

UNIVERSIDADE FEDERAL DO PIAUÍ
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS
CURSO BACHARELADO EM SISTEMAS DE INFORMAÇÃO

SYSNUT – UM SISTEMA DE AUXÍLIO À NUTRICIONISTA

LINNEU MAGNO LOPES DE SOUSA

PICOS-PI

2017

LINNEU MAGNO LOPES DE SOUSA

SYSNUT - UM SISTEMA DE AUXÍLIO À NUTRICIONISTA

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharel em Sistemas de Informação, Campus Senador Helvídio Nunes de Barros, da Universidade Federal do Piauí como parte dos requisitos para obtenção do referido grau.

Orientador: Prof. MSc. Patricia Medyna Lauritzen de Lucena Drumond.

PICOS-PI

2017

Dedicatória

AGRADECIMENTOS

RESUMO

A alimentação é muito importante na vida das pessoas. Devido a vida corriqueira destas pessoas, aumentou-se a demanda pelos *fast-food*, alimentos estes ricos em substâncias prejudiciais a saúde. A preocupação com a melhora e qualidade de vida faz com que as pessoas busquem cada vez mais o nutricionista. Por outro lado, o nutricionista encontra um trabalho repetitivo e que necessita de organização, do contrário pode cometer erros que comprometam o seu trabalho com o paciente. Diante dessas dificuldades, surge a demanda por ferramentas que auxiliem o trabalho do nutricionista, possibilitando um trabalho com mais agilidade e que atenda às necessidades do seu cotidiano profissional. O sistema deste trabalho propõe uma ferramenta de fácil aprendizagem, ao tempo que pretende proporcionar uma boa experiência para o profissional que a utiliza.

Palavras-chave: Alimentação, *Python* e *Django*, *Systema Web*, Sistema de apoio ao nutricionista.

ABSTRACT

Food is very important in people's lives. Due to the everyday life of these people, the demand for fast food, which is rich in substances harmful to health, has increased. The concern with the improvement and quality of life causes people to seek more and more the nutritionist. On the other hand, the nutritionist finds a repetitive work and needs organization, otherwise he can make mistakes that compromise his work with the patient. Faced with these difficulties, there is a demand for tools that help the work of the nutritionist, enabling a more agile work and that meets the needs of their professional life. The system of this work proposes an easy-to-learn tool, while also aiming to provide a good experience for the professional who uses it.

Keywords: Food, Python and Django, Web System, Nutritionist support system.

LISTA DE FIGURAS

Figura 1 -	Exemplo de Código HTML	16
Figura 2 -	Código CSS	
Figura 3 -		
Figura 4 -		
Figura 5 -		
Figura 6 -		
Figura 7 -		
Figura 8 -		
Figura 9 -		
Figura 10 -		
Figura 11 -		
Figura 12 -		
Figura 13 -		
Figura 14 -		

LISTA DE TABELAS

Tabela 1 - <i>Product BackLog</i>	26
---	----

LISTA DE SIGLAS E ABREVIATURAS

DCNT	Doenças Crônicas Não Transmissíveis
CSS	<i>Cascading Style Sheets</i>
DRY	<i>Don't Repeat yourself</i>
ETA	Efeito Térmico dos Alimentos
GPL	<i>General Public License</i>
HTML	<i>HyperText Markup Language</i>
JSON	<i>JavaScript Object Notation</i>
MVC	<i>Model-View-Controller</i>
MTV	<i>Model-Template-View</i>
NSF	<i>National Science Foundation</i>
OMG	<i>Object Management Group</i>
ORM	<i>Object-relational mapper</i>
PSF	<i>Python Software Foundation</i>
SP	<i>Story points</i>
SGBD	Sistema de Gerenciamento de Banco de Dados
SQL	<i>Structured Query Language</i>
TMB	Taxa Metabólica Basal
GET	Gasto Energético Total
URL	<i>Uniform Resource Locator</i>
UML	<i>Unified Modeling Language</i>
VN	Valor de negócio
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1 INTRODUÇÃO

A alimentação é um dos momentos mais importantes na vida das pessoas tanto por fatores biológicos logicamente, como por fatores sociais, culturais e científicos. É por meio dela que obtemos a energia necessária para desenvolver nossas atividades diárias e outros nutrientes primordiais a nossa saúde, como vitaminas e minerais. (PROENÇA, 2010).

O Brasil atualmente passa por uma etapa de transição na alimentação de sua população, a mesma que antes enfrentava a desnutrição, nos dias atuais apresenta alta prevalência de obesos, isso pode ser explicado pela mudança drástica na alimentação. Com o aumento da renda básica das famílias, o *fast food* tornou-se presente nas mesas da população brasileira. Alimentos como este, são ricos em gordura saturada, açúcar e sódio precursores de diversas doenças crônicas não transmissíveis (DCNT) como, diabetes mellitus tipo II, doenças no sistema cardiovascular, obesidade mórbida entre outras (SCHUSTER et al., 2015).

A preocupação com a alimentação e melhora na qualidade de vida se faz cada dia mais presente no cotidiano de todos, o nutricionista, profissional responsável pela prescrição dietoterápica, tem ganho mais cada vez mais espaço no mercado de trabalho, perante sua importância para manutenção da saúde do ser humano (BRASIL, 2014).

Diante da importância do seu trabalho, o nutricionista tem buscado melhorias no seu atendimento, que por vezes se torna cansativo perante a vasta quantidade de cálculos para análise do estado nutricional do paciente, das necessidades energéticas e para adequação do cardápio. A necessidade de agilizar e aprimorar o atendimento nutricional acarretou o surgimento de softwares que atendessem as necessidades básicas como resolução de equações e maior organização dos dados dos pacientes (VIEIRA, 2009).

1.1 Objetivo

Esse trabalho teve como objeto o desenvolvimento de um sistema para nutricionistas, voltado para o atendimento nutricional de seus pacientes, atendendo todas as necessidades primarias de um atendimento individualizado.

1.2 Organização do Documento

Este trabalho será apresentado nos próximos capítulos organizado da seguinte maneira:

- Capítulo 2 - Estado da Arte: Capítulo onde será relatado as principais tecnologias utilizadas durante o desenvolvimento, descrevendo seus principais conceitos.
- Capítulo 3 – Problemática: Capítulo onde será descrito os principais problemas relatados que o sistema solucionou.
- Capítulo 4 – Sistema Desenvolvido e seus Resultados: Capítulo onde é apresentado o sistema desenvolvido e seus respectivos resultados.
- Capítulo 5 – Conclusões e Trabalhos Futuros: Neste capítulo, relata-se os conhecimentos que o sistema proporcionou e os principais problemas enfrentados. Relata-se também as intenções futuras em relação ao mesmo.

2 ESTADO DA ARTE

Durante o desenvolvimento de um software, é comum a utilização de vários recursos para que se possa concluir o objetivo com excelência. Neste caso, trata-se de uma aplicação *web* e, portanto, não é diferente. Foram utilizadas várias ferramentas e tecnologias que auxiliaram na obtenção do objetivo. As principais serão abordadas neste capítulo.

2.1 Servidor

Os sistemas *web* funcionam através de requisições: o usuário (cliente) faz uma requisição de uma página para o servidor e retorna uma resposta para ele através do navegador. A resposta é interpretada pelo navegador e entregue pelo servidor para o usuário (NIEDERAUER, 2017).

2.1.1 *Python*

Python é uma linguagem de programação de altíssimo nível, orientada a objeto e de tipagem dinâmica e forte, interpretada e interativa. O nome *Python* foi dado por Guido van Rossum do programa da TV Britânica *Monty Python's Flying Circus* e a linguagem foi criada por ele, na década de 90, no Instituto Nacional de Pesquisa para Matemática e Ciência da Computação da Holanda (CWI). O *Python* foi concebido a partir de outra linguagem existente na época, chamada ABC.

A intenção do autor ao criar a linguagem é que fosse de altíssimo nível e que tivesse características importantes de outras linguagens de programação. *Python* é livre, ou seja, tem código aberto, com licença compatível com a *General Public License* (GPL). A especificação da linguagem é mantida pela *Python Software Foundation* (PSF). A linguagem *Python* apresenta uma série de vantagens e recursos interessantes que foram inspirados e aproveitados de outras linguagens de sucesso, tornando-a assim uma linguagem bastante flexível (BORGES, 2010).

Python possui diversas características e estruturas de alto nível (listas, dicionários, entre outros) e uma coleção de módulos prontos para uso, além de outras

ferramentas que podem ser utilizadas para auxiliar no desenvolvimento, dentre elas citamos o *Django*, que será abordado no tópico a seguinte.

Segundo a *IEEE Spectrum*, Python lidera o ranking de linguagens mais utilizadas (DIAKOPOULOS e CASS, 2017), seguido por C e Java. Existem também implementações de *Python* para *.NET* (*IronPython*), *JVM* (*Jython*), entre outros.

2.1.2 *Django*

O *Django* é um *framework* de desenvolvimento web criado por Jacob Kaplan-Moss, Adrian Holovaty e Simon Willison em 2003. O *Django* utiliza o conceito de *DRY* – *Don't Repeat yourself* (“não repita a si mesmo”), que propõe que cada parte de um sistema deve possuir uma representação única.

O *Django* possui vários componentes com funções específicas. Neto (2010) alguns de seus principais componentes:

- **ORM – Object-relational mapper (Mapeador objeto-relacional):** é uma técnica de mapeamento objeto relacional que permite fazer uma relação dos objetos com os dados que estes representam.
- **Template System:** linguagem para criação de templates (*HTML*, *XML*, *JSON*, etc.) usados na geração de páginas dinâmicas.
- **Sistema de administração:** Interface de administração própria do framework (*Django-admin*).
- **URL (Uniform Resource Locator) dispatcher:** Processador de URLs do sistema que executando funções específicas feitas pelo desenvolvedor, possibilitando URL's amigáveis.
- **Internacionalização:** Permite que o sistema seja traduzido para diversos idiomas.
- **Formulários:** geração automática de formulários e facilitação na manipulação dos dados enviados por meio deles.
- **Segurança:** gerenciamento de autenticação de usuários e controle de permissões.
- **Outros componentes:** serialização de dados, sistema de testes automatizado, entre outros.

O *Django* utiliza o padrão de projeto MTV (*Model-Template-View*) para desenvolvimento, que possui essencialmente a mesma lógica que o MVC (*Model-View-Controller*), muito utilizado em outras linguagens (NETO, 2010).

O padrão de projeto MVC divide-se em três camadas: Modelo, Visão e Controlador. O Modelo é o responsável pela comunicação com os dados armazenados no banco de dados que serão visualizados na camada de Visão. A Visão, por sua vez, é responsável pela apresentação da aplicação. E por último, o controlador, responsável por administrar todo o fluxo da aplicação (LEMOS et al., 2013).

2.1.3 Banco de dados MySQL

O *MySQL*, criado pela empresa *MySQL AB* na Suécia, é um SGBD que utiliza a linguagem SQL como interface. A *MySQL AB* foi adquirida pela *Sun Microsystems* em 2008, a qual posteriormente foi comprada pela *Oracle*, incluindo o SGBD *MySQL*. Atualmente, esse é um dos bancos de dados mais utilizados do mundo, com mais de 10 milhões de instalações e um dos motivos para essa popularidade é devido a fácil integração a também popular linguagem PHP (MOTTIN et al., 2014).

2.2 Cliente

O lado cliente de uma aplicação *web* é quem realiza a requisição e se comunica com o servidor, ao tempo que recebe a resposta do mesmo. Em outras palavras, o lado cliente é quem estabelece a comunicação com o usuário.

2.2.1 HTML

HTML é uma sigla em inglês que significa Linguagem para Marcação de Hipertexto (*HyperText Markup Language*). Na aplicação, é usada para denotar os elementos da página, tais como caixas de seleção, caixas de texto, etc. Sua função é definir apenas a estrutura da página (FOLLE, 2012).

A Figura 1 é um exemplo simples de uma página com código HTML, mostrando algumas tags HTML e suas funcionalidades de marcação:

```
<html>
<head>
  <title>Minha primeira página HTML</title>
</head>
<body>
  Corpo da página.
</body>
</html>
```

Figura 1 – Exemplo de código HTML

Fonte: O autor (2017)

2.2.2 *Cascading Style Sheets – CSS*

CSS (*Cascading Style Sheets*) – linguagem utilizada para tratamento visual dos elementos da página *web* (FOLLE, 2012). Este é, portanto, a tecnologia responsável por tratar cada elemento da página *web* visualmente, atribuindo cores e posições diferentes, dependendo da preferência do programador. Possui sintaxe simples e assim como as demais tecnologias, utiliza-se de várias palavras em inglês para especificar os diferentes estilos de propriedades da página.

```
body{
  background-color: white;
  font-size: 10px;
  font-weight: bold;
}
```

Figura 2 - Código CSS

Fonte: O autor (2017)

2.2.3 *JavaScript e JQuery*

Segundo Folle (2012), *JavaScript* “é uma linguagem de script executada pelos navegadores que permite o acesso e manipulação programática de objetos de uma página *web*”. Possibilita então que um campo para digitar uma data, seja formatado

automaticamente (com barras), ou que um campo seja desabilitado dinamicamente, etc.

Jquery é portanto, uma biblioteca de funções prontas de *JavaScript*, que interagem com o HTML e possibilitam uma melhor experiência ao usuário. Foi lançada em 2006, no BarCamp, de Nova York, por John Resig e é utilizada por milhares de sites visitados pelo mundo. Segundo a W3TECHS (2017), é a biblioteca JavaScript mais popular dentre as existentes.

2.3 Engenharia de Software

Segundo Rohden, “a Engenharia de Software propõe métodos sistemáticos com o uso adequado de ferramentas e técnicas, que levam em consideração o problema a ser resolvido, necessidades dos clientes e os recursos disponíveis”. Considerando estes elementos, para obter software de qualidade, esta engenharia é imprescindível, pois leva em consideração os principais elementos a serem estudados e avaliados.

2.3.1 Scrum

Scrum é uma metodologia de desenvolvimento ágil que prioriza explicitamente o retorno de investimento. O *Product Owner* (pessoa que define os itens importantes que compõe o *Product Backlog*) tem a função de maximizar e priorizar as atualizações do *Product Backlog*, que é a lista de funcionalidades desejadas no desenvolvimento do produto, de forma que os itens de maior valor para o cliente em cada momento sejam implementados primeiro. Dessa forma, o incremento ao produto realizado ao final de cada *Sprint*(itens desenvolvidos em um ciclo) gera retorno ao cliente por diversas vezes ao longo do projeto (MACHADO e MEDINA, 2009).

Machado e Medina enfatiza ainda que saber responder as mudanças é mais importante do que seguir um plano. Por se tratar de um *framework* ágil, *Scrum* encara as mudanças como parte natural do processo de desenvolvimento, através das atualizações do *Product Backlog*, onde as novas solicitações do cliente podem ser introduzidas no próximo *Sprint*, gerando vantagem competitiva para as empresas.

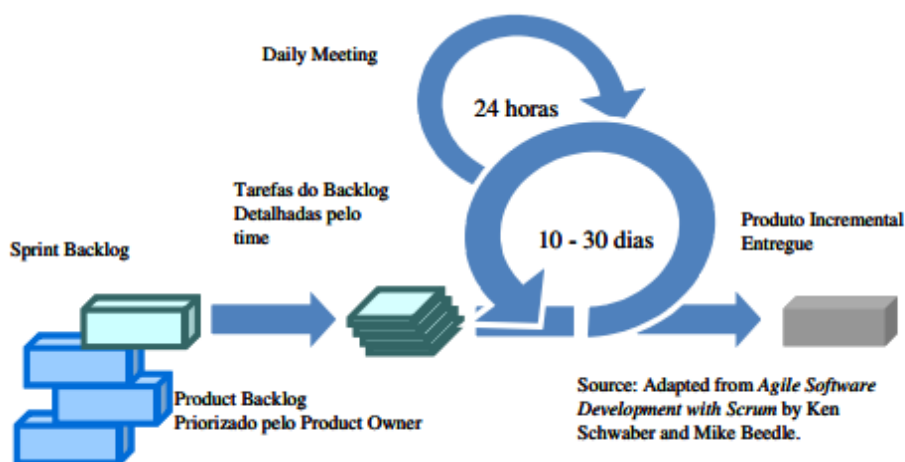


Figura 3 - Ciclo de Desenvolvimento Scrum

Fonte: VARASCHIM (2009, P.03)

A figura 3 apresenta o ciclo de desenvolvimento *Scrum*, que é dividido em *Sprints*, que podem durar de 10 a 30 dias e no final do *sprint* é mostrado ao *Product Owner*. Durante cada *sprint* são realizadas verificações do andamento do projeto. Um conjunto de *sprints* que formam uma versão utilizada do produto é chamado de *release*.

2.4.2 Linguagem de Modelagem Unificada (*Unified Modeling Language – UML*)

Segundo Costa (2001), A UML surgiu da união de métodos anteriores para análise e projeto de sistemas orientados a objetos e em 1997 passou a ser aceita e reconhecida como um padrão potencial de notação para modelagem de múltiplas perspectivas de sistemas de informações pela OMG (*Object Managment Group*). A UML oferece um suporte direto para o projeto e implementação de cada perspectiva do sistema em desenvolvimento e também uma notação para sua representação. Neste sistema, a UML foi importante durante todo o processo sobretudo para visualização das funcionalidades e sua posterior implementação, bem como novas solicitações feitas pelo cliente. A UML apresenta atores e onde podem estar representados suas ações no sistema, facilitando a visualização do papel destes no sistema.

3 PROBLEMÁTICA

Durante o acompanhamento do paciente, o nutricionista muitas vezes precisa consultar diversos autores para efetuar um trabalho com mais precisão. Esta, por muitas vezes, vem a ser um ponto negativo, se não houver uma ferramenta que auxilie no agendamento de consultas e possibilite acompanhar a evolução do quadro.

3.1 Entendendo o Problema

O acompanhamento nutricional individualizado é extremamente importante para o paciente obter resultados relevantes de acordo com o objetivo traçado pelo nutricionista, visto que, cada indivíduo possui uma necessidade energética diferente e a distribuição dos macronutrientes e micronutrientes é realizada de acordo com seu tipo de organismo, objetivo a ser alcançado e estado nutricional, mediante avaliação antropométrica.

A nutrição é uma ciência exata, torna-se necessário o uso de diversas fórmulas e equações no dia a dia do nutricionista, seja para avaliar o estado nutricional do paciente ou para esquematizar um plano alimentar, fórmulas essas que nem sempre são sinônimos de praticidade e simplicidade. Esse fator pode trazer perdas significativas ao profissional da Nutrição, visto que são necessários vários cálculos para prestar atendimento adequado ao seu público-alvo.

Diante disso, faz-se necessário o uso de ferramentas computacionais que forneçam a inclusão de refeições, cálculo das necessidades energéticas e análise do cardápio auxiliando o profissional durante seu atendimento, trazendo mais comodidade e rapidez ao seu serviço prestado.

3.2 O Sistema Web

Diante as necessidades relatadas pelos graduandos e professores, foi possível fazer um levantamento de requisitos com histórias de usuário. Durante essa etapa, foi feito um relatório sobre as dificuldades que o sistema pudesse suprir através de uma ou mais funcionalidades.

Para iniciar o desenvolvimento, foi necessário ter ciência de quais as literaturas mais usadas pelos nutricionistas e a partir deles, poder criar um sistema que efetue cálculos e crie relatórios com base na literatura preferida pelo profissional. Abaixo, será citado quais as principais fórmulas utilizadas com base na literatura.

Taxa Metabólica Basal (TMB)

A taxa metabólica basal (TMB) é a quantidade de energia necessária para a conservação das funções vitais do organismo, representando a maior parte do consumo energético diário em humanos (cerca de 50% a 70%). Sendo calculada em condições padrão de jejum, repouso físico e mental em ambiente calmo com controle de temperatura, iluminação e barulhos (RUIZ JR et, al., 2011, HARRIS & BENEDICT, 1919).

A TMB sofre grande influência da massa magra, sexo, idade, composição corporal e predisposição genética. Fatores como o funcionamento do sistema nervoso e os hormônios tireoidianos, também contribuem para diferença da TMB entre os indivíduos. Para a estimativa da TMB, foram desenvolvidas várias equações matemáticas, utilizando variáveis de fácil mensuração e de baixo custo, como idade, altura e massa corporal total. Entre tantas equações, serão utilizadas as seguintes: Harris e Benedict (1919 e 1984) e Cunningham (1996) por possuírem grande aceitabilidade e credibilidade pelas entidades relacionadas a Nutrição (WEIJS PJM et, al., 2008).

Gasto Energético Total (GET)

O GET compreende a soma de todos os gastos energéticos diários a seguir: A taxa metabólica basal (TMB) que compreende o gasto energético necessário para a consumação das funções vitais do organismo; o gasto energético da atividade física ou fator de atividade (FA), que representa o gasto calórico com as atividades físicas do cotidiano e o exercício físico; e o efeito térmico dos alimentos (ETA), relacionado com a digestão, a absorção e o metabolismo dos alimentos. Em indivíduos saudáveis, a TMB corresponde aproximadamente a 60% a 70% do gasto diário, o ETA entre 5% e 15% e o GEAF ou FA de 15% a 30%, sendo este último o elemento que mais varia entre os indivíduos (HILL et, al., 2006).

Esse cálculo torna-se indispensável durante o atendimento nutricional, visto que ele é responsável pela individualização da sua dieta, de forma que possa atender todas as necessidades calóricas do mesmo, auxiliando na melhora do quadro nutricional do paciente (PEDROSA et, al., 2007)

O sistema deve então focar nos dados que estas fórmulas podem apresentar como resultado, a fim de que possa facilitar o atendimento por parte do nutricionista, ao tempo que o paciente obtenha um resultado com maior satisfação.

4 IMPLEMENTAÇÃO/RESULTADOS/CENÁRIO

O nutricionista solicitou um sistema que pudesse auxiliar nos cálculos da rotina de atendimento do paciente e também que pudesse agendar suas consultas, ao tempo em que pudesse visualizar o progresso do paciente. Também foi solicitado que fosse possível adicionar cardápios as suas consultas para que o paciente possa seguir suas recomendações.

4.1 Descrição da Implementação

Durante todo o desenvolvimento, foi necessário conhecer e entender os procedimentos de todas as partes (nutricionista e paciente), desde sua primeira visita ao consultório até o acompanhamento do nutricionista com prescrição cardápio, para que fosse possível atender às necessidades descritas, ao tempo que os resultados fossem mais claros e objetivos.

O sistema foi desenvolvido utilizando a metodologia *Scrum* (sub tópico 2.4.1), onde as análises são feitas sobre as histórias de usuário, sendo que cada uma delas representa uma necessidade do usuário do sistema. Primeiramente, foi definido o *Product BackLog*, com as histórias de usuário relatadas pelo *Product Owner*. Abaixo, um breve resumo do *Product BackLog* (Tabela 1).

ID	AÇÃO	DESCRIÇÃO	CRITÉRIOS DE ACEITAÇÃO	VALOR DE NEGÓCIO (VN)	ESTIMATIVA (STORY POINTS - SP)	ROI (VN/ SP)	SPRINT
1	Login	Sendo um administrador, preciso realizar login no sistema para que possa ter acesso ao sistema como administrador.	Verificar o autenticidade; Exibir página do usuário				

ID	AÇÃO	DESCRIÇÃO	CRITÉRIOS DE ACEITAÇÃO	VALOR DE NEGÓCIO (VN)	ESTIMATIVA (STORY POINTS - SP)	ROI (VN/ SP)	SPRINT
2	Login	Sendo um nutricionista, preciso realizar login no sistema para que possa ter acesso ao sistema como nutricionista.	Verificar o autenticidade; Exibir página do usuário.				
3	Cadastro	Sendo um nutricionista, preciso cadastrar um novo paciente para que possa fazer sua consulta e seu cardápio.	Verificar o preenchimento dos campos;				
4	Cadastro	Sendo um paciente, preciso informar os dados antropométricos para que o nutricionista possa realizar a anamnese alimentar do paciente com mais precisão.	Verificar o preenchimento dos campos;				
5	Cadastro	Sendo um nutricionista, preciso anexar documentos como laudos médicos para que ateste algum tipo de patologia que o paciente possa ter.	Verificar o preenchimento dos campos;				
6	Cadastro	Sendo um nutricionista, preciso cadastrar o fator de atividade física do paciente	Verificar o preenchimento dos campos;				

ID	AÇÃO	DESCRIÇÃO	CRITÉRIOS DE ACEITAÇÃO	VALOR DE NEGÓCIO (VN)	ESTIMATIVA (STORY POINTS - SP)	ROI (VN/ SP)	SPRINT
		para que o cardápio se adeque a sua rotina.					
7	Cadastro	Sendo um nutricionista, preciso relacionar alimentos que o paciente tenha intolerância, alergia para que possa ser feito o cardápio com suas particularidades .	Verificar o preenchimento dos campos;				
8	Cadastro	Sendo um nutricionista, preciso que o paciente relate sinais ou sintomas para que possa evitar algum tipo de reação a algum alimento ao qual não pode consumir.	Verificar o preenchimento dos campos;				
9	Cadastro	Sendo um nutricionista preciso cadastrar as principais patologias do paciente para que o cardápio esteja de acordo com suas necessidades.	Verificar o preenchimento dos campos;				
10	Cadastro	Sendo um nutricionista, preciso cadastrar uma opção diferente de cada refeição na dieta para que	Verificar o preenchimento dos campos;				

ID	AÇÃO	DESCRIÇÃO	CRITÉRIOS DE ACEITAÇÃO	VALOR DE NEGÓCIO (VN)	ESTIMATIVA (STORY POINTS - SP)	ROI (VN/ SP)	SPRINT
		o paciente possa optar por outro consumo, se preferir.					
11	Cadastro	Sendo um nutricionista, preciso definir o objetivo da dieta do paciente para que a orientação seja direcionada ao objetivo.	Verificar o preenchimento dos campos;				
12	Cadastro	Sendo um nutricionista, preciso realizar uma ou mais consultas de um paciente para que possa ser feita uma análise completa.	Verificar o preenchimento dos campos;				
13	Cadastro	Sendo um nutricionista, preciso que as consultas tenham datas marcadas para que seja possível organizar uma agenda para os pacientes.	Verificar o preenchimento dos campos;				
14	Cadastro	Sendo um nutricionista, preciso definir se a dieta está disponível ou não para o paciente para que o mesmo só tenha acesso quando a orientação final for dada.	Verificar o preenchimento dos campos;				
15	Cadastro	Como administrador preciso	Verificar dados da planilha;				

ID	AÇÃO	DESCRIÇÃO	CRITÉRIOS DE ACEITAÇÃO	VALOR DE NEGÓCIO (VN)	ESTIMATIVA (STORY POINTS - SP)	ROI (VN/SP)	SPRINT
		importar planilhas de alimentos para facilitar a escolha pelo nutricionista	Exibir mensagem de sucesso.				
16	Relatório	Sendo um nutricionista, preciso observar a evolução do paciente durante minhas consultas para poder concluir o trabalho.	Verificar consultas existentes; Exibir relatório com os dados;				

Tabela 1 - *Product BackLog*

O *Product BackLog* possui oito atributos, especificados a seguir:

1. *ID*: identifica unicamente uma história do *Product BackLog*.
2. *Ação*: define onde deve ser executado a história.
3. *Descrição*: contém a história de usuário
4. *CrITÉrios de aceitação*: contém os critérios para que a ação possa ser executada com êxito.
5. *Valor de Negócio*: define a importância da história.
6. *Estimativa (Story Points)*: estima o custo do (na visão do desenvolvedor) para se implementar a história.
7. *ROI (VN/SP)*: retorno do investimento em relação ao custo.
8. *Sprint*: define em que *Sprint* a funcionalidade foi implementada.

Com base nas histórias citadas na Tabela 1, o sistema foi dividido em 2 subsistemas. O primeiro define as ações do administrador e do nutricionista. A Figura 4 mostra o diagrama de casos de uso.

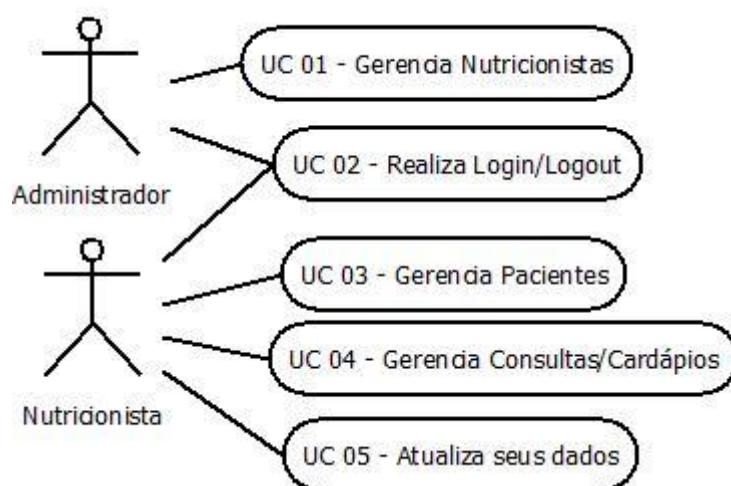


Figura 4 - Casos de uso administrador e nutricionista

Fonte: O autor (2017)

- **UC01 – Gerencia Nutricionistas**
 - **Ator:** Administrador
 - **Descrição:** O usuário administrador pode cadastrar, excluir e editar os nutricionistas do sistema.
- **UC02 – Realiza login e logout**
 - **Ator:** Administrador e Nutricionista
 - **Descrição:** Todos os usuários devem se autenticar, para que o sistema os identifique e defina suas permissões de acesso.
- **UC03 – Gerencia Pacientes**
 - **Ator:** Nutricionista
 - **Descrição:** O nutricionista pode cadastrar, editar e excluir pacientes do sistema.
- **UC04 – Gerencia Consultas/Cardápios**
 - **Ator:** Nutricionista
 - **Descrição:** O nutricionista pode cadastrar, editar e excluir consultas e cardápios feitas sobre um paciente.
- **UC05 – Atualiza seus dados**
 - **Ator:** Nutricionista
 - **Descrição:** O nutricionista pode editar dados a seu respeito, como Nome, E-mail, Senha de Login, etc.

O segundo subsistema, deixa claro as ações do nutricionista em relação ao paciente. A Figura 5 mostra o diagrama de casos de uso desse subsistema.

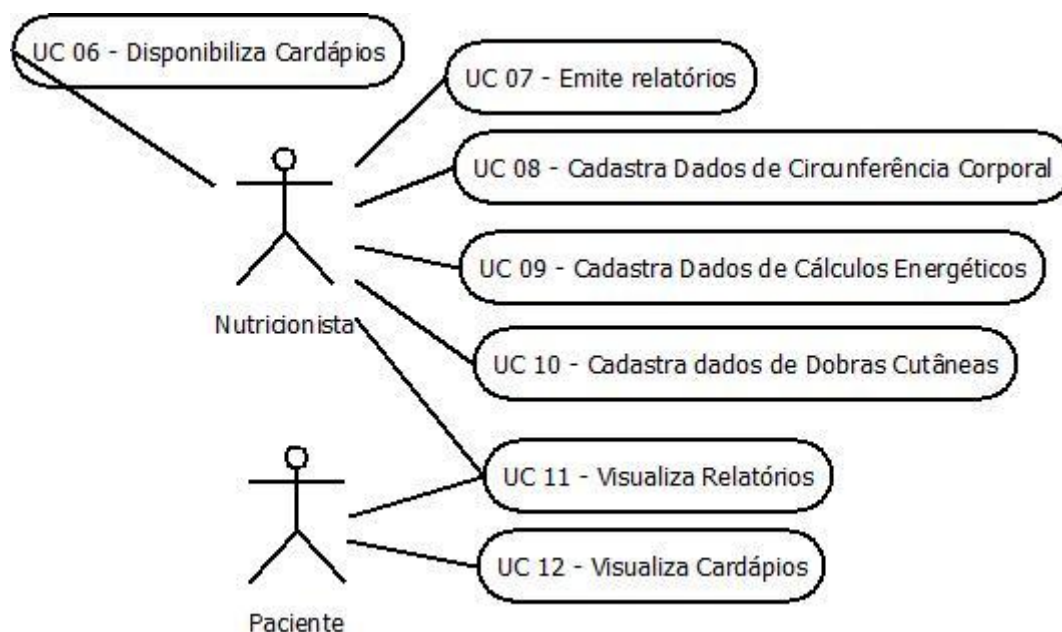


Figura 5 - Casos de uso nutricionista e paciente

Fonte: O autor(2017)

O diagrama da Figura 5 visa deixar claro as principais ações do nutricionista em relação ao paciente e o que o paciente pode fazer. Abaixo, segue a descrição detalhada de cada caso de uso

- **UC06 – Disponibiliza cardápios**
 - **Ator:** Nutricionista
 - **Descrição:** Uma vez cadastrado, o nutricionista define se o cardápio está ou não disponível para o paciente.
- **UC07 – Emite relatórios**
 - **Ator:** Nutricionista
 - **Descrição:** O nutricionista é capaz de emitir e disponibilizar relatórios ao paciente.
- **UC08 – Cadastrar Dados de Circunferência Corporal**
 - **Ator:** Nutricionista
 - **Descrição:** O nutricionista deve cadastrar dados de circunferência corporal, como circunferência da cintura, abdômen, etc.
- **UC09 – Cadastrar Dados de Cálculos Energéticos**
 - **Ator:** Nutricionista
 - **Descrição:** O nutricionista deve cadastrar dados de cálculos energéticos, como peso, altura, massa livre, etc.
- **UC10 – Cadastrar dados de Dobras Cutâneas**
 - **Ator:** Nutricionista

- **Descrição:** O nutricionista deve cadastrar dados de dobras cutâneas, como dobra tricipital, abdominal, etc.
- **UC11 – Visualiza Relatórios**
 - **Ator:** Paciente
 - **Descrição:** O paciente deve ser capaz de visualizar relatórios emitidos pelo nutricionista.
- **UC12 – Visualiza Cardápios**
 - **Ator:** Paciente
 - **Descrição:** O paciente deve ser capaz de visualizar relatórios emitidos pelo nutricionista.

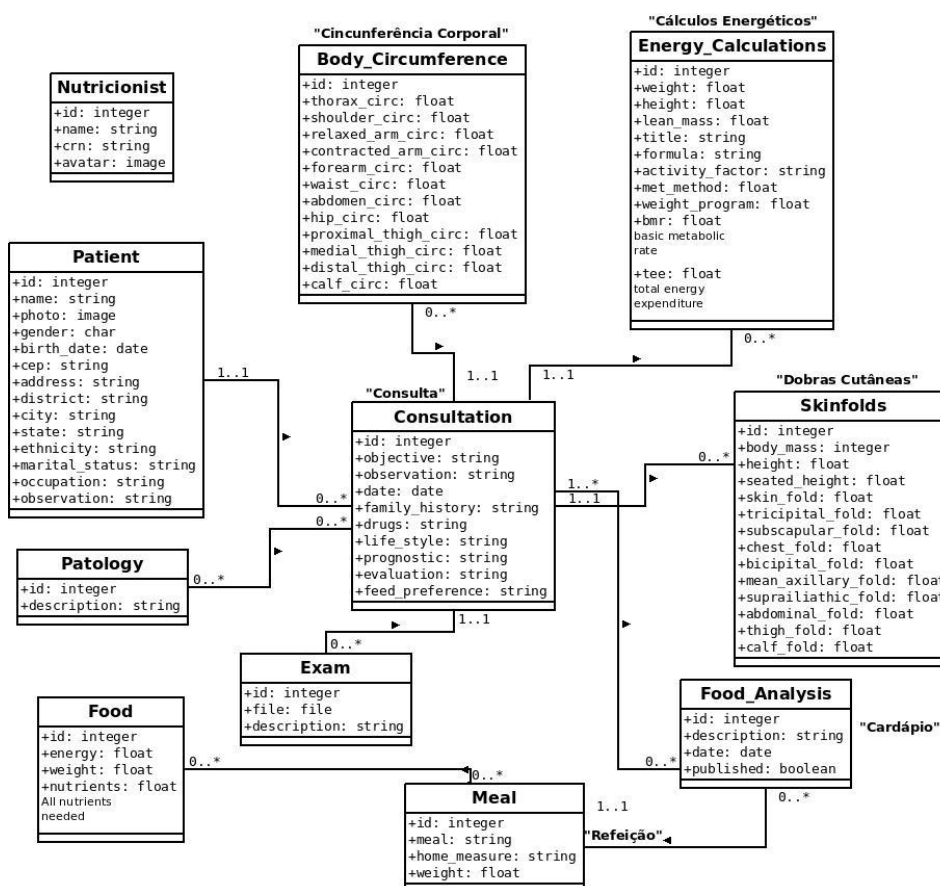


Figura 6 - Diagrama de classes entrada

Fonte: O autor(2017)

O nutricionista cadastra novos pacientes. O paciente possui dados como dobras cutâneas, circunferência corporal e dados para cálculos energéticos. O paciente também pode possuir patologias, onde estas podem interferir na sua alimentação. Também poderá ser cadastrado exames feitos sobre um paciente. Dados como etnia, profissão podem influenciar no fator de atividade física e portanto, enfatiza-se a presença destes no diagrama da Figura 6.

Cada alimento, possui micro e macro nutrientes como energia, carboidratos, lipídios, etc. Estes devem ser previamente cadastrados antes de prescrever o cardápio.

Durante o cadastro de um novo cardápio, o nutricionista seleciona o alimento, o horário em que será consumido, a quantidade normal, a medida caseira e se este será publicado ou não, podendo ser editado posteriormente.

4.2 Resultados

Com base no que foi apresentado no sub tópico 4.1, foi desenvolvido o sistema. No decorrer deste tópico será apresentado as principais telas do sistema, explicando suas principais funcionalidades. As Figuras 7 e 8 ilustram os menus com todas as funcionalidades disponíveis no sistema.

**Figura 7 - Menu do
Nutricionista**

Fonte: O autor(2017)

Figura 8 – Menu do Paciente

Fonte: O autor(2017)

A Figura 7 mostra todas as funcionalidades do Nutricionista. O Nutricionista ainda pode emitir relatórios a partir da lista de pacientes acessível no menu. Cada funcionalidade dessa fica disponível a partir do momento que o Nutricionista realiza Login no sistema. A Figura 8 mostra o menu do Paciente. O paciente pode obter dados do relatório ou do cardápio apenas digitando seu ID do sistema.

O *Django* trata as requisições através de suas URL's. A Figura 9 mostra a lista de usuários.

Figura 9 - Lista de usuários cadastrados

Fonte: O autor(2017)

A Figura 10 mostra a tela de cadastro de paciente, onde é possível cadastrar seus dados pessoais e os demais dados para fins de acompanhamento do paciente.

Figura 10 - Tela de cadastro de pacientes.

Fonte: O autor (2017)

As Figuras 11 e 12 mostram as telas de cadastro de cardápio e consulta, respectivamente. É durante a consulta que o nutricionista cadastra os dados de acompanhamento do paciente. O cardápio só poderá ser cadastrado se a consulta tiver sido cadastrada previamente.

Figura 11 - Tela de cadastro de cardápio

Fonte: O autor (2017)

O cadastro de cardápio consta dados do paciente, como nome completo e data em que a consulta foi realizada. Em seguida, a descrição do cardápio, disponibilidade

e os alimentos a serem cadastrados, que devem ter horário de consumo, quantidade e medida caseira.

Figura 12 - Tela de cadastro de consultas

Fonte: o autor(2017)

No cadastro de consultas, o nutricionista cadastra informações de histórico familiar do paciente, patologias, fármacos que utiliza, além de dados de acompanhamento como dobras cutâneas, circunferência corporal, entre outros.

Figura 5 – Relatório do Paciente

Fonte: O autor (2017)

A Figura 13 mostra o relatório do paciente, onde é possível ver todos os dados durante as consultas que foram realizadas e os dados de acompanhamento que foram obtidos em cada uma delas.

Figura 14 – Cardápio do Paciente

Fonte: O autor (2017)

A Figura 14 mostra o cardápio com a descrição e todas as suas informações nutricionais (macro e micro nutrientes), além do nome do paciente para quem foi prescrito.

Todas as funcionalidades aqui apresentadas, atualmente, encontram-se implementadas, onde os usuários do sistema começam a receber treinamento a fim de entender como ele funciona.

5 CONCLUSÃO E TRABALHOS FUTUROS

Dentre as pessoas pesquisadas, constatou-se que nenhuma delas tinham interesse em pagar pelas ferramentas que estavam disponíveis no mercado. Por esse motivo, pretendeu-se desenvolver uma ferramenta simples e que atendesse as principais necessidades relatadas por estes, ao tempo que tivesse um baixo custo financeiro para as mesmas.

Para solucionar tais problemas, desenvolveu-se um sistema web com as principais funcionalidades elencadas nas histórias de usuário, possibilitando assim que qualquer usuário (nutricionista ou paciente) pudesse acessá-lo de qualquer dispositivo com acesso à internet.

Antes de começar o desenvolvimento, estudou-se o problema e entendeu-se quais as principais necessidades: o trabalho repetitivo e cansativo de calcular as informações nutricionais de um cardápio, a falta de uma agenda que interaja com os dados de acompanhamento nutricional. Com base nisso, foi feito o levantamento de requisitos através das histórias de usuário para só então passar para a fase de desenvolvimento.

Como trabalhos futuros, planeja-se ampliar o sistema de modo que o paciente possa interagir mais com o nutricionista e posteriormente, disponibilizar um aplicativo móvel, para que o paciente possa receber notificações enviadas pelo seu próprio nutricionista, facilitando e melhorando o acompanhamento do mesmo.

REFERENCIAS

- BORGES, L. E. Python para desenvolvedores. Rio de Janeiro: 2ª edição, 2010, disponível em <https://ark4n.files.wordpress.com/2010/01/python_para_desenvolvedores_2ed.pdf> , Acesso em 05 de novembro de 2017.
- BRASIL. Ministério da Saúde. Secretaria de Atenção à Saúde. Departamento de Atenção Básica. Guia alimentar para a População Brasileira. 2. ed. Brasília: Ministério da Saúde, 2014.
- DIAKOPOULOS, N.;CASS, S.. Interactive: The Top Programming Languages 2017. IEEE Spectrum. 2017. Disponível em < <https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2017>>, acessado em: 05 de novembro de 2017.
- FOLLE, P. A. Editando áreas georreferenciadas no Google Maps. Universidade Federal do Rio Grande do Sul. Porto Alegre, 2012.
- HARRIS, J.A.; BENEDICT, F.G. A Biometric Study of Basal Metabolism in man. Boston: Carnegie Institution of Washington, 1919.
- HILL, J.O. Understanding and addressing the epidemic of obesity: an energy balance perspective. Endocr Rev. 2006;27(7):750-61.
- MOTTIN, L. R.; COSTA, J. F.; ANIBOLETE, R. C. Troca-troca de roupas. Universidade Federal do Paraná. Curitiba, 2014.
- LEMONS, M. F.; OLIVEIRA, P. C.; RUELA, L. C.; SANTOS, M. S.; SILVEIRA, T. C; REIS, J. C. S. Aplicabilidade da arquitetura MVC em uma aplicação Web (WebApps).UNIFENAS. Alfenas. 2013.
- MACHADO, M.; MEDINA, S. G. SCRUM – Método Ágil: uma mudança cultural na Gestão de Projetos de Desenvolvimento de Software. Revista Científica Intr@ciência. 2009.
- NETO, O. S. Desenvolvimento Web com Python & Django. Triveos, 2010.
- NIEDERAUER, J. Desenvolvendo web sites com PHP. 3ª Ed. Novatec, São Paulo, Brasil, 2017.
- PEDROSA R.G.; DONATO J. J.; ARAÚJO J. J.A.; TIRAPEGUI J. Gasto energético: componentes, fatores determinantes e mensura- ção. In: Angelis RC, Tirapegui J, organizador. Fisiologia da nutrição humana: aspectos básicos, aplicados e funcionais. São Paulo: Atheneu; 2007.
- PROENÇA, R.P.C. Alimentação e globalização: algumas reflexões. Cienc. Cult. vol.62 no.4 São Paulo Oct. 2010.
- ROHDEN, R. B. A importância da Engenharia de Software na Computação nos dias atuais. Universidade Regional do Noroeste do Estado do Rio Grande do Sul. Ijuí. Disponível em: <http://www.dai.ifma.edu.br/~karladf/Disciplinas/EngenhariaSoftware_SI/Artigo_Importancia_ES_UNIJUI.pdf>. Acesso em: 05 de novembro de 2017.

RUIZ JR, ORTEGA B. O.; RODRIGUEZ G.; ALKORTA P.; LAMBAYAN I. Validity of resting energy expenditure predictive equations before energy-restricted diet intervention in obese women. PLoS ONE 2011; 9:1-11.

SCHUSTER, J.; OLIVEIRA, A. M.; BOSCO, S. M. D. O papel da nutrição na prevenção e no tratamento de doenças cardiovasculares e metabólicas. Revista da Sociedade de Cardiologia do Estado do Rio Grande do Sul, Porto Alegre, v.1, n.28, p.1-6, 2015.

VARASCHIM, J. D. Implantando o SCRUM em um Ambiente de Desenvolvimento de Produtos para Internet. Pontifícia Universidade Católica Do Rio De Janeiro. Rio de Janeiro, 2009.

VIEIRA, F. G. P. F. P.; FEIO, L. C.; ASSIS, M. A. A.; PERES, M. A.; VASCONCELOS, F. A. G. Comparison of nutritional values of ten menus based on four computer programs. Rev. Nutr. vol.22 no.1 Campinas Jan./Feb. 2009.

W3TECHS. Usage of JavaScript libraries for websites. Novembro de 2017. Disponível em: < https://w3techs.com/technologies/overview/javascript_library/all>. Acesso em: 06 de novembro de 2017.

WEIJS P.J.M.; KRUIZENGA H.M.; DIJK A.E.; BARBARA S.; MEIJ B.S.; LANGIUS, J.A.E. et al. Validation of predictive equations for resting energy expenditure in adult outpatients and inpatients. Clin Nutr 2008; 27:150-7.