

Aplicação para Gestão de Entradas em Armazém no Banco Alimentar Contra a Fome

Proposta de Projeto

Orientador: Pedro R. M. Inácio(inacio@di.ubi.pt)

Coorientador: Alexandre Soares(alex.mp.soares@gmail.com)

Objetivos

O objetivo principal deste projeto é desenvolver uma aplicação de gestão de entradas para o Banco Alimentar contra a Fome da Cova da Beira (BACB). O BACB costuma fazer várias campanhas anualmente, tipicamente em fins de semana, que concretizam momentos de grande azáfama e em que recebem contributos de várias lojas. O funcionamento do BACB depende da participação voluntária de muitas pessoas que desempenham várias tarefas nesses dias. Por exemplo, algumas dessas pessoas são os condutores designados e dirigem-se de viatura às lojas para recolher contributos. Outras pessoas recebem esses contributos no armazém, fazendo a triagem e também o registo do que é recebido.

De forma a melhor gerir as entradas dos contributos, o BACB desenvolveu uma folha de cálculo [2] que pode ser usada por vários operadores e que funciona bem, a menos que algum dos operadores apague alguma fórmula, ainda que por engano. Assim, configura-se ideal desenvolver uma aplicação que, partindo do trabalho que já foi feito, permita que qualquer operador insira valores sem risco de cometer erros de eliminação de fórmulas. Deve permitir também que um administrador, com credenciais próprias aceda para alterações/correções de valores. A aplicação deve ainda oferecer ao administrador uma série de perspetivas integradas com listagens por loja, contributos, cadeias comerciais e concelhos.

Pelo que acima foi dito, este projeto incorpora ainda o desenho de um modelo de dados e desenvolvimento de uma base de dados de suporte adequado. A interface de administrador deve permitir ainda a configuração de vários parâmetros e a edição desse modelo de dados. Espera-se que também se possam exportar alguns dados tratados ou produzidos na aplicação para um formato legível por folhas de cálculo (e.g., Excel).

Durante a execução do projeto (na primeira fase) deve ser feito um levantamento de requisitos mais detalhado do que o que se coloca aqui, mas identificam-se já alguns dados com que a deve lidar. Para o operador:

1. deve ser possível escolher a proveniência do contributo que está a chegar a armazém;
2. ao dar entrada de um contributo, o sistema automaticamente grava, com os restantes dados, a data e a hora;
3. após escolher a proveniência, o operador pode introduzir o peso por categoria ou produto, numa perspetiva com uma lista de *items* intuitiva;
4. pode ainda fazer consulta dos totais relativos a dados atuais;

Para o administrador:

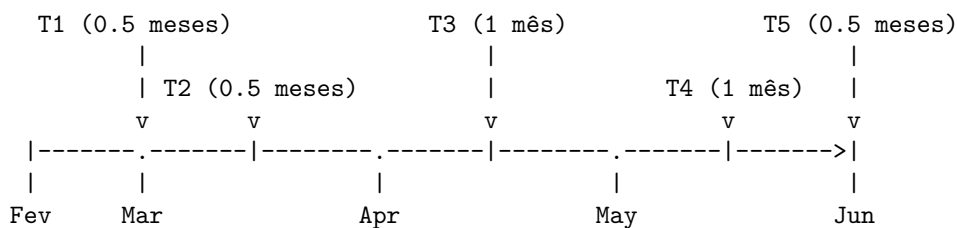
1. o acesso administrador requer um código de autenticação e é feito a partir de um link modesto na primeira perspetiva da aplicação;
2. o administrador pode alterar vários aspetos da base de dados, nomeadamente introduzir, editar ou eliminar lojas, concelhos e produtos;
3. o administrador pode tonificar/ajustar vários parâmetros da aplicação, nomeadamente número de instituições e número de entregas;
4. o administrador pode fazer correções a dados de entrada de contributos feitos por operadores;
5. o administrador tem acesso a perspetivas integradas de listagens completas por loja, concelho e cadeias comerciais.

Todas as *interfaces* devem ser o mais simples possível. O acesso *operador normal* não precisa palavra-passe, enquanto que a de administrador precisa. A aplicação pode ser desenvolvida usando tecnologias web (HTML5+CSS+PHP+Apache) em modo servidor local, embora talvez seja mais adequada a implementação em Python ou Java com uma biblioteca de interface. A aplicação deve ser desenhada para Windows. A documentação deve ser rica e uniforme e devem ser usados repositórios de código para manutenção e gestão de versões (e.g., github). Deve apontar-se para fluxos de utilização intuitivos e inclusão de mensagens simples sobre como funcionam as várias partes da aplicação. A aplicação deve ter em consideração aspetos de desempenho.

Dada a sua natureza, este projeto requer conhecimentos sólidos em Programação e Base de Dados. O(a) aluno(a) terá uma oportunidade de solidificar o seu conhecimento nas várias áreas abrangidas por este projeto.

Tarefas a Realizar e Cronologia

- T1** Contextualização com os objetivos propostas e preparação do ambiente de trabalho (0,5 meses);
- T2** Análise detalhada dos requisitos; desenho do modelo e da base de dados (0,5 meses);
- T3** Implementação do protótipo da aplicação (1 mês);
- T4** Testes e aprimoramentos (1 mês);
- T5** Escrita do relatório de projeto (0,5 meses).



Requisitos Técnicos / Académicos

Ter boas classificações e conhecimentos em programação e bases de dados.

Elementos de Avaliação a Entregar

Para além do relatório, o(a) estudante deverá entregar todo o código fonte desenvolvido, bem documentado, bem como um pacote de instalação da aplicação (se pertinente).

Resultados Esperados

- * O desenho da base de dados de suporte à aplicação;
- * Um protótipo totalmente funcional e testado da aplicação;
- * 1 relatório de projeto [1].

Referências Bibliográficas

[1] C. Collberg and S. Kobourov, Self-plagiarism in Computer Science, Communications of the ACM, 48(4): 88 - 94, 2005.

[2] Paulo Pinheiro, Folha de Cálculo de Campanhas BACB, 2019.