As pessoas tem necessidade de se organizarem financeiramente, mas isso não é tão simples para a maioria, a saída muitas vezes é realizar esse controle através do Excel ou até mesmo manualmente. Com tal dificuldade existente surge a possibilidade de criar softwares que realizem o controle de gastos diário exibindo ao fim do mês ou ano quanto foi gasto e como o que. Existe uma grande variedade de aplicações nesse ramo, mas em sua grande maioria acabam sendo complexas por exigir do usuário dados de uma conta bancaria, saldo inicial, dentre outras informações que não satisfazem quem busca algo simples e mais dinâmico.

O grupo possui tal necessidade e até então não encontrou uma saída simples em sistemas desse tipo, assim sendo resolveu buscar uma alternativa que elimine esse problema e que seja a solução para outras pessoas que também buscam algo do tipo.

O sistema deverá ser o mais simples possível, exigindo o mínimo de informações necessárias e ao fim gerar um extrato simples e objetivo do que foi gasto durante o período de tempo escolhido.

• Hierarquia de Classes

Para aplicação utiliza extends Application e para fazer a mudança de tela utiliza extends StackPane

• Implementação de Interfaces

Para a mudança de tela se cria uma interface, a qual se chama ControlledScreen, que é implementada nos controles de fxml onde tb se implementa a interface Initializable.

• Encapsulamento de Atributos e exposição com métodos de acesso (get/set)

Os atributos que vem da interface(Button, TextField) são declarados como private ocorrendo o seu encapsulamento e é necessário usar get para acessa-los. Os métodos criados para os botões da interface tb são declarados private.

• Uso de Polimorfismo

• Construtores parametrizados

• Tratamento de Exceções

Se utiliza tratamento de exceções ao fazer a leitura e escrita dos dados de login no arquivo texto.

• Interface gráfica interativa que processe eventos de teclado e mouse

Todo o programa foi feito utilizando interface gráfica, com botões e textfields onde o usuário pode interagir.

• Armazenamento e recuperação de dados persistentes (por serialização ou arquivos-texto)

Os dados de login são armazenados em um arquivo texto

• Estruturas equivalentes a Collections (usando no mínimo uma das classes de cada interface principal: List, Set e Map)

ArrayList que é utilizada para armazenar o login e senha

• Geração de arquivo para instalação/execução do sistema fora do ambiente de desenvolvimento