

UNIVERSIDADE FEDERAL DOS VALES DO JEQUITINHONHA E MUCURI
FACULDADE DE CIÊNCIAS EXATAS E TECNOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

**DESENVOLVIMENTO DE UM SISTEMA WEB PARA GERENCIAMENTO DE
CLIENTES DA ACADEMIA STUDIO FITNESS DE DIAMANTINA/MG**

Rodolpho Freitas Melo Ferreira

Diamantina
2016

UNIVERSIDADE FEDERAL DOS VALES DO JEQUITINHONHA E MUCURI
FACULDADE DE CIÊNCIAS EXATAS E TECNOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

**DESENVOLVIMENTO DE UM SISTEMA WEB PARA GERENCIAMENTO DE
CLIENTES DA ACADEMIA STUDIO FITNESS DE DIAMANTINA/MG**

Rodolpho Freitas Melo Ferreira

Orientador:

Luiz Filipe Carreiro Salazar

Trabalho de Conclusão de Curso
apresentado ao Curso de Sistemas de
Informação, como parte dos requisitos
exigidos para a conclusão do curso.

Diamantina

2016

Dedico esse trabalho a meus pais e minha família

AGRADECIMENTOS

Agradeço primeiramente a Deus.

Agradeço meus pais e minha família pelo apoio em toda essa caminhada.

Agradeço meus companheiros de república.

Agradeço aos amigos que me auxiliaram nessa jornada.

Agradeço ao Luiz Filipe Carreiro Salazar, meu orientador, pela ajuda e orientação na conclusão do trabalho.

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema web para gerenciar os clientes da academia Studio Fitness localizada em Diamantina, Minas Gerais, utilizando a metodologia de desenvolvimento Scrum. A academia possui como forma de armazenamento de dados, fichas impressas pelos funcionários, com isso o processo de armazenamento não é feito de forma segura.

O objetivo é construir um sistema web para armazenar todos os dados importantes. O sistema também retorna para os usuários (instrutores, fisioterapeutas), a evolução de um determinado cliente ao praticar suas atividades físicas. Uma ferramenta auxiliar, a agenda para marcação de atendimentos. O sistema também conta com o cadastro de movimentos diários, relacionados com a parte financeira.

Foram utilizados conceitos sobre engenharia web e de software, e ferramentas para o auxílio no desenvolvimento, como linguagens PHP, HTML, Javascript, banco de dados em MySQL. Após o desenvolvimento do sistema os testes foram executados e o sistema implantado na academia.

Palavras-chave: Scrum, desenvolvimento, sistema web

ABSTRACT

This paper presents the development of a web system to manage the costumers in Studio Fitness, which is a gym, located in Diamantina, Minas Gerais, using a development methodology Scrum. The gym uses sheets printed by employees as their unique form of data storage, spending too much physical space and resources for the storage process. The goal is build a web system to store all the important data. The system also features an auxiliary tool, the agenda for dialing calls. Concepts have been used on web and software engineering, and tools to aid in development, such as PHP, HTML, Javascript and MySQL database. When the development was conclude, tests were performed and the system deployed in the gym.

Keywords: Scrum , development , web system

LISTA DE FIGURAS

Figura 1 - Funcionamento do Scrum.....	16
Figura 2 – Burn Down Chart.....	17
Figura 3 – Tela de configuração do XAMPP.....	19
Figura 4 - Caso de uso.....	25
Figura 5 - Modelagem do banco de dados.....	31
Figura 6 – Burndown chart Sprint 1.....	32
Figura 7 – Tela da lista de todos administradores.....	34
Figura 8 – Tela de cadastro de administradores.....	35
Figura 9 – Tela de edição de administradores.....	36
Figura 10 – Mensagem de alerta de exclusão de administradores.....	37
Figura 11 – Burndown chart da Sprint 2.....	38
Figura 12 – Tela de login.....	41
Figura 13 – Lista de todos clientes.....	42
Figura 14 – Tela de cadastro de clientes.....	43
Figura 15 – Tela de edição de clientes.....	44
Figura 16 – Mensagem de alerta de exclusão de clientes.....	44
Figura 17 – Lista de todos instrutores.....	46
Figura 18 – Tela de cadastro de instrutores.....	47
Figura 19 – Tela de edição de instrutores.....	48
Figura 20 – Mensagem de alerta de exclusão de instrutores.....	49
Figura 21 – Burndown chart Sprint 3.....	50
Figura 22 – Lista de todas avaliações.....	52
Figura 23 – Primeira parte do cadastro de avaliações.....	53
Figura 24 – Segunda parte do cadastro de avaliações.....	54
Figura 25 – Primeira parte do cadastro de avaliações.....	55
Figura 26 – Terceira parte do cadastro de avaliações.....	56
Figura 27 – Quarta parte do cadastro de avaliações.....	57
Figura 28 – Quinta parte do cadastro de avaliações.....	58
Figura 29 – Sexta parte do cadastro de avaliações.....	59
Figura 30 – Sétima parte do cadastro de avaliações.....	60
Figura 31 – Burndown chart da sprint 4.....	61
Figura 32 – Burndown chart da Sprint 5.....	65

Figura 33 – Burndown chart da Sprint 6.....	67
Figura 34 – Tela home com a agenda de atendimento.....	70
Figura 35 – Burndown chart da Sprint 7.....	71
Figura 36 – Tela de movimentos diários.....	73
Figura 37 – Burndown chart da Sprint 8.....	74
Figura 38 – Burndown chart da Sprint 9.....	76

LISTA DE TABELAS

Tabela 1 – Product backlog.....	26
Tabela 2 – Product backlog com os custos.....	28
Tabela 3 – Sprint backlog da Sprint 1.....	29
Tabela 4 – Sprint backlog da Sprint 2.	33
Tabela 5 – Sprint backlog da Sprint 3.....	39
Tabela 6 – Sprint backlog da Sprint 4.....	51
Tabela 7 - Sprint backlog da Sprint 5.....	63
Tablela 8 – Sprint backlog da Sprint 6.....	66
Tabela 9 – Sprint backlog da Sprint 7.....	69
Tabela 10 – Sprint backlod da Sprint 8.....	72
Tabela 11 – Sprint backlog da Sprint 9.....	75
Tabela 12 – Caso de teste do banco de dados.	79
Tabela 13 – caso de teste de funcionalidade.	80
Tabela 14 – Caso de teste de configuração.....	81

LISTA DE ABREVIATURAS E SIGLAS

CSS – Cascading Style Sheet

GNU - General Public License

HTTP - HyperText Transfer Protocol

PHP - Hypertext Preprocessor

SUMÁRIO

1	INTRODUÇÃO	
2	REFERENCIAL TEÓRICO.....	11
2.1	Engenharia de Software voltada para web.....	11
2.2	Sistemas de informações gerenciais.....	12
2.3	Interação Humano Computador.....	12
2.4	Indicadores de desempenho.....	13
3	METODOLOGIA.....	14
3.1	Scrum.....	14
3.1	Funcionamento do Scrum.....	14
3.1.2	Membros e responsabilidades.....	16
3.2	Ferramentas utilizadas.....	17
3.2.1	XAMPP.....	17
3.2.2	Apache.....	18
3.2.3	3.3. 3 HTML.....	18
3.2.4	PHP.....	19
3.2.5	MySQL.....	20
3.2.6	CSS.....	21
3.2.7	Adobe Dreamweaver CS6.....	22
3.2.8	Adobe Fireworks CS6.....	22
3.2.9	Navicat.....	22
3.2.10	Javascript.....	22
4	RESULTADOS.....	23
4.1	Desenvolvimento e especificação do sistema.....	23
4.2	Sprint 1.....	27
4.3	Sprint 2.....	30
4.4	Sprint 3.....	36
4.5	Sprint 4.....	47
4.6	Sprint 5.....	58

4.7 Sprint 6.....	62
4.8 Sprint 7.....	65
4.9 Sprint 8.....	68
4.10 Sprint 9.....	71
5 VALIDAÇÃO E TESTES.....	74
5.1 Tipos de teste.....	75
5.2 Casos de teste.....	76
6 IMPLANTAÇÃO.....	79
7 CONCLUSÃO.....	79
8 REFERÊNCIAS.....	81

1 INTRODUÇÃO

Na academia Studio Fitness – Diamantina, Minas Gerais, há uma grande quantidade de clientes que fazem uso de seus serviços como musculação, pilates, fisioterapia. Na academia os registros dos clientes são feitos a partir de fichas de papel que contém as atividades que irão ser executadas durante os dias que o cliente escolher. O registro das atividades é feito pelos educadores físicos, no caso de musculação e os fisioterapeutas formulam as atividades do pilates, tudo registrado em fichas que são mantidas em um arquivo. Portanto, o cadastro dos clientes feito em um Sistema Web, acessado por um navegador que necessita de simples recursos de software e hardware, poderia facilitar e informatizar o processo de cadastro, consulta e gestão dos clientes da academia.

Este trabalho de conclusão de curso é também um projeto com foco no desenvolvimento de um sistema web que abrange todos os clientes e profissionais da academia Studio Fitness. Ao se inserir os dados no sistema, é possível condensar os registros em informações e analisar o desempenho físico dos clientes por meio de indicadores. Estes deverão auxiliar o processo de tomada de decisão dos profissionais da academia, permitindo uma melhor avaliação dos resultados obtidos pelos clientes auxiliaram os profissionais

Atualmente a informatização de serviços é uma ferramenta que auxilia muito as empresas, proporcionando resultados rápidos, eficientes e também satisfação pessoal. Segundo Oliveira (2004), um sistema de informação gerencial pode, em determinadas condições, trazer benefícios para as empresas, tais como, redução dos custos das operações e centralização de decisões, melhoria na produtividade, no acesso às informações, na tomada de decisões, na estrutura organizacional, nos serviços oferecidos e realizados, adaptação da empresa para superar acontecimentos inesperados, com isso as empresas tornam o processo de gestão mais eficiente, garantindo o diferencial de atuação e subsequentemente, vantagem competitiva.

Ao utilizar um sistema web, o processamento e o armazenamento dos dados serão feitos de forma rápida e eficiente, tendo como vantagem o acesso das informações de diferentes locais, em qualquer horário que desejar e realizando suas

tarefas a um custo reduzido. A proposta do projeto é facilitar a entrada e armazenamento dos dados dos clientes da academia e organizar as informações através de um sistema web.

O objetivo geral do projeto é: Desenvolver e implementar/implantar um sistema Web de gerenciamento de clientes para a empresa

E os objetivos específicos:

- Utilização de linguagens/ferramentas de desenvolvimento web como, PHP para as funções do sistema, HTML para estruturação das páginas e CSS para estilização;
- Implantar um banco de dados para armazenar as informações pertinentes;
- Na fase de desenvolvimento, utilizar um método ágil de desenvolvimento de software, de acordo com o sistema proposto;
- Realizar testes e avaliação da ferramenta, acompanhando a interação que o sistema promover aos usuários.

2 REFERENCIAL TEÓRICO

1.1 Engenharia de software voltada para web

O avançado tecnológico e de multimídias abriram espaço para um novo meio de comunicação, a internet. Segundo Castells (2001), a internet é um meio de comunicação que permite, pela primeira vez, a comunicação de muitos com muitos, num momento escolhido, em escala global. Com isso, as aplicações web suportam um conjunto complexo de funcionalidades e informações, são desenvolvidas em várias linguagens e estilos, fazem reuso de diversos componentes, realizam interações com os usuários, com bases de dados e diferentes aplicações (OFFUTT, 2002). De acordo com Pressman (2002) ao desenvolver uma aplicação web devem ser considerados alguns critérios:

- **Imediatismo:** nas aplicações web têm-se a necessidade de acessos de vários usuários simultaneamente, sobrecarregando o tráfego de informação. Com isso, o desenvolvedor deve se preocupar e utilizar métodos de análise, implementação, planejamento para que essas aplicações gerem resultados rápidos.
- **Segurança:** as aplicações estão em rede, com isso existe a dificuldade de controlar o número de usuários que farão acessos das aplicações. Portanto o desenvolvimento precisa garantir a segurança e a integridade dos dados, limitando o acesso a determinados grupos de usuários e, conceder privilégios de acesso aos outros.
- **Estética:** não tão menos importante que o desenvolvimento, a estética das aplicações web são de suma importância. Uma boa interface torna o acesso mais confortável e promove uma melhor compreensão. Apresentar os dados de forma objetiva e organizada, a fim de induzir o usuário a adquirir determinado serviço ou produto.

1.2 Sistemas de Informações Gerenciais

Atualmente no cenário competitivo, as empresas necessitam utilizar ferramentas de otimização, gerenciando as informações em bases de dados para, conseguir resultados rápidos e satisfatórios. Essas ferramentas de otimização

também ajudam na tomada de decisões com base nas informações gerenciadas, diminuindo o tempo de resposta da organização com o mercado. Uma das ferramentas que cumprem essas exigências são os sistemas de informações gerenciais (SIG), são de suma importância para sobrevivência das empresas no cenário mundial dos negócios. Resumindo, os sistemas de informações gerenciais são conjuntos de dados brutos que são transformados em informações estruturadas e organizadas auxiliando no processo de decisão e aprimoração de resultados. Segundo Oliveira (2004) os sistemas de informações gerenciais são indispensáveis na interpretação dos dados para tomada de decisões, auxiliando na interpretação dos dados sem esperar por conversões demoradas de sistemas informatizados.

1.3 Interação Humano Computador

A interação humano-computador é o conjunto de diálogos, processos e ações que o usuário humano usa para interagir com um computador. Segundo Carvalho (1994), o avanço das tecnologias fez do computador uma ferramenta cada vez mais indispensável na execução das atividades humanas, onde na maioria dos ambientes e situações do dia, há um computador presente, como por exemplo uma compra efetuada em uma loja ou supermercado, uso de eletrodomésticos, serviços bancários, dentre vários.

1.4 Indicadores de desempenho

Com os dados inseridos no sistema implementado, poderá avaliar com peculiaridade cada cliente cadastrado, de forma rápida, simples e com baixo custo de processamento. Avaliar é conhecer, é contrastar, é dialogar, é indagar, é argumentar, é deliberar, é raciocinar, é aprender. Em termos gerais, decididamente comprometido com a racionalidade prática e crítica, quem avalia quer conhecer, valorizar, sopesar, distinguir, discernir, definir o valor de uma ação humana, de uma atividade, de um processo, de um resultado. Avaliar é construir conhecimento por vias heurísticas de descoberta. Quem avalia com intenção formativa quer conhecer a qualidade dos processos e dos resultados. (MENDEZ, 2002). Um indicador é uma ferramenta que permite a obtenção de informações sobre uma dada realidade, tendo como característica principal a de poder sintetizar diversas informações, retendo apenas o significado essencial dos aspectos analisados (MITCHELL, 2004).

Os indicadores de desempenho são utilizados para alcançar as metas definidas pela empresa e indicam quais melhorias são necessárias para o melhor desempenho dos processos. Segundo Takashina e Flores (1995) um indicador deve ser gerado criteriosamente para assegurar a disponibilidade dos dados e resultados mais relevantes no menor tempo possível e com o menor custo. Para selecionar indicadores, segundo Kligerman et al. (2007), apresentam algumas métricas, como:

- Existência de dados base;
- Possibilidade de comparação com critérios/padrões/metapas existentes;
- Possibilidade de agilidade na atualização do indicador.

Segundo Trzesniak (1998), as vantagens dos indicadores de desempenho são: vantagem competitiva, fidelidade dos clientes, aumento da lucratividade e redução de custos, crescimento e desenvolvimento da empresa, satisfação do cliente com inovações nos produtos ou serviços que atendam suas necessidades e expectativas.

O sistema web implementado terá como principal indicador de desempenho o de qualidade. Os indicadores de qualidade são também conhecidos como indicadores da satisfação do cliente e usuários, medindo como o produto ou serviço é visto pelos clientes e a capacidade de atender seus requisitos.

3 METODOLOGIA

3.1 Scrum

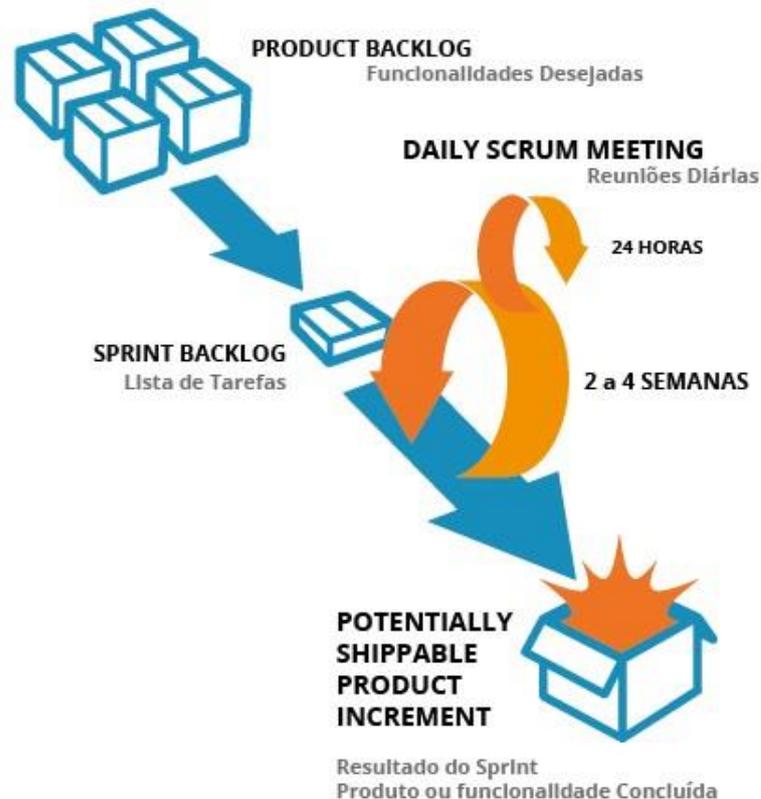
Das várias metodologias ágeis existentes, a Scrum foi escolhida como objeto de estudo principalmente por ser uma das mais aceitas, e por conter em suas práticas muito esforço de gerenciamento de projeto. A ferramenta Scrum é eficiente na gestão dos riscos, do tempo e dos recursos utilizados, dando agilidade no processo de desenvolvimento. O scrum é um método de desenvolvimento ágil bastante objetivo, com papéis bem definidos e simples adaptação. Segundo Schwaber (2004), o Scrum não é um processo previsível pois não delimita o que fazer em determinadas circunstâncias. No Scrum há um conjunto de práticas que

torna tudo mais visível, permitindo aos adeptos saber o que está acontecendo nas etapas do projeto, com isso, fazer os devidos ajustes visando cumprir os objetivos ao longo do tempo. O Scrum não indicará o que fazer quando ocorrer um problema mas os problemas serão facilmente identificados.

3.1.2 Funcionamento do Scrum

No método Scrum os projetos são divididos em ciclos chamados de *Sprint*. O *Sprint* é um tempo, definido, para a execução de algumas atividades, geralmente duram de 2 a 4 semanas. O *Product Backlog* é uma lista que contém as funcionalidades que deverão ser desenvolvidas no projeto. Quando iniciada uma *Sprint* há uma reunião de planejamento ou *Sprint Planning Meeting* onde o *Product Owner* prioriza os itens do *Product Backlog* e a equipe escolhe as atividades que serão desenvolvidas durante o *Sprint*. Depois de alocar as atividades, essas são transferidas do *Product Backlog* para o *Sprint Backlog*.

Figura 1 – Funcionamento do Scrum.



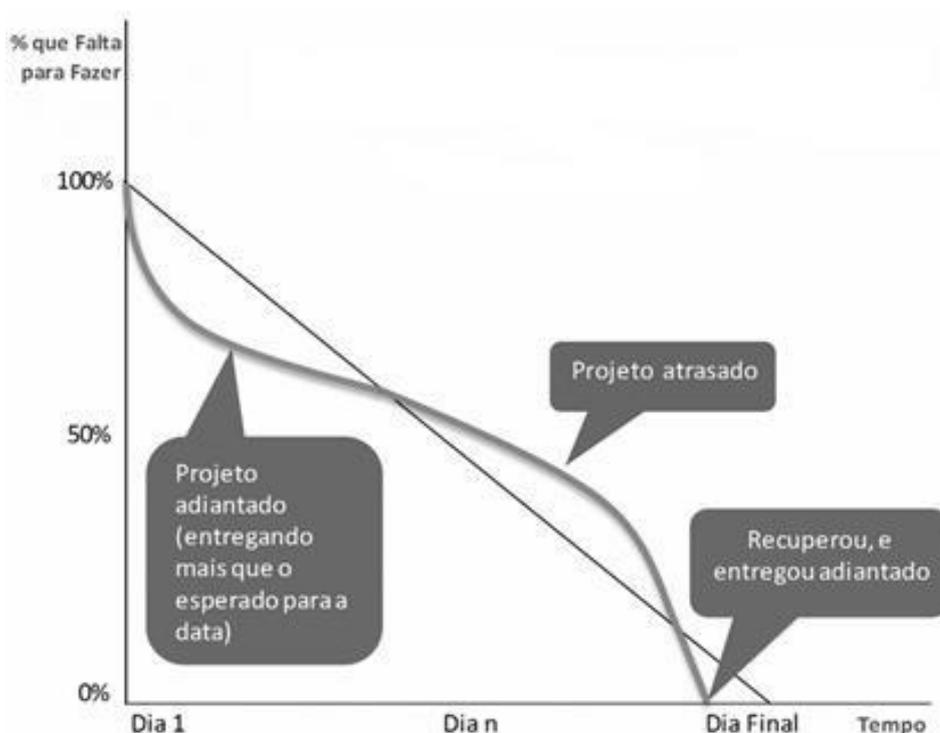
Retirada de: <http://www.brq.com/metodologias-ageis/>

A equipe possui um quadro de trabalho (chamado também de *Kanban*), usado para organizar as atividades do *Sprint Backlog*. Esse quadro é muito útil, pois auxilia a equipe no acompanhamento do progresso da *Sprint*. As etapas das atividades são separadas por estados: Para fazer, Em desenvolvimento, Testando e Finalizado.

Diariamente, a equipe se reúne para uma breve reunião, composta de todos os membros da equipe, chamada de *Daily Scrum* e tem como objetivo dar a cada participante o direito de falar o que ele fez no dia anterior, o que fará no dia corrente e se existe empecilhos para a execução das atividades. Ao término de uma *Sprint* a equipe apresenta as funcionalidades desenvolvidas em uma *Sprint Review Meeting* mostrando o que foi concluído. Com a *Sprint Retrospective* identifica-se o que está funcionando bem e quais melhorias devem ser feitas e o time planeja o início de um novo *Sprint*. O *Burn Down Chart* é um gráfico do trabalho restante em

comparação com o trabalho realizado. É composto de dois eixos (geralmente, o eixo vertical Y representa a quantidade de trabalho e no eixo horizontal X, representa-se o tempo) e é muito útil para alertar a equipe de atrasos no projeto, se as execuções das atividades estão adiantadas e prever quando a atividade será concluída. Uma linha é traçada com a representação da execução da atividade e uma curva representa o esforço realizado pela equipe e é esperado que a execução das tarefas leve a curva de início no eixo Y ao encontro do eixo X representando o término na execução das atividades. Abaixo um exemplo do *Burn Down Chart*:

Figura 2 – Burn Down Chart.



Fonte: <http://www.brq.com/metodologias-ageis/>

3.1.3 Membros e responsabilidades

No scrum, os membros têm responsabilidades e papéis, são eles os principais:

- *Product Owner*: é responsável na definição dos requisitos do produto/serviço, prioriza os requisitos de acordo com sua importância, pode mudar os requisitos e prioridades dos Sprints, aceita ou nega os resultados das Sprints

- *Scrum Master*: é responsável por garantir a produtividade da equipe, facilitando a colaboração entre áreas e funções, eliminando empecilhos e protegendo a equipe de interferências. Participa das reuniões diárias, planejamentos e revisões do Sprint, garantindo que as atividades estão sendo seguidas.
- *Scrum Team*: é composto pelos membros que desenvolverão as atividades propostas. Selecionam quais itens serão desenvolvidos primeiro, realizando o que for pertinente durante a Sprint.

3.2 Ferramentas Utilizadas

As ferramentas listadas abaixo foram escolhidas conforme experiências que tive em estágios, projeto e ao longo do curso de sistemas de informação na Universidade Federal dos Vales do Jequitinhonha e Mucuri (UFVJM) tendo como objetivo facilitar o desenvolvimento da aplicação web proposta para resolução do problema.

3.2.1 XAMPP

O XAMPP é uma multiplataforma de software livre, composto por uma base de dados MySQL, o servidor web Apache e interpretadores de linguagens de script como PHP, ele trabalha como um servidor web livre, destaca-se pela fácil utilização. A licença do programa é GNU – *General Public License* desenvolvido pela Apache Friends disponível para Microsoft Windows, Linux, MacOS. O XAMPP é instalado em poucos segundos, com algumas configurações específicas ao sistema em alguns de seus componentes, estará funcionando como servidor web. Atualizações regulares são feitas para incorporar as últimas versões dos seus componentes MySQL, PHP, Apache. Ele conta também com módulos, os mais utilizados são o phpMyAdmin e o OpenSSL. Permite ao programador de website testar o andamento de suas tarefas de implementação em seu computador sem necessitar de conexão com à internet.

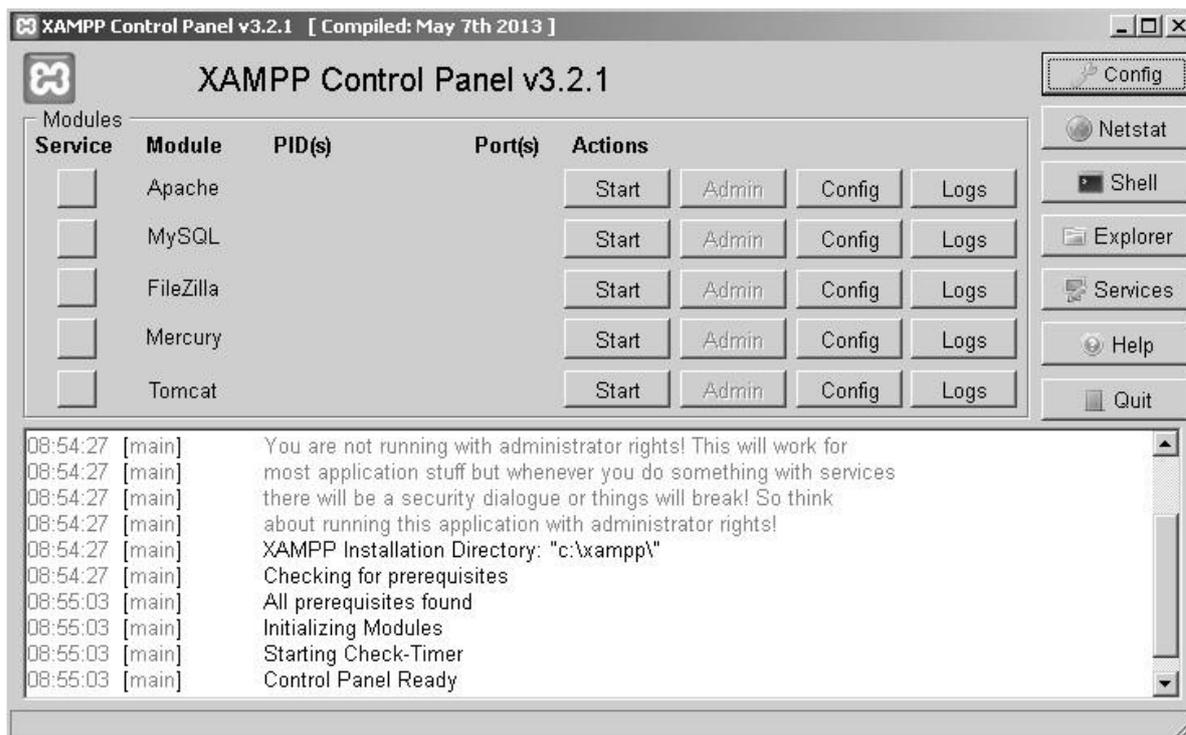


Figura 3 - Tela de configuração do XAMPP.

3.2.2 Apache

O Apache ou Apache Server é um servidor desenvolvido por Rob McCool quando ainda trabalhava na NCSA - *National Center of Supercomputing Applications*, tratasse de software livre, onde qualquer um pode estudar e modificar seu código. O Apache era o servidor Web mais usado no mundo, tendo um domínio de mais de 60% (ALECRIM, 2006). Optou-se por este servidor, o Apache por ser um servidor de performance excelente, compatível com várias plataformas, requer pouca capacidade de hardware para funcionar perfeitamente e por ser seguro. Ele é capaz de executar códigos na linguagem PHP em conjunto com o banco de dados MySQL e atua como servidor HTTP - *Hyper-Text Transfer Protocol* e FTP - *File Transfer Protocol*.

3.2.3 HTML

HTML – *HyperText Markup Language* ou Linguagem de Marcação de Texto é uma linguagem que define as páginas web na internet, a partir de um

conjunto de etiquetas chamadas de *tags* definem a forma que o texto e demais elementos da página serão apresentados. Uma linguagem de programação de fácil aprendizado, permitindo qualquer usuário criar uma página web. Para programar em HTML utilizasse um editor de texto, no arquivo são adicionadas as *tags* necessárias, como por exemplo, <P> significa um parágrafo, <A> é um link, negrito, após adicionadas as *tags* basta salvar o editor de texto como extensão .html ou .htm. Podemos resumir hipertexto de acordo com Silva (2008) como todo o conteúdo inserido em um documento para a web e que tem como principal característica a possibilidade de se interligar a outros documentos web.

3.2.4 PHP

O PHP – *Hypertext Preprocessor* é uma linguagem de programação que funciona do lado do servidor, ou seja, é executada no servidor web antes da página ser enviada ao cliente pela internet, podendo realizar acessos ao banco de dados, conexões em rede e outros procedimentos para criação da página final. O PHP é uma das linguagens mais utilizadas na web. Milhões de sites no mundo inteiro utilizam PHP. A principal diferença em relação às outras linguagens é a capacidade que o PHP tem de interagir com o mundo web, transformando totalmente os websites que possuem páginas estáticas (NIEDERAUER, 2011). Especialmente utilizada para o desenvolvimento web e contém HTML embutido em sua codificação. É delimitado pelas instruções de processamento *tags* de (início <?php) e (fim ?>). O código é executado no servidor, gerando o código HTML que finalmente é enviado para o cliente. A linguagem oferece muitos recursos para um programador avançado e é de fácil aprendizado para programadores iniciantes, permitindo escrever *scripts* em pouco tempo.

3.2.5 MySQL

Um dos sistemas mais utilizados, o MySQL é um sistema para gerenciamento de banco de dados, é desenvolvido, distribuído e tem suporte da MySQL AB. Possui licença GNU – *General Public License*, ou seja, qualquer usuário pode estudá-lo ou alterá-lo conforme sua necessidade. Essa ferramenta é otimizada para aplicações web e vastamente utilizada na internet. A maioria dos serviços de hospedagem de sites oferecem esse sistema, juntamente com a linguagem PHP,

onde ambos trabalham com uma eficaz sinergia. Outro ponto forte do MySQL é sua disponibilidade para a maioria dos sistemas operacionais, como Microsoft Windows, Linux e MacOS. O servidor de banco de dados MySQL é extremamente rápido, confiável, e fácil de usar. Apesar de estar em constante desenvolvimento, o Servidor MySQL oferece hoje um rico e proveitoso conjunto de funções. A conectividade, velocidade, e segurança fazem com que o MySQL seja altamente adaptável para acessar bancos de dados na Internet (SUEHRING, 2002).

3.2.6 CSS

CSS – *Cascading Style Sheet* traduzindo do termo em inglês temos folhas de estilo em cascata. Uma folha de estilo em cascata é um simples mecanismo que permite adicionar estilos, como por exemplo, cores, espaçamentos, fontes em documentos web. As CSS têm por finalidade devolver à marcação HTML/XML o propósito inicial da linguagem (SILVA, 2012). Usando a linguagem HTML para estruturação das páginas web, teremos apenas a marcação dos conteúdos da mesma, não fornecendo informações sobre o visual das apresentações dos elementos como tamanho das fontes, cores dos textos, posicionamento e todo a estilização da página, restando às CSS a função de apresentar visualmente os elementos presentes na página. CSS + HTML: “HTML para estruturar e CSS para apresentar.” (SILVA, 2012). Uma regra CSS pode conter inúmeras declarações separadas por ponto e vírgula. Veja o exemplo a seguir (A terminologia mostrada é a adotada pelo W3C):

```
p { color: blue;
background-
color:white;
font-size: 15px;
}
```

No exemplo temos os componentes a seguir da regra CSS:

- **Seletor:** é o elemento p. Seletores para elementos da marcação são denominados seletores tipos.
- **Declarações:** color: blue; background-color: white; font-size: 15px;

- **Propriedades:** color, que define a cor dos textos contidos em p, background-color, que define a cor de fundo do elemento p e font-size, que define o tamanho da fonte para o elemento p.

3.2.7 Adobe Dreamweaver CS6

O Adobe Dreamweaver é um software de desenvolvimento web desenvolvido pela Adobe Systems, Macromedia compatível com os sistemas operacionais Microsoft Windows e MacOS. Possui suporte para várias tecnologias web, como, HTML, CSS, PHP, JavaScript dentre outros. É uma poderosa ferramenta para solução profissional na criação de aplicativos e sites. Possui ferramentas para modelos visuais, funções de desenvolvimento de aplicações e também conta com suporte para edição de código oferecendo a qualquer programador criar sites com rapidez utilizando visuais incríveis. Tem como vantagem a combinação de design com desenvolvimento, permitindo ao usuário facilidade na implementação de suas aplicações. Sua interface ajuda muito na hora de trabalhar, simples, com ícones e demais componentes que são facilmente visualizados.

3.2.8 Adobe Fireworks CS6

O Fireworks é um editor de imagem e desenho vetorial desenvolvido pela Adobe, Macromedia compatível com os sistemas operacionais Microsoft Windows e MacOS. Tem como foco a publicação gráfica na web, possui suporte para formatos como .PNG e GIFS animados. Tem como vantagem uma interface limpa e clara de fácil utilização além de possuir uma ótima compreensão de imagens.

3.2.9 Navicat

O Navicat é uma ferramenta que executa múltiplas conexões com o banco de dados, agilizando a administração dos mesmos. Ele permite conectar a MySQL, PostgreSQL dentre outros, simultaneamente em um único programa. Escolhi esse software pelo motivo de usá-lo nos estágios que fiz no período de graduação, tendo assim uma experiência com a ferramenta.

3.3.10 Javascript

O Javascript é uma linguagem interpretada de programação, ou seja, o código fonte é executado por um interpretador em seguida executado no processador ou no sistema operacional. Surgiu em 1995 criada por Brendan Eich. A linguagem funciona a partir de scripts implementados, esses scripts são executados e controlados pelo navegador web utilizado pelo usuário, interagindo diretamente com o cliente, sem a necessidade de ser carregado do servidor. Tem comunicação sem sincronismo, alterando o conteúdo do documento exibido para o usuário final.

4 RESULTADOS

4.1 Desenvolvimento e especificação do sistema

Nesse capítulo, serão descritas as etapas seguidas de acordo com a metodologia do artefato escolhido, o Scrum. Primeiramente, foi feito o levantamento dos requisitos a partir de reuniões com o proprietário da academia. A academia é localizada na cidade de Diamantina, uma cidade do interior de Minas Gerais, é uma academia de médio porte que atende em média 100 alunos por dia. O sistema desenvolvido tem como foco os alunos que executam suas atividades e os profissionais da empresa, os instrutores e fisioterapeutas. Os profissionais da empresa serão beneficiados com o sistema no processo de tomada de decisão, a partir da avaliação do desempenho do cliente acompanhado pelo educador físico, e no processo de armazenamento dos dados. O cliente será beneficiado com um melhor acompanhamento de suas atividades pelos instrutores físicos ou fisioterapeutas e uma melhor disponibilização de seus dados para eventuais consultas. Em seguida iniciou-se o processo de implementação do sistema. As sprints guiaram as etapas de desenvolvimento, organizando e auxiliando, a alcançar os resultados e nos prazos estipulados. Na Figura 5 foram representados as funcionalidades e os atores (entidades externas que interagem com o sistema).

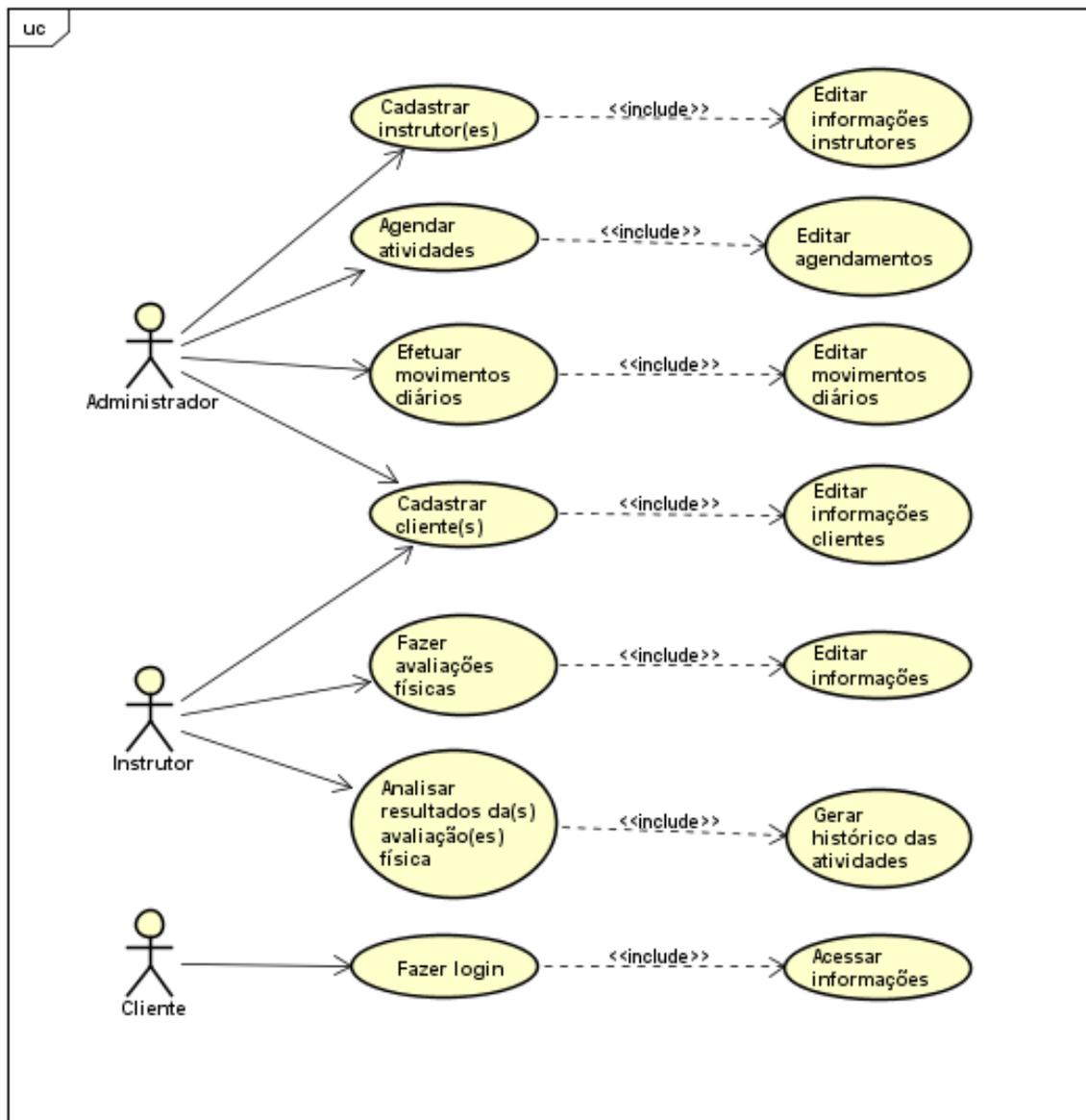


Figura 4 - Caso de uso.

Atores:

1- Administrador: é quem efetua e modifica os cadastros dos usuários do sistema (clientes e instrutores), agenda ou edita os horários marcados dos clientes, de acordo com suas atividades, efetua os movimentos diários (pagamentos e outras atividades dos clientes).

2- Instrutor: o instrutor efetua e modifica as avaliações físicas dos clientes, de acordo com suas atividades e objetivos (exemplo: pilates, musculação, perda de peso, condicionamento físico), analisa os resultados das

avaliações físicas efetuadas e ou gera histórico das avaliações de determinado cliente.

3- Cliente: os clientes só podem logar no sistema e consultar suas informações (ficha de atividades, dados pessoais).

Product Backlog

O *Product Backlog* lista as funcionalidades desejadas do sistema, de acordo com os requisitos levantados, a Tabela 1 mostra as prioridades das funcionalidades a serem implementadas.

Funcionalidades	Prioridades
Modelagem do banco de dados	1
Cadastro de administradores	1
Cadastro de instrutores e clientes	2
Cadastro de avaliações físicas	2
Análise dos resultados das avaliações físicas	3
Módulo que mostra para o usuário os resultados das análises das avaliações físicas	3
Agendamento de atendimento	4
Efetuar movimentos diários	4
Design das páginas do sistema	5

Tabela 1 – *Product backlog*.

Reunião para planejamento das *sprints*

Após construir o *Product Backlog*, o planejamento das tarefas foi feito, definindo a quantidade de tempo (em dias) que cada funcionalidade deve gastar. Levando em conta aspectos técnicos e funcionais a Tabela 2 mostra as estimativas em custo/dias.

Funcionalidades	Prioridades	Custo/dias
Modelagem do banco de dados	1	5
Cadastro de administradores	1	10
Cadastro de instrutores e clientes	2	10
Cadastro de avaliações físicas	2	10
Módulo de análise dos resultados das avaliações físicas	3	10
Módulo que mostra para o usuário os resultados das análises das avaliações físicas	3	10
Agendamento de atendimento	4	10
Efetuar movimentos diários	4	10
Design das páginas do sistema	5	10

Tabela 2 – *product backlog* com os custos.

Com os custos de cada funcionalidade definidos no *Product Backlog*, conseguiu-se definir a meta da primeira *Sprint*. A primeira *Sprint* é a modelagem do banco de dados.

1.2 Sprint 1

Modelagem do banco de dados:

Com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 3.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Modelagem banco de dados	1	5
Sub tarefa 1	Criação das classes do banco de dados	1.1	2
Sub tarefa 2	Criação das tabelas do banco de dados	1.2	2
Sub tarefa 3	Revisão e criação das chaves estrangeiras, relacionamentos das tabelas do banco de dados	1.3	1

Tabela 3 – Sprint backlog da Sprint 1.

Início da Sprint: com o *Sprint Backlog* pronto dá-se início a *Sprint*, visando alcançar as metas criadas seguindo o planejamento traçado. A primeira *Sprint* é a modelagem do banco de dados, base que guardará os dados relevantes para funcionamento do sistema, criando as classes, tabelas, chaves estrangeiras e relacionamentos de acordo com os requisitos coletados.

Reuniões diárias: com as reuniões diárias acompanhou-se que as sub tarefas foram executadas no custo planejado, cumprindo com eficiência a meta da *Sprint*.

Revisão da Sprint: no final do prazo estipulado, verificou que a Sprint foi finalizada com sucesso, cumprindo os requisitos e os custos previstos. Então o *Sprint Backlog* foi concluído com sucesso composto pelas sub tarefas: criação das classes do banco de dados, criação das tabelas do banco de dados, revisão e criação das chaves estrangeiras, relacionamentos das tabelas do banco de dados. A Figura 6 mostra o modelo do banco de dados e seus componentes, como resultado da primeira Sprint.



Figura 5 - modelagem do banco de dados.

O Burndown Chart é visto na seção 3.1.1 Funcionamentos do scrum, é um gráfico do trabalho restante em comparação com o trabalho realizado. É composto de dois eixos (geralmente, o eixo vertical Y representa a quantidade de trabalho e no eixo horizontal X, representa-se o tempo) e é muito útil para alertar a equipe de

atrasos no projeto, se a execução das atividades está adiantada e prever quando a atividade será concluída. Assim podemos acompanhar o progresso da Sprint, a Figura 7 mostra o gráfico dessa primeira Sprint.

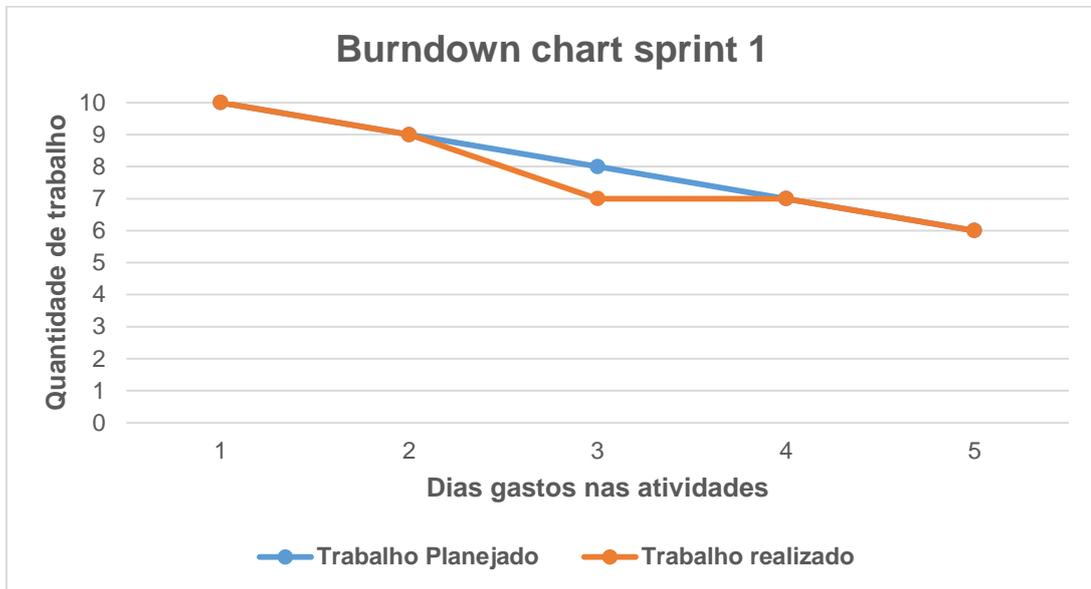


Figura 6 – Burndown chart Sprint 1.

No gráfico podemos ver experiência do desenvolvedor para modelar o banco de dados do sistema, dos dias 2 a 4 as atividades foram finalizadas antes do tempo previsto, conseguindo cumprir as atividades nos prazos estipulados para finalizar as tarefas da Sprint. Após concluir a primeira Sprint, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na Sprint 2: Cadastro de administradores

4.3 Sprint 2

Cadastro de administradores

Com o *Product Backlog* é definida a meta que deverá ser alcançada com a segunda Sprint, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 4.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Cadastro de administradores	1	10
Sub tarefa 1	Implementação das classes	1.1	2
Sub tarefa 2	Implementação dos controlers e viewers	1.2	3
Sub tarefa 3	Implementação das funções (salvar, editar e excluir)	1.3	3
Sub tarefa 4	Desenvolvimento das funções de controle do banco de dados	1.4	2

Tabela 4 – *Sprint backlog da Sprint 2.*

Início da Sprint: O objetivo da segunda *Sprint* foi implementar a página de cadastro de administradores e suas funcionalidades.

Reuniões diárias: nas reuniões foi visto que no início da *Sprint* as atividades foram executadas nos prazos estipulados e no decorrer das demais atividades, de implementação, por exemplo, algumas dificuldades apareceram atrasando um pouco a realização das tarefas. Os problemas foram resolvidos com

alguns dias e as tarefas foram desenvolvidas normalmente afim de finalizar a segunda *Sprint*, que foi finalizada como esperado.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada como esperado, cumprindo os requisitos e os custos previstos. Foi realizado uma revisão de tudo que foi feito, para evitar futuros problemas. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Implementação das classes, Implementação dos *Controllers* e *Viewers*, Implementação das funções (salvar, editar e excluir), Desenvolvimento das interações com o banco de dados. As Figuras 8 e 9 ilustram o que foi produzido na terceira *Sprint*.

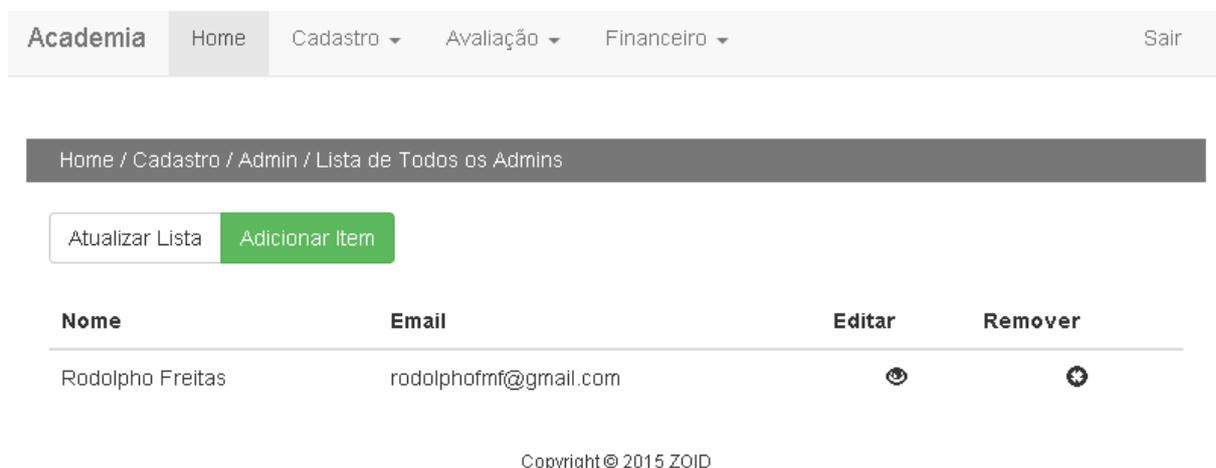


Figura 7 – tela da lista de todos administradores.

The image shows a web application interface for administrator registration. At the top, there is a navigation bar with the logo 'Academia' and menu items: 'Home', 'Cadastro' (with a dropdown arrow), 'Avaliação' (with a dropdown arrow), 'Financeiro' (with a dropdown arrow), and 'Sair'. Below the navigation bar is a breadcrumb trail: 'Home / Cadastro / Admin / Cadastrar Admins'. The main content area contains three input fields: 'Nome Completo *' with the placeholder text 'Nome Completo', 'Senha *' with the placeholder text 'Senha', and 'Email *' with the placeholder text 'Email'. Above the first two fields are two buttons: 'Voltar' (orange) and 'Salvar' (green). At the bottom center, there is a copyright notice: 'Copyright © 2015 ZOID'.

Figura 8 – tela de cadastro de administradores.

Nas Figuras 8 e 9 são mostradas as páginas de cadastro de administradores. Na Figura 8 é mostrada a lista de administradores já cadastrados no sistema, mostrando para o usuário quem está armazenado no banco de dados, e permitindo também editar e/ou remover algum cliente que desejar. Na Figura 9 mostra os campos que deverão ser preenchidos para cadastrar um novo cliente tais como: nome completo, senha e email. Após inserir as informações corretamente, basta clicar em salvar que o novo administrador será cadastrado, ou clicar em voltar se não desejar efetuar o cadastro.

Na Figura 10 ilustra a tela de edição de administradores, ao clicar no botão de editar o usuário é redirecionado para a página mostrada, permitindo editar os dados que ele desejar, como vemos podendo editar o nome completo do administrador escolhido, o *email* não é permitido editar, para evitar erros no

funcionamento o desenvolvedor achou melhor desativar a edição. Após editar o que desejar basta clicar em salvar que a edição será executada. Ele pode também excluir todos os dados, basta clicar em excluir, uma mensagem de alerta é retornada para o usuário, perguntando se ele deseja mesmo excluir, evitando cliques acidentais e uma exclusão não desejada, como ilustrado na Figura 11.

Academia Home Cadastro ▾ Avaliação ▾ Financeiro ▾ Sair

Home / Cadastro / Admin / Editar Admin

Voltar Excluir Salvar

Nome Completo * Rodolpho Freitas

Email * rodolphofnrf@gmail.com

Copyright © 2015 ZOID

Figura 9 – tela de edição de administradores.

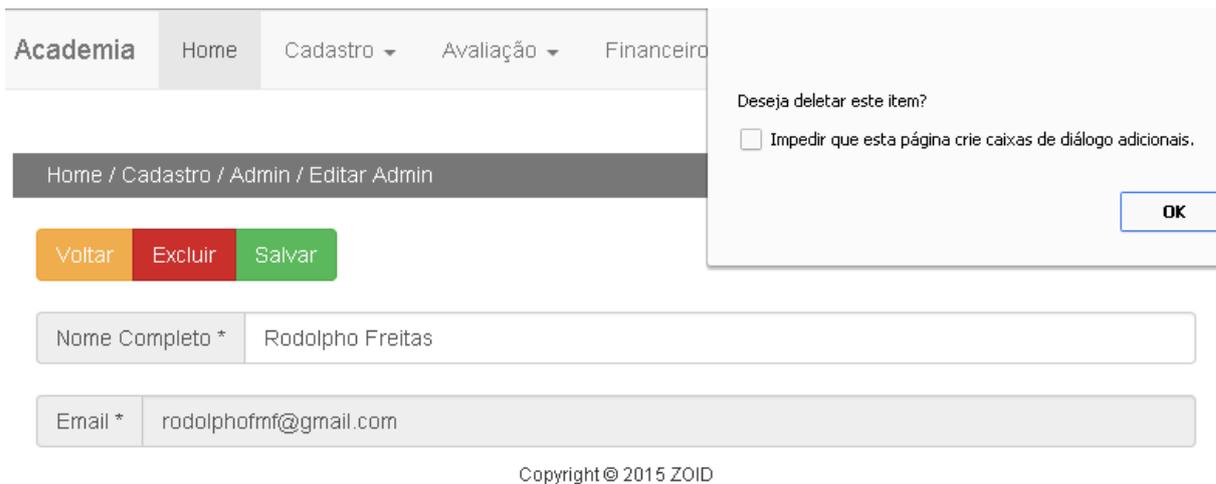


Figura 10 – mensagem de alerta de exclusão de administradores.

A mesma mensagem de alerta acontece ao clicar no botão remover presente na página de lista de administradores, ilustrada na Figura 8.

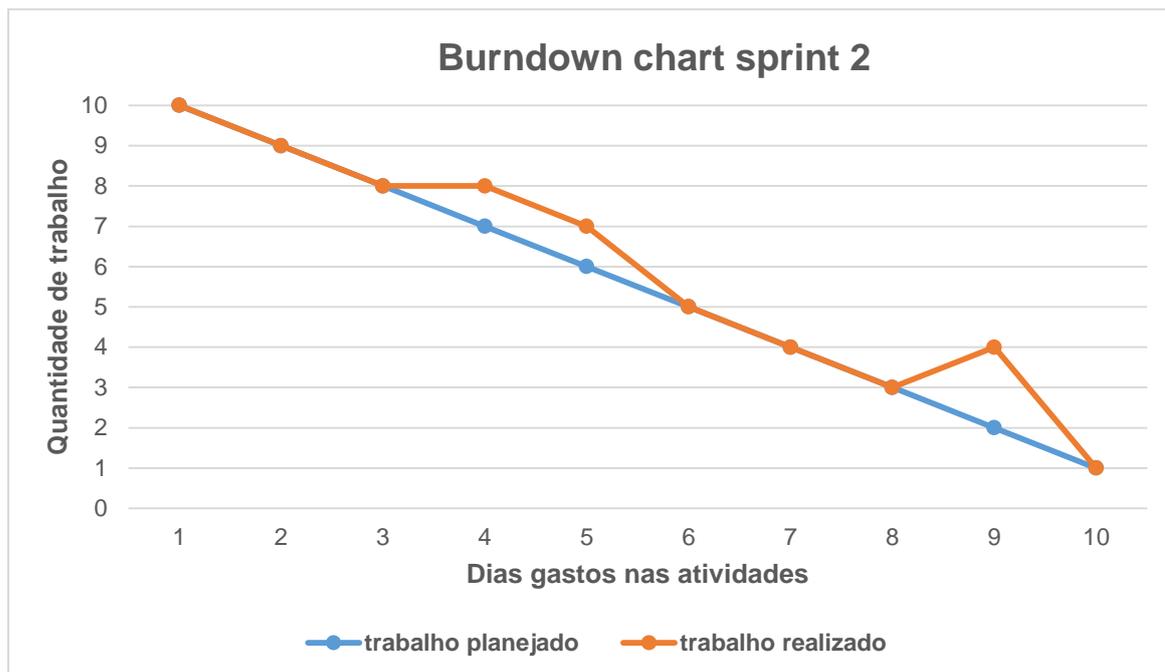


Figura 11 – *Burndown chart da Sprint 2*

No gráfico ilustrado na Figura 12 podemos observar que as tarefas estavam sendo executadas corretamente no prazo estipulado até o terceiro dia, onde ocorreram alguns problemas durante a implementação dos *controllers* e *viewers* das páginas. Após resolver os empecilhos, as atividades foram cumpridas no prazo até o oitavo dia, onde houve problemas no desenvolvimento das funções de controle do banco de dados. Os problemas foram resolvidos com alguns dias e as atividades foram cumpridas no prazo previsto, finalizando com sucesso a segunda Sprint.

Após concluir a segunda *Sprint*, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na Sprint 3: Cadastro de instrutores e clientes.

4.4 Sprint 3

Cadastro de instrutores e clientes

Primeiramente com o *Product Backlog* é gerada a meta que deverá ser alcançada com a terceira Sprint, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 5.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Cadastro de instrutores e clientes	1	10
Sub tarefa 1	Implementação das classes	1.1	2
Sub tarefa 2	Desenvolvimento da página de login	1.2	1
Sub tarefa 3	Implementação dos <i>controllers</i> e <i>viewers</i>	1.3	3
Sub tarefa 4	Implementação das funções (salvar, excluir, editar)	1.4	3
Sub tarefa 5	Desenvolvimento das interações com o banco de dados	1.5	2

Tabela 5 – Sprint backlog da Sprint 3.

Início da Sprint: O objetivo da terceira *Sprint* foi implementar a página de cadastro de instrutores e clientes e suas funcionalidades, e também desenvolver a página de *login* do sistema.

Reuniões diárias: nas reuniões foi visto que no início da *Sprint* as atividades foram executadas nos prazos estipulados e no decorrer das demais

atividades, de implementação, por exemplo, algumas dificuldades e gargalos apareceram, atrasando a realização das tarefas. Após resolver tais problemas, as tarefas foram desenvolvidas normalmente afim de finalizar a terceira *Sprint*, que foi finalizada com sucesso.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada com sucesso, cumprindo os requisitos e os custos previstos. Foi realizado uma revisão de tudo que foi feito, para evitar futuros problemas. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Implementação das classes, Desenvolvimento da página de *Login*, Implementação dos *Controllers* e *Viewers*, Implementação das funções (salvar, excluir, editar), Desenvolvimento das interações com o banco de dados. As Figuras 13 e 14 ilustram o que foi produzido na terceira *Sprint*.

Academia Studio Fitness

Login

Email do Usuário	<input type="text" value="Email de Usuário"/>
Senha	<input type="password" value="Senha"/>

Copyright © 2015 ZOID

Figura 12 – tela de login.

Na página ilustrada na Figura 13 o usuário efetua o *login* no sistema a partir do seu nome de usuário e senha, escolhido por ele e cadastrado pelo administrador. Tendo assim acesso e permitindo-o efetuar suas atividades.

Atualizar Lista Adicionar Item

Nome	Data de nascimento	Profissão	Telefone 1	Telefone 2	Email	Atividade	Endereço
Rodolpho Freitas	12/06/1989	estudante	3193961575		rodolphofmf@gmail.com	musculação	av silvio felicio dos santos, 253

Copyright © 2015 ZOID

Figura 13 – lista de todos clientes.

[Voltar](#) [Salvar](#)

Nome Completo *	Nome Completo
Data de nascimento *	Data de nascimento
Sexo *	Escolha uma das opcoes abaixo ▾
Profissão *	Profissão
Telefone fixo	Telefone 1
Celular	Telefone 2
Estado civil *	Estado civil
Email *	Email
Atividade *	Atividade
Endereco *	Endereco
Senha *	Senha

Copyright © 2015 ZOID

Figura 14 – tela de cadastro de clientes.

Nas Figuras 14 e 15 são mostradas as páginas de cadastro de clientes. Na Figura 14 é mostrada a lista de clientes já cadastrados no sistema, mostrando para o usuário quem está armazenado no banco de dados, e permitindo também editar e/ou remover algum cliente que desejar. Na Figura 15 mostra os campos que deverão ser preenchidos para cadastrar um novo cliente tais como: nome completo, data de nascimento, sexo, profissão, telefone 1 e telefone 2 se o cliente tiver mais de um número, estado civil, *email*, atividade que o cliente deseja praticar na academia (pilates, musculação, fisioterapia e outras), endereço e senha. Se as informações forem inseridas corretamente, basta clicar em salvar que o novo cliente será cadastrado, ou clicar em voltar se não desejar efetuar o cadastro.

Na Figura 16 ilustra a tela de edição de clientes, ao clicar no botão de editar o usuário é redirecionado para a página mostrada, permitindo editar os dados que ele desejar, como vemos podendo editar o nome completo, data de nascimento, sexo, profissão, telefone fixo, celular, estado civil, atividade e endereço do cliente escolhido. Após editar o que desejar basta clicar em salvar que a edição será executada. Ele pode também excluir todos os dados, basta clicar em excluir, uma mensagem de alerta é retornada para o usuário, perguntando se ele deseja mesmo excluir, evitando cliques acidentais e uma exclusão não desejada, como ilustrado na Figura 17.

Academia	Home	Cadastro ▾	Avaliação ▾	Financeiro ▾	Sair
-----------------	------	------------	-------------	--------------	------

Home / Cadastro / Cliente / Editar Clientes	
---	--

Voltar	Excluir	Salvar
--------	---------	--------

Nome Completo *	Rodolpho Freitas
Data de nascimento *	19/89/0612
Sexo *	Masculino ▾
Profissão *	estudante
Telefone fixo	(31) 9396-1575
Celular	Telefone 2
Estado civil *	solteiro
Email *	rodolphofmf@gmail.com
Atividade *	musculação
Endereco *	av silvio felicio dos santos, 253

Copyright © 2015 ZOID

Figura 15 – tela de edição de clientes

Academia Home Cadastro ▾ Avaliação ▾ Financeiro

Home / Cadastro / Cliente / Editar Clientes

Voltar
Excluir
Salvar

Deseja deletar este item?

Impedir que esta página crie caixas de diálogo adicionais.

OK

Nome Completo * Rodolpho Freitas

Data de nascimento * 19/89/0612

Sexo * Masculino ▾

Profissão * estudante

Telefone fixo (31) 9396-1575

Celular Telefone 2

Estado civil * solteiro

Email * rodolphofmf@gmail.com

Atividade * musculação

Endereço * av silvio felicio dos santos, 253

Copyright © 2015 ZOID

Figura 16 – mensagem de alerta de exclusão de clientes

A mesma mensagem de alerta acontece ao clicar no botão remover presente na página de lista de clientes, ilustrada na Figura 14.

Nas Figuras 18 e 19 são mostradas as páginas de cadastro de instrutores. Na Figura 18 é mostrada a lista de instrutores já cadastrados no sistema, mostrando para o usuário quem está armazenado no banco de dados, e permitindo também editar e/ou remover algum instrutor que desejar. Na Figura 19 mostra os campos que deverão ser preenchidos para cadastrar um novo instrutor, tais como: nome completo, função do instrutor, cpf, rg, email e senha. Se as informações forem inseridas corretamente, basta clicar em salvar que o novo instrutor será cadastrado, ou clicar em voltar se não desejar efetuar o cadastro.

Atualizar Lista Adicionar Item

Nome	Função	CPF	RG	Email	Editar	Remover
Rodolpho	instrutor pilates	222.222.222-22	10137333	rodolphofmf@gmail.com		

Copyright © 2015 ZOID

Figura 17 – lista de todos instrutores.

[Voltar](#) [Salvar](#)

Nome Completo *	Nome Completo
Função do Instrutor *	Função do Instrutor
CPF *	CPF
RG *	RG
Email *	Email
Senha *	Senha

Copyright © 2015 ZOID

Figura 18 – tela de cadastro de instrutores.

Na Figura 20 ilustra a tela de edição de instrutores, ao clicar no botão de editar o usuário é redirecionado para a página mostrada, permitindo editar os dados que ele desejar, como vemos podendo editar o nome completo, função do instrutor, cpf, rg,. Após editar o que desejar basta clicar em salvar que a edição será executada. Ele pode também excluir todos os dados, basta clicar em excluir, uma mensagem de alerta é retornada para o usuário, perguntando se ele deseja mesmo excluir, evitando cliques acidentais e uma exclusão não desejada, como ilustrado na Figura 21.

[Voltar](#) [Excluir](#) [Salvar](#)

Nome Completo *	Rodolpho
Função do Instrutor *	instrutor pilates
CPF *	222.222.222-22
RG *	10137333
Email *	rodolphofmf@gmail.com

Copyright © 2015 ZOID

Figura 19 – tela de edição de instrutores

Academia Home Cadastro ▾ Avaliação ▾ Financeiro

Home / Cadastro / Instrutor / Editar Instrutores

Voltar Excluir Salvar

Nome Completo *	Rodolpho
Função do Instrutor *	instrutor pilates
CPF *	222.222.222-22
RG *	10137333
Email *	rodolphofmf@gmail.com

Deseja deletar este item?

Impedir que esta página crie caixas de diálogo adicionais.

OK

Copyright © 2015 ZOID

Figura 20 – mensagem de alerta de exclusão de instrutores

A mesma mensagem de alerta acontece ao clicar no botão remover presente na página de lista de instrutores, ilustrada na Figura 18.

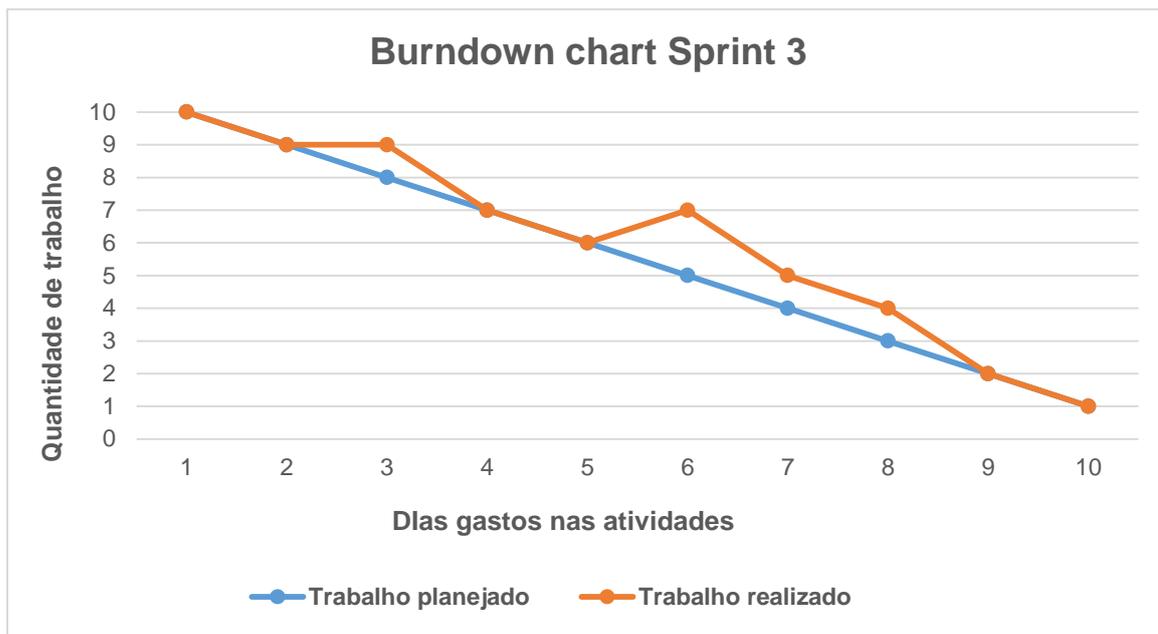


Figura 21 – Burndown chart Sprint 3

No gráfico ilustrado na Figura 22 podemos observar que as tarefas estavam sendo executadas corretamente no prazo estipulado até o segundo dia, onde ocorreram alguns problemas durante a implementação dos *controllers* e *viewers* das páginas. Após resolver as pendências, as atividades foram cumpridas no prazo até o quinto dia, onde houve problemas na implementação das funções (salvar, editar e excluir). Os problemas foram resolvidos com alguns dias e as atividades foram cumpridas com um pequeno atraso mas finalizando com sucesso a terceira *Sprint*.

Após concluir a terceira *Sprint*, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na *Sprint 4*: Cadastro de avaliações físicas.

4.5 Sprint 4

Cadastro de avaliações físicas

Usando o *Product Backlog* é definida a meta que deverá ser alcançada com a quarta *Sprint*, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 6.

	Funcionalidades	Prioridade	Custo/dias
Tarefa Principal	Cadastro de avaliações físicas	2	10
Sub tarefa 1	Implementação das classes necessárias	2.1	3
Sub tarefa 2	Implementação dos controlers e viewers	2.2	3
Sub tarefa 3	Desenvolvimento das interações com o banco de dados	2.3	3
Sub tarefa 4	Revisão final do desenvolvimento	2.4	1

Tabela 6 – *Sprint backlog da Sprint 4*

Início da Sprint: O objetivo da quarta *Sprint* foi implementar a página de cadastro de avaliações físicas e suas funcionalidades.

Reuniões diárias: nas reuniões foi visto que no início da *Sprint* as atividades foram executadas nos prazos estipulados e no decorrer das demais atividades, de implementação, por exemplo, algumas dificuldades apareceram, conseqüentemente, atrasando a realização das tarefas. Após corrigir os erros de desenvolvimento no processo de comunicação com o banco de dados, as tarefas foram executadas afim de finalizar a quarta *Sprint* sem atrasos, que foi finalizada com sucesso.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada corretamente, cumprindo os requisitos e os custos previstos. Foi realizado

uma revisão do que foi produzido, para evitar problemas futuros. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Implementação das classes, Implementação dos *Controllers* e *Viewers*, Desenvolvimento das interações com o banco de dados, Revisão final do desenvolvimento. As Figuras 23 e 24 ilustram o que foi produzido na quarta *Sprint*.

Academia Home Cadastro ▾ Avaliação ▾ Financeiro ▾ Sair

Home / Cadastro / Avaliação / Lista de Todas as Avaliações

Atualizar Lista Adicionar Item

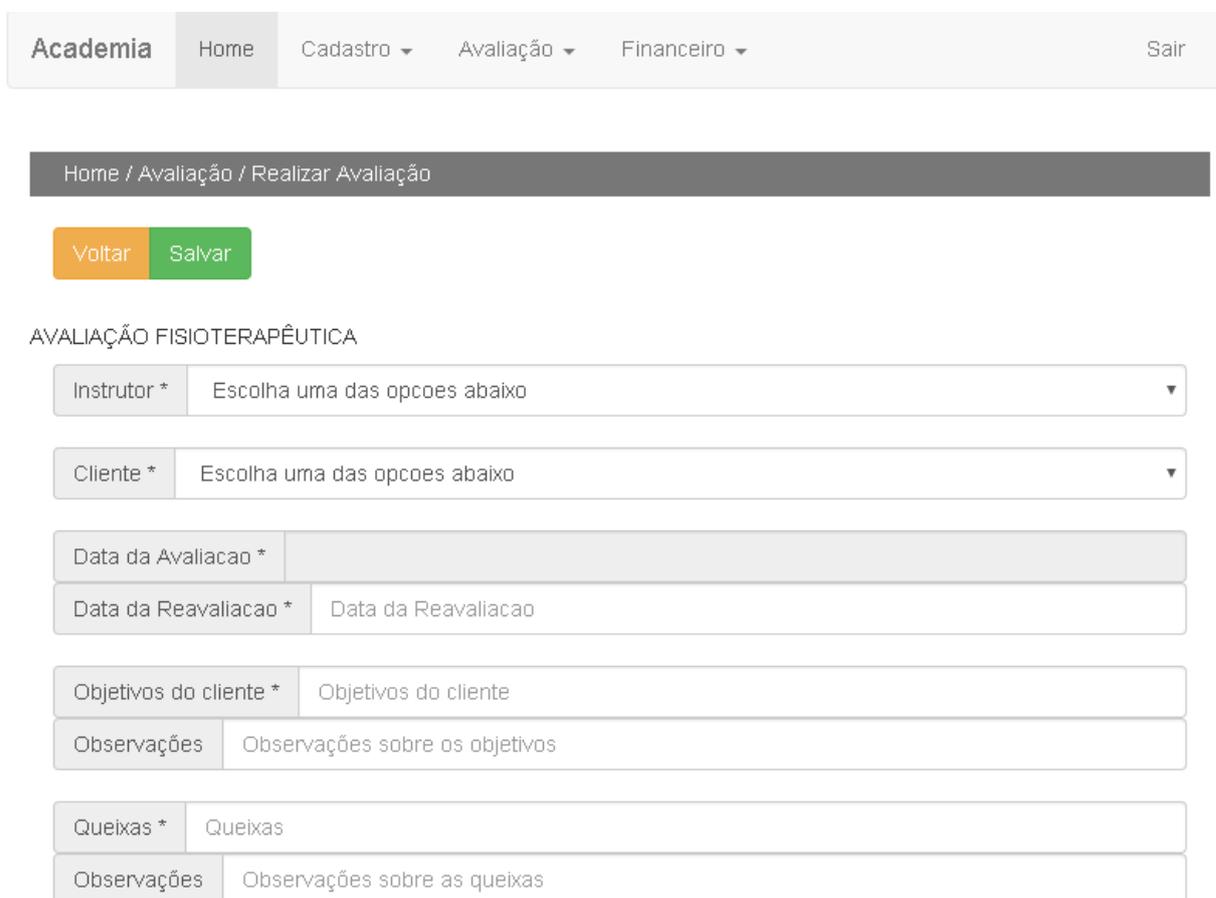
Id Avaliação	Instrutor	Cliente	Editar
1	Rodolpho	Rodolpho Freitas	

Copyright © 2015 ZOID

Figura 22 – lista de todas avaliações.

Na Figura 23 é mostrada a lista de avaliações já cadastrados no sistema, mostrando para o usuário as avaliações armazenadas no banco de dados, indicando a id da respectiva avaliação, o instrutor que executou o cadastro e o cliente que foi avaliado, e permitindo também editar as informações que desejar.

Por se tratar de um número grande de campos que devem ser preenchidos, vamos analisar por partes o processo de cadastro de avaliações. Na Figura 24 observamos o início do cadastro da avaliação fisioterapêutica,



Academia Home Cadastro ▾ Avaliação ▾ Financeiro ▾ Sair

Home / Avaliação / Realizar Avaliação

Voltar Salvar

AVALIAÇÃO FISIOTERAPÊUTICA

Instrutor * Escolha uma das opções abaixo ▾

Cliente * Escolha uma das opções abaixo ▾

Data da Avaliação *

Data da Reavaliação *

Objetivos do cliente *

Observações

Queixas *

Observações

Figura 23 – primeira parte do cadastro de avaliações.

Para efetuar o cadastro de uma avaliação primeiramente o instrutor deve escolher o seu próprio nome no campo instrutor, em seguida escolher o cliente avaliado. A data da avaliação é preenchida automaticamente pelo sistema por uma função implementada no código, em seguida preencher uma data para reavaliar o cliente. A reavaliação é efetuada geralmente depois de 1 a 2 meses de atividades físicas exercidas pelo cliente. Continuando, ele preenche os objetivos do cliente (musculação, perda de peso, condicionamento físico, lazer ou outros), se houver alguma observação importante preencher o campo de observações. Queixas dos clientes podem ser, geralmente, dores, sedentarismo ou outras. Se houver alguma observação pertinente, preencher em observações abaixo do campo queixas. Após

preencher todos os campos obrigatórios marcados com asterisco, basta clicar em salvar e concluir a avaliação. Se quiser cancelar o cadastro é clicar no botão voltar e retornar para a página da lista de avaliações cadastradas. Na Figura 25 observamos a sequência do cadastro de uma avaliação.

PAR-Q

1.Seu médico já lhe disse que você possui algum problema cardíaco e recomendou atividades físicas apenas sob supervisão médica?

Resposta	Escolha uma das opções abaixo	▼
----------	-------------------------------	---

2.Você tem dor no peito provocada por alguma atividade física?

Resposta	Escolha uma das opções abaixo	▼
----------	-------------------------------	---

3.Você teve dor no peito no último mês?

Resposta	Escolha uma das opções abaixo	▼
----------	-------------------------------	---

4.Você já perdeu a consciência em alguma ocasião ou sofreu alguma queda em virtude de tontura?

Resposta	Escolha uma das opções abaixo	▼
----------	-------------------------------	---

5.Você tem algum problema ósseo ou articular que poderia se agravar com as atividades físicas propostas?

Resposta	Escolha uma das opções abaixo	▼
----------	-------------------------------	---

6.Algum médico já lhe prescreveu medicamento para Pressão Arterial ou para o Coração?

Resposta	Escolha uma das opções abaixo	▼
----------	-------------------------------	---

7.Você tem conhecimento, por alguma informação médica ou pela própria experiência, de algum motivo que poderia impedi-lo de participar de atividades físicas sem supervisão médica?

Resposta	Escolha uma das opções abaixo	▼
----------	-------------------------------	---

Anamnese	Anamnese
----------	----------

Figura 24 – segunda parte do cadastro de avaliações.

Dando sequência ao processo de cadastro o instrutor deve preencher as perguntas do campo Par-q, selecionando uma das respostas Sim ou Não, e preencher o campo anamnese, uma entrevista com o cliente a fim de obter um diagnóstico mais completo da sua saúde. Na Figura 26 ilustra os próximos campos a serem preenchidos.

DADOS VITAIS/MEDIDAS

FC (Repouso) *	FC (Repouso)
Pressão arterial *	Pressão arterial
Peso *	Peso
Estatura *	Estatura

HÁBITOS DE VIDA

Tabagista	Tabagista
Etilista	Etilista
Medicamentos *	Medicamentos

HISTÓRIA FAMILIAR

Cardiopatias *	Cardiopatias
Hipertensão *	Hipertensão
Diabetes *	Diabetes

Figura 25 – primeira parte do cadastro de avaliações.

O instrutor continua o cadastro da avaliação, dessa vez preenchendo os dados vitais como frequência cardíaca em repouso, a pressão arterial e o peso do cliente. As medições são feitas pelo instrutor e inseridas nos respectivos campos. Em seguida os hábitos de vida são preenchidos, se o cliente fuma, ingere bebidas alcoólicas ou usa medicamentos. Em seguida é perguntado ao cliente seu histórico familiar, se algum membro da família é portador de algum problema cardíaco, de hipertensão ou diabetes. Após preencher os campos citados o instrutor dá sequência ao processo de cadastro ilustrado na Figura 27.

HISTÓRIA PREGRESSA

Lesões Músculo-Esqueléticas *	Lesões Músculo-Esqueléticas
Sistema Cardio-respiratório *	Sistema Cardio-respiratório
Cirurgias/Internações	Cirurgias/Internações
Alergias	Alergias
Outras	Outras

Figura 26 – terceira parte do cadastro de avaliações.

Na seção de história pregressa o instrutor pergunta ao cliente se ele tem lesões em algum músculo ou em algum osso de seu corpo, pergunta se ele já foi submetido a alguma cirurgia ou se foi internado em alguma ocasião, e se há alergias. Se houver alguma informação para relatar, o instrutor preenche o campo outras. Na Figura 28 observamos os próximos campos para preencher da avaliação.

FLEXIBILIDADE

Legenda

01 - Excelente

02 - Normal

03 - Ligeiramente Encurtado

04 - Moderadamente Encurtado

05 - Severamente Encurtado

Adutor Esquerdo	Adutor Esquerdo
Adutor Direito	Adutor Direito
Peitoral Esquerdo	Peitoral Esquerdo
Peitoral Direito	Adutores Direito
Reto Femoral Esquerdo	Reto femoral Esquerdo
Reto Femoral Direito	Adutores Direito
Isquiotibais Esquerdo	Isquiotibais Esquerdo
Isquiotibais Direito	Adutores Direito
Tríceps Sural Esquerdo	Tríceps Sural Esquerdo
Tríceps Sural Direito	Tríceps Sural Direito

Figura 27 – quarta parte do cadastro de avaliações.

Na seção de flexibilidade o instrutor preenche os campos destinados, com base na legenda mostrada, indicando qual o estado de flexibilidade dos músculos do cliente. 01 para estado Excelente, 02 Normal, 03 Ligeiramente encurtado, 04 Moderadamente encurtado e 05 Severamente encurtado. O instrutor faz as medições necessárias, examinando o cliente, preenchendo os campos com seus respectivos estados. Segue na Figura 29 a próxima seção a ser preenchida.

PERIMETRIA (CM)

Ombro	Centímetros
Tórax	Centímetros
Cintura	Centímetros
Abdômen	Centímetros
Quadril	Centímetros
Braço Esquerdo	Centímetros
Braço Direito	Centímetros
Antebraço Esquerdo	Centímetros
Antebraço Direito	Centímetros
Coxa Esquerda	Centímetros
Coxa Direita	Centímetros
Panturrilha Esquerda	Centímetros
Panturrilha Direita	Centímetros
Pescoço	Centímetros

Obs: Medidas realizadas sem contração. Braço ao lado do corpo, levemente abduzido.

Figura 28 – quinta parte do cadastro de avaliações.

Na seção de perimetria o instrutor usa uma fita métrica para obter as informações necessárias para o cadastro. Após colher as medidas do cliente, ele preenche os campos destinados para cada membro do corpo. A medida é dada em centímetros. Uma observação é feita nessa seção, atentando o instrutor para a postura correta para a medição (Observação: medidas realizadas sem contração. Braço ao lado do corpo, levemente abduzido) evitando assim possíveis erros na coleta dos dados. Na próxima seção damos continuidade ao cadastro da avaliação.

DOBRAS CUTÂNEAS

Bíceps	Milímetros
Tríceps	Milímetros
Subescapular	Milímetros
Peitoral	Milímetros
Subaxilar	Milímetros
Abdominal	Milímetros
Supra Iílica	Milímetros
Quadríceps	Milímetros
Panturrilha	Milímetros
Resultado Atual (%)	Resultado Atual (%)
Resultado Desejável (%)	Resultado Desejável (%)
Peso Gordo (Kg)	Peso Gordo (Kg)
Peso Magro (Kg)	Peso Magro (Kg)

Figura 29 – sexta parte do cadastro de avaliações.

Na seção de dobras cutâneas, ilustrada na Figura 30 o instrutor usa um aparelho chamado adipômetro, também chamado de compasso de dobras cutâneas, para obter a composição corporal do cliente a partir da espessura em milímetros. Após efetuar a medição das dobras necessárias ele preenche os campos da seção. Os cálculos de resultado atual, peso gordo e peso magro são feitos pelos intrutores ou fisioterapeutas, e colocados em seus campos. No campo de resultado desejável o cliente responde ao instrutor qual sua expectativa de resultado ao praticar suas atividades, e então, o campo é preenchido. Após preencher todos os campos o instrutor avança para a última seção do cadastro.

AVALIAÇÃO POSTURAL

VISTA ANTERIOR

Cabeça	Opções
Opções: Alinhada, Inclinação, Rodada	
Ombros	Opções
Opções: Alinhado, Inclinado	
Pelve	Opções
Opções: Alinhada, Elevada, Rodada	
Joelhos	Opções
Opções: Alinhado, Valgo, Varo	
Pés	Opções
Opções: S/ alterações (D)(E), Pronado (D)(E), Supinado (D)(E)	

VISTA LATERAL

Cabeça	Opções
Opções: Alinhada, Protusa	
Ombros	Opções
Opções: Alinhado, Protuso	
Torácica	Opções
Opções: Fisiológica, Aumentada, Retificada	
Lombar	Opções
Opções: Fisiológica, Aumentada, Retificada	
Pelve	Opções
Opções: Alinhada, Antrovertida, Retrovertida	
Joelhos	Opções
Opções: Alinhado, Recurvatum, Semi-Fletido	

VISTA POSTERIOR

Coluna	Opções
Opções: Alinhada, Escoliose, Em S, Em C, Concavidade (D)(E)	
Escápulas	Opções
Opções: Alinhadas, Aladas (D)(E), Retraídas (D)(E)	
Observações	Observações

Copyright © 2015 ZOID

Figura 30 – sétima parte do cadastro de avaliações.

Na última seção de cadastro ilustrada na Figura 31, a avaliação postural, o instrutor analisa a postura do cliente, observando de vários pontos de vista diferentes, vista anterior, lateral e posterior. O instrutor preenche os campos de acordo com as opções recomendadas abaixo de cada campo, correspondentes para as partes do corpo do cliente.

Preenchidos todos os campos obrigatórios (o instrutor deve preencher todos os campos marcados com asterisco) basta clicar no botão salvar para concluir a avaliação, se ele quiser cancelar basta clicar no botão voltar.

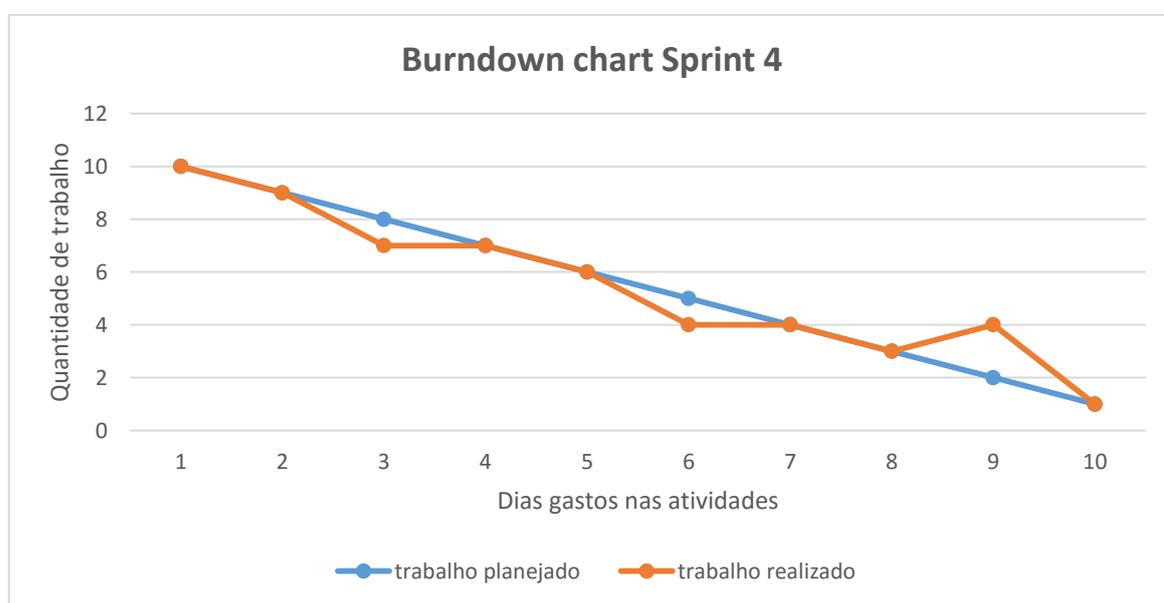


Figura 31 – Burndown chart da Sprint 4

No gráfico ilustrado na Figura 32, podemos observar que as tarefas estavam sendo executadas corretamente no prazo estipulado, algumas atividades foram executadas antes do prazo, como visto nos dias 2 a 4 e 5 a 7, devido a experiência de desenvolvimento adquirida nas *Sprints* anteriores. Os problemas apareceram no dia 8, onde foi realizada a revisão das atividades, foram descobertos erros no desenvolvimento no processo de salvar as informações no banco de dados, causando atraso para terminar a Sprint. Após corrigir os erros as atividades foram finalizadas com um pequeno atraso mas apesar das pendencias a quarta *Sprint* foi concluída com sucesso.

Após concluir a quarta *Sprint*, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na *Sprint 5*: Análise dos resultados das avaliações físicas.

4.6 Sprint 5

Análise dos resultados das avaliações físicas

Com o *Product Backlog* foi definida a meta que deverá ser alcançada com a quinta *Sprint*, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 7.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Análise dos resultados das avaliações físicas	3	10
Sub tarefa 1	Implementação das classes necessárias	3.1	2
Sub tarefa 2	Implementação dos controllers e viewers	3.2	1
Sub tarefa 3	Implementação das funções (salvar, editar, excluir)	3.3	1
Sub tarefa 4	Desenvolvimento das interações com o banco de dados	3.4	1
Sub tarefa 5	Desenvolvimento do módulo de análise das avaliações físicas	3.5	4
Sub tarefa 6	Revisão e teste do código desenvolvido	3.6	1

Tabela 7 - Sprint backlog da Sprint 5.

Início da Sprint: O objetivo da quinta *Sprint* foi implementar a página de análise dos resultados das avaliações físicas e suas funcionalidades.

Reuniões diárias: nas reuniões foi visto que no início da *Sprint* as atividades foram executadas nos prazos estipulados e no decorrer das demais atividades, de implementação, por exemplo, muitas dificuldades surgiram, conseqüentemente, atrasando a realização das tarefas. Após muito estudo, revisões conseguiu corrigir os erros de desenvolvimento, as tarefas foram executadas com muito esforço por parte do desenvolvedor. A *Sprint 5* até o momento foi a mais trabalhosa de desenvolver, por conter muitas funções complexas que demandaram muito tempo e esforço. Por se tratar de um grande número de dados e campos preenchidos, houve dificuldade no processo de salvar no banco e também na parte que o sistema efetua os cálculos. Após superar os imprevistos afim de finalizar a quinta *Sprint*, a mesma foi finalizada corretamente mas como dito, com grande atraso.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada corretamente, cumprindo os requisitos e os custos. Foi realizado uma revisão do que foi desenvolvido, para evitar problemas futuros. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Implementação das classes, Implementação dos *Controlers* e *Viewers*, Desenvolvimento das interações com o banco de dados, Desenvolvimento do módulo de análise das avaliações físicas, Revisão e teste do código desenvolvido.

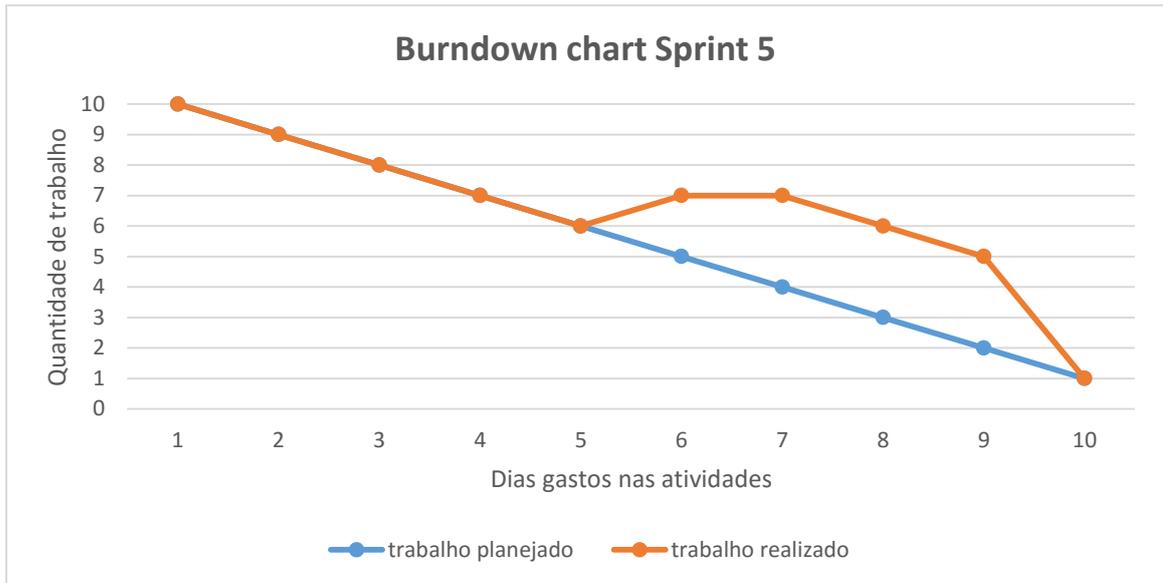


Figura 32 – Burndown chart da Sprint 5.

No gráfico ilustrado na Figura 33, podemos observar que as tarefas estavam sendo executadas corretamente no prazo estipulado. Os problemas apareceram no dia 5, onde foram descobertos erros no desenvolvimento no processo de salvar as informações no banco de dados, devido ao grande número de informações que precisavam ser armazenadas no banco causando atraso para terminar a *Sprint*. Tais erros foram resolvidos com muito estudo e esforço do desenvolvedor. Após corrigir os erros as atividades foram finalizadas com um atraso significativo mas apesar das pendências a quinta *Sprint* foi concluída com sucesso.

Após concluir a quinta *Sprint*, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na *Sprint 6*: Módulo que mostra para o usuário os resultados das análises das avaliações físicas.

4.7 Sprint 6

Módulo que mostra para o usuário os resultados das análises das avaliações físicas

Com o *Product Backlog* foi definida a meta que deverá ser alcançada com a sexta *Sprint*, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 8.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Módulo que mostra para o usuário os resultados das análises das avaliações físicas	3	10
Sub tarefa 1	Implementação das classes utilizadas	3.1	1
Sub tarefa 2	Implementação das funções	3.2	2
Sub tarefa 3	Desenvolvimento das ligações com o banco de dados	3.3	2
Sub tarefa 4	Implementação do módulo que retorna os resultados das análises das avaliações físicas	3.4	4
Sub tarefa 4	Revisão e teste do produto desenvolvido	3.5	1

Tablela 8 – Sprint backlog da Sprint 6.

Início da Sprint: O objetivo da sexta *Sprint* foi implementar o módulo que mostra para o usuário os resultados das análises das avaliações físicas.

Reuniões diárias: nas reuniões foi visto que no início da *Sprint* as atividades foram executadas nos prazos estipulados e no decorrer das demais atividades, de implementação, por exemplo, algumas dificuldades apareceram, com isso houve atraso na realização das atividades. Para resolver os gargalos durante a implementação foi necessário muito estudo e trabalho do desenvolvedor, e com várias revisões conseguiu corrigir os erros de desenvolvimento, as tarefas foram executadas. A *Sprint* 6 também foi muito trabalhosa de implementar, por conter várias funções complexas que demandaram muito tempo e esforço. Após superar todas adversidades a *Sprint* foi finalizada corretamente.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada corretamente, cumprindo os requisitos e os custos. Foi realizado uma revisão do que foi desenvolvido, para evitar problemas futuros. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Implementação das classes utilizadas, Implementação das funções, Desenvolvimento das ligações com o banco de dados, Implementação do módulo que retorna os resultados das análises das avaliações físicas, Revisão e teste do produto desenvolvido.

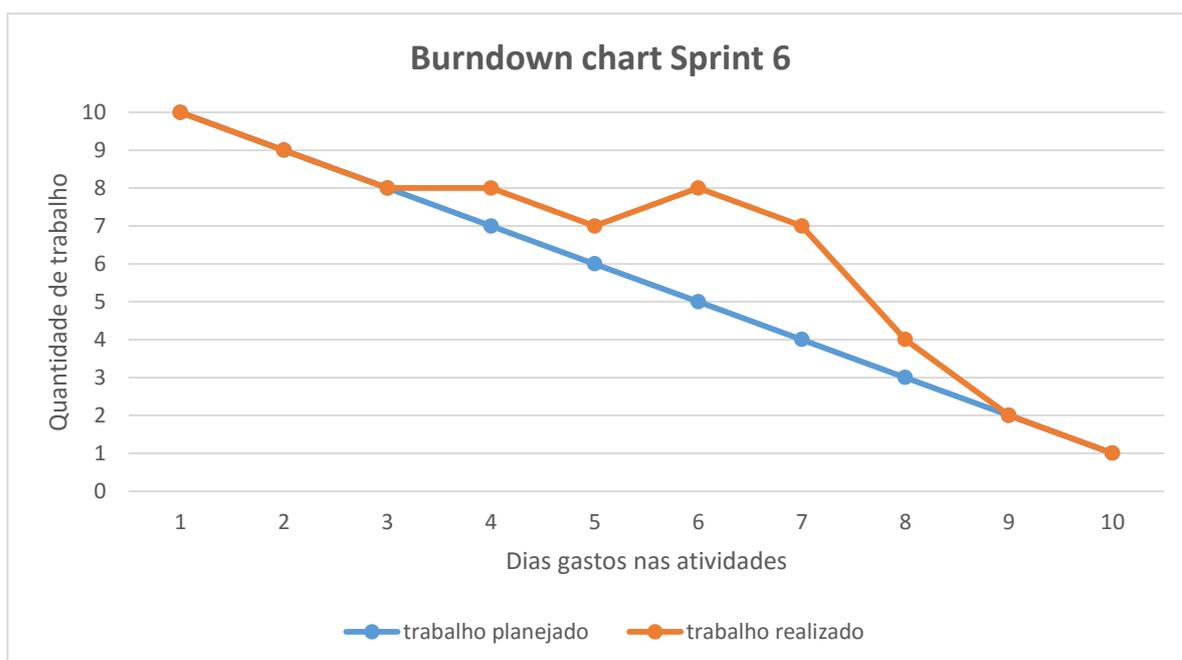


Figura 33 – Burndown chart da Sprint 6

No gráfico ilustrado na Figura 34, podemos observar que as tarefas estavam sendo executadas corretamente no prazo previsto. Os problemas apareceram no dia 4, onde foram descobertos erros no desenvolvimento de algumas funções e outros erros nas operações do banco de dados, devido a várias requisições feitas, para o retorno de informações importantes, causando atraso para terminar a *Sprint*. Tais erros foram resolvidos com muito estudo e esforço do desenvolvedor. Após corrigir os erros as atividades foram finalizadas com algum atraso mas apesar dos problemas a sexta *Sprint* foi concluída com sucesso. Após concluir a sexta *Sprint*, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na *Sprint 7*: Agendamento de atendimento.

4.8 Sprint 7

Agendamento de atendimento

Com o *Product Backlog* foi definida a meta que deverá ser alcançada com a sétima *Sprint*, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 9.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Agendamento de atendimento	4	10
Sub tarefa 1	Desenvolvimento das estruturas	4.1	2
Sub tarefa 2	Desenvolvimento das funções	4.2	3
Sub tarefa 3	Implementação da agenda	4.3	3
Sub tarefa 4	Revisão de código	4.4	2

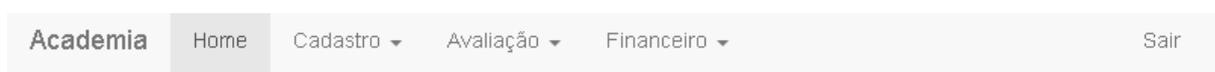
Tabela 9 – *Sprint backlog* da *Sprint 7*.

Início da Sprint: O objetivo da sétima *Sprint* foi desenvolver o agendamento de atendimento, com isso o usuário pode marcar compromissos e consultas com seus clientes.

Reuniões diárias: nas reuniões foi visto que no início da *Sprint* as atividades foram executadas nos prazos previstos e não houve muitas dificuldades durante a implementação da agenda, devido a experiência do desenvolvedor em trabalhos já executados por ele em relação a implementação de agendas em sistemas web.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada corretamente, cumprindo os requisitos e os custos. Foi realizado uma revisão do que foi desenvolvido, para evitar problemas. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Desenvolvimento das estruturas, Desenvolvimento das funções, Implementação da agenda, Revisão de código. A Figura 35 e ilustra o que foi produzido na sétima *Sprint*.

Na Figura 35 é mostrada a página home onde está presente a agenda de horários. A agenda é dividida em colunas. A primeira coluna é composta pelos horários que a academia geralmente funciona. A segunda coluna é destinada ao cliente que deseja atendimento. A terceira coluna é designada para o tipo de atendimento requisitado, como por exemplo, uma consulta de fisioterapia. O usuário insere na coluna cliente o cliente que marcou o atendimento, relacionado com o horário escolhido, e o tipo de atendimento é inserido na última coluna.



Olá, Bem vindo a Academia Studio Fitness 

Agenda de atendimentos		
Horários	Cliente	Tipo de atendimento
07:00	Rodolpho Freitas	Fisioterapia
08:00		
09:00		
10:00		
11:00		
12:00		
14:00		
15:00		
16:00		
17:00		
18:00		
19:00		
20:00		
21:00		

Copyright © 2015 ZOID

Figura 34 – tela *home* com a agenda de atendimento.

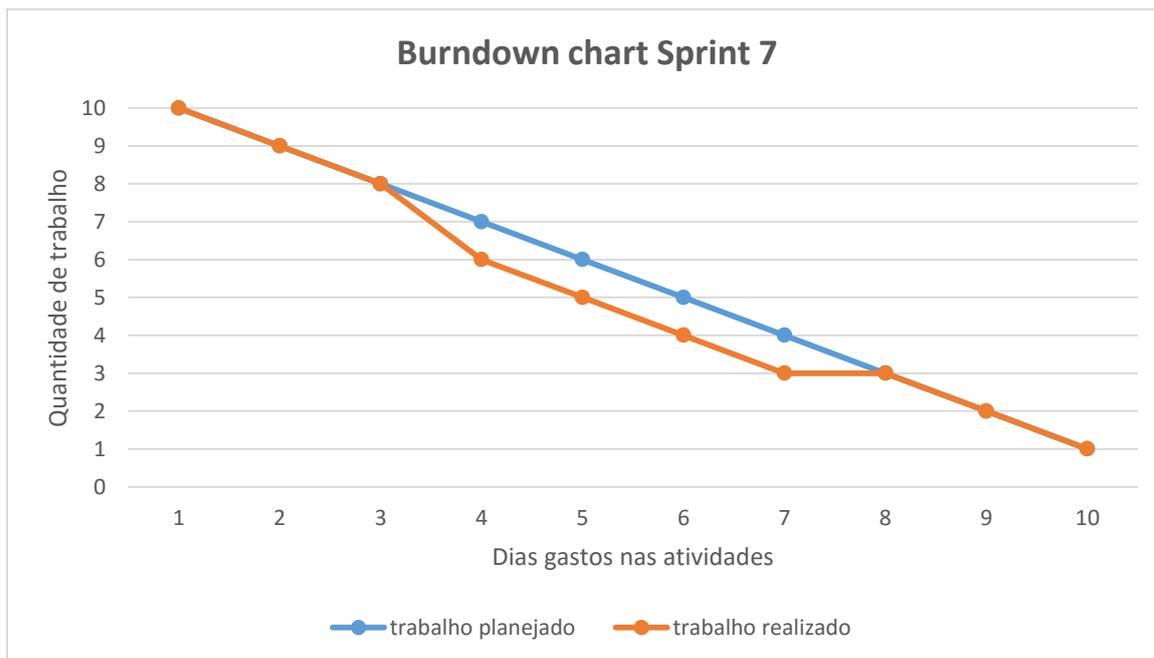


Figura 35 – Burndown chart da Sprint 7

No gráfico ilustrado na Figura 36, observamos que as tarefas foram executadas corretamente no prazo previsto, sem dificuldades. Do dia 3 ao dia 8 as tarefas foram finalizadas antes dos prazos estipulados, devido a experiência do programador em desenvolver as atividades requisitadas, adiantando a conclusão da sétima *Sprint*. Após concluir a sétima *Sprint*, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na *Sprint 8*: Efetuar movimentos diários.

4.9 Sprint 8

Efetuar movimentos diários

Com o *Product Backlog* foi definida a meta que deverá ser alcançada com a oitava *Sprint*, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 10.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Efetuar movimentos diários	4	10
Sub tarefa 1	Implementar as estruturas necessárias	4.1	2
Sub tarefa 2	Desenvolver as tabelas dos movimentos diários	4.2	2
Sub tarefa 3	Implementação das funções	4.3	3
Sub tarefa 4	Desenvolver as ligações com o banco de dados	4.4	2
Sub tarefa 5	Revisão e teste do produto desenvolvido	4.5	1

Tabela 10 – Sprint backlog da Sprint 8.

Início da Sprint: O objetivo da oitava *Sprint* foi implementar a página que o usuário efetua os movimentos diários, como pagamentos ou qualquer movimentação monetária ocorrida na academia.

Reuniões diárias: nas reuniões foi visto que logo nos primeiros dias houve alguns problemas no desenvolvimento das tabelas dos movimentos, mas tais

problemas foram resolvidos rapidamente pelo desenvolvedor, seguindo normalmente a execução da Sprint até sua finalização.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada com sucesso, cumprindo os requisitos e os custos. Foi realizado uma revisão do que foi desenvolvido a fim de evitar erros futuros. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Implementar as estruturas necessárias, Desenvolver as tabelas dos movimentos diários, Implementação das funções, Desenvolver as ligações com o banco de dados, Revisão e teste do produto desenvolvido. A Figura 37 ilustra o que foi produzido na oitava *Sprint*.

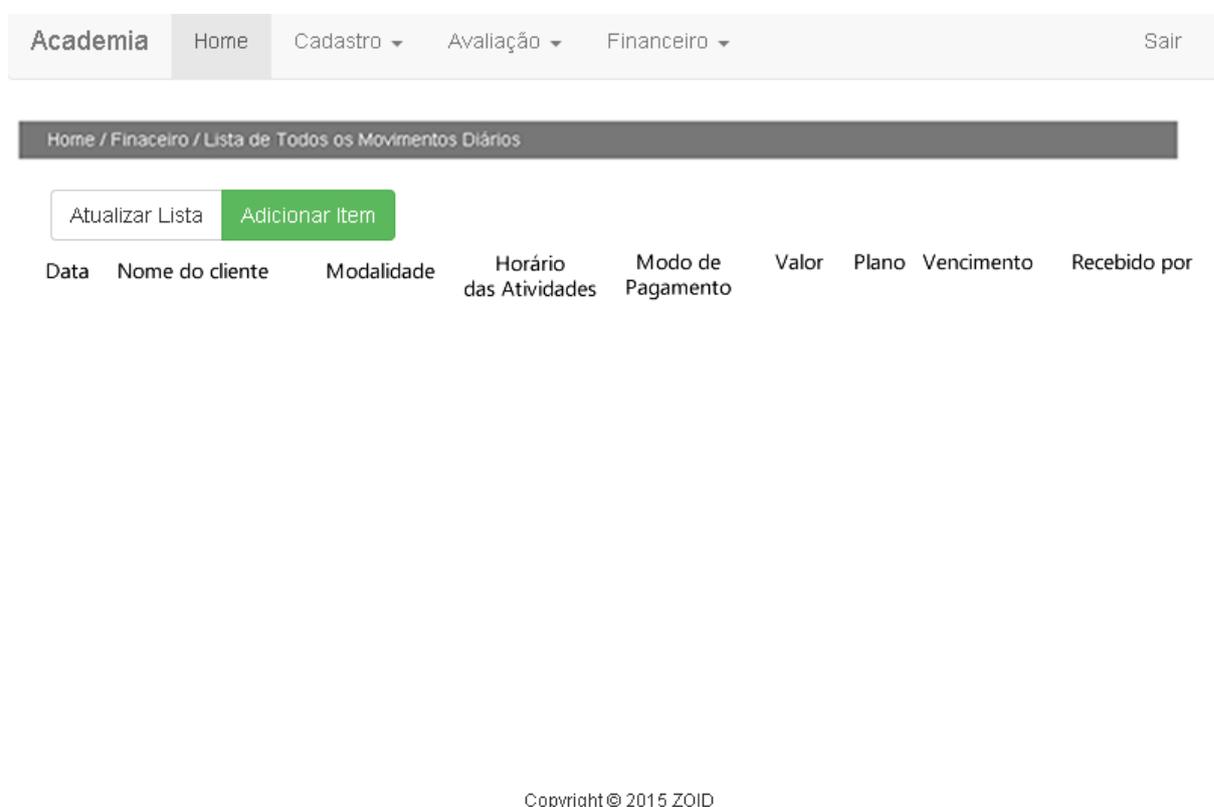


Figura 36 – Tela de movimentos diários.

Na seção de movimentos diários, o usuário pode adicionar os pagamentos efetuados pelos clientes em determinados dias. A seção conta com o campo de data, preenchido com a data que o pagamento foi efetuado, o nome do cliente que efetuou o mesmo, a modalidade que ele exerce na academia, o horário de suas atividades (manhã, tarde ou noite), o modo de pagamento (cheque, cartão de crédito ou débito,

dinheiro), o valor pago, o plano (mensal, trimestral, semestral), a data de vencimento do pagamento, e por quem foi recebido o pagamento.

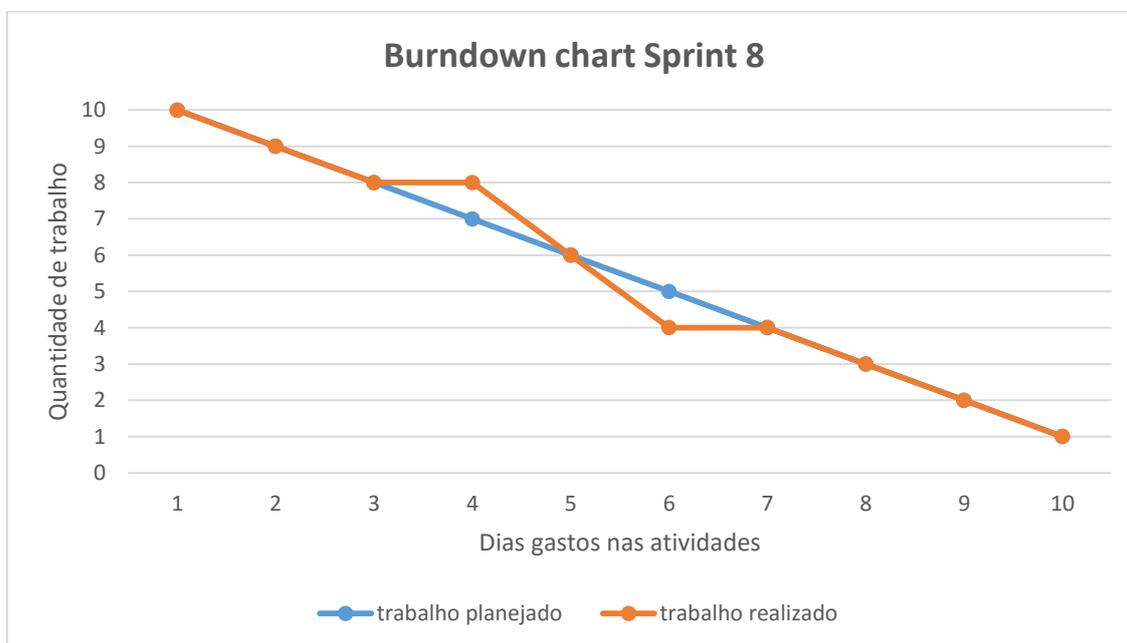


Figura 37 – Burndown chart da Sprint 8.

No gráfico ilustrado na Figura 38, observamos que no terceiro dia houve problemas no desenvolvimento das tabelas dos movimentos diários, devido à alta quantidade de dados necessários causando dificuldades em encaixá-las no tamanho da página. Apesar dos problemas, o desenvolvedor conseguiu resolvê-los rapidamente, conseguindo finalizar a *Sprint* sem mais atrasos.

Após concluir a oitava *Sprint*, foram geradas as prioridades das próximas tarefas, os prazos estipulados e o que seria executado na *Sprint* 9: Design das páginas do sistema.

4.10 Sprint 9

Design das páginas do sistema.

Com o *Product Backlog* foi definida a meta que deverá ser alcançada com a nona *Sprint*, com o *Sprint Backlog* subdivide-se a tarefa principal em pequenas tarefas, listadas na Tabela 11.

	Funcionalidades	Prioridade	Custo/dias
Tarefa principal	Design das páginas do sistema	5	10
Sub tarefa 1	Implementação dos códigos CSS necessários	5.1	3
Sub tarefa 2	Desenvolvimento do layout das paginas	5.2	3
Sub tarefa 3	Edição das imagens utilizadas nas paginas	5.3	3
Sub tarefa 4	Revisão das layouts e resoluções	5.4	1

Tabela 11 – Sprint backlog da Sprint 9.

Início da Sprint: O objetivo da nona *Sprint* foi desenvolver o design das páginas do sistema, envolvendo o layout e imagens utilizadas na busca de uma melhor aparência das páginas, utilizando design responsivo para melhor acesso em diferentes tamanhos de telas.

Reuniões diárias: nas reuniões foi visto as atividades foram bem executadas pelo desenvolvedor, sem muitos atrasos. Houve poucas dificuldades nessa Sprint. As dificuldades que surgiram foram rapidamente resolvidas, levando a Sprint a uma conclusão bem sucedida.

Revisão da Sprint: no final do prazo estipulado, verificou que a *Sprint* foi finalizada com sucesso, cumprindo os requisitos e os custos. Foi realizado uma revisão do que foi desenvolvido a fim de evitar erros futuros. Portanto os *Sprints Backlogs* foram finalizados com sucesso: Implementação dos códigos CSS necessários, Desenvolvimento do layout das páginas, Edição das imagens utilizadas nas páginas, Revisão das layouts e resoluções.

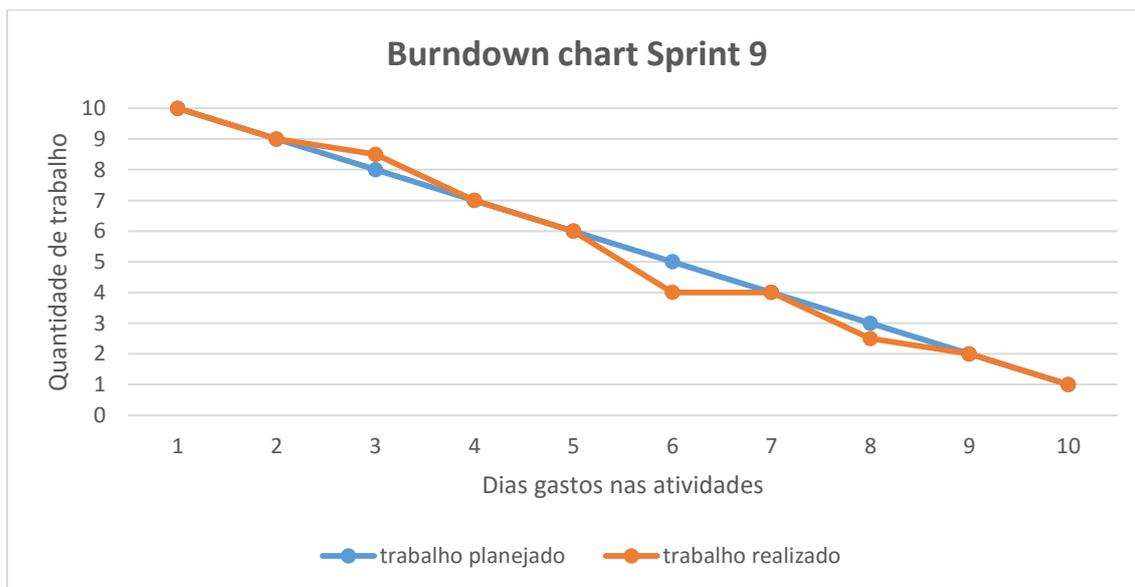


Figura 38 – Burndown chart da Sprint 9.

No gráfico ilustrado na Figura 39, observamos que no segundo dia houve problemas na codificação em CSS, devido à complexidade de implementação, mas tudo foi resolvido rapidamente pelo programados e as atividades seguiram normalmente. Algumas tarefas foram finalizadas antes do prazo, como podemos ver, nos dias 5 a 7 e 7 a 9. Portanto a Sprint foi finalizada como esperado.

Finalmente o processo de implementação foi finalizado com sucesso seguindo as referências do método ágil Scrum, escolhido para organizar o processo de desenvolvimento.

5 VALIDAÇÃO E TESTES

O objetivo desse capítulo é verificar se o sistema desenvolvido tem um perfeito funcionamento e se o produto final atende as expectativas dos usuários. De acordo com (SOMMERVILLE, 2007) podem ser usados dois tipos de testes, o teste de validação e o teste de defeitos. O teste de defeitos tem como objetivo encontrar erros na execução do sistema, e o teste de validação verifica se o sistema desenvolvido atende tudo aquilo que os usuários requisitaram.

Na etapa de verificação e validação é onde procura-se possíveis erros do sistema. Nessa fase o desenvolvedor efetuou repetitivos testes realizados manualmente a fim de encontrar e corrigir tais problemas de funcionalidade. Foi necessária experiência e conhecimento aprofundado do código para identificar e corrigir com eficiência os erros, comuns e outros mais complexos, do sistema. Erros comuns como falha ao salvar no banco de dados, botões de funções que não executam nada, erros de interface, foram rapidamente resolvidos. Mas os erros nas classes, erros de javascript, erros nas funções de PHP, por serem erros morosos de resolver demandaram mais tempo e conhecimento no momento de procura e correção de erros, precisando assim de um maior domínio do código do sistema. Quando um erro foi encontrado o mesmo foi corrigido e o sistema revalidado. Foi testado cada campo destinado as entradas dos dados e suas interfaces, realizados com profissionais da academia, instrutores e fisioterapeutas, verificando se o modo como são inseridos está intuitivo para o usuário, as conexões com o banco de dados foram constantemente testadas a fim de garantir que os dados estão sendo armazenados de forma correta, em sua respectiva tabela, em seu respectivo campo. As opções de editar e excluir também foram testadas no mesmo modo, verificando se o método de edição está intuitivo ao usuário e se a edição está sendo armazenada de forma correta. O modulo que mostra os resultados das avaliações foi testado com mais atenção, por ser a parte mais complexa do sistema, garantindo que o método de saída dos dados está claro e completo para o usuário.

Condições de erros foram inseridas em praticamente todos os processos de cadastros, evitando que dados incorretos sejam inseridos e que o usuário não esqueceu de preencher nenhum campo do cadastro.

Manualmente os testes foram feitos, passo a passo, tela por tela. Os testes foram executados com dados fictícios, apenas para testar as funcionalidades.

Com as execuções dos testes foi garantido que o sistema efetua o cadastro das informações (dados dos clientes e instrutores, administradores, avaliações, agendamentos e movimentos financeiros), edição as informações, exclusão das informações. Todo o processo de teste foi executado junto a etapa de implementação, facilitando assim o encontro dos erros. Após testar o sistema, uma versão foi disponibilizada para o usuário validar e verificar se atendia suas expectativas. Com isso as telas de cadastro se parecem muito com as fichas utilizadas pela academia, facilitando sua interação com o sistema.

5.1 Tipos de teste

1. Testes no banco de dados

- Verificar a consistência do banco de dados.
- Verificar as colunas e as linhas das tabelas do banco de dados.
- Verificar as classes do banco de dados.
- Verificar as chaves estrangeiras e primárias das tabelas do banco de dados.
- Verificar as conexões com o banco de dados.
- Verificar se os dados estão sendo inseridos no banco.
- Verificar se as informações podem ser recuperadas.
- Verificar se os dados não estão redundantes.

2. Teste de funcionalidade

- Verificar se o usuário consegue de cadastrar os dados necessários.
- Verificar se o usuário consegue editar as informações que desejar.
- Verificar se o usuário consegue excluir as informações que desejar.
- Verificar se o usuário consegue usar o módulo de comparação das avaliações dos clientes.
- Verificar se consegue fazer os agendamentos.
- Verificar se é possível fazer os movimentos diários.

3. Teste de configuração

- Verificar se o hardware é adequado a utilização do sistema.

5.2 Casos de teste

Nessa seção são mostrados os casos de teste realizados e sua forma que foram testados. Na Tabela 12 podemos ver o caso de teste do banco de dados.

	CASO DE TESTE
Teste	1. Teste do banco de dados
Objetivo	Verificar o desempenho do banco de dados
Entradas	Dados inseridos em diversos campos e formatos no banco de dados
Resultados desejados	Dados inseridos com sucesso no banco de dados Todos os dados sejam preenchidos Conexões com o banco funcionando bem Dados armazenados nos seus respectivos campos das tabelas
Início	Desenvolvimento correto do banco de dados Execução do banco de dados
Execução	Inserir os dados nas tabelas do banco de dados Execução de <i>queries</i> para testar se tudo está correto
Interrupções	Se houver alguma interrupção, tentar corrigir o problema e realizar o teste novamente
Resultados	Testes realizado com sucesso

Tabela 12 – Caso de teste do banco de dados.

Na Tabela 13 podemos ver a execução do teste de funcionalidade.

	CASO DE TESTE
Teste	2. Teste de funcionalidade
Objetivo	Verificar se os usuários conseguem utilizar as interfaces do sistema
Entradas	As interfaces são apresentadas para os usuários
Resultados desejados	Os usuários compreenderam as telas e suas funções Os dados foram inseridos nos campos, corretamente Todos os campos foram preenchidos O visual das telas agradaram os usuários
Início	Preparar as páginas para o acesso Preencher as informações necessárias
Execução	Todos os campos devem ser preenchidos Tentar inserir dados inválidos Tentar salvar sem preencher todos os dados Efetuar os cadastros Editar, excluir informações
Interrupções	Se houver alguma interrupção, tentar corrigir o problema e realizar o teste novamente
Resultados	Erros nas tentativas de salvar cadastros incompletos Erros ao preencher dados inválidos
Observações	Todos os erros encontrados foram alertados e corrigidos, e os testes foram executados novamente

Tabela 13 – caso de teste de funcionalidade.

Na Tabela 14 Podemos ver a execução do teste de configuração

	CASO DE TESTE
Teste	3. Teste de configuração
Objetivo	Verificar se o sistema está configurado corretamente
Entradas	Instalar o sistema na academia Verificar se o <i>hardware</i> suporta o sistema
Resultados desejados	Implantar o sistema com sucesso
Início	Executar o sistema
Execução	O XAMPP será instalado O diretório do sistema será criado O sistema será carregado pelo navegador As funcionalidades serão testadas garantindo o perfeito funcionamento do sistema
Interrupções	Se houver alguma interrupção, tentar carregar o sistema novamente e seguir com a configuração
Resultados	O sistema foi instalado com sucesso
Observações	O sistema foi carregado com sucesso e suas funcionalidades foram testadas e executadas com sucesso

Tabela 14 – Caso de teste de configuração.

6 IMPLANTAÇÃO

O processo de implantação do sistema na academia Studio Fitness foi efetuado com a presença do proprietário da empresa com quem foi coletado os pré-requisitos. O sistema foi recebido com animação pelo usuário final, os profissionais da empresa, e foi bem aceito. Primeiramente foi instalado o XAMPP e o Navicat, XAMPP para ser o servidor independente, e o Navicat para gerenciar o banco de dados. Depois de instaladas as ferramentas, o sistema desenvolvido foi instalado no computador principal da academia, que fica na recepção da mesma e foi instalado também no notebook do proprietário da academia. Os dados de clientes reais foram inseridos com o objetivo de observar seu funcionamento. Foi relatado que a semelhança das interfaces das telas com as fichas de papel utilizadas para armazenar as informações dos clientes facilitou muito o uso e compreensão do sistema.

Em conversa com o proprietário, foi relatado que o sistema desenvolvido será de grande importância para a empresa, que auxiliará no processo de gestão dos clientes e suas informações e também na organização das tarefas exercidas pelos profissionais da academia como, fisioterapeutas e instrutores. O processo de alimentação de informações no sistema e seu uso ainda vem sendo feito pelos profissionais da empresa.

7 CONCLUSÃO

Este trabalho teve como objetivo propor uma solução para gerenciar os clientes e demais cadastro de informações na Academia Studio Fitness de Diamantina, Minas Gerais, por meio do desenvolvimento de um Sistema de Informação baseada na Web.

O sistema foi implementado seguindo a metodologia do Scrum e as demais ferramentas utilizadas permitiram que o sistema fosse desenvolvido, alcançando os objetivos por ele oferecidos. Adquiri experiências importantíssimas, como por exemplo, ao iniciar um projeto deve-se atentar para a coleta dos requisitos e definir bem o planejamento de desenvolvimento, fundamental para acompanhar os riscos, os recursos utilizados e o tempo gasto na execução das tarefas necessárias.

A interface das telas foi muito próxima da interface aplicada nas fichas de cadastro utilizadas na academia, isso proporcionou satisfação dos profissionais da academia, instrutores e fisioterapeutas. Uma interface de interação intuitiva é fundamental em um sistema. Portanto, percebeu-se que a metodologia definida e executada no contexto desse trabalho proporcionou organização e coerência, e foi vital para que os objetivos fossem alcançados.

Propostas de continuidade:

- Implementar um servidor próprio da academia e aplicar o sistema no mesmo.
- Melhorar o layout/design do sistema com base nas opiniões dos usuários.
- Disponibilizar o sistema online.
- Implementar um aplicativo mobile para os clientes interagirem melhor pelos seus dispositivos móveis.

8 REFERÊNCIAS

- ALECRIM, Emerson. **Conhecendo o Servidor Apache**. Artigo publicado em, v. 15, 2006.
- BAZZOTTI, Cristiane; GARCIA, Elias. A importância do sistema de informação gerencial para tomada de decisões. **Cascavel**. Disponível em:< <http://www.unioeste.br/campi/cascavel/ccsa/VISeminario/Artigos%20apresentados%20em%20Comunica%C3%A7%C3%B5es/ART>, v. 203, p. 1-5, 2007.
- CASTELLS, Manuel. Internet y la sociedad red. **La factoría**, v. 14, p. 15, 2001.
- DE CARVALHO, José Oscar Fontanini. O papel da interação humano-computador na inclusão digital. **Transinformação**, v. 15, n. 3, 2012.
- KLIGERMAN, Débora Cynamon et al. Sistemas de indicadores de saúde e ambiente em instituições de saúde. **Ciênc. saúde coletiva**, v. 12, n. 1, p. 199-211, 2007.
- MÉNDEZ, Juan Manuel Álvarez; PAIS, Paulo. **Avaliar para conhecer, examinar para excluir**. 2002.
- MITCHELL, Gordon. Problems and fundamentals of sustainable development indicators. 2006.
- NIEDERAUER, Juliano. **Desenvolvendo websites com PHP**. São Paulo: Novatec, 2004.
- OFFUTT, Jeff. Quality attributes of web software applications. **IEEE software**, n. 2, p. 25-32, 2002.
- OLIVEIRA, Djalma de Pinho Rebouças de. Sistemas de informações gerenciais: estratégicas, táticas, operacionais. **São Paulo: Atlas**, 2004.
- PRESSMAN, Roger S. **Engenharia de software**. McGraw Hill Brasil, 2011.
- SCHWABER, K.; BEEDLE, Mike. **Scrum: Software Development with Scrum**. 2002.
- SILVA, Maurício Samy. **Criando sites com HTML: sites de alta qualidade com HTML e CSS**. Novatec Editora, 2008.

SILVA, Maurício Samy. **CSS3: desenvolva aplicações web profissionais com uso dos poderosos recursos de estilização das CSS3**. Novatec Editora, 2011.

SUEHRING, Steve. Mysql: a bíblia. In: **MySQL: a bíblia**. Campus, 2002.

TAKASHINA, Newton Tadachi; FLORES, Mário César Xavier. **Indicadores da qualidade e do desempenho: como estabelecer metas e medir resultados**.

Qualitymark Editora Ltda, 1996.

TRZESNIAK, Piotr. Indicadores quantitativos: reflexões que antecedem seu estabelecimento. **Ciência da Informação**, v. 27, n. 2, p. 159-164, 1998.