

## Requisitos do projeto final

O enunciado do problema se encontra no pdf *Ementa* postado no sigaa no primeiro dia do trabalho. No entanto segue um resumo.

Deve-se desenvolver um sistema web (não importa qual linguagem utilizada) chamado de EasyMarket (EM). O EM é um sistema para o controle e gerenciamento de uma padaria que funciona também como restaurante “à la carte”.

O sistema tem 3 tipos de usuários:

- 1- Funcionário Comum
- 2- Gerente
- 3- Dono do estabelecimento

As permissões de cada tipo de funcionário. Por exemplo, o dono do estabelecimento pode fazer tudo que o funcionário comum e o gerente pode fazer mais alguma outra tarefa. O gerente não pode fazer tudo que o dono do estabelecimento pode fazer, mais pode fazer tudo que o funcionário comum pode fazer mais alguma tarefa.

Nesse sistema, cada cliente recebe um IDentificador (código de barras por exemplo) quando entra na loja. A medida que o cliente consome algo no restaurante, é armazenado esse “algo” como um consumo em sua loja. Quando o cliente passar no caixa, os produtos consumidos serão somados com os produtos em seu carrinho de compra (da padaria em si). Quando o cliente finaliza a compra o ID é liberado (ou seja, é zerado as compras e consumos desse ID) para que possa ser usado por um novo cliente.

Outro ponto importante é que o funcionário comum pode assumir diferentes papéis (garçom, cozinheiro, zelador, caixa). Porém todos os funcionários podem executar essas tarefas. Não existe restrição.

Em relação ao restaurante, o sistema deve permitir que se registre pedidos (de um cardápio já registrado) e que outro funcionário na cozinha possa pegar esses pedidos (na ordem que foram feitos) para preparar e produzi-los.

O sistema deve permitir as seguintes funcionalidades (Funcionalidade → tipo de usuário que pode usar a funcionalidade)

- CRUD de funcionários → b
- CRUD de produtos na padaria e cardápio → b
- “Lista de chamada” para ponto dos funcionários registrando o dia e a hora que funcionário chegou e saiu → b
- Realizar pedidos no restaurante → a
- Listar pedidos feitos no restaurante em sequência → a
- Gerenciar autenticação dos usuários
- Gerenciar o uso de catracas para controle de entrada e saída do estabelecimento, liberando a passagem do cliente caso ele tenha pago sua conta (ou seja, ID zerado)
- Exibição de relatórios de vendas para que o gerente e o dono do comércio avalie o desempenho financeiro do estabelecimento → b

Legenda: a → só dono do estabelecimento

b → gerente e dono do estabelecimento

c → gerente, dono do estabelecimento e funcionário

De resto, imaginem o cenário e tirem as dúvidas com o cliente que vai comprar o sistema de vocês. No caso, o professor da disciplina é o cliente e vocês são os programadores que desenvolvem o sistema.

Consultar o cliente é essencial para que ele descreva todas funcionalidades que ele deseja no programa, conforme os métodos SCRUM e XP.

### Requisitos de documentação

- 1- O Trabalho deve conter o diagrama de classes de todo o sistema
- 2- O diagrama relacional de todo banco de dados
- 3- Diagrama de pacotes de todo o sistema
- 4- 2 diagramas de objeto em ocasiões distintas escolhidas pelo grupo
- 5- O diagrama de caso de uso dos principais casos de uso (faz parte do trabalho, vocês definirão os casos de uso importantes). Deve haver o diagrama de sequência do Login e do gerenciamento da realização de pedidos (ou seja, o fluxo que mostre a realização do pedido no restaurante, passando pela sua produção até a entrega desse pedido na mesa)
- 6- 2 Diagramas de Máquina de Estado (fica a cargo do grupo escolher o que representar)
- 7- 1 diagrama de Atividade (fica a cargo do grupo escolher o que representar)