

	Curso: Téc. em Desenvolvimento de Sistemas	
	Disciplina: Programação Web Back-End	Turma: 366 (2025.1)
	Professor(a): Bruno Leal	Data:
	Aluno(a):	Matrícula:

Avaliação Prática 02

Sistema de Controle de Investimentos em Ações (Bolsa de Valores do Brasil)

Objetivo do Projeto

Desenvolver uma aplicação web completa utilizando **Node.js com o framework Express** e a engine de templates **EJS**, seguindo o padrão de arquitetura **MVC (Model-View-Controller)**. A aplicação deve permitir o controle de usuários e o gerenciamento de suas operações de investimento em ativos da bolsa de valores.

Funcionalidades Requeridas

Módulo de Usuários

- Cadastro de usuários com validação de dados (nome, email, senha).
- Login com autenticação de sessão e criptografia da senha.
- Logout.
- Middleware para proteger rotas privadas (somente usuários logados podem acessar suas operações).
- Cada usuário só pode visualizar, editar ou excluir **suas próprias operações**.

Módulo de Operações de Investimento

Cada operação deve conter os seguintes campos:

- Data da operação
- Tipo de operação: compra ou venda
- Código do ativo
 - deve seguir o padrão de quatro letras maiúsculas seguidas de 1 ou 2 números.
 - Ex: ITSA4, BBSE3, BCFF11, GOGL34.

- Quantidade
- Preço unitário
- Valor bruto (calculado)
- Taxa B3 (calculado)
 - 0.03% do valor bruto da operação
- Valor líquido (calculado)
 - operação de compra: valor bruto + taxas
 - operação de venda: valor bruto - taxas

Funcionalidades:

- CRUD completo (Create, Read, Update, Delete) para operações.
- Visualização de todas as operações do usuário.
- Visualização de todas as operações do usuário de um ativo específico (pelo código do ativo)
- Cálculo automático do valor bruto e valor líquido na criação e edição de operações.

Arquitetura e Requisitos Técnicos

Padrão MVC

- Models: Representam os dados da aplicação e regras de negócio (ex: Usuário, Operação).
- Views: Arquivos `.ejs` com layouts reutilizáveis e templates dinâmicos.
- Controllers: Lógica de controle entre os modelos e as views.
- Routers: Separação clara de rotas para usuários e operações.

Tecnologias e Ferramentas

- Node.js
- Express.js
- EJS (layouts com `include` e `partial`)
- Express-session para controle de sessão (login/logout) no banco de dados
- Banco de dados (PostgreSQL ou outro banco de dados relacional)
- Obs.: **não** pode utilizar frameworks ORM como Sequelize ou Knex