

## **DESAFIO 02 – Desenvolvendo uma aplicação Backend de apoio à desabrigados de enchentes**

O projeto consiste no desenvolvimento de uma estrutura para Backend que compreenda em uma aplicação para auxílio à desabrigados das enchentes, utilizando as tecnologias e conhecimentos aprendidos no curso.

### **Entrega**

O desafio será em individual e o código deverá ser disponibilizado no GitHub.

### **Envio do e-mail**

Crie um e-mail com o seguinte assunto:

- [Prog Bolsas MAI/24] Desafio 2 – Aplicação Backend usando Java

Exemplo de texto para o corpo do e-mail:

Olá instrutores,

Estou enviando o link Repositório do desafio 2 do Programa de Bolsas Backend (Spring Boot), como solicitado.

Link do repositório: [link do repositório no GitHub]

Envie o e-mail para os seguintes destinatários:

Para:

- [franciele.ciostek@compasso.com.br](mailto:franciele.ciostek@compasso.com.br)
- [diego.bonetti@compasso.com.br](mailto:diego.bonetti@compasso.com.br)
- [angelo.fabris@compasso.com.br](mailto:angelo.fabris@compasso.com.br)
- [gladson.ramos@compasso.com.br](mailto:gladson.ramos@compasso.com.br)
- [gabryel.melo@compasso.com.br](mailto:gabryel.melo@compasso.com.br)

## Apresentação

As apresentações serão realizadas a partir do dia 15/07/2024 com sorteio da ordem de apresentação a ser realizada de forma antecipada em horário ao que seria a Daily, sendo importante a presença de todos os colegas.

Para a apresentação será necessário explicar o código e apresentar a aplicação funcionando. O tempo para as apresentações será de 15 a 20 minutos. Caso ultrapasse o tempo delimitado, será descontado nota da apresentação

## Avaliação

Serão geradas duas notas avaliativas:

- *Nota da Implementação*: Os Instrutores avaliarão o desenvolvimento das telas a utilização dos itens citados nas Regras de Aplicação e o uso de boas práticas de programação aprendidos durante os cursos.
- *Nota da Apresentação*: Nessa nota iremos avaliar o domínio com relação ao código a ser apresentado e a postura durante a apresentação.

## Agenda

Data	O que faremos	Observação
01/07/2024	Início do Desafio	Lançamento do Desafio durante a Daily.
01/07 a 12/07	Daily com Tira Dúvidas	A reunião da Daily será reservada para consulta de dúvidas
11/07/2024	Sorteio da ordem de Apresentações	Serão sorteadas as ordens de Apresentação. Para quem quiser apresentar primeiro será dada prioridade.
12/07/2024	Prazo para envio do Link do Github aos mentores.	Prazo final para envio do Link do Github
15/07/2024	Último dia para edição no Github e data final do Desafio.	<b>O prazo será até as 10:00 horas. Após isso, alterações serão descontadas da Nota da Implementação</b>
15/07/2024	Início das Apresentações conforme sorteio realizado.	As apresentações serão realizadas das 10:30h às 11:30h na data e nas seguintes até que todos os bolsistas realizem suas apresentações;

## Regras da Aplicação

### Requisitos do Sistema

- **Tecnologias:** Java 17
- **Framework:** Nesse momento não será utilizado Spring Boot;
- **Banco de dados:** De livre escolha.
- **Boas práticas:** O código deve ser desenvolvido seguindo as boas práticas de desenvolvimento de software, para garantir a qualidade e a manutenibilidade.
- **Versionamento:** o código fonte deve estar no repositório do GitHub.

### Visão Geral da Aplicação:

O projeto é uma aplicação de backend para auxiliar desabrigados em enchentes. Ele controla entradas e saídas, calcula necessidades de alimentos com base em perfis (crianças, adolescentes, adultos e idosos) e mostra a duração do estoque. Além disso, calcula a previsão de estoque com base no consumo por faixa etária, fornecendo dietas diárias específicas para cada grupo. Os requisitos incluem algoritmos de cálculo de estoque com base em padrões de consumo e sua integração na lógica de controle de estoque da aplicação.

### Features da Aplicação:

#### 1. Registro de Doações

- Funcionalidade de cadastro, leitura, edição e exclusão de itens doados.
- Cada lote de itens cadastrados será direcionado a um centro de distribuição específico, que deve ser indicado no momento do cadastro. Existem 3 centros de distribuição, listados ao final deste documento.
- Os itens obrigatórios são:
  - Roupas:
    - Agasalhos (M/F) (Infantil/PP/P/M/G/GG)
    - Camisas (M/F) (Infantil/PP/P/M/G/GG)
  - Produtos de higiene:
    - Sabonete
    - Escova de dentes
    - Pasta de dentes
    - Absorventes
  - Alimentos
    - Arroz
    - Feijão
    - Leite
- Cada centro de distribuição tem a capacidade de armazenar 1000 itens de cada tipo (independente de tamanho), o sistema deverá acusar quando o limite de um centro de distribuição for atingido, impedindo de adicionar o excedente.

- **Opcional**: expandir a lista básica de itens doados.

## 2. Registro de Abrigos

- Registro de abrigos aptos a receber as doações: cadastro, leitura, edição e exclusão de abrigos.
- Deverá ser armazenado o nome, endereço, responsável, telefone e email de contato, capacidade e ocupação (%) de cada abrigo.
- Todos os dados inseridos devem passar por uma validação.
- Ao buscar um abrigo, além dos dados acima, deve-se listar a quantidade de cada item recebido pelo abrigo.
- **Opcional**: implementar filtros para consultar abrigos por capacidade, ocupação e quantidade recebida de cada item.

## 3. Ordem de Pedido

- Os abrigos listarão suas necessidades (item e atributos), e a aplicação indicará em quais centros de distribuição os itens podem ser encontrados.
- A listagem será ordenada pelo centro que dispõe da quantidade solicitada. Por exemplo, se 100 escovas de dentes forem solicitadas e o centro A tiver 80, o B tiver 450 e o C tiver 50, a ordem será B, A, C. Se nenhum centro tiver a quantidade total, o primeiro da lista será o que tiver a quantidade mais próxima da solicitada.
- O abrigo poderá enviar uma ordem de pedido para um ou mais centros de distribuição, conforme a necessidade.
- **Opcional**: listar os centros de distribuição por ordem de proximidade ao abrigo solicitante.

## 4. Checkout de Itens

- Cada centro de distribuição terá uma lista de pedidos para aceitar ou recusar.
- Se o pedido for aceito, o centro deverá deduzir do total disponível a quantidade de cada item enviado.
- Em caso de recusa, o motivo deverá ser informado.
- É necessário armazenar a quantidade de itens enviados para cada abrigo.
- **Opcional**: dar prioridade na listagem de pedido para abrigos que estão há mais tempo sem receber doações.

## 5. Transferência de doações

- Transferência de itens entre centros de distribuição ou devolução de itens para um abrigo em caso de excedente, atualizando os totais de cada item em cada local.

- Cada centro de distribuição tem um limite de 1000 itens de cada, e cada abrigo tem um limite de 200 itens de cada.
- **Opcional**: criar alertas de excedente (quando a quantidade ultrapassar 900) e de escassez (quando a quantidade for inferior a 100) para facilitar as transferências.

### **Centros de Distribuição:**

Os centros de distribuição serão:

- Centro de Distribuição Esperança. Av. Boqueirão, 2450 - Igara, Canoas - RS, 92032-420
- Centro de Distribuição Prosperidade. Av. Borges de Medeiros 1501 Porto Alegre CEP: 90119900
- Centro de Distribuição Reconstrução. R. Dr. Décio Martins Costa, 312 - Vila Eunice Nova, Cachoeirinha - RS, 94920-170

### **Bônus:**

#### **Visão Geral da Aplicação:**

O projeto é uma aplicação de Backend para auxiliar desabrigados em enchentes. Ele controla entradas e saídas, calcula necessidades de alimentos com base em perfis (crianças, adolescentes, adultos e idosos) e mostra a duração do estoque. Além disso, calcula a previsão de estoque com base no consumo por faixa etária, fornecendo dietas diárias específicas para cada grupo. Os requisitos incluem algoritmos de cálculo de estoque com base em padrões de consumo e sua integração na lógica de controle de estoque da aplicação.

#### **1. Controle de Entrada e Saída de Pessoas Desabrigadas**

- A aplicação deve ser capaz de registrar a entrada e saída de pessoas desabrigadas em um abrigo. Isso inclui a data e hora da entrada e saída, bem como detalhes pessoais como nome, idade e sexo.
2. Visualização da População do Abrigo:
- A aplicação deve ser capaz de mostrar a população atualizada do abrigo.
  - A população deve ser exibida com agrupamento por faixa etária (crianças até 3 anos, crianças de 4 a 8 anos, crianças de 9 a 12 anos, adolescentes de 13 a 18 anos, adultos de 19 a 59 anos, idosos a partir de 60 anos) e para cada faixa etária o sexo.

### 3. Controle de Saída de Estoque

- A aplicação deve ser capaz de mostrar o estoque global de 4 produtos não-perecíveis: arroz, feijão, leite em pó e café em pó. E o usuário deve ser capaz de remover produtos do estoque pela aplicação.

### 4. Cálculo da Necessidade de Alimentos:

- A aplicação deve ser capaz de calcular a necessidade de alimentos básicos com base na quantidade de pessoas no abrigo e no perfil de cada pessoa (crianças, adolescentes, adultos e idosos) e o sexo. Isso deve ser feito considerando a seguinte dieta diária:
  - Crianças (até 3 anos): Leite em pó: 50g, Arroz: 35g, Feijão: 15g
  - Crianças (4 a 8 anos): Leite em pó: 50g, Arroz: 50g, Feijão: 25g
  - Crianças (9 a 12 anos): Leite em pó: 50g, Arroz: 66g, Feijão: 33g
  - Mulheres Adolescentes (13 a 18 anos): Leite em pó: 50g, Arroz: 83g, Feijão: 42g, Pó de café: 20g
  - Homens Adolescentes (13 a 18 anos): Leite em pó: 50g, Arroz: 100g, Feijão: 50g, Pó de café: 20g
  - Mulheres Adultas (19 a 59 anos): Leite em pó: 50g, Arroz: 83g, Feijão: 42g, Pó de café: 30g
  - Homens Adultos (19 a 59 anos): Leite em pó: 50g, Arroz: 100g, Feijão: 50g, Pó de café: 30g
  - Idosos (a partir de 60 anos): Leite em pó: 50g, Arroz: 66g, Feijão: 33g, Pó de café: 20g

### 5. Previsão de Estoque:

- A aplicação deve ser capaz de calcular a duração do estoque atual com base no consumo diário de alimentos por pessoa. Isso deve ser feito considerando a quantidade de pessoas dentro do abrigo e a dieta diária de cada pessoa.

### Dicas:

Ler arquivo CSV em Java: <https://www.youtube.com/watch?v=xLDViuYlqGM&t=1195s>

Ler arquivo CSV em Java com Biblioteca OpenCSV:  
<https://www.youtube.com/watch?v=Y7UviY05U78&t=338s>