'Salvar'.

Figura 24 - Protótipo Alterar Objetivos

APÊNDICE B.11 –Alterar Origem

Um protótipo de tela de um aplicativo móvel com o título "Alterar Origem". A interface possui um fundo escuro com elementos em tons de cinza e branco. No topo, há uma barra de status com ícones de Wi-Fi e bateria. Abaixo, o título "Alterar Origem" está centralizado. O formulário contém campos de entrada para "Nome", "Endereço", "Logradouro", "Complemento", "Bairro", "Cidade" e "Fone", todos em branco. Um botão "Salvar" está localizado na parte inferior esquerda.

TCC

Alterar Origem

Nome

Endereço

Logradouro

Complemento

Bairro

Cidade

Fone

Salvar

Figura 24 - Protótipo Alterar Origem