



Projeto final - PHP

UTFundos de Investimentos

Necessidades apresentadas pelo cliente

1. O sistema deve permitir que o usuário (investidor) realize depósitos e saques na sua conta de investimentos, ao realizar o depósito o sistema deve dar algumas opções de prazos mínimos para saque (dias, meses e anos), quanto maior o prazo selecionado maior é o rendimento (o rendimento de um mês é igual 1% do valor investido), caso o usuário saque antes do prazo cobrar uma taxa administrativa de 0,5% para cada mês que o dinheiro estava investido.
2. O sistema deve permitir gerar relatórios de rendimentos para que o usuário (investidor) acompanhe os seus ganhos. O cliente acha interesse implementar vários filtros para visualização dos dados como por período, por investimento e etc.
3. O sistema deve permitir que o usuários (gestor de investimento) faça investimentos com o dinheiro depositado pelos investidores. O objetivo dos gestores de fundos é maximizar os ganhos e bater suas metas definidas pelo usuário (administrador), a meta padrão é render 2% ao mês para conseguir pagar os investidores.
4. O sistema deve permitir que o usuário (gestor de investimentos) tenha acesso ao dashboard de ações, onde ele pode acompanhar até quatro ações ao mesmo tempo (as quais ele selecionou), com botões de compra e venda de fácil acesso.
5. O sistema deve permitir que o usuário (gestor de investimentos) acompanhe a progressão de ganhos da sua carteira de investimentos (ações que ele possui dinheiro investido).
6. O sistema deve permitir que o usuário (administrador) realize o cadastro de novos usuários (administrador, gestor e investidor). Para o usuário (gestor) ele pode definir um valor limite para compra de investimentos, sendo sempre menor que o valor de dinheiro em caixa (valor depositado pelos investidores) e definir a meta de rendimento mensal para cada gestor.
7. O sistema deve permitir que o usuário (administrador) acompanhe a progressão de ganhos da empresa (fluxo de caixa), por gestor e por

investidor.

8. Para aumentar a veracidade o sistema deve consultar os dados das ações em API de bolsa de valores.
 - a. <https://www.alphavantage.co>
 - b. <http://meumobi.github.io/stocks%20apis/2016/03/13/get-realtime-stock-quotes-yahoo-finance-api.html>

Entregas do projeto

- 1. Definição dos requisitos de software (Funcionais e Não-Funcionais).**
 - a. Alguns requisitos já foram apresentados pelo "cliente".
 - b. A equipe deve analisar se os requisitos já apresentados são suficientes para implementação do projeto.
 - c. Modelar os requisitos quando necessário (observar a complexidade de cada um).
 - d. Dica: novos requisitos serão bem-vindos.
 - e. Dica: Não esqueça que o cronograma é limitado.
 - f. Não esquecer de entregar o relatório de atividades.
- 2. Arquitetura de software, Padrões e Tecnologias que serão utilizadas na implementação.**
 - a. Descrever a arquitetura de software escolhida e justificar as suas escolhas.
 - b. Exemplos de arquitetura de software: Cliente-Servidor, MVC, SOA, Microserviços, etc.
 - c. Não esquecer de entregar o relatório de atividades.
- 3. Modelagem do banco de dados, Scripts de criação e inserção utilizados.**
 - a. Entregar o modelo lógico e físico. Lembrar de descrever as tabelas utilizadas no banco de dados.
 - b. Entregar os scripts de criação em um arquivo separado .zip.
 - c. Não esquecer de entregar o relatório de atividades.
- 4. Diagrama de classes e Mockups da interfaces do sistema.**
 - a. Entregar o diagrama de classes, não esquecer de seguir a arquitetura apresentada anteriormente e descrever as classes utilizadas.
 - b. Entregar o mockup de todas interfaces que serão construídas no sistema.
 - c. Não esquecer de entregar o relatório de atividades.
- 5. Implementação e apresentação do sistema desenvolvido.**
 - a. Entregar todo o código fonte e arquivos necessários para execução do projeto. Não será aceito link do github.
 - b. Preparar apresentação de venda do projeto desenvolvido. Nessa apresentação montar um slide apresentado os pontos fortes do projeto

- e porque o cliente deve escolher o seu projeto em relação aos outros. Apresentar o sistema o funcionamento do sistema em tempo real.
- c. Não esquecer de entregar o relatório de atividades.

Notas sobre a entrega do projeto

1. As entregas serão incrementais e todos os grupos criarão um relatório do projeto contendo a seguinte estrutura:
 - a. Capa (conter o nome dos alunos do grupo)
 - b. Requisitos (Funcionais e Não-Funcionais)
 - i. Relatório de atividades
 - c. Arquitetura de software, Padrões e Tecnologias
 - i. Relatório de atividades
 - d. Modelagem do banco de dados
 - i. Relatório de atividades
 - e. Diagrama de classes
 - f. Interfaces do sistema (mockups)
 - i. Relatório de atividades referente ao diagrama de classes e mockups.
 - g. Relatório de atividades referente às atividades de implementação.
2. Entregar todos os arquivos utilizados para entrega como relatório (PDF), scripts, imagens e etc em único arquivo zipado, com o seguinte nomenclatura "aluno1_aluno2_aluno3.zip"
3. Arquivos corrompidos são de inteira responsabilidade do aluno, o aluno deve verificar os arquivos que foram enviados. O professor irá corrigir os arquivos que foram submetidos nas entregas.
4. Se não for enviado o código-fonte quando solicitado, o grupo automaticamente receberá nota zero sobre aquela atividade.
5. Casos de flagrantes de cópias de trabalho ocasionarão em nota zero para todos os trabalhos semelhantes.
6. Os grupos poderão ser entrevistados pelo professor individualmente, de acordo com a necessidade, em data e horários a serem definidos.
7. A indentação de código, comentários e qualidade da solução farão parte da avaliação do projeto.
8. A apresentação do projeto também fará parte da avaliação do projeto.
9. Os alunos do grupos serão avaliados de forma individualmente.
10. O engajamento dos alunos no projeto ponderarão as notas individuais e do grupo.