

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Colegiado de Sistemas de Informação**



UFOP
Universidade Federal
de Ouro Preto

**Desenvolvimento de um sistema Web
para utilização e gerenciamento de
dados de Cupons Fiscais e Saúde**

Bernardo Gomes de Abreu

**TRABALHO DE
CONCLUSÃO DE CURSO**

ORIENTAÇÃO:
Filipe Nunes Ribeiro
COORIENTAÇÃO:
Sérgio Evangelista Silva

**Agosto, 2016
João Monlevade/MG**

ANEXO IV - Ata de Defesa

ATA DE DEFESA

Aos 08 dias do mês de agosto de 2016, às 10 horas, na sala _____ do Instituto de Ciências Exatas e Aplicadas, foi realizada a defesa de Monografia pelo aluno **Bernardo Gomes de Abreu**, sendo a Comissão Examinadora constituída pelos professores: Prof. MSc. Filipe Nunes Ribeiro, Prof. Dr. Sérgio Evangelista Silva, Prof. MSc. Vicente José Peixoto de Amorim e Profa. MSc. Helen de Cássia Sousa da Costa Lima.

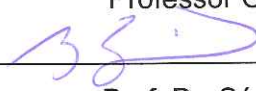
O candidato apresentou a monografia intitulada: "*Desenvolvimento de um Servidor Web para gerenciamento de dados de Cupons Fiscais e Saúde*". A comissão examinadora deliberou, por unanimidade, pela aprovação do candidato, com nota 9,0 (nove), concedendo-lhe o prazo de 15 dias para incorporação das alterações sugeridas ao texto final.

Na forma regulamentar, foi lavrada a presente ata que é assinada pelos membros da Comissão Examinadora e pelo graduando.

João Monlevade, 08 de agosto de 2016



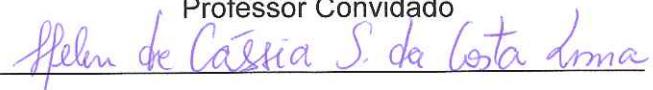
Prof. MSc. Filipe Nunes Ribeiro
Professor Orientador/Presidente



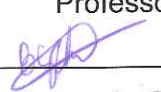
Prof. Dr. Sérgio Evangelista Silva
Professor Coorientador



Prof. MSc. Vicente José Peixoto de Amorim
Professor Convidado



Profa. MSc. Helen de Cássia Sousa da Costa Lima
Professor Convidado



Bernardo Gomes de Abreu
Graduando



UFOP
Universidade Federal
de Ouro Preto

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
COLEGIADO DO CURSO DE SISTEMAS DE INFORMAÇÃO

ANEXO V – Folha de Aprovação
Curso de Sistemas de Informação

FOLHA DE APROVAÇÃO DA BANCA EXAMINADORA

Desenvolvimento de um Servidor Web para gerenciamento de dados de Cupons Fiscais e Saúde

Bernardo Gomes de Abreu

Monografia apresentada ao Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto como requisito parcial da disciplina CSI499 – Trabalho de Conclusão de Curso II do curso de Bacharelado em Sistemas de Informação e aprovada pela Banca Examinadora abaixo assinada:

Prof. MSc. Filipe Nunes Ribeiro

DECSI - UFOP

Prof. Dr. Sérgio Evangelista Silva

DEENP - UFOP

Prof. MSc. Vicente José Peixoto de Amorim

DECSI - UFOP

Profa. MSc. Helen de Cássia Sousa da Costa Lima

DECSI - UFOP

João Monlevade, 8 de Agosto de 2016



UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E APLICADAS
COLEGIADO DO CURSO DE SISTEMAS DE INFORMAÇÃO

UFOP

Universidade Federal
de Ouro Preto

ANEXO III – Termo de Responsabilidade

TERMO DE RESPONSABILIDADE

Eu, Beauvardo Gomes de Azevedo,
declaro que o texto do trabalho de conclusão de curso intitulado
"DESENVOLVIMENTO DE UM SISTEMA WEB PARA UTILIZAÇÃO
E GERENCIAMENTO DE DADOS DE CUPONS FISCAIS E SAÚDE" é de
minha inteira responsabilidade e que não há utilização de texto, material fotográfico, código
fonte de programa ou qualquer outro material pertencente a terceiros sem as devidas
referências ou consentimento dos respectivos autores.

João Monlevade, 08 de AGOSTO de 2016


Assinatura do aluno

Bernardo Gomes de Abreu

**Desenvolvimento de um sistema Web para
utilização e gerenciamento de dados de Cupons
Fiscais e Saúde**

Orientador: Filipe Nunes Ribeiro

Coorientador: Sérgio Evangelista Silva

Monografia apresentada ao curso de Sistemas de Informação do Departamento de Computação e Sistemas da Universidade Federal de Ouro Preto como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação

Universidade Federal de Ouro Preto

João Monlevade

Agosto de 2016

Bernardo Gomes de Abreu

Desenvolvimento de um sistema Web para utilização e gerenciamento de dados de Cupons Fiscais e Saúde/ Bernardo Gomes de Abreu. – João Monlevade, 08 de agosto de 2016-

Orientador: Filipe Nunes Ribeiro

Monografia (graduação) – Universidade Federal de Ouro Preto, 08 de agosto de 2016.

CDU 02:141:005.7

Agradecimentos

Agradeço todos os dias à Deus por tudo, e em especial por ter me colocado junto a pessoas tão generosas que fizeram de mim um alguém melhor. Sem o apoio Dele não teria sido colocado em uma família tão generosa e atenciosa que em todos os momentos está preocupada em buscar o melhor para mim.

Agradeço ao meu Avô Zizi, por tudo. Sempre foi gentil me levando para jogar bola e para festas. Nunca houve tempo ruim, sempre conversando e ajudando. Minha avó Ney, sempre tão gentil e preocupada, até demais. Sempre foi um amor, cozinhando as melhores comidas possíveis!

E claro, não poderia faltar o agradecimento à minha GRANDE Mãe Soraya, que apesar de todas as dificuldades de criar uma família quase sozinha, sempre me ajudou em tudo.

Acredito que sou uma pessoa de muita sorte por ter esses três pilares como base. Sem ensinamentos e a educação que me proveram, não seria a pessoa realizada que sinto que sou.

Também agradeço ao meu pai Laercio que, apesar de distante, sempre me apoiou quando precisei. Sempre recordo de bons momentos de pai e filho.

Também agradeço ao meu irmão que sempre foi amigo. E minha grande Yasmine Melo por estar ao meu lado nos últimos 3 anos e nos próximos todos que ainda virão. Novamente agradeço a Deus por tê-los ao meu lado.

Meus agradecimentos ao Colégio Tiradentes no qual eu estudei por toda minha vida. Aprendi muito com todos os professores que compartilharam comigo seu conhecimento.

Agradeço também à Ordem Demolay, onde conheci bons amigos e reafirmei valores importantes que carrego comigo até hoje.

Agradeço à Republica PCC, que nesses 4 anos em que fui morador, aprendi muita coisa e melhorei meu jeito de ser. Conheci pessoas fantásticas que contribuíram muito em minha vida. Lá não existem amigos, e sim irmãos.

Agradeço à UFOP que contribuiu muita nessa etapa importante da minha formação. À todos os professores que contribuíram em meu crescimento. Em especial aos professores Filipe e Sérgio que me acompanharam na etapa de TCC.

Enfim, meu muito obrigado a todos que contribuíram para a conclusão dessa etapa que se encerra hoje.

Resumo

No dia-a-dia muitos tipos de dados são gerados, mas a maioria deles se perder ou são armazenados incorretamente. Se isso acontecer, os dados não podem ser convertidos em informações úteis. O objetivo deste trabalho final é criar um sistema de coleta e armazenamento de dados a partir do qual podemos extrair informações úteis para os usuários. Para atingir esse objetivo, foram desenvolvidas duas aplicações: um sistema para o usuário final acessar as informações em um único local pelo armazenamento em nuvem, um aplicativo móvel para receber e armazenar informações sobre saúde e um terceiro aplicativo que foi modificado para obter a compra dados.

Palavras-chaves: Computação em nuvem. *Web*. Aplicativo Móvel. Armazenamento de Dados. Desenvolvimento.

Abstract

In day-to-day many kinds of data are generated but most of them get lost or are stored incorrectly. If it happens, data can not be converted into useful information. The purpose of this final paper is to create a system of data collection and storage from which we can extracted useful information for the users. In order to achieve this goal, two applications were developed: a system to the end user access the information in a single place by the cloud storage, a mobile application to receive and store information about health and a third application that was modified to obtain purchase data.

Key-words: *Cloud computing. Textit Web. Mobile Application. Data storage. Development.*

Lista de ilustrações

Figura 1 – Computação em nuvem.	20
Figura 2 – Cupom X NFCe.	24
Figura 3 – Taxa de mortalidade	25
Figura 4 – Suporte a tag Date Time em formulário. Fonte: http://caniuse.com/feat=input-datetime	31
Figura 5 – Sintaxe de HTML a esquerda e código renderizado no navegador a direita.	32
Figura 6 – Regra de <i>CSS</i>	33
Figura 7 – Sintaxe de <i>HTML</i> com <i>CSS</i> a esquerda e código renderizado no navegador a direita.	33
Figura 8 – Fonte: Eduardo Palma - Framework para desenvolvimento web.	35
Figura 9 – Modelo UML do ciclo de vida do <i>software</i>	38
Figura 10 – Módulo Cupom Fiscal	40
Figura 11 – Módulo Saúde	40
Figura 12 – Modelagem Conceitual	42
Figura 13 – Modelagem Lógica	43
Figura 14 – App Saúde na nuvem. Tela de login.	47
Figura 15 – Cenários de comunicação com o servidor.	48
Figura 16 – Arquitetura do sistema	50
Figura 17 – Funcionamento do sistema	51
Figura 18 – Tela de <i>login</i>	54
Figura 19 – Página de cupons fiscais	55
Figura 20 – Modal(tela) de detalhes de um cupom fiscal.	55
Figura 21 – Tela de busca de biodados	56
Figura 22 – Gráfico anual por estabelecimento	57
Figura 23 – Gráfico de intervalo selecionado dividido por tipo de estabelecimento	57
Figura 24 – Gráfico anual por tipo de estabelecimento	58
Figura 25 – Página do gráfico mensal de batimentos cardíacos	58
Figura 26 – Página do gráfico mensal de pressão arterial	59
Figura 27 – Tela principal mobile	61
Figura 28 – Novo registro de pressão arterial	61

Lista de tabelas

Tabela 1 – Hipertensão	27
Tabela 2 – Loja hospedada na nuvem: Fonte(Amazon Web Services Ação)	30
Tabela 3 – Caso de uso detalhado: Registrar Usuário	41

Lista de abreviaturas e siglas

API	Interface de Programação de Aplicativos
COFINS	Contribuição para o Financiamento da Seguridade Social
DLL	Biblioteca de vínculo dinâmico
ECF	Emissor de cupom fiscal
ICMS	Imposto Sobre Circulação De Mercadorias E Serviços
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
MVC	Modelo, Visão, Controlador
MVT	Modelo, <i>View</i> (Controlador), <i>Template</i> (Visão)
OMS	Organização Mundial da Saúde
PIS	Programa de Integração Social
REST	Transferência de Estado Representacional
XML	Linguagem Extensível de Marcação Genérica

Sumário

	Introdução	19
1	OBJETIVOS	21
1.1	Objetivo Geral	21
1.2	Objetivos Específicos	21
2	REFERENCIAL TEÓRICO	23
2.1	Cupons Fiscais	23
2.1.1	Nota Fiscal do Consumidor Eletrônica(NFC-e)	24
2.2	Doenças Crônicas	24
2.2.1	Diabetes	26
2.2.2	Hipertensão	27
2.3	Computação em nuvem	28
2.3.1	Vantagens da Computação em Nuvem	29
2.3.2	Acesso aos dados	30
2.3.2.1	HTML	30
2.3.2.2	CSS	32
2.3.2.3	JavaScript	32
2.3.2.4	Python	34
2.3.2.5	Framework	34
2.3.2.6	Django	35
3	ENGENHARIA DE SOFTWARE	37
3.1	Análise de Requisitos	38
3.2	Casos de Uso	39
3.2.1	Diagrama de Caso de Uso	39
3.2.2	Caso de Uso Detalhado	41
3.3	Banco de Dados	42
3.3.1	Modelagem Conceitual	42
3.3.2	Modelagem Lógica	43
4	DESENVOLVIMENTO	45
4.1	Cupons Fiscais - Dispositivos em estabelecimentos	45
4.2	Saúde - Móvel	46
4.3	Arquiteturas	49
4.4	Ferramentas	52

4.4.1	Ambiente	52
4.4.2	Bibliotecas e Frameworks	53
5	EXECUÇÃO DOS SISTEMAS	53
5.1	Sistema Web	53
5.1.1	<i>Login</i>	54
5.1.2	Buscas	54
5.1.2.1	Cupom Fiscal	54
5.1.2.2	Saúde - Pressão Arterial, Batimentos Cardíacos, Glicose	55
5.1.3	Relatórios	56
5.1.3.1	Cupom Fiscal	56
5.1.3.2	Saúde - Pressão Arterial, Batimentos Cardíacos, Glicose	58
5.2	Aplicativo Móvel	60
6	CONCLUSÃO	63
6.1	Trabalhos Futuros	64

REFERÊNCIAS	65
------------------------------	-----------

ANEXOS	69
-------------------------	-----------

	ANEXO A – CASOS DE USO	71
A.0.1	Caso Dois	71
A.0.2	Caso Três	72
A.0.3	Caso Quatro	73
A.0.4	Caso Cinco	74
A.0.5	Caso Seis	75
A.0.6	Caso Sete	76
A.0.7	Caso Oito	77
A.0.8	Caso Nove	78
A.0.9	Caso Dez	79
	ANEXO B – EXECUTANDO O SISTEMA	81
B.1	Sistema Web	81
B.1.1	Executando	81
B.1.2	Banco de Dados	81
B.1.3	Testes	82
B.2	Aplicativo Mobile	82

Introdução

Passada a Era Industrial, estamos vivendo hoje a Era da Informação. Também conhecida como Era Digital ou Era Tecnológica. Uma das características mais marcantes do atual período histórico é, sem dúvida, a ampliação da capacidade de armazenamento e memorização de informações, dados e formas de conhecimentos.

Toda essa evolução quanto à informação está diretamente ligada aos avanços técnicos que permitiram o progresso e a consolidação da globalização pelo mundo, no qual as distâncias foram encurtadas e a velocidade das transformações tecnológicas foi acelerada. A informação é a nova matéria prima. As tecnologias se desenvolveram para atuar sobre a informação propriamente dita.

São situações corriqueiras na sociedade da informação: Educação à distância, bibliotecas digitais, videoconferência, correio eletrônico, grupos de "bate-papo", além de voto eletrônico, banco on-line, vídeo sob demanda, comércio eletrônico e trabalho à distância. Tudo isso é hoje parte integrante da vida diária na maioria dos grandes centros urbanos no mundo. E a base de todos esses serviços é a informação disponível de modo online ao usuário.

Com o avanço da tecnologia, muitos documentos importantes vêm sendo armazenados em computadores (HD's, HD's externos, *pen drives*, CD's, etc.), mas infelizmente esses *hardwares* estão sujeitos a vírus, danos pelo tempo, danos por mau uso, de modo que toda a informação contida neles fique vulnerável e passível de perda.

Por isso a computação em nuvem e o armazenamento em nuvem se tornaram o método preferencial para a distribuição de informações e funcionalidade online. Cada parte desta infraestrutura é provida como um serviço e, estes são normalmente alocados em centros de dados, utilizando *hardware* compartilhado para computação e armazenamento (BUYYA R.; BRANDIC, 2009b).

A palavra "nuvem" faz alusão à ideia de um local onde qualquer usuário tenha acesso através da Internet, de modo que, fique possível acessar todas as suas informações (fotos, documentos, vídeos, entre outros). Para visualizar seus arquivos em nuvens é necessário um equipamento ligado à internet (*Tablet*, Celular, *Notebook*, etc) e pelo navegador ou algum meio de acesso da nuvem como aplicativo ou programa, os arquivos são acessados.

No cotidiano da era da informação existe uma série de transações que geram documentos que devem ser armazenados, como cupons fiscais, boletins acadêmicos, exames de saúde, comprovantes em geral. Mesmo com a forte presença da computação em nuvem, que permite o armazenamento de dados e processamento remoto, boa parte dos documentos

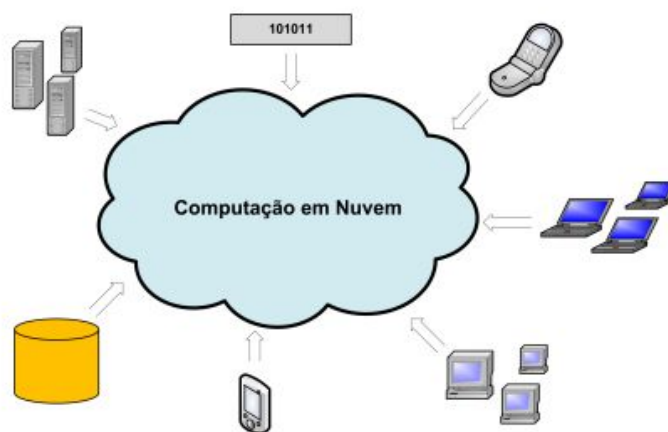


Figura 1 – Computação em nuvem.

mencionados anteriormente ainda são registrados em papel.

Este trabalho surgiu como sugestão de continuidade do Trabalho de Conclusão de Curso (TCC) da aluna Jéssica Pires de Freitas, que tem o título "Cupom Fácil: Sistema de envio de cupons fiscais para dispositivos móveis" e consistia em um sistema com a função de obter e enviar cupons fiscais para o cliente. O mesmo foi desenvolvido em dois sistemas distintos. Um deles desenvolvido na linguagem JAVA, para funcionar em qualquer sistema operacional, obtendo as informações da memória da impressora fiscal durante o lançamento de uma nota fiscal eletrônica. O outro sistema consistia em um aplicativo (APP) desenvolvido para a plataforma de telefones móveis que possuem o sistema operacional *Android*.

O projeto desenvolvido nesse presente Trabalho de Conclusão de Curso tem o intuito de receber os documentos diversos gerados no dia-a-dia do brasileiro e realizar seu armazenamento para controle e futuras consultas.

O presente trabalho foi iniciado em janeiro de 2014 e finalizada a codificação em dezembro de 2014. Por motivos profissionais, a parte escrita do trabalho foi postergada, perante a requerimento de trancamento de período, sendo reiniciada em fevereiro de 2016.

Nesse intervalo entre o final do desenvolvimento e a apresentação, foi apresentado um trabalho de conclusão de curso cujo tema era parecido com o módulo de saúde apresentado neste trabalho. Todavia, o foco principal deste trabalho não foi sobreposto pois o tema principal foi a criação de um sistema de armazenamento de dados onde o usuário poderia armazenar dados pessoais.

1 Objetivos

1.1 Objetivo Geral

O objetivo principal deste trabalho é criar um ambiente de nuvem que permita o armazenamento de diversos tipos de documentos gerados das transações cotidianas. A partir desse ambiente o usuário irá visualizar os dados armazenados tendo a opção de gerar relatórios de controle.

Para o trabalho foram desenvolvidos dois módulos para esse mesmo objetivo. O primeiro armazena dados relacionados a informações de compras do usuário. Com os dados acessíveis na nuvem, reduz-se a necessidade da emissão dos cupons fiscais em papel, reduzindo o consumo do mesmo.

O segundo módulo faz o armazenamento dos dados resultados de medições de informações de saúde importantes para boa qualidade de vida tais como: pressão arterial, nível de glicose no sangue e batimento cardíaco. Essas informações foram chamadas no contexto deste trabalho de Biodados. Assim, tanto o médico quanto o paciente podem ter acesso à todas essas informações de maneira a fazer o acompanhamento dos níveis de forma eficiente.

1.2 Objetivos Específicos

Para a realização desse projeto é necessário o desenvolvimento de alguns sistemas que em conjunto serão o ambiente em nuvem. São eles:

- Desenvolvimento de um servidor *web* que possuirá sistema de login, endereços(*urls*) das páginas amigáveis, interfaces responsivas. Além disso será modular compostos pelos seguintes partes:
 - Cupom Fiscal: manipulará os dados sobre cupons fiscais presente no ambiente.
 - Saúde: sua função será manipular os dados da saúde do usuário que serão enviados pelo aplicativo.

Cada módulo será composto por duas partes. O *backend*, que trata-se da parte invisível do *software* para o usuário, mas é nele que está implementado toda as regras de negócios, como: *webservices* e *API's* da aplicação. A outra parte é *frontend* que são as telas de interface para o usuário interagir com o sistema.

- Um aplicativo na linguagem *Android* para o usuário enviar os biodados ao servidor sobre a medição da saúde.

2 Referencial Teórico

O presente trabalho aborda áreas distintas, não ficando apenas na área da computação. Por isso, visando uma melhor compreensão do mesmo, serão apresentados conceitos, tecnologias e informações que foram utilizados no seu desenvolvimento.

Na seção 2.1 abordaremos os cupons fiscais. A seção 2.2 será voltada para doenças crônicas. A computação em nuvem junto com as tecnologias que permitem o seu acesso será explicado na seção 2.3 .

2.1 Cupons Fiscais

Para o funcionamento do Estado, são necessários meios de captação de recursos financeiros. O modo mais amplamente utilizado no mundo é o imposto.

Um dos impostos recolhidos no Brasil trata-se da taxa que incide sobre a venda de produtos ou prestação de serviços. O valor que o estabelecimento paga ao estado é um percentual de cada movimentação financeira realizada. Como exemplo desses impostos possuímos ICMS (Imposto sobre Circulação de Mercadorias e Serviços), PIS (Programa de Integração Social) e COFINS (Contribuição para Financiamento da Seguridade Social).

O Brasil tem a maior carga tributária da América Latina e também supera os países ricos na média. Um dos meios para combater essa ação ilícita foi a criação de um documento que substituiu a antiga nota fiscal: o Cupom Fiscal. Através dele, a Receita Federal consegue realizar uma apuração com maior precisão sobre os tributos que devem ser pagos ao Estado. A emissão desse documento é obrigatória para todas as empresas de comércio que faturam a partir de R\$ 120.000,00 ao ano.

O documento mencionado acima, é emitido através de um equipamento chamado de ECF (Emissor de Cupom Fiscal). O ECF é o equipamento de automação comercial com capacidade de emitir documentos fiscais e realizar controles dos valores referentes a operações de circulação de mercadorias ou a prestações de serviços, e é composto por *hardware*, sistema e um certificado digital.

Os documentos gerados pelo ECF são autenticados, após esse procedimento, são impressos em papel e alguns dados são armazenados no equipamento. Esses dados são periodicamente enviados para a Secretaria da Fazenda (SEFAZ) através de uma conexão local com a internet para controle. Vale ressaltar que antes da utilização desse aparelho, é preciso que ele passe por uma homologação e um processo de lacração para evitar manipulações.

2.1.1 Nota Fiscal do Consumidor Eletrônica(NFC-e)

A NFC-e visa ser uma alternativa totalmente eletrônica para os documentos fiscais que estamos habituados a receber nas operações de compra. É bastante parecida com a versão 2.0 da NF-e, porém é completamente digital.

A proposta deste documento é substituir o cupom fiscal e a nota fiscal de venda. A utilização do NFC-e traz consigo algumas vantagens como o fim das impressões em papéis; envio em tempo real das informações para a Secretaria da Fazenda; e fim da necessidade de um *hardware* específico (equipamentos ou impressoras fiscais). Com isso também serão dispensadas homologações, lacrações, registros, parametrizações, dentre outras exigências.

Cupom Fiscal	NFC-e																																																																																													
RAZÃO SOCIAL DA EMPRESA RUA FULANO DE TAL, 123 - SÃO PAULO - SP CNPJ:00.000.000/0001-00 IE:000.000.000 IM:0.000.000-3 01/01/2013 23:59:59 CCF:000001 COD:000001 CNPJ/CPF consumidor: 000.000.000-00 CUPOM FISCAL <table><tr><th>ITEM</th><th>CODIGO</th><th>DESCRICAO</th><th>QTD.</th><th>UN.</th><th>VL.UNIT(R\$)</th><th>ST</th><th>TAT</th><th>VL.ITEM(R\$)</th></tr><tr><td>001</td><td>11202230</td><td>AGUA COCOKERO 200 ML</td><td>1.000</td><td>UN</td><td>x 1,85</td><td>N1</td><td>A</td><td>1,85</td></tr><tr><td>002</td><td>15154888</td><td>COCACOLA PET 2.5 LT</td><td>1.000</td><td>UN</td><td>x 4,65</td><td>N1</td><td>A</td><td>4,65</td></tr><tr><td>003</td><td>25220041</td><td>AZEITE GALO LT 500ML</td><td>1.000</td><td>UN</td><td>x 14,95</td><td>N1</td><td>A</td><td>14,95</td></tr><tr><td>004</td><td>25145153</td><td>SORVETE HDAZS PT 1 LT</td><td>1.000</td><td>UN</td><td>x 20,60</td><td>N1</td><td>A</td><td>20,60</td></tr><tr><td colspan="8">Subtotal R\$</td><td>42.05</td></tr><tr><td colspan="8">DESCONTO-ICMS 10.00%</td><td>-4.20</td></tr><tr><td colspan="8">TOTAL R\$</td><td>37.85</td></tr><tr><td colspan="8">Dinheiro</td><td>37.85</td></tr></table> 2>02>hp+lpRRtbxjhg+powerEdByvinCO+T\$mi#*&u=bvgff4 2PM ZPH/1FIT LOGGER ECF-IF VERSAO:01.01.01 ECF:001 LOJA:0001 OPERADOR:001 CGCAPINSAC 01/01/2013 23:59:59V FAB:ZP0000000000	ITEM	CODIGO	DESCRICAO	QTD.	UN.	VL.UNIT(R\$)	ST	TAT	VL.ITEM(R\$)	001	11202230	AGUA COCOKERO 200 ML	1.000	UN	x 1,85	N1	A	1,85	002	15154888	COCACOLA PET 2.5 LT	1.000	UN	x 4,65	N1	A	4,65	003	25220041	AZEITE GALO LT 500ML	1.000	UN	x 14,95	N1	A	14,95	004	25145153	SORVETE HDAZS PT 1 LT	1.000	UN	x 20,60	N1	A	20,60	Subtotal R\$								42.05	DESCONTO-ICMS 10.00%								-4.20	TOTAL R\$								37.85	Dinheiro								37.85	NFC-e RAZÃO SOCIAL DA EMPRESA CNPJ: 00.000.000/0000-99 IE: 000000000000 00 Endereço Completo (logradouro, número, bairro, município, UF) DANFE NFC-e - Documento Auxiliar da Nota Fiscal de Consumidor Eletrônica Não permite aproveitamento de crédito de ICMS ÁREA DO DETALHE DA VENDA <table><tr><td>QTD. TOTAL DE ITENS</td><td>7</td></tr><tr><td>VALOR TOTAL R\$</td><td>54,95</td></tr><tr><td>FORMA PAGAMENTO</td><td>VALOR PAGO R\$</td></tr><tr><td>Dinheiro</td><td>4,95</td></tr><tr><td>Cheque</td><td>50,00</td></tr><tr><td colspan="2">Tributos Totais Incidentes (Lei Federal 12.741/2012) R\$ 9.999.999,99</td></tr></table> ÁREA DE MENSAGEM FISCAL NFC-e nº 000000001 Série 001 Emissão DD/MM/AAAA hh:mm:ss Via Consumidor Consulte pela Chave de Acesso em www.nfce.sefaz.uf.gov.br CHAVE DE ACESSO 4321 9205 5565 1101 1250 5300 1000 0040 7720 7054 8065 CONSUMIDOR CNPJ / CPF (ID Estrangeiro : 000000000000000000) NOME DO CONSUMIDOR Endereço Completo (logradouro, número, bairro, município, UF) Consulte via leitor de QR Code  Protocolo de autorização: 1412130000008502 DD/MM/AAAA hh:mm:ss	QTD. TOTAL DE ITENS	7	VALOR TOTAL R\$	54,95	FORMA PAGAMENTO	VALOR PAGO R\$	Dinheiro	4,95	Cheque	50,00	Tributos Totais Incidentes (Lei Federal 12.741/2012) R\$ 9.999.999,99	
ITEM	CODIGO	DESCRICAO	QTD.	UN.	VL.UNIT(R\$)	ST	TAT	VL.ITEM(R\$)																																																																																						
001	11202230	AGUA COCOKERO 200 ML	1.000	UN	x 1,85	N1	A	1,85																																																																																						
002	15154888	COCACOLA PET 2.5 LT	1.000	UN	x 4,65	N1	A	4,65																																																																																						
003	25220041	AZEITE GALO LT 500ML	1.000	UN	x 14,95	N1	A	14,95																																																																																						
004	25145153	SORVETE HDAZS PT 1 LT	1.000	UN	x 20,60	N1	A	20,60																																																																																						
Subtotal R\$								42.05																																																																																						
DESCONTO-ICMS 10.00%								-4.20																																																																																						
TOTAL R\$								37.85																																																																																						
Dinheiro								37.85																																																																																						
QTD. TOTAL DE ITENS	7																																																																																													
VALOR TOTAL R\$	54,95																																																																																													
FORMA PAGAMENTO	VALOR PAGO R\$																																																																																													
Dinheiro	4,95																																																																																													
Cheque	50,00																																																																																													
Tributos Totais Incidentes (Lei Federal 12.741/2012) R\$ 9.999.999,99																																																																																														

Figura 2 – Cupom X NFCE.

Fonte:<http://vinco.com.br/nfc-e>

A Figura 2 ilustra um cupom fiscal e um Documento Auxiliar da NFC-e (DANFE NFC-e). Caso o cliente queira visualizar os itens da NFC-e basta utilizar um leitor de QRCode, que ao ler o código exibirá o *link* da compra. O *link* direciona o usuário para o site da Receita que contém todas as informações sobre a compra.

2.2 Doenças Crônicas

Existem doenças que demoram muito tempo para se curarem, ou até mesmo não tem cura. Essas são chamadas de doenças crônicas. Em alguns casos, as doenças crônicas são assintomáticas, ou seja, o portador não apresenta nenhum sintoma, o que pode levar o indivíduo uma identificação tardia da doença.

Geralmente essas doenças estão ligadas com o avanço de idade, má alimentação e sedentarismo. As doenças crônicas mais conhecidas são o Tabagismo, Epilepsia, Hipertensão,

Diabetes, Alzheimer, Parkinson, HIV e Hepatites. Elas podem ser divididas em dois grupos: doenças crônicas transmissíveis e não transmissíveis.

De acordo com a OMS(Organização Mundial da Saúde), esse tipo de doença é a causa de 72% das mortes por ano no mundo. No Brasil, de acordo com dados do IBGE, mais de 75% da população com mais de 60 anos tem alguma doença crônica. De acordo com o Ministério da Saúde, no ano de 2009, as que mais mataram foram as cardiovasculares (31,3%), o câncer (16,2%), as doenças respiratórias crônicas (5,8%) e o diabetes mellitus (5,2%).

A Figura 3 mostra o resultado de uma pesquisa realizada em 2008 sobre as pessoas que declararam possuir alguma doença crônica. Percebemos que a taxa é altíssima para pessoas com idade avançada.

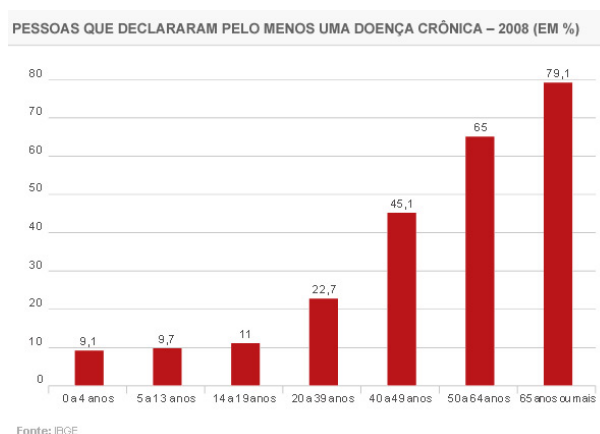


Figura 3 – Taxa de mortalidade

Apesar das estatísticas acima, mesmo sendo portador de alguma doença crônica, o indivíduo pode levar uma vida longa se fizer mudanças no estilo de vida, como adoção de dieta, hábitos saudáveis e prática de exercícios físicos.

O governo brasileiro está trabalhando para combater essas doenças. No caso das doenças que são provocadas pelo sedentarismo, foi criado o Plano de Ações Estratégicas para o Enfrentamento das Doenças Crônicas Não Transmissíveis no Brasil. Esse programa determinou a implantação de polos com infraestrutura, equipamentos e profissionais qualificados para a orientação de práticas corporais, atividades físicas e lazer. A meta era construir até 2014 cerca de 4 mil unidades para estimular o exercício físico. No caso do Tabagismo, o governo trabalha aumentando o imposto dos produtos derivados do tabaco, o que provocou um declínio no uso do cigarro de 34% da população em 1980 para 15% em 2010.

2.2.1 Diabetes

Diabetes é uma doença crônica que provoca a alta concentração de açúcar no sangue. Ela está relacionada ao hormônio insulina que, nos casos de diabéticos, não age adequadamente ou não é produzida na quantidade necessária pelo pâncreas.

Nos indivíduos que não possuem diabetes, após a alimentação, o sistema digestivo transforma parte do alimento ingerido em açúcar, que é levado até as células como forma de energia. O paciente que possui a diabetes tem problemas na distribuição desse açúcar para a célula, assim, a glicose fica presente em grande quantidade no sangue.

A diabetes ocorre em grande escala no Brasil. De acordo com o Ministério da Saúde, existem cerca de 11 milhões de indivíduos com essa doença no Brasil, um número muito alto. Maior ainda é a previsão da Federação Internacional de Diabetes, que prevê 592 milhões de diabéticos para 2035 para o planeta.

Existem dois tipos de diabetes:

- Tipo I: O sistema imunológico apresenta uma anomalia que produz anticorpos que atacam as células que produzem insulina. Dessa forma, o pâncreas perde completamente a capacidade de produzir o hormônio. Esse tipo de diabetes ocorre em cerca de 10% dos diabéticos e geralmente é hereditária e apresenta os primeiros sintomas já na infância.
- Tipo II: É o mais comum, ocorre em cerca de 90% dos diabéticos. O problema é que a produção de insulina existe, mas a pessoa apresenta resistência aos efeitos da mesma, ou ela pode ser insuficiente para manter um nível de glicose normal. Normalmente acometem pessoas com mais de 40 anos, acima do peso, sedentárias e sem bons hábitos alimentares.

Existem vários sintomas para a diabetes, mas os principais são: aumento da vontade de urinar, sede excessiva, aumento de infecções bacterianas, cansaço e fraqueza.

O tratamento é em grande parte uma reeducação do indivíduo. Sendo os principais tratamentos a atividade físicas e o controle da dieta, evitando a ingestão de açúcares, para quem possui o tipo 2. Para os indivíduos com o tipo 1 é necessário o a aplicação de insulina todos os dias. Aos que possuem a tipo 2 devem fazer um autoexame em casa para saber se precisam tomar a insulina naquele dia.

De acordo com o site Endocrino.org.com.br, no tratamento é recomendável que a glicose fique entre 70 a 100 mg/dL. Abaixo dos 70 mg/dL é considerado hipoglicemia e se ficar maior que 100 mg/dL é considerado hiperglicemia. A medida dL significa 1 litro dividido por 10.

2.2.2 Hipertensão

A hipertensão é uma doença crônica e silenciosa que consiste no aumento da pressão arterial (PA), o que pode causar problemas em vários órgãos como coração, cérebro, olhos e rins. Essa pressão está relacionada com o bombeamento do sangue pelo coração através das artérias.

O coração é o órgão responsável por bombear o sangue através de vasos ou artérias para todo o corpo. Nesse bombeamento existe pressão que deve ser maior que a presente nos vasos, o que permite a circulação do sangue por todo corpo. Contudo, esses vasos podem estar contraídos ou bloqueados, o que eleva a pressão arterial mais normal.

A pressão arterial é medida através de duas pressões:

- Sistólica, conhecida como máxima, é a pressão que o coração precisa para bombear o sangue pelo corpo, ou seja, a pressão que o sangue faz contra os vasos. A medida normal é de 120 mmHg (milímetro de mercúrio). Se a pressão for maior ou igual a 140 mmHg, é considerado hipertensão.
- Diastólica, também conhecida como mínima, é a pressão exercida pelos os vasos ao sangue que está sendo bombeado. A medida considerada normal é de 80mmHg, sendo hipertensão com a medida maior ou igual a 90mmHg.

A pressão alta pode ocorrer em qualquer faixa etária, desde de crianças até em idosos que são as pessoas com maiores incidências dessa doença. De acordo com Sociedade Brasileira De Hipertensão ([Sociedade Brasileira de Hipertensao, 2016](#)) essa enfermidade está presente em 25% da população brasileira adulta e em 50% dos idosos, já nos jovens em 5%. Por estar presente em índices elevados, ela é responsável por 40% dos infartos, 80% dos derrames e 25% de casos de insuficiência renal terminal.

Na Tabela 1 são ilustrados os tipos e as classificações da hipertensão.

Tabela 1 – Hipertensão

Classificação da PA	PA sistólica (mmHg)	PA diastólica (mmHg)
Ótima	<120	<80
Normal	< 130	< 85
Limítrofe	130-139	85-89
Hipertensão estágio 1	140–159	90–99
Hipertensão estágio 2	160-179	100-109
Hipertensão estágio 3	> 180	> 110
Hipertensão sistólica isolada	> 140	< 90

Não existe uma causa específica para essa doença, mas sim um conjunto de fatores que potencializa seu aparecimento. Entre os principais, estão a hereditariedade, idade

avançada, falta de exercícios físicos, má alimentação, sal em excesso, álcool, tabagismo, obesidade e níveis altos de colesterol.

Para o tratamento é indicado uma rotina saudável, evitando os fatores de risco. Sugere-se manter o peso ideal, realizar exercícios diários, abandonar o fumo, não utilizar o sal em excesso, evitar carnes vermelhas, derivados do leite integral, açúcares e doces. Caso o paciente tenha uma pressão com medição superior a 180mmHg é indicado o uso de medicamento que tem a função de dilatar os vasos sanguíneos para diminuir a pressão. Além disso, é sugerido que os portadores façam o acompanhamento regularmente da variação a pressão através do aparelho esfigmomanômetro para o controle.

2.3 Computação em nuvem

A Internet cada dia mais torna-se uma ferramenta essencial na vida dos usuários e principalmente nas empresas. Com o aumento de dispositivos móveis e internet sem fio, a população fica conectada em todos os lugares, e não mais somente em casa ou no trabalho.

Estamos acostumados a armazenar todo tipo de arquivo e a utilizar aplicações em nossa estrutura local, isto é, instaladas em nossos próprios computadores ou dispositivos. A principal vantagem dessa forma de armazenamento ou uso de aplicações, é na maioria das vezes ter acesso a esses recursos de maneira *off-line*.

Por outro lado, a não ser que haja compartilhamento de rede, todos os dados gerados ficam restritos a um único equipamento. E mesmo que haja uma rede compartilhada, como é o caso de ambientes corporativos, a prática on premise, de possuir infraestrutura no local, pode gerar algumas limitações, como a necessidade de possuir uma licença de determinado *software* para cada computador, por exemplo.

Nos tempos atuais, com a constante evolução da tecnologia computacional e das telecomunicações, o acesso à internet está cada vez mais amplo e rápido, o que propiciou a popularização desse novo conceito: *Cloud Computing*.

Computação em nuvem é uma tendência recente de tecnologia que tem por objetivo proporcionar serviços de tecnologia da Informação sob demanda com pagamento baseado no uso.

Conhecida no Brasil como computação em nuvem, a *cloud computing* refere-se basicamente à ideia de acessar e utilizar qualquer aplicação por meio da internet. Sejam eles editores de texto, jogos, etc. Tudo isso como se tivessem instalados em computadores locais.

A computação em nuvem é dividida em três modelos de serviços. São eles:

Infraestrutura como serviço(IaaS) que permite a contratação servidores virtuais (e outros dispositivos de infraestrutura) ao invés de comprar servidores, roteadores, dentre

outros *hardwares*;

Software como Serviço(SaaS) tem como sua principal função entregar a aplicação ao cliente final. É um modelo onde você utiliza algum *software* e paga por sua utilização, não sendo necessária a compra de licenças.

Plataforma como Serviço(PaaS) é o modelo intermediário, entre o SaaS e o IaaS, nele é oferecido um ambiente para o desenvolvedor criar e ou executar o projeto, utilizando os *softwares* e a infraestrutura necessária e adequada à aplicação desenvolvida (OLIVEIRA et al., 2014; ANTONIORICARDO, 2013).

Neste trabalho foram utilizados os conceitos de PaaS e SaaS. O PaaS foi empregado para o desenvolvimento, teste e produção com o sistema sendo hospedado em um servidor nuvem que poderia ser escalado, aumentando ou diminuindo as suas funcionalidades. Para o futuro cliente, o sistema funcionará com SaaS, com a possibilidade de ser acessado de qualquer dispositivo com acesso à internet.

2.3.1 Vantagens da Computação em Nuvem

Uma das maiores vantagens da computação em nuvem é não ter que investir substancialmente em centro de processamento de dados e servidores potentes, uma vez que tudo é executado em servidores remotos. Além disso, não é necessário ao usuário conhecer toda a estrutura que há por trás, como as configurações de *hardware*, como o escalonamento é feito, quantos servidores existem, etc.

Muitas aplicações em cloud computing são gratuitas, e quando é necessário pagar, o pagamento é proporcional aos recursos ou tempo utilizados.

Outro ponto positivo é poder acessar dados, arquivos e aplicativos a partir de qualquer lugar que tenha acesso à internet. Porém, um dos pontos negativos da computação em nuvem é a desconfiança gerada pelo fato de manter em um ambiente virtual informações importantes para as empresas. A segurança dessas informações é um ponto de dúvida entre alguns usuários ou potenciais usuários.

Ao utilizar a computação em nuvem é possível conseguir um custo variável mais baixo devido à economia de escala alcançada com o uso compartilhado de determinada aplicação por centenas de milhares de clientes. Na Tabela 2 temos a ilustração do panorama de uma loja *web* que exemplifica essa característica.

O número de visitas ao site aumentou cinco vezes, de 100 mil para 500 mil. Como esse sistema está na nuvem, através da escalabilidade, a quantidade de servidores foi aumentada de 1 para 4. Nesse cenário, era de se esperar que o custo sobre essa operação fosse alto, mas foi de 395 dólares. Apesar do aumento do fluxo de visitas, essa situação trouxe uma redução no valor do custo por visita ao site.

Anteriormente, a cada 100.000 visitas, o custo era de U\$142.37, o que leva ao valor de 0,0014237 centavos de dólar por visita. Com o cenário de 500.000 acessos o custo foi para U\$ 538.09 o que equivale a 0.00107618 centavos de dólar por visita.

Tabela 2 – Loja hospedada na nuvem: Fonte(Amazon Web Services Ação)

Serviço	Uso em janeiro	Uso em fevereiro	Cobrança de fevereiro (US\$)	Aumento (US\$)
Visitas ao site	100.000	500.000		
CDN	CDN 26 mil solicitações + 25 GB de tráfego	131 mil solicitações + 125 GB de tráfego	113,31	90,64
Arquivos estáticos	Usados 50 GB de armazenamento	Usados 50 GB de armazenamento	1,50	0,00
Balanceador de carga	748 horas + 50 GB de tráfego	748 horas + 250 GB de tráfego	20,30	1,60
Servidores web	1 servidor = 748 horas	4 servidores = 2.992 horas	204,96	153,72
Banco de dados (748 horas)	Servidor pequeno + armazenamento de 20 GB	Servidor grande + armazenamento de 20 GB	170,66	128,10
Tráfego (tráfego enviado para a internet)	51 GB	255 GB	22,86	18,46
DNS	2 mil solicitações	10 mil solicitações	4,50	3,20
Custo total			538,09	395,72

2.3.2 Acesso aos dados

Os dados são acessados muitas vezes através de páginas *web*. Uma página da *web* é formada pela junção de três tecnologias que fornecem estrutura, estilo e interação. O cliente ao requisitar uma página é enviado um endereço(URL) ao servidor. Quando a requisição chega ao servidor, ela é tratada através do sistema de *urls* que utiliza expressões regulares para encontrar o padrão e retorna a resposta da solicitação.

Essas tecnologias da *web* serão descritas detalhadamente nas seções a seguir:

2.3.2.1 HTML

As páginas da *web* são estruturadas através do *Hypertext Markup Language* (Linguagem de Marcação de Hipertexto). A criação da página ocorre através de etiquetas que informam ao navegador como ele deve apresentar o conteúdo. Essas etiquetas são conhecidas como *tags*, através das quais é informado se aquela estrutura é um parágrafo

tag ou uma tabela *tag table*. O número de etiquetas é grande, existindo cerca de 116 representações de estruturas na versão 5.

As recomendações da linguagem são desenvolvidas pela W3C e suas licenças são gratuitas. Atualmente está em sua quinta versão, e trouxe importantes *tags* como *nav* e *navbar*, ambas para informar estrutura de navegação. O benefício dessas novas *tags* são a acessibilidade para os sites, uma vez que leitores de site, *crawlers* (robôs que coletam informações de páginas da *web*) ou até mesmo ferramentas de acessibilidade identificam com maior precisão o que aquela estrutura representa.

A utilização do HTML em algumas situações, como o uso de suas novas *tags*, pode ser um problema para os desenvolvedores. A linguagem HTML é desenvolvida a partir das recomendações da W3C, e cabe às empresas como Mozilla, Google e Microsoft implementarem as características em seus navegadores. Dessa forma, algumas *tags* funcionam em alguns navegadores e em outros não.

Na Figura 4 é ilustrado o suporte a *tag Data and Time input types*. Os navegadores em vermelho não tem suporte; Os em amarelo tem suporte parcial; E o em verde tem suporte completo para especificação.

Date and time input types  - LS Global 60.24% + 8.48% = 68.72%

Form field widget to easily allow users to enter a date, time or both, generally by using a calendar/time input widget.

Current aligned Usage relative Show all

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
8								4.3	
9			45					4.4	
10	12	42	46			8.4		4.4.4	
11	13	43	47	9	34	9.2	8	46	47
	14	44	48	9.1	35	9.3			
		45	49		36				
		46	50						

Figura 4 – Suporte a tag Date Time em formulário. Fonte: <http://caniuse.com/feat=input-datetime>

Um exemplo da sintaxe do HTML e do código renderizado no navegador pode ser visualizado na Figura 5.

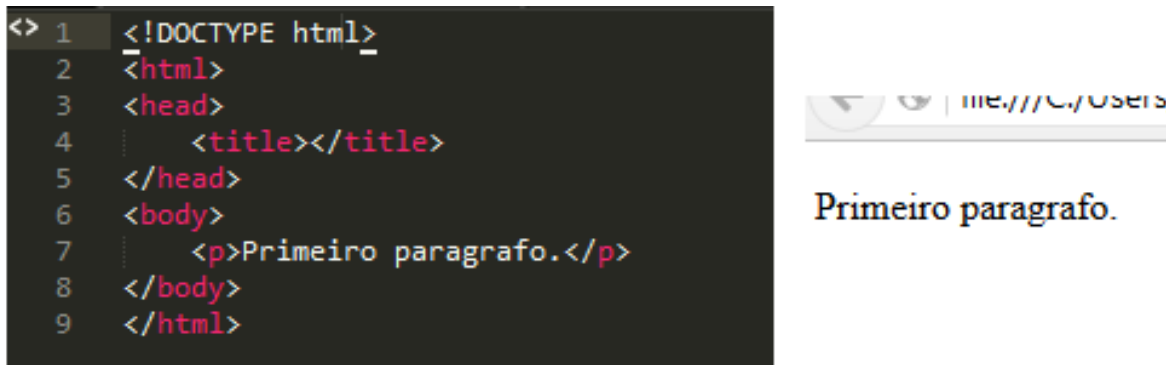


Figura 5 – Sintaxe de HTML a esquerda e código renderizado no navegador a direita.

2.3.2.2 CSS

Uma página *HTML* não tem detalhes, cor ou estilo. Essa transformação pode ser obtida através do uso do *CSS* que é a abreviação para o termo em inglês *Cascading Style Sheet*, traduzido para o português como Folhas de Estilo em Cascata. Essa tecnologia tem a função de descrever como é o elemento, ou seja, define o estilo, como a cor ou tamanho de letra de um elemento numa página web.

A tecnologia *CSS* tem a única e exclusiva função de estilizar a página. Através da estilização pode-se atribuir algum efeito. Um exemplo é uso de atributos, como ao passar o mouse em alguma estrutura realizar a troca de cor da letra.

A especificação do *CSS* é desenvolvida pela W3C e pode ser encontrada na página oficial da especificação no site W3¹. Atualmente ele está em sua terceira versão.

O *CSS* funciona através de seletores. Cada *tag* do *HTML* ou ID utilizado na construção da página são seletores. Além disso, a tecnologia possui artifícios que permitem criar classes que são conjuntos de estilos para atribuir a elementos. Assim adiciona-se um efeito específico às estruturas que possuem a classe.

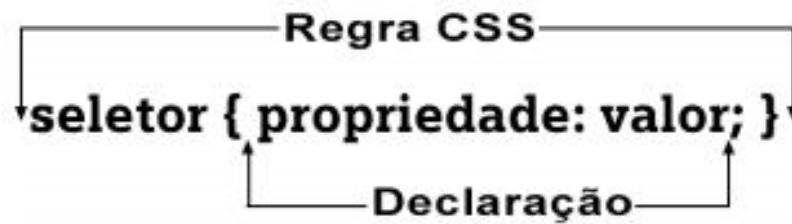
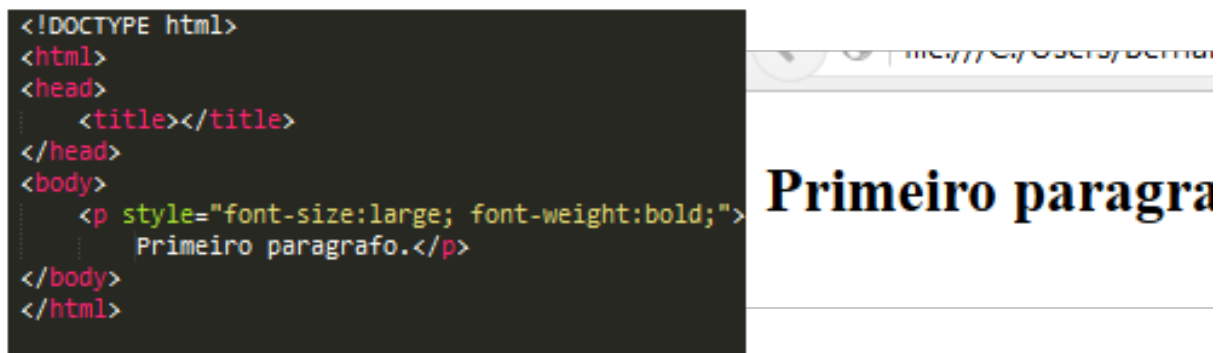
A Figura 6 exemplifica a atribuição de uma regra a um seletor desta tecnologia.

Um exemplo de uso do *HTML* com *CSS* pode ser visualizado na Figura 7.

2.3.2.3 JavaScript

JavaScript (*JS*) é uma linguagem de programação interpretada, criada em 1995 por *Brendan Eich* e atualmente segue o padrão encontrado em <http://www.ecma-international.org/ecma-262/5.1/> > *ECMAScript* 5.1.

¹ <https://www.w3.org/Style/CSS/specs.en.html>

Figura 6 – Regra de *CSS*Figura 7 – Sintaxe de *HTML* com *CSS* a esquerda e código renderizado no navegador a direita.

Por padrão ela funciona somente no navegador e não interage com outras partes do computador do usuário, por exemplo criando arquivos ou executando comandos. A linguagem foi desenvolvida para fornecer interatividade nas páginas da *web*, pois antes eram estáticas, compostas somente por *HTML* e *CSS*.

O JS conta com um interpretador que executa operações em tempo real sem a necessidade de enviar dados para o servidor, executando no próprio navegador. Assim, através do *JavaScript* podemos fazer operações como validações de dados. Essa possibilidade de tratar informação no próprio navegador permite que servidor não se sobrecarregue. Um exemplo é o caso dos formulários, que através do uso da linguagem é possível validar se todos os campos estão preenchidos e com dados corretos, antes de enviar para o servidor.

O código JS a seguir tem a funcionalidade de validar se o *e-mail* digitado em um campo de entrada tem o formato de um *e-mail* válido.

```
function is_email(email){
  er = /^[a-zA-Z0-9][a-zA-Z0-9\._-]+@[a-zA-Z0-9\._-]+\.[a-zA-Z0-9]{2}/;
  if( !er.exec(email) ) return false;
  else return true;
}
```

2.3.2.4 Python

Python é uma linguagem de programação divulgada em 1991 por Guido van Rossum. Apesar dos seus 25 anos, ela acompanha as mudanças da tecnologia e está sempre em desenvolvimento e aperfeiçoamento seguindo algumas filosofias e leis que podem ser acessadas no PEP 20 que está anexo a esse documento. Ela encontra-se em duas versões estáveis, a 2.7.11 e a mais nova 3.5.1.

Essa linguagem possui várias características entre as quais se destacam: a indentação, o que obriga o código a ser sempre organizado e legível; ser interpretada, o que permite a flexibilidade de mudança entre os sistemas operacionais; multiparadigma, o que possibilita o desenvolvimento em várias abordagens diferentes como orientada a objetos, imperativa, funcional e procedural.

Python é bastante utilizado pelas universidades do exterior como por exemplo Universidade *Oxford*, no Reino Unido e Universidade da Califórnia, nos Estados Unidos. Como também em grandes empresas, órgãos e desenvolvedores de *softwares* como *Google INC*, Rede Globo, Nasa, *Washington Post* e *The Guardian*.

2.3.2.5 Framework

Os sistemas computacionais oferecem soluções para problemas que normalmente são parecidos e recorrentes, o que torna a reutilização do código algo bastante útil. Assim surgiram os *frameworks*. Estes são códigos interligados e comuns a determinados problemas. Existem vários *frameworks* usados em diferentes situações, como: *Ionic Framework*, usado para desenvolvimento móvel; *XUnit Framework*, usado em teste de código automatizados ou para games, como o *Phaser*.

Segundo (MATTSSON, 1996), um *framework* é uma arquitetura desenvolvida com o objetivo de atingir a máxima reutilização, representada como um conjunto de classes abstratas e concretas, com grande potencial de especialização. A utilização de *frameworks* no desenvolvimento de aplicações traz benefícios originados de suas características principais: são modulares, reusáveis, extensíveis e eventualmente assumem o controle da execução invocando métodos da aplicação quando necessário (inversão de controle).

A diferença entre bibliotecas e *frameworks* é que as funções do *framework* são interligadas e nas bibliotecas não há interligação. A Figura 8 ilustra os componentes que formam o *DjangoFramework*, que foi utilizado para o desenvolvimento deste projeto. Nela é possível visualizar que o *Django* é formado e depende dos componentes *Settings*(Configurações), *Templates*, *RootURLS*(Links principais), *Database*(Banco de dados) E *APPS*(Módulos), os quais dependem de outro sub-nível de componentes que são *Models*(Modelos), *Views*(Controlador), *Templates*(Telas). O conjunto desses componentes interagindo entre si formam o *framework*.



Figura 8 – Fonte: Eduardo Palma - Framework para desenvolvimento web.

Das vantagens que um *framework* nos fornece, temos:

- Padronização
- Manutenção
- Reutilização do código
- Qualidade

Apesar de todas as vantagens em utilizar *framework* para desenvolvimento, existem também algumas dificuldades, que são:

- Curva de aprendizado
- Requisitos e dependências do sistema

No desenvolvimento *web* a utilização de *framework* tornou-se uma boa prática, uma vez que leva à padronização e organização do código. Como a maioria dos projetos são desenvolvidos por grupos de pessoas que muitas vezes trabalham em funcionalidades diferentes, a utilização do *framework* permite reunir os códigos de equipes distintas no mesmo projeto.

2.3.2.6 Django

No desenvolvimento deste trabalho de conclusão de curso, foi necessária a utilização de um *framework* robusto para realizar operações de recebimento do *XML* e exibição e manipulação da informação. Após um estudo dos melhores *frameworks* gratuitos do mercado voltados para o desenvolvimento da metodologia ágil, os que se destacaram foram *Symphony*, da linguagem *PHP* que é mantido pela empresa SensioLabsNetwork; *Ruby On Rails*, da linguagem *Ruby* que é desenvolvido pela equipe *Rails Core Team*; e *DjangoFramework* da linguagem *Python* com a empresa *Django Software Foundation*.

Das opções citadas acima, a escolha foi o *Django*, pois é desenvolvido na linguagem *Python*, sendo assim, pode ser executado em qualquer servidor tanto *Linux* ou *Windows*. A escolha por esse *framework* se deu por utilizar tecnologias ágeis com várias funcionalidades implementadas como sessão de usuário, *logs*, *URLS* amigáveis, *REST*, *template engine*, sistema de configuração avançado, sistema de testes e por adotara tecnologia *MVC*. Além de todas essas características positivas ele é desenvolvido modularmente. Sendo assim, um módulo não interfere no outro, com futuras implementações nos sistemas livre de incompatibilidades com as atuais existentes.

As três principais áreas do *Django* são os *Models*, no qual estão as classes que são convertidas em tabelas de bancos de dados; as *Views*, onde são recebidas as solicitações do usuário, essas são processadas e retornam em forma de página *html* ou *json*. Em outros *frameworks* essas views são chamados de *controllers*.; *Templates* são as páginas *html* que são processadas por um *template engine* que realiza algumas operações como processar uma lista de dados em *python* convertendo para uma lista em *html*. Essas três áreas são geralmente conhecidas como MVC, mas no Django são *MTV*.

Esse *framework* segue o princípio *DRY* ("*Don't repeat yourself*"), ou "Não se repita". Assim, ele convencionou que o usuário não deve reescrever o mesmo código duas vezes. Um exemplo dessa convenção seria como descrito no *models* anteriormente, o *framework* cria para o usuário as tabelas no banco de dados através das classes escritas o que reduz código de criação de *SQL*.

3 Engenharia de Software

O ponto chave de toda criação é o planejamento. Desenvolver um projeto que necessita de meses ou até anos para ser finalizado e durante o processo descobrir que ele não atende ao usuário, ou possui problemas na estrutura, são situações que nenhum desenvolvedor deseja passar.

Para melhorar o gerenciamento de um projeto e diminuir a possibilidade de erros, entra em cena a Engenharia de Software. Ela está ligada ao desenvolvimento de sistemas complexos dentro do prazo, com alta qualidade, e custos viáveis. Aspectos muito importantes para o sucesso de um projeto.

Fritz Bauer, citado no livro de (PRESSMAN, 2011), definiu o conceito de Engenharia de Software de forma bastante clara: "Engenharia de *Software* é o estabelecimento e o emprego de sólidos princípios de engenharia de modo a obter *software* de maneira econômica, que seja confiável e funcione de forma eficiente em máquinas reais".

De acordo com o IEEE (Instituto de Engenheiros Eletricistas e Eletrônicos) "Engenharia de Software é: (1) A aplicação de uma abordagem sistemática, disciplinada e quantificável no desenvolvimento, na operação e na manutenção de *software*; isto é, a aplicação de engenharia de *software*. (2) O estudo de abordagens como definido em (1)".

Como todo produto industrial, o produto de *software* tem seu ciclo de vida:

- ele é projetado para atender a uma necessidade;
- é especificado, quando essas necessidades são traduzidas em requisitos viáveis;
- é desenvolvido, transformando-se em um conjunto formado por código, modelos, documentos e dados;
- passa por procedimentos de aceitação e é entregue ao cliente;
- entra em operação e passa por manutenções quando necessário;
- é retirado de operação ao final de sua vida útil.

É interessante observar que a codificação, que representa a escrita final de um programa em forma inteligível para um computador, é apenas uma pequena parte do ciclo de vida.

Conforme falamos, o sucesso de um *software* envolve diversas tarefas como: comunicação, análise de requisitos, modelagem de projeto, construção de *software*, testes e suporte.

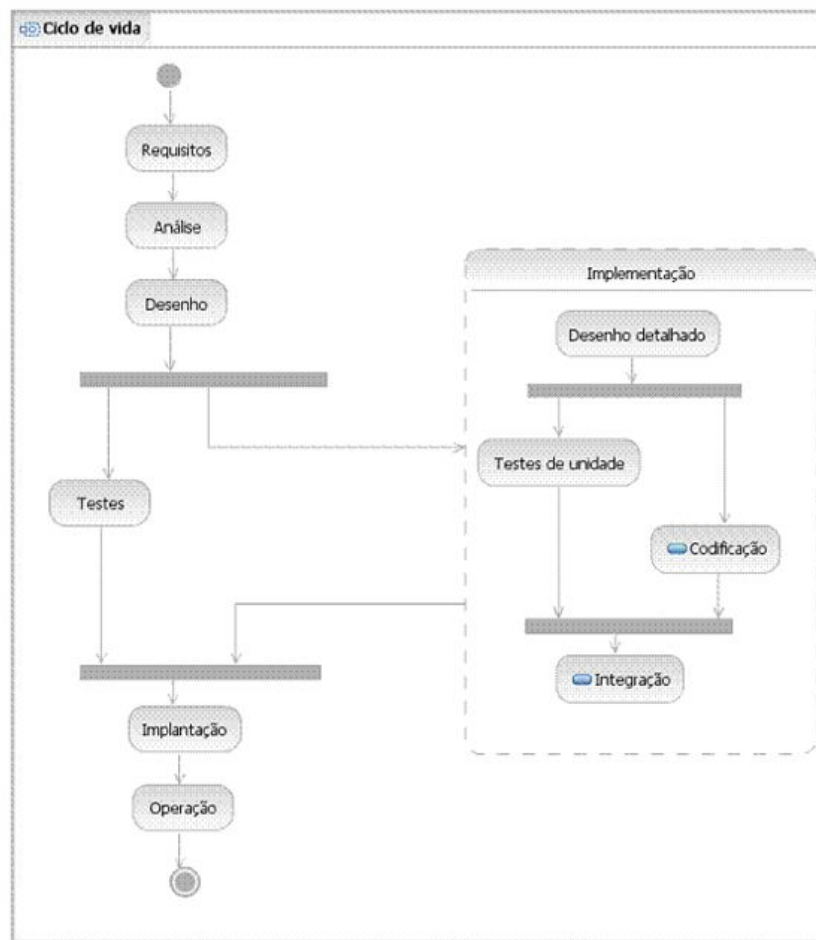


Figura 9 – Modelo UML do ciclo de vida do *software*

O projeto descrito nesse trabalho utilizou a área da engenharia de *software* que chama análise de requisitos em que foram utilizados os casos de usos para apoiar o desenvolvimento. Após essa análise foi realizada a modelagem de bancos de dados levando em consideração as estruturas de dados identificadas nos casos de usos detalhados. Nas sessões seguintes serão explanados com detalhes as áreas utilizadas.

3.1 Análise de Requisitos

A análise de requisitos é o processo de levantamento das necessidades do cliente. Nessa etapa é feita o reconhecimento, a avaliação e a especificação do problema. A coleta de informações geralmente é realizada através das seguintes técnicas:

- Entrevista, etapa na qual os analistas conversam com o cliente;
- Questionário, nessa técnica várias perguntas são respondidas podendo ser abertas ou fechadas. Esse método geralmente é utilizado quando as pessoas têm muito conhecimento sobre o problema;

- Observação, nessa técnica o analista de requisito vai in loco observar as situações dia-a-dia do lugar;
- *Brainstorming*, que são reuniões realizadas com todo o nível de usuário do futuro projeto. Dessa forma surgem várias ideias e pontos de vistas, são todos ouvidos e depois são escolhidas as melhores ideias para o desenvolvimento;

Nesse projeto, a análise foi baseada no TCC Cupom Fácil: sistema de envio de cupons fiscais para dispositivos móveis e nas reuniões *Brainstormings* com os orientadores.

3.2 Casos de Uso

Após a conclusão da análise, iniciou-se uma nova fase do projeto chamada Casos de Uso, na qual desenvolveu-se diagramas UML e histórias de utilização do projeto. (PRESSMAN, 2011) define essa etapa como história estilizada sobre como um usuário final interage com o sistema sob um conjunto de circunstâncias específicas.

Um caso de uso é um recurso para documentar com o máximo de detalhes uma funcionalidade do sistema. Na documentação é descrito quem irá utilizar a funcionalidade, que são os atores; fluxos para utilização; pré-condições e o qual será o resultado final da funcionalidade.

Essa etapa foi dividida em duas partes: o diagrama, responsável por representar as funções que cada usuário pode utilizar no sistema; e parte dos casos de uso, que descreve cada uma dessas funções. Assim, ao final desta fase obtivemos uma visão prévia de como seria o sistema.

3.2.1 Diagrama de Caso de Uso

Esses diagramas fornecem uma ilustração sobre os atores e as funcionalidades principais do sistema. Os atores são os usuários que utilizaram o sistema e existem dois:

- O ator Usuário: é um usuário cadastrado no sistema que após efetuar o login no site poderá utilizar todas as funcionalidades.
- O ator Estabelecimento: é um estabelecimento cadastrado que possui o sistema de envio de cupom fiscal para o sistema.

Geralmente um diagrama engloba todo o sistema, porém, esse foi projetado para ser modular. Foi criado um caso de uso para cada módulo, pois um módulo não tem interação com o outro.

A Figura 10 representa o diagrama do módulo fiscal que apresenta as funcionalidades de cada ator. Nela é possível visualizar que o estabelecimento tem a responsabilidade de enviar cupom fiscal para o sistema. E o ator usuário tem as funcionalidades de realizar buscas, visualizar detalhes de cupons e gráficos baseado nos documentos fiscais. Já a Figura 11 representa o módulo de saúde que possui 6 funcionalidades separadas entre sistema web e aplicativo *mobile*. Nesse módulo existe somente um ator que é o usuário que na web tem a sua disposição as funcionalidade de realizar buscas e gerar relatórios. No aplicativo é permitido o envio de dados e visualizar os biodados recentes.

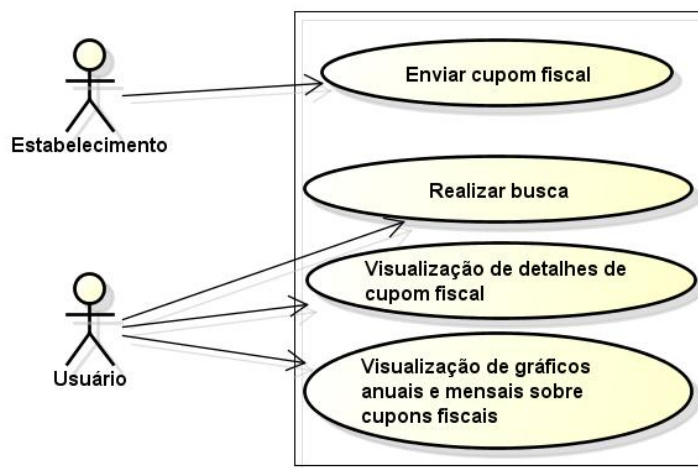


Figura 10 – Módulo Cupom Fiscal

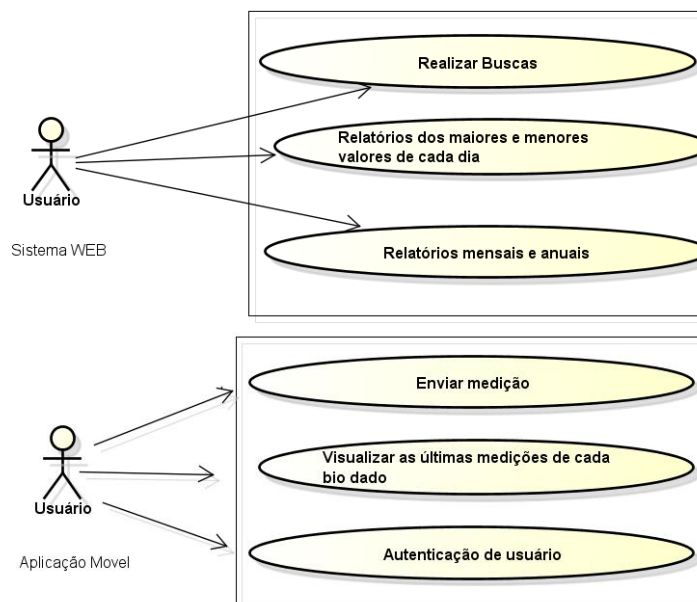


Figura 11 – Módulo Saúde

3.2.2 Caso de Uso Detalhado

Nos diagramas das Figuras 10 e 11 são observados os atores e as funcionalidades, mas não existe uma descrição de como todo o processo deve funcionar. Porém, existem os casos de usos detalhados a seguir, através dos quais serão apresentadas as informações que descrevem as funcionalidades.

Não existe um padrão único a ser seguido no desenvolvimento de casos de uso. Nesse projeto foi escolhido o padrão que contém nome, descrição, atores e fluxo básico. E alguns itens opcionais como pré-condições, fluxo alternativo, estruturas de dados.

Segue um caso de uso que ilustra o cenário em que o cliente realiza um cadastro no sistema.

Tabela 3 – Caso de uso detalhado: Registrar Usuário

Nome	Registrar Usuário
Descrição Sucinta	Usuário registra-se no sistema
Atores	Usuário Visitante
Pré-Condições	—
Fluxo Básico	1. Usuário entra no site. 2. Na tela inicial do sistema existe formulários de cadastro e usuário de login. 3. Usuário preenche todos os campos do formulário de cadastro. 4. Usuário clica no botão registrar. 5. Sistema redireciona para a página de login. 6. Caso de uso encerrado.
Fluxo Alternativo	1. Alternativa ao passo 5. • Sistema não realiza o cadastro pois o usuário informado já existe. • Sistema retorna ao passo 3 com alertas do erro 2. Alternativa ao passo 5. • Sistema não realiza o cadastro pois as duas senhas não são idênticas. • Sistema retorna ao passo 3 com alertas do erro.
Estrutura de Dados	1. Registro. • Nome do usuário • Senha • E-mail • Tipo do usuário (Físico ou Jurídico) • CPF ou CNPJ

No final do trabalho, existem outros casos de uso em anexo.

3.3 Banco de Dados

A modelagem do banco de dados é uma parte crucial para esse projeto, pois os dados enviados pelos usuários serão armazenados no sistema. Com todos os requisitos coletados e os casos de uso escritos, iniciou-se a modelagem, tendo essa uma abordagem clássica com banco de dados relacional.

3.3.1 Modelagem Conceitual

A primeira modelagem apresentada na Figura 12 foi conceitual que de acordo com Carlos Alberto Heuser é uma descrição do banco de forma independente de implementação do sistema gerenciador de banco de dados. Normalmente essa abordagem é utilizada em demonstrações com poucos detalhes para clientes com alto nível de abstração. O cliente consegue visualizar as entidades em que serão salvos seus dados, mas não em quais colunas (HEUSER, 2008).

A modelagem realizada consiste em um diagrama de entidade relacional que ilustra os aspectos do sistema. Apesar do projeto ser modular, a diagramação foi única, optando por não separá-la em módulos.

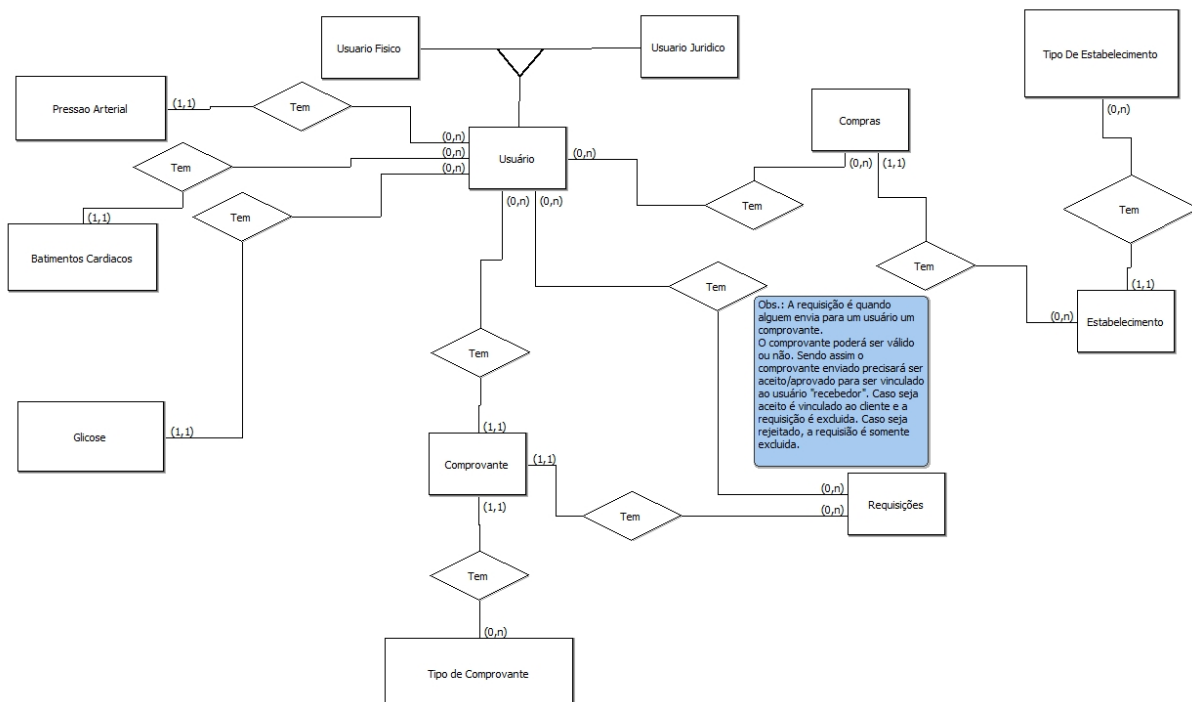


Figura 12 – Modelagem Conceitual

3.3.2 Modelagem Lógica

Podem observar na Figura 13 a modelagem lógica do projeto, que além das tabelas e relacionamentos, contém os atributos com seus tipos de dados definidos. Dessa forma o banco de dados é ilustrado de maneira mais detalhada.

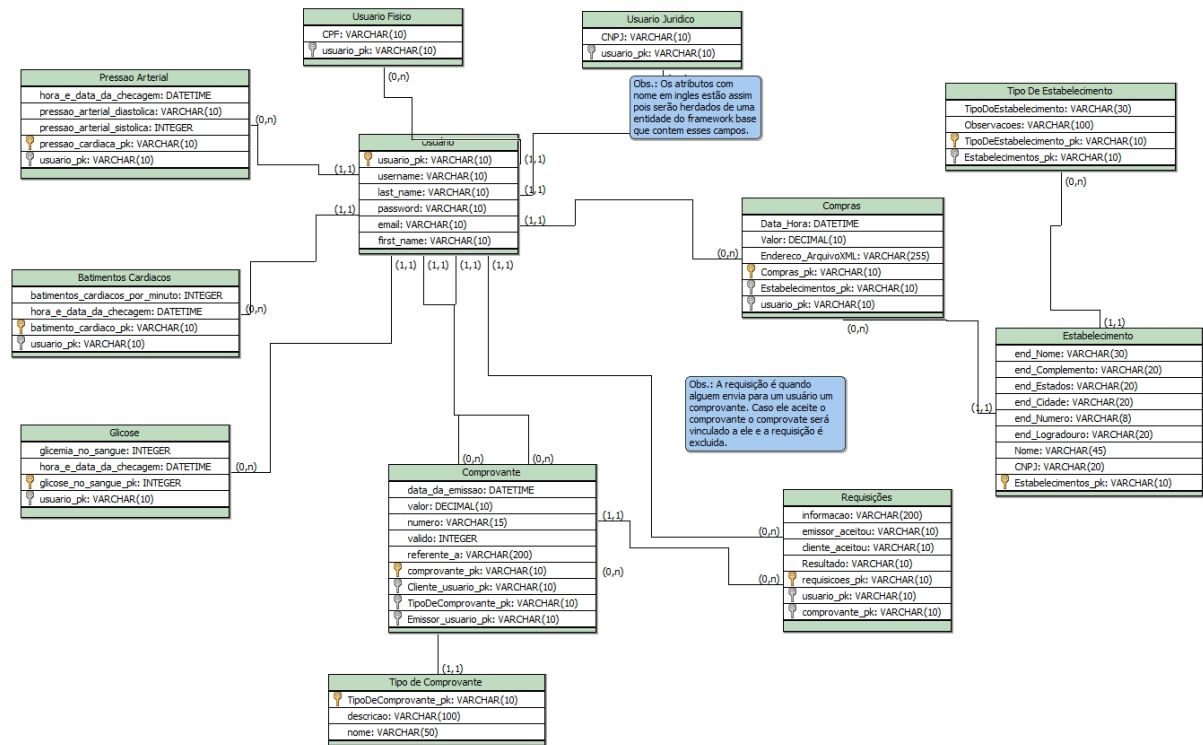


Figura 13 – Modelagem Lógica

4 Desenvolvimento

O presente trabalho demonstra a utilização do armazenamento de informações na nuvem em dois módulos desenvolvidos: o Módulo A, que gerencia cupons fiscais; e o Módulo B, que controla informações de saúde. Esses módulos fazem parte de um sistema que está hospedado em um servidor *cloud*. Uma premissa desse sistema é que as informações sempre deverão estar disponíveis para os usuários.

Nas seções a seguir serão descritos o contexto de utilização, as tecnologias e técnicas utilizadas no desenvolvimento dos módulos. Na seção 4.1 abordará o módulo de cupons fiscais, a seção 4.2 abordará a saúde, na seção 4.3 será explanado sobre arquiteturas e na seção 4.4 será explicado sobre as ferramentas.

4.1 Cupons Fiscais - Dispositivos em estabelecimentos

Mesmo com todo seu benefício, o cupom fiscal é um documento ao qual, em geral, a população não dá muita importância. Em situações cotidianas, muitas vezes ele não é retirado do estabelecimento, é perdido e até mesmo descartado. O motivo desse descaso é a grande quantidade de cupons que o consumidor recebe, já que para toda venda é gerado um documento.

Propondo melhorias para esse cenário, a aluna Jéssica de Freitas realizou um trabalho de conclusão de curso para otimizar o modo de armazenamento dos cupons fiscais.

A proposta consistia em que no momento de emissão do cupom, o cliente pudesse solicitar que o cupom fiscal fosse enviado para o seu celular via transmissão *bluetooth*. Essa prática reduziria o gasto de papel pelo estabelecimento, e evitaria que o cliente guardasse inúmeros comprovantes de compras realizadas.

O projeto foi desenvolvido para o sistema operacional *Android* e o resultado foi um completo aplicativo móvel para armazenamento dos documentos, e outras funcionalidades, como o controle dos gastos através de gráficos.

Após a conclusão desse projeto, notou-se uma necessidade em relação ao armazenamento dos cupons, pois esses eram armazenados somente no celular do cliente, ficando assim suscetíveis a fatores como roubo do celular ou ocupar todo o espaço disponível no aparelho.

Como solução para essa necessidade, iniciou-se o desenvolvimento do módulo de Cupons Fiscais nas nuvens, cujo objetivo principal é a recepção, armazenamento, manipulação e exibição das informações para os usuários. Ele foi desenvolvido para

solucionar esse problema do armazenamento dos documentos, deixando-os disponíveis para todos dispositivos que possuem internet e possam conectar à nuvem do Documento Fácil.

O desenvolvimento deste módulo começou através da proposta do trabalho Cupom Fácil: Sistema de envio de cupons fiscais para dispositivos móveis onde é utilizado as *DLL's* do fabricante do emissor ECF para obter as informações dos cupons emitidos. A partir desse momento criou-se uma bifurcação, pois, foi necessário o desenvolvimento de um meio para enviar as informações para os servidores.

No trabalho base o envio era através da tecnológica *Bluetooth* do computador de frente de vendas para celulares. Neste presente trabalho desenvolvido a escolha foi o desenvolvimento de um novo código que realizasse uma requisição do tipo *POST* do computador de frente de vendas para o servidor enviando um arquivo em formato *XML* com os detalhes das compras.

No lado servidor, foi desenvolvido um método com a finalidade de receber e manipular a requisição do cliente. O tratamento consiste em obter os dados em forma de *bytes*, transformá-los em um arquivo *XML* e armazená-lo em pastas do sistema divididas por ano, mês e dia para facilitar futuras buscas.

Durante o desenvolvimento foi discutido se todos os dados presentes no *XML* seriam inseridos no banco de dados. Foi convencionado que somente algumas informações como CPF do cliente, valor total da compra e o caminho para o *XML* seriam gravadas no banco de dados. Essa opção foi justificada, pois, o sistema ficaria sobrecarregado ao analisar todos os mínimos detalhes da compra, uma vez que a maioria dos usuários gostaria de ter acesso a relatórios mais gerais. Contudo, não foi descartada a possibilidade da visualização de cupons únicos. Caso o cliente solicitasse os dados, o sistema buscaria pelo caminho armazenado para encontrar e obter as informações para retornar para o cliente.

O banco de dados desse módulo foi projetado para armazenar informações das compras, de estabelecimentos e tipo de estabelecimento. É importante ressaltar que a escolha dessa modelagem permite a adição de novos estabelecimentos e tipo de estabelecimentos ao sistema. Com a inserção sendo facilmente realizada, o sistema não fica travado por limitações técnicas. Toda a lógica é baseada nos dados de tabelas e não em conjuntos estáticos como enumerações previamente definidas.

4.2 Saúde - Móvel

Para algumas doenças é necessário realizar o controle diário de alguns índices, geralmente realizado através de medições. Após a aferição é feita uma análise. Em algumas situações o biodado não é armazenado. Em outras, quando é armazenado, essa rotina é realizada por mecanismos ineficientes. Um exemplo seriam as aferições sendo gravadas

manualmente, o que dificulta o manuseio posteriormente. Para melhorar esse cenário, foi proposto e desenvolvido um módulo no sistema para armazenar esses dados e um aplicativo de celular para o cliente enviar seus dados e acompanhar seu histórico.

Para esse módulo foram escolhidos três índices para o controle que estão presentes em grande escala da população brasileira que são:

- Pressão Arterial, dividida em sistólica e diastólica.
- Glicose.
- Batimentos cardíacos.

O funcionamento desse módulo necessita que o cliente possua o aplicativo instalado em seu celular e através dele envie os dados. O aplicativo foi desenvolvido para ser o mais simples possível, de forma que qualquer pessoa possa usar. A Figura 15 apresenta a tela de login desenvolvida pelo aplicativo.



Figura 14 – App Saúde na nuvem. Tela de login.

Ele é composto pelas telas de lista para exibir os dados enviados e telas para enviar os dados. A utilização do aplicativo é descrita a seguir:

1. Ao abrir, é solicitado uma autenticação que é efetuada através do usuário cadastrado no sistema. Nesse momento é salvo na memória o usuário e senha para posteriormente utilizar em outras requisições.
2. Através de abas na parte inferior, é selecionado o fator que a pessoa deseja utilizar.
3. Ao abrir a tela, é obtida do servidor a lista das últimas 10 medições. Nas telas de lista, existe um botão na parte inferior para enviar os dados.
4. Caso entre na tela de cadastro de dados, é exibido um formulário para o servidor.

Toda a comunicação entre o aplicativo e o servidor é realizada através da tecnologia RESTFUL que utiliza o máximo que a comunicação HTTP oferece. Através de um mesmo endereço é possível realizar todas as operações básicas utilizadas em um banco de dados que são: criar, ler, atualizar e deletar, pois, o REST permite indicar o tipo da requisição através das flags GET, POST, PUT ou PATCH, e DELETE. Essa tecnologia foi escolhida, pois, é segura e todos celulares modernos a possuem. Além disso, outros dispositivos como vestíveis que a utilizam podem futuramente comunicar com o sistema.



Figura 15 – Cenários de comunicação com o servidor.

A segurança das requisições é realizada através das informações no cabeçalho. O aplicativo armazena na memória o usuário e a senha do cliente atualmente logado. Ao comunicar com o sistema essas informações são enviadas. No lado do servidor é feita a checagem se os dados estão corretos, e, caso estejam, é realizada a operação solicitada.

Os dados do módulo saúde não ficam restritos ao aplicativo para a visualização. Eles podem ser consultados através do sistema *online*, onde é possível gerar alguns gráficos e realizar pesquisas por datas.

Os biodados deste módulos são aferidos manualmente pelo usuário. Após esse procedimento, os dados são inseridos nos formulários presentes no aplicativo e enviados para o sistema.

A Figura 15 apresenta uma característica de segurança em dois cenários distintos de comunicação do aplicativo. Em ambos, ao realizar uma requisição é adicionado no header(cabeçalho) tokens para garantir a autenticação.

4.3 Arquiteturas

O projeto é composto por vários elementos que interagem entre si, tanto *software* como *hardware*. Com a junção destes elementos formou-se uma arquitetura única. Arquitetura pode ser definida de acordo com (JAZAYERI; LINDEN, 2000) como conjunto de componentes e seus relacionamentos, que deve satisfazer os requisitos funcionais e não-funcionais do sistema.

A Figura 16 ilustra todos os componentes que são utilizados pelo sistema divididos em 4 camadas.

- *Hardware*: composta pelos componentes físicos, ou seja, equipamentos utilizados no sistema, que são: servidor, computador, impressora fiscal, celular.
- Comunicação: são utilizados dois meios, um deles é a Internet através do protocolo http; e o outro é comunicação via cabo USB/SERIAL que recupera os dados da impressora fiscal.
- *Software*: nessa camada estão alocados os *softwares* desenvolvidos, sendo um o sistema que é responsável por receber e retornar informações, como também registrar e realizar a autenticação dos usuários; o aplicativo móvel, que envia e recebe informações; e o servidor de cupom fácil, que obtêm as informações da impressora fiscal e envia para o servidor na nuvem.
- Dados: são os tipos de armazenamento utilizados no sistema que são: o arquivo XML, que é trafegado no módulo de cupons fiscais; e o banco de dados que está no servidor que armazena as informações.

Para ilustrar o funcionamento dessa arquitetura foi criado o diagrama presente na Figura 17. Através dele visualiza-se o sistema, que como foi dito previamente, é separado em dois módulos que não comunicam entre si mas comunicam com o servidor. Além desses

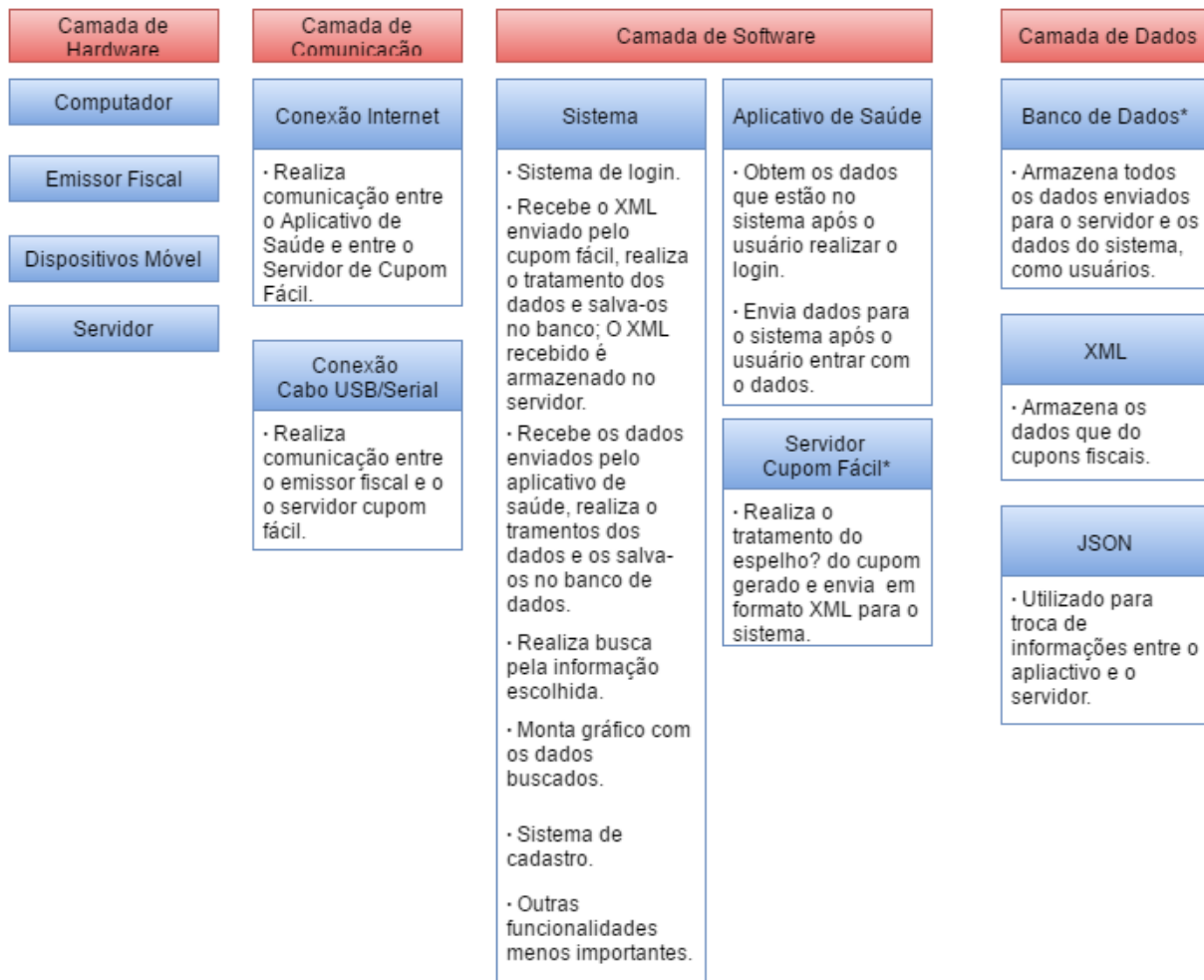


Figura 16 – Arquitetura do sistema

módulos, outro exemplo da utilização do sistema são clientes acessando o portal para a visualização dos dados.

Para facilitar a compreensão do diagrama, a seguir serão explicados os principais passos do sistema:

1. Ao abrir o aplicativo são solicitadas as credenciais do usuário.
2. O usuário efetua o login no sistema. Caso seja válido, é redirecionado para uma tela com seus dados. Nesse momento ele pode escolher entre duas tarefas:

A Enviar novos dados para o servidor.

B Visualizar seus últimos dados.

3. A Caso o usuário escolha enviar dados, é exibida uma tela com formulário, que após ser preenchido é enviado para o servidor. Esse envio é realizado através de

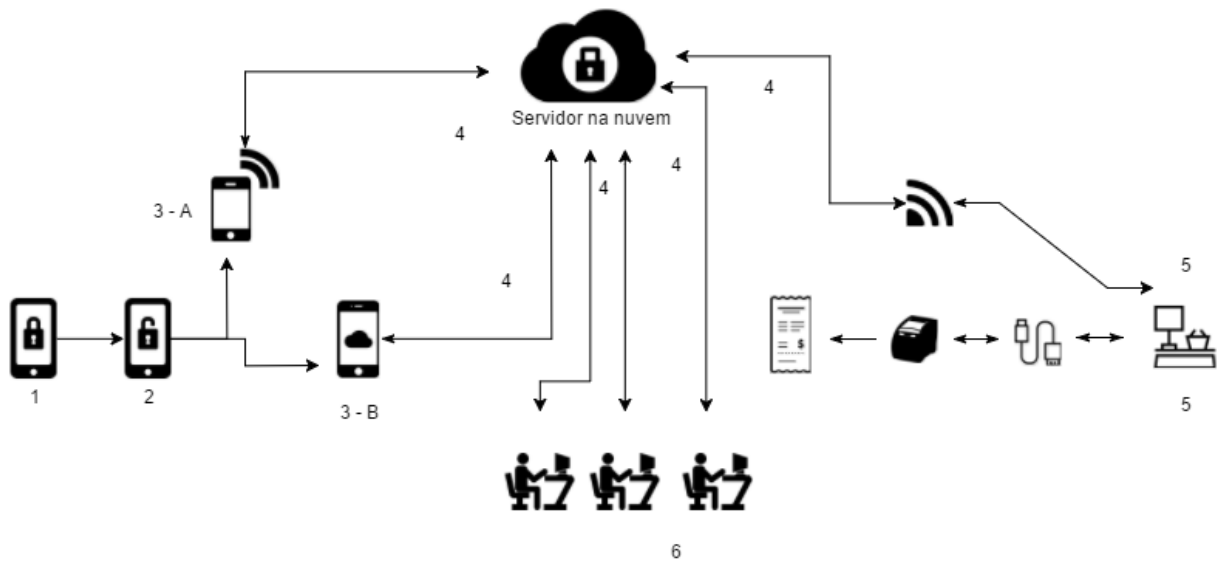


Figura 17 – Funcionamento do sistema

requisição HTTP do tipo POST com cabeçalhos de autenticação preenchidos e os dados inseridos no corpo.

3. B Nessa requisição foi solicitada uma lista com os últimos registros. Nesse caso é retornado uma lista contendo os dados em formato JSON. Toda requisição exige uma resposta.

4. Nesse momento o servidor recebe a requisição, realiza as validações necessárias para não permitir o envio de dados inválidos ou de usuário não credenciado. Após a validação, é realizada a rotina requerida. O *framework Django*, através das rotas e os métodos vinculados a elas, organiza as solicitações (de onde vêm, onde serão processados e para quem retornarão).

5. Nesse momento, o cliente está em algum estabelecimento que possua o servidor fácil finalizando uma compra. O servidor fácil não gera o cupom fiscal, mas sim, sistemas de terceiros presentes na máquina. O cliente solicita que os dados sejam enviados para o sistema. A partir disso, o servidor fácil busca na memória da impressora fiscal os dados do último cupom lançado. Após os encontrar, ele gera um arquivo XML com os dados e o envia para o servidor na nuvem como anexo através de uma requisição HTTP do tipo POST. Para cada estabelecimento que possua o servidor fácil é gerado uma autenticação para validações entre servidor e sistema local.

6. Nesse cenário, usuários entram e navegam no sistema. Vale lembrar que para navegar é necessário possuir usuário e senha. Caso não possua, na página inicial do sistema é exibido um formulário de registro. Durante a navegação é possível

visualizar gráficos e realizar pesquisas por dados com alguns filtros sendo o mais comumente utilizado o por datas.

No sistema não existe a quantidade máximo de usuários logados simultaneamente, isso se deve ao fato do *framework* utilizar uma da tecnologia que pode ser escalável.

Vale ressaltar que para toda requisição realizada ao servidor é feito o que está descrito no passo 4, pois, a segurança das informações e do servidor foi um dos requisitos desse projeto.

4.4 Ferramentas

Equipamentos externos foram utilizados durante o desenvolvimento do projeto, principalmente na etapa de testes. O uso desses equipamentos foi essencial para simular como seria a realidade de uso e o tempo gasto para cada operação, pois, os emuladores não ofereciam métricas exatas.

Um equipamento utilizado foi um celular Moto G da primeira geração da *Motorola Solutions* com o sistema operacional *Android* 4.3. Ele foi utilizado visando testar a comunicação entre o aplicativo de saúde e o sistema. Esses testes foram realizados enviando e recebendo informações.

Outro equipamento utilizado foi uma impressora térmica FS700 de cupons fiscais da empresa Daruma Tecnologia. Nela foram realizados os testes de envio do XML de cupons fiscais para o servidor.

4.4.1 Ambiente

Para desenvolver um sistema é necessário um conjunto de elementos como editor de texto, linguagem de desenvolvimento, sistema operacional, bibliotecas e *frameworks*. Neste trabalho foram utilizados Windows 7 e Linux Mint 13.04 como sistemas operacionais, e o GIT para controle de versões.

Para o desenvolvimento do sistema foi utilizado o Sublime Text 3 como editor de texto, Python3 como linguagem do desenvolvimento e o *framework* DjangoFramework como *backend*.

Para o desenvolvimento do aplicativo foi utilizado o SDK Android 4.1, a linguagem Java e o IDE Eclipse Luna. Além desses itens citados, foram utilizados alguns *frameworks* e bibliotecas adicionais que serão descritas posteriormente.

As tecnologias presentes são de licenças livres não sendo necessário o pagamento pelo uso das mesmas. Além disso, elas não possuem restrições em relação ao sistema operacional, pois são voltadas para serem utilizadas com Python ou para a *web*.

4.4.2 Bibliotecas e Frameworks

Foram utilizadas algumas bibliotecas e *frameworks* para fornecer funcionalidades prontas ao sistema. A utilização desses é justificada pelo fato do foco do trabalho não ser o desenvolvimento das funcionalidades presentes nelas.

- Bootstrap: *framework* utilizado para padronizar a interface do sistema. Ele é composto de componentes HTML, Javascript e CSS. <<http://getbootstrap.com/>>
- Chart.js: biblioteca utilizada para plotar gráficos. Ela é leve e com o estilo visual que lembra o Bootstrap. <<http://www.chartjs.org/>>
- jQuery: biblioteca Javascript para manipulação de elemento do HTML. É utilizado nas páginas do sistema e pelo Bootstrap. <<https://jquery.com/>>
- django-bootstrap3-datetimepicker: biblioteca utilizada para adicionar um componente para escolha de datas com interface do Bootstrap aos formulários de buscas. <<https://github.com/nkuniyiko/django-bootstrap3-datetimepicker>>
- Apache HttpComponents: biblioteca utilizada para enviar dados do servidor do cupom fácil para o sistema através de requisições HTTP. <<https://hc.apache.org/>>
- Django-Rest-Framework: Responsável por tratar as requisições de REST provenientes do app de saúde. <<http://www.django-rest-framework.org/>>

5 Execução dos Sistemas

5.1 Sistema Web

Os usuários utilizam o sistema *web* para visualizar suas informações. O sistema é composto por vários módulos, mas são imperceptíveis perante aos usuários, somente no código que se vê a diferença.

Todas as páginas do sistema, com exceção da tela inicial e a de *login*, são padronizadas seguindo um *layout* formado por uma barra de navegação e o corpo da página dividida em 3 partes sendo uma divisão de 15% 70% e 15%.

As funcionalidades do sistema são divididas em: Armazenar dados; consultar dados; gerar relatórios. Essas serão descritas a seguir.

5.1.1 Login

Com o sistema sendo executado em algum servidor de hospedagem como *Gunicorn* ou *Apache* é possível acessá-lo através de um navegador. Ao acessar a página inicial será exibido dois formulários um de cadastro e o outro de *login*, como ilustrado na Figura. 18

Figura 18 – Tela de *login*

5.1.2 Buscas

Nas seções 5.1.2.1 e 5.1.2.2 serão apresentadas as formula de realizar buscas no sistema online.

5.1.2.1 Cupom Fiscal

Ao efetuar o *login* no sistema, o usuário será redirecionado para a tela de cupons fiscais que é ilustrado na Figura 19. Nessa página, existe uma lista resumida contendo os dez últimos cupons do usuário armazenados no sistema. Cada item da lista é a representação de um cupom que possuirá três informações básicas as quais são: o nome do estabelecimento, o de valor e a data. E um botão que ao ser clicado mostrará informações detalhadas sobre o cupom fiscal.

Além da lista, no lado direito existe um formulário de busca que através das informação inseridas é realizada pesquisa no histórico do usuário por tipo, estabelecimento e intervalos de datas. Assim, o usuário buscará por seus cupons fiscais de forma efetiva como ilustra a Figura 19 mostra uma busca realizada dentro de um intervalo de datas e pelo tipo de estabelecimento Supermercado.

Estabelecimento	Valor	Data	Mais informações?
Merceria 4 M	R\$ 119,10	26/04/2016	Detalhes
Carrefour	R\$ 105,45	18/04/2016	Detalhes
Comercial Franco e Lima	R\$ 113,30	13/04/2016	Detalhes
Comercial Aktem	R\$ 35,60	22/03/2016	Detalhes
Andrade e Campanha	R\$ 759,00	18/03/2016	Detalhes
Merceria e Sacolao California	R\$ 703,90	15/02/2016	Detalhes
Comercial Aktem	R\$ 59,90	12/02/2016	Detalhes
Comercial Franco e Lima	R\$ 757,00	09/02/2016	Detalhes
Merceria 4 M	R\$ 27,75	07/02/2016	Detalhes
Carrefour	R\$ 384,00	03/02/2016	Detalhes

Página 1 de 2. [próximo →](#)

Figura 19 – Página de cupons fiscais

Descrição	Preço Unitário	Quantidade	Preço Total
Laranja	R\$ 1,00	2	R\$ 2,00
Escova	R\$ 5,00	2	R\$ 10,00
Whisky	R\$ 70,00	3	R\$ 210,00
Vodka	R\$ 11,00	5	R\$ 55,00
Pizza	R\$ 5,00	0	R\$ 0,00
Lasanha	R\$ 7,00	4	R\$ 28,00
mamao	R\$ 2,00	5	R\$ 10,00

[Fechar](#)

Figura 20 – Modal(tela) de detalhes de um cupom fiscal.

5.1.2.2 Saúde - Pressão Arterial, Batimentos Cardíacos, Glicose

A área destinada a apresentação das informações sobre saúde engloba pressão arterial, batimentos cardíacos e glicose.

Cada página apresenta uma lista com os 10 últimos registros efetuados no sistema. Nessa lista, seus itens apresentam informações sobre a doença. No caso da pressão arterial é exibido os dados da sistólica e diastólica; No batimento é exibido o próprio e por último na Glicose é apresentado o dado de Glicemia.

Além dessas informações, em todo item é apresentado a data e hora do recebimento da informação pelo servidor. Assim, o usuário poderá acompanhar seus dados várias vezes no mesmo dia.

No lado direito de cada página, existe um formulário que contém campos para escolher datas com intuito de realizar busca dentro daquele intervalo de tempo. Assim, os dados exibidos na lista previamente mencionada, podem ser customizados para um período desejado.

A Figura 21 apresenta uma busca de biodados. Nela é possível visualizar ao lado



Extra Hipermercado

Data: 28/11/2016 Hora: 13:09:17

Valor: R\$ 315,00

Descrição	Preço Unitário	Quantidade	Preço Total
Laranja	R\$ 1,00	2	R\$ 2,00
Escova	R\$ 5,00	2	R\$ 10,00
Whiskey	R\$ 70,00	3	R\$ 210,00
Vodka	R\$ 11,00	5	R\$ 55,00
Pizza	R\$ 5,00	0	R\$ 0,00
Lasanha	R\$ 7,00	4	R\$ 28,00
mamão	R\$ 2,00	5	R\$ 10,00

Fechar

Figura 21 – Tela de busca de biodados

direito um formulário de data e na parte central da imagem uma tabela com os dados encontrados.

5.1.3 Relatórios

Através dos relatórios, o usuário do sistema obterá informações sintetizadas do seus registros.

As telas que compõem essa parte do sistema são padronizadas sendo compostas por gráficos no centro da página e no lado esquerdo um formulário permite buscar informações em intervalos escolhidos. A seguir serão apresentados alguns gráficos.

5.1.3.1 Cupom Fiscal

Nos relatórios de cupom fiscal são apresentados três distintos gráficos para uma visualização detalhada e simples das informações.

O primeiro corresponde aos gastos anuais de um estabelecimento. Através dele o usuário escolherá o tipo de estabelecimento, estabelecimento e o ano para customizar o gráfico. A apresentação dos dados é realizada em forma de barra separado por meses.

O segundo é um gráfico que apresenta a comparação entre os tipos de estabelecimento cadastrados no sistema em um intervalo de tempo. A customização dos dados se dá pela escolha do intervalo de datas em um formulário.

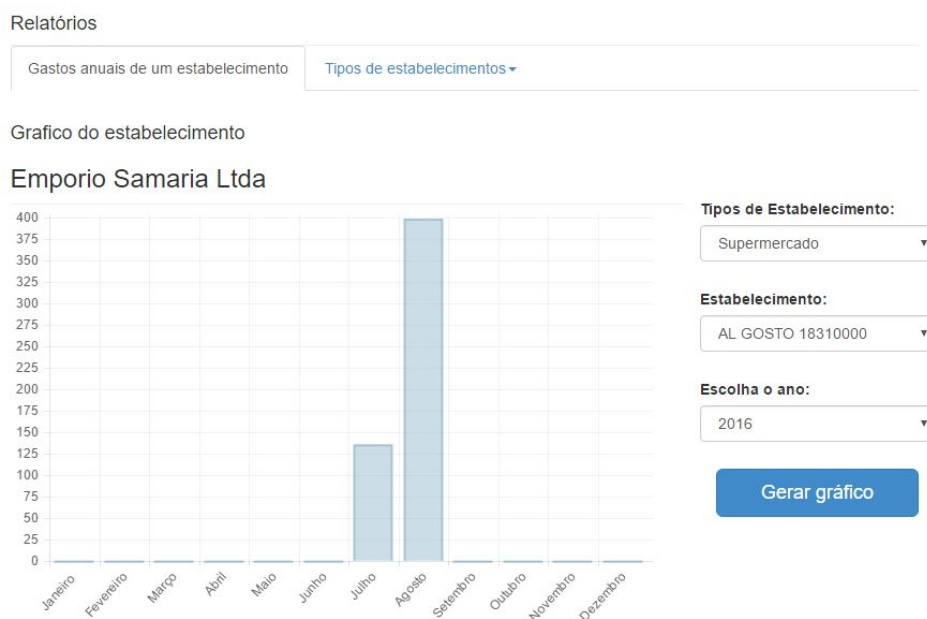


Figura 22 – Gráfico anual por estabelecimento

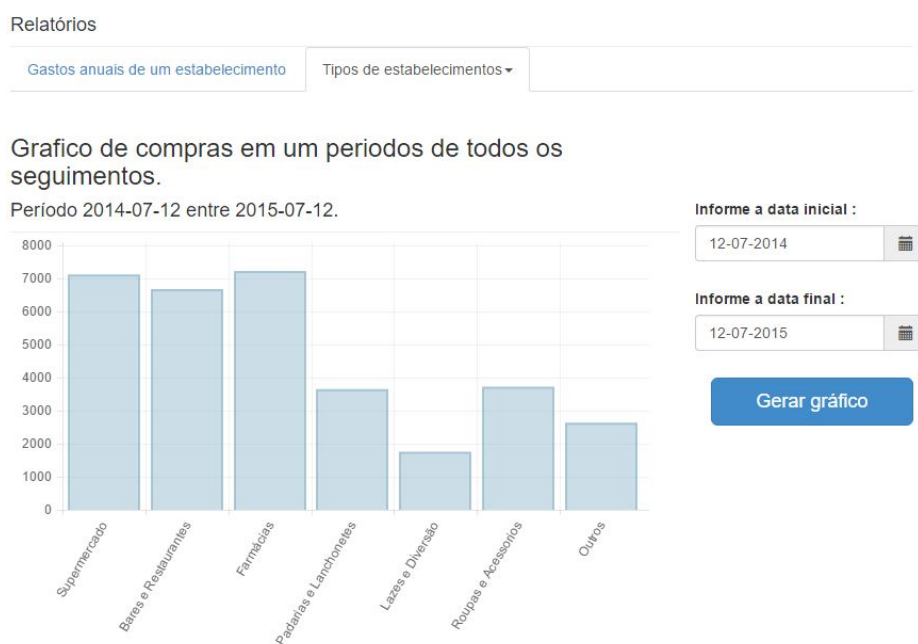


Figura 23 – Gráfico de intervalo selecionado dividido por tipo de estabelecimento

O terceiro apresenta um somatório de valores gastos por mês em um determinado tipo de estabelecimento escolhido pelo usuário. A apresentação do resultado é semelhante ao do primeiro gráfico.



Figura 24 – Gráfico anual por tipo de estabelecimento

5.1.3.2 Saúde - Pressão Arterial, Batimentos Cardíacos, Glicose

Para cada biodado existem dois gráficos para o acompanhamento. O primeiro contém informações mensais sintetizadas com o menor e maior valor do dia do usuário, com isso, é possível ter um acompanhamento de picos do usuário. Podendo identificar dias ou até mesmo intervalo na semana que o usuário não está seguindo as recomendações.

O segundo gráfico é referente ao dia do usuário. Nele é possível observar todas as medições que o usuário realizou no dia pesquisado.

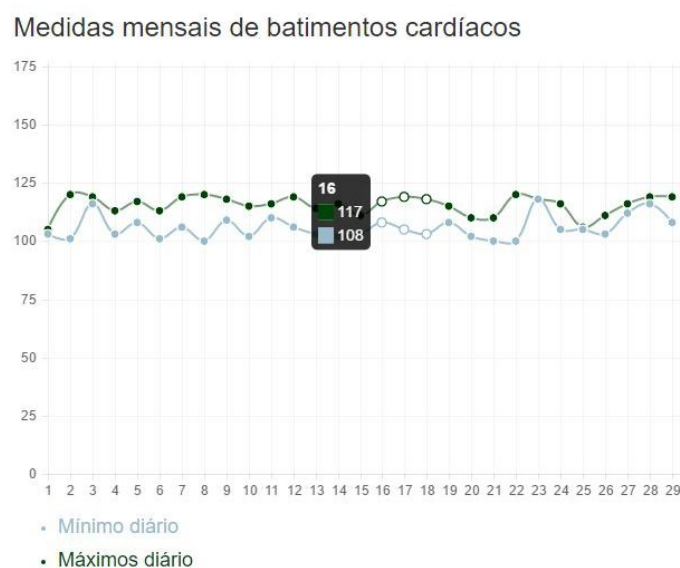


Figura 25 – Página do gráfico mensal de batimentos cardíacos

Os gráficos mensais são em formato de linha contendo uma com os menores valores e uma com os maiores valores. A Figura 25 ilustra um gráfico mensal de batimento cardíacos. Os diários não são diferentes visualmente dos mensais. A diferença é que não mostram duas linhas por gráfico e sim uma.

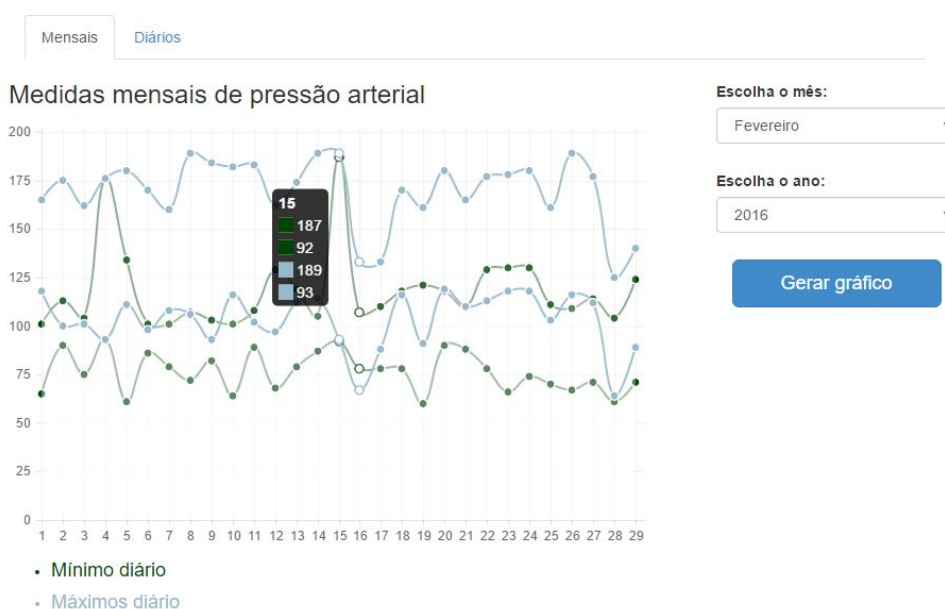


Figura 26 – Página do gráfico mensal de pressão arterial

5.2 Aplicativo Móvel

Para a comunicação entre o usuário e o sistema no contexto de saúde foi escolhido um aplicativo de celular pois a utilização seria rápida e prática.

A partir desta premissa foi desenvolvido um aplicativo simples que possui duas funcionalidades. A primeira e sendo a principal é enviar dados do cliente para o servidor. A segunda é visualizar as últimas medições do biodados enviados para o sistema.

Para o usuário utilizar o aplicativo é necessário que tenha uma conta no sistema *web* deste trabalho. Na primeira tela do aplicativo são solicitadas credenciais válidas. Após o *login* ser efetuado com sucesso, o usuário é direcionado para a tela principal do aplicativo que pode ser observado na Figura 27.

Na tela principal em sua parte superior, existem três opções (*Tabs*) Pressão Arterial, Batimentos Cardíacos e Glicose que são utilizadas para escolher o contexto dos dados que são exibidos.

Acima dessas opções existe um botão com o título de Novo Registro que é utilizado para o usuário enviar dados para o sistema. Para cada contexto de dados é exibida uma janela diferente solicitando os dados pertinentes para o contexto. A janela para pressão arterial pode ser visualizada na Figura 28.

Caso o usuário deseje enviar os dados para o servidor basta entrar com os dados e clicar no botão gravar.

É válido ressaltar que a aferição dos biodados é realizada pelo usuário com suas forma usuais. O aplicativo não realiza a medição, sendo sua principal função armazenar e enviar os dados para o servidor.

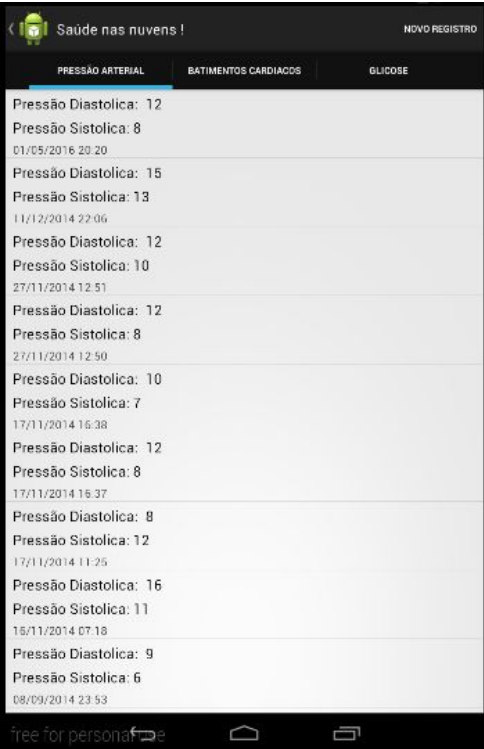


Figura 27 – Tela principal mobile

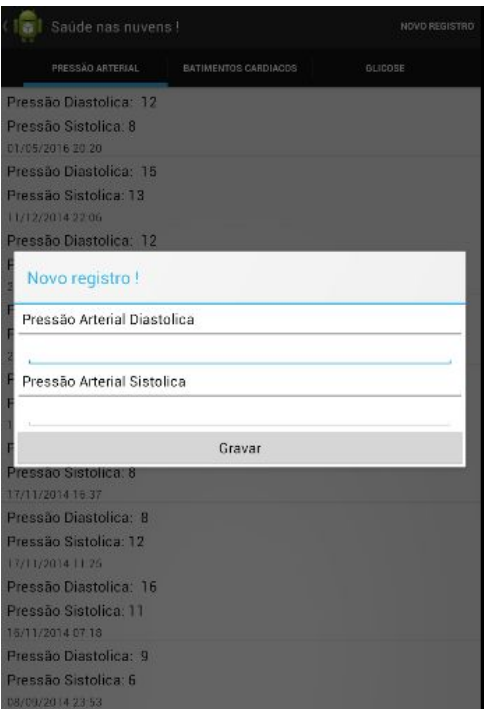


Figura 28 – Novo registro de pressão arterial

6 Conclusão

O presente trabalho apresenta um sistema *online* na nuvem que armazena dados de usuários sobre diferentes contextos como fiscal e saúde. Esse sistema foi projetado de forma modular, o que possibilita futuras adições de módulos sem comprometer os atuais. Além disso, utiliza tecnologia para ser executado em servidores na nuvem o que lhe dá a possibilidade de ser escalável. Outra característica importante do projeto é que o *framework*, que é base do sistema, tem tecnologia de acesso a múltiplos banco de dados, assim, em certo momento quando a base for gigantesca será possível dividir os bancos por áreas e sub áreas. Além dessas características, o sistema comunica com diferentes *hardwares* para obtenção de dados.

Para o desenvolvimento do trabalho, foi necessária a criação de dois projetos em diferentes áreas da computação. O primeiro trata-se de um sistema *web* que utiliza o *framework Django* como base, e possui área de autenticação e formulário para cadastro de novos usuários. Usuários não autenticados tem acesso somente à página inicial. Após realizar a autenticação, o usuário tem permissão para visualizar os dados que o mesmo enviou para o sistema, que são dados fiscais e de saúde. Uma vez inseridos os dados no sistema, ele é capaz de analisar informações e gerar relatórios.

O segundo sistema desenvolvido foi um aplicativo para dispositivos *Android*. Esse sistema é utilizado apenas no módulo de saúde e sua principal função é enviar biodados através de comunicação *HTTP* com tecnologia *RESTFUL*. Além do envio dos dados, o *app* permite a visualização de forma rápida das últimas 10 aferições de biodados.

Por último, a criação de algumas novas funções no sistema base para a coleta de dados das impressoras fiscais e envio para o sistema *web*. Essa parte do projeto é utilizada somente no módulo fiscal.

A utilização deste sistema apoia o usuário no controle de suas informações, tanto com dados de saúde, como dados de compras. O módulo de saúde, permite ao usuário acompanhar seus registros, e em eventuais enfermidades, o médico do mesmo pode visualizar o histórico do paciente. E essas informações reunidas em um único local permite ao médico e paciente uma visão geral do estado de saúde do usuário.

No módulo fiscal, o usuário realiza o controle das suas compras, visualizando onde estão concentrados seus maiores gastos, onde há possibilidade de poupar e assim, consegue gerir melhor suas finanças.

6.1 Trabalhos Futuros

O principal fundamento desse projeto é o armazenamento de dados, transformando-os em informações úteis na vida do usuário. Nessa mesma linha de pensamento, vários outros dados poderiam ser armazenados. Continuando o trabalho no módulo de área da saúde, poderiam ser armazenados no sistema dados como receitas, vacinas ou alergias. Se desenvolvido um módulo na área de educação, poderia-se armazenar notas desde o primário até a graduação, pós graduação; artigos; provas realizadas; etc. No ramo do trabalho, poderia-se armazenar informações sobre empresas e projetos no qual o trabalhou

Além do registro de diferentes dados, o sistema pode ser adaptado para fazer diversas análises de dados. Exemplos de funções incrementais no módulo fiscal seriam:

- De acordo com o histórico de compras dos usuários ativos do sistema, obter o estabelecimento que vende determinado produto com menor preço;
- Ou a partir de uma determinada lista de compras, selecionar um número definido de estabelecimentos, mostrando onde comprar cada item, de maneira a minimizar o custo total da compra, levando em conta os custos de deslocamento;

No módulo de saúde, poderia-se extrair dados de maneira a fazer levantamentos estatísticos sobre a incidência de doenças e assim, direcionar campanhas de prevenção para o local.

Por fim, uma melhoria que traria mais comodidade ao usuário, seria uma parceria com um fabricante de medidores de biodados. O aparelho seria desenvolvido de maneira a enviar automaticamente os dados das medições para o sistema, sem a necessidade da inclusão manual dos dados.

Referências

- ANÁLISE de Requisitos. Disponível em: <<http://www.governancamunicipal.sp.gov.br/conteudo/arquivos/Analise%20de%20requisitos.pdf>>. Acesso em: 9 de abril 2016. Nenhuma citação no texto.
- ANTONIORICARDO. *O que é SaaS, IaaS e PaaS em Cloud Computing? (Conceitos básicos)*. 2013. Disponível em: <<https://antonioricardo.org/2013/03/28/o-que-e-saas-iaas-e-paas-em-cloud-computing-conceitos-basicos/>>. Citado na página 29.
- BEAZLEY, D.; JONES, B. K. *Python Cookbook*. [S.l.]: Novatec Editora Ltda, 2013. Nenhuma citação no texto.
- BRASIL pode ter 80Disponível em: <<http://www.sbh.org.br/geral/noticias.asp?id=69>>. Acesso em: 31 de março 2016. Nenhuma citação no texto.
- BUYA R., Y. C. S. V. S. B. J.; BRANDIC, I. *Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility*. 2009b. Citado na página 19.
- CASOS De Uso. Disponível em: <<http://www.dsc.ufcg.edu.br/~sampaio/cursos/2007.1/Graduacao/SI-II/Uml/diagramas/usecases/usecases.htm>>. Acesso em: 9 de abril 2016. Nenhuma citação no texto.
- DIRETRIZES para o cuidado das pessoas com doenças crônicas nas redes de atenção à saúde e nas linhas de cuidado prioritárias. [S.l.]. Disponível em: <http://bvsms.saude.gov.br/bvs/publicacoes/diretrizes%20_cuidado_pessoas%20_doencas_cronicas.pdf>. Acesso em: 30 de março 2016. Nenhuma citação no texto.
- DOCUMENTAÇÃO Bootstrap. Disponível em: <<http://getbootstrap.com/getting-started/>>. Acesso em: 16 de abril 2016. Nenhuma citação no texto.
- DOCUMENTAÇÃO ChartJS. Disponível em: <<http://www.chartjs.org/docs/>>. Acesso em: 16 de abril 2016. Nenhuma citação no texto.
- DOCUMENTAÇÃO Django. Disponível em: <<https://docs.djangoproject.com/en/1.7/>>. Acesso em: 16 de abril 2016. Nenhuma citação no texto.
- DOCUMENTAÇÃO Django Rest. Disponível em: <<http://www.django-rest-framework.org/>>. Acesso em: 16 de abril 2016. Nenhuma citação no texto.
- DOCUMENTAÇÃO JQuery. Disponível em: <<http://api.jquery.com/>>. Acesso em: 16 de abril 2016. Nenhuma citação no texto.
- DOENÇAS crônicas, Capítulo 24. [S.l.]. Disponível em: <http://hesperian.org/wp-content/uploads/pdf/pt_wtnd_2009/pt_wtnd_2009_24.pdf>. Acesso em: 31 de março 2016. Nenhuma citação no texto.
- DOENÇAS Transmissíveis e Não-Transmissíveis. Disponível em: <http://www.paho.org/bra/index.php?option=com_content&view=article&id=569:-conceito&catid=901:bra-03-a-doencas-nao-transmissiveis&Itemid=539>. Acesso em: 31 de março 2016. Nenhuma citação no texto.

ELMAN, J.; LAVIN, M. *Django Essencial*. [S.l.]: O'Reilly, 2015. Nenhuma citação no texto.

EMISSOR de Cupom Fiscal (ECF) - informações e instruções. Disponível em: <http://www.fazenda.mg.gov.br/empresas/ecf/>. Acesso em: 2 de abril 2016. Nenhuma citação no texto.

GREENFIELD, D.; GREENFELD, D.; ROY, A. *Two Scoops of Django: Best Practices for Django 1.6*. [S.l.]: Novatec Editora Ltda, 2013. Nenhuma citação no texto.

GUIA Grátis do Cupom Fiscal Eletrônico. Disponível em: <http://www.cupomfiscaleletronico.org/cupom-fiscal-eletronico/>. Acesso em: 2 de abril 2016. Nenhuma citação no texto.

HEUSER, C. A. *Projeto de Banco de Dados*. [S.l.]: Artmed, 2008. Citado na página 42.

HIPERTENSÃO. Disponível em: <http://www.minhavidacom.br/saude/temas/hipertensao/>. Acesso em: 31 de março 2016. Nenhuma citação no texto.

HIPERTENSÃO Arterial. Disponível em: <http://www.abc.med.br/p/22140/hipertensao+arterial.htm>. Acesso em: 31 de março 2016. Nenhuma citação no texto.

JAZAYERI, A. R. M.; LINDEN, F. *Software Architecture for Product Families: Principles and Practice*. 2000. Citado na página 49.

LECHETA, R. R. *Google Android - 4ª Edição*. [S.l.]: Novatec Editora Ltda, 2013. Nenhuma citação no texto.

LECHETA, R. R. *AWS para Desenvolvedores*. [S.l.]: Novatec Editora Ltda, 2014. Nenhuma citação no texto.

M., M. *Evolution and Composition Object-Oriented Frameworks*. Tese (Doutorado) — Department of Software Engineering and, University of Karlskrona/Ronneby, 2000. Nenhuma citação no texto.

MATTSSON, M. *Object-oriented Frameworks - A survey of methodological issues*. 1996. Citado na página 34.

MENEZES, N. N. C. *Introdução à Programação com Python - 1ª Edição*. [S.l.]: Novatec Editora Ltda, 2010. Nenhuma citação no texto.

NFC-E. Disponível em: <http://vinco.com.br/nfc-e/>. Acesso em: 2 de abril 2016. Nenhuma citação no texto.

O que é Pressão Arterial Sistólica e Diastólica? Disponível em: <http://cienciadotreinamento.com.br/o-que-e-pressao-arterial-sistolica-e-diastolica/>. Acesso em: 31 de março 2016. Nenhuma citação no texto.

O que é um cupom fiscal? Disponível em: <http://controlware.com.br/o-que-e-um-cupom-fiscal>. Acesso em: 2 de abril 2016. Nenhuma citação no texto.

O que é um framework? Disponível em: <http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/frame/oque.htm>. Acesso em: 9 de abril 2016. Nenhuma citação no texto.

OLIVEIRA, A. L. de et al. Computação em nuvens. *REVISTA DE TRABALHOS ACADÊMICOS*, v. 0, n. 1, 2014. ISSN 2179-1589. Disponível em: <<http://revista.universo.edu.br/index.php?journal=1reta2&page=article&op=view&path%5B%5D=1188&path%5B%5D=892>>. Citado na página 29.

PERGUNTAS frequentes. Disponível em: <<http://www.sbh.org.br/geral/faq.asp>>. Acesso em: 31 de março 2016. Nenhuma citação no texto.

PRESSMAN, R. S. *Engenharia De Software*. [S.l.]: Bookman, 2011. Citado 2 vezes nas páginas 37 e 39.

RAMALHO, L. *Python Fluente*. [S.l.]: Novatec Editora Ltda, 2015. Nenhuma citação no texto.

RUSCHEL MARIANA SUSAN ZANOTTO, W. C. d. M. H. *Computação em Nuvem*. Disponível em: <<http://www.ppgia.pucpr.br/~jamhour/RSS/TCCRSS08B/Welton%20Costa%20da%20Mota%20-%20Artigo.pdf>>. Nenhuma citação no texto.

SAÚDE lança plano para reduzir taxa de mortalidade por doenças crônicas. Disponível em: <<http://www.brasil.gov.br/saude/2012/04/saude-lanca-plano-para-reduzir-taxa-de-mortalidade-por-doencas-cronicas>>. Acesso em: 30 de março 2016. Nenhuma citação no texto.

SILVA, M. S. *Fundamentos de HTML5 e CSS3*. [S.l.]: Novatec Editora Ltda, 2015. Nenhuma citação no texto.

Sociedade Brasileira de Hipertensao. *O que e hipertensao*. 2016. Disponível em: <<http://www.sbh.org.br/geral/oque-e-hipertensao.asp>>. Acesso em: 31 de março 2016. Citado na página 27.

SOMMERVILLE, I. *Engenharia De Software*. [S.l.]: Pearson - Addison Wesley, 2007. Nenhuma citação no texto.

WITTIG, A.; WITTIG, M. *Amazon Web Services em Ação*. [S.l.]: Manning Publications Co., 2015. Nenhuma citação no texto.

Anexos

ANEXO A – Casos de Uso

A.0.1 Caso Dois

Nome	Login do Usuário
Descrição Sucinta	Usuário realiza o login no sistema
Atores	Usuário Cadastrado
Pré-Condições	O usuário precisar ter realizado o passo um
Fluxo Básico	1. Usuário entra no site. 2. Na tela inicial do sistema existem formulários de cadastro e de login do usuário. 3. Usuário preenche os campos do formulário de login. 4. Usuário clica no botão entrar. 5. Sistema redireciona para a página de inicial do sistema. 6. Caso de uso encerrado.
Fluxo Alternativo	1. Alternativa ao passo 5. • O sistema verifica se o usuário e senha são válidas. Mas o usuário não é valido. • Sistema retorna ao passo 3 com alertas de erro. 2. Alternativa ao passo 5. • O sistema verifica se o usuário e senha são válidas. Mas a senha não é válida. • Sistema retorna ao passo 3 com alertas de erro.
Estrutura de Dados	1. Login • Nome do usuário • Senha

A.0.2 Caso Três

Nome	Cupons Fiscais - Buscas
Descrição Sucinta	Usuário realiza busca por cupons fiscais no sistema.
Atores	Usuário Cadastrado
Pré-Condições	1. O usuário ter realizado o caso um. 2. O usuário ter realizado o caso dois.
Fluxo Básico	1. Navega até a área de buscas de cupons fiscais. 2. O sistema apresenta um formulário com opções da busca. 3. O usuário realiza o preenchimento dos campos como bem deseja. 4. O usuário clica no botão pesquisar. 5. O sistema retorna a resposta que pode ser uma mensagem avisando que não encontrou itens ou uma lista com os dados encontrados. 6. Caso de uso encerrado.
Fluxo Alternativo	—
Estrutura de Dados	1. TDA Busca. • Tipo de Estabelecimento • Estabelecimento • Data Inicial • Data Final

A.0.3 Caso Quatro

Nome	Cupons Fiscais - Detalhes
Descrição Sucinta	Após realização da busca por cupons fiscais com resultado é possível visualizar os detalhes dos cupons.
Atores	Usuário Cadastrado
Pré-Condições	1. O usuário ter realizado o caso um. 2. O usuário ter realizado o caso dois. 3. O usuário ter realizado o caso três.
Fluxo Básico	1. Na lista apresentada existe um botão(com valor) detalhes, o usuário clica neste botão. 2. O sistema exibe uma janela com os detalhes daquele cupom. 3. Caso de uso encerrado.
Fluxo Alternativo	—
Estrutura de Dados	—

A.0.4 Caso Cinco

Nome	Cupons Fiscais - Relatórios – Gastos Anuais de um estabelecimento
Descrição Sucinta	O usuário solicita ao sistema um relatório em formato de barras que pode ser customizado.
Atores	Usuário Cadastrado
Pré-Condições	1. O usuário ter realizado o caso um. 2. O usuário ter realizado o caso dois.
Fluxo Básico	1. Navega até a área de cupons fiscais - relatórios, seleciona o relatório Gastos Anuais de um estabelecimento. 2. O sistema apresenta um formulário com as opções do relatório. 3. O usuário preenche as opções desejadas. 4. O usuário clica no botão pesquisar. 5. O sistema retorna um gráfico com o resultado. 6. Caso de uso encerrado.
Fluxo Alternativo	—
Estrutura de Dados	1. TDA Relatório. • Tipo de Estabelecimento • Estabelecimento • Ano

A.0.5 Caso Seis

Nome	Cupons Fiscais - Relatórios – Compras em um período de todos os seguimentos.
Descrição Sucinta	O usuário solicita ao sistema um relatório em formato de barras que pode ser customizado.
Atores	Usuário Cadastrado
Pré-Condições	1. O usuário ter realizado o caso um. 2. O usuário ter realizado o caso dois.
Fluxo Básico	1. Navega até a área de cupons fiscais - relatórios, seleciona o Compras em um período de todos os seguimentos. 2. O sistema apresenta um formulário com as opções do relatório. 3. O usuário preenche as opções desejadas. 4. O usuário clica no botão pesquisar. 5. O sistema retorna os resultados que são uma mensagem que não nada ou um gráfico com os resultados. 6. Caso de uso encerrado.
Fluxo Alternativo	—
Estrutura de Dados	1. TDA Relatório. • Data Inicial • Data Final

A.0.6 Caso Sete

Nome	Cupons Fiscais - Relatórios – Gráfico por tipo de estabelecimento.
Descrição Sucinta	O usuário solicita ao sistema um relatório em formato de barras que pode ser customizado.
Atores	Usuário Cadastrado
Pré-Condições	1. O usuário ter realizado o caso um. 2. O usuário ter realizado o caso dois.
Fluxo Básico	1. Navega até a área de cupons fiscais - relatórios, seleciona o gráfico por tipo de estabelecimento. 2. O sistema apresenta um formulário com as opções do relatório. 3. O usuário preenche as opções desejadas. 4. O usuário clica no botão pesquisar. 5. O sistema retorna os resultados que são uma mensagem que não nada ou um gráfico com o resultado. 6. Caso de uso encerrado.
Fluxo Alternativo	—
Estrutura de Dados	1. TDA Relatório. • Tipo de Estabelecimento • Ano

A.0.7 Caso Oito

Nome	Login no aplicativo
Descrição Sucinta	Usuário realiza login no aplicativo de saúde utilizando as credenciais criadas no web site
Atores	Usuário Registrado
Pré-Condições	—
Fluxo Básico	1. Usuário entra no app. 2. Na tela inicial do sistema é exibido um formulários de login. 3. O usuário preenche os campos de usuário e senha. 4. Usuário clica no botão entrar. 5. O Sistema autentica as credenciais. 6. O aplicativo redireciona para a página de inicial. 7. Caso de uso encerrado.
Fluxo Alternativo	1. Alternativa ao passo 6. • O sistema não encontra usuário cadastrado com os dados informados. • O aplicativo permanece na tela de login.
Estrutura de Dados	1. Login. • Usuário • Senha

A.0.8 Caso Nove

Nome	Visualizar informações no app.
Descrição Sucinta	O usuário visualizará os biosdados enviados por o mesmo. Os biodados devem ser divididos por tipo
Atores	Usuário Cadastrado
Pré-Condições	1. O usuário ter realizado o caso treze.
Fluxo Básico	1. O usuário encontra-se logado no sistema. 2. O mesmo através de <i>tab</i> superiores escolhe o tipo de biodado a ser visualizado. 3. Caso de uso encerrado.
Fluxo Alternativo	—
Estrutura de Dados	—

A.0.9 Caso Dez

Nome	APP Saúde – Enviar biodados.
Descrição Sucinta	O usuário enviará biodados das medições realizadas.
Atores	Usuário Cadastrado
Pré-Condições	1. O usuário ter realizado o caso treze.
Fluxo Básico	1. O usuário encontra-se logado no sistema. 2. No canto superior é exibido um botão com o valor novo registro. 3. O usuário clica no botão. 4. Um uma nova janela com campos é exibida. IMPORTANTE, para cada tipo de bio dado deverá ser exibido uma janela com os dados necessários. 5. O usuário clica no botão gravar e os dados são enviados para o servidor e gravados. 6. Caso de uso encerrado.
Fluxo Alternativo	—
Estrutura de Dados	1. TDA Variado de acordo com o biodado. • Batimento* • Glicose* • Pressão Arterial*

ANEXO B – Executando o sistema

B.1 Sistema Web

Para executar o sistema é necessário possuir um computador com a linguagem *Python* instalada. No caso de computadores com *Windows*, a linguagem pode ser obtida através do site <<http://www.python.org.br>>. Nos casos de usuários *Linux*, a linguagem faz parte do sistema.

Outro requisito para execução é o usuário possuir a biblioteca do *Python* chamada *PIP* que pode ser encontrada em <<https://pip.pypa.io/en/stable/>>. A função desta é realizar a instalação das dependências necessárias para o projeto.

O código do sistema pode ser obtido no repositório GitHub através do LINKREPOSITORIO [assim que apresentar irei publicar e atualizar aqui]. Caso o usuário possua o sistema *GIT* instalado na máquina e o código pode ser obtido executando o seguinte comando *GIT clone* LINKREPOSITORIO. Se o usuário não possuir *GIT* e deseja instalá-lo poderá fazer o download através do site oficial <<https://git-scm.com/>>.

Todo o procedimento de execução é realizado através de *CMD* no caso de *Windows* ou *Terminal* no caso de *Linux*.

Abra o *CMD* ou o *Terminal* e navegue até a pasta onde o código está armazenado. Nessa pasta existe um arquivo chamado *requirements.txt* que armazena todas as dependências necessárias para a execução do sistema. Para a instalação destas dependências execute o comando *pip install -r requirements.txt*.

Com isso, estamos aptos a executar o sistema.

B.1.1 Executando

Para executar o sistema utilize o comando *python manage.py runserver*. Com isso, através do *browser* é possível acessar o sistema no endereço 127.0.0.1:8000.

B.1.2 Banco de Dados

Para criar banco de dados ou atualizar alguma modificação realizada nos *models.py* execute os seguintes comandos: *python manage.py makemigrations* e *python manage.py migrate*.

B.1.3 Testes

Durante o desenvolvimento viu-se a necessidade de dados simulando a utilização do sistema. Com isso, foi desenvolvido alguns *scripts* que geram dados. Os *scripts* podem ser visto através do comando `python manage.py -h`.

Um exemplo de utilização para a criação de batimentos cardíacos segue o comando: `python manage.py bpm_populate_db --usuarioId=1 --qtdDeDias=10 --qtdPorDia=2`. Com a execução desse comando seria criado para o usuário com id 1 por 10 dias 2 registros de batimento.

B.2 Aplicativo Mobile

Para utilizar o APP Android, o usuário deverá possuir a APK que pode ser encontrada no seguinte repositório LINKREPOSITORIO. Ainda neste repositório está armazenado todo o código do aplicativo.

Uma informação pertinente em relação ao APP é que existe um arquivo localizado em `src/../../util/conexaoUtil.java` que controla a conexão com o sistema online. Esse controle é realizado através da variável `URL_BASE` que armazena o endereço do servidor. Caso esse dado mude é necessário modificar a informação contida na variável.