

Modelo de projeto do Sistema Pizza.Net

Marcus Vinicius Marchiori Ribeiro - 11811BSI224
Matheus José dos Reis – 11811BCC027
Matheus Humberto G. F. Costas - 11811BCC006
Nayara Terezinha Nunes – 11911BCC006
Thiago Antônio Moreira -11411BSI235
Victor Hugo Marra Araújo - 11721BCC041

Uberlândia – MG
Setembro de 2020

1. Sistema PizzaNet

O sistema permitirá a realização, gerenciamento e entrega de pedidos da Pizzaria Entrega Rápida, o sistema poderá ser acessado pela web ou por aplicativo para dispositivos móveis. Possui fácil utilização, devido ao público alvo possuir pouco conhecimento técnico em informática. A ferramenta não visa substituir profissionais, mas sim facilitar o trabalho dos funcionários e aumentar a eficiência do serviço prestado.

2. Tecnologias

Para o desenvolvimento do sistema serão utilizadas as seguintes especificações das tecnologias:

- O back-end do sistema será hospedado em provedor que mantém 96% de disponibilidade como a AWS, Azure e etc;
- O back-end será feito em Java com Spring Boot;
- Persistência do banco Spring Data, JPA e Hibernate;
- Projeto de padrão REST;
- Arquitetura MVC;
- O banco de dados será o PostgreSQL;
- O app será feito em Dart com o framework Flutter;
- A documentação do sistema deve seguir o padrão Swagger;
- Metodologia orientada a objetos.

3. Divisão dos pacotes seguindo MVC

Os pacotes de view irão conter as classes e projetos do front-end/web/mobile, todas as classes que serão expostas e apresentadas para os usuários do sistema;

Os pacotes de model irão conter a camada de serviço e modelagem, regras de negócios e cumprimento dos requisitos funcionais;

Os pacotes de controller, irão conter os web containers e JSP servlets de páginas HTML.

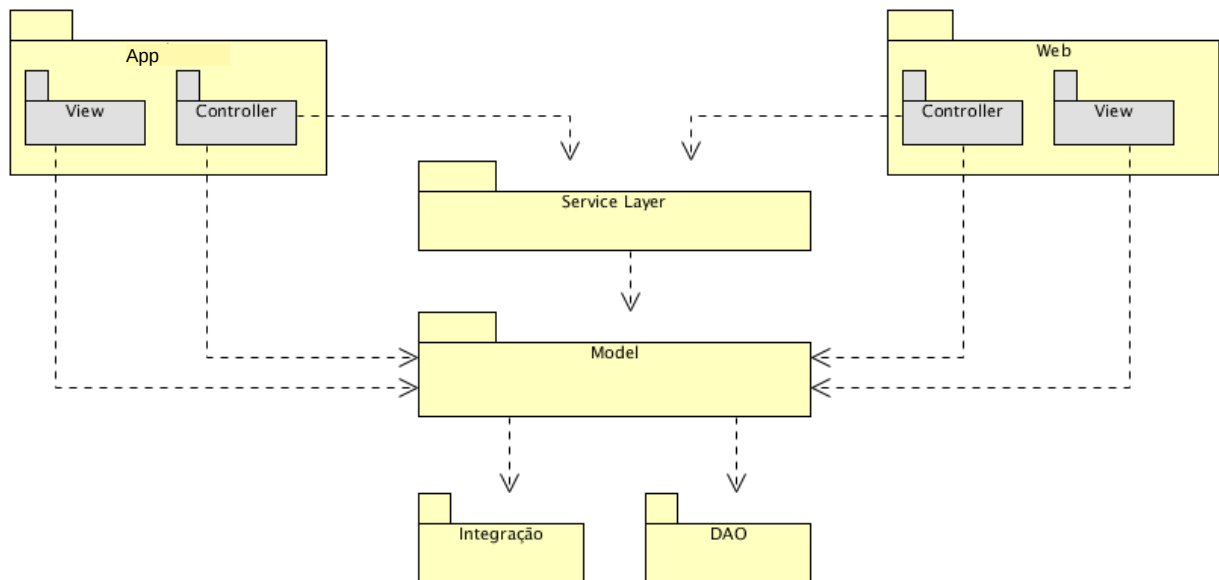


Figura 01 – Modelo Visualização Controle

4. Descrição das camadas

Na documentação da arquitetura foi definido que para o PizzaNet será usado o modelo MVC, a seguir encontram-se as especificações e pacotes de cada camada. A figura 02 exemplifica o uso do MVC.

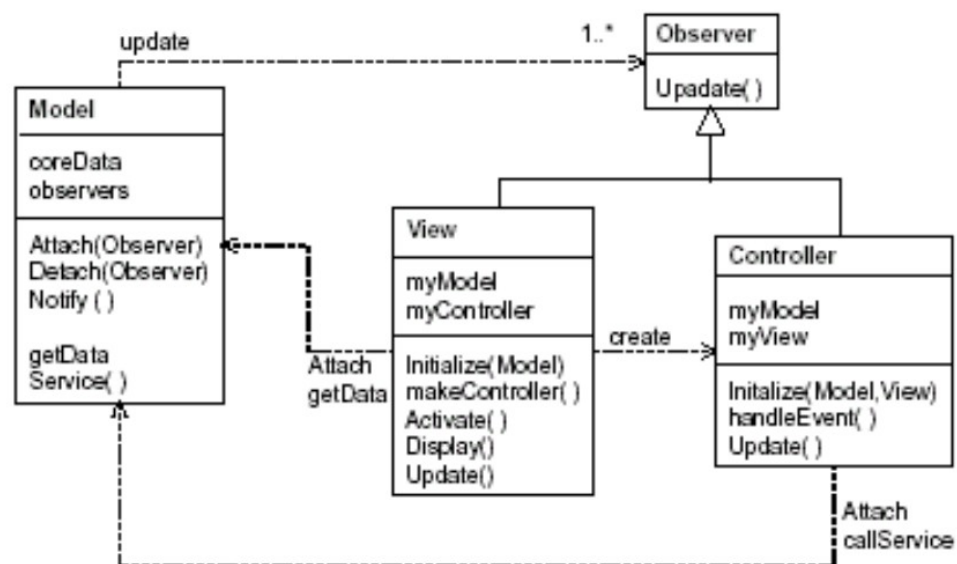


Figura 02 – Pacotes MVC

Uma breve descrição de como funciona o sistema usando esse tipo de modelo de projeto:

O atributo action da tag Form HTML irá apontar para um servlet que receberá as informações. O controlador vai ser implementado com Servlet, e irá receber os dados do cliente. Ele processará as informações e seleciona o visualizador

adequado à resposta. Quando for necessário a servlet usa o bean funcional para armazenar ou recuperar informações do banco de dados. Próximo passo, o servlet instancia os beans do modelo. O modelo, que será implementado com JavaBeans, é criado pelo controlador, que irá preencher seu conteúdo. O visualizador, que é implementado como JSP e selecionado pelo controlador, recebe os dados do cliente, acessa o modelo para extrair os dados e formata a página de saída para o usuário. Então, após todo esse processo, a página é enviada para o cliente. Separado dessa maneira pode-se alterar as telas ou componentes visuais do sistema sem modificar as classes responsáveis pela lógica da aplicação. O model é um projeto Spring Boot será feito em Java, e guardará todas as regras de negócio, exeções, persistência do banco com o Spring Data.

Como exemplo, usando esse processo, mostraremos como funciona para armazenar os cartões dos clientes que estão cadastrados no PizzaNet:

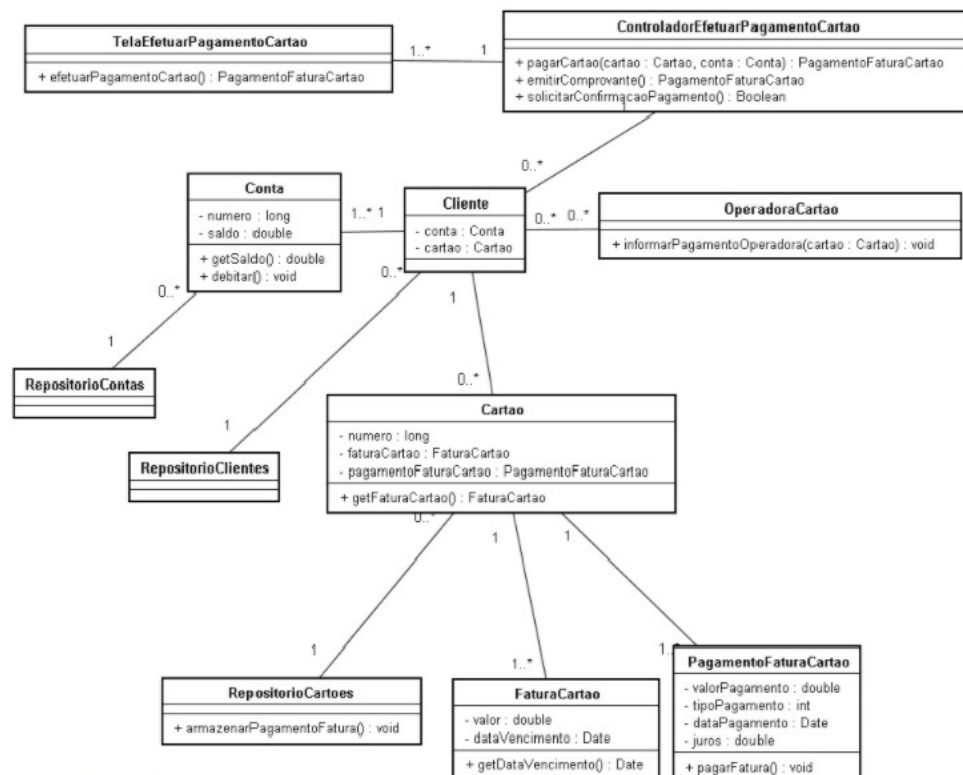


Figura 03 – Pagamento com cartão

A camada de View está relacionada com as telas (interfaces), que o cliente irá interagir. As Figuras 04, 05 06 e 07 mostram exemplos de interface web, no qual o cliente poderá logar no sistema, escolher seus pedidos, visualizar histórico, promoções, criar sua pizza, confirmar pedido e endereço da entrega, cancelar pedido e acompanhar pedido.

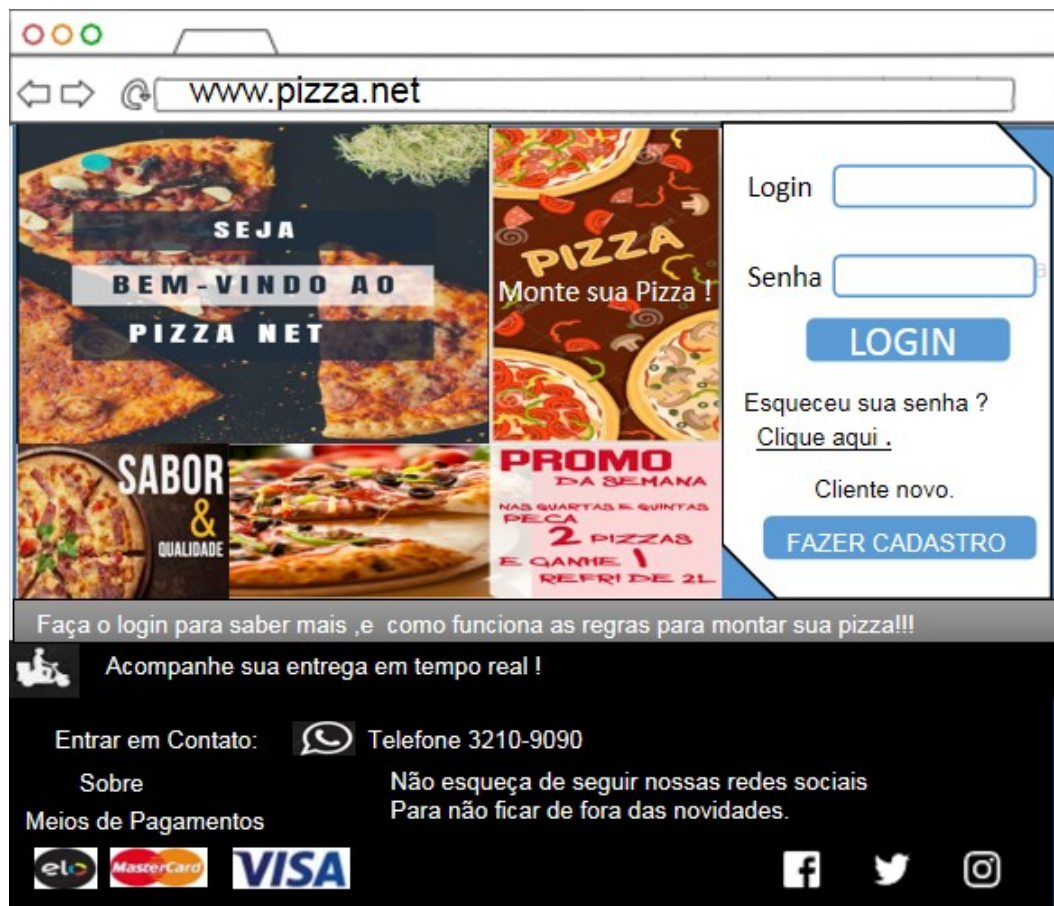


Figura 04 – Interface Login WEB



Figura 05 - Visualizar histórico, promoções e criar sua pizza

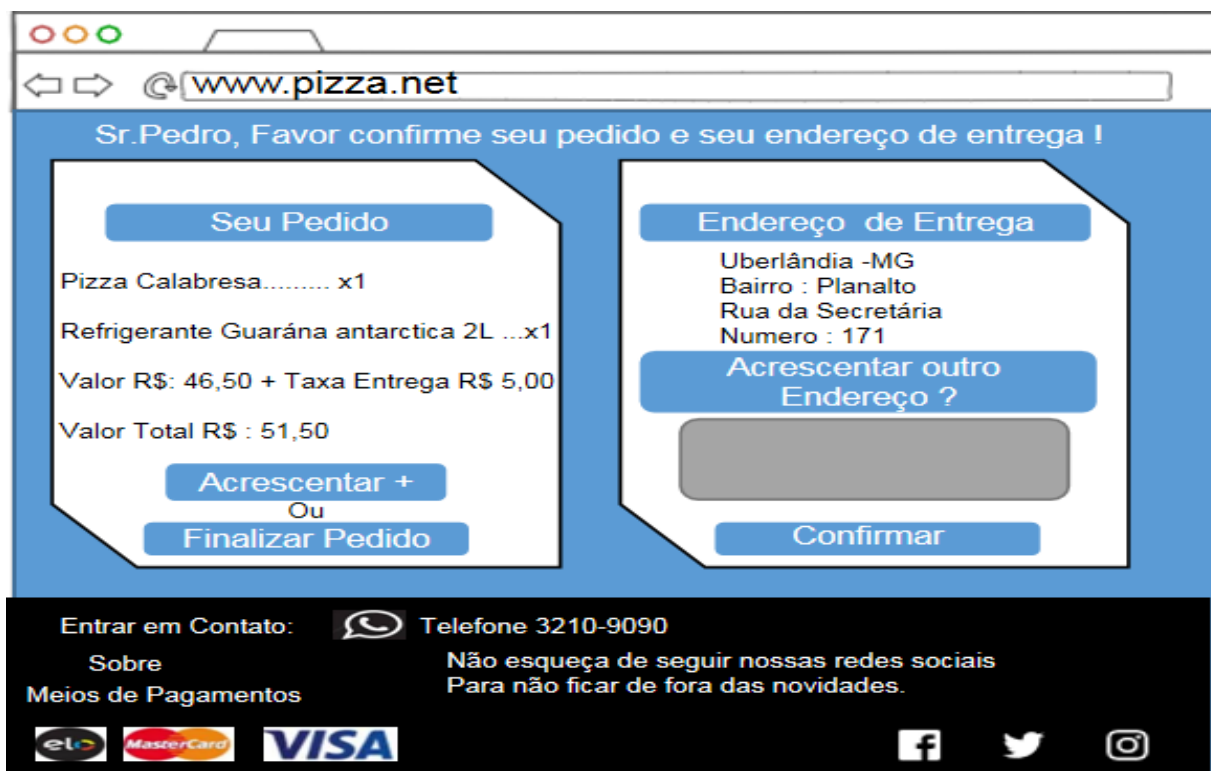


Figura 06 - Confirmação de pedido e confirmação de endereço.

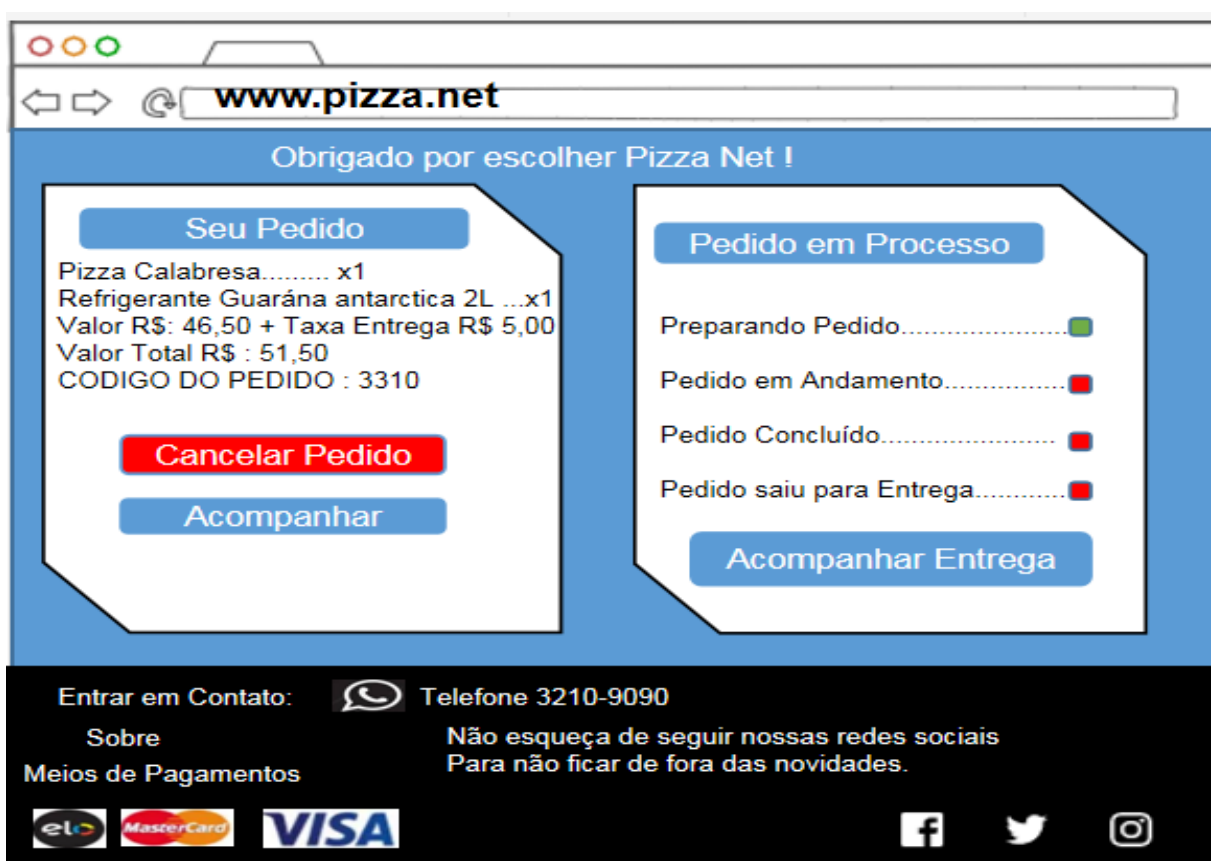


Figura 07 - Cancelar pedido e Acompanhar Pedido.