

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
COINT - TECNOLOGIA EM SISTEMAS PARA INTERNET
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET

FELIPE WEIBER

SISTEMA DE GERENCIAMENTO DE PIZZARIA

TRABALHO DE CONCLUSÃO DE CURSO

GUARAPUAVA
2017

FELIPE WEIBER

SISTEMA DE GERENCIAMENTO DE PIZZARIA

Trabalho de Conclusão de Curso apresentado ao Curso de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Tecnólogo.

Orientador: Prof. Me. Paulo Henrique Soares

Coorientador: Prof. Me. Andres Jessé Porfirio

GUARAPUAVA
2017

RESUMO

WEIBER, Felipe. Sistema de Gerenciamento de Pizzaria. 2017. 28 f. Trabalho de Conclusão de Curso – Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuava, 2017.

Sistemas de informação são ferramentas que administradores utilizam para atingir eficiência e produtividade nas operações, podendo afetar o comportamento e as práticas de negócios. Podem ser definidos como um conjunto de componentes inter-relacionados que coletam, processam, armazenam e distribuem informações para apoiar a tomada de decisão, coordenar e auxiliar no controle de uma organização. Podem ser utilizados também para identificar problemas e visualizar informações mais complexas. Este projeto tem como objetivo desenvolver um sistema de informação para pizzarias. Ele será composto por módulos para controle das transações da pizzaria, como por exemplo, cadastro de clientes, cadastro de produtos, gerenciamento de entregas e mesas, entre outras operações. É composto também por um módulo para auxiliar na tomada de decisão, onde as informações serão apresentadas em forma de relatórios e gráficos, auxiliando o gerente, por exemplo, na elaboração de promoções.

Palavras-chave: Aplicações Web. Engenharia de software x Gerência. PHP (Linguagem de programação de computador).

ABSTRACT

WEIBER, Felipe. Pizzeria Management System. 2017. 28 f. Trabalho de Conclusão de Curso – Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuava, 2017.

Information systems are tools that managers use to achieve efficiency and productivity in operations and can affect behavior and business practices. It can be defined as a set of interrelated components that collect, process, store and distribute information to support decision making, coordinate, and assist in the control of an organization. It can also be used to identify problems and view information that is more complex. This project aims to develop an information system for pizzerias. It will be composed of modules to control the transactions of the pizzeria, such as customer registration, product registration, delivery and table management, among other operations. It is also composed by a module to aid in decision making, where information will be presented in the form of reports and graphs, assisting the manager, for example, in the elaboration of promotions.

Keywords: Web applications. Software engineering x Management. PHP (Computer program language).

LISTA DE FIGURAS

Figura 1 – Fluxo do MVC com Laravel	4
Figura 2 – Captura de tela da versão gráts instalada para teste.	7
Figura 3 – Captura de tela do sistema, pedido.	8
Figura 4 – Tela do sistema SysPizza	9
Figura 5 – Captura de tela, tela de pedido.	9
Figura 6 – A figura mostra os módulos em cada dispositivo.	10
Figura 7 – Exibição dos relatórios na web.	10
Figura 8 – Captura de imagem da tela inicial do sistema.	11
Figura 9 – Captura de imagem do sistema, com exemplo de listagem de pedidos.	11
Figura 10 – Captura de imagem do sistema, menu financeiro.	12
Figura 11 – Estrutura da Base de Dados.	17
Figura 12 – Teste unitário básico do sistema.	18
Figura 13 – Dashboard do sistema.	19
Figura 14 – Tela de cadastro de usuário.	20
Figura 15 – Tela de cadastro de cliente.	21
Figura 16 – Tela de cadastro promoções.	21
Figura 17 – Tela de cadastro de produto.	22
Figura 18 – Tela de abertura de novo pedido.	23
Figura 19 – Exibição das mesas.	23
Figura 20 – Resumo do pedido.	24
Figura 21 – Tela de pagamento.	24
Figura 22 – Gráfico dos produtos mais vendidos.	25
Figura 23 – Relatório de produtos.	25

LISTA DE QUADROS

Quadro 1 – Características dos sistemas apresentados. 12

LISTA DE ABREVIATURAS E SIGLAS

BD	Banco de dados
CSS	Folha de Estilo em Cascata (do inglês <i>Cascading Style Sheets</i>)
HTML	Linguagem de Marcação de Hipertexto (do inglês <i>HyperText Markup Language</i>)
HTTP	Protocolo de Transferência de Hipertexto (do inglês <i>Hypertext Transfer Protocol</i>)
MVC	Modelo, Visualização, Controlador (do inglês <i>Model View Controller</i>)
PHP	Pré-processador de hipertexto (do inglês Hypertext Preprocessor)
SI	Sistema de informação
SO	Sistema Operacional

SUMÁRIO

1 – INTRODUÇÃO	1
1.1 OBJETIVOS	1
1.1.1 Objetivo Geral	1
1.1.2 Objetivos Específicos	2
1.2 ESTRUTURA DA MONOGRAFIA	2
2 – REVISÃO DE LITERATURA	3
2.1 FUNDAMENTAÇÃO TEÓRICA	3
2.1.1 MySQL	3
2.1.2 Apache	3
2.1.3 Linguagem de programação: PHP	4
2.1.3.1 Laravel	4
2.1.4 HTML, CSS e Javascript	5
2.1.4.1 Materialize	5
2.1.4.2 AngularJS	5
2.1.4.3 Google Charts Tool	6
2.1.4.4 DataTable	6
2.2 ESTADO DA ARTE	6
2.2.1 Eclética Tecnologias	6
2.2.2 Pizzaria Free	7
2.2.3 SysPizza	8
2.2.4 Consumer	9
2.2.5 Comparativo	12
3 – METODOLOGIA	13
4 – DESENVOLVIMENTO	14
4.1 FUNCIONALIDADES DO SISTEMA	14
4.2 REQUISITOS DO SISTEMA	14
4.2.1 Requisitos Funcionais	14
4.2.2 Requisitos não funcionais	15
4.2.3 User Stories, Histórias de Usuários	15
4.3 BASE DE DADOS	16
4.4 TESTE	16
5 – SISTEMA	19
5.1 CONFIGURAÇÕES, CONTROLE DE ACESSO e DASHBOARD	19

5.2 CADASTROS	20
5.2.1 Cadastros de Usuários, Categorias e Taxas de entrega	20
5.2.2 Cadastros de Clientes, Produtos e Promoções	20
5.3 PEDIDOS	22
5.4 RELATÓRIOS	24
6 – CONCLUSÃO	26
6.1 TRABALHOS FUTUROS	26
Referências	27

1 INTRODUÇÃO

A utilização da informação como ferramenta de estratégia está cada vez mais presente no cotidiano das organizações. Sistemas computacionais são cada vez mais utilizados para criar, armazenar e transferir informações. Tais sistemas são chamados de Sistemas de Informação (SI).

Um SI é um conjunto de componentes inter-relacionadas que coleta, manipula, armazena e dissemina dados e informação e fornece um mecanismo de realimentação para atingir um objetivo. As empresas podem utilizar os sistemas de informação para aumentar receitas e reduzir custos ([REYNOLDS; STAIR, 2011](#)).

Os SI são utilizados por empreendedores e por proprietários de negócios para alcançar seus clientes. Representantes de vendas os usam para anunciar produtos, comunicar-se com os clientes e analisar as tendências de venda. Independente do porte da organização, seja uma pequena empresa ou uma grande corporação, todas precisam de um SI para controlar suas operações, principalmente se tratando de operações contábeis, fiscais, financeiras e de recursos humanos. Estes sistemas precisam também, fornecer informações corretas para pessoas específicas da organização com intuito de auxiliar na tomada de decisão.

A utilização de um SI no ramo de pizzarias é de grande valia. Sua aplicação seria na coleta de dados de vendas, na gestão de funcionários, controle de entrada de produtos utilizados como ingredientes, entre outras aplicações.

Desta forma, o presente trabalho tem como objetivo desenvolver um sistema de informação para pizzaria, que proporcione a coleta dos dados, por meio do registro de vendas e controle financeiro. Estes dados serão representados por meio de gráficos e tabelas. Assim, o gerente poderá tomar melhores decisões visando redução de desperdícios e consequente redução de custos, impactando nos resultados da empresa, possibilitando inclusive um aumento no lucro.

O sistema desenvolvido neste trabalho, além de gerenciamento de vendas e controle financeiro, exibirá os pedidos para entrega em um mapa, auxiliando na localização do pedido, bem como, cadastro de clientes. As informações corretas, como o endereço, serão obtidas de um *webservice*. O sistema possuirá também uma área específica para visualização de relatórios e acompanhamento do andamento dos processos.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Criar um sistema de informação para pizzarias, com controle de vendas, cadastros de clientes e gráficos para auxiliar nas decisões.

1.1.2 Objetivos Específicos

- Fornecer um sistema com armazenamento de informações em banco de dados;
- Desenvolver um módulo para cadastro de clientes;
- Desenvolver um módulo para cadastro de produtos;
- Desenvolver módulo de pedidos;
- Desenvolver módulo administrativo para configurações, visualização de relatórios e gráficos;
- Implementar um módulo para gerenciamento de vendas externas (*delivery*) e vendas locais

1.2 ESTRUTURA DA MONOGRAFIA

O presente trabalho está organizado da seguinte forma: Revisão de Literatura, com os principais conceitos teóricos relacionados ao tema, Estado da Arte, mostrando alguns trabalhos (sistemas) correlatos, Metodologia, contendo os procedimentos executados para elaboração do trabalho, Desenvolvimento, abordando sobre as tarefas executadas em diferentes etapas do desenvolvimento, Sistema, apresentando o produto (sistema) desenvolvido com detalhamento de seus recursos e funcionalidades, e por fim, Conclusão, contendo as considerações finais sobre o trabalho, bem como, sugestões de trabalhos futuros.

2 REVISÃO DE LITERATURA

Este capítulo apresenta as definições dos conceitos relacionados com o presente trabalho e o estado da arte.

2.1 FUNDAMENTAÇÃO TEÓRICA

Neste item serão apresentadas as tecnologias utilizadas no desenvolvimento do sistema para pizzaria. O sistema será desenvolvido seguindo o padrão Model-View-Controller (MVC), que segundo [Zervaas \(2008\)](#) é geralmente utilizado em projetos de aplicações web. Model representa a lógica do sistema, View representa a interação com o usuário, Controller faz a intermediação entre Model e View.

2.1.1 MySQL

De acordo com [ORACLE \(2017\)](#), MySQL é o BD de código aberto mais conhecido no mundo. Com desempenho comprovado, confiabilidade e facilidade de uso. O MySQL é uma das principais opções de BD para aplicativos baseados na Web, usado por propriedades da Web de alto perfil, incluindo *Facebook*, *Twitter*, *YouTube*.

O MySQL possui um mecanismo chamado *InnoDB*, que adere o modelo *ACID* (Atomicidade, Consistência, Isolamento, Durabilidade), para garantir que não haja falhas, dados inconsistentes e dados repetidos ou corrompidos. Possui suporte para backups.

O BD possui fácil integração com a linguagem de programação PHP, que foi a base de desenvolvimento do sistema deste projeto.

2.1.2 Apache

O Servidor HTTP Apache é um servidor web que foi escolhido neste trabalho para centralizar o sistema. No servidor estará a lógica do sistema e este será responsável por enviar dados para os clientes ([FOUNDATION, 2017a](#)).

Características:

- Software Livre, qualquer um pode utilizar;
- Multiplataforma, pode ser instalado em diversos SO, como *Windows*, e *Linux*;
- Suporte ao protocolo HTTP, que é o protocolo utilizado pelos navegadores web;
- Capaz de executar código PHP, que foi utilizado para o desenvolvimento do sistema, *Perl*, *Shell Script*, etc.

2.1.3 Linguagem de programação: PHP

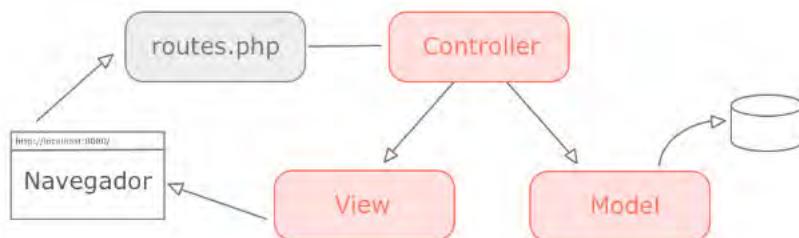
O PHP é uma linguagem de *script* de código aberto e uso geral, utilizado para o desenvolvimento web. Ele é interpretado do lado do servidor, pode-se coletar dados de formulários, gerar páginas com conteúdo dinâmico. Possui suporte a vários bancos de dados ([PHP, 2017](#)).

Para o desenvolvimento do sistema foi utilizado um *framework* PHP chamado Laravel.

2.1.3.1 Laravel

Laravel é um *framework* PHP, é utilizado para agilizar o processo de desenvolvimento web. Ele é definido com o padrão MVC ([TURINI, 2015](#)).

Figura 1 – Fluxo do MVC com Laravel



Fonte: [Turini \(2015\)](#)

O *Laravel* possui suporte a diversos BD como *MySQL* e *Postgress*. Possui um arquivo de configuração que permite mudar o BD de acordo com a preferência. Para estruturar o BD são usados arquivos do *Laravel* que possuem operações para criação do BD ou até mesmo atualização, facilitando a instalação do sistema no servidor.

O *Laravel* possui também uma biblioteca que fornece uma configuração para compilação de *css* e *javascript* usando vários pré-processadores. Pode ser compilado o *css* em dois tipos, para produção, onde os arquivos gerados são minificados para ter um tamanho final pequeno, permitindo o tráfego menor pela rede quando requisitados, e para desenvolvimento, onde os arquivos permanecem com seu código original para facilitar na hora de analisar o código gerado.

Características do Laravel:

- Possui um site oficial <https://laravel.com/>, possui documentação completa;
- Integração com vários bancos de dados;
- Curva de aprendizado pequena.

Por se tratar de um *framework* completo o Laravel foi a base do sistema, nele foi feita a comunicação com BD assim como a interface do usuário.

2.1.4 HTML, CSS e Javascript

HTML (HyperText Markup Language) é uma linguagem de marcação de hipertexto para criação de páginas web. Essa linguagem é interpretada por navegadores, criando o conteúdo exibido ([W3C, 2017](#)).

CSS é a abreviação de Cascading Style Sheets que em português é “folha de estilo”, é utilizada para definir a apresentação das páginas na internet que utilizam linguagem de marcação, isto é, é usado para personalizar páginas criadas em HTML. O CSS define como serão os elementos, separando assim o conteúdo da formatação ([W3C, 2017](#)).

Javascript é uma linguagem de programação para HTML. Onde HTML define o conteúdo da página, o CSS define a formatação, e o Javascript é usado para programar o comportamento das páginas web. Quando uma página web é carregada, o navegador cria um Document Object Model (DOM), que é possível ser manipulado com Javascript.

Com Javascript é possível se comunicar com o servidor utilizando o método AJAX que faz requisições no servidor sem a necessidade de recarregar a página web ([W3C, 2017](#));

2.1.4.1 Materialize

Materialize é um *framework front-end* moderno e responsivo baseado no Material Design da Google (é uma linguagem visual criada para os usuários que sintetiza os princípios do design com a inovação e a possibilidade de tecnologia e ciência.). Ele é um *framework* para desenvolvimento ágil, possui estilos CSS e funções já prontas de acordo com as especificações, minimizando o trabalho pesado do desenvolvedor ([MATERIALIZE, 2017](#)).

Possui uma documentação completa disponível no site oficial “<http://materializecss.com/>”.

Neste projeto foi desenvolvido um sistema responsivo e agradável ao usuário, com a utilização do materialize se torna mais prático o desenvolvimento e possível de se alcançar estas características.

2.1.4.2 AngularJS

O *AngularJS* é um *framework front-end* que utiliza a linguagem de programação Javascript. É estruturado no padrão MVC, permitindo uma organização no código, e permite a utilização de diversas funcionalidades como a criação de componentes para reuso de código, validação de formulário, manipulação da DOM, possui funções para acesso via AJAX ao servidor ([SESHADRI; GREEN, 2014](#)). A versão para o desenvolvimento foi a versão 1.6.6.

O Angular possui documentação completa disponível no site oficial. Com esse *framework* as responsabilidades são divididas, deixando o servidor responsável pelas lógicas de negócio e o armazenamento de dados e o angular responsável pela renderização das telas e buscas de dados no servidor.

2.1.4.3 Google Charts Tool

O Google Charts é uma ferramenta gratuita para criação de gráficos interativos. É compatível com vários navegadores, possui uma ampla variedade de gráficos, tabelas e também mapas, permitindo dados dinâmicos.

No sistema foi utilizado o Google Charts para a apresentação dos relatórios, esta ferramenta possui como característica não exigir a preocupação com detalhes visuais, o programador deve ficar atento apenas com os dados a serem exibidos ([GOOGLE, 2017](#)).

2.1.4.4 DataTable

Datatables é um plugin para a *jQuery*¹. É uma ferramenta que adiciona controles de interações avançados nas tabelas HTML. Possui em sua implementação paginação, busca instantânea, com múltipla ordenação por coluna, pode ser aplicado temas de *bootstrap*, *Foundation*, etc.. Também possui configurações de internacionalização permitindo configuração de idiomas desejado. Suporta várias fontes de dados, como Ajax, Javascript, processamento do lado do servidor.

Neste projeto foi utilizado uma versão que integra junto com o *laravel*. No front-end é utilizado o *DataTable*, onde foi configurado a internacionalização para apresentar os textos no idioma português do Brasil e a comunicação para acessar a rota de busca no servidor via Ajax. No servidor foi configurado a rota com um método que utiliza uma biblioteca específica para integração do *DataTable*.

2.2 ESTADO DA ARTE

Atualmente existem vários sistemas com foco em pizzaria, que efetuam diversos serviços, como cadastro de clientes, cadastro de produto, controle de entrega, identificação de chamada com um software chamado BINA², entre outros. As subseções a seguir apresentam alguns sistemas disponíveis para este ramo de mercado.

2.2.1 Eclética Tecnologias

A empresa [Ecletica \(2016\)](#), com mais de 25 anos de experiência no setor de automação comercial e com foco em *Food Service*, possui um sistema para Soluções em pizzarias. Esse sistema é modular e possui os módulos de: Central de *delivery*, pedidos web, e-garçom, aplicativos, cardápio eletrônico, módulo *master*. Possui ainda um serviço específico para realizar entregas com busca de endereço por CEP, taxa de entrega por CEP, venda de produto de

¹*jQuery* é um biblioteca javascript rica em recursos, fácil de ser usada e funciona na maioria dos navegadores ([FOUNDATION, 2017b](#)).

²BINA é um dispositivo eletrônico que identifica o número do telefone de quem está ligando, é um identificador de chamada, utilizado por sistemas para captura do número ([BINA, 2009](#)).

vários sabores, aplicativo para *delivery* ou retirada no balcão, disponível em *Android* e *IOS*, baixas no estoque por receitas e baixas no estoque automático.

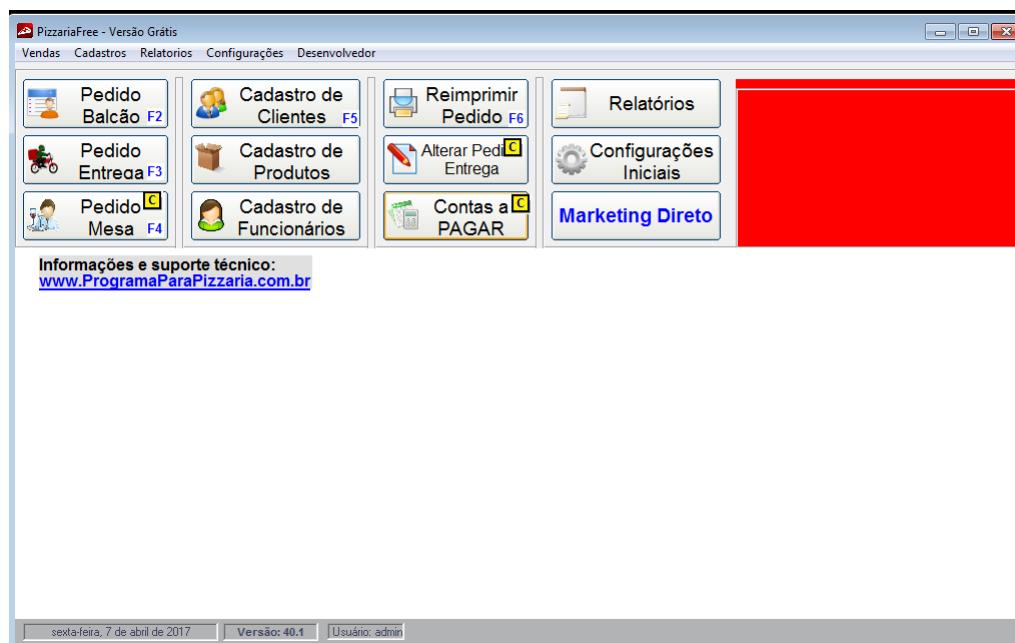
2.2.2 Pizzaria Free

[PizzariaFree \(2015\)](#) é um sistema que possui recursos como marketing, envio de *sms* para os clientes no módulo Marketing Direto, cadastros de clientes com alertas caso o cliente possua algum tipo de alergia de ingredientes, cadastro de produtos com opções de preços diferentes para promoções, funciona em ambiente de rede e é possível integrar com impressoras. O sistema possui relatórios para controle interno e recurso com BINA para identificar o número da ligação recebida. Este sistema possui uma versão grátil e uma versão completa, que é vendida comercialmente com valores por planos divulgados no site. A Figura 2 apresenta a tela inicial do sistema.

Para conhecer melhor o sistema, foi instalada a versão gratuita para testes. Esta é disponibilizada para o SO Windows e não possui todos os recursos da versão completa, porém foi possível entender seu funcionamento e conhecer as principais características do software.

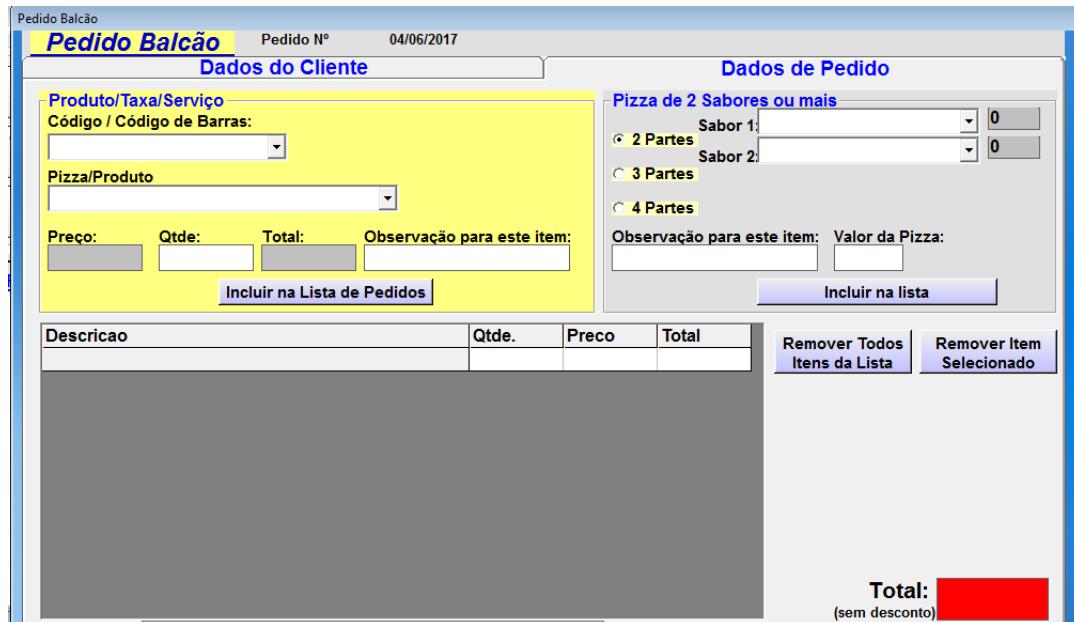
O pizzaria Free possui uma interface simples, roda somente em *desktop*, possui atalhos para facilitar a interação com o usuário. Podem ser feitos pedidos por mesa, ou para entrega separadamente. O cadastro de cliente possui uma opção para buscar o endereço utilizando o CEP, entretanto é necessário estar conectado à internet. A Figura 3 mostra a tela de pedido feito no balcão onde são mostradas as opções de pizzas para inserir no pedido, mostrando valores e totais.

Figura 2 – Captura de tela da versão grátil instalada para teste.



Fonte: O autor

Figura 3 – Captura de tela do sistema, pedido.



Fonte: O autor

2.2.3 SysPizza

O [SysPizza \(2017\)](#) também se encaixa em sistemas desenvolvidos para pizzarias, tendo como principais características: Vendas de balcão, *delivery* e mesas. Foi desenvolvido no ano de 2007, sendo atualizado de acordo com a necessidade de cada pizzaria. Atende centenas de pizzarias no Brasil, Argentina e Uruguai. Possui gravador de chamadas e cadastro de valores para diferentes dias da semana e assim como os sistemas apresentados anteriormente, possui um sistema de BINA para identificação de chamadas, busca de endereço por CEP. A Figura 4 apresenta uma tela do sistema.

Foi instalada uma versão grátil para testes, logo após instalada, o sistema mostra informações a respeito de como usar seus recursos. Como outros sistemas apresentados neste trabalho, este só pode ser instalado em SO Windows. O sistema em sua versão gratuita não possui um BD, as transações são salvas somente em arquivos.

Figura 4 – Tela do sistema SysPizza



Fonte: Website do desenvolvedor

Figura 5 – Captura de tela, tela de pedido.



Fonte: O autor

2.2.4 Consumer

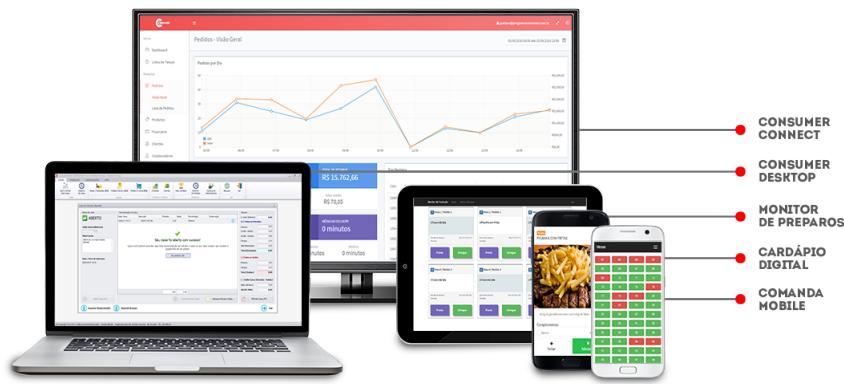
O programa para Pizzarias e *Delivery Consumer* de RAL (2017), está instalado em mais de 55 mil estabelecimentos em todo o Brasil. Possui pedidos para mesas, pedidos para *delivery*, frente de caixa, sistema BINA, impressora na rede e módulos financeiros. Pode-se visualizar os pedidos no *google Maps*, cadastros de produtos e clientes, resumo financeiro, relatório de desempenho mensal. A Figura 6 apresenta o sistema em diferentes dispositivos e a Figura 7 mostra um gráfico de pedidos por dia.

Para conhecer melhor o sistema, foi instalada uma versão *premium* para testes, essa versão ficou disponível por 15 dias sem custo, diferente dos outros sistemas apresentados, esse possui conexão com BD *Firebird*, possui cardápios que podem ser visualizados por um aplicativo *mobile*, e tem suporte a web para visualização de relatórios para acompanhar o estabelecimento, também possui uma funcionalidade de mostrar em monitor qual pedido está pronto.

O sistema possui uma interface agradável e de fácil utilização. Na opção de pedidos por entregas, pode-se visualizar o mapa para ajudar o atendente a se localizar e passar as informações corretamente. Pode ser feito pedido por número da mesa.

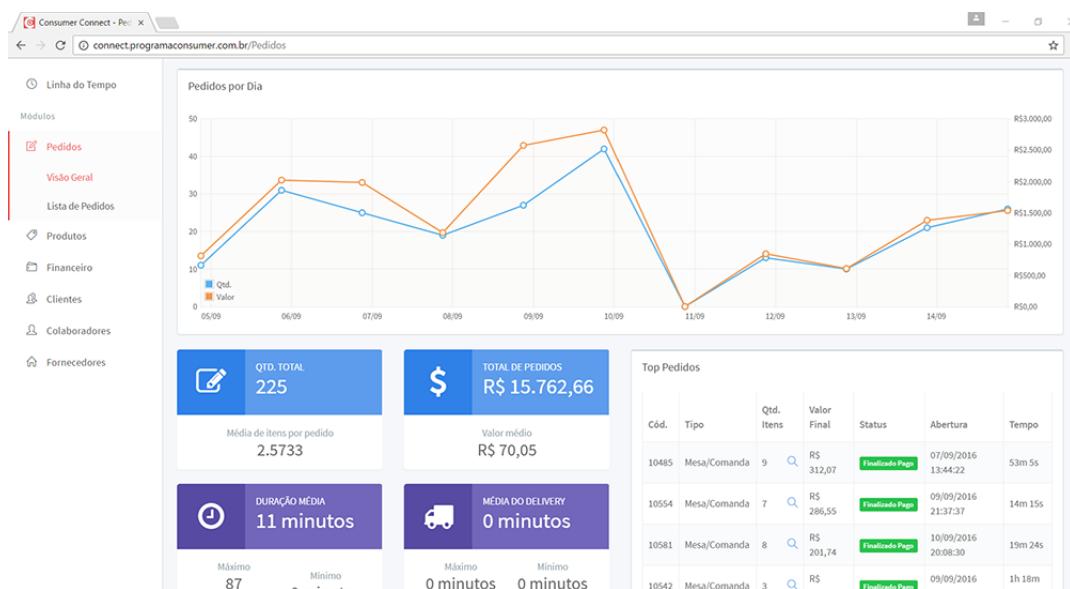
A Figura 8, mostra a tela inicial do sistema, pode-se observar que possui um menu intuitivo com todas as funcionalidades do sistema. A Figura 9, mostra a listagem de pedidos, mostrando também um mapa para se poder observar o percurso para entrega. E a Figura 10, mostra o menu financeiro do sistema, onde é possível cadastrar fornecedores, cadastrar contas e ter um indicador do faturamento.

Figura 6 – A figura mostra os módulos em cada dispositivo.



Fonte: Website do desenvolvedor

Figura 7 – Exibição dos relatórios na web.



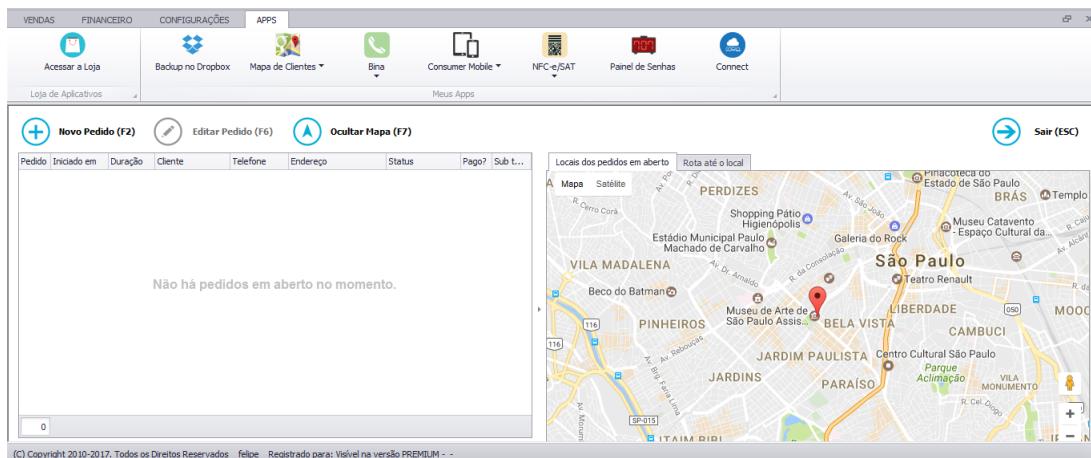
Fonte: Website do desenvolvedor

Figura 8 – Captura de imagem da tela inicial do sistema.



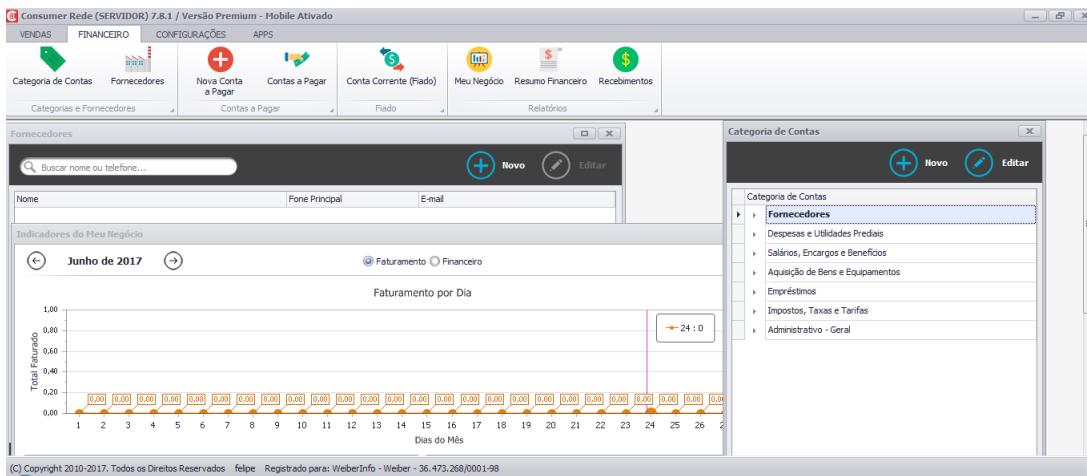
Fonte: O autor

Figura 9 – Captura de imagem do sistema, com exemplo de listagem de pedidos.



Fonte: O autor

Figura 10 – Captura de imagem do sistema, menu financeiro.



Fonte: O autor

2.2.5 Comparativo

O quadro abaixo mostra algumas das características dos sistemas apresentados. Todos os sistemas na versão grátis não são multiplataforma e utilizam um sistema de arquivo para salvar os dados. Somente o sistema Consumer possui armazenamento em um BD e recursos Web, que são somente os relatórios para acompanhar a empresa.

Quadro 1 – Características dos sistemas apresentados.

Características	Eclética Tecnologias	Pizzaria Free	SysPizza	Consumer
1. Multiplataforma				
2. BD				X
3. Salvar em arquivo	X	X	X	X
4. Recursos WEB				X
5. Relatórios	X	X	X	X
6. Busca por CEP	X	X	X	X
7. Mobile				X

3 METODOLOGIA

Neste tópico é apresentada a metodologia utilizada para o desenvolvimento do sistema. Os passos são apresentados a seguir:

1. Levantamento de requisitos: definir os requisitos que o sistema deve cumprir para o funcionamento. Foi feita uma busca do que é necessário para o funcionamento de uma pizzaria;
2. Desenvolvimento do BD: criar o banco de dados com as tabelas e atributos necessários para o sistema;
3. Desenvolvimento do Sistema: foi feita a estrutura do sistema com base nos requisitos;
4. Desenvolvimento dos cadastros: primeiramente foram desenvolvidos os cadastros do sistema e funcionalidade de mostrar todos os dados;
5. Desenvolvimento dos módulos de pedidos e entregas: Foi desenvolvido módulo do processo de pedidos e entregas com funcionalidades para visualizar, abrir e finalizar;
6. Desenvolvimento de relatórios do sistema: foi desenvolvido a exibição dos relatórios das informações no BD, mostrando gráficos e tabelas;
7. Desenvolvimento do controle de acesso: foi feito o controle de acesso para permitir ao usuário acessar somente o que possuir permissão;
8. Testes: efetuar os diferentes tipos de teste no sistema, os testes unitários e testes funcionais para garantir que não há erros.
9. Efetuar cadastros de todos os formulários;
10. Efetuar testes de desempenho para analisar a capacidade do sistema;

4 DESENVOLVIMENTO

Neste trabalho foi desenvolvido um SI para pizzaria. O sistema foi dividido em dois níveis: nível de usuário normal e nível de administrador, no nível administrador é possível efetuar os cadastros acompanhar os relatórios, configurar informações do sistema, e no nível de usuário é permitido somente o processo de transações de dados.

O sistema foi desenvolvido utilizando tecnologias web, permitindo que possa ser acessado de qualquer dispositivo, computador, notebook, *tablet*, entre outros. Para o funcionamento do sistema é necessário um servidor, que pode estar em uma rede local ou externa, os demais terminais podem ser acessados utilizando o navegador web disponível nos próprios dispositivos.

O sistema foi desenvolvido utilizando o framework css Materialize para fins de aproveitar o seu design amigável e responsivo, se adaptando assim a qualquer dispositivo que possua um navegador de internet.

4.1 FUNCIONALIDADES DO SISTEMA

As funcionalidades que foram desenvolvidas no sistema são:

- Cadastro de clientes com busca de endereço por CEP;
- Cadastro de Produtos: pizzas, bebidas, aperitivos, entre outros;
- Módulo delivery: controle de entregas e retiradas no balcão com visualização dos pedidos que saíram para entrega e quais já foram entregues;
- Módulo mesa: controle dos pedidos efetuados nas mesas, pedidos ficam em aberto até receberem o pagamento;
- Visualizador de pedido: ideal para os funcionários, como garçom e pizzaiolo saberem quais pedidos estão na vez;
- Módulo de cadastro: disponível para usuários de nível administrador, neste módulo que possui os cadastros de clientes, produtos, categorias, usuários entre outros.
- Módulo de relatório: disponível para usuários de nível administrador.

4.2 REQUISITOS DO SISTEMA

Para o desenvolvimento do sistema foi feito um levantamento preliminar dos requisitos do sistema, que são divididos em requisitos funcionais e não funcionais.

4.2.1 Requisitos Funcionais

- Cadastro de usuários para poder acessar o sistema;
- Cadastro de produtos: Pizzas, Bebidas;
- Cadastro de clientes com nome e dados de contato, endereço e telefone/celular;

- Buscar clientes pelo número do telefone ou pelo nome;
- Registrar pedidos dos clientes tanto nas mesas quanto em *delivery* (entrega);
- Monitoramento dos pedidos;
- Exibição de relatórios do sistema: Produtos mais vendidas; Bairros mais vendidos na cidade;
- Tela de pagamento a partir de um pedido feito;
- Deve ser possível cadastrar um cliente na abertura do pedido;
- Possuir preço normal e preço promocional;

4.2.2 Requisitos não funcionais

- Funcionamento em ambiente web;
- Compatibilidade com navegadores padrões;
- Busca de endereço por CEP em api de terceiros;
- Linguagem de programação PHP;
- Acesso ao sistema com usuário e senha;

4.2.3 *User Stories*, Histórias de Usuários

Os requisitos podem ser organizados utilizando *User Stories*¹. Seguem algumas *User Stories* com critérios de aceitação criados para o desenvolvimento do sistema deste projeto:

Como um administrador eu quero cadastrar um usuário;

- O administrador deve acessar o sistema;
- O administrador deve informar os dados do novo usuário;
- O sistema deve confirmar a inclusão;

Como um usuário eu quero acessar o sistema;

- O usuário deve informar um login e senha;
- O sistema deve validar os dados e então redirecionar para o sistema;
- O sistema encontrando falha na autenticação, deve informar os erros ao usuário;

Como um administrador do sistema eu quero cadastrar uma pizza com tamanhos diferentes;

- O sistema deve mostrar os tamanhos de pizza cadastrados para o cadastro de pizza;
- O usuário deve adicionar os preços para os diferentes tamanhos de pizza;

Como um administrador do sistema eu quero cadastrar bebidas;

- O usuário deve selecionar o tipo de produto;
- O usuário deve informar os dados do produto;
- O sistema deve mostrar a confirmação do cadastro;

Como um usuário eu quero abrir um pedido, filtrando clientes por telefone ou nome.

- O usuário deve informar o telefone ou nome para filtrar;

¹*User Stories* são artefatos utilizados em sistema que utilizam metodologia de desenvolvimento ágil ([PRIMO, 2011](#)).

- O sistema deve mostrar um pedido apto a receber produtos;
Como um usuário eu quero poder inserir produtos em um pedido aberto;
- O usuário deve buscar o produto desejado;
- O usuário deve informar a quantidade;
- O sistema deve salvar o produto no pedido;
Como um usuário eu quero poder remover um produto do pedido;
- O usuário deve selecionar um pedido em aberto;
- O usuário deve selecionar o produtos;
- O usuário deve confirmar a exclusão do produto;
- O sistema deve atualizar o pedido;
Como um usuário eu quero fechar um pedido em aberto;
- O usuário deve selecionar um pedido em aberto;
- O usuário deve informar a forma de pagamento;
- O usuário deve confirmar o fechamento;

4.3 BASE DE DADOS

Com base nos requisitos informados, foi desenvolvida a base de dados para melhor se adequar o sistema. A base de dados foi elaborada no modelo relacional, onde as tabelas se relacionam umas com as outras. A Figura 11 apresenta a base de dados final que foi utilizada no sistema.

4.4 TESTE

A última etapa do desenvolvimento são os teste para garantir compatibilidades e confiabilidade no sistema. Foram feitos testes em dois navegadores bastante utilizados, *chrome* (Versão 61.0.3163.100 (Versão oficial) 64 bits) e *firefox* (Versão 54.0 (64-bit)), sendo compatíveis com ambos.

Foram feitos também testes unitários e funcionais, onde é verificado rotas para acesso ao sistema e teste de cadastro de cliente. Para executar esses testes foi utilizado uma ferramenta chamado *phpUnit* que vem por padrão no framework *laravel*, a Figura 12 mostra a execução destas tarefas.

Figura 11 – Estrutura da Base de Dados.



Fonte: O autor

Figura 12 – Teste unitário básico do sistema.

```
fweiber@acer:~/Documentos/fwpizza$ ./phpunit
PHPUnit 6.4.3 by Sebastian Bergmann and contributors.

.
.
.
Time: 2.69 seconds, Memory: 16.00MB

OK (11 tests, 11 assertions)
fweiber@acer:~/Documentos/fwpizza$ █
```

Fonte: O autor

5 SISTEMA

Este capítulo tem por finalidade apresentar o sistema desenvolvido e os recursos disponíveis nele.

O SI de pizzaria é uma ferramenta para auxiliar uma pizzaria, possibilitando e facilitando controle das operações. O sistema auxilia na venda de produtos da pizzaria, desde a retirada de balcão, entregas e vendas no local, pedidos de mesa. Na sequência serão apresentadas as funções do sistema e imagens ilustrativas.

5.1 CONFIGURAÇÕES, CONTROLE DE ACESSO e DASHBOARD

Ao acessar o sistema é necessário informar o e-mail e senha do usuário, assim que autenticado o usuário é levado para dentro do sistema onde pode acessar os módulos que possui permissão. Caso esqueça a senha pode-se redefinir a senha clicando em esqueci minha senha, assim é enviado um e-mail com link para redefinição de senha no sistema. O sistema já vem com um usuário com nível de administrador para o primeiro acesso.

Depois de acessado o sistema, o usuário pode entrar na tela de configurações para configurar os dados da pizzaria, como o nome da empresa, logotipo, endereço, quantidade de mesas que serão utilizadas. Só o administrador tem permissão de alterar esses dados.

No *Dashboard* aparecem informações atuais do sistema, como últimos pedidos, quantidade de pedidos em aberto, quantos pedidos saíram para entrega, e também oferece links para acesso direto a determinados cadastros e abertura de pedido. A Figura 13 mostra o dashboard do sistema.

Figura 13 – Dashboard do sistema.

The screenshot shows the FW Pizza system dashboard. On the left, there's a sidebar with navigation links: Dashboard, Cadastros, Relatórios, Pedidos (5 Abertos), Pedidos Mesa (2 Abertos), Painel de Pedidos, Pedidos para Entrega (2), and Pagamentos. The main area has a header 'DASHBOARD' and a user dropdown 'Administrador'. Below the header are four cards: 'Balcão' (2), 'Mesa' (2), 'Entrega' (3), and 'Total' (7). Each card has a 'VER...' button. Below these cards is a section titled 'Últimos pedidos' with a table:

ID	Cliente	Status	Abertura	Atualização	Situação
45	Felipe	Aberto	29/10/2017 20:32	04/11/2017 19:42	Esperando retirada
47	Mesa 2	Aberto	30/10/2017 01:52	30/10/2017 01:52	
34	Felipe	Aberto	27/10/2017 23:51	30/10/2017 01:45	Entregue

Fonte: O autor

5.2 CADASTROS

5.2.1 Cadastros de Usuários, Categorias e Taxas de entrega

No cadastro de usuário é informado e-mail, senha, nome, nível de acesso e função. O nível de acesso do usuário tem duas opções, administrador que possui acesso a todas as funções do sistema e usuário que só tem permissão para efetuar venda e cadastrar cliente na hora de abrir o pedido. A Figura 14 mostra o formulário para o cadastro de um novo usuário.

O cadastro de categoria possui somente o campo para inserir o nome da categoria. A categoria é utilizada para agrupar os produtos. A taxa de entrega é utilizada para facilitar no valor da entrega do pedido.

Figura 14 – Tela de cadastro de usuário.

Fonte: O autor

5.2.2 Cadastros de Clientes, Produtos e Promoções

Os cadastros de clientes, produtos e promoções são diferenciados, neles existem funções especiais para auxiliar no cadastro, como mostra a Figura 15.

No cadastro de cliente, ao informar o CEP, automaticamente o sistema pesquisa em um *webservice* chamado VIACEP¹, o endereço, bairro, cidade e estado, somente precisando informar o número da casa, também é feito uma busca das coordenadas de latitude e longitude do endereço, para poder se localizar mais facilmente no mapa.

¹VIACEP é um *webservice* para consultas de CEP gratuito, passando o CEP ele retorna o endereço, nos formatos *json*, *xml*, *piped* ou *query* ([VIACEP, 2017](#)).

Figura 15 – Tela de cadastro de cliente.

Fonte: O autor

No cadastro de promoções foi criada uma regra de promoção por semana, assim o usuário pode informar os dias em que será utilizado o preço promocional do produto. Na Figura 16 mostra o formulário para o cadastro de promoções.

Figura 16 – Tela de cadastro promoções.

Fonte: O autor

Para o cadastro de produto primeiramente é necessário informar o tipo de produto, se é pizza ou não, como mostra a Figura 17 no botão pizza, caso não seja pizza, o cadastro é normal, preenche-se os dados, onde são informados dois preços, um preço principal e outro preço promocional, por fim é salvo o produto, esses produtos são informados normalmente na tela de listagem. Se for pizza o cadastro é um pouco diferente, é cadastrado um produto genérico e é necessário informar os valores para os diferentes tamanhos de pizza, com isso são

salvos 4 produtos, um genérico e mais 3 produtos com características diferentes. Na listagem de produtos só aparece o produto genérico.

Também no cadastro de produto é possível inserir uma imagem para ilustrar o produto.

Figura 17 – Tela de cadastro de produto.

Tamanho	Código	Preço	Preço Promocional
Grande			
Médio			
Pequeno			

Fonte: O autor

5.3 PEDIDOS

Os pedidos são divididos em três tipos, podem ser para entrega, para retirada no balcão e podem ser feitos por mesa. Os pedidos de balcão e para entrega são parecidos. Para criar um pedido deve ser informado o cliente, caso não exista pode ser cadastrado um cliente na hora do pedido, se o pedido for do tipo de entrega, são automaticamente preenchidos os campos de endereço com os dados de endereço do cliente, podendo esse ser mudado, para pedidos do tipo balcão os campos de endereço não aparecem por não ser necessário.

A Figura 18 mostra o formulário para abrir um pedido, nele é possível escolher o tipo de pedido, se é para entrega ou para retirada no balcão.

Figura 18 – Tela de abertura de novo pedido.

Fonte: O autor

Para os pedidos do tipo mesa, há uma listagem das mesas, esses são mostrados de acordo as configurações de quantidade. Nesta listagem pode-se abrir um pedido ou solicitar pagamento, como mostra na Figura 19.

Figura 19 – Exibição das mesas.

Fonte: O autor

Depois de abertos os pedidos habilita-se a opção para inserção do produto, que pode ser buscado por meio do nome do produto ou do código, depois de selecionado é informado a quantidade do item desejado, caso o item buscado esteja dentro da promoção é mostrado o valor promocional. Na tela de pedido existe uma aba para o resumo, que informa os totais e dá opções para fechamento do pedido e solicitação do pagamento. Para os pedidos do tipo entrega, há um link para exibir o endereço no mapa para o entregador poder se localizar, como mostra na Figura 20.

Figura 20 – Resumo do pedido.

Código	Descrição	Quantidade	Preço
10	Pizza calabresa (Grande)	1.00	10.00
123	Coca cola	1.00	10.00
			Taxa de entrega: R\$ 5.00 Total: R\$ 25.00

FECHAR PEDIDO **PAGAR PEDIDO**

Fonte: O autor

Além da listagem das mesas, há a listagem de pedidos que estão em entrega e há um painel de pedidos, que mostra a última situação do pedido atualizada.

Para o pagamento de pedido há uma tela mostrada na Figura 21 que busca todos os pedidos em abertos que não possuem pagamentos, assim pode-se pagar e manter o pedido em aberto ou pagar e fechar o pedido.

Figura 21 – Tela de pagamento.

PAGAMENTO PEDIDO N°: 48

Meio de pagamento	Dinheiro
Valor a pagar	100,00
Valor Pago	100
Troco	0

CANCELAR **PAGAR**

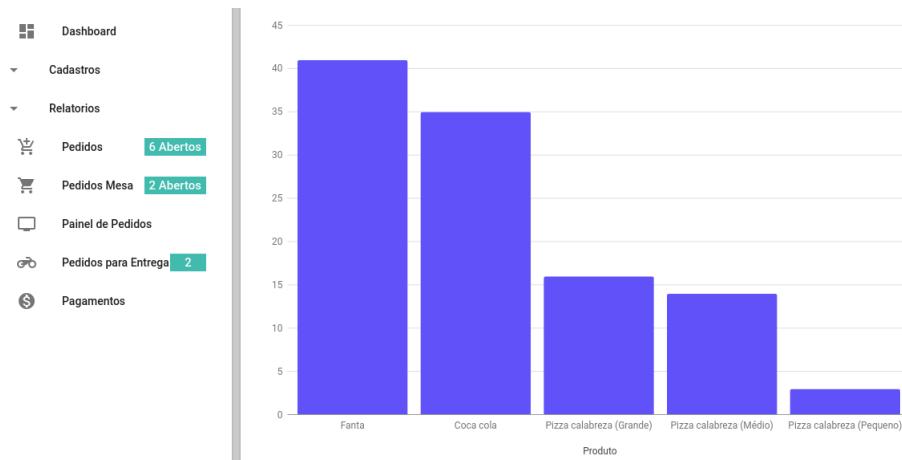
Fonte: O autor

5.4 RELATÓRIOS

Todos os relatórios possuem filtro por período.

O sistema possui relatórios para pedidos e produtos, o relatório de produto mostra quais os produtos mais vendidos, o resultado é mostrado em uma tabela e também exibido em um gráfico como é exibido na Figura 22.

Figura 22 – Gráfico dos produtos mais vendidos.



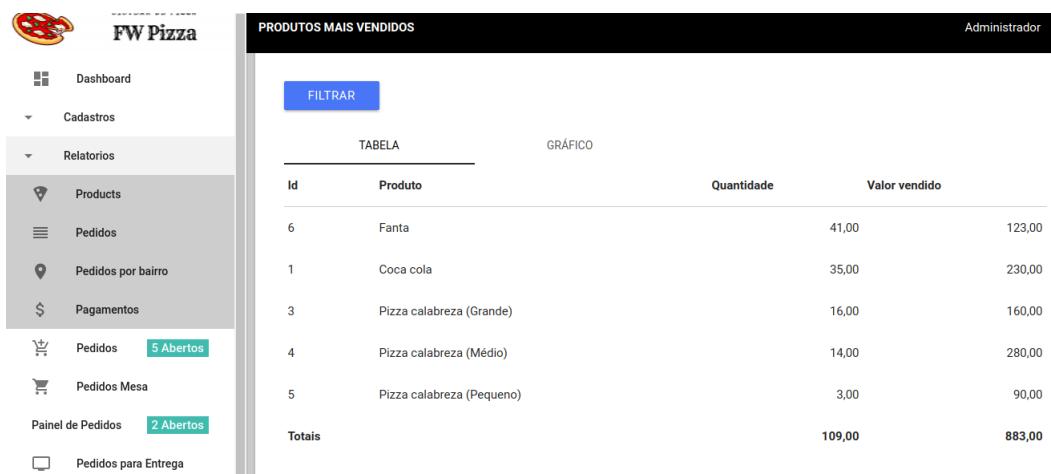
Fonte: O autor

Os relatórios de pedidos são dois, um mostra a quantidade de pedido efetuados por bairro, e o outro mostra a quantidade por tipo, mesa, entrega ou balcão.

Os relatórios de venda, mostram os valores de todos os pagamentos.

A Figura 23 mostra o relatório de produtos com seus totais, pode-se efetuar filtrar por um período específico.

Figura 23 – Relatório de produtos.



Fonte: O autor

6 CONCLUSÃO

No presente trabalho, inicialmente, foram analisados diversos *softwares* de gerenciamento de pizzaria existentes no mercado. Com isso, foram elencadas uma série de funcionalidades que foram utilizadas na implementação do sistema proposto. O objetivo inicial que era desenvolver uma aplicação capaz de atender não somente as demandas operacionais da pizzaria, mas também gerar uma série de informações estatísticas com capacidade de auxiliar no processo de tomada de decisões, foi alcançado.

O sistema desenvolvido pode ser utilizado em pizzarias para auxiliar no processo de gestão e com pequenas alterações também pode ser utilizado em outros estabelecimentos que fornecem serviço de vendas de produtos por local ou por entrega.

6.1 TRABALHOS FUTUROS

Para a continuidade desse trabalho podem ser implementadas outras funcionalidades, como um módulo mobile, um gerador de cardápios e integração com site institucional e fiscal.

Referências

- BINA. **BINA**. 2009. Disponível em: <<http://www.dicionarioinformal.com.br/bina/>>. Acesso em: 12 de abril de 2017. Citado na página 6.
- ECLETICA. **Soluções para Pizzarias**. 2016. Disponível em: <<http://www.ecletica.com.br/solucoes-para-food-service/pizzarias/>>. Acesso em: 2 de abril de 2017. Citado na página 6.
- FOUNDATION, A. S. **APACHE Site oficial**. 2017. Disponível em: <<https://www.apache.org/>>. Acesso em: 27 de maio de 2017. Citado na página 3.
- FOUNDATION, T. jQuery. **DataTables**. 2017. Disponível em: <<https://jquery.com/>>. Acesso em: 2 de novembro de 2017. Citado na página 6.
- GOOGLE. **Google Chart**. 2017. Disponível em: <<https://developers.google.com/chart/>>. Acesso em: 5 de abril de 2017. Citado na página 6.
- MATERIALIZE. **Site Oficial, 2017**. 2017. Disponível em: <<http://materializecss.com/>>. Acesso em: 31 de maio de 2017. Citado na página 5.
- ORACLE. **ORACLE Site oficial**. 2017. Disponível em: <<https://www.oracle.com/br/mysql/index.html>>. Acesso em: 27 de maio de 2017. Citado na página 3.
- PHP. **PHP Site oficial**. 2017. Disponível em: <https://secure.php.net/manual/pt_BR/intro-whatis.php>. Acesso em: 20 de maio de 2017. Citado na página 4.
- PIZZARIAFREE. **Pizzaria Free**. 2015. Disponível em: <<http://programaparapizzaria.com.br/>>. Acesso em: 27 de março de 2017. Citado na página 7.
- PRIMO, G. **User Stories – O que são? Como Usar?** 2011. Disponível em: <<http://blog.myscrumhalf.com/2011/10/user-stories-o-que-sao-como-usar/>>. Acesso em: 31 de maio de 2017. Citado na página 15.
- RAL. **Programa para Pizzarias e Delivery Consumer**. 2017. Disponível em: <<https://www.programaconsumer.com.br/programa-para-pizzarias>>. Acesso em: 27 de março de 2017. Citado na página 9.
- REYNOLDS, G.; STAIR, R. **Princípios de sistemas de informação**. São Paulo: Cengage Learning, 2011. Citado na página 1.
- SESHADRI, S.; GREEN, B. **Desenvolvendo com AngularJS: Aumento de Produtividade com Aplicações Web Estruturadas**. Novatec Editora, 2014. ISBN 9788575224090. Disponível em: <<https://books.google.com.br/books?id=ZEgcBQAAQBAJ>>. Citado na página 5.
- SYSPIZZA. **SysPizza**. 2017. Disponível em: <<http://www.syspizza.com.br/>>. Acesso em: 27 de março de 2017. Citado na página 8.
- TURINI, R. **PHP e Laravel Crie aplicações web como um verdadeiro artesão**. [S.I.]: Casa Do Código, 2015. Citado na página 4.
- VIACEP. **VIACEP**. 2017. Disponível em: <<https://viacep.com.br>>. Acesso em: 05 de novembro de 2017. Citado na página 20.

W3C. **Site Oficial, 2017.** 2017. Disponível em: <<https://www.w3schools.com/css/>>. Acesso em: 31 de maio de 2017. Citado na página 5.

ZERVAAS, Q. **Aplicações Práticas de Web 2.0.** [S.I.]: Alta Books, 2008. Citado na página 3.