

# Desenvolvido por

- Breno F Pinho - TIA: 41932110
- Bruno N Santiago - TIA: 41933613
- Guilherme B Pereira - TIA: 32060785

DISCIPLINA: Computação distribuída

TURMA: 6N

## Projeto

### Servidor de Dados

- Os parâmetros são enviados ao servidor de dados pela *query string* da requisição. Exemplo `/getSaldo?id_negoc=1&conta=2`
- A falha na verificação do token aborta a requisição com status `401` (Não autorizado).
- Os parâmetros de cada requisição são validados e existe um código de erro para cada falha conforme a tabela abaixo:

constante	valor	descrição
LOCKED_ACCOUNT	-1	Conta bloqueada para escrita
INVALID_ACCOUNT	-2	Nenhuma conta foi passada na requisição
INVALID_VALUE	-3	Nenhum valor ou um valor inválido foi passado na requisição
NOT_FOUND	-4	A conta passada na requisição não existe no banco de dados

### Servidor de Negócio

- Os parâmetros são enviados ao servidor pela url da requisição, conforme enunciado.
- O valor `id_negoc` passado ao servidor de dados tem o formato a seguir: `IP : PORTA`.
- Caso haja algum erro na requisição, a operação é abortada com um código HTTP conforme a tabela abaixo:

código	descrição	motivo
401	Não autorizado	O token não é válido ou não foi informado
403	Proibido	A conta de origem/destino estão bloqueadas para escrita
404	Não encontrado	A conta não foi encontrada ou o valor é inválido

### Cliente

- O cliente pode ser inicializado com um token e utiliza um token padrão caso nenhum seja passado.
- O cliente é um objeto com quatro métodos públicos: `deposito`, `saque`, `saldo` e `transferencia`.
- Foi criado uma classe `ClientErr` para tratamento de erros
- O cliente tem tratamento de erros e todas as operações retornam uma *string* no formato abaixo:

```
body: <CORPO RESPOSTA>, status: <STATUS RESPOSTA>
```

### Script de testes

- O script de testes inicializa um cliente utilizando tokens válidos e inválidos e tenta fazer operações diversas em contas existentes e não existentes.
- Todo o resultado é tratado e exibido em tela.
- O script de execução contém uma lista de tokens válidos.

- O script de execução é configurado com as constantes abaixo:

```
# quantidade de contas no servidor de dados
```

```
NUM_CONTAS = 10
```

```
# endereco dos servidores de negocio
```

```
BUSINESS_SERVERS = ["http://localhost:5001", "http://localhost:5002", "http://localhost:5003"]
```

```
# endereco de autenticao
```

```
AUTHENTICATION_URL = "http://localhost:5000/autentica"
```