

Trabalho

Implementar um programa, com interface gráfica de usuário (GUI), para ser utilizado para controlar as receitas e despesas domésticas de uma pessoa.

As receitas são todos os recebimentos que a pessoa obtém e são classificadas em recebimento de salário, décimo terceiro, férias e outras receitas. As despesas são gastos que a pessoa tem e são classificadas em gastos com pagamento de alimentação, transporte, residência, saúde, educação, entretenimento e outras despesas.

O programa deve atender aos seguintes requisitos funcionais (o programa deve permitir):

- 1) Incluir receitas. Uma receita deve ser categorizável e deve ser possível informar a data em que a receita ocorreu (ou vai ocorrer);
- 2) Incluir as despesas. Uma despesa deve ser categorizável. Deve ser possível informar a data em que a despesa ocorreu (ou vai ocorrer);
- 3) Consultar o saldo que a pessoa tem disponível até a data atual;
- 4) Consultar o saldo que a pessoa tem disponível, independente do período;
- 5) Listar todas as receitas lançadas;
- 6) Listar todas as despesas lançadas;
- 7) Listar todos os lançamentos (receitas e despesas) ordenados por data. A cada lançamento, exibir como aquele lançamento impactou no saldo. Isto é, totalizar o saldo até aquele momento, assim como ocorre num extrato;

Também devem ser atendidos aos seguintes requisitos não funcionais:

- 8) Deve utilizar do conceito de herança;
- 9) Os dados submetidos pelo usuário devem ser gravados, bem como o programa deve recuperar os dados do disco quando o programa for iniciado. Os arquivos de dados devem ser formatados em arquivo no formato .csv. Este requisito será detalhado nas semanas seguintes.

MÉTODO DE AVALIAÇÃO

É esperado que seja construído:

- **[2,0 pontos]** Diagrama de classes detalhando todas as classes e relacionamentos, com código fonte compatível com o diagrama. O diagrama de classes deve conter apenas as classes da camada de negócios. Utilizar a linguagem UML para desenhar o diagrama. O software para desenhar o diagrama de classes é de livre escolha da equipe, deixo duas sugestões, o software draw.io (DRAWIO, 2023) ou o software Astah UML, versão *Student* (ASTAH, 2023).
- **[5,0 pontos]** Implementação do programa, atendendo aos requisitos funcionais e construído utilizando-se a arquitetura em duas camadas. As classes da camada de negócio devem ser devidamente documentadas com o estilo `javadoc`;
- **[1,0 ponto]** Construção de um plano de testes para validar as classes da camada de negócios;
- **[2,0 pontos]** Implementação do plano de testes através do framework JUnit. Todos os métodos públicos (que não são *getter* e *setter*) devem possuir testes.

As classes do projeto devem ser organizadas em pacotes conforme a arquitetura de duas camadas. O projeto deve conter apenas uma classe com o método `main()`.

Os trabalhos serão apresentados para a turma na última semana de aula. Cada equipe terá um tempo a ser definido pelo professor para apresentar seu trabalho.



MATERIAL A SER SUBMETIDO NO AVA

No AVA deverá ser submetido:

- O projeto contendo os arquivos fontes exportados pelo Netbeans (usar a opção Arquivo > Exportar Projeto > Para Zip). Não utilizar bibliotecas de terceiros;
- O plano de testes;
- A implementação do plano de testes.
- O Diagrama de Classes UML (Enviar no formato pdf).

RESTRIÇÕES E PRAZO

- O trabalho deve ser feito em equipe de 3 pessoas.
- O trabalho deve ser entregue até o dia 26/11 as 23:59.

REFERENCIAS

DRAWIO. Disponível em: <<https://www.drawio.com/>>. Acesso em: outubro 2023

ASTAH. Astah Community. **Astah - Software Design Tools for Agile Teams**. Disponível em: <<https://astah.net/products/free-student-license/>>. Acesso em: outubro 2023