

# Sistema de gerenciamento de comandas e mesas

Gabriel de Mello Flores, Ricardo Frohlich da Silva

<sup>1</sup>Universidade Franciscana (UFN) - Santa Maria - RS - Brazil

gabriel.flores@ufn.edu.br, ricardo.frohlich@ufn.edu.br

**Abstract.** *This project aims to present the development of a web system to help in the organization of restaurants and bars, given the large number existing in Brazil and understanding that technology has always been one of the greatest allies of entrepreneurs. It still has a vast amount that does not have any type of management and control software over its business. In view of this, the objective is to design a system in order to manage more focused on the part of commands and tables, controlling the occupation, values, orders and other features. In order to streamline, automate, organize and make service more efficient, making customer satisfaction as high as possible.*

**Resumo.** *Este trabalho tem como objetivo apresentar o desenvolvimento de um sistema web para auxiliar na organização de restaurantes e bares, visto o grande número de estabelecimentos deste ramo existentes no Brasil. Desse modo, compreendendo que a tecnologia sempre foi uma das maiores aliadas dos empreendedores, ainda existe uma vasta quantidade que não possui nenhum tipo de software de gestão e controle sobre os seus negócios. Diante disso, foi projetado um sistema a fim de gerenciar as comandas e mesas, controlando a ocupação, valores, pedidos e entre outras funcionalidades. Portanto, com a finalidade de agilizar, automatizar, organizar e tornar o atendimento mais eficiente, fazendo com que a satisfação do cliente se torne a maior possível.*

## 1. Introdução

Com o grande número de restaurantes que existem no Brasil, segundo [Abrasel 2021], existem mais de 1 milhão de negócios no ramo de bares e restaurantes, representando atualmente 2,7% do PIB nacional. Somado a isso, o hábito de comer fora de casa é crescente no país e, no último levantamento feito pelo IBGE, correspondia a 31,1% de todo gasto dos brasileiros com alimentos.

Para que as empresas possam competir em um cenário global com crescimento sustentável, é necessário cada vez mais conhecimento em administração, para tomar decisões em termos de estratégia, governança, tecnologia, fusões e aquisições, entre outras importantes questões, o que impulsiona a demanda por conhecimento e aperfeiçoamento no uso das ferramentas de gestão [Rigby 2005].

Segundo [Lima et al. 2017], analisando os processos de um grande restaurante, nota-se alguns problemas, como atraso no atendimento ao cliente, pedidos anotados incorretamente, demora no fechamento das contas e entre outras ocorrências, ocasionados pelo sistema de comandas em papel. Todos esses fatores contribuem para o descontentamento do cliente e, conseqüentemente, a não fidelização e, até mesmo, a perda destes, o que não é nada interessante para o empreendedor e seu estabelecimento.

Desta forma, foi desenvolvido um sistema web para auxiliar no controle destes restaurantes, especificamente na parte de comandas e controle de mesas com intuito de agilizar, organizar e diminuir a chance do erro, tanto de pedidos, quanto de valores.

### **1.1. Objetivo Geral**

Este trabalho tem por objetivo apresentar o desenvolvimento de um sistema web para controle de comandas e mesas de restaurantes e bares, a fim de auxiliar no controle e organização do estabelecimento.

### **1.2. Objetivos Específicos**

Para este trabalho, os objetivos específicos são:

- Estudar modelos de sistemas para restaurantes, assim como modelos de comandas;
- Levantar os requisitos para a criação do sistema desenvolvido deste trabalho;
- Desenvolver um sistema web com a linguagem PHP e o *framework* Laravel;
- Utilizar o *framework* Bootstrap para criação do layout das páginas;
- Utilizar o banco de dados MySQL para controle dos dados.

## **2. Referencial teórico**

Nesta seção serão abordados conceitos pertinentes ao desenvolvimento deste trabalho.

### **2.1. Sistemas de informação**

Um sistema de informação é o resultado da utilização de computadores e de tecnologia de telecomunicações com intuito de transformar dados em informações que serão utilizadas para apoiar a decisão dos gestores. O uso do hardware (componente físico), do software (programas de computadores) e de recursos humanos pela área da informática são os responsáveis por dar forma a esse sistema. São eles que geram, armazenam, processam e transmitem informações, através de dados fornecidos, utilizado para a execução de dados, e, assim, auxiliam nos processos de tomada de decisões [Carlos 2015].

De acordo com [Audy et al. 2009], os sistemas de informação disponibilizam as informações que permitem à organização analisar problemas e oportunidades e criar novos produtos, serviços e formas de operação que propiciem a obtenção de vantagens competitivas. Uma vantagem competitiva pode ser considerada como a razão pela qual o cliente escolhe uma empresa determinada, e não a empresa concorrente.

Desse modo, evidencia-se que as empresas estão sempre tentando melhorar a eficiência de suas operações a fim de conseguir maior lucratividade. Das ferramentas de que os administradores dispõem, as tecnologias e os sistemas de informação estão entre as mais importantes para atingir altos níveis de eficiência e produtividade nas operações, especialmente quando combinadas com mudanças no comportamento da administração e nas práticas de negócio [Laudon and Laudon 2017].

### **2.2. Tecnologias**

Nesta subseção serão apresentados as tecnologias necessárias para o desenvolvimento deste trabalho.

### 2.2.1. PHP

O PHP (Hypertext Preprocessor) é uma linguagem de *script* livre de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML (HyperText Markup Language) [PHP 2022].

O PHP permite incorporar fragmentos de códigos em páginas de HTML, código esse que é interpretado à medida que suas páginas são carregados aos usuários. O PHP também serve como uma linguagem de que facilita a conexão de suas páginas Web com o banco de dados do lado servidor [Converse, Tim and Park, Joyce 2003].

### 2.3. Padrão MVC

O MVC (*Model View Controlle*) é uma aplicação que segue o *Design Pattern Model View Controller* que tem as suas classes separadas em três grandes grupos de responsabilidades. A intenção principal ao utilizarmos o *Design Pattern MVC* é não misturarmos em uma mesma classe responsabilidades diferentes [Dall'Oglio 2018]. Segundo [Carlos 2017] Suas classes são:

- *Model*: Todas as regras negociais do sistema juntamente com o acesso ao banco de dados para a realização de qualquer operação, estarão contidas nessa camada;
- *View*: A interface gráfica de comunicação entre o usuário e o software é situada nessa camada;
- *Control*: O controlador é responsável por gerenciar a comunicação (envio de requisições e dados) entre uma camada e outra.

Na Figura 1, é possível ver uma representação visual de um padrão MVC.

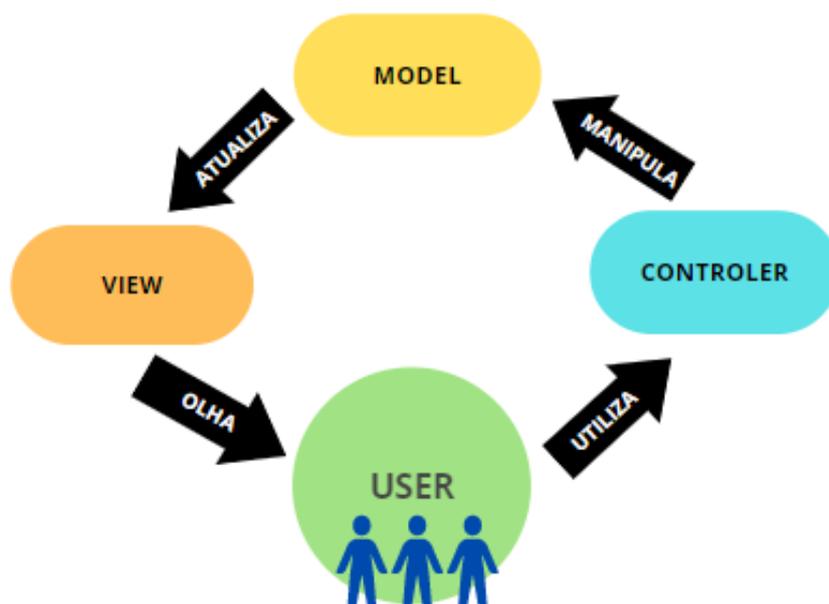


Figura 1. Representação MVC. [Olanrewaju et al. 2015]

### 2.3.1. Laravel

Laravel é um *framework* web PHP livre e de código aberto, destinado ao desenvolvimento de aplicações web seguindo o padrão arquitetural *Model View Controlle* (MVC) [Laravel 2022].

O Laravel possui um sistema modular gerenciador de dependências, vários modos de acesso a bancos de dados e diversos utilitários que ajudam no desenvolvimento do sistema. Enquanto o *view* do Laravel utiliza a *engine* do Blade Template que traz ferramentas para facilitar a escrita de um código HTML junto com dados dinâmicos de forma limpa e sem duplicação [Silva et al. 2017].

### 2.3.2. Bootstrap

O Bootstrap é um *framework* para desenvolvimento *front-end*, desenvolvido em CSS (*Cascading Style Sheet*) e Javascript, proporciona um design mais elegante as páginas HTML (*Hypertext Markup Language*) proporciona uma maior rapidez e praticidade ao desenvolvimento [Bootstrap 2022].

Segundo [Costa 2014], algumas das principais vantagens de se utilizar o Bootstrap são:

- Vasta documentação;
- Feito para trabalhar com layouts responsivos;
- Inúmeros componentes a disposição;
- Mantém padrões;
- Funciona em todos os navegadores atuais (Chrome, Safari, Firefox, IE, Opera).

### 2.3.3. MySQL

Segundo [MySQL 2022], o MySQL é um sistema de gerenciamento de banco de dados SQL de código aberto popular por seu desempenho e confiabilidade, tornou-se a principal escolha de banco de dados para aplicativos baseados na Web, é desenvolvido, distribuído e suportado pela Oracle Corporation. Sendo suas principais características:

- Bancos de dados relacionais;
- Código aberto;
- O MySQL Database Server é confiável e escalável;
- MySQL Server funciona em sistemas cliente/servidor ou embutidos;
- Uma grande quantidade de softwares MySQL contribuídos estão disponíveis.

## 2.4. FDD

De acordo com [Foggetti 2015], a FDD (*Feature-Driven Development*) foi criada em 1999 por Jeff de Luca, Peter Coad e Stephen Palmer, durante um desenvolvimento de um projeto. Foi definido um processo ágil com as características:

- Iteração;
- Qualidade;

- Resultados tangíveis e frequentes;
- Possuir relatórios de progresso;
- Apreciado por clientes, gerentes e desenvolvedores.

*Feature-Driven Development* é uma metodologia mista, pois possui alguns processos rigorosos, como modelagem, planejamento e controle do projeto, e características de projetos ágeis, como foco no código, interação constante com o cliente e entrega de versões do projeto [Foggetti 2015].

Desse modo, correlacionando com [Silva et al. 2009], o autor salienta que são cinco os processos da metodologia ágil FDD: Desenvolver um modelo abrangente, construir uma lista de funcionalidades, planejar através de funcionalidades, projetar através de funcionalidades e construir através de funcionalidades. Como pode ser visualizado na Figura 2

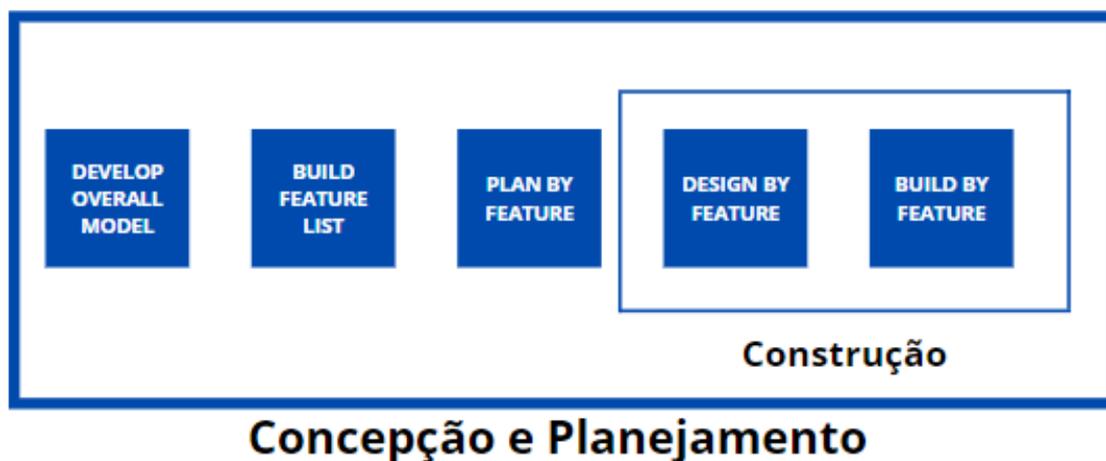


Figura 2. Metodologia ágil FDD

## 2.5. Trabalhos relacionados

Nesta seção serão apresentados trabalhos relacionados com temáticas e/ou as tecnologias semelhantes.

### 2.5.1. Sistema Web de Gerenciamento de Pedidos de Restaurantes

O trabalho desenvolvido por [Santos et al. 2021] tem como objetivo permitir o gerenciamento e controle dos pedidos, estoque e caixa para gerar sinergia e resultados positivos para a empresa.

O sistema foi desenvolvido para funcionar em ambiente web, sendo utilizado ferramentas para desenvolvimento web. Para o projeto foi realizado um PDV (Ponto de Venda) onde pode-se criar os pedidos e uma área de configuração onde pode-se criar categorias e incluir produtos no banco de dados.

Com este trabalho evidencia-se que é possível a utilização de ferramentas modernas de administração nos restaurantes e que este investimento só traz benefícios no atendimento e gerenciamento dos estabelecimentos [Santos et al. 2021].

### **2.5.2. Sistema de Gestão de Pedidos para Restaurantes e Lanchonetes**

O autor [Cruz 2015] apresentou um projeto que tem como objetivo automatizar o processo de pedido e entrega em restaurantes, lanchonetes e afins, visando diminuir as perdas por erros de anotações, entre outros problemas que resultam em uma longa espera dos clientes, causando constrangimentos e estresse.

Para a solução deste problema foi realizado um aplicativo Android para controle de comanda. Com foco de inovar o atendimento, proporcionar um serviço ágil, minimizando os erros e constrangimentos gerados pelo método manual de realizar o pedido.

Como resultado, o sistema desenvolvido promove aos seus usuários mais mobilidade, independência, precisão e agilidade em suas atividades. Com isso, os problemas citados como causa de transtorno e lentidão podem ser minimizados ao máximo. Tudo isso implica no bom andamento de uma pizzaria, restaurante ou lanchonete, pois a satisfação dos clientes é o principal fator para o lucro e crescimento deste tipo de empresa [Cruz 2015].

### **2.5.3. SmartFood: Sistema de Gerenciamento de Pedidos**

A agilidade no atendimento, sendo essencial para qualidade, é de suma importância para o consumidor, pretendendo isso [Gomes et al. 2018] realizou uma aplicação web com acesso local, voltada para estabelecimentos do ramo alimentício, a qual permite aos clientes acessar o cardápio, bem como realizar o pedido e pagar a comanda através de um dispositivo móvel direto da mesa.

Conforme exposto no referencial teórico, ainda há no setor alimentício espaço para introdução de tecnologias para automatizar o atendimento ao cliente, de forma a agilizar o processo. Desse modo, levantou-se uma série de sistemas com características semelhantes às do projeto, com a finalidade de verificar as soluções e tendências do mercado.

O sistema busca evitar possíveis atrasos causados pela falta de atenção de funcionários com a alta demanda de pedidos dos clientes. Desse modo, utilizar um sistema como o SmartFood, possivelmente, aumenta a velocidade do atendimento, o que ajuda reduzir a evasão de clientes [Gomes et al. 2018].

### **2.5.4. Considerações sobre os trabalhos relacionados**

O trabalho desenvolvido por [Santos et al. 2021] tem algumas semelhanças com a proposta deste trabalho referente a parte do controle de mesa e caixa para gerar sinergia e resultados positivos para o estabelecimento.

Já o trabalho realizado por [Cruz 2015] apresenta funcionalidades para o controle das comandas e mesas, sendo utilizadas para a aplicação móvel onde será aplicado neste trabalho como o formato de controle de mesas, o qual é possível verificar quais estão sendo ocupadas.

Segundo [Gomes et al. 2018] além de um sistema de controle de comanda agilizar o procedimento, também traz a abordagem de reduzir custos, uma vez que irá diminuir a

chance de erro na hora do pedido e anotação do que foi consumido.

Os trabalhos apresentados evidenciam importantes pontos para a estruturação da proposta deste trabalho, pois focaram em significativos itens para a criação de um sistema para auxiliar os estabelecimentos tanto na praticidade, velocidade e redução de custos.

### 3. Proposta

O propósito fundamental deste trabalho foi desenvolver um sistema web para controle de comandas e mesas de restaurantes, bares, etc. Decidiu-se focar nesta área sendo perceptível a importância de possibilitar controle, organização e praticidade dos atendimentos nesses estabelecimentos.

Primeiramente após o cliente ser alocado em uma mesa, ele realiza o pedido ao garçom, que por sua vez grava no sistema o pedido e a mesa ocupada. Na primeira parte da Figura 3 é possível ver uma ilustração deste processo.

Na segunda parte da Figura 3 representa o momento que é registrado o pedido no sistema, e logo após é informado para a cozinha a realização do preparo da comida.

Com o pedido preparado pela cozinha é informado ao garçom via sistema que está finalizado, desta forma o pedido é levado até a mesa informada. Na quarta parte da Figura 3 é possível visualizar o cliente fechando a conta, no qual o garçom informa ao sistema que a conta será encerrada, assim poderá ser realizado o pagamento via API.

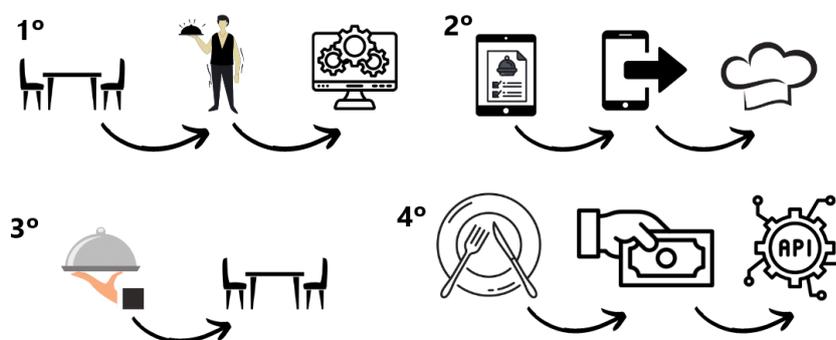


Figura 3. Processo do sistema

Além disso, para o desenvolvimento foi utilizado a linguagem PHP, o *framework* Bootstrap, *framework* Laravel e o banco de dados MySQL. Em relação ao planejamento do projeto foi utilizada a metodologia FDD e o modelo foi dividido em cinco etapas que são:

- Desenvolver um modelo geral;
- Construir uma lista de funcionalidades;
- Planejamento por funcionalidades;
- Projetar por funcionalidades;
- Construir por funcionalidades;

### 3.1. Desenvolver um modelo inicial

De acordo com [Foggetti 2015], a primeira etapa do modelo FDD é feita a criação de um modelo de classe e objetos de negócios. Também é realizada uma análise macro do escopo do sistema, onde nas próximas etapas este escopo vai sendo quebrado em partes para que possa ser detalhado.

O modelo de classe e objetos consiste no diagrama de domínio que pode ser visto na Figura 4. Este diagrama refere-se ao projeto a ser desenvolvido, sendo possível obter uma visão conceitual.

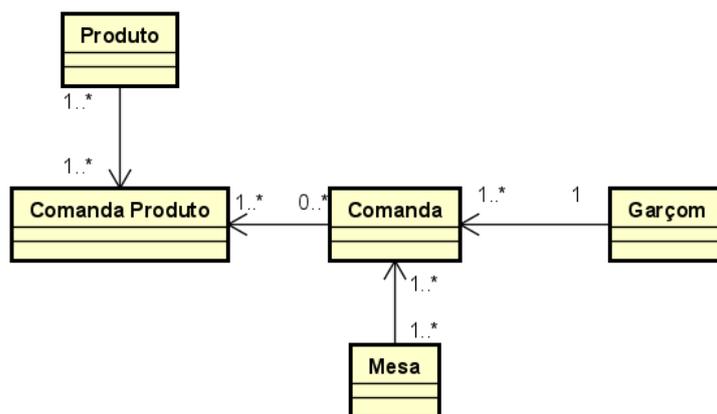


Figura 4. Diagrama de domínio

### 3.2. Construir uma lista de funcionalidades

Com base no conhecimento da primeira etapa, foi realizada a decomposição das funcionalidades do domínio em áreas de assunto, as atividades de negócios dentro delas e as etapas dentro de cada atividade de negócios, formando assim a lista de recursos categorizados [FDD 2022].

Dessa forma, foram construídas listas contendo todos os requisitos funcionais (RF) e requisitos não funcionais (RFN) que fazem parte da estrutura do sistema (Apêndice A).

### 3.3. Planejamento por funcionalidades

Segundo [Foggetti 2015], nesta etapa é definida a sequência de desenvolvimento com alguns critérios considerados:

- Dependências de funções;
- Carga de trabalho;
- Complexidade das funções;
- Priorização das funcionalidades de maior risco;
- Marcos externos ao projeto, como pontos de checagem.

Como pode ser visto na Tabela 1, é apresentado o tempo de desenvolvimento de cada funcionalidade que faz parte do sistema.

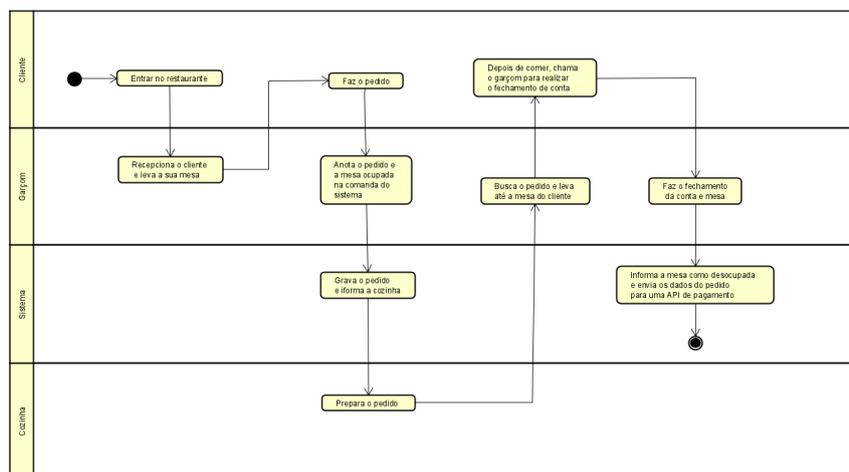
**Tabela 1. Planejamento por funcionalidade**

Sequência	Funcionalidade	Tempo Previsto
RF1	Gerenciar Login	20 Horas
RF2	Gerenciar Produtos	30 Horas
RF3	Gerenciar Pedidos	60 Horas
RF4	Gerenciar Usuários	20 Horas
RF5	Gerenciar Mesas	30 Horas
Total		160 Horas

### 3.4. Projetar por funcionalidade

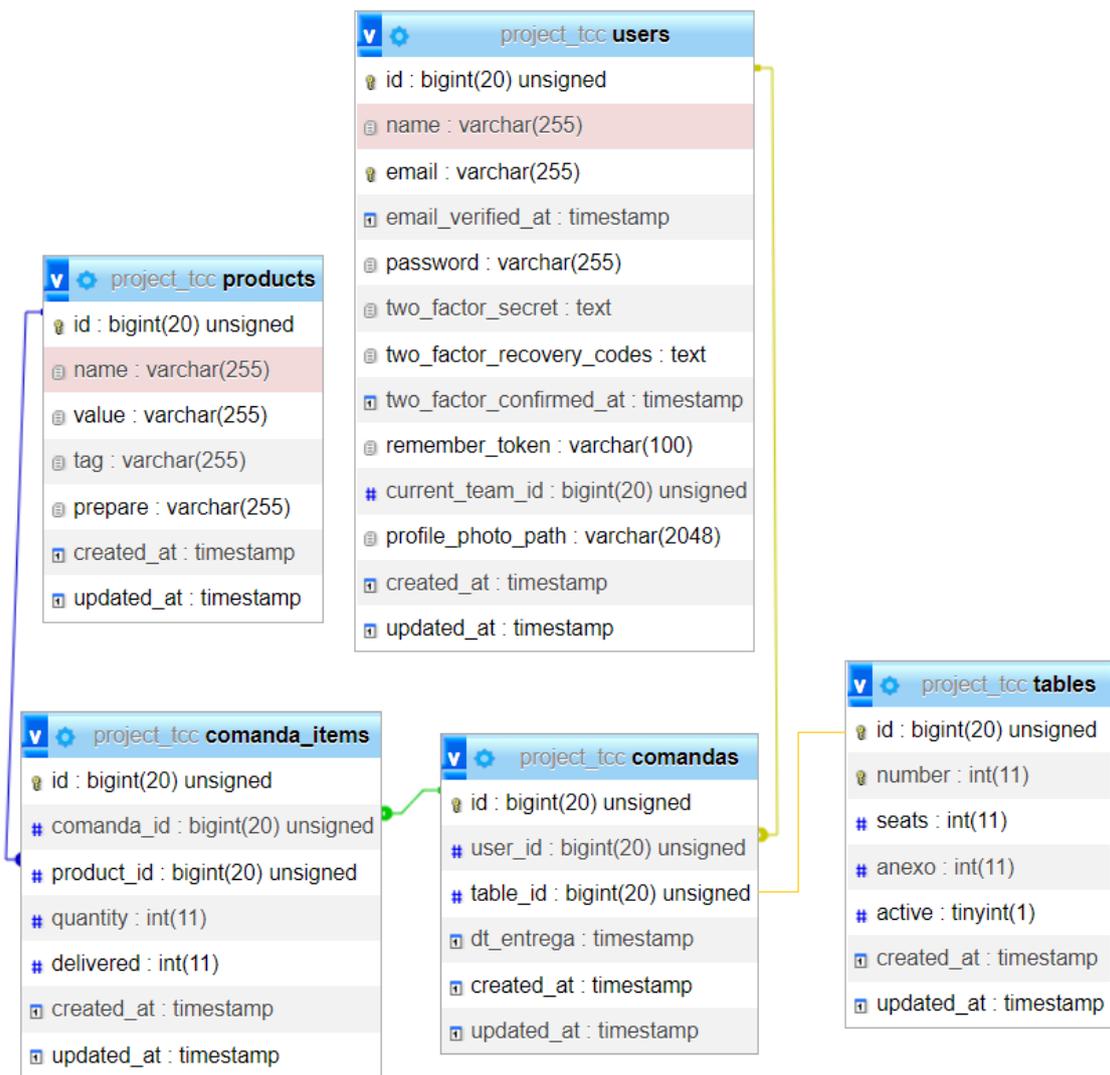
Nesta etapa foi criada uma modelagem com as funcionalidades a serem desenvolvidas. É criada de acordo com funcionalidades específicas e divididas em classes, métodos e atributos [FDD 2022].

No diagrama de atividade, apresentado na Figura 5, é possível ter uma visão das atividades do sistema.



**Figura 5. Diagrama de Atividade**

O diagrama de tabelas, auxilia na representação da estrutura do banco de dados, na Figura 6 é mostrado este diagrama e suas entidades.



**Figura 6. Diagrama de tabelas**

### 3.5. Construir por funcionalidade

Nesta etapa do FDD, é apresentado dentre os códigos implementados os considerados mais relevantes para o sistema.

Uma das funcionalidades mais importantes do sistema é o gerenciamento das comandas dos estabelecimentos, podendo com que o garçom consiga visualizar, inserir, editar e excluir comandas. A Figura 7 apresenta a função responsável pelo armazenamento dos dados da comanda no banco de dados.

A Figura 7 mostra a manipulação de um array de dados que é enviada ao salvar uma comanda, após manipular é salva a comanda, produtos e mesas em suas respectivas tabelas no banco de dados.

---

```

public function store()
{

    $arr = request()->all();
    $idComanda = Comanda::create(array_merge($this->validateComanda(), [
        'user_id' => request()->user()->id,
    ])->id;

    $tableId = $arr['table_id'];
    unset($arr['_token']);
    unset($arr['table_id']);

    foreach ($arr as $key=>$value){

        if(substr($key,0, 8) == 'table_id'){

            $request = request()->validate([
                'table_id'.substr($key,8) => 'max:255'
            ]);

            $id = $request['table_id'.substr($key,8)];

            $table = Table::find($id);
            $table->anexo = $tableId;
            $table->save();
        }

        if(substr($key,0, 10) == 'product_id'){

            $request = request()->validate([
                'product_id'.substr($key,10) => 'max:255',
                'quantity'.substr($key,10) => 'max:255',
                'delivered'.substr($key,10) => 'max:255',
            ]);

            $request['product_id'] = $request['product_id'.substr($key,10)];
            unset($request['product_id'.substr($key,10)]);
            $request['quantity'] = $request['quantity'.substr($key,10)];
            unset($request['quantity'.substr($key,10)]);
            $request['delivered'] = $request['delivered'.substr($key,10)];
            unset($request['delivered'.substr($key,10)]);
            ComandaItem::create(array_merge($request, ['comanda_id' => $idComanda]));
        }

    }
}

```

**Figura 7. Função Store, referente a gravação de informações das comandas no banco de dados**

Outra função importante é o adicionar de produtos na comanda, na Figura 8 é possível visualizar o evento atribuído a classe addProduto, aonde é adicionado o produto na comanda com os valores ajustados na tabela.

```

$(' .addProduto').click(function(){
  @foreach ($products as $product)
  if($('#selectProduto').val() == {{ $product->id }}){
    var html = "<tr row='{{ $product->id }}'>"
    + "<th> <input type='text' class='hidden' name='product_id' value='{{ $product->id }}'></input>{{ $product->name }} </th>"
    + "<td> R$ {{ $product->value }}</td>"
    + "<td tipo='quantidade' valor='{{ $product->value }}'>"
    + " <button type='button' style=' font-size: 10px; padding: 5px; margin-right: 9px;' class='menosQuant quant btn btn-danger'><i class='fa"
    + " <input style=' background-color: transparent; width: 50px;' type='text' id='total' name='delivered' value"
    + " <button type='button' style=' font-size: 10px; padding: 5px; margin-left: 9px;' class='maisQuant quant btn btn-success'><i class='fa-"
    + '</td>'
    + '<td tipo="total">'
    + " <button type='button' style=' font-size: 10px; padding: 5px; margin-right: 5px;' class='menosQuant btn btn-danger'><i class='fa-solid fa-minu"
    + " <input style=' background-color: transparent; width: 50px;' type='text' id='total' name='delivered' value"
    + " <button type='button' style=' font-size: 10px; padding: 5px; margin-left: 5px;' class='maisQuant btn btn-success'><i class='fa-solid fa-plus"
    + '</td>'
    + "<td> <span id='valor' value='{{ $product->id }}'>R$ <span>{{ $product->value }}</span></span></td>"
    + "</tr>";
  }
  @endforeach
  $('#tableProdutos tbody').prepend(html);
  $('# .maisQuant').unbind();
  $('# .menosQuant').unbind();
  $('# .quant').unbind();
  alteraQuant();
  alteraTotais();
  $('# .quant' ).trigger( "click" );
});

```

Figura 8. Evento responsável por adicionar produtos na comanda

## 4. Resultados

Como resultado do presente trabalho, um sistema web foi desenvolvido com foco no gerenciamento de mesas e comandas em estabelecimentos alimentícios. O sistema utiliza de interface simplificada para melhor compreensão do cliente.

Na Figura 9, é apresentada a interface contendo o cadastro da comanda, contendo os produtos e mesas. Ao adicionar uma comanda é possível adicionar a qual mesa esta comanda está associada e assim como as mesas anexadas. Para parte de produtos, é possível adiciona-los junto a sua quantidade e o quanto desses já entregue, tendo uma visão dos valores unitários bem como o total.

×
Comanda

Gabriel de Mello Flores

Mesa  Anexar Mesa +

Pedido

Produto	Preço	Quantidade	Entregue	Total
X-Filé	R\$ 16	<span style="color: red;">-</span> <input type="text" value="1"/> <span style="color: green;">+</span>	<span style="color: red;">-</span> <input type="text" value="0"/> <span style="color: green;">+</span>	R\$ 16
X-Calabresa	R\$ 14	<span style="color: red;">-</span> <input type="text" value="2"/> <span style="color: green;">+</span>	<span style="color: red;">-</span> <input type="text" value="0"/> <span style="color: green;">+</span>	R\$ 28
X-Coração	R\$ 19	<span style="color: red;">-</span> <input type="text" value="2"/> <span style="color: green;">+</span>	<span style="color: red;">-</span> <input type="text" value="0"/> <span style="color: green;">+</span>	R\$ 38
Refrigerante	R\$ 6	<span style="color: red;">-</span> <input type="text" value="5"/> <span style="color: green;">+</span>	<span style="color: red;">-</span> <input type="text" value="5"/> <span style="color: green;">+</span>	R\$ 30

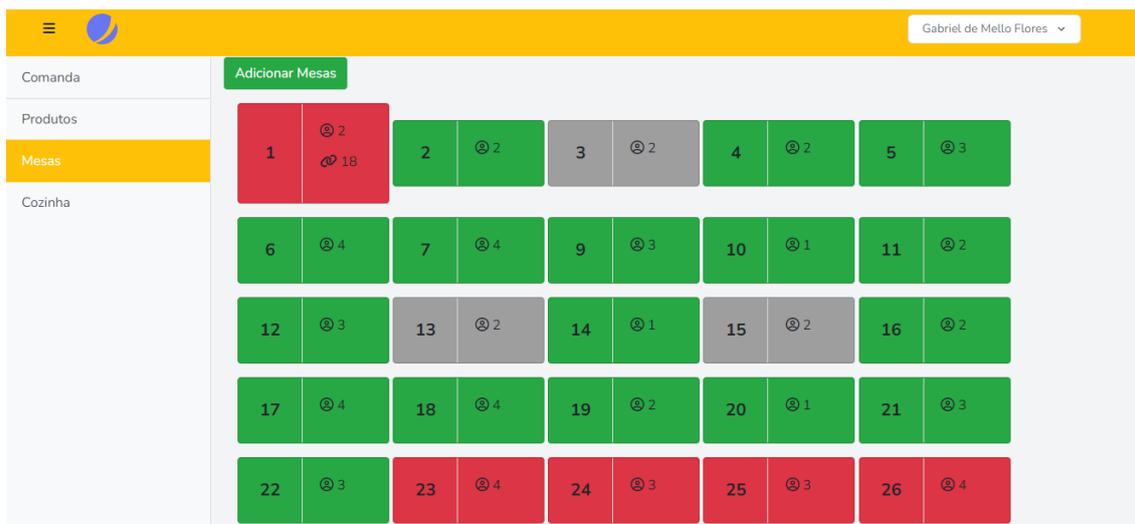
Add Produto

Total
R\$ 112

Cancelar
Salvar

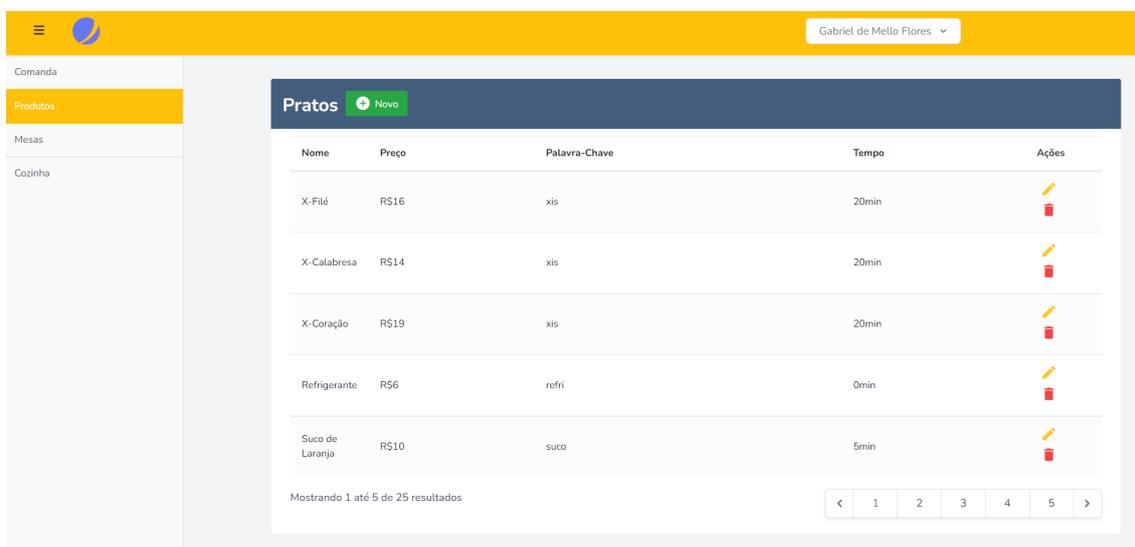
Figura 9. Tela de comanda

Já na Figura 10 é possível visualizar a tela de mesas na qual se pode ver a listagem das mesas, nela você também pode adicionar, editar e excluir estas mesas. As cores são para fácil visualização do usuário no estado dessas mesas, onde cinza significa mesas não disponíveis para utilização, vermelha mesas que estão ocupadas no momento e verde para mesas disponíveis. Em cada uma das mesas, vem a informação do número de lugares que elas podem comportar e para caso de mesas juntas, é possível ver um cadeado com o número da mesa que está acoplada nela.



**Figura 10. Tela das mesas**

Na Figura 11 é possível visualizar a tela de produtos. Nela é listada os produtos e é possível cadastrar, editar e excluir estes produtos. Ao clicar no botão no botão para cadastrar ou editar, um modal irá ser criado com os campos, nome, valor, tag e tempo de preparo.



**Figura 11. Tela de produtos**

## 5. Conclusões

Este trabalho teve, como objetivo geral, a realização de um sistema web que controle as comandas e mesas de restaurantes e bares, a fim de melhorar o controle e reduzir os erros desses estabelecimentos, com a finalidade de aprimorar a relação com o cliente.

Através de estudos realizados nos trabalhos correlacionados que foram apresentados, foi possível compreender melhor as falhas destes estabelecimentos, assim como formas de solucionar estes problemas.

A metodologia FDD foi utilizada, tendo em vista uma melhor visão de tudo que irá ser desenvolvido, já que foi dividida as funcionalidades do projeto, dando prioridade as tarefas principais em paralelo a correção de problemas que poderão aparecer, uma vez utilizando suas boas práticas.

Portanto, de maneira geral, com o auxílio da linguagem e os *frameworks* selecionados, que facilitaram o desenvolvimento do sistema, e com as fases do projetos divididos no cronograma de trabalho, foram aplicados na práticas tudo que foi abordado no presente trabalho.

Em relação a trabalhos futuros, planeja-se desenvolver uma conexão via API de pagamento, para o cliente fazer o pagamento da comanda assim que encerrada. Também busca-se aplicar conceitos de SaaS(*Software* como serviço) visando a comercialização do sistema e posteriormente aplicar validações para testes do sistema.

## Referências

- Abrasel (2021). *Trabalho, profissionalização e investimentos pelo desenvolvimento do Brasil*. Disponível em: <https://pb.abrasel.com.br/abrasel/>. Acesso realizado em março de 2022.
- Audy, J. L. N., de Andrade, G. K., and Cidral, A. (2009). *Fundamentos de sistemas de informação*. Bookman editora.
- Bootstrap (2022). *Bootstrap · The most popular HTML, CSS, and JS library in the world*. Disponível em: <https://getbootstrap.com/>. Acesso em: Março/2022.
- Carlos, H. (2015). *Sistemas de Informação Gerenciais em RH*. Editora Pearson.
- Carlos, R. N. (2017). Estudo de tecnologias atuais para elaboração de arquitetura de sistemas desenvolvidos em php. *aham*.
- Costa, F. F. (2014). Design responsivo para web com bootstrap.
- Cruz, J. (2015). Sistema de gestão de pedidos para restaurantes e lanchonetes.
- Dall'Oglio, P. (2018). *PHP Programando com orientação a Objetos*. Novatec Editora.
- FDD (2022). *Feature Driven Development*. Disponível em: <http://www.featuredrivendevelopment.com/>. Acesso realizado em abril de 2022.
- Foggetti, C. (2015). *Gestão ágil de projetos*. Editora Pearson.
- Gomes, J., Santiago, L., Silva, A., Corrêa, B., and Costa (2018). Smartfood: Sistema de gerenciamento de pedidos.

- Laravel (2022). *Laravel - The PHP Framework For Web Artisans*. Disponível em: <https://laravel.com/>. Acesso em: Março/2022.
- Laudon, K. and Laudon, J. (2017). *Sistemas de Informação Gerenciais - 7ª edição*. Editora Pearson.
- Lima, T., Pedroso, A., Neto, J., and Barbieri, P. (2017). Sistema de cardápio digital para bares, restaurantes e similares.
- MySQL (2022). *MySQL Documentation*. Disponível em: <https://www.mysql.com/>. Acesso em: Maio/2022.
- Olanrewaju, R. F., Islam, T., and Ali, N. (2015). An empirical study of the evolution of php mvc framework. In *Advanced Computer and Communication Engineering Technology*. Springer.
- PHP (2022). *PHP:Hypertext Preprocessor*. Disponível em: <https://php.net/>. Acesso em: Março/2022.
- Rigby, D. K. (2005). *Management tools 2005: An executive's guide*. Bain & Company, Inc.
- Santos, D., Machado, G., and Gomes, L. (2021). Sistema web de gerenciamento de pedidos de restaurantes.
- Silva, D. R. et al. (2017). Sistema para elaboração e controle da planilha de pontuação docente: desenvolvido utilizando o framework laravel e php. Insitituto Federal de Educação, Ciência e Tecnologia de Goiás.
- Silva, F., Hoentsch, S. C., and Silva, L. (2009). Uma análise das metodologias Ágeis fdd e scrum sob a perspectiva do modelo de qualidade mps. br. *Scientia Plena*, 5(12).
- Converse, Tim and Park, Joyce (2003). *PHP: a bíblia*. Gulf Professional Publishing.

## **6. Apêndices**

### **6.1. Apêndice A**

Como pode se ver abaixo está listado os requisitos funcionais e não funcionais.

#### RF1 - Gerenciar Login

- RF 1.1 - Cadastrar Usuários
- RF 1.2 - Consultar Usuários
- RF 1.3 - Alterar Usuários
- RF 1.4 - Excluir Usuários

Descrição: Cadastro, consulta, alteração e exclusão de usuários do sistema.

#### RF2 - Gerenciar Clientes

- RF 2.1 - Cadastrar Clientes
- RF 2.2 - Consultar Clientes
- RF 2.3 - Alterar Clientes
- RF 2.4 - Excluir Clientes

Descrição: Cadastro, consulta, alteração e exclusão de clientes do sistema.

#### RF3 - Gerenciar Produtos

- RF 3.1 - Cadastrar Produtos
- RF 3.2 - Consultar Produtos
- RF 3.3 - Alterar Produtos
- RF 3.4 - Excluir Produtos

Descrição: Cadastro, consulta, alteração e exclusão de produtos do sistema.

#### RF4 - Gerenciar Pedidos

- RF 4.1 - Cadastrar Pedidos
- RF 4.2 - Consultar Pedidos
- RF 4.3 - Alterar Pedidos
- RF 4.4 - Excluir Pedidos

Descrição: Cadastro, consulta, alteração e exclusão de pedidos do sistema.

#### RF5 - Gerenciar Garçons

- RF 5.1 - Cadastrar Garçons
- RF 5.2 - Consultar Garçons
- RF 5.3 - Alterar Garçons
- RF 5.4 - Excluir Garçons

Descrição: Cadastro, consulta, alteração e exclusão de garçons do sistema.

#### RF6 - Gerenciar Mesas

- RF 6.1 - Cadastrar Mesas
- RF 6.2 - Consultar Mesas
- RF 6.3 - Alterar Mesas
- RF 6.4 - Excluir Mesas

Descrição: Cadastro, consulta, alteração e exclusão de mesas do sistema.

RNF01 - Utilização da Linguagem PHP

RNF02 - Utilização do *Framework* Laravel

RNF03 - Utilização do *Framework* Bootstrap

RNF04 - Utilização do SGBD MySQL

## 6.2. Apêndice B

Na Figura 12, mostra as estruturas das classes do sistema, possibilitando a visualização de seus atributos e métodos.

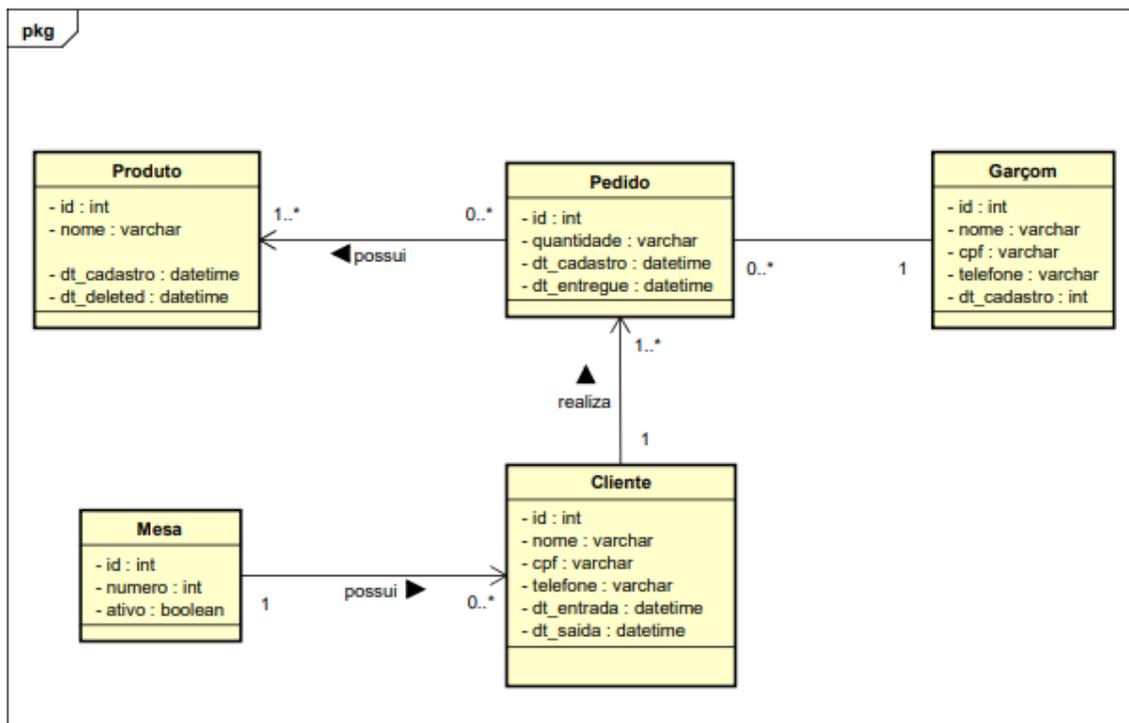


Figura 12. Diagrama de Classes