



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

JUNIOR APARECIDO RODRIGUES DA CRUZ

**SISTEMA DE GESTÃO DE PEDIDOS PARA RESTAURANTES E
LANCHONETES**

Assis, SP

2015

JUNIOR APARECIDO RODRIGUES DA CRUZ

**SISTEMA DE GESTÃO DE PEDIDOS PARA RESTAURANTES E
LANCHONETES**

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, analisado pela seguinte comissão examinadora:

Orientador: Célio Desiró.

Área de Concentração: Análise e Desenvolvimento de Sistemas.

Assis, SP

2015

FICHA CATALOGRÁFICA

Cruz, Junior Aparecido Rodrigues.

Sistema de Gestão de Pedidos para Restaurantes e Lanchonetes / Junior Aparecido Rodrigues da Cruz. Fundação Educacional do Município de Assis – FEMA – Assis, 2015.
46p.

Orientador: Prof. Esp. Célio Desiró

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis – IMESA

1. Android 2. C# 3. Mobilidade

CDD: 001.61
Biblioteca da FEMA

SISTEMA DE GESTÃO DE PEDIDOS PARA RESTAURANTES E LANCHONETES

JUNIOR APARECIDO RODRIGUES DA CRUZ

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, analisado pela seguinte comissão examinadora:

Orientador: Célio Desiró.

Analisador: Douglas Sanches da Cunha.

Assis, SP

2015

DEDICATÓRIA

Dedico este Trabalho

Primeiramente a Deus, por tudo que me proporciona, a minha namorada e minha família pelo apoio constante e a todas as pessoas que acreditaram em minha capacidade.

AGRADECIMENTOS

Agradeço a Deus por ter me dado sabedoria, saúde e muita força para seguir em frente durante toda essa jornada.

A minha namorada por ter tido muita paciência durante todo o desenvolvimento deste trabalho e pelo apoio incondicional que tem me dado.

Ao meu orientador Célio Desiró, e todos os professores que me apoiaram e caminharam comigo durante toda essa fase de aprendizado.

Aos meus familiares que me deram muita força durante todo o curso.

Agradeço os meus amigos e as pessoas que ajudaram direta ou indiretamente para conclusão deste trabalho.

“Não confunda derrotas com fracasso nem vitórias com sucesso. Na vida de um campeão sempre haverá algumas derrotas, assim como na vida de um perdedor sempre haverá vitórias. A diferença é que, enquanto os campeões crescem nas derrotas, os perdedores se acomodam nas vitórias.”

Roberto Shinyashiki

RESUMO

A tecnologia é uma das principais aliadas dos empreendedores que sonham em alavancar seus negócios sem precisar gastar muito dinheiro. Isso também é verdadeiro nos processos de negócios que, quando automatizados, podem fazer com que as atividades se tornem mais ágeis e produtivas, demandando menos tempo e esforço dos funcionários e setores responsáveis.

Quanto mais ágil, rápido e eficiente for o atendimento, maior é a satisfação do cliente.

Este projeto tem como objetivo automatizar o processo de pedido e entrega em restaurantes, lanchonetes e afins, visando diminuir as perdas por erros de anotação, entre outros problemas que resultam em uma longa espera dos clientes, causando constrangimentos e estresse.

Em substituição ao papel e caneta, todos os pedidos serão feitos pelo garçom utilizando um tablet ou um smartphone com sistema operacional Android, possibilitando que os pedidos sejam enviados e distribuídos a lugares diferentes sem que os garçons tenham que levá-los, aumentando, conseqüentemente, a eficiência e o tempo disponível para atender outros clientes.

Palavras-chave: Android, C#, Mobilidade.

ABSTRACT

Technology is one of the main allies of entrepreneurs who dream of promote their business without spending too much money. This is also true in business processes that, when automated, can make the activities become more agile and productive, requiring less time and effort of employees and responsible sectors.

The more agile, fast and efficient the service, greater customer satisfaction.

This project aims to automate the process of order and delivery in restaurants, snack bars and the like in order to reduce losses due to clerical error, or other problems that result in a long wait of customers, causing embarrassment and stress.

Instead of pen and paper, all the requests will be made by the waiter using a tablet or a smartphone with Android operating system, enabling these requests to be sent to different places without the waiters have to take them, increasing consequently the time available to serve other clients.

Key-words: Android, C#, Mobility.

LISTA DE ILUSTRAÇÕES

Figura 1 – WBS	21
Figura 2 - Mapa Mental do Sistema Desktop	22
Figura 3 - Mapa mental do aplicativo mobile	23
Figura 4 - Diagrama de Caso de Uso Geral	24
Figura 5 – Diagrama Caso de Uso Fazer Login	25
Figura 6 – Diagrama Caso de Uso Cadastrar Produto	26
Figura 7 - Diagrama de Caso de Uso Cadastrar Categoria.....	27
Figura 8 - Diagrama de Caso de Uso Cadastrar Adicionais	28
Figura 9 - Diagrama de Caso de Uso Cadastrar Garçon	29
Figura 10 - Diagrama de Caso de Uso Fazer Pedido.....	30
Figura 11 - Diagrama de Caso de Uso Consultar Conta	31
Figura 12 - Diagrama de Caso de Uso Solicitar Fechamento da Comanda.....	32
Figura 13 - Diagrama Entidade e Relacionamento.....	33
Figura 14 - Diagrama de Atividade.....	34
Figura 15 - Diagrama de Classe.....	35
Figura 16 - Tela de Login	36
Figura 17 - Tela das Mesas.....	37
Figura 18 - Tela de Categoria.....	38
Figura 19 - Tela do Pedido	39
Figura 20 - Tela de login	40
Figura 21 - Tela principal.....	40
Figura 22 - Tela do Caixa.....	41
Figura 23 - Tela das Mesas.....	41

LISTA DE TABELAS

Tabela 1 - Especificação de Caso de Uso Fazer Login.....	25
Tabela 2 - Especificação de Caso de Uso Cadastrar Produto	26
Tabela 3 - Especificação de Caso de Uso Cadastrar Categoria	27
Tabela 4 - Especificação de Caso de Uso Cadastrar Adicionais.....	28
Tabela 5 - Especificação de Caso de Uso Cadastrar Garçom	29
Tabela 6 - Especificação de Caso de Uso Fazer Pedido	30
Tabela 7 - Especificação de Caso de Uso Consultar Conta.....	31
Tabela 8 - Especificação de Caso de Uso Solicitar Fechamento da Comanda	32

SUMÁRIO

1. INTRODUÇÃO	14
1.1 OBJETIVO	15
1.2 PÚBLICO ALVO.....	15
1.3 JUSTIFICATIVA.....	15
1.4 ESTRUTURA DO TRABALHO	16
2. TECNOLOGIAS E FERRAMENTAS UTILIZADAS.....	17
2.1 METODOLOGIA DE ANÁLISE	17
2.2 DESENVOLVIMENTO DO SISTEMA.....	17
2.3 ARMAZENAMENTO E RECUPERAÇÃO DE DADOS	19
2.4 WEB SERVICES.....	19
3. PROJETO DO SISTEMA.....	21
4. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA	22
4.1 MAPA MENTAL DO SISTEMA	22
4.2 DIAGRAMAS DE CASO DE USO	24
4.2.1 Caso de Uso Geral	24
4.2.2 Fazer Login.....	25
4.2.3 Cadastrar Produto.....	26
4.2.4 Cadastrar Categoria.....	27
4.2.5 Cadastrar Adicionais.....	28
4.2.6 Cadastrar Garçom	29
4.2.7 Fazer Pedido.....	30
4.2.8 Consultar Conta.....	31
4.2.9 Solicitar Fechamento da Comanda.....	32
4.3 DIAGRAMA ENTIDADE E RELACIONAMENTO.....	33
4.4 DIAGRAMA DE ATIVIDADES.....	34
4.5 DIAGRAMA DE CLASSE.....	35

5. RESULTADOS.....	36
5.1 APLICATIVO MOBILE	36
5.1.1 Tela de Login	36
5.1.2 Tela das Mesas.....	37
5.1.3 Tela de Categoria	38
5.1.4 Tela do Pedido.....	39
5.2 SISTEMA DESKTOP	40
5.2.1 Tela de Login	40
5.2.2 Tela Principal	40
5.2.3 Tela do Caixa.....	41
5.2.4 Tela das Mesas.....	41
6. CONCLUSÃO	42
ANEXO I	44

1. INTRODUÇÃO

A tecnologia tem evoluído com tamanha agilidade e aprimoramento, que nos faz refletir sobre o quanto precisamos estar atualizados para podermos acompanhar e viabilizar sistemas que possam proporcionar melhor qualidade de vida para as pessoas.

Na atualidade, uma das coisas mais importantes nas nossas vidas é o tempo. Pensando nisso e no grande crescimento no setor alimentício, percebe-se a necessidade de melhorar o atendimento ao consumidor. Novas técnicas surgem a todo o momento, como por exemplo, o uso dos dispositivos móveis. Com eles é possível proporcionar um atendimento com maior capacidade de abrangência e rapidez.

Restaurantes e lanchonetes, na maioria dos casos, ainda utilizam o tradicional formulário em papel para anotar os pedidos. Este método muitas vezes se mostra ineficiente em dias de muito movimento gerando diversos problemas.

Pedidos anotados errados, com rasuras ou ilegíveis, dificuldade de leitura pelos cozinheiros, deslocamento excessivo dos garçons entre o cliente e a cozinha são alguns problemas que ocorrem no cotidiano destes estabelecimentos, muitas vezes causando ineficiência no atendimento e insatisfação do cliente.

Analisando este cenário, o projeto apresenta um sistema que automatiza os pedidos de lanchonetes e restaurantes para maior praticidade e qualidade do serviço.

1.1 OBJETIVO

O objetivo deste trabalho é apresentar o desenvolvimento de um sistema de gestão de pedidos para restaurantes ou lanchonetes desenvolvido na plataforma Android, na busca de agilidade e um atendimento diferenciado.

Os principais objetivos do trabalho são:

- Disponibilizar um aplicativo *mobile* para gestão do pedido com o cardápio completo.
- Disponibilizar no aplicativo a consulta da conta do cliente e o que foi consumido em tempo real.
- Disponibilizar para o administrador do sistema a manutenção do cardápio e o fechamento da conta de cada cliente.

1.2 PÚBLICO ALVO

Este trabalho se destina a estabelecimentos do ramo gastronômico, como restaurantes e lanchonetes, que queiram informatizar e inovar o atendimento, proporcionar um serviço ágil, minimizando os erros e constrangimentos gerados pelo método manual de realizar o pedido.

1.3 JUSTIFICATIVA

A maioria dos restaurantes ainda utiliza o tradicional bloco de papel para anotar o pedido, acarretando diversos problemas como:

- O produto recebido é diferente do que foi pedido pelo cliente.
- O atendimento é muito demorado.
- O papel chega à cozinha rasurado e ilegível.
- O garçom precisa obrigatoriamente levar o pedido até a cozinha.
- O garçom precisa ter uma boa escrita para que seja entendido pelos cozinheiros.

Estas são algumas situações rotineiras, principalmente em dias de grande movimento. A automatização do processo de pedidos e na entrega dos produtos pode reduzir bastante essas falhas, garantindo uma maior segurança e rapidez no atendimento.

1.4 ESTRUTURA DO TRABALHO

Este trabalho está estruturado em seis capítulos:

O primeiro capítulo apresenta os objetivos, público alvo e justificava, demonstrando a proposta do trabalho.

O segundo capítulo aborda as tecnologias utilizadas no levantamento de requisitos e no desenvolvimento do sistema.

O terceiro capítulo Apresenta o WBS – *Work Breakdown Structure* e o cronograma utilizado para o desenvolvimento do software.

O quarto capítulo se concentra na especificação do sistema, no mapa mental e nos diagramas (caso de uso, entidade e relacionamento, atividades e classe).

O quinto capítulo demonstra o resultado do trabalho, com algumas telas do sistema desenvolvido.

O sexto capítulo é a conclusão geral do projeto.

2. TECNOLOGIAS E FERRAMENTAS UTILIZADAS

Neste capítulo são abordadas as tecnologias e ferramentas utilizadas para fazer a documentação e o desenvolvimento do sistema.

2.1 METODOLOGIA DE ANÁLISE

A linguagem utilizada para fazer a modelagem do sistema foi a UML. A Unified Modelling Language (UML) é uma linguagem ou notação de diagramas para especificar, visualizar e documentar modelos de 'software' orientados a objetos. Seu objetivo é auxiliar a visualizar o desenho e a comunicação entre objetos. Ela permite que desenvolvedores visualizem os produtos de seu trabalho em diagramas padronizados, e é muito usada para criar modelos de sistemas de software e tem uma utilização limitada para outros paradigmas de programação. A UML não fornece suporte semântico e visual que substitua a linguagem de programação, ou seja, ela não está orientada a nenhum código e não é uma linguagem de programação (LIMA, 2012). O *Astah-Community* foi utilizado para fazer a modelagem do sistema.

2.2 DESENVOLVIMENTO DO SISTEMA

O projeto é composto de dois sistemas, a versão *mobile*, a ser utilizada pelos garçons para anotar os pedidos dos clientes e então enviar para ser impresso na cozinha ou no bar, e a versão *desktop* para o controle das mesas e fechamento dos pedidos.

A versão *mobile* foi desenvolvida com a linguagem de programação java. Para desenvolver a versão desktop foi utilizada a ferramenta Visual Studio 2013, plataforma da Microsoft destinada a desenvolvedores que trabalham com a linguagem de programação C# e com o framework .NET. Sua principal função é auxiliar programadores na criação de aplicações para o Windows, e, sendo ele o

sistema dominante do mercado, o Visual Studio é a plataforma mais popular do mundo (TECHTUDO, 2015).

Junto com essa ferramenta foi utilizada a linguagem de programação C# (lê-se “cê sharp”), uma linguagem de programação orientada a objetos, criada junto com a arquitetura da plataforma .NET da Microsoft. Grande parte das classes do Framework .NET foram escritas com essa linguagem.

Diversos desenvolvedores participaram do projeto de sua criação, porém o principal envolvido no projeto foi o engenheiro Anders Hejlsberg que era desenvolvedor de compiladores na Borland, que além do C# também criou Turbo Pascal e o Delphi.

A linguagem C# foi criada influenciada pelas linguagens C++ e Java. As inovações do C# permitem escrita de aplicativos de maneira rápida, mantendo a expressividade e elegância das linguagens C-Style. A sintaxe utilizada pelo C# é relativamente fácil, o que diminui o tempo de aprendizado.

Depois que entende-se como funciona, não há mais motivos para utilizar outra linguagem complicada, pois ela tem o poder do C++ e é simples como o Visual Basic (LOTAR, 2015).

A plataforma *mobile* escolhida para desenvolver o aplicativo é o Android. Android é um sistema operacional baseado em Linux. Em outras palavras o Android é um sistema operacional desenvolvido para dispositivos móveis que permite a seus desenvolvedores criarem aplicações na linguagem de programação Java.

Esta plataforma teve seu desenvolvimento iniciado em 2003 pela empresa Android Inc. Em 2005, o Google adquiriu a empresa que hoje lidera o desenvolvimento do Android. Porém não é apenas o Google que colabora com o seu desenvolvimento, existe um grupo formado por grandes empresas como a Motorola, LG, Samsung, Sony Ericson e muitas outras. Esse grupo é chamado de Open Handset Alliance (OHA), a formação desse grupo foi com a intenção de padronizar uma plataforma de código aberto e livre para celulares. Além de contar com a participação dessas empresas, o código que forma o sistema operacional Android também tem recebido contribuições de programadores espalhados por todo o mundo, uma vez que ele é software aberto.

A plataforma Android desfruta hoje de um papel de destaque no mercado, tanto pela quantidade significativa de dispositivos produzidos como também por oferecer uma API rica, disponibilizando fácil acesso a vários recursos de hardware, tais como Câmera Wi-Fi e GPS, além de boas ferramentas para o desenvolvedor (MONTEIRO 2012).

O aplicativo *mobile* foi desenvolvido utilizando a IDE Eclipse com o *plug-in* Android Developer Tools (ADT) junto com o Android SDK. O *Software Development Kit* (Kit de Desenvolvimento de Software em tradução livre) inclui diversas ferramentas. Dentre elas se destacam o emulador do sistema android, ferramentas de *debugging* e o editor de componentes visuais, além de fornecer uma API completa para a linguagem Java.

2.3 ARMAZENAMENTO E RECUPERAÇÃO DE DADOS

Para o armazenamento das informações foi utilizado o banco de dados Microsoft SQL Server.

O MS SQL Server criado por volta de 1988, é um gerenciador de Banco de dados relacional feito pela Microsoft. É um Banco de dados robusto e usado por sistemas corporativos dos mais diversos portes. Sua principal função é a de armazenar e recuperar dados solicitados por outras aplicações de software seja aqueles no mesmo computador ou aqueles em execução em outro computador através de uma rede (incluindo a Internet). A SQL permite o aumento de produtividade operacional no desenvolvimento de sistemas, tem uma estrutura sintática muito próxima da linguagem humana, por isso se tornou a linguagem básica utilizada em vários sistemas de gerenciamento de banco de dados (MANZANO, 2012).

2.4 WEB SERVICES

Com o avanço da internet e o intuito de interligar sistemas heterogêneos surgiu a tecnologia que chamamos de web services, proveniente de um consórcio formado

por grandes empresas como IBM, Microsoft e BEA, entre outras pertencentes ao W3C.

Segundo Adorno (2014, p.14):

Os web services são uma tecnologia de integração de sistemas empregada principalmente em ambientes heterogêneos, traduzindo: utilizando essa tecnologia, podemos desenvolver softwares ou componentes de softwares capazes de interagir, seja enviando ou recebendo informações, com outros softwares, não importando a linguagem de programação em que estes foram desenvolvidos, o sistema operacional em que rodam e o hardware que é utilizado.

Houve a necessidade de utilizar essa tecnologia para realizar a comunicação do aplicativo android com o sistema desktop. A linguagem utilizada para criação do web service foi o php por ser uma linguagem open source e de aprendizagem.

PHP é uma linguagem de criação de scripts embutida em HTML, usada originalmente apenas para o desenvolvimento de aplicações presentes e atuantes no lado do servidor.

Os principais produtos existentes no mercado são as Active Server Pages (ASP) da Microsoft, o Coldfusion da Allaire e as Java Server Pages (JSP), da Sun. O PHP é, às vezes, chamado de “o ASP de código fonte Aberto” (CONVERSE e PARK 2001, p.03).

3. PROJETO DO SISTEMA

Este capítulo aborda as fases do desenvolvimento do projeto por meio de uma WBS e apresenta o cronograma adotado para o desenvolvimento do software.

A figura 1 demonstra a WBS – Work Breakdown Structure – uma técnica que consiste em decompor um projeto em seus elementos componentes. É uma imagem gráfica da hierarquia do projeto, decomposto nível a nível em subprojetos e finalmente em tarefas (DAYCHOUW, 2007).



Figura 1 – WBS

4. ANÁLISE E ESPECIFICAÇÃO DO SISTEMA

O Levantamento de requisitos e a documentação do sistema é a parte mais importante do projeto. Este capítulo apresenta a especificação e os componentes do mesmo.

4.1 MAPA MENTAL DO SISTEMA

Mapas Mentais apoiam a organização de ideias, de conhecimento através de uma visualização intuitiva e amigável, além de apresentarem grande versatilidade visual, são considerados um padrão mundial para criação, gerenciamento e comunicação de ideias (VARGAS, 2007).

A Figura 2 e a 3 respectivamente mostra o Mapa mental do aplicativo *Mobile* e do sistema *Desktop*.

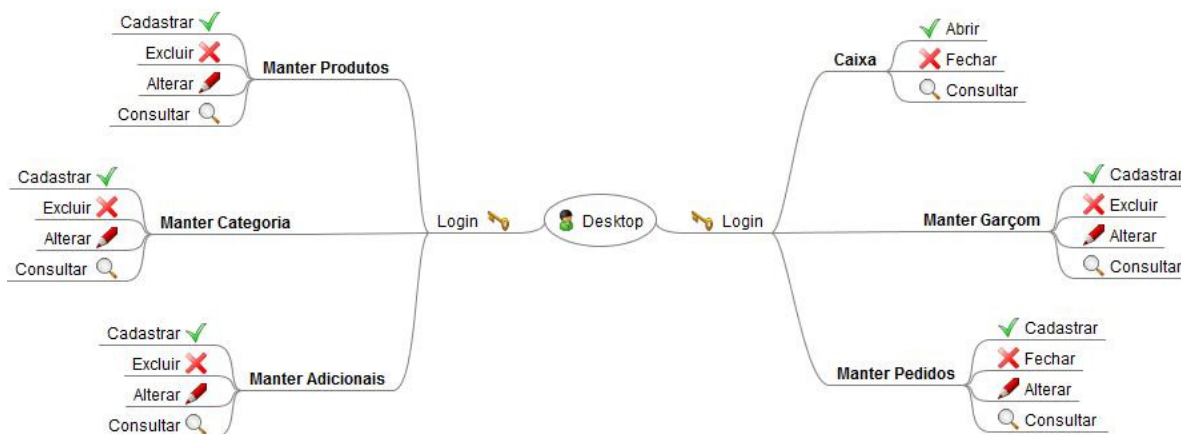


Figura 2 - Mapa Mental do Sistema Desktop

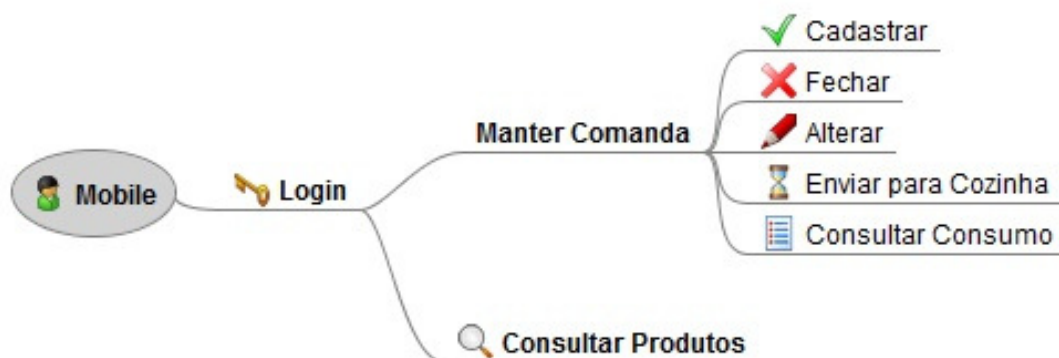


Figura 3 - Mapa mental do aplicativo mobile

4.2 DIAGRAMAS DE CASO DE USO

Um caso de uso descreve uma sequência de ações que representam um cenário principal (perfeito) e cenários alternativos, com o objetivo de demonstrar o comportamento de um sistema (ou parte dele), através de interações com atores (MELO, 2010).

4.2.1 Caso de Uso Geral

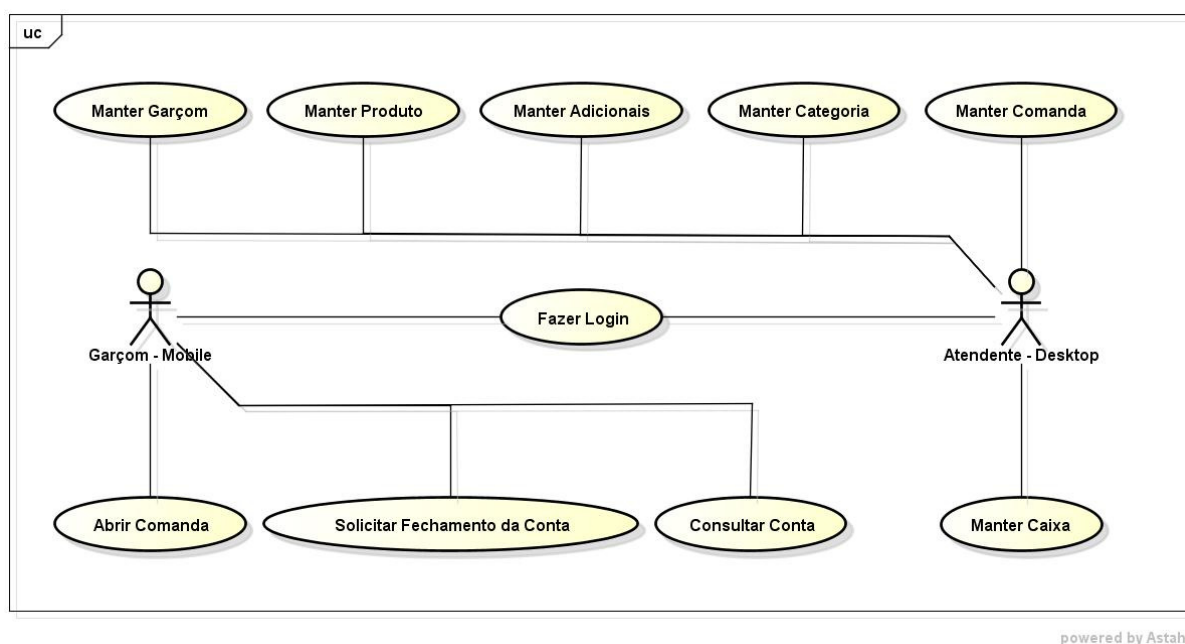
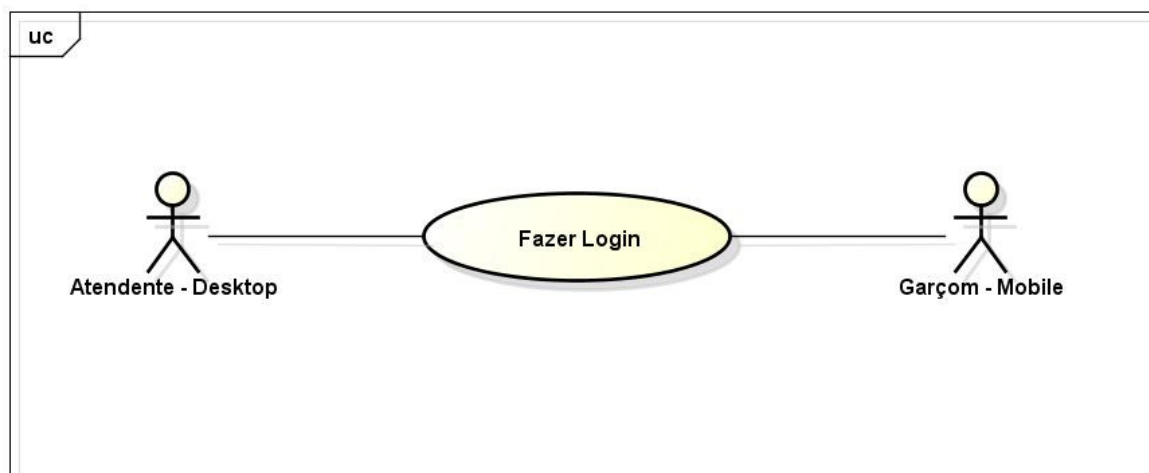


Figura 4 - Diagrama de Caso de Uso Geral

4.2.2 Fazer Login



powered by Astah

Figura 5 – Diagrama Caso de Uso Fazer Login

Nome do Caso de Uso	Fazer Login
Ator(es)	Atendente, Garçom
Resumo	Permite o usuário acessar o sistema
Pré Condições	O usuário deve conter um registro no sistema
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no sistema.
Fluxo Principal	1. Usuário informa nome de login e senha. 2. O Sistema verifica os registros informados, e libera o acesso [A1] .
Fluxos Alternativos A1	1. Usuário cancela a entrada ao sistema. 2. O Sistema encerra.
Fluxos de Exceções	1. Usuário digita login ou senha incorreta. 2. O Sistema emite mensagem de erro, e cancela a entrada.

Tabela 1 - Especificação de Caso de Uso Fazer Login.

4.2.3 Cadastrar Produto

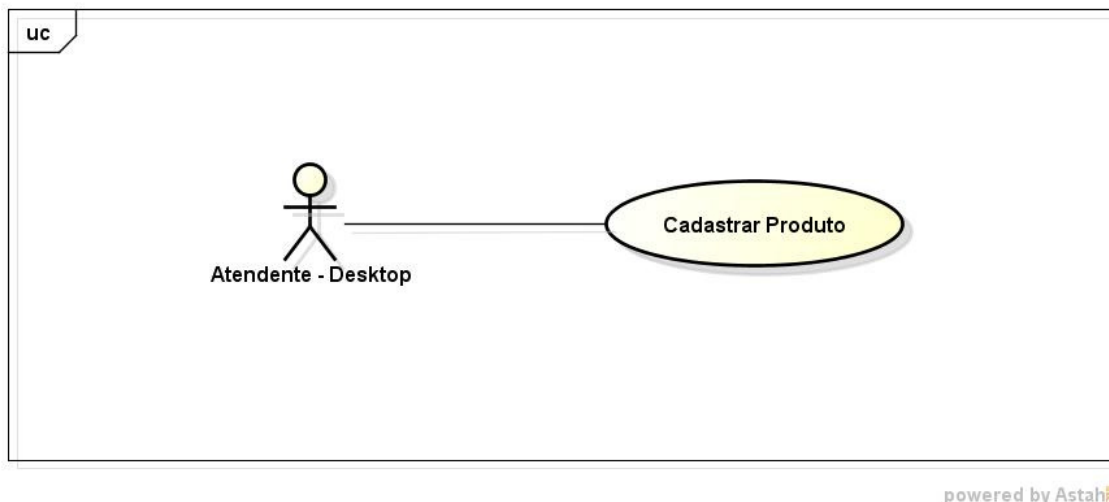


Figura 6 – Diagrama Caso de Uso Cadastrar Produto

Nome do Caso de Uso	Cadastrar Produto
Ator(es)	Atendente
Resumo	Cadastrar um Novo Produto
Pré Condições	Usuário deve estar conectado no Sistema
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. Clicar no menu Cadastrar Produto. 2. Sistema irá abrir o formulário para o cadastramento. 3. Usuário deverá preencher todos os campos, e clicar em salvar. 4. Sistema irá emitir uma mensagem confirmando o cadastro. 5. Usuário clica em "OK" [A1].
Fluxos Alternativos A1	<ol style="list-style-type: none"> 1. Usuário cancela o cadastro de produto. 2. O Sistema fecha a janela de cadastro.
Fluxos de Exceções	<ol style="list-style-type: none"> 1. Usuário não preenche os campos corretamente. 2. O Sistema emite uma mensagem de erro.

Tabela 2 - Especificação de Caso de Uso Cadastrar Produto

4.2.4 Cadastrar Categoria

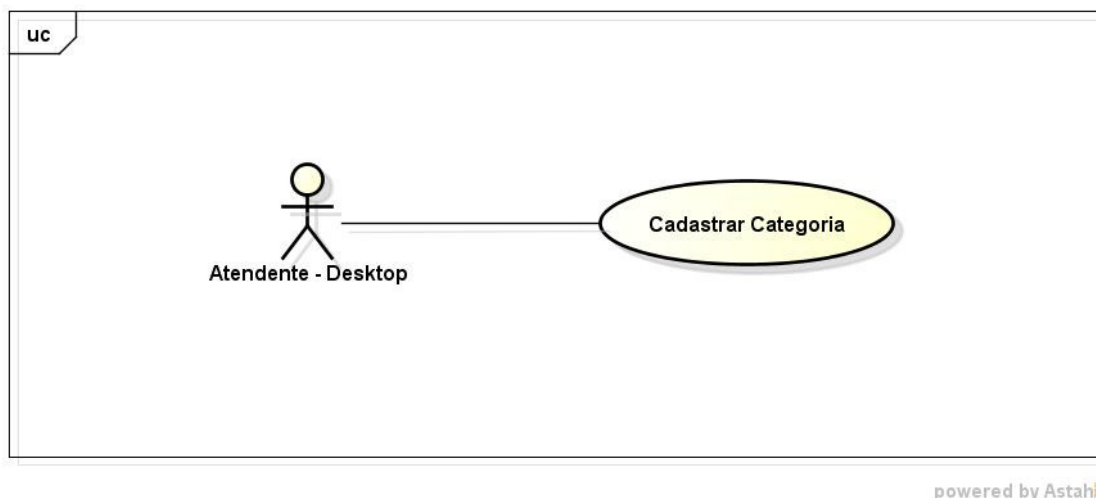


Figura 7 - Diagrama de Caso de Uso Cadastrar Categoria

Nome do Caso de Uso	Cadastrar Categoria
Ator(es)	Atendente
Resumo	Cadastrar uma Nova Categoria de Produto
Pré Condições	Usuário deve estar conectado no Sistema
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. Clicar no menu Cadastrar Categoria. 2. Sistema irá abrir o formulário para o cadastramento. 3. Usuário deverá preencher todos os campos, e clicar em salvar. 4. Sistema irá emitir uma mensagem confirmando o cadastro. 5. Usuário clica em "OK" [A1].
Fluxos Alternativos A1	<ol style="list-style-type: none"> 1. Usuário cancela o cadastro de Categoria. 2. O Sistema fecha a janela de cadastro.
Fluxos de Exceções	<ol style="list-style-type: none"> 1. Usuário não preenche os campos corretamente. 2. O Sistema emite uma mensagem de erro.

Tabela 3 - Especificação de Caso de Uso Cadastrar Categoria

4.2.5 Cadastrar Adicionais

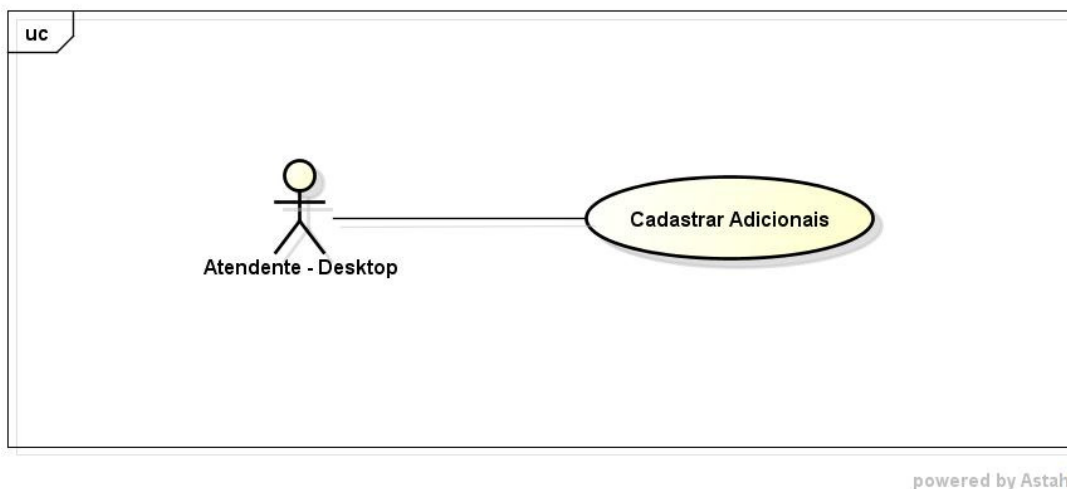
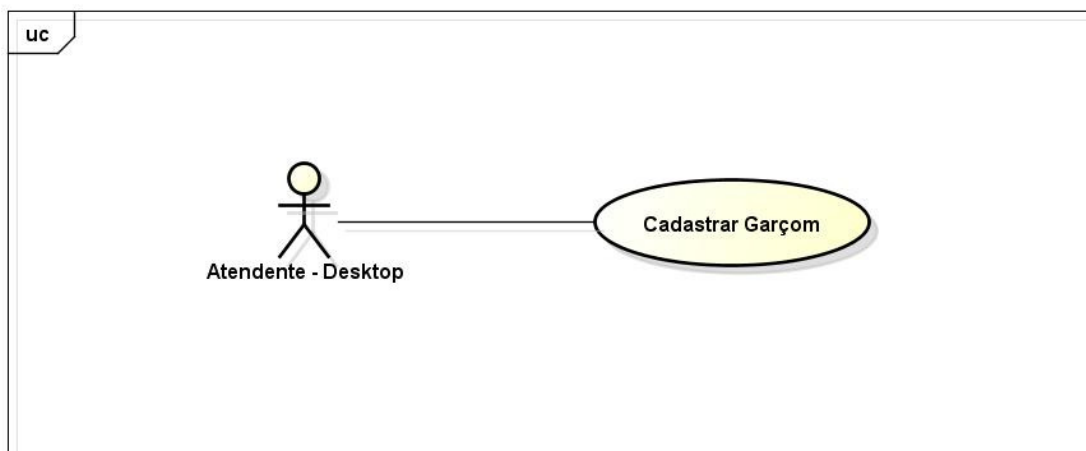


Figura 8 - Diagrama de Caso de Uso Cadastrar Adicionais

Nome do Caso de Uso	Cadastrar Adicionais
Ator(es)	Atendente
Resumo	Cadastrar Adicionais para lanches, pizzas etc.
Pré Condições	Usuário deve estar conectado no Sistema
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. Clicar no menu Cadastrar Adicional. 2. Sistema irá abrir o formulário para o cadastramento. 3. Usuário deverá preencher todos os campos, e clicar em salvar. 4. Sistema irá emitir uma mensagem confirmando o cadastro. 5. Usuário clica em "OK" [A1].
Fluxos Alternativos A1	<ol style="list-style-type: none"> 1. Usuário cancela o cadastro de Adicional. 2. O Sistema fecha a janela de cadastro.
Fluxos de Exceções	<ol style="list-style-type: none"> 1. Usuário não preenche os campos corretamente. 2. O Sistema emite uma mensagem de erro.

Tabela 4 - Especificação de Caso de Uso Cadastrar Adicionais

4.2.6 Cadastrar Garçom



powered by Astah

Figura 9 - Diagrama de Caso de Uso Cadastrar Garçom

Nome do Caso de Uso	Cadastrar Garçom
Ator(es)	Atendente
Resumo	Cadastrar Garçom.
Pré Condições	Usuário deve estar conectado no Sistema
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no sistema.
Fluxo Principal	<ol style="list-style-type: none"> 1. Clicar no menu Cadastrar Garçom. 2. Sistema irá abrir o formulário para o cadastramento. 3. Usuário deverá preencher todos os campos, e clicar em salvar. 4. Sistema irá emitir uma mensagem confirmando o cadastro. 5. Usuário clica em "OK" [A1].
Fluxos Alternativos A1	<ol style="list-style-type: none"> 1. Usuário cancela o cadastro de Garçom. 2. O Sistema fecha a janela de cadastro.
Fluxos de Exceções	<ol style="list-style-type: none"> 1. Usuário não preenche os campos corretamente. 2. O Sistema emite uma mensagem de erro.

Tabela 5 - Especificação de Caso de Uso Cadastrar Garçom

4.2.7 Fazer Pedido

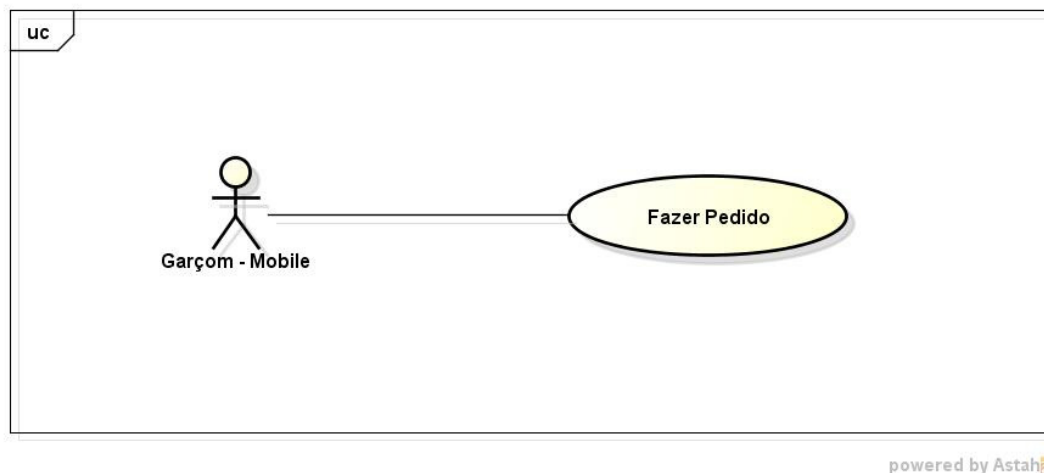
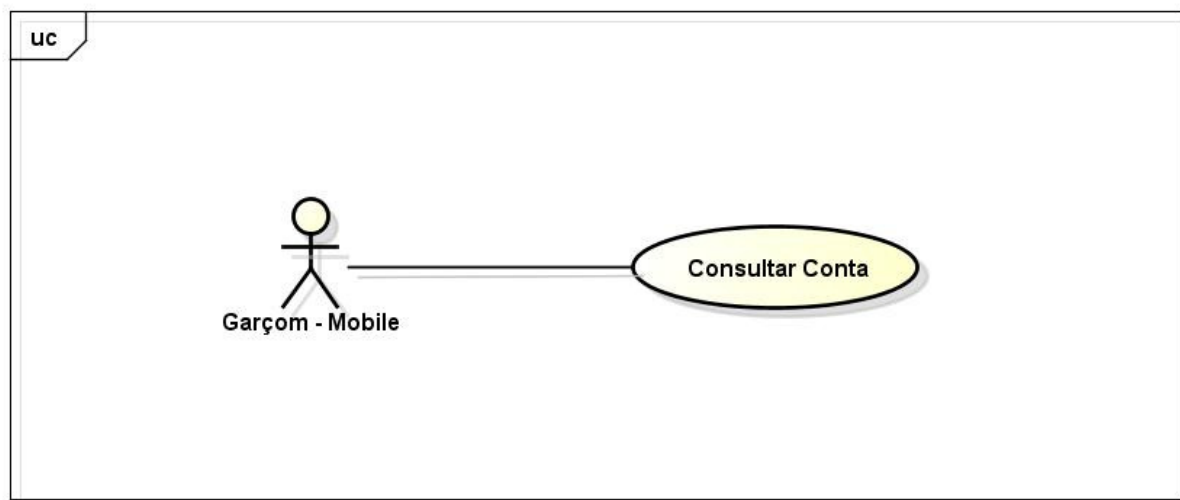


Figura 10 - Diagrama de Caso de Uso Fazer Pedido

Nome do Caso de Uso	Fazer Pedido
Ator(es)	Garçom
Resumo	Fazer um Pedido
Pré Condições	Usuário deve estar conectado no aplicativo
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no aplicativo
Fluxo Principal	<ol style="list-style-type: none"> 1. Garçom seleciona uma mesa 2. O aplicativo irá abrir as categorias dos produtos. 3. Garçom seleciona a categoria desejada. 4. O aplicativo abre os produtos referentes à categoria selecionada. 5. Garçom seleciona o produto desejado. 6. O aplicativo abre a janela para seleção de quantidade, e para especificar alguma observação. 7. Garçom seleciona a quantidade e se necessário especifica alguma observação e clica em OK [A1]. 8. O Item é adicionado ao carrinho para ser enviado para preparo. 9. O ícone da mesa selecionada é alterado para cor vermelha.
Fluxos Alternativos A1	<ol style="list-style-type: none"> 1. Garçom cancela o Pedido. 2. O Sistema fecha a janela de pedido.

Tabela 6 - Especificação de Caso de Uso Fazer Pedido

4.2.8 Consultar Conta



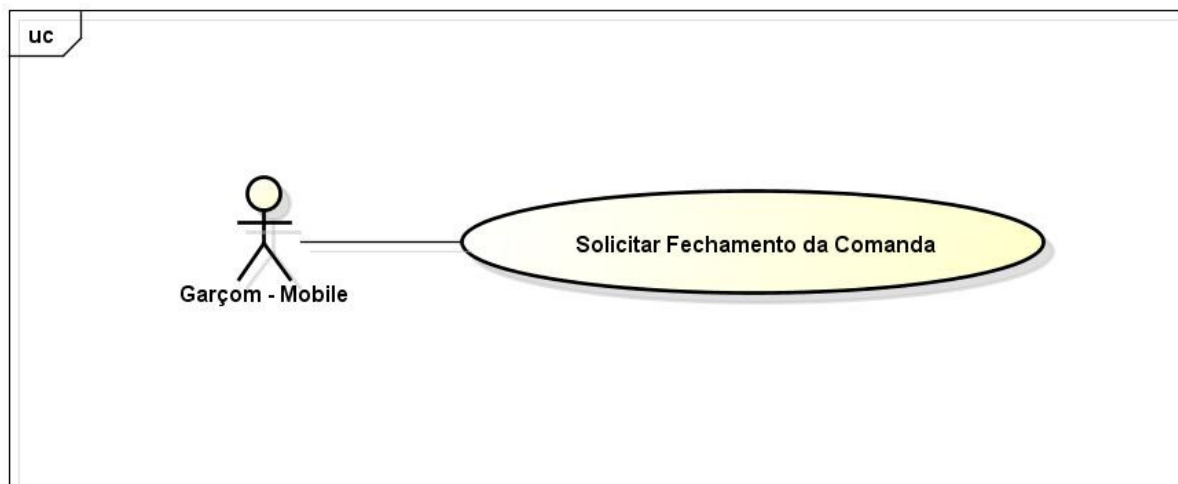
powered by Astah

Figura 11 - Diagrama de Caso de Uso Consultar Conta

Nome do Caso de Uso	Consultar Conta
Ator(es)	Garçom
Resumo	Consultar o consumo da mesa
Pré Condições	Usuário deve estar conectado no aplicativo
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no aplicativo
Fluxo Principal	<ol style="list-style-type: none"> 1. Garçom seleciona uma mesa. 2. O aplicativo abre as categorias dos produtos [A1]. 3. Garçom seleciona o ícone da conta da mesa [A1]. 4. O aplicativo abre o resumo da conta [A1].
Fluxos Alternativos A1	<ol style="list-style-type: none"> 1. Garçom fecha a janela. 2. O aplicativo abre a janela aberta antes de acessar a janela de conta.

Tabela 7 - Especificação de Caso de Uso Consultar Conta

4.2.9 Solicitar Fechamento da Comanda



powered by Astah

Figura 12 - Diagrama de Caso de Uso Solicitar Fechamento da Comanda

Nome do Caso de Uso	Fazer solicitar fechamento da comanda
Ator(es)	Garçom
Resumo	Solicitar fechamento da comanda
Pré Condições	Usuário deve estar conectado no aplicativo
Pós-Condições	O usuário terá acesso a todos os dados e movimentações no aplicativo
Fluxo Principal	<ol style="list-style-type: none"> 1. Garçom seleciona uma mesa. 2. O aplicativo abre as categorias dos produtos [A1]. 3. Garçom seleciona o ícone da conta da mesa [A1]. 4. O aplicativo abre o resumo da conta [A1]. 5. Garçom clica em fechar conta. 6. O aplicativo emite uma mensagem de confirmação. 6. O ícone da mesa selecionada é alterada para cor azul indicando fechamento de conta.
Fluxos Alternativos A1	<ol style="list-style-type: none"> 1. Garçom fecha a janela. 2. O aplicativo abre a janela aberta antes de acessar a janela de conta.

Tabela 8 - Especificação de Caso de Uso Solicitar Fechamento da Comanda

4.3 DIAGRAMA ENTIDADE E RELACIONAMENTO

O Modelo Entidade Relacionamento (também chamado Modelo ER, ou simplesmente MER), como o nome sugere, é um modelo conceitual utilizado na Engenharia de Software para descrever os objetos (entidades) envolvidos em um domínio de negócios, com suas características (atributos) e como elas se relacionam entre si (relacionamentos) (DEVMEDIA).

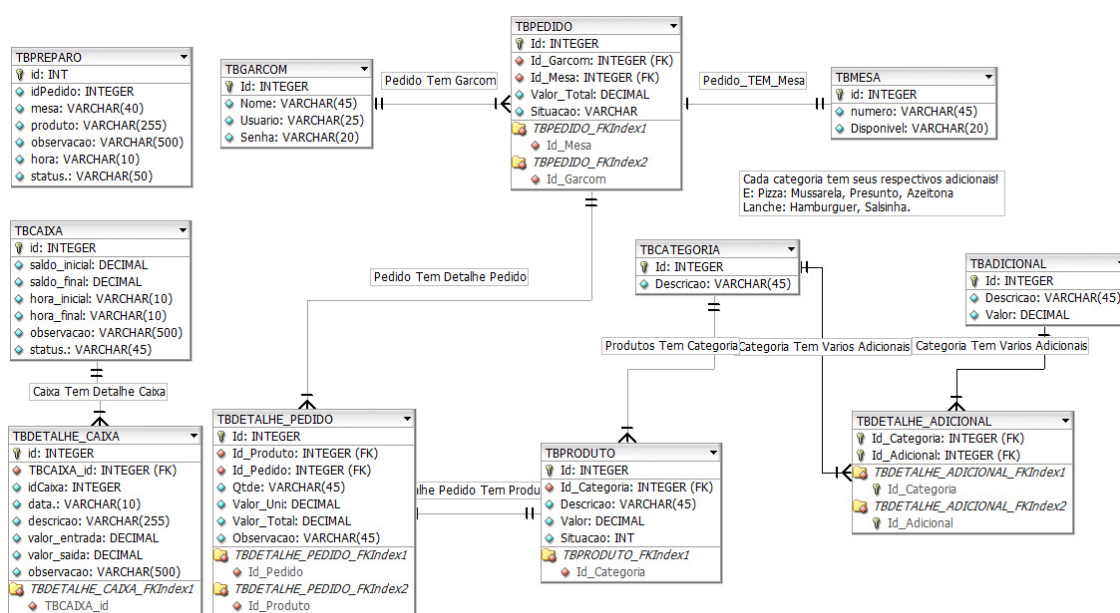


Figura 13 - Diagrama Entidade e Relacionamento

4.4 DIAGRAMA DE ATIVIDADES

O Diagrama de Atividade se preocupa em descrever os passos a serem percorridos para a conclusão de uma atividade específica, muitas vezes representada por um método com um certo grau de complexidade, podendo, no entanto, modelar um processo completo (Guedes, 2009).

A figura 14 representa o Diagrama de Atividade sobre a Realização de Pedido.

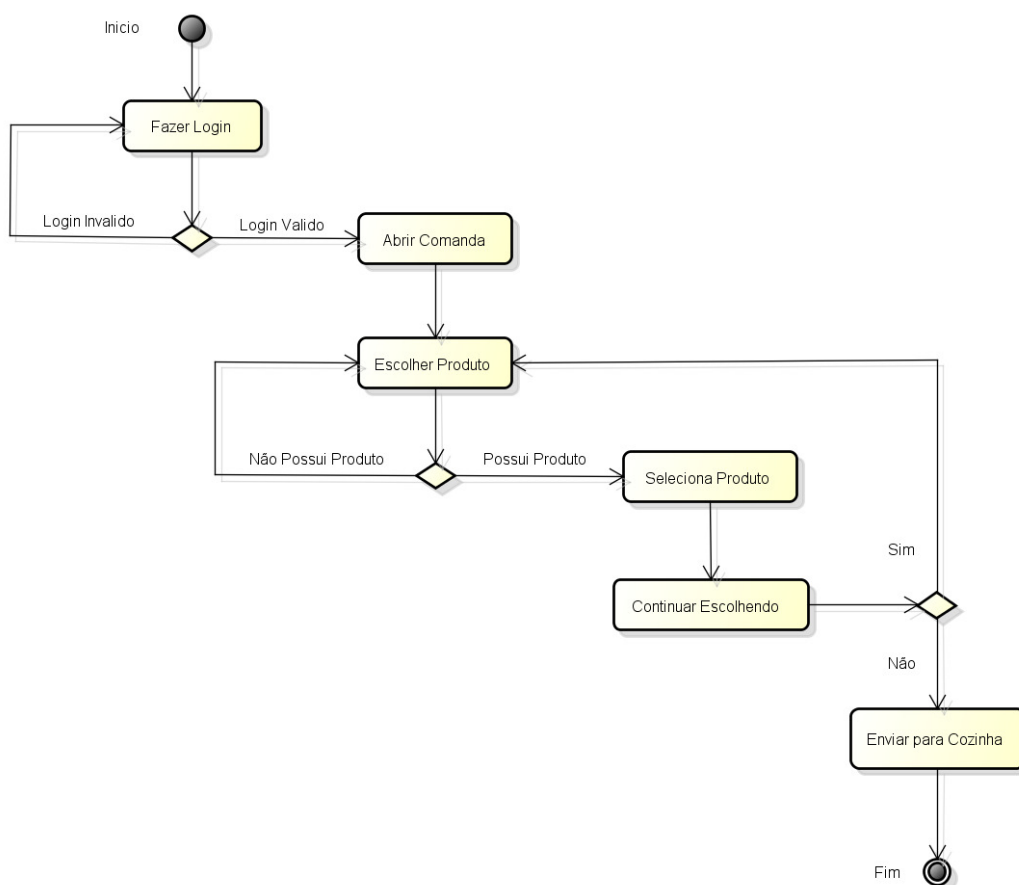
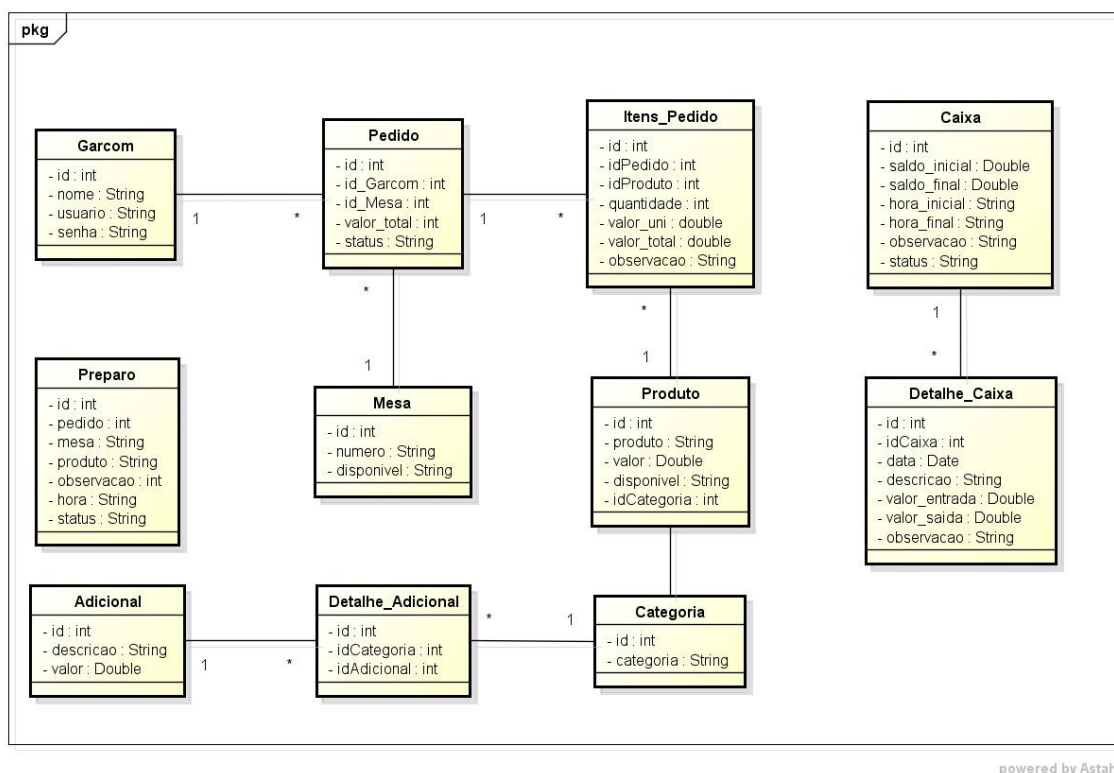


Figura 14 - Diagrama de Atividade

4.5 DIAGRAMA DE CLASSE

Este é o diagrama mais utilizado e o mais importante da UML, servindo de apoio para a maioria dos outros diagramas. Como o próprio nome diz, esse diagrama define a estrutura das classes utilizadas pelo sistema, determinando os atributos e métodos possuídos por cada classe, além de estabelecer como as classes se relacionam e trocam informações entre si (Guedes, 2009).

A figura 15 demonstra o diagrama de Classe do Sistema proposto.



powered by Astah

Figura 15 - Diagrama de Classe

5. RESULTADOS

5.1 APLICATIVO MOBILE

Abaixo estão algumas telas do aplicativo *mobile*.

5.1.1 Tela de Login

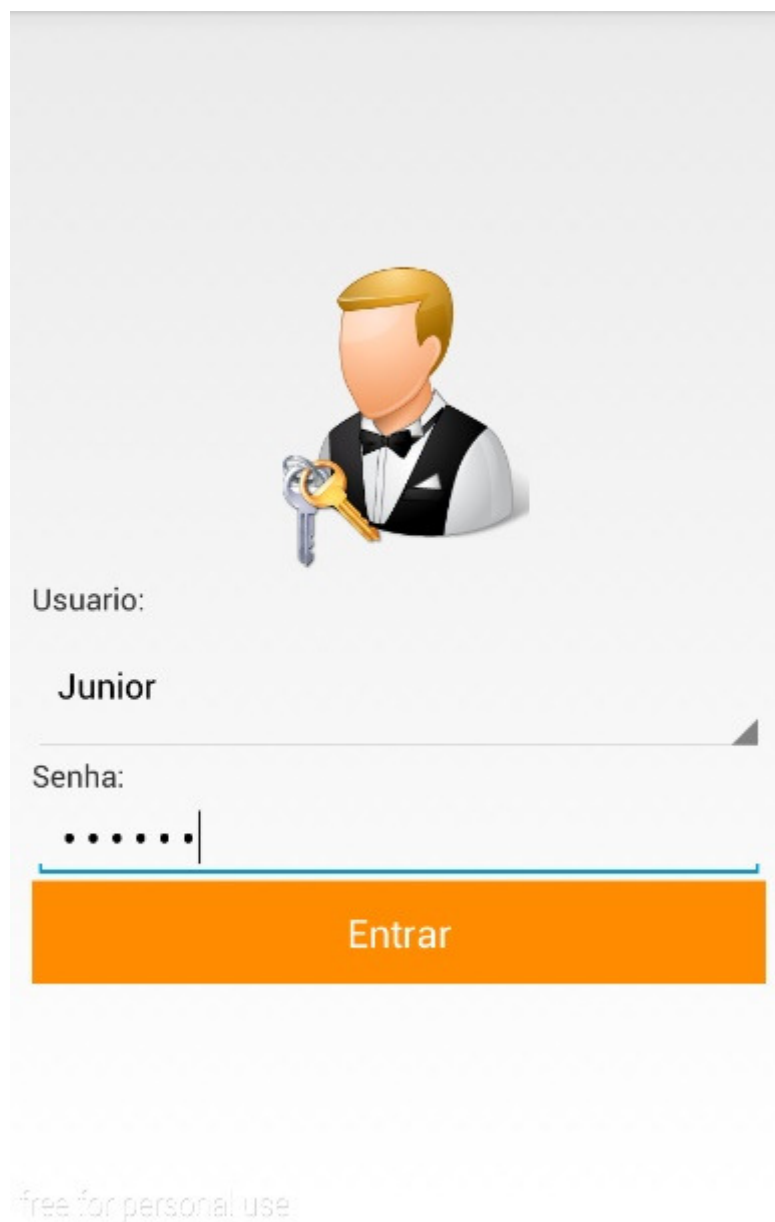


Figura 16 - Tela de Login

5.1.2 Tela das Mesas

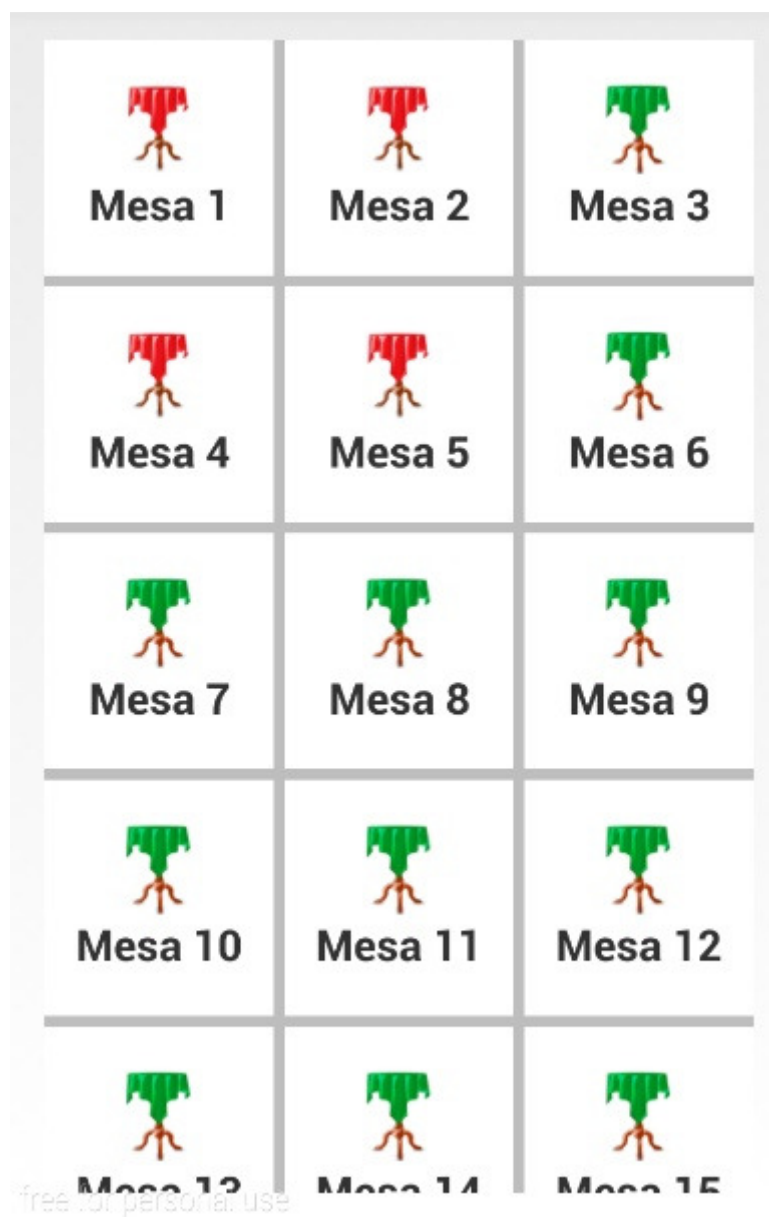


Figura 17 - Tela das Mesas

5.1.3 Tela de Categoria

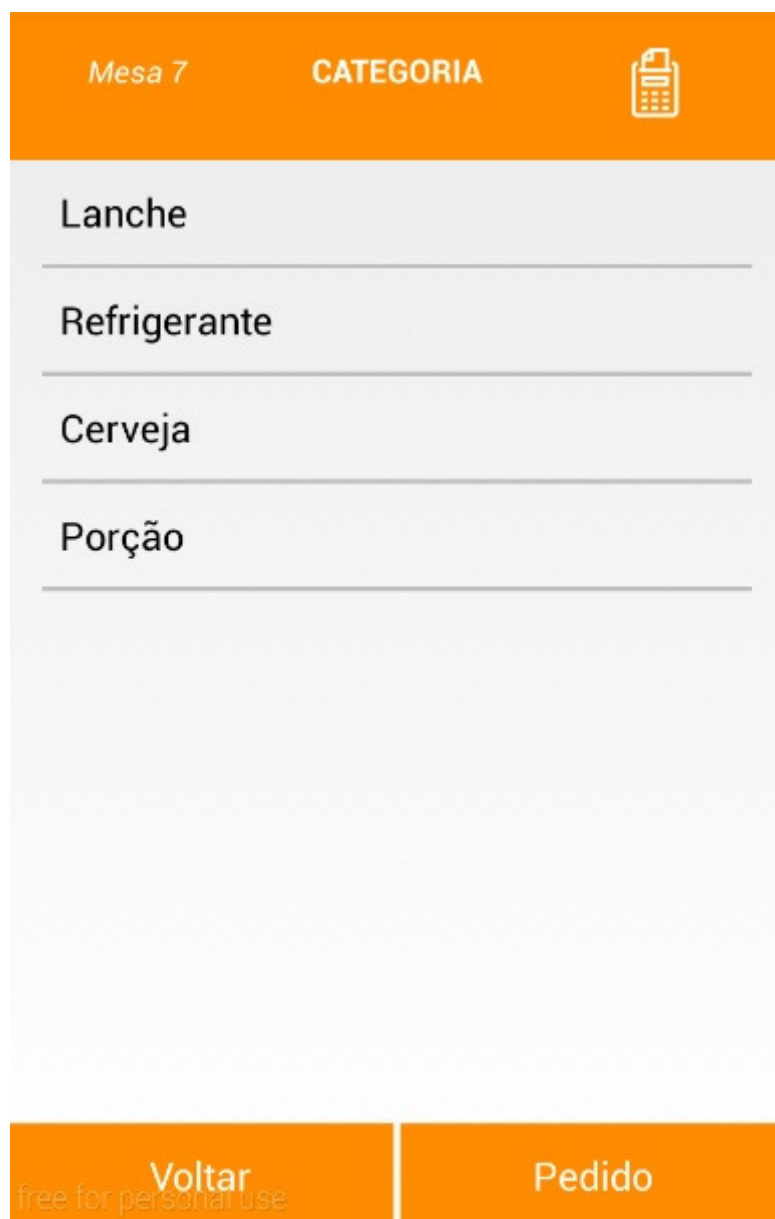


Figura 18 - Tela de Categoria

5.1.4 Tela do Pedido



Figura 19 - Tela do Pedido

5.2 SISTEMA DESKTOP

Abaixo estão algumas telas do sistema *desktop*.

5.2.1 Tela de Login

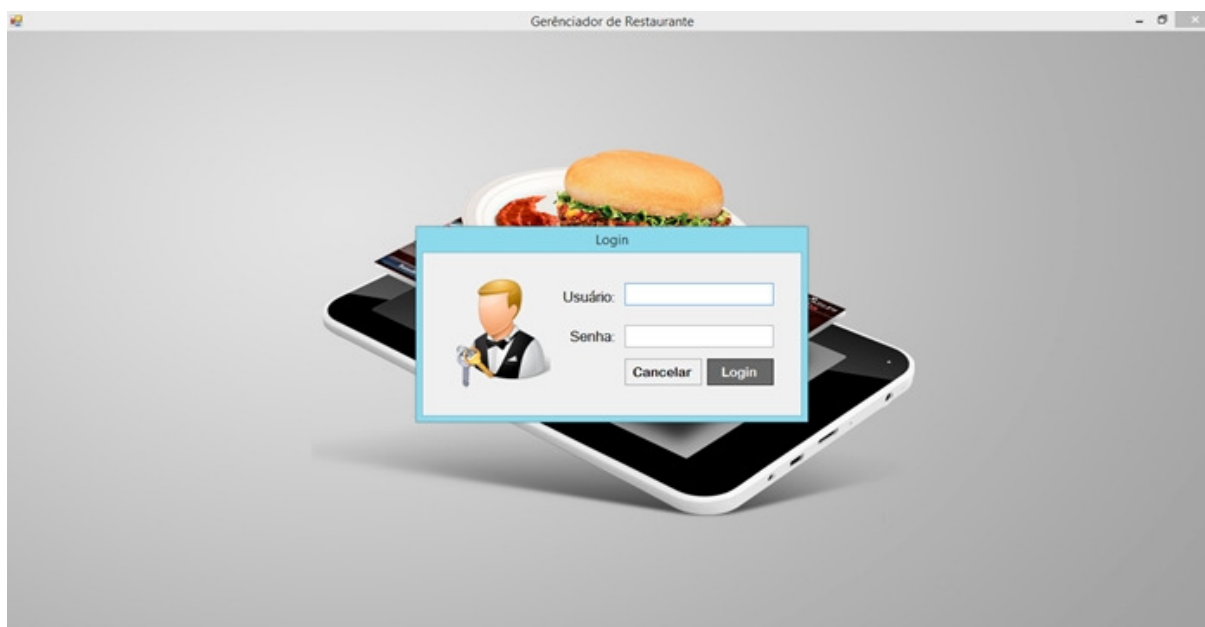


Figura 20 - Tela de login

5.2.2 Tela Principal

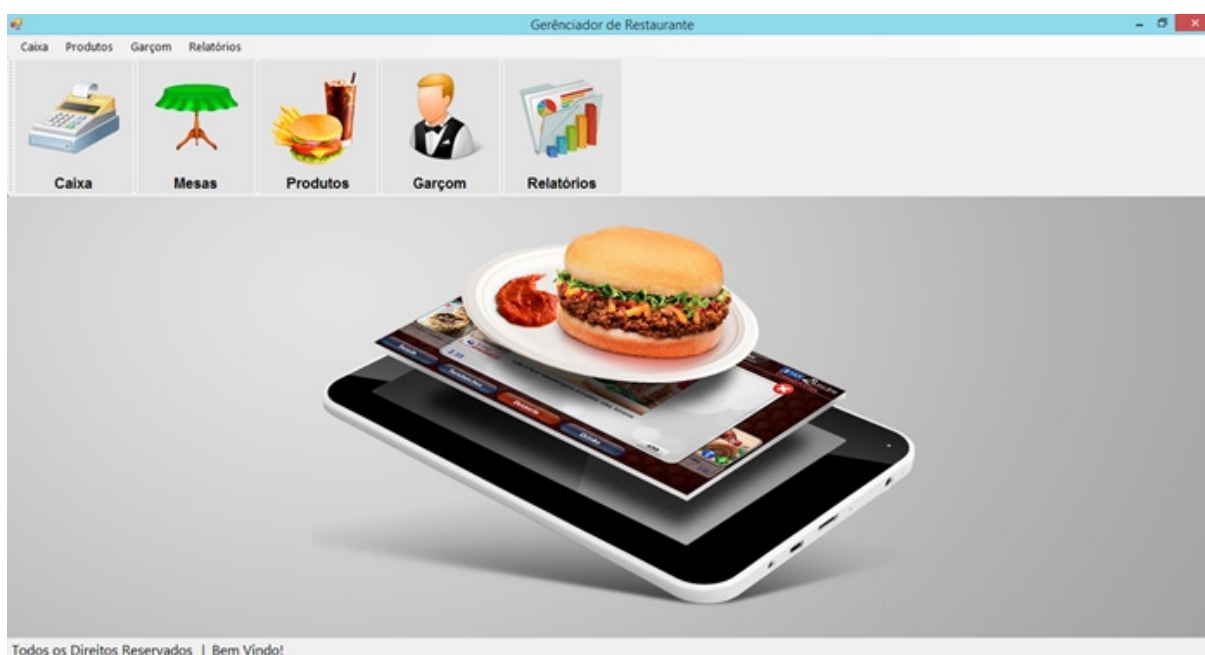


Figura 21 - Tela principal

5.2.3 Tela do Caixa

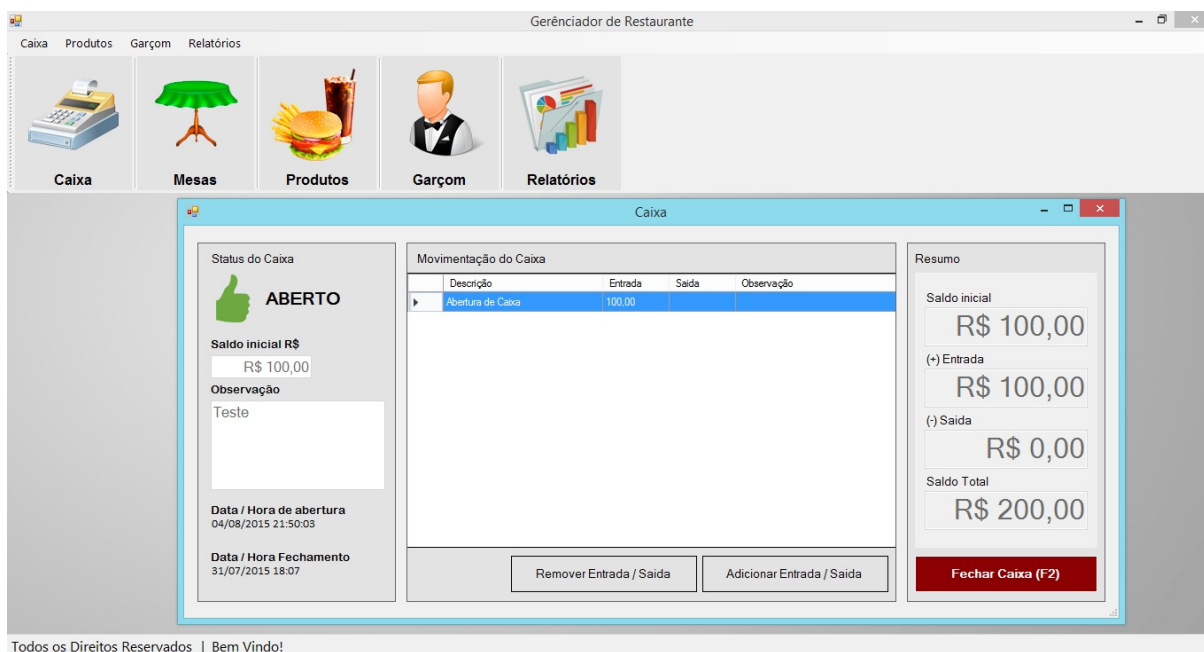


Figura 22 - Tela do Caixa

5.2.4 Tela das Mesas

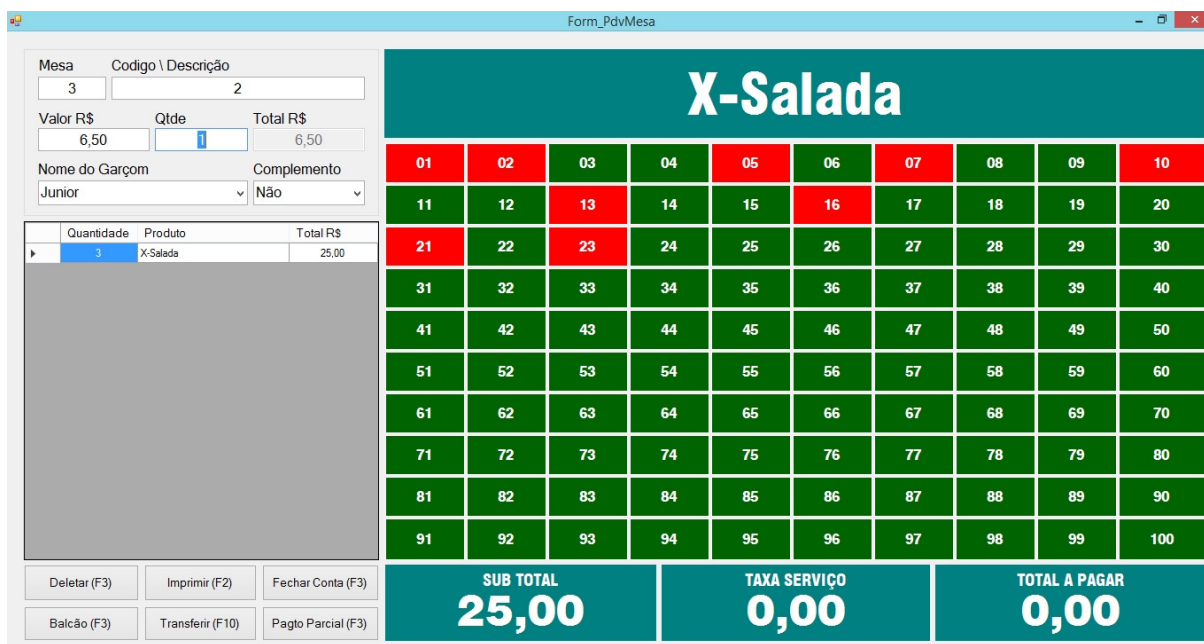


Figura 23 - Tela das Mesas

6. CONCLUSÃO

Este software atenderá de forma eficaz as necessidades do Restaurante, trazendo agilidade e organização para empresa. Além disso, trará facilidade ao acesso das informações, para melhor controle e auxílio no gerenciamento.

O uso de UML e diagramas feitos durante a análise e levantamento de requisitos foram fundamentais para compreender e identificar as principais funcionalidades do sistema.

Durante a fase de implementação os frameworks utilizados foram essenciais e facilitaram muito o desenvolvimento, a complexidade de acessar os dados no banco SQL Server fora simplificados com o uso do Entity Framework.

A maior dificuldade na execução desse projeto, foi o desenvolvimento do aplicativo android e a integração do mesmo com o banco de dados SQL Server, por ser a primeira experiência com a plataforma android e com web services.

Como resultado, pode-se concluir que o objetivo traçado neste trabalho foi alcançado, pois 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.

REFERÊNCIAS

- ADORNO, Daniel Gomes. Web Services SOAP em Java: 2ª edição: Novatec, 2014.
- CONVERSE, Tim; PARK, Joyce. PHP 4 A Bíblia. Campus, 2001.
- DAYCHOUW, Merhi. 40 Ferramentas e técnicas de Gerenciamento. Brasport, 2007.
- GUEDES, Gilleanes T. A. UML 2 – Guia Prático. Novatec, 2009.
- LIMA, Adilson da Silva. UML 2.3 Do Requisito á Solução: Erica, 2012.
- MANSANO, José Augusto N. G. SQL Server 2012 Express Guia Pratico e Interativo: Erica, 2012.
- MARTINS, José Carlos Cordeiro. Gerenciando projetos de desenvolvimento de software com PMI, RUP, e UML. Ed. 5: Brasport, 2010.
- MELO, Ana Cristina. Desenvolvendo aplicações com UML 2.2 Do conceito á implementação: Brasport, 2010.
- RODRIGUES, Joel. Modelo Entidade Relacionamento (MER) e Diagrama Entidade-Relacionamento (DER). Disponível em: <<http://www.devmedia.com.br/modelo-entidade-relacionamento-mer-e-diagrama-entidade-relacionamento-der/14332>>. Acesso em: 02 de Fev. 2015.
- STELLMAN, Andrew. Use a Cabeça C#. Ed. 2: Alta Books, 2011.
- TECHTUDO, Visual Studio: crie apps em C# e .NET e exporte para múltiplas plataformas. Disponível em: <<http://www.techtudo.com.br/tudo-sobre/visual-studio.html>>. Acesso em: 21 de Jan. 2015.
- VARGAS, Ricardo Viana. Manual Prático do Plano de Projeto utilizando o PMBOK Guide. Ed. 3. Brasport, 2009.

ANEXO I

Neste anexo são apresentados os códigos utilizados para que as informações dos garçons sejam acessadas pelo aplicativo Android. O primeiro código apresenta o método “buscarGarcom”, feito em Java, que realiza a solicitação ao webservice e armazena o resultado em uma lista. O segundo código, feito em PHP, faz parte do webservice que recebe as solicitações e retorna um JSON com as informações dos garçons. Por fim, é apresentado o JSON com a lista retornada pelo webservice.

METODO PARA BUSCAR GARÇOM (ANDROID)

```
public static ArrayList<Garcom> burcarGarcom () {  
    String retorno = "";  
    ArrayList<Garcom> listaGarcom = new ArrayList<Garcom>();  
    try{  
        HttpClient httpClient = new DefaultHttpClient();  
        HttpPost httpPost = new HttpPost(Conexao.getConexao());  
        ArrayList<NameValuePair> parametros = new ArrayList<NameValuePair>();  
        parametros.add(new BasicNameValuePair("metodo", "buscarGarcom"));  
        httpPost.setEntity(new UrlEncodedFormEntity(parametros));  
        HttpResponse resposta = httpClient.execute(httpPost);  
        HttpEntity entity = resposta.getEntity();  
        retorno = EntityUtils.toString(entity);  
    }  
    try {  
        JSONArray arrayJson = new JSONArray(retorno);  
        JSONObject json;  
        //Converte a string json em objeto  
        for(int i = 0; i < arrayJson.length(); i++){
```

```

        json = new JSONObject();
        json = arrayJson.getJSONObject(i);
        Garcom garcom = new Garcom();
        garcom.setId(json.getInt("id"));
        garcom.setNome(json.getString("nome"));
        garcom.setUsuario(json.getString("usuario"));
        garcom.setSenha(json.getString("senha"));
        listaGarcom.add(garcom);
    }
} catch (JSONException e) {
    Log.e("log_tag", "Error parsing data "+e.toString());
}
}
}
catch(NullPointerException e){ e.printStackTrace(); }
catch(ClientProtocolException e){ e.printStackTrace(); }
catch(IOException e){ e.printStackTrace(); }
return listaGarcom;
}

```

METODO QUE RETORNA OS GARÇONS CADASTRADOS NO BANCO (PHP)

```

if($metodo === "buscarGarcom"){
    $sql = "SELECT * from Funcionario where perfil = 'Atendente'";
    $stmt = sqlsrv_query($conn, $sql);
    $obj = array();
    $row = array();

    while ($obj = sqlsrv_fetch_array($stmt, SQLSRV_FETCH_ASSOC)){

```

```
$row[] = $obj;
}
sqlsrv_free_stmt( $stmt);
$json = json_encode($row);
echo $json;
}
```

RESPOSTA JSON

```
[{"id":1,"nome":"Administrador","email":null,"telefone":null,"perfil":"Atendente","usuario":"admin","senha":"123"},{"id":2,"nome":"Junior Aparecido Rodrigues da Cruz","email":"juniorccna@hotmail.com","telefone":"33711922","perfil":"Atendente","usuario":"junior","senha":"123"}]
```