

OrderofService 2.0

Autor: Clayton Franklin Queiroz

Orientador: Luciana Rocha Cardoso

Curso: Análise e Desenvolvimento de Sistemas Período: 6º

Área de Pesquisa: Ciências Exatas e da Terra

RESUMO: O presente artigo tem como finalidade demonstrar a importância de se ter um controle a respeito da entrada e saída de equipamentos de uma loja de informática. A proposta é desenvolver um sistema para cadastro de clientes e controle de entrada e saída de equipamentos que são deixados para reparo na assistência técnica. A ideia central é facilitar não só o controle de aparelhos que entram e saem da assistência, mas também o gerenciamento de dados de clientes que ficarão armazenados para que no futuro utilize esses dados novamente. O sistema foi implementado no ambiente de programação Visual Studio 2017 e como SGBD optou-se em utilizar o MySQL. Entre as várias linguagens de programação que o Visual Studio disponibiliza optou-se pelo Visual Basic devido às vantagens que a mesma disponibiliza, tais como: Fácil compreensão e fácil utilização; o que gera um grande ponto positivo para a criação do *software*, além disso, o Visual Basic permite desenvolver rápidas aplicações, fornece vários recursos para os programadores, e existem diversas informações sobre a linguagem na internet, o que auxilia em caso de dúvidas.

PALAVRAS – CHAVE: Ordem de Serviço. Gerenciamento. Visual Studio. MySQL.

1. INTRODUÇÃO

O presente artigo, sob o tema Ordem de serviço, tem por objetivo apresentar as funcionalidades e as vantagens do *software* “OrderofService 2.0”, que faz o controle de entrada e saída de equipamentos de informática que precisam de reparo tais como *notebooks*, *desktops*, *mouses*, teclados, caixinhas de som, estabilizador e entre outros equipamentos. Sendo assim, levanta-se como problema principal a falta de organização e gerenciamento de clientes e equipamentos. Observa-se que no presente momento o gerenciamento de dados de clientes e equipamentos são feitas a partir de ordens de serviços impressas em uma gráfica e todos os dados são passados a mão para as fichas de cadastro de ordens de serviço e também que todas as vezes que um cliente leva seu aparelho para reparar sendo a primeira vez ou não, são pedidos seus dados pessoais para abrir a nova ordem de serviço, o que ocasiona a demora na hora do atendimento ao cliente.

Dado este fato: Qual *software* poderia automatizar o processo de ordem de serviço? Neste contexto surgiu o “OrderofService 2.0” que tem como ideia central facilitar no atendimento ao cliente, gerenciando dados de clientes e equipamentos que são deixados em determinada assistência para reparo. Sendo assim o usuário do sistema cadastra todos os dados de um cliente, esses dados ficarão salvos no banco de dados para serem utilizados futuramente, após o cadastro do cliente caso for a primeira vez que ele deixa o aparelho, o usuário já poderá cadastrar o equipamento que será deixado para reparo e emitindo uma guia para o cliente. Assim facilitando o controle de equipamentos que entram e saírem da assistência e facilitando também no atendimento ao cliente.

A esse respeito, tem-se como metodologia a confecção de pesquisa de campo na empresa de assistência técnica Lopes Informática.

A partir de então, encontra-se substrato a confirmação de hipótese de automação desse trabalho é necessária, para haver um maior gerenciamento e controle de clientes e de equipamentos deixados para reparo na empresa.

2. REFERENCIAL TEÓRICO

Neste capítulo será abordada toda a pesquisa bibliográfica utilizada no trabalho. Os temas que são apresentados englobam os assuntos relacionados ao desenvolvimento do *software* utilizando *Microsoft Visual Studio 2017*, na linguagem *Visual Basic*, e para o armazenamento de dados, o SGBD MySQL. Também são apresentadas as ferramentas e tecnologias utilizadas durante o desenvolvimento.

2.1. ANÁLISE DE SOFTWARE

Na opinião de Pressman (2006, p.29), “*software* é uma ferramenta importante porque afeta quase todos os aspectos de nossas vidas e tornou-se pervasivo no comércio, na cultura e em nossas atividades cotidianas”. É importante destacar que ele torna o serviço automático, trazendo muito mais confiabilidade e agilidade nos processos.

Ainda Pressman (2006 p.30),

O termo *software* engloba muito mais do que um simples programa de computador, dependendo do seu tamanho e qual função irá exercer, um *software* pode ser até uma série de programas separados. Tendo em vista um *software* não é apenas um programa mas também toda a documentação associada que explica como utilizar o sistema e os dados de configuração necessários para fazer com que esses programas operem corretamente. Porém para que se consiga um bom resultado ao se desenvolver um *software* é necessário que haja disciplina tanto no projeto quanto na construção do protótipo e do sistema final.

Para se desenvolver um *software* é importante lembrar-se das suas fases de desenvolvimento escolhendo algum modelo de desenvolvimento que auxilie o desenvolvedor na tentativa de colocar ordem uma atividade inerentemente caótica.

Sommerville (2003, p.47) afirma que,

“Por esta razão ao se desenvolver um sistema de computador deve-se antes de tudo pensar nas especificações, pois possuem a finalidade de descrever as funcionalidades e restrições do *software*, no desenvolvimento, já que o *software* deve ser produzido de modo que atenda às suas especificações, na validação, pois o *software* tem de ser validado para garantir que ele faça o que o cliente deseja, e na possível evolução do mesmo já que ele deve evoluir para atender às necessidades de mudança do cliente”.

Dado a um bom processo de desenvolvimento Pressman (2006) afirma que

“um sistema de computador é bem sucedido quando ele apresenta um bom desempenho rodando sem erros, quando seu manuseio é simples, e além de tudo, quando ele atende a todas as necessidades e especificações do cliente na fase de análise. No entanto para que se chegue em um bom resultado é necessário que haja disciplina tanto no projeto quanto na construção do protótipo e no sistema final”.

Como ambiente de desenvolvimento do *software* optou-se pelo *Microsoft Visual Studio 2017* utilizando a linguagem de programação Visual Basic por se tratar de uma linguagem de fácil utilização e de fácil compreensão.

Como SGBD optou-se em utilizar o MySQL que é um servidor de banco de dados SQL gratuito e de código fonte aberto que tem como características principais: Rapidez, confiança, estabilidade, multitarefa e multiusuários.

Para a construção do projeto optou-se em utilizar o ciclo de vida da engenharia denominado prototipagem que aborda basicamente uma visão evolutiva do desenvolvimento do *software*, este tipo de ciclo conta com a participação dos usuários em cada fase do projeto, assim os usuários podem avaliar o protótipo e ter uma boa noção de como ficará o projeto final, evitando as chances de erro e principalmente de insatisfação após o projeto pronto. A principal vantagem em utilizar a prototipagem é que impede que após todo o projeto criado, o cliente fique insatisfeito, pois o mesmo participou “teoricamente” da criação do projeto. Como desvantagem relacionamos que o cliente muitas vezes vê o protótipo funcionando e acha que está pronto ou que em pouco tempo estará pronto a versão final e não leva em consideração que o protótipo é apenas uma versão que consegue funcionar precariamente e que tem por finalidade a avaliação e ajustes para satisfazer as necessidades do cliente.

2.2. PROTOTIPAÇÃO

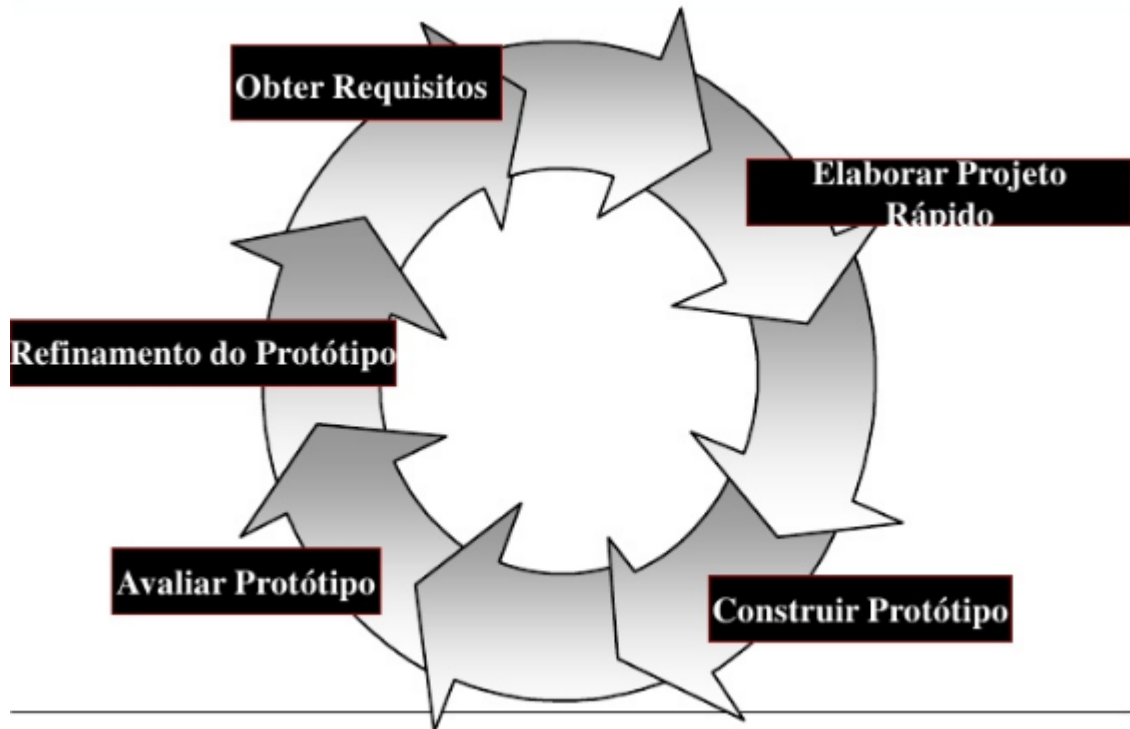
Sommerville (2003, p.72)

“Para reduzir as incertezas sobre o que o sistema irá fazer é necessário que se faça uma análise cuidadosa do mesmo bem como previsões sistemáticas de requisitos e isso só será possível se um protótipo do sistema estiver disponível”.

A prototipação ajuda na exibição do *software* antes que o mesmo fique pronto para que o cliente possa ter uma ideia do projeto final, de acordo com Dennis *et al* (2005) recebe o nome de protótipo a versão inicial de um sistema de *software*,

“ela é utilizada para mostrar os conceitos, e experimentar as opções de projeto e, em geral, para conhecer mais sobre os problemas e suas possíveis soluções, esta etapa tem enorme importância, pois é a hora em que vamos tirar as ideias recolhidas do papel e passar a entendê-las na forma física. Para que os usuários possam fazer experiências há a necessidade do desenvolvimento rápido de um protótipo, isso ajuda também a controlar os custos de desenvolvimento além de apoiar atividades importantíssimas para o desenvolvimento tais como, levantamento e validação de requisitos”.

Figura 1: Modelo de Prototipação



Fonte: SlideShare

De acordo com SILVA e PÁDUA (2010) o modelo de prototipação pode ser dividido nos seguintes itens:

- **Obter Requisitos**: Cliente e desenvolvedor definem os objetivos gerais do *software*, identificam quais requisitos o *software* terá e o que ele irá fazer.
- **Elaborar projeto rápido**: Representação dos aspectos do *software* que são visíveis ao usuário.
- **Construir protótipo**: Implementação rápida do projeto.
- **Avaliar protótipo**: Cliente e desenvolvedor avaliam o protótipo.
- **Refinamento do protótipo**: Cliente e desenvolvedor refinam os requisitos do *software* a ser desenvolvido.

Esta etapa é de fundamental importância, pois permite que os usuários do sistema realizem testes para ver se o sistema está de acordo com suas exigências, na maioria das vezes realizando esses testes faz com que tenham novas ideias para melhoria do *software* desenvolvido, podendo até identificar pontos positivos e negativos. Outra função importante do protótipo é que com ele consegue diminuir os riscos antes da implantação do sistema.

Segundo MATERA (2012, p.15),

“Inserir a fase de prototipação no início do projeto assegura um bom resultado final. É um investimento válido, pois diminui as chances de grandes erros serem identificados tardiamente, gerando um custo alto para o projeto, além de proporcionar ao cliente a possibilidade de

mudar o que não está e acordo com o desejado logo no início do projeto, é eficiente também para a coleta de requisitos de interface, melhorando inclusive a usabilidade do sistema e etc”.

SILVA e PÁDUA (2010, p.23) afirmam que:

É perigoso adotar-se prototipação sem um comprometimento de todas as partes envolvidas, pois esta técnica pode ser visualizada como desperdício de dinheiro, e que após a apresentação de várias versões de protótipos, equipe de desenvolvedores e clientes podem perder o entusiasmo, além de que, os clientes podem confundir o protótipo com o sistema verdadeiro, criando assim falsas expectativas com relação a prazos.

A fase de prototipação é importante no desenvolvimento, por quanto se deve atentar para ela e sobre a sua real funcionalidade, assim como todas as fases ela necessita do envolvimento de toda equipe. Deve-se enfatizar também que o protótipo é apenas uma visão do sistema e que ele não deve ser usado pelo cliente.

2.3. Linguagem de desenvolvimento *Visual Studio 2017*

O Microsoft Visual Studio é um ambiente de desenvolvimento integrado para os sistemas operacionais do Windows. Ele suporta várias linguagens de programação, como Visual C++, Visual C #, Visual J# e Visual Basic .NET, como ambientes de desenvolvimento *web* como ASP.NET. Embora eles já desenvolveram as técnicas necessárias para muitas outras extensões. O Visual Studio permite aos desenvolvedores criar aplicativos, sites e aplicações web e serviços *web* em qualquer ambiente que suportam a plataforma .NET. Assim você pode criar aplicativos que estão interligados entre estações de trabalho, páginas *web* e dispositivos móveis.

A linguagem utilizada para criação do *software* é o Visual Basic, que conta com sua facilidade de ser utilizado e também por ser facilmente compreendida por especialistas, bem como para usuários iniciantes.

O Visual Basic foi escolhido para desenvolvimento do sistema por se tratar de uma linguagem de programação de fácil compreensão e fácil utilização o que gera um grande ponto positivo para a criação do *software*, além disso, o Visual Basic permite desenvolver rápidas aplicações, fornece vários recursos para os programadores, e existem diversas informações sobre a linguagem na internet, o que auxilia em caso de dúvidas. Esses e outros fatores fizeram com que se escolhesse o Visual Basic como linguagem de programação apropriada para elaboração do projeto.

2.4. LEVANTAMENTO DE REQUISITOS

O levantamento de requisitos é uma das partes mais importantes do processo, pois que resultara no desenvolvimento do *software*. Entender o que o cliente deseja é o fator de grande importância, visando a melhor condição para satisfazê-lo e suprir as necessidades e expectativas. O levantamento de requisitos deve conter informações suficientes para permitir que comece a construção do

sistema. Conforme proposto por SOMMERVILLE (2003, p.83) “um requisito é tratado como funcional quando descreve um serviço ou função que o sistema deve realizar”.

Para Dennis & Wixon:

Levantamento de requisitos nada mais é do que as informações do que o sistema deve ser ou quais características ele deve possuir. Nesta etapa o cliente e o desenvolvedor trabalham juntos para recolher as informações de tudo que o sistema deve fornecer, bem como o desempenho exigido do mesmo e as restrições de hardware para que o sistema funcione corretamente conforme previsto.

Segundo SOMMERVILLE (2003) esta etapa tem a finalidade de converter todos os requisitos propostos pelo cliente em uma lista que precisa ser simples e precisa, contendo informações que serão usadas no restante da fase de análise, criando assim os casos de uso que mais tarde se expandem com base nos requisitos.

Os requisitos podem ser classificados de várias formas, uma forma bastante aceitável é que a classificação seja entre requisitos funcionais e não-funcionais. Um requisito funcional aborda o que o sistema deve fazer. Os requisitos não funcionais declaram características de qualidade que o sistema deve possuir e que estão relacionadas às suas funcionalidades.

O levantamento de requisitos para a criação do *software* “OrderofService 2.0” deu-se a partir da pesquisa de campo realizada na empresa Lopes Informática. Durante o levantamento de requisitos pode-se coletar todas as informações necessárias para dar início ao projeto, e pode-se observar que o grande problema da empresa estava relacionado ao gerenciamento de dados de clientes e no controle dos equipamentos que eram deixados na empresa, sendo assim, o *software* tem por finalidade resolver este problema.

2.5. REQUISITOS DO SISTEMA

O *software* elaborado tem restrição de funcionar apenas na plataforma *Windows*. O OrderofService 2.0 necessita de alguns requisitos mínimos para que o usuário o tenha em pleno funcionamento, são eles plataforma *Windows* 7 ou superior com suporte para arquitetura de x32 e x64 bits; Memória RAM de no mínimo 2 GB e Processador de 2.0 GHz.

2.6. BANCO DE DADOS

Sobre a história do MySQL, OFICINADANET (2017) afirma que

“a linguagem foi criada na Suécia por dois suecos e um finlandês: David Axmark, Allan Larsson e Michael “Monty” Widenius, que têm trabalhado juntos desde a década de 1980. Hoje seu desenvolvimento e manutenção empregam aproximadamente 400 profissionais no mundo inteiro, e mais de mil contribuem testando o software, integrando-o a outros produtos, e escrevendo a respeito dele. No dia 16 de Janeiro de 2008, a MySQL AB, desenvolvedora do MySQL foi adquirida pela Sun Microsystems, por US\$ 1 bilhão, um preço jamais visto no setor de licenças livres. No dia 20 de Abril de 2009 a Oracle compra a Sun Microsystems e todos o seu produtos,

incluindo o MySQL. A Comissão Europeia ainda não aprovou a aquisição”.

Para a construção da base de dados do *software* optou-se em utilizar o MySQL, pois é sem dúvida uma das opções mais confiáveis e econômicas que existe, além de ser gratuito o que barateia significativamente o valor final do projeto ele conta com outras vantagens como possuir código fonte aberto, facilidade de programação, pode ser utilizado em qualquer tipo de aplicação desde as mais simples até as mais robustas e entre outras mais vantagens que tornam o MySQL o SGBD para melhor atender na composição do *software*.

De acordo com o DEVMIDIA (2017),

O MySQL é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês *Structured Query Language*) como interface. É atualmente um dos bancos de dados mais populares, com mais de 10 milhões de instalações pelo mundo. Entre os usuários do banco de dados MySQL estão: NASA, Friendster, Banco Bradesco, Dataprev, HP, Nokia, Sony, Lufthansa, U.S Army, US. Federal Reserve Bank, Associated Press, Alcatel, Slashdot, Cisco Systems, Google CanaVialis S.A. e outros.

O MySQL além de gerenciar o banco de dados, foi escolhido pelas suas diversas vantagens. Ele é gratuito o que faz com que barateie significativamente o valor final do projeto; possui código fonte aberto; é de fácil programação e aprendizado; ele pode ser totalmente modificado; também possui funções importantes que auxiliam durante o processo de desenvolvimento; Pode ser utilizado em qualquer tipo de aplicação desde as mais simples até as mais robustas; O mercado disponibiliza diversos profissionais com experiência no uso do banco de dados; É multi-plataforma; E permite que sejam implementados regras de segurança no servidor.

2.7. TESTE DE SOFTWARE

Segundo DENNIS e WIXOM (2005, p.33):

Ao se desenvolver um sistema, uma fase muito importante a ser levada em consideração é a fase de testes, pois é nela que serão descobertos os possíveis erros, inconsistências e falhas no *software*. Um erro frequente ao se desenvolver um sistema é querer testá-lo assim que os primeiros módulos são concluídos.

O Teste de *software* é a etapa de controle de qualidade, esta etapa serve para assegurar que o *software* está funcionando perfeitamente da maneira esperada e tem como principal objetivo revelar falhas do sistema para que possam ser corrigidas até que o produto final atinja a qualidade desejada. Geralmente é a última etapa na construção de um sistema.

A fase de testes do *software* foi dividida em etapas para ter a aprovação de que tudo esteja funcionando perfeitamente. Dentre essas etapas obteve os seguintes testes:

- Testes de integração que garante que um ou mais componentes combinados estejam funcionando.

- Teste operacional que garante que a aplicação possa rodar muito tempo sem falhar.
- Teste de regressão, toda vez que algo for mudado, deve ser testada toda aplicação novamente.
- Teste de caixa-preta que testa todas as entradas e saídas desejadas, neste teste não se preocupa com o código e cada saída indesejada é vista como erro.
- Teste de caixa-branca, tem por objetivo testar o código, pois as vezes existem partes do código que nunca foram testadas.
- Teste de performance que verifica se o tempo de resposta é o desejado para o momento de utilização da aplicação.
- Teste de aceitação do usuário, esta fase o usuário analisa o *software* e vê se está de acordo, nesta etapa se estiver alguma crítica como por exemplo um botão pequeno, são feitos os ajustes até estar de acordo com o desejado.
- Teste de configuração, testa se o *software* funcionará perfeitamente no *hardware* que será instalado.
- Teste de usabilidade, este teste é mais focado na experiência do usuário, se a interface é de fácil compreensão, se o menu não está confuso ou algo do tipo.

3. METODOLOGIA

A presente pesquisa foi realizada na empresa Lopes Informática, uma assistência técnica de computadores, com os funcionários da mesma. Foi feita uma coleta de dados para o levantamento de requisitos. Para o desenvolvimento do projeto foi utilizada caso de uso para melhor compreensão das necessidades do cliente. Ela é qualitativa pois todos os dados foram retirados do dia a dia da empresa. Quanto à natureza a ideia da pesquisa aplicada é: O *software* pode ser útil e pode ajudar dentro de uma empresa?

A linguagem de programação escolhida para desenvolvimento do projeto foi *Visual Basic*. Ela foi escolhida por permite desenvolver de forma rápida e fácil controle visual com barras de rolagem, interruptores e indicadores, além de formulários que um usuário pode preencher. O *Visual Basic* oferece suporte completo para programação orientada a objeto, incluindo encapsulamento, herança e polimorfismo.

O banco de dados escolhido foi o MySQL, o que levou a escolher ele foi primeiramente por se tratar de um *software* livre e com capacidade de até 300 mil registros em seu banco de dados, e também devido a características importantes como excelente desempenho e estabilidade, facilidade no uso, suporta praticamente qualquer plataforma atual, compatível com a maioria dos *softwares* de desenvolvimento e etc.

4. DISCUSSÃO DE RESULTADOS

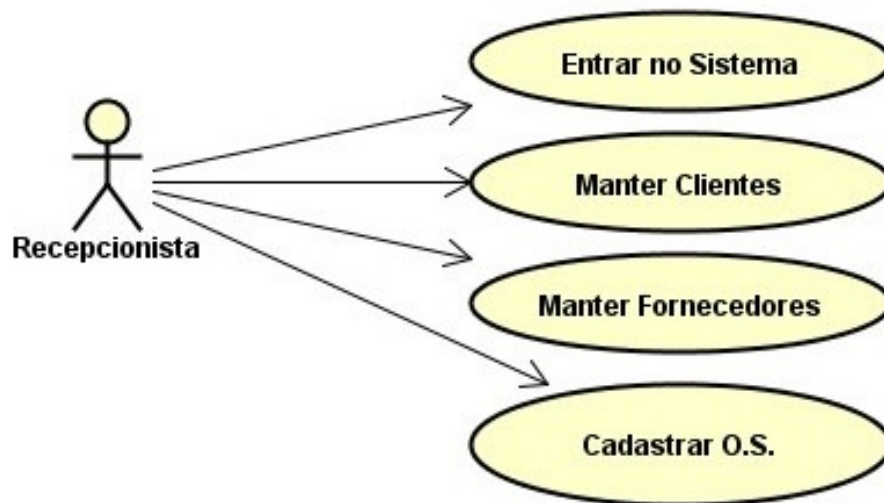
4.1 DIAGRAMAS

O diagrama que se segue, o de Caso de Uso, objetiva descrever as funcionalidades propostas para um novo sistema que será projetado, é uma

excelente ferramenta para o levantamento dos requisitos funcionais do sistema mostrando o ponto de vista do usuário do sistema.

DIAGRAMA 1: Caso de Uso

Figura 2: Diagrama de Caso de Uso



Fonte: Acervo Pessoal

Especificação funcional do Caso de Uso

Nome: Cadastrar O.S.

Objetivo: Cadastrar ordens de serviço para empresas de informática.

Ator: Recepcionista.

Cenário Principal

1. O sistema exibe uma tela contendo a apuração de ordens de serviços cadastradas, contendo os seguintes campos:
 - 1.1. Nome
 - 1.2. Técnico
 - 1.3. Data
 - 1.4. Total
2. Os dados aparecem por ordem de cadastro.

Cenário Alternativo 1

1. O sistema apresenta uma tela de novo cadastro contendo as seguintes informações:
 - 1.1. Caixa de seleção com clientes já cadastrados.
 - 1.2. Digitar busca de equipamento existente ou inclui novo equipamento.
 - 1.3. Digitar busca de peças existentes ou inclui novas peças.

- 1.4. Digitar busca de serviços existentes ou inclui novo serviço.
- 1.5. Caixa de seleção com técnicos já cadastrados.
- 1.6. Digitar alguma observação.
- 1.7. Caixa de seleção de status.
2. A recepcionista informa os dados da nova ordem de serviço.
3. A recepcionista salva a nova ordem de serviço.
4. O sistema exibe mensagem de ordem de serviço salva.

Cenário Alternativo 2

1. O sistema exibe tela de consulta contendo as seguintes informações:
 - 1.1. Digitar código da ordem de serviço
 - 1.2. Digitar nome do cliente
2. O sistema retorna a busca

Cenário Alternativo 3

1. O sistema exibe campo de exclusão de ordem de serviço
2. O sistema exclui a ordem e serviço

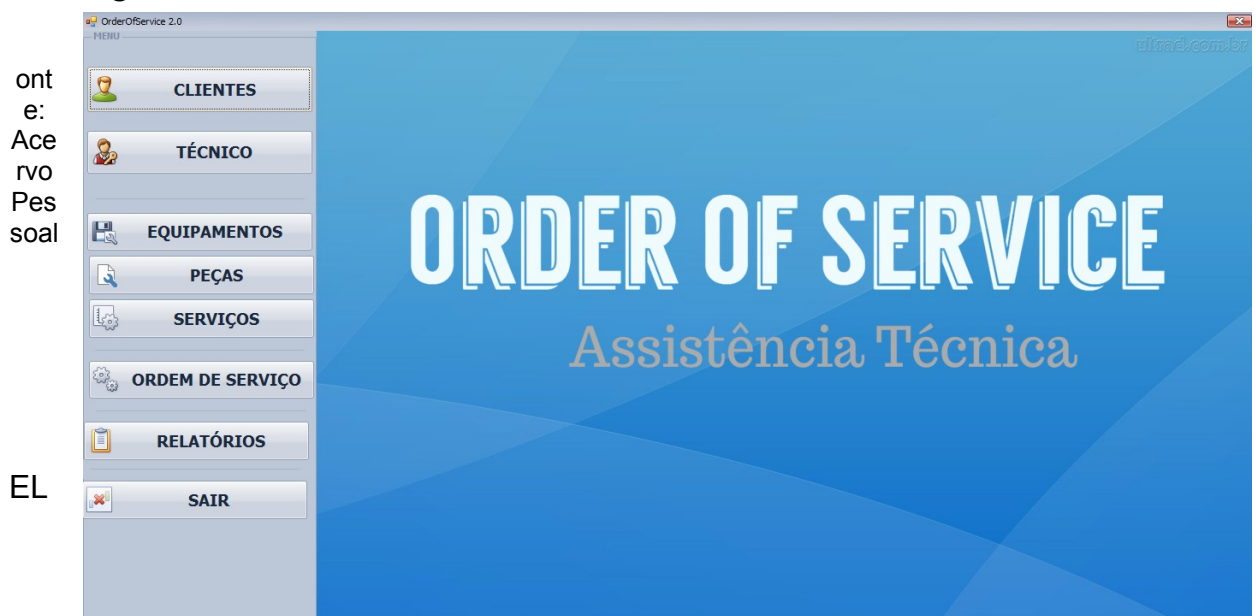
Os demais diagramas se encontram no Apêndice B.

4.2 TELAS

Nesta etapa, serão apresentados as principais telas presentes no sistema inferindo que todas as demais telas estarão presentes no Apêndice A.

TELA 01 – Menu inicial do sistema: Esta tela representa a tela inicial do sistema onde todas as funções do sistema podem ser executadas. Nela encontramos os botões para acessar o cadastro de clientes, cadastro de técnicos, cadastro de equipamentos, cadastro de peças, cadastro de serviços, cadastro da ordem de serviço, encontramos também o botão que dá acesso a parte de relatórios de ordens de serviços e o ultimo botão é o botão sair do sistema. Esses botões e suas respectivas telas se encontram explicadas no Apêndice A.

Figura 3: Menu inicial do sistema OrderOfService



A 02 – Cadastro de ordens de serviço: Nesta tela o usuário do sistema preenche os dados e o sistema realiza o cadastro.

Figura 4: Cadastro de Ordens de Serviço

Fon
te:
Ace
rvo
Pes
soal

O
sist
em
a
pos
sui
o
cad
ast



ORDEN DE SERVIÇO....

ORDEN DE SERVIÇO Nº 1

CLIENTE EQUIPAMENTO PEÇAS SERVIÇOS TÉCNICO OBSERVAÇÃO STATUS

SELECIONA UM CLIENTE

NOME

 CANCELAR  CONFIRMA **TOTAL R\$ 0**

ro dos clientes. A tela se encontra no apêndice A Figura 6: Clientes.

5. CONCLUSÃO

O presente trabalho apresentou a implementação de um *software* para auxiliar na organização das ordens de serviço de uma informática, denominado *OrderofService 2.0*. O sistema foi implementado utilizando o *Microsoft Visual Studio 2017*, na linguagem *Visual Basic*, seguindo os princípios de uma arquitetura orientada a objeto a qual a linguagem se submete. Ele irá facilitar o cadastro de clientes e aparelhos que serão deixados em determinada assistência para reparo, assim o usuário do sistema cadastra os dados de um cliente, esses dados ficarão salvos no banco de dados para serem utilizados futuramente, após o cadastro do cliente caso for a primeira vez que ele deixa o aparelho, o usuário já poderá cadastrar o produto que será deixado para reparo e emitindo uma guia para o cliente. Assim facilitando o controle de aparelhos que entrarem e saírem da assistência.

O desenvolvimento deste projeto para o gerenciamento dos dados de clientes e dos equipamentos deixados na empresa irá propiciar uma melhoria no atendimento do cliente, pois o usuário não vai precisar toda vez ficar coletando os dados pessoais do cliente, esses dados a partir de agora ficarão salvos no banco de dados e toda vez que for necessário a consulta a esses dados é só fornecer o nome do cliente para o programa que todos os dados ficarão disponíveis, assim poupando tempo na hora do atendimento, e quando for necessário a consulta em relação a algum equipamento que foi deixado para reparo, o usuário em pouco tempo conseguirá passar ao cliente todos dados relacionados ao reparo de seu aparelho, mantendo assim uma qualidade de atendimento ao cliente e melhor organização de sua empresa.

6. REFERÊNCIAS

DENNIS, Alan & WIXOM, Barbara Haley. **Análise e Projeto de Sistemas**. Rio de Janeiro: LTC, 2005, 2ª. Ed.

DEVMEDIA. MySQL, 2017. Disponível em:< <http://www.devmedia.com.br/mysql-instalando-e-configurando-revista-sql-magazine-99/24354> >. Acesso em: 28 de Setembro. 2017.

MATERA Systems. Protótipos 2012. Disponível em:< <http://www.matera.com.br/2012/12/27/beneficios-da-utilizacao-de-prototipos-no-desenvolvimento-de-sistemas/> >. Acesso em: 29 de Setembro. 2017.

OFICINA DA NET. Mysql – o que é? 2017. Disponível em:< https://www.oficinadanet.com.br/artigo/2227/mysql_-_o_que_e >. Acesso em: 17 de maio. 2017.

PRESSMAN, Roger S. **Engenharia de Software**. 6.ed. São Paulo: McGraw – Hill, 2006.

SILVA e PÁDUA. Prototipação – Engenharia de Usabilidade [2010]. Disponível em:<<http://homepages.dcc.ufmg.br/~clarindo/arquivos/disciplinas/eu/material/transparencias/topicos/9-prototipacao.pdf> >. Acesso em: 29 de Setembro. 2017.

SOMMERVILLE, I. **Engenharia de Software**. SP: Prentice Hall Brasil, 2003.

VISUAL STUDIO BLOG. Microsoft Visual Studio, 2012. Disponível em:< <https://progvisualestudio.wordpress.com/historia> >. Acesso em: 16 de maio. 2017.

WIKIPEDIA. Visual Basic, 2016. Disponível em:< https://pt.wikipedia.org/wiki/Visual_Basic >. Acesso em: 16 de maio. 2017.

7. APÊNDICES

Apêndice A: Telas

Nesta etapa são apresentados todas as telas que compõe o sistema OrderOfService.

TELA 03 – Login: Nesta tela será informada o nome de usuário e a senha para que se possa ter acesso ao sistema.

Figura 5: Login

The image shows a login interface with a light blue background. On the left, there are two text input fields. The first field is labeled 'Usuario' in a bold, black, sans-serif font. The second field is labeled 'Senha' in a bold, black, sans-serif font. Below these fields are two buttons. The first button is labeled 'SAIR' and features a red circular icon with a white 'X' on the left. The second button is labeled 'ENTRAR' and features a black circular icon with a white checkmark on the left. To the right of the input fields is a square placeholder for a user profile picture, containing a dark blue silhouette of a person's head and shoulders.

Fonte: Acervo Pessoal

TELA 04 – Clientes: Nesta tela consegue-se ter acesso a lista de clientes cadastrados no sistema, consegue-se também cadastrar um novo cliente, editar algum cliente já cadastrado e também excluir algum cliente do banco de dados

Figura 6: Clientes

The screenshot shows a window titled "FERRAMENTA DE BUSCA ...". It features a search bar at the top, a table with two columns labeled "Cod" and "Descrição", and a toolbar at the bottom with buttons for "SAIR", "NOVO", "EDITAR", and "EXCLUIR".

Fonte: Acervo Pessoal

TELA 05 – Cadastrar novo cliente: Nesta tela é possível cadastrar um novo cliente preenchendo todos os dados e em seguida clicando no botão salvar para que todas as informações possam ser salvas no banco de dados.

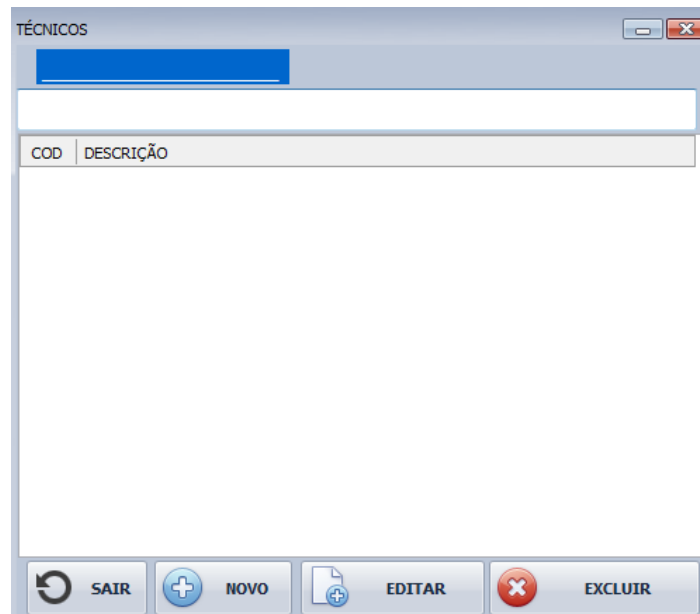
Figura 7: Cadastrar novo cliente

The screenshot shows a window titled "CADASTRO DE PESSOAS". It has a tab labeled "GERAL" and a form with the following fields: "COD", "NOME", "Nº DOCUMENTO", "EMAIL", "TELEFONE" (with area and number fields), "CELULAR" (with area and number fields), "CEP" (with state and number fields), "CIDADE", "UF" (dropdown), "RUA", "Nº", "BAIRRO", and "COMPLEMENTO". At the bottom, there are "SAIR" and "SALVAR" buttons.

Fonte: Acervo Pessoal

TELA 06 – Técnicos: Nesta tela conseguiremos ter acesso a lista de técnicos cadastrados no sistema, conseguiremos também cadastrar um novo técnico, editar algum técnico já cadastrado e também excluir algum técnico do banco de dados.

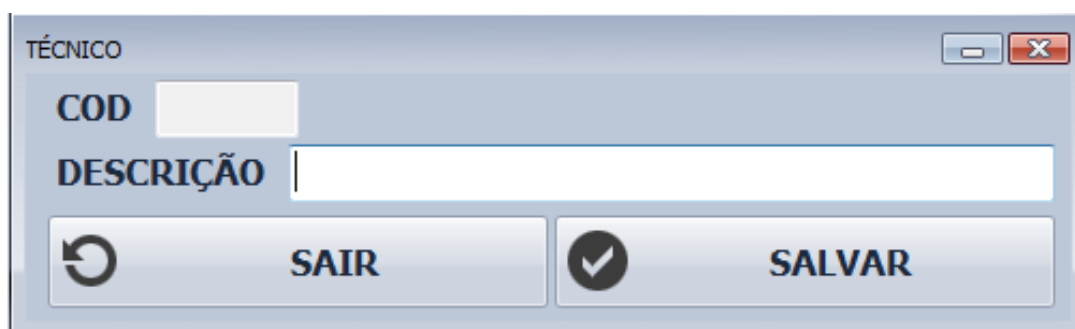
Figura 8: Técnicos



Fonte: Acervo Pessoal

TELA 07 – Cadastrar novo técnico: Nesta tela é possível cadastrar um novo técnico preenchendo o campo descrição com o nome do novo técnico e em seguida clicando no botão salvar para que a informação possa ser salva no banco de dados.

Figura 9: Cadastrar novo técnico



Fonte: Acervo Pessoal

TELA 08 – Equipamentos: Nesta tela conseguiremos ter acesso a lista de equipamentos cadastrados no sistema, conseguiremos também cadastrar um novo equipamento, editar algum equipamento já cadastrado e também excluir algum equipamento do banco de dados.

Figura 10: Equipamentos

A interface 'BUSCA EQUIPAMENTO' apresenta uma barra de busca no topo, seguida por uma tabela com os cabeçalhos 'COD', 'DESCRIÇÃO' e 'NUMERO'. Abaixo da tabela, há uma barra de ferramentas com os botões 'SAIR' (com ícone de seta circular), 'NOVO' (com ícone de plus), 'EDITAR' (com ícone de documento e plus) e 'EXCLUIR' (com ícone de X vermelho).

Fonte: Acervo Pessoal

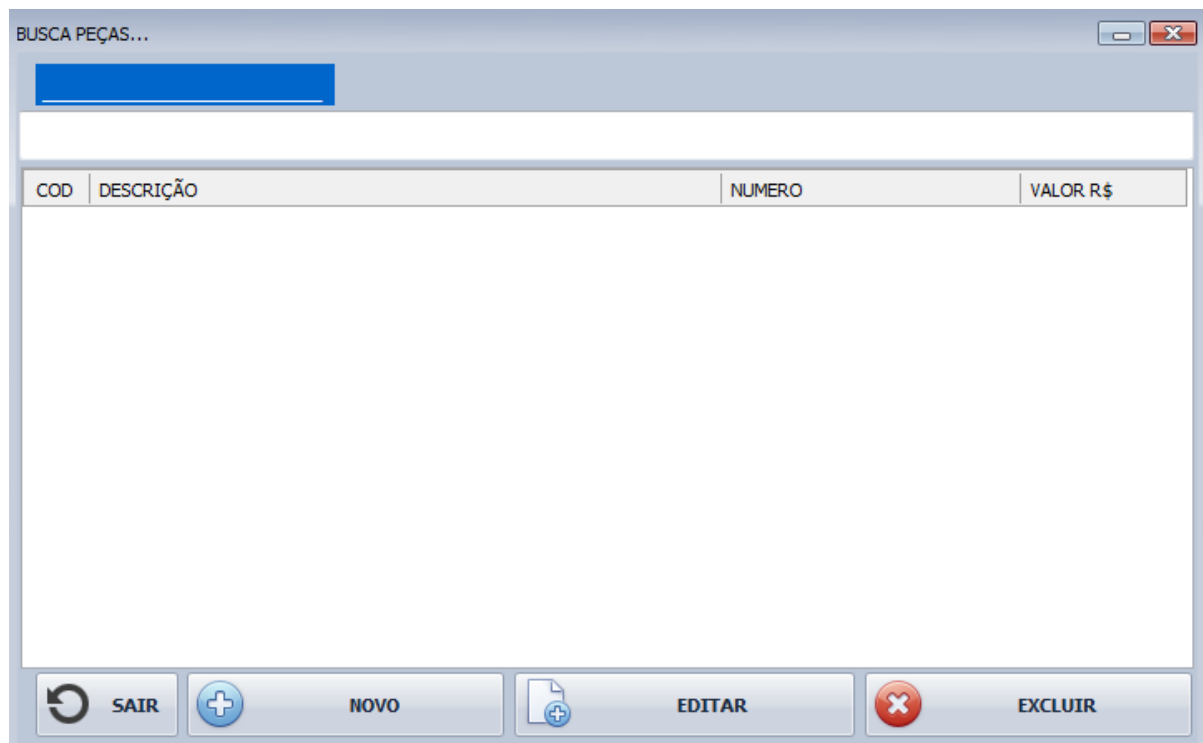
TELA 09 – Cadastrar novo equipamento: Nesta tela é possível cadastrar um novo equipamento preenchendo todos os dados e em seguida clicando no botão salvar para que todas as informações possam ser salvas no banco de dados.

Figura 11: Cadastrar novo equipamento

A interface 'EQUIPAMENTO' para cadastro contém campos de entrada para 'COD', 'DESCRIÇÃO' e 'NUMERO DE SERIE'. Na base da tela, há dois botões: 'SAIR' (com ícone de seta circular) e 'SALVAR' (com ícone de checkmark).

Fonte: Acervo Pessoal

TELA 10 – Peças: Nesta tela conseguiremos ter acesso a lista de peças cadastrados no sistema, conseguiremos também cadastrar uma nova peça, editar alguma peça já cadastrada e também excluir alguma peça do banco de dados.

Figura 12: Peças


BUSCA PEÇAS...

COD	DESCRIÇÃO	NUMERO	VALOR R\$
-----	-----------	--------	-----------

SAIR NOVO EDITAR EXCLUIR

Fonte: Acervo Pessoal

TELA 11 – Cadastrar novas peças: Nesta tela é possível cadastrar uma nova peça preenchendo todos os dados e em seguida clicando no botão salvar para que todas as informações possam ser salvas no banco de dados.

Figura 13: Cadastrar novas peças


PEÇAS.....

COD

DESCRIÇÃO

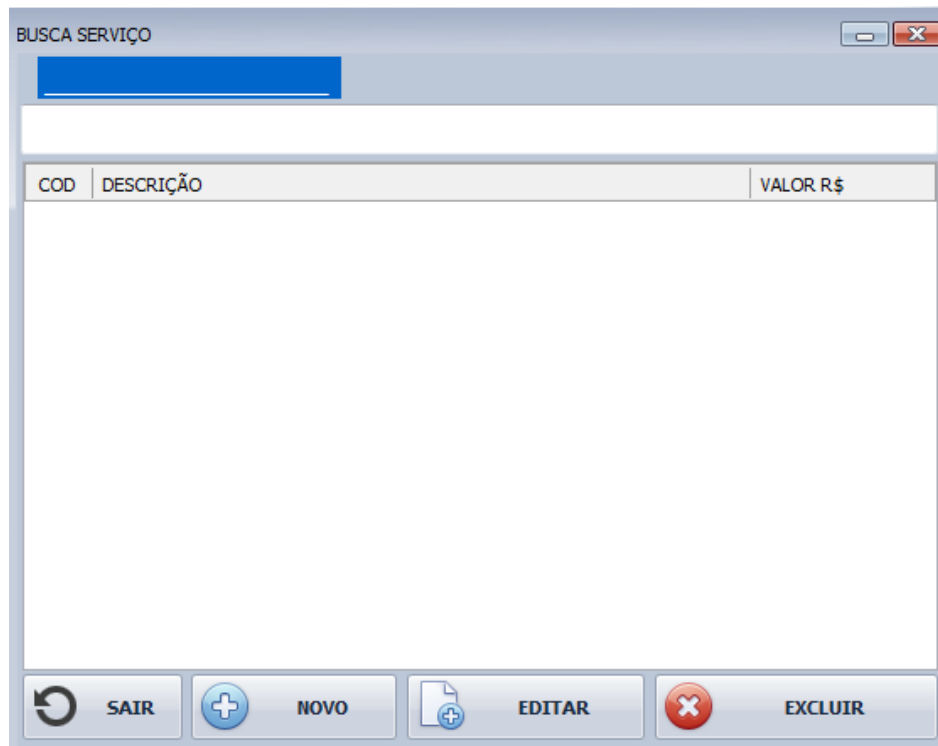
NUMERO DE SERIE

VALOR R\$

SAIR SALVAR

Fonte: Acervo Pessoal

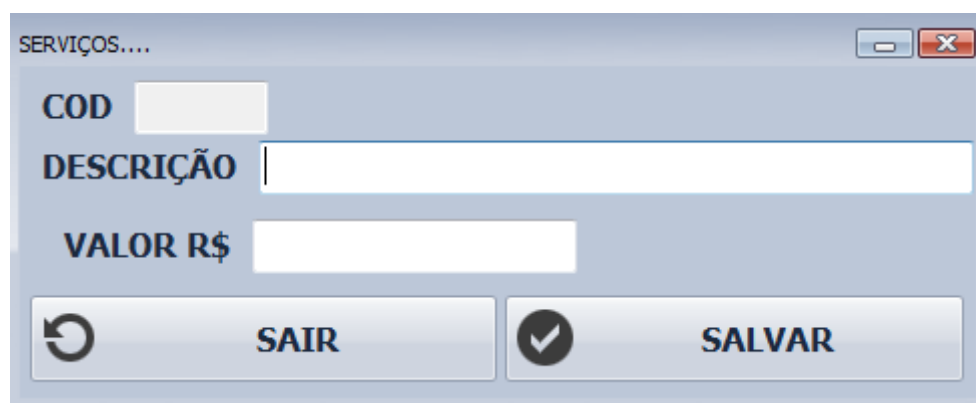
TELA 12 – Serviços: Nesta tela conseguiremos ter acesso a lista de serviços cadastrados no sistema, conseguiremos também cadastrar um novo serviço, editar algum serviço já cadastrado e também excluir algum serviço do banco de dados.

Figura 14: Serviços

A janela 'BUSCA SERVIÇO' apresenta uma interface de pesquisa. No topo, há um campo de busca. Abaixo dele, uma tabela com as seguintes colunas: COD, DESCRIÇÃO e VALOR R\$. A tabela está atualmente vazia. Na base da janela, há uma barra de ferramentas com quatro botões: 'SAIR' (com ícone de seta curva), 'NOVO' (com ícone de mais), 'EDITAR' (com ícone de documento e mais) e 'EXCLUIR' (com ícone de X vermelho).

Fonte: Acervo Pessoal

TELA 13 – Cadastrar novo serviço: Nesta tela é possível cadastrar um novo serviço preenchendo todos os dados e em seguida clicando no botão salvar para que todas as informações possam ser salvas no banco de dados.

Figura 15: Cadastrar novo serviço

A janela 'SERVIÇOS....' é utilizada para o cadastro de novos serviços. Ela contém três campos de entrada: 'COD', 'DESCRIÇÃO' e 'VALOR R\$'. Na base da janela, há duas botões: 'SAIR' (com ícone de seta curva) e 'SALVAR' (com ícone de marca de seleção).

Fonte: Acervo Pessoal

TELA 14 – Ordem de Serviço: Nesta tela conseguiremos ter acesso a lista de ordens de serviços cadastrados no sistema, conseguiremos também cadastrar uma nova ordem de serviço e também visualizar alguma ordem de serviço já cadastrada.

Figura 16: Ordem de Serviço

BUSCA ORDEM DE SERVIÇO...

COD	NOME	TECNICO	DATA	TOTAL	STATUS

SAIR NOVA OS VISUALISAR OS

Fonte: Acervo Pessoal

TELA 15 – Selecionar cliente da O.S: Neste campo o usuário selecionar o nome do cliente que está deixando o aparelho para reparo.

Figura 17: Selecionar cliente da O.S

ORDEM DE SERVIÇO Nº 1

CLIENTE EQUIPAMENTO PEÇAS SERVIÇOS TÉCNICO OBSERVAÇÃO STATUS

SELECIONA UM CLIENTE

NOME

CANCELAR CONFIRMA TOTAL R\$ 0

Fonte: Acervo Pessoal

TELA 16 – Lista de equipamentos: Nesta tela é possível ver os equipamentos que o cliente está deixando e no clicando no botão “+” é possível adicionar mais equipamentos.

Figura 18: Lista de equipamentos

A imagem mostra uma janela de software intitulada "ORDEN DE SERVIÇO Nº 1". No topo, há uma barra de navegação com abas: "CLIENTE", "EQUIPAMENTO", "PEÇAS", "SERVIÇOS", "TÉCNICO", "OBSERVAÇÃO" e "STATUS". A aba "EQUIPAMENTO" está selecionada. Abaixo das abas, há uma tabela com três colunas: "id", "descricao" e "numero". A tabela está vazia. No canto inferior direito da tabela, há um botão com um símbolo de "+" azul. Na barra inferior da janela, há dois botões: "CANCELAR" (com um ícone de X vermelho) e "CONFIRMA" (com um ícone de checkmark verde). À direita desses botões, está o texto "TOTAL R\$ 0".

Fonte: Acervo Pessoal

TELA 17 – Selecionar equipamentos: Nesta tela teremos acesso a lista de equipamentos cadastrados no sistema, assim selecionamos o equipamento que corresponde ao que o cliente estará deixando e clicamos em “incluir novo equipamento” para que o equipamento selecionado seja adicionado a ordem de serviço do cliente.

Figura 19: Selecionar equipamento

A janela 'BUSCA EQUIPAMENTO' possui uma barra de busca no topo. Abaixo dela, há uma tabela com os seguintes cabeçalhos: COD, DESCRIÇÃO e NUMERO. A tabela está vazia. No rodapé da janela, há um botão com um ícone de mais (+) e o texto 'INCLUIR NOVO EQUIPAMENTO'.

Fonte: Acervo Pessoal

TELA 18 – Lista de peças: Nesta tela é possível ver as peças que serão utilizadas para reparo do equipamento do cliente e clicando no botão “+” é possível adicionar mais peças.

Figura 20: Lista de peças

A janela 'ORDEM DE SERVIÇO Nº 1' possui uma barra de navegação com as seguintes opções: CLIENTE, EQUIPAMENTO, PEÇAS, SERVIÇOS, TÉCNICO, OBSERVAÇÃO e STATUS. A aba 'PEÇAS' está selecionada. Abaixo, há uma tabela com os seguintes cabeçalhos: id, descricao, numero e valor. A tabela está vazia. No rodapé da janela, há dois botões: 'CANCELAR' (com um ícone de X vermelho) e 'CONFIRMA' (com um ícone de checkmark verde). À direita dos botões, há o texto 'TOTAL R\$ 0'. Um botão com um ícone de mais (+) está localizado no canto inferior direito da área da tabela.

Fonte: Acervo Pessoal

TELA 19 – Selecionar peças: Nesta tela teremos acesso a lista de peças cadastrados no sistema, assim selecionamos as peças que serão utilizadas para reparo do equipamento do cliente e clicamos em “incluir nova peça” para que a peça selecionado seja adicionada a ordem de serviço do cliente.

Figura 21: Selecionar peças

The screenshot shows a window titled "BUSCA PEÇAS...". At the top, there is a blue search bar. Below it is a table with the following headers: COD, DESCRIÇÃO, NUMERO, and VALOR R\$. The table is currently empty. At the bottom left, there is a blue button with a white plus sign. At the bottom right, there is a button labeled "INCLUIR NOVA PEÇA".

Fonte: Acervo Pessoal

TELA 20 – Lista de serviços: Nesta tela é possível ver os serviços que serão feitos no equipamento do cliente e clicando no botão “+” é possível adicionar mais serviços.

Figura 22: Lista de serviços

The screenshot shows a window titled "ORDEM DE SERVIÇO Nº 1". At the top, there is a tabbed interface with the following tabs: CLIENTE, EQUIPAMENTO, PEÇAS, SERVIÇOS, TÉCNICO, OBSERVAÇÃO, and STATUS. The "SERVIÇOS" tab is currently selected. Below the tabs is a large empty area for the list of services. At the bottom right of this area, there is a blue button with a white plus sign. At the bottom left, there are two buttons: "CANCELAR" (with a red X icon) and "CONFIRMA" (with a green checkmark icon). At the bottom right, there is a label "TOTAL R\$ 0".

Fonte: Acervo Pessoal

TELA 21 – Selecionar serviços: Nesta tela teremos acesso a lista de serviços cadastrados no sistema, assim selecionamos os referidos serviços que serão feitos para reparo do equipamento do cliente e clicamos em “incluir novo serviço” para que o serviço selecionado seja adicionado a ordem de serviço do cliente.

Figura 23: Selecionar serviços

The screenshot shows a window titled "BUSCA SERVIÇO". At the top, there is a blue search bar. Below it is a table with three columns: "COD", "DESCRIÇÃO", and "VALOR R\$". The table is currently empty. At the bottom of the window, there is a button with a plus icon and the text "INCLUIR NOVO SERVIÇO".

Fonte: Acervo Pessoal

TELA 22 – Selecionar técnico: Neste campo o usuário selecionar o nome do técnico que ficará responsável por reparar o equipamento deixado pelo cliente.

Figura 24: Selecionar técnico

The screenshot shows a window titled "ORDEM DE SERVIÇO Nº 1". It has a tabbed interface with tabs for "CLIENTE", "EQUIPAMENTO", "PEÇAS", "SERVIÇOS", "TÉCNICO", "OBSERVAÇÃO", and "STATUS". The "TÉCNICO" tab is selected. Below the tabs, there is a dropdown menu labeled "SELECIONA UM TÉCNICO". Below the dropdown, there is a large text box labeled "NOME". At the bottom of the window, there are three buttons: "CANCELAR" (with a red X icon), "CONFIRMA" (with a green checkmark icon), and "TOTAL R\$ 0".

Fonte: Acervo Pessoal

TELA 23 – Observação: Nesta tela o usuário poderá estar anotando alguma observação que achar necessário.

Figura 25: Observação

The screenshot shows a software window titled 'ORDEM DE SERVIÇO Nº 1'. At the top, there is a tab bar with seven tabs: 'CLIENTE', 'EQUIPAMENTO', 'PEÇAS', 'SERVIÇOS', 'TÉCNICO', 'OBSERVAÇÃO' (which is selected), and 'STATUS'. The main area of the window is a large, empty rectangular box for text entry. At the bottom of the window, there is a footer bar containing three elements: a red 'X' icon next to the text 'CANCELAR', a green checkmark icon next to the text 'CONFIRMA', and the text 'TOTAL R\$ 0'.

Fonte: Acervo Pessoal

TELA 24 – Status: Nesta tela será possível selecionar o *status* que o aparelho se encontra. Quando o *status* estiver em aberto significa que o aparelho ainda está em processo de reparo. Quando o status estiver como cancelado significa que o cliente não autorizou o serviço e se o status estiver como concluído significa que o aparelho já foi retirado pelo cliente.

Figura 26: Status

The screenshot shows the same software window 'ORDEM DE SERVIÇO Nº 1', but with the 'STATUS' tab selected. The main area of the window now displays the text 'STATUS DA ORDEM DE SERVIÇO' above a single-line dropdown menu. The footer bar remains the same, with 'CANCELAR', 'CONFIRMA', and 'TOTAL R\$ 0'.

Fonte: Acervo Pessoal

TELA 25 – Relatórios: Nesta tela podemos gerar o relatório inserindo uma data de início e uma de fim e também podemos selecionar se queremos os relatórios de entrada ou os de saída.

Figura 27: Relatórios

RELATÓRIOS

SELECIONE.... DATA DE INICIO 30/11/2017 DATA FINAL 30/11/2017 GERAR

id	nome	total	descricao
----	------	-------	-----------

SAIR

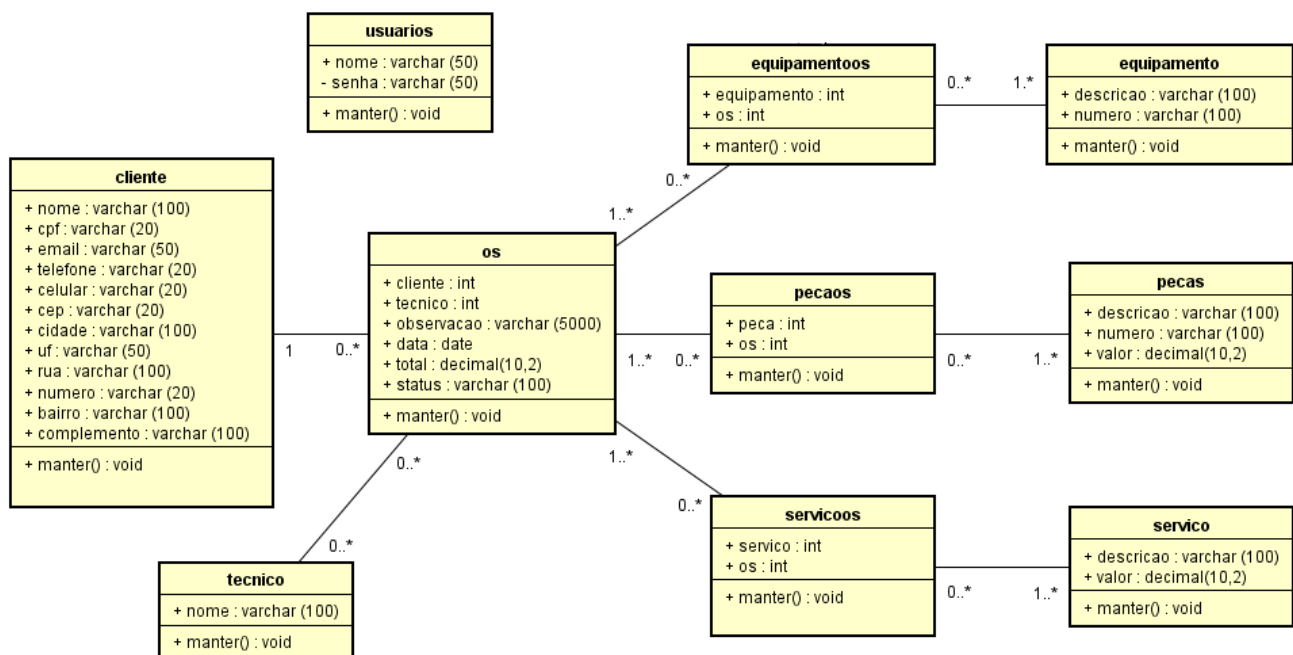
Fonte: Acervo Pessoal

Apêndice B: Diagramas

Nesta etapa são apresentados os demais diagramas utilizados para a modelagem do sistema OrderOfService.

DIAGRAMA 2: Classes: O Diagrama de Classes permite a visualização dos componentes da base de dados do Sistema.

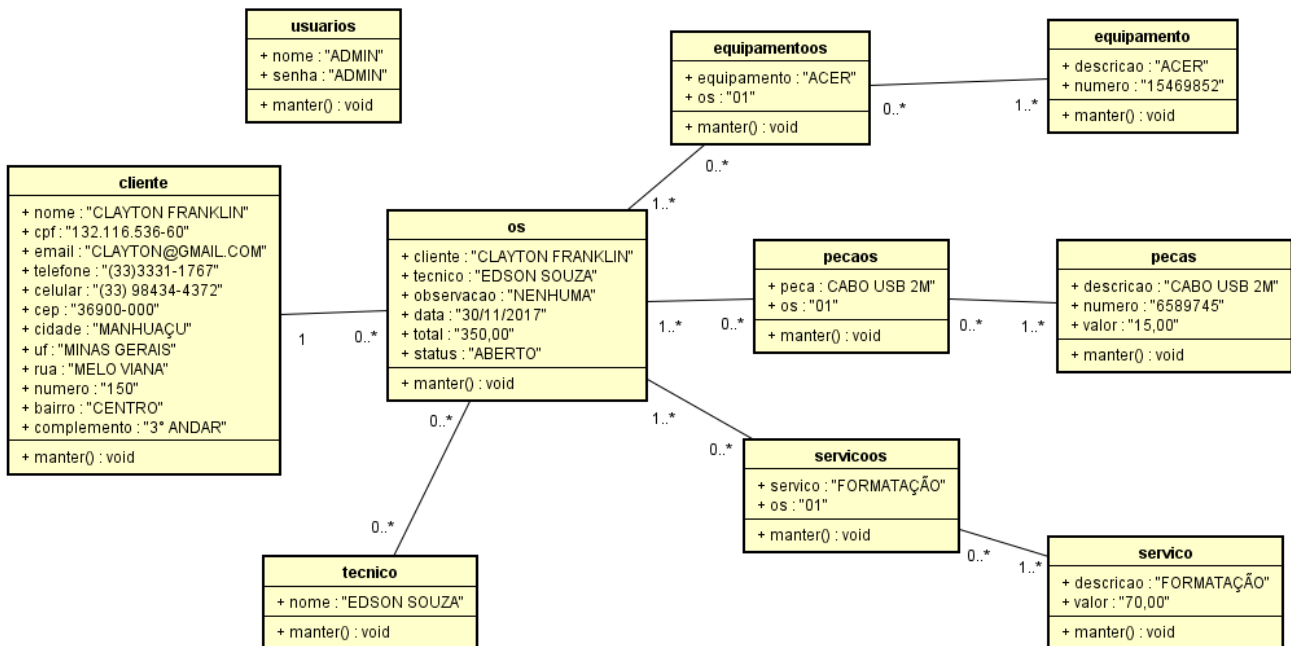
Figura 28: Diagrama de Classes



Fonte: Acervo Pessoal

DIAGRAMA 3: Objetos: O Diagrama de Objetos é bem parecido com o de Classes, a diferença é que o de Objetos representa uma instância de uma classe específica, ou seja, mostra os valores armazenados pelos objetos das classes num determinado intervalo de tempo.

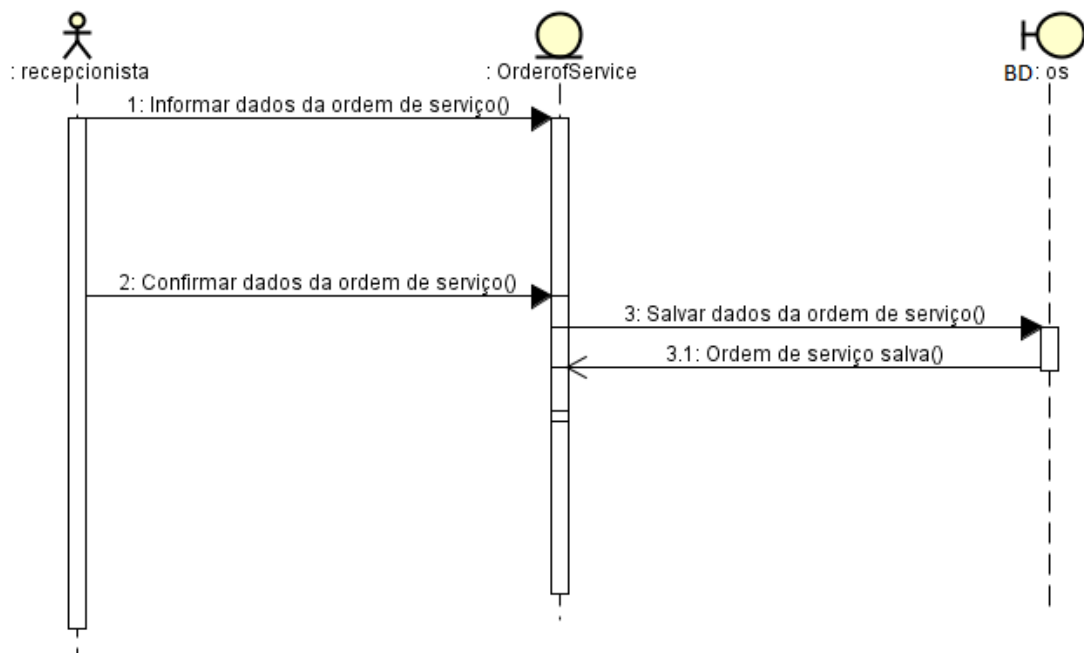
Figura 29: Diagrama Objeto



Fonte: Acervo Pessoal

DIAGRAMA 4: Sequência: Este tipo de diagrama descreve ao longo de uma linha de tempo a sequência de comunicações entre objetos de um sistema de informação.

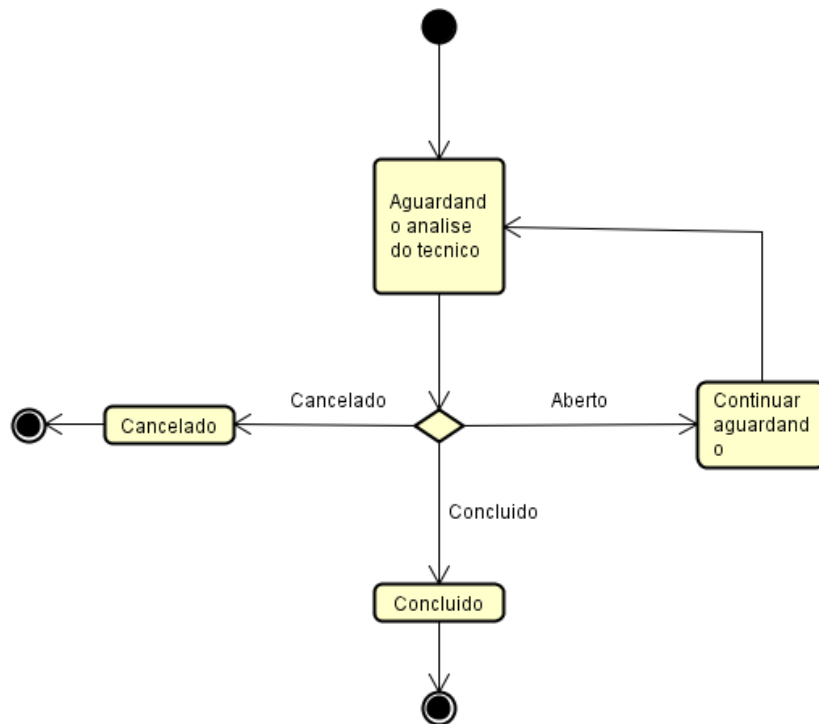
Figura 30: Diagrama de Sequência



Fonte: Acervo Pessoal

DIAGRAMA 5: Máquina de Estado: A função principal do Diagrama de Máquina de Estado é ilustrar o comportamento dos objetos quando reagem a estímulos externos a eles.

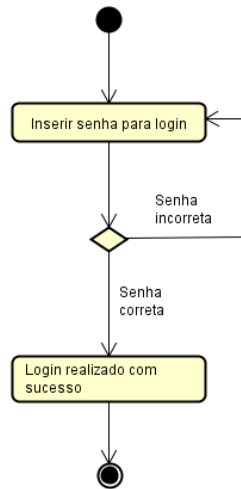
Figura 31: Diagrama de Máquina de Estado



Fonte: Acervo Pessoal

DIAGRAMA 6: Atividades: Este diagrama representa os aspectos dinâmicos, pode ser utilizado para modelar um sistema de informação, alguns módulos deste, uma pequena parte do código de um dos seus programas, um algoritmo ou os processos (fluxos de trabalho).

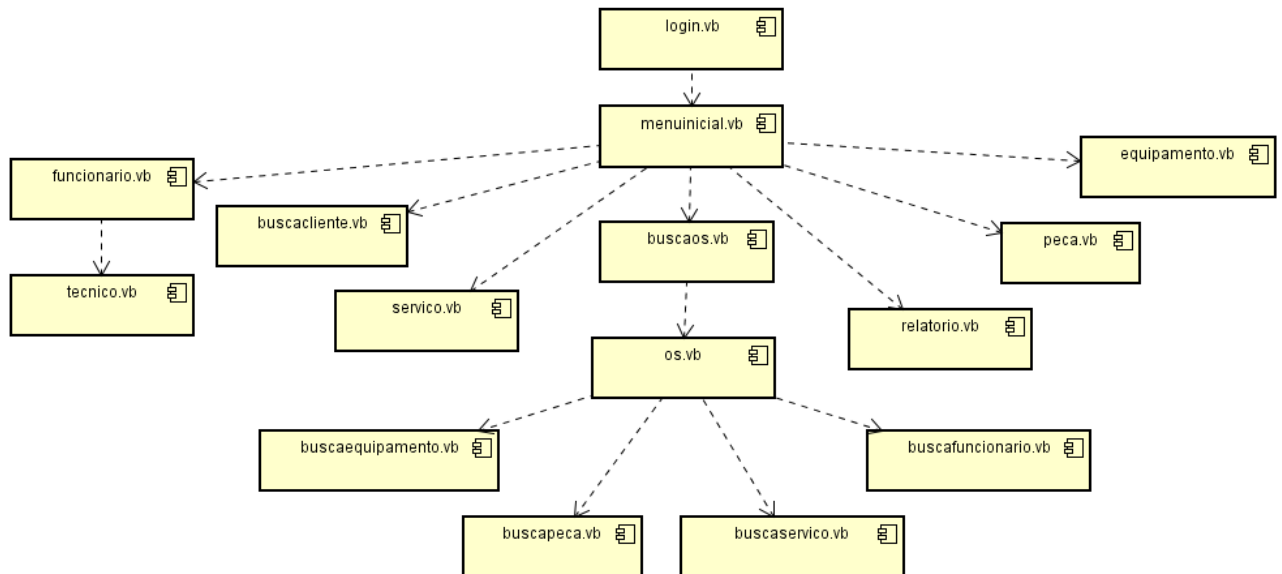
Figura 32: Diagrama de Atividades



Fonte: Acervo Pessoal

DIAGRAMA 7: Componentes: É função do Diagrama de Componente documentar como os arquivos físicos estão estruturados, organizados e se relacionam, assim permitindo uma melhor compreensão.

Figura 33: Diagrama de Componentes



Fonte: Acervo Pessoal