



FACULDADE DE TECNOLOGIA DE AMERICANA

Curso de Graduação em Análise e Desenvolvimento de Sistemas

CLIZAN WILLIAN SANTANA LOPES

SISTEMA DE INFORMAÇÃO PARA ACADEMIA

Americana

2017

CLIZAN WILLIAN SANTANA LOPES

SISTEMA DE INFORMAÇÃO PARA ACADEMIA

Monografia apresentada a Faculdade de Tecnologia de Americana do Estado de São Paulo, como requisito parcial para obtenção do Grau de Análise e Desenvolvimento de Sistemas.

Área de concentração: Programação.

Americana, SP

2017

FICHA CATALOGRÁFICA – Biblioteca Fatec Americana - CEETEPS
Dados Internacionais de Catalogação-na-fonte

L851s LOPES, Clizan Willian Santana

Sistema de informação gerencial para academia de musculação. / Clizan Willian Santana Lopes. – Americana: 2017.

57f.

Monografia (Curso de Tecnologia em Análise e Desenvolvimento de Sistemas) - - Faculdade de Tecnologia de Americana – Centro Estadual de Educação Tecnológica Paula Souza

Orientador: Prof. Esp. Antonio Alfredo Lacerda

1. Sistemas de informação 2. Inovação tecnológica I. LACERDA, Antonio Alfredo II. Centro Estadual de Educação Tecnológica Paula Souza – Faculdade de Tecnologia de Americana

CDU: 681.518

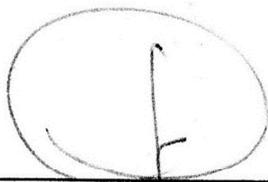
Clizan Willian Santana Lopes

SISTEMA DE INFORMAÇÃO PARA ACADEMIA

Trabalho de graduação apresentado como exigência parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas pelo CEETEPS/Faculdade de Tecnologia – FATEC/ Americana.
Área de concentração: Sistema de Informação.

Americana, 26 de junho de 2017.

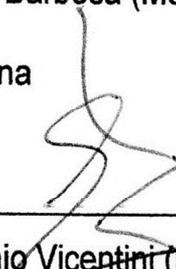
Banca Examinadora:



Antônio Alfredo Lacerda (Presidente)
Especialista
Fatec Americana



Anderson Luiz Barbosa (Membro)
Mestre
Fatec Americana



Eduardo Antonio Vicentini (Membro)
Mestre
Fatec Americana

DEDICATÓRIA

Dedico este trabalho à minha família, em especial aos meus pais, minha noiva e amigos de sala que me auxiliaram e incentivaram o meu crescimento profissional.

AGRADECIMENTOS

Agradeço a Deus pela minha vida, e pela a oportunidade de estar realizando a conclusão deste trabalho.

A minha noiva Jessica Fernanda dos Santos Carnavalli e aos meus pais Elizabete Batista de Santana e Nilton Cezar Lopes que me incentivaram incondicionalmente em todos os momentos difíceis desta jornada, fornecendo abundantemente motivação, carinho e compreensão a fim de que pudesse desenvolver e concluir esta missão.

Ao meu orientador Antônio Alfredo Lacerda por me auxiliar no decorrer no trabalho de conclusão do curso de análise e desenvolvimento de sistemas.

Aos meus amigos e colegas de trabalho que de alguma maneira ajudaram na coleta de informações para a conclusão deste trabalho.

RESUMO

Durante muitos anos, a gestão da academia foi muito simples, feita de papéis impressos que poderiam ser facilmente deslocados e / ou danificados. O avanço tecnológico e o desenvolvimento de sistemas de gerenciamento de informações nas empresas, facilitou o gerente, fornecendo a otimização necessária do controle de informações. O principal objetivo deste trabalho é desenvolver uma aplicação *web*, cuja finalidade é auxiliar no gerenciamento de academias de musculação. Os conceitos utilizados nas análises e desenvolvimento buscam ajudar o usuário a tomar decisões através dos relatórios gerados. O trabalho propõe uma manobra criativa para que empresas esportivas e atividades de atividades físicas possam ter um aliado para suas empresas. A inovação provocada pelas tecnologias de informação e comunicação cria uma demanda de sistema mais eficiente e eficaz todos os dias. Aqui reside a necessidade do presente trabalho.

Palavras-Chaves: Avanço tecnológico; Academia; Desenvolvimento; Sistema de informação.

ABSTRACT

For many years, the management of the gym was very simple, made of printed papers that could be easily displaced and / or damaged. Technological advancement and the development of information management systems in companies, facilitated the manager, providing the necessary optimization of information control. The main objective of this work is to develop a web application, whose purpose is to assist in the management of bodybuilding academies. The concepts used in the analysis and development seek to help the user to make decisions through the generated reports. The work proposes a creative maneuver so that sports companies and activities of physical activities can have an ally for their companies. Innovation brought about by information and communication technologies creates a more efficient and effective system demand every day. Herein lies the necessity of the present work.

Keywords: Technological advancement; Academy; Implementation; Information System.

LISTA DE FIGURAS

Figura 1: Estrutura dos cinco componentes	3
Figura 2: Estrutura do sistema de informação.....	4
Figura 3: Evolução do sistema de informação.....	5
Figura 4: Classificação dos sistemas de informação.....	6
Figura 5: Sintaxe HTML	14
Figura 6: Script do PHP junto ao HTML	16
Figura 7: Código HTML e JavaScript	18
Figura 8: Código HTML e JavaScript (link externo).....	18
Figura 9: Link de acesso externo CSS	19
Figura 10: CSS aplicado na seção head da página HTML.....	20
Figura 11: Símbolo do MER	23
Figura 12: Exemplo tabela.....	24
Figura 13: Área de atuação do IHC	26
Figura 14: Diagrama de Caso de Uso	31
Figura 15: Diagrama de Classe.....	32
Figura 16: Diagrama de Sequência Tela Login	33
Figura 17: Diagrama de sequência tela cadastro aluno	34
Figura 18: Diagrama de sequência tela avaliação.....	34
Figura 19: Diagrama de sequência tela treino.....	35
Figura 20: Diagrama de sequência tela relatório.....	36
Figura 21: Diagramas de Atividade tela login	37
Figura 22: Diagrama de atividade tela cadastro aluno	38
Figura 23: Diagrama de atividade tela avaliação.....	39
Figura 24: Diagrama de atividade tela treino.....	40
Figura 25: Diagrama de atividade tela relatório.....	40
Figura 26: MER do sistema SIGAM	41
Figura 27: Tela login SIGAM	42
Figura 28: Tela menu do sistema (SIGAM)	43
Figura 29: Tela cadastro Cliente	43
Figura 30: Tela de avaliação	44
Figura 31: Ficha treino aluno.....	44
Figura 32: Tela relatório	45

SUMÁRIO

INTRODUÇÃO	1
1. SISTEMA DE INFORMAÇÃO	3
1.1. SISTEMA DE INFORMAÇÃO NAS ORGANIZAÇÕES	4
1.2. TIPOS DE SISTEMAS DE INFORMAÇÃO	5
2. ENGENHARIA DE SOFTWARE	10
2.1. UML	10
2.2. MODELAGEM DE SOFTWARE	11
2.3. DIAGRAMAS	12
3. LINGUAGENS DE PROGRAMAÇÃO	13
3.1. HTML	13
3.1.1. Sintaxe HTML	13
3.2. PHP	15
3.2.1. Sintaxe PHP	15
3.3. JAVA SCRIPT	16
3.3.1. Sintaxe JavaScript	17
3.4. CSS	19
4. BANCO DE DADOS	21
4.1. HISTÓRIA DO BANCO DE DADOS	21
4.2. MODELAGEM DE DADOS	22
4.3. TABELAS	24
5. INTERFACE HOMEM MÁQUINA	26
5.1. INTERFACE	27
6. SISTEMA DE INFORMAÇÃO PARA ACADEMIA (PROJETO - SIA)	30
6.1. DIAGRAMA DE CASO DE USO	30
6.2. DIAGRAMA DE CLASSE	31
6.3. DIAGRAMAS DE SEQUÊNCIAS	33
6.4. DIAGRAMAS DE ATIVIDADES	36
6.5. BANCO DE DADOS MYSQL	41
6.6. INTERFACE SAI	42
7. CONCLUSÃO	46
REFERÊNCIAS	47

INTRODUÇÃO

Atualmente, com os avanços tecnológicos e o seu crescimento no mercado, um fator que está bastante evidente é a importância de sistemas integrados no ambiente empresarial, uma vez que, é nesta específica área de T.I. (Tecnologia da Informação) que o sistema de informação (SI) se apresenta como o principal responsável por impor novos conceitos no desenvolvimento de softwares, por exemplo: redução na rotina sistêmica, aumento da produtividade, melhorias referentes às manutenções, entre outros.

Um sistema de informação (SI) possui auxílio e recursos disponibilizados pela área de T.I., tais como o acesso, armazenamento e recuperação dos dados, em virtude de sua flexibilidade e facilidade. A implantação de sistemas de informação tem gerado grande impulso nos processos e rotinas de empresas dos mais variados tipos e ramos de atuação. De forma geral, pode-se dizer que um sistema de informação busca atrelar as necessidades que o cliente possui correlação a sistemas eficientes, contendo somente o necessário para armazenamento de informação e modos de acessibilidade dos dados.

O objetivo deste trabalho é desenvolver uma aplicação *web*, que irá auxiliar no controle de gestão de uma academia, tendo como diferencial agilidade, precisão e baixo custo, podendo ser acessado através de dispositivos móveis (*WAP*) e online (*Website*).

Este trabalho busca sanar problemas relacionados ao gerenciamento de informação, fazendo o armazenamento seguro dos dados, proporcionando uma fonte rápida para obtenção de dados.

Para a realização deste trabalho será utilizado bibliografias impressas e digitais referentes a sistema de informação, engenharia de software, linguagens de programação *web* e banco de dados.

Este trabalho foi estruturado em sete capítulos, sendo que o primeiro conceitua o sistema de informação e seus tipos, o segundo traz uma definição sobre engenharia de software e abrange sobre uml, modelagem de software e diagramas, o terceiro descreve linguagens de programação *web* além de descrever HTML, PHP, JavaScript e CSS, além de descrever suas sintaxe de programação, o quarto capítulo aborda o que venha ser banco de dados, bem como sua história, como

realiza a modelagem de dados e como surge uma tabela, o quinto traz a definição do que é interface homem e maquina (HCI) e o que venha a ser interface, o sexto conceitua sobre o desenvolvimento do sistema SIGAM, módulos propostos, diagramas que compõem o sistema, bancos de dados utilizado e interfaces do software e, por fim, o sétimo trata-se da conclusão de todo o trabalho.

1. SISTEMA DE INFORMAÇÃO

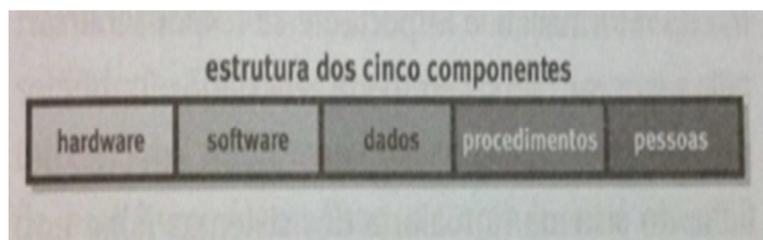
Um sistema de informação é um conjunto de componentes que interagem entre si a fim de produzir, coletar, armazenar e distribuir informações. Tais recursos auxiliam na tomada de decisões de uma organização, pois além de conter dados importantes, auxiliam seus usuários a analisar problemas complexos e criar novos produtos.

Um sistema de informação (SI) pode ser definido tecnicamente como um conjunto de componentes inter-relacionados que coletam (ou recuperam), processam, armazenam e distribuem informações destinadas a apoiar a tomada de decisões, a coordenação e o controle de uma organização. (LAUDON E LAUDON, 2011, p.12).

Pode-se dizer que um sistema de informação é um complexo de informações que integram para um determinado fim, cujo objetivo desta interação é criar novas informações. Para que estas informações possam ser processadas é necessário que haja um conjunto de componentes (Figura 1).

Segundo Kroenke (2013, p.28) “esses cinco componentes estão presentes em todo sistema de informação, do mais simples ao mais complexo”.

Figura 1: Estrutura dos cinco componentes

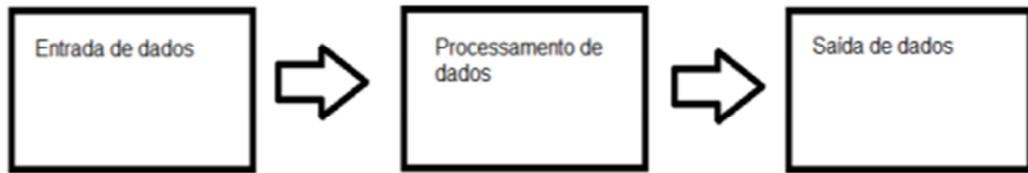


Fonte: David M. Kroenke, 2013, p.29.

“Em essência, cada subsistema age como um coletor e processador de dados para a produção de determinadas informações” (BIO. 1991, p. 29).

Os componentes listados anteriormente são responsáveis por operar qualquer sistema computacional. Conceitualmente é possível verificar a estrutura de um sistema de informação composto por dados esses elementos interagem entre si para uma determinada finalidade criar um novo processo.

Figura 2: Estrutura do sistema de informação



Fonte: Sérgio Rodrigues Bio, 1991, p. 29.

Conclui-se que um sistema de informação é um conjunto de procedimentos (subsistema), em que todas as ações realizadas são de forma repetitiva, iniciada, controlada, executada e finalizada, sua essência tem por base a sequência cronológica e a sua implementação é transformada em resultados de ações.

1.1. SISTEMA DE INFORMAÇÃO NAS ORGANIZAÇÕES

“Sistemas de informação empresariais são sistemas de informação que abrangem toda a empresa e prestam suporte às atividades de diversos departamentos.” (KROENKE, 2013, p. 160).

O sistema de informação é um fator decisivo para todas as organizações, pois seus recursos são importantes e indispensáveis para os objetivos e aplicações da empresa. A utilização destes recursos faz com que as informações tornam-se mais organizadas, seguras, gravadas e recuperadas, assim possibilita aos gestores agir com mais segurança e acerto na tomada de decisões.

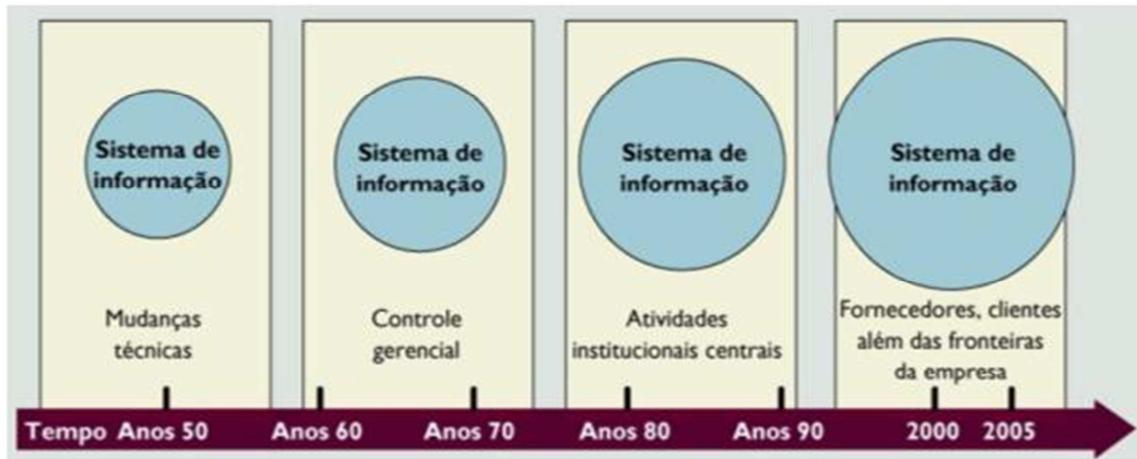
Segundo Laudon e Laudon (2011, p.14),

Os sistemas de informação são parte integrante das organizações. E, embora nossa tendência seja pensar que a tecnologia da informação está alterando as organizações e empresas, trata-se, na verdade, de uma via de mão dupla: a história e a cultura das empresas também determinam como a tecnologia é e deveria ser usada.

Em meados de 1950, muitas empresas administravam suas informações através de registro impressos, mas, com o avanço tecnológico muitas companhias passaram obter sistemas computadorizados, os dados não eram mais armazenados em papéis, e na medida em que as empresas passaram a investir em tecnologia, o empreendimento em sistemas tornou-se indispensável para a realização de

atividades diárias. É possível notar a dimensão em que um sistema de informação proporcionou aos clientes, fornecedores e até mesmo concorrentes do mesmo setor.

Figura 3: Evolução do sistema de informação



Fonte: Keneth C. Laudon e Jane P. Laudon, 2006 p. 17.

Até a metade da década de 1950, as empresas administravam tudo em registros impressos e divulgavam suas informações em papel. Durante os últimos 50 anos, cada vez mais as informações organizacionais e o fluxo delas entre os principais atores dos negócios foram computadorizados. (LAUDON; LAUDON 2011, p, 41).

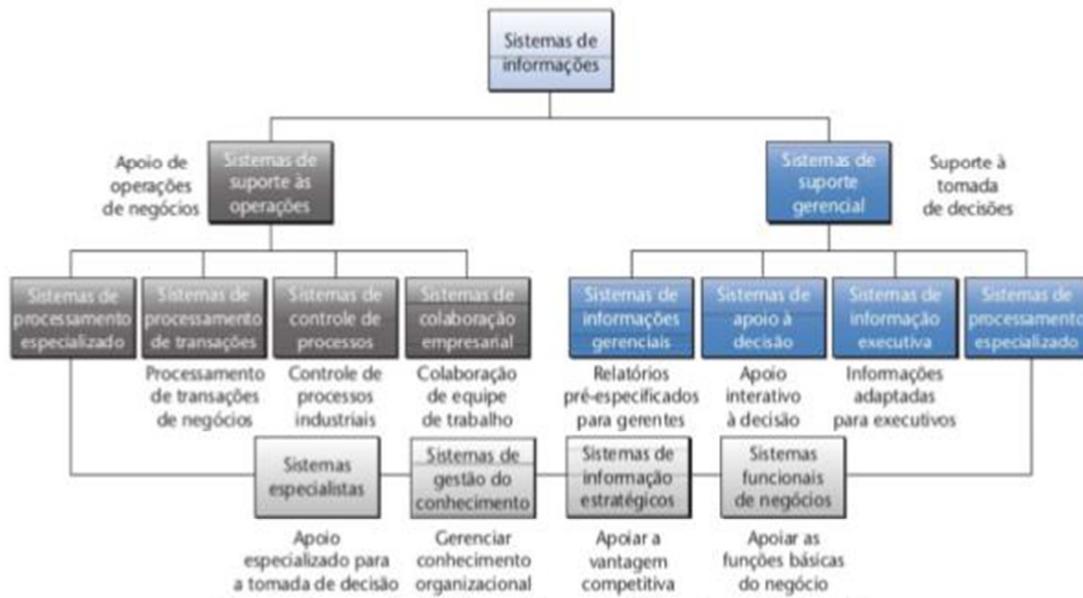
Em virtude dos fatos mencionados acima, as organizações encontraram em sistemas de informação oportunidades de aprimorar-se de tal forma que pudesse enfrentar seus muitos desafios e problemas.

1.2. TIPOS DE SISTEMAS DE INFORMAÇÃO

“Uma empresa tem sistemas para apoio à tomada de decisão e às atividades de diferentes níveis da organização”. (LAUDON; LAUDON, 2011, p. 42).

Em uma organização existem diversos tipos de sistemas contribuindo para cada nível organizacional, e, por sua vez o administrador necessita compreender o papel de cada um, pois são estas habilidades que auxiliam no apoio a tomada de decisões e a forma em que ações comportam-se em determinados níveis empresariais.

Figura 4: Classificação dos sistemas de informação



Fonte: James A. O'Brien e George M. Marakas, 2013, p.11.

Os sistemas de apoio às operações têm por objetivo processar transações de negócios, controle de processos industriais, apoiarem a comunicação e colaboração e a atualização de bancos de dados corporativos. Os níveis pertencentes ao setor de operações são classificados da seguinte forma:

Para O'Brien e Marakas (2013, p. 11-12) "Os sistemas de processamento de transação são um exemplo importante de sistemas de suporte de operações que registram e processam os dados que resultam de transações de negócios".

- **Sistemas de processamento de transações:** A função deste sistema é coletar, armazenar e executar informações rotineiras, repetitivas planejadas e processadas pela empresa. Exemplos: Controle de estoque, sistema de processamento de venda e de contabilidade.

Segundo O'Brien e Marakas (2013, p. 12) "Os sistemas de controle de processo monitoram e controlam processos físicos".

- **Sistemas de controle de processo:** Controlam e monitoram os processos físicos constantemente através de dispositivos ligados ao computador. Exemplos: Fabricação de veículos e energia.

"Os sistemas de colaboração empresarial aprimoram as comunicações e a produtividade da equipe e de grupos de trabalho e incluem aplicações que são por

vezes chamadas *sistemas de automação de escritório*". (O'BRIEN; MARAKAS, 2013, p. 12).

- **Sistema de colaboração empresarial:** São softwares que facilitam a execução de trabalhos em grupos dentro da empresa. Exemplos: Sistemas de e-mail e chat.

Os sistemas gerenciais se preocupam em fornecer informações que auxiliem os administradores a tomarem decisões eficazes. As subdivisões pertencentes a este grupo são classificadas da seguinte maneira:

"Os sistemas de informação gerencial (MIS) fornecem informações na forma de relatórios e exibições em telas para gerentes e muitos profissionais de negócios". (O'BRIEN; MARAKAS, 2013, p. 12).

- **Sistema de informação gerencial (SIG):** Sistemas capazes de resumir e fornecer informações através de relatórios para dar auxílio à tomada de decisão e ajudar a organização atingir suas metas, fornecendo aos níveis hierárquicos mais elevados um melhor controle e planejamento. Exemplos: Sistema de relatórios de vendas e demonstrativos financeiros mensais.

Para O'Brien e Marakas (2013, p. 13) "Os sistemas de suporte de decisão (*decision support systems* – DSS) dão suporte direto do computador aos gerentes durante o processo de tomada de decisão [...]".

- **Sistema de apoio à decisão (SAD):** A função deste sistema é traçar uma direção para o planejamento estratégico, colaborando para resoluções de problemas organizacionais rotineiros e conseqüentemente auxilia no gerenciamento de dados específicos. Exemplos: Planilhas eletrônicas e sistemas de análise estática.

Os sistemas de informação executiva (*executive information systems* – EIS) fornecem aos executivos e gerentes informações fundamentais a partir de uma ampla variedade de fontes internas e externas em exibições em tela de fácil utilização. (O'BRIEN; MARAKAS, 2013, p.13).

- **Sistema de informação executiva (SIE):** São sistemas tecnológicos computadorizados projetados para apoiar os altos executivos da organização, além de possuir características similares ao sistema de informação gerencial e dos sistemas de apoio à decisão, fornecendo informações fundamentais de fácil acesso, podendo ser obtida rapidamente e identificar problemas e determinar objetivos estratégicos. Exemplos: Sistemas de fácil acesso às

análises do desempenho do negócio para dar suporte ao planejamento estratégico.

Existem ainda outras categorias de sistemas, que prestam suporte tanto às aplicações operacionais quanto gerenciais, esses tipos são definidos a seguir:

“Os sistemas especialistas, por exemplo, podem fornecer recomendações adequadas a pequenas tarefas operacionais, como diagnóstico de equipamento, ou a decisões administrativas, como o gerenciamento da carteira de empréstimo”. (O'BRIEN; MARAKAS, 2013, p. 14).

- **Sistemas especialistas:** Aplicam habilidades de raciocínio, conhecimento e experiência, que são destinadas a solucionar problemas em campos específicos. Exemplo: treinamento de grupos de pessoas, monitor de processo.

Segundo O'Brien e Marakas (2013, p. 14) “Os sistemas de gestão de conhecimento baseiam-se no conhecimento e apoiam a criação, organização e disseminação de conhecimento de negócios para empregados e gerente em toda a empresa”.

- **Sistema de gestão de conhecimentos:** Esse tipo de sistema colabora com a identificação, criação e organização de conhecimento dos negócios dentro da empresa. Exemplo: Acesso a internet às melhores práticas de negócios.

Para O'Brien e Marakas (2013, p. 14) “[...] os sistemas de informação estratégicas aplicam a tecnologia da informação a produtos, serviços ou processos de negócios de uma empresa para ajuda-la a obter vantagem estratégica sobre seus concorrentes [...]”.

- **Sistemas de informação estratégica:** Ajuda a empresa estabelecer objetivo estratégico, visando analisar as necessidades da empresa e obter vantagens competitivas sobre seus concorrentes. Exemplo: Sistema de resposta eficiente ao cliente, rastreamento de embarque e sistemas e-commerce.

“[...] sistemas de informação integrados ou interfuncionais que fornecem uma variedade de funções”. (O'BRIEN; MARAKAS, 2013, p.14)

- **Sistemas funcionais do negócio:** Concentra-se em aplicações operacionais e administrativas básicas. Exemplo: Sistema de suporte a aplicação contábeis, financeiras.

Por este motivo a importância de um administrador conhecer e entender os diferentes tipos de sistemas existentes, para obter vantagem ou reduzir desvantagens competitivas em relação aos concorrentes. Além disso, é necessário saber onde será onde ele será implementado e como será a sua utilização, em desktops, notebooks ou em dispositivos móveis.

2. ENGENHARIA DE SOFTWARE

A engenharia de software é uma área da computação responsável por vincular inúmeros fatores para que possam auxiliar no desenvolvimento de um software. Tal conceito foi exposto por Fritz Bauer no ano de 1969 em uma conferência do Comitê de Ciência da Organização do Trabalho Atlântico Norte (Ocean).

“Há 60 anos, ninguém poderia prever que o software se tornaria uma tecnologia indispensável para negócios [...]” (PRESSMAN, 2016, p. 2).

De acordo com o autor, antigamente não era possível evidenciar a importância de uma aplicação da engenharia de software no âmbito empresarial, portanto, atualmente é possível notar os benefícios que a implantação de um software trouxe ao mercado, dentre eles destacam-se a praticidade, redução de tempo e custos. Estes recursos são utilizados em determinados meios, tais como telecomunicações, meios de transportes, ciências humanas, ciências contábeis, entre outras.

“A engenharia de software engloba processos, métodos e ferramentas que possibilitam a construção de sistemas complexos baseados em computador dentro do prazo e com qualidade”. (PRESSMAN, 2011, p. 48).

Com base na concepção acima, pode-se afirmar que a engenharia de software abrange ferramentas que auxilia as atividades, métodos que orientam a realização destas atividades, processos que definem as atividades, bem como o produto e a qualidade de processo do software. Deste modo, é possível garantir a qualidade e a produtividade de um software.

2.1. UML

Uma das áreas que vem possuindo maior destaque pela sua praticidade e auxílio a engenheiro de software é a UML (*Unified Modeling Language*).

“A UML – Unified Modeling Language ou Linguagem de Modelagem Unificada – é uma linguagem visual utilizada para modelar softwares baseados no paradigma de orientação a objetos.” (GUEDES, 2011, p.19).

Baseado nas informações de Guedes, a UML é uma linguagem de modelagem visual composta por vários diagramas que tem a função de auxiliar no desenvolvimento de um software.

A UML é uma linguagem unificada de modelagem orientada a objetos. Surgiu após a fusão de outras três metodologias, o método de Booch, OMT (*Object Modeling Technique*) de Rumbaugh e OOSE (*Object - Oriented Software Engineering*) de Jacobson conhecido como os “Os três amigos”. Seu objetivo é auxiliar e exemplificar de modo simples a visualização do desenho e da interação entre objetos.

A estruturação inicial do projeto se desenvolveu através da junção dos métodos de Booch e OMT de Jacobson, resultando o lançamento de um novo método unificado no final de 1995. Subsequentemente, Rumbaugh se junta a Booch e Jacobson na Rational Software (financiadora do projeto), implementando e unificando seu método orientado a objetos (OOSE), resultando no lançamento da primeira versão da UML propriamente dita, no ano de 1996.

Ao decorrer dos anos a UML, após o recebimento de feedbacks sobre a ferramenta utilizada pela área de engenharia de software, foi se desenvolvendo (aperfeiçoando) e atualmente a sua versão disponível no mercado é a 2.4.1. A ferramenta tem como objetivo auxiliar a visualização da comunicação entre objetos (características, comportamento e estrutura que o software irá conter). Aplicando um conjunto de técnicas de notação gráficas é possível criar modelos visuais do software de sistemas concorrentes e distribuídos.

2.2. MODELAGEM DE SOFTWARE

A real necessidade de se modelar um software dar-se-á quando este possui um propósito (finalidade), seja da aplicação mais simples até a mais complexa, apesar de ser uma modelagem técnica. Portanto é possível basear-se nas funcionalidades do sistema como: tamanho, complexidade, prototipação, análise de requisitos, custos, entre outros.

“A modelagem é uma parte central de todas as atividades que levam à implantação de um bom software.” (BOOCH; RUMBAUGH; JACOBSON, 2006, p.4).

Mediante a referência citada anteriormente é possível afirmar que a ferramenta UML propôs ao mercado tecnológico novas formas de planejar um

software, apesar de possuir termo e informações técnicas é evidente que ao ser utilizado traz benefícios na criação de um sistema, minimizando falhas e até mesmo alterações do projeto.

2.3. DIAGRAMAS

“São desenhados para permitir a visualização de um sistema sob diferentes perspectivas; nesse sentido, um diagrama constitui uma projeção de um determinado sistema.” (BOOCH; RUMBAUGH; JACOBSON, 2006, p.26).

Na UML cada diagrama tem sua finalidade, dentre de seus muitos objetivos é possível destacar como o principal que a usabilidade dos diagramas é o fornecimento de múltiplas visões do sistema a ser modelado, pois além de permitir que cada diagrama complemente o outro (anterior) e atinge a completude (qualidade) da modelagem.

3. LINGUAGENS DE PROGRAMAÇÃO

Para Scheidt (2015, p. 1) o "Desenvolvimento *web* é matéria fundamental dentro dos cursos de informática, devido à popularização do acesso à internet e à consequente demanda por aplicações do tipo *web*".

De acordo com o pensamento do autor Scheidt, podemos afirmar que a linguagem *web* são utilizadas para desenvolver páginas ou aplicações que rodam na *web*, dentre elas destacam o HTML, PHP, JavaScript e o CSS. Mais adiante veremos as definições destas linguagens de programação.

3.1. HTML

De acordo com Carril (2012, p.3) "O HTML é uma linguagem de marcação, utilizada para formatar páginas *Web* com texto e informações separadamente".

Atualmente umas das estruturas mais utilizadas para o desenvolvimento *web*, o HTML é denominado uma linguagem de marcação cuja função é formatar páginas *web* com textos e dados independentes, sendo assim é possível denomina-la como uma plataforma poderosa, pois consegue unir outras linguagens *web* ao mesmo projeto.

"A home Page é a primeira página de um site, ou a página de entrada." (CARRIL, 2012, p.3).

Baseado na informação citado por Carril, todos os sites eletrônicos disponíveis ou que venha a ser desenvolvido devem possuir uma home Page (página inicial). Estas páginas são normalmente chamadas de index ou default, quando utilizado na linguagem de marcação HTML, cujo objetivo é realizar navegações (link) com outras interfaces que o projeto venha a consistir.

3.1.1. Sintaxe HTML

Segundo o autor Silva (2015, p. 41) "Uma tag HTML é constituída por uma ou mais letras, ou letra e número, que devem ser escritas entre os sinais < e >. A tag de fechamento é igual à tag de abertura com o sinal de barra / na frente".

Pode-se afirmar que as marcações utilizadas para formatação de páginas *web* ocorrem por meio de tags, estas por sua vez aperfeiçoam os *layouts* das

páginas já desenvolvidas ou em fase de desenvolvimento e são responsáveis pela dinamicidade que dão à página *web*. Elas são representadas por letras e números e impreterivelmente estão contidas entre os símbolos < e >. O que determina a finalização de uma tag é a utilização da barra /, este é o diferencial da tag de abertura.

“No documento HTML possui duas partes: head (cabeçalho) e body (corpo)”. (CARRIL, 2012, p.7).

Os códigos HTML são escritos em duas seções que desempenham funcionalidades diferentes, estas práticas possibilitam uma melhor visualização do software como um todo, uma das informações mais importante está localizada no cabeçalho (home) é nela será atribuído o nome da página através da tag <title> (título), o desenvolvimento de formulários e, ou *layouts* do *website* são desenvolvidos na seção body (corpo).

Conforme visto nesta seção para criar-se uma pagina HTML é necessário possuir vários elementos dentre eles a escolha de ferramentas capazes de realizar tal ação. Uma das ferramentas mais utilizada para o desenvolvimento *web* é o bloco de notas, ferramenta nativa do sistema operacional Windows.

Figura 5: Sintaxe HTML

```
<html>
  <head>
    <title> Olá mundo </title>
  </head>
  <body>
    Olá mundo!
  </body>
</html>
```

Fonte: Autor do Trabalho (2016)

Como é possível analisar, a imagem (figura 5) demonstra a estrutura utilizada para a criação de uma página HTML, isto é, todo documento cuja extensão for .html deverá conter esta estrutura, para que assim o navegador possa interpreta-lo e executa-lo.

3.2. PHP

“O PHP surgiu como uma linguagem representada por um subconjunto de scripts executados sobre a linguagem Perl, criada por Rasmus Lerdof em 1994.” (RAMOS; SILVA; ÁLVARO; AFONSO, 2007, p. 6).

Desenvolvido no ano de 1994 pelo o então engenheiro de software Rasmus Lerdof o PHP tinha como objetivo principal acompanhar a quantidade de acessos que seu currículo online iria obter. Os scripts iniciais eram bem similar ao da linguagem Perl, porém continha limitações.

“O PHP é uma linguagem para a criação de scripts para a *Web* do lado servidor, que pode ser incorporada em HTML, ou utilizada como um binário independente (embora a primeira utilização seja muito mais comum)”. (CONVERSE; PARK, 2003, p. 3).

Geralmente, o PHP pode ser definido como uma linguagem para criação de scripts que ao ser embutida em páginas HTML as torna dinâmicas fornecendo indeterminadas ações. Seu código é aberto e compatível com diversos servidores *web* dentre eles destaca-se o Apache.

[...] o PHP coloca-se como uma das mais poderosas ferramentas para os profissionais de desenvolvimento e programação voltada para a Internet, sendo capaz de reduzir bruscamente a quantidade de tempo necessário para a realização de um projeto ou mesmo permitir que novos mecanismos sejam criados de maneira que se possa obter um produto final diferenciado e digno de destaque. (BRUNO; ESTROZI; NETO, 2010, p. 27).

Nos dias atuais o uso do PHP tem se mostrado uma potencia no mercado de programação para internet, muito em decorrer das suas funcionalidades e redução no tempo de desenvolvimento do software. A utilização desta linguagem abrange vastos campos, dentre eles o mais comum, devido à praticidade, é fazer a conexão com sistemas gerenciadores de banco de dados.

3.2.1. Sintaxe PHP

“O PHP é uma linguagem de programação interpretada pelo servidor que fica embutida junto do código HTML”. (BRUNO; ESTROZI; NETO, 2010, p.29).

Os códigos PHP são responsáveis por interagir com os servidores *web* locais ou remotos, isto é, ele pode estar em qualquer parte do código HTML que mesmo assim o servidor irá interpreta-lo e logo após executa-lo.

Pode-se dizer que o servidor é responsável por percorrer todo arquivo a fim de procurar um conjunto de tags especiais que compõe o PHP. Estas tags são: `<?php` e `?>`, determinantes para dizer se um arquivo é composto ou não por códigos PHP.

Figura 6: Script do PHP junto ao HTML

```
<html>
  <head>
    <title>Exemplo 1</title>
  </head>
  <body>
    <?php
      echo("Olá! Este texto é resultado de um script PHP");
    ?>
  </body>
</html>
```

Fonte: Odemir M. Bruno, Leandro F. Estrozi e João E. S. Batista Neto, 2010, p. 29-30

O exemplo esboçado pela imagem mostra a forma mais correta de utilizar o PHP na estrutura HTML, isto é, mesmo que os códigos possam ser aplicados em qualquer parte do arquivo, é mais comum utilizar os scripts inseridos entre as tags `<body>` da página *web*.

3.3. JAVASCRIPT

No ano de 1995 surgiu o JavaScript uma linguagem desenvolvida por Brendan Eich e integrantes da companhia Netscape, cujo propósito era fornecer aos usuários páginas *web* mais dinâmicas, tornando o seu uso mais agradável (preciso). Inicialmente a linguagem foi criada para rodar nos servidores da Netscap, mas após obter sucesso em seu lançamento passou a ser implementada em outros navegadores, logo, transformando-se em uma ferramenta importante e indispensável para o uso da internet.

"JavaScript nos permite fazer scripts do conteúdo HTML e da apresentação CSS de documentos em navegadores *Web* [...]" (FLANAGAN, 2012, p. 10).

Uma das características do JavaScript está relacionado a diversidade de possibilidades de programação que ela fornece as páginas *web*, atuando na transformação e processamento dos dados enviados e recebidos, interagindo com as marcações e exibições da linguagem HTML e o aprimoramento desse conteúdo proporcionado pelo CSS.

"A ampla maioria dos sites modernos usa JavaScript e todos os navegadores modernos - em computadores de mesa, consoles de jogos, tablets e smartphones - incluem interpretadores JavaScript, tornando-a linguagem de programação mais onipresente da história." (FLANAGAN, 2012 p.1).

Atualmente o JavaScript não só evolui como propôs novos recursos na área de desenvolvimento de softwares, abrangendo e atendendo as demandas de mercado, por exemplo em seu surgimento era possível criar aplicações apenas para *web*, que ao decorrer dos anos foi modelado para ser utilizado na criação de aplicativos mobiles e desktops.

3.3.1. Sintaxe JavaScript

"O JavaScript é a linguagem utilizada na camada de comportamento e é uma linguagem de *scripting*, o que significa que tem que ser utilizada em conjunto com outra linguagem (como é o caso do HTML) ou aplicação".(REMOALDO, 2008, p. 34).

Baseado na definição de Remoaldo, o JavaScript ou de forma mais abreviada o JS é uma linguagem de programação dinâmica, utilizada nos navegadores da *web*, possibilitando controlar o comportamento das páginas *web* e seus utilizadores através de scripts., mas é necessário que o JS seja utilizada em conjunto com outra linguagem (HTML) ou aplicação.

As ações desta linguagem são programadas entre as tags `<script>` e `</script>`, ou seja, tudo que estiver contido entre estes delimitadores será responsável por dar dinamicidade à página *web*.

A inserção dos códigos JS na página HTML pode ocorrer de duas maneiras, a primeira é quando os códigos são usados de forma direta, isto é, os scripts da linguagem são inseridos diretamente nas tags `<head>` do arquivo, possibilitando assim um acesso rápido e eficiente caso haja alguma alteração. Outra forma de

implementação pode ser realizada através de links, que oferece a vantagem de manipular um arquivo que contenha apenas linguagem JS.

"Pode incluir código JavaScript num elemento `<script>`, que está muitas vezes localizado na secção `<head>` de um documento HTML [...]".(REMOALDO, 2008, p.35).

Figura 7: Código HTML e JavaScript

```
<html>
  <head>
    <script><!--Início do Java Script -->
        /*Códigos são implementados aqui para a funcionalidade*/
    </script><!--Fim do Java Script -->
  </head>
</html>
```

Fonte: Autor do Trabalho (2016)

O primeiro exemplo traz uma das formas em que a linguagem JavaScript pode ser utilizada, no caso junto ao HTML.

"Pode embeber código JavaScript directamente no conteúdo HTML [...]". (REMOALDO, 2008, p.35).

Figura 8: Código HTML e JavaScript (link externo)

```
<html>
  <head>
    <script src = "teste.js" type = "text/javascript"> </script>
    /*Link externo do JavaScript*/
  </head>
</html>
```

Fonte: Autor do Trabalho (2016)

A segunda imagem (figura 8) é responsável por demonstrar como o JavaScript pode ser utilizado pelo formulário HTML. Neste caso toda a programação é feita no arquivo teste.js, arquivo este que contém toda programação de scripts necessários para dar dinamicidade a página *web*.

3.4. CSS

“O termo CSS, proveniente de Cascading Style Sheets, que significa Folhas de Estilos em Cascata, é uma linguagem de estilo (ou de formatação) fantástica para a construção do *layout* de suas páginas ou sites.” (JOBSTRAIBIZER, 2009, p. 06).

O CSS é uma linguagem de programação cujo objetivo é projetar a apresentação de um documento *web*. Esta por sua vez, tem proporcionado modificações no desenvolvimento de páginas, em decorrência de oferecer mecanismo robusto que possibilita a estilização de um aplicativo.

A utilização do CSS é possível por dois métodos, à primeira ocorre através da criação de um arquivo (folha) que para o funcionamento será referenciado na página principal, ou seja, aquela página que contém as linguagens de marcação; a segunda forma de ocorrer à estilização é aplicar a formatação diretamente nas tags ou diretamente no documento que compõe o HTML ou derivados.

Figura 9: Link de acesso externo CSS

```
<!DOCTYPE html>
<html>
<head>
<link href="estilo.css" rel="stylesheet"
type="text/css"/>
</head>
<body>
<p>conteúdo do parágrafo 1</p>
</body>
</html>
```

Fonte: Felipe Alex Scheidt, 2015, p. 13.

Conforme mostrado na imagem (figura 9) as regras CSS está contida no arquivo `estilo.css`, que pode ser acessado pelo HTML através do link especificado na página.

Figura 10: CSS aplicado na seção head da página HTML

```
<!DOCTYPE html>

<html>

<head>

<style>

  p{

    font-weight: bold; /* fonte do tipo negrito */

  }

</style>

</head>

<body>

  <p>conteúdo do parágrafo 1</p>

</body>

</html>
```

Fonte: Felipe Alex Scheidt, 2015, p. 12.

A imagem acima mostra outra forma de vincular o código CSS a uma página HTML, uma das formas mais utilizada até então, isto, devido apresentar facilidade de uso e acesso rápido aos códigos de animações (CSS).

4. BANCO DE DADOS

“Um banco de dados é uma coleção de dados relacionados. Com dados, queremos dizer fatos conhecidos que podem ser registrados e possuem significado implícito.” (ELMASRI; NAVATHE, 2011, p.3).

Um sistema gerenciador de banco de dados pode ser conceituado como uma coleção de dados inter-relacionados, capazes de manipular informações através de outros softwares e diferentes usuários. Estes dados são armazenados de forma estrutural, funcionalidade que impossibilita a redundâncias dos dados.

Os bancos de dados e sua tecnologia têm um impacto importante sobre o uso crescente dos computadores. É correto afirmar que os bancos de dados desempenham um papel crítico em quase todas as áreas em que os computadores são usados, incluindo negócios, comércio eletrônico, engenharia, medicina, genética, direito, educação e biblioteconomia. (ELMASRI; NAVATHE, 2011, p.2).

Atualmente os bancos de dados são essências para a era computacional, é possível armazenar, coletar, alterar e excluir informações, sua evolução ajudou a nos depender cada vez mais da tecnologia da informação, hoje diversas atividades são conceituada a um sistema gerenciador de banco de dados. Por exemplo, ir ao banco retirar ou depositar, agendar um hotel, fazer uma comprar online, etc.

4.1. HISTÓRIA DO BANCO DE DADOS

Segundo Elmasri e Navathe (2011, p.15) “A maioria desses sistemas de banco de dados era implementada em computadores mainframes grandes e caros, começando em meados da década de 1960 e continuando nos anos de 1970 e 1980”.

Por volta das décadas de 1960 e 1970 começaram a surgir fundamentos de banco de dados nas empresas, muitos observadores descobriram que além de custoso era inviável para a companhia empregar um número grande de pessoas que tinham como trabalho armazenar e organizar arquivos. A fim de sanar o problema existente, pesquisadores iniciaram a busca por encontrar solução eficaz e eficiente objetivando na mecanização do processo.

O anseio da busca a investigação tornou viral, até que no ano de 1970 um pesquisador da IBM - Ted Codd - publicou o primeiro artigo sobre banco de dados

relacional, baseado em no uso de cálculo e álgebra relacional permitindo que usuários armazenassem e recuperassem grande quantidade de informações. Sendo uma pessoa visionária o mesmo estipulou que era necessário um sistema intermediário onde o usuário seria capaz de acessar informações através de comandos em inglês, estas informações estariam contidas em tabelas.

A IBM então decidiu investir mais no projeto o que levou a montar uma equipe que seria responsável por projetar um sistema de banco de dados relacional que eventualmente se tornaria um produto. Então em meados de 1976 a pequena startup Oracle lança o primeiro sistema comercial de banco de dados.

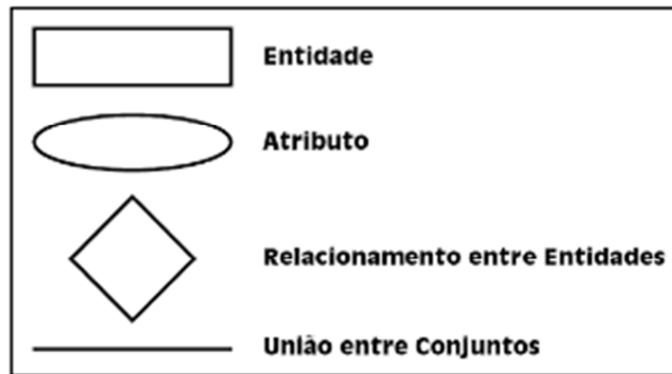
4.2. MODELAGEM DE DADOS

A modelagem de dados pode ser considerada um diagrama responsável por demonstrar as interações das informações e das tabelas, esses processos de modelagem são realizados anteriormente a criação do banco de dados, o que possibilita averiguar a integridade das informações para que não haja falhas na aplicação.

A modelagem de dados é uma área da TI (Tecnologia da Informação) que estuda e apresenta ferramentas para descrever dados de forma conceitual, ou seja, utilizando símbolos e gráficos. A simbologia utilizada favorece uma representação mais precisa e rápida dos dados, permitindo que qualquer banco de dados seja projetado antes de sua real criação. (LOBO, 2008, p.9).

Baseado nas informações do autor é correto afirmar que a modelagem de dados é uma ferramenta conceitual, cuja finalidade implica diretamente na junção dos dados e a forma em que se relacionam, ou seja, pode-se definir como a base para a criação do banco de dados.

“O modelo de Entidades e Relacionamentos possui símbolos que representam as entidades, relacionamentos e o conjunto de atributos que vão designar o banco de dados.” (LOBO, 2008, p.10).

Figura 11: Símbolo do MER

Fonte: Edson Junio Rodrigues Lobo, 2008, p. 10.

A representação das estruturas (figura 10) proporciona a compreensão, visualização e comportamento das informações através de mecanismos que irá compor o banco de dados. Abaixo é possível identificar a funcionalidade de cada um destes elementos:

Descrição dos símbolos:

- **Entidade:** É um conjunto de informações conhecida ou manipulada pelo banco de dados.
- **Atributos:** Informações que representam uma entidade.
- **Relacionamento entre entidades:** É a união entre as entidades, cuja função é representar o relacionamento dos atributos envolvidos.
- **União entre conjuntos:** Como o próprio nome sugere, tem por característica demonstrar a união de uma ou mais entidades.

Os relacionamentos entre as entidades quando ocorrem tem por característica possuir uma cardinalidade, é ela que representa a quantidade de objetos envolvidos no relacionamento. A seguir iremos analisar os tipos de relacionamentos possíveis:

- **1 para 1:** Ocorre quando um registro de determinada entidade, se relaciona com apenas um único registro de outra entidade. Exemplo: Uma pessoa possui apenas um CPF, e aquele CPF só pertence a uma pessoa.
- **1 para muitos:** Um elemento de uma entidade pode se relacionar com vários elementos de outras entidades. Exemplo: Em um plano de sócio clube, uma

pessoa pode ter vários dependentes, porém cada dependente só pode estar ligada a um usuário principal.

- **Muitos para muitos:** Neste tipo de relacionamento, um ou mais elementos da tabela se relacionam com um ou mais registro de outra tabela. Exemplo: Em um sistema escolar, um professor pode ensinar em um ou mais curso, e o cada curso pode ter mais de um professor.

Contudo, em virtude dos fatos mencionados anteriormente, a principal função do MER (Modelo Entidade e Relacionamento) é criar estruturas dos dados permitindo que as regras de negócios sejam produzidas antes da criação do banco de dados.

4.3. TABELAS

Segundo Lobo (2008, p. 57) "As tabelas de um banco de dados são estruturas de armazenamento. Elas são criadas para armazenar um conjunto de valores que serão utilizados posteriormente para gerar informações em um sistema".

As tabelas são à base de todos os bancos de dados relacional cuja responsabilidade é organizar os dados armazenados de maneira lógica, isto é, os dados contidos na tabela serão organizados e separados por linhas e colunas, onde cada informação armazenada estará de acordo com o seu respectivo tipo, podendo ser numérico, alfanumérico, datas, entre outros.

Figura 12: Exemplo tabela

Usuario	
	CodUsuario: int
	usuario: Texto(20)
	senha: Texto(20)

Fonte: Elaborado pelo autor do trabalho.

De forma exemplificada, a imagem ilustrada acima tem por obrigatoriedade demonstrar como as informações são armazenadas dentro de uma tabela no SGBD, além de atribuir os dados corretamente aos seus determinados tipos.

5. INTERFACE HOMEM COMPUTADOR

Para Barbosa e Silva (2010, p.10) o HCI "é uma disciplina interessada no projeto, implementação e avaliação de sistemas computacionais interativos para o uso humano [...]".

O HCI (*Interface Human-Computer*) é uma área multidisciplinar cujo objetivo é estudar e definir métodos para o projeto, a fim de satisfazer as necessidades do usuário, ou seja, as aplicações devem buscar meios de atingir a usabilidade, aplicabilidade e comunicabilidade, disponibilizando mecanismo simples para facilitar o uso do cliente final.

Para Barbosa e Silva (2010, p.12) "Alguns conhecimentos e técnicas importados de outras áreas além da Computação são adaptados às necessidades de IHC".

Figura 13: Área de atuação do IHC



Fonte: <http://wdxpt.blogspot.com.br/2011/08/o-que-e-ihc-parte-i.html>

A interação humana computador tem a característica de pesquisar e agregar o conhecimento de outras áreas como: psicologia, sociologia, antropologia, *design* e engenharia, e devido à interação destas diversas etapas é possível ter uma melhor compreensão sobre os mecanismos envolvidos no uso de sistemas computacionais interativos.

Portanto, nos dias atuais o estudo do IHC tornou-se indispensável para o desenvolvimento de qualquer aplicação computacional, que além de exibir a relação de como o usuário se comunica com a máquina mostra também como a tecnologia corresponde à interação do usuário.

5.1. INTERFACE

"A interface com usuário determina os processos de interação possíveis, à medida que determina o que ele pode falar ou fazer, de que maneira e em que ordem". (BARBOSA; SILVA. 2010 p. 26).

A interface é o único meio de comunicação entre o usuário e o sistema, ou seja, é através dela que o usuário consegue disparar e receber os resultados destas ações. Contudo, é nítida a importância da interface para o desenvolvimento de um software, visto que, é ela a responsável por analisar individualmente cada *layout*, verificando se os mesmos se atendem os requisitos para o funcionamento. As análises tem o proposito de identificar e atuar nas anormalidades encontradas, para entender melhor este processo será tipificado abaixo cada uma das etapas realizadas.

Para a Revistabw (2013, p.1) visibilidade do status do sistema é "[...] entender as informações que ele necessita do sistema comunicando estas mensagens em uma linguagem adequada e construir um sistema que forneça elementos que digam ao usuário o que o sistema está continuamente fazendo".

- **Visibilidade do status do sistema:** Está relacionado à forma em que o sistema irá se comportar quando um usuário realizar uma determinada ação.

Segundo a Revistabw (2013, p.1) a interação entre a interface e o mundo real afirma que "o sistema deve falar a linguagem do usuário e não a linguagem técnica".

- **Relacionamento entre a interface do sistema e o mundo real:** A finalidade é contextualizar a comunicação entre o sistema e o usuário.

Conforme a Revistabw (2013, p.1) a liberdade e controle do usuário permitem que "o usuário, sempre que desejar, deve poder abortar uma tarefa ou retornar ao ponto anterior. O sistema não pode impedir uma operação do usuário [...]".

- **Liberdade e controle do usuário:** É explicar ao usuário o motivo em que determinada ação solicitada pelo mesmo não pode ser cancelada.

“o sistema sempre deve utilizar o mesmo padrão de ícones, símbolos e de palavras”. (REVISTABW, 2013, p.1).

- **Consistência e Padronização:** O sistema deverá realizar a mesma ação independente do *layout* que o usuário estiver conectado, deverá conter as mesmas funcionalidades e os botões básicos e as cores devem seguir o mesmo padrão da aplicação corrente.

De acordo com a Revistabw (2013, p.1) para prevenir erros de um determinado sistema é necessário “[...] criar mecanismos que possam prevenir os erros mais básicos do usuário”.

- **Prevenção de erros:** é criar mensagens informativas sobre o que pode ocasionar se tal arquivo for alterado e estabelecer formatos obrigatórios de campos, permitindo preenchimento automático do campo, minimizando erros de digitação.

Para Revistabw (2013, p.1) define reconhecimento e não lembrança como sendo “[...] elementos de diálogo que permita que o usuário manipular o sistema, mas sem sobrecarregar sua capacidade de memorização”.

- **Reconhecimento e não lembrança:** o sistema deve conter informações capazes de orientar os usuários.

Segundo a Revistabw (2013, p.1) a flexibilidade e eficiência de uso defendem que “o sistema deve ser fácil para uso por usuários comuns, mas deve ser flexível para permitir que usuários avançados possam ter ganho de desempenho”.

- **Flexibilidade e eficiência de uso:** é o modo em que a aplicação será projetada para atender usuários leigos e a flexibilidade que terá para auxiliar usuários avançados.

Conforme a Revistabw (2013, p.1) “o texto e o *design* do sistema devem ser sempre simples e objetivos”.

- **Estética e *design* minimalista:** as informações devem ser claras e objetivas ao usuário, para que o mesmo consiga realizar ações facilmente.

“A mensagem de erro nunca pode intimidar ao usuário”. (REVISTABW, 2013, p.1).

- **Ajudar os usuários a reconhecer, diagnosticar e sanar erros:** as mensagens devem ser claras e explicativas no intuito de auxiliar o usuário e não intimidá-lo.

“um sistema eficiente deve ser tão fácil de utilizar que o usuário não precise de maior ajuda”. (REVISTABW, 2013, p.1).

- **Ajuda e documentação:** um bom *design* consegue evitar a necessidade de ajuda no uso do sistema, mas ainda assim, é necessário possuir uma documentação cuja finalidade é auxiliar o usuário em caso de dúvida.

Em virtude dos fatos mencionados, conclui-se que a interface se faz necessária em toda aplicação seja ela *web*, desktop, ou para dispositivos móveis, pois é ela que possibilita que o usuário possa ter compreensão de como o sistema irá se comportar ao desempenhar determinadas atividades.

6. SISTEMA DE INFORMAÇÃO GERENCIAL PARA ACADEMIA DE MUSCULAÇÃO (PROJETO - SIGAM)

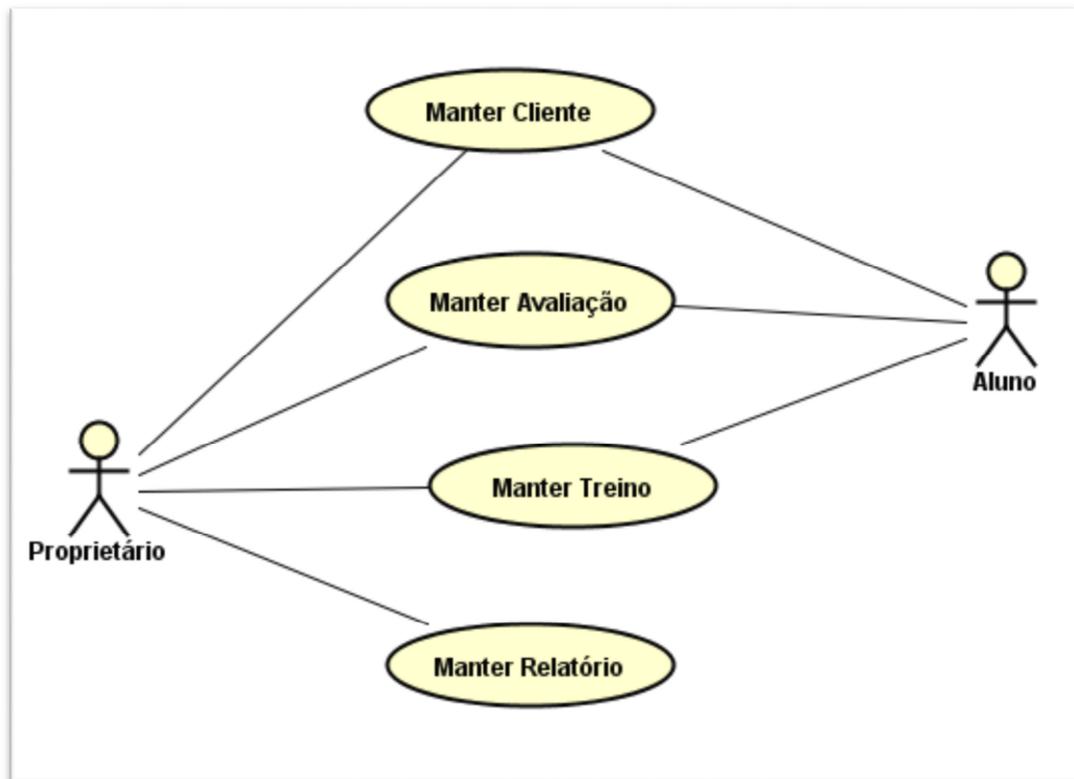
Consiste em um sistema de informação gerencial voltado para academia de musculação, o SIGAM uma ferramenta *web*, desenvolvido no ambiente Windows versão 10, que possibilitará um melhor gerenciamento da academia de musculação, garantindo ao proprietário acesso das informações através de dispositivos móveis como: smartphones, tablets, notebooks, além de ser acessado localmente por uma máquina desktop que estiver disponível no local.

- **Modulo login:** Validar autenticação do usuário, a fim de evitar que qualquer individuo possa ter acesso.
- **Modulo cadastro de clientes:** Responsável por armazenar informações básicas dos alunos, além de auxiliar o proprietário a realizar diversas atividades para que o indivíduo alcance seu objetivo.
- **Modulo ficha treino:** Auxiliará o aluno a desempenhar atividades adequadas de acordo com a avaliação física realizada pelo professor.
- **Modulo Avaliação:** Irá submeter o aluno a realizar diversas atividades físicas que posteriormente será analisada pelo instrutor.
- **Modulo relatório:** Fornecer informações sobre os alunos e atividades extras realizadas na academia.

6.1. DIAGRAMA DE CASO DE USO

O diagrama de casos de uso é o diagrama mais geral e informal da UML, utilizado normalmente nas fases de levantamento e análise de requisitos do sistema, embora venha ser consultado durante todo o processo de modelagem e possa servir de base para outros diagramas. (GUEDES, 2011, p. 30).

Com base no que foi descrito por Guedes é correto afirmar que o diagrama de casos de uso é um diagrama simples e comum, mas de extrema importância para UML, pois além de não possuir detalhamentos mais aprofundados, serve como base para outros diagramas. Geralmente seu objetivo é esquematizar informações coletas no levantamento de requisitos.

Figura 14: Diagrama de Caso de Uso

Fonte: Elaborado pelo autor do trabalho.

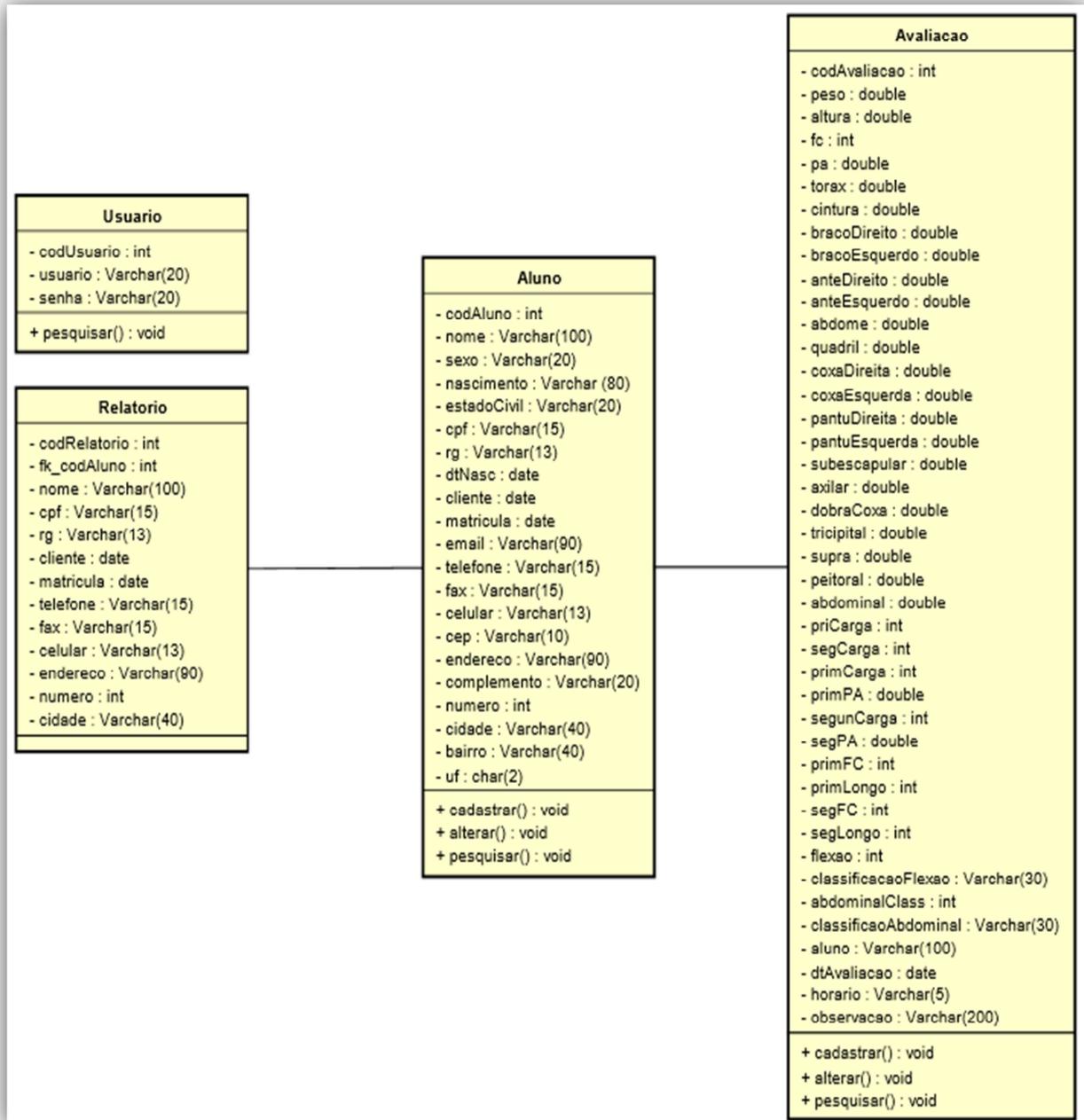
O proprietário fará o gerenciamento dos dados dos alunos, além de realizar a avaliação física, montar treino e posteriormente acessá-los quando necessário para meios de comparação.

6.2. DIAGRAMA DE CLASSE

Seu principal enfoque está em permitir a visualização das classes que compõem o sistema com seus respectivos atributos e métodos, bem como em demonstrar como as classes do diagrama se relacionam, complementam e transmitem informações entre si. (GUEDES, 2011, p.101).

Um dos diagramas mais importantes da UML é o diagrama de classe, pois se utiliza informações coletadas no levantamento de requisitos de forma que é possível visualizar as interações dos dados, além de servir como base para a produção de outros diagramas.

Figura 15: Diagrama de Classe



Fonte: Elaborado pelo autor do trabalho.

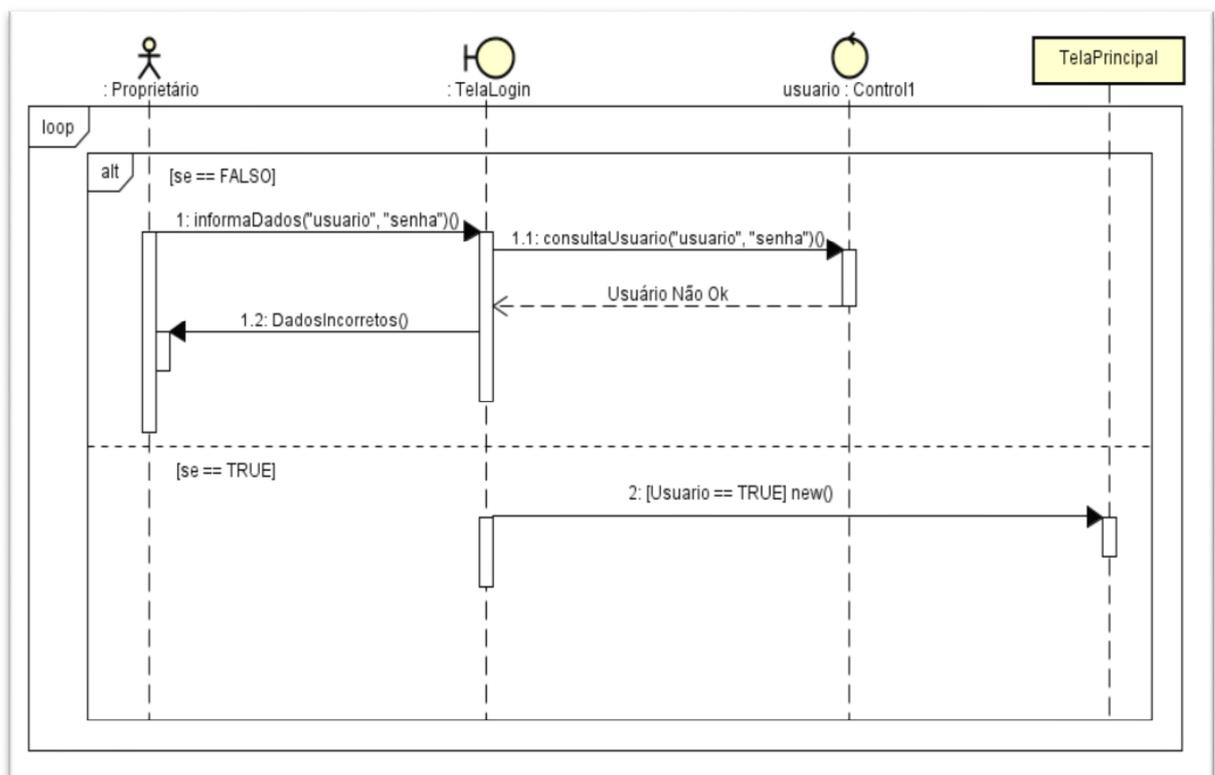
A imagem é responsável por demonstrar a interação entre as classes do sistema.

6.3. DIAGRAMAS DE SEQUÊNCIAS

“O diagrama de sequência é um diagrama comportamental que se preocupa com a ordem temporal em que as mensagens são trocadas entre os objetos envolvidos em um determinado processo.” (GUEDES, 2011, p.33).

Conceitualmente embasado na opinião do autor, as informações se interagem de forma sequencial de tal modo que passam a realizar determinadas ações.

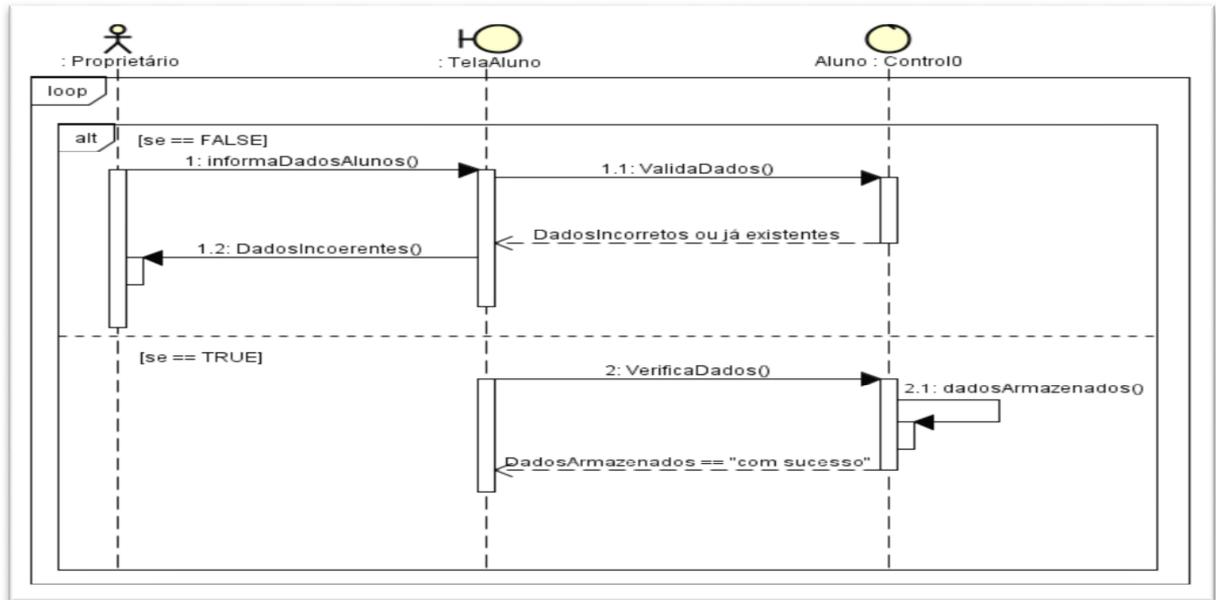
Figura 16: Diagrama de Sequência Tela Login



Fonte: Elaborado pelo autor do trabalho.

A imagem anexada acima demonstra a ação que o usuário do sistema fará para logar ao sistema e ter acesso a outras funcionalidades do sistema.

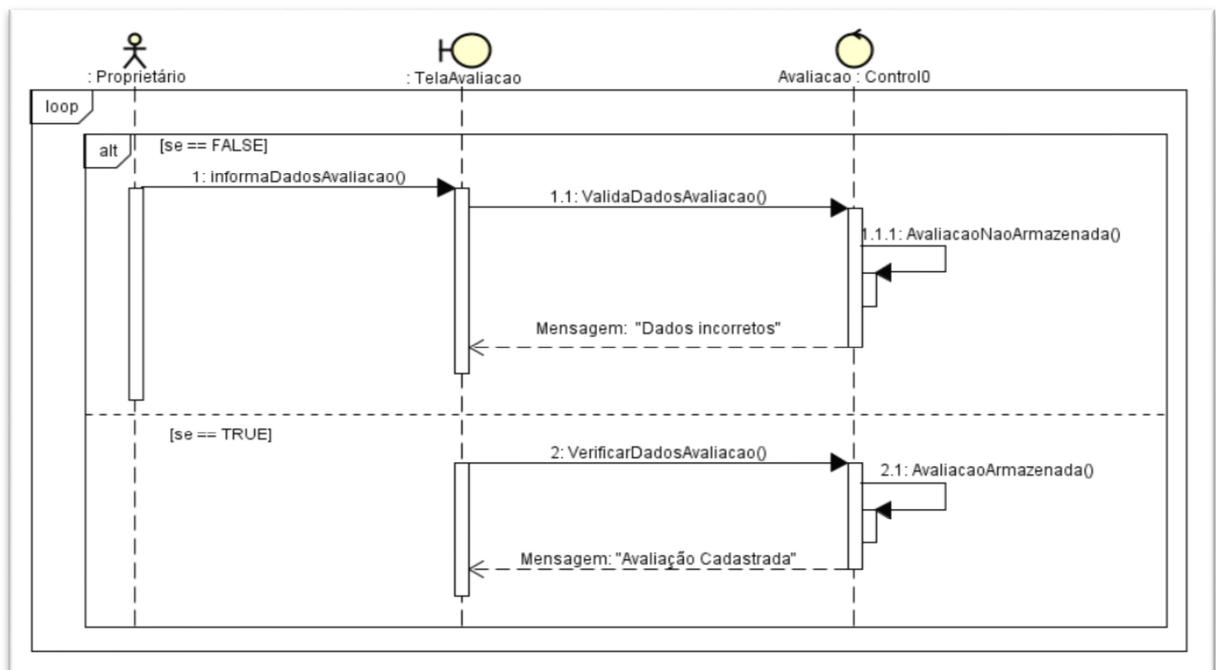
Figura 17: Diagrama de seqüência tela cadastro aluno



Fonte: Elaborado pelo autor do trabalho.

A figura ilustrada acima demonstra a seqüência a ser seguida pelo administrador do sistema quando o cadastro de um novo aluno é realizado.

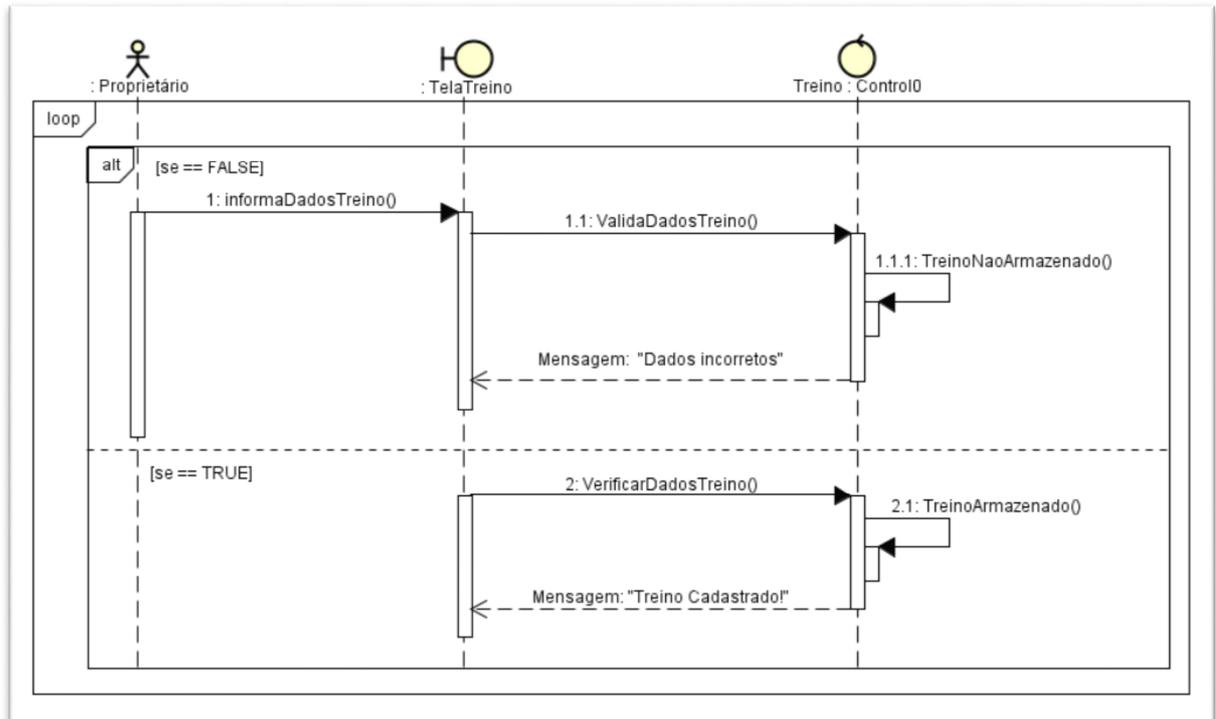
Figura 18: Diagrama de seqüência tela avaliação



Fonte: Elaborado pelo autor do trabalho.

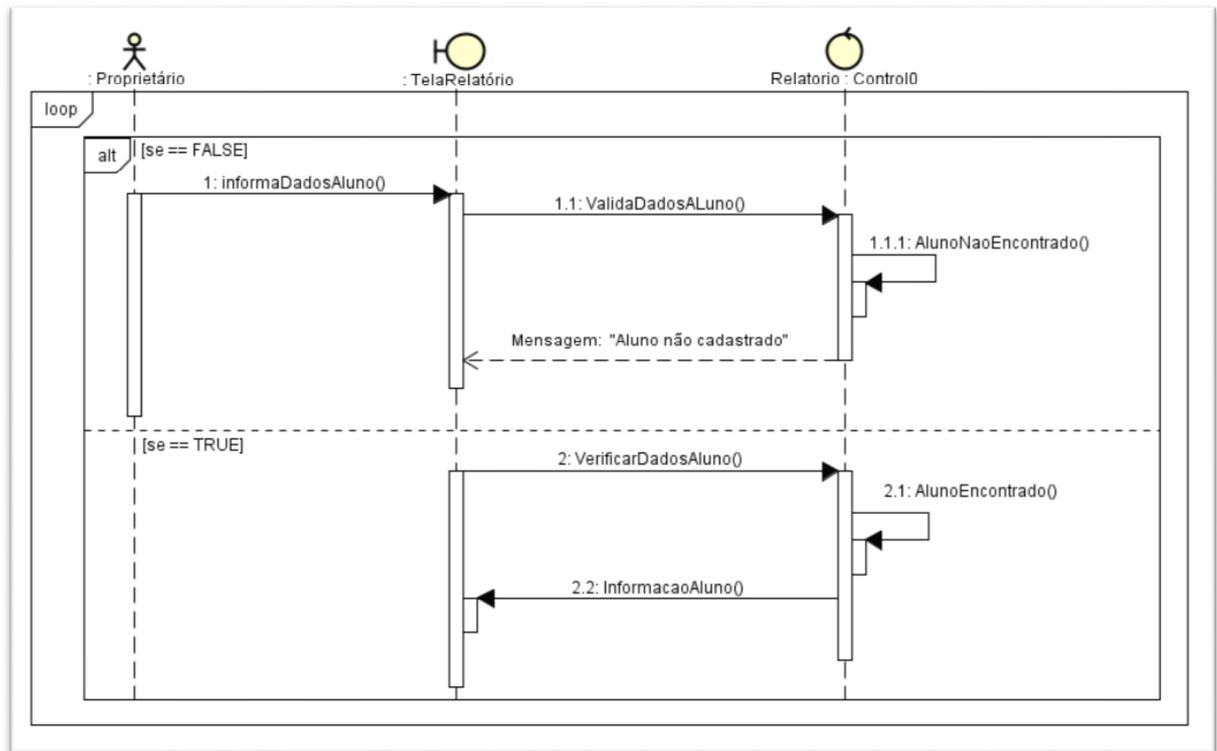
A imagem tem por objetivo demonstrar a ação realizada pelo usuário do sistema ao desempenhar determinadas atividades, como a de cadastrar a avaliação do aluno.

Figura 19: Diagrama de sequência tela treino



Fonte: Elaborado pelo autor do trabalho.

A imagem mostra a sequência tomada pelo usuário ao realizar a criação de uma nova ficha de treino para que o aluno possa ter acesso aos exercícios que irá desempenhar para atingir seu objetivo.

Figura 20: Diagrama de sequência tela relatório

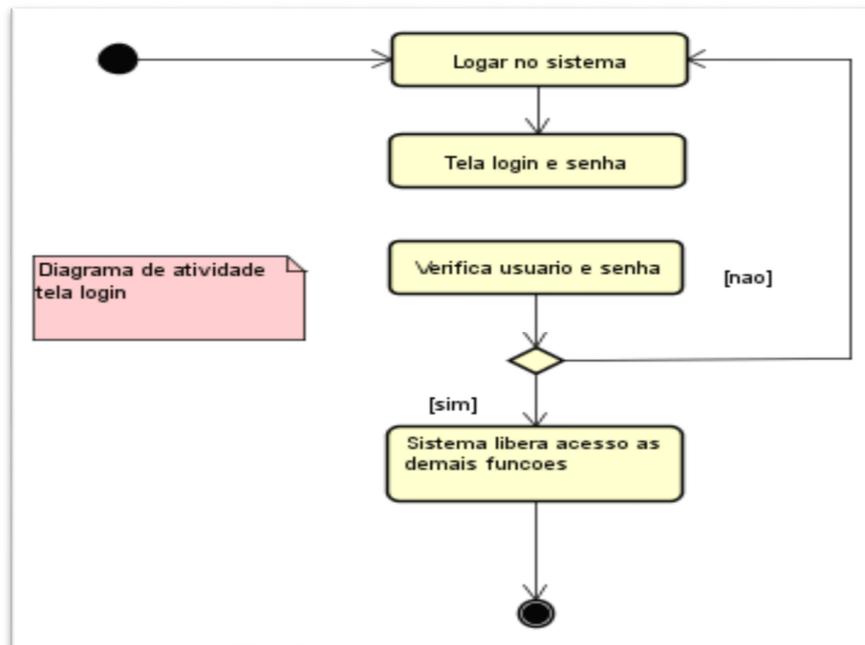
Fonte: Elaborado pelo autor do trabalho.

A figura ilustrada acima tem por responsabilidade demonstrar as ações tomadas pelo administrador da academia quando houver a necessidade de gerar relatórios que contemplam a mensalidade e o período em que o aluno frequenta a academia.

6.4. DIAGRAMAS DE ATIVIDADES

“A modelagem de atividade enfatiza a sequência e condições para coordenar comportamentos de baixo nível.” (GUEDES, 2011, p. 277).

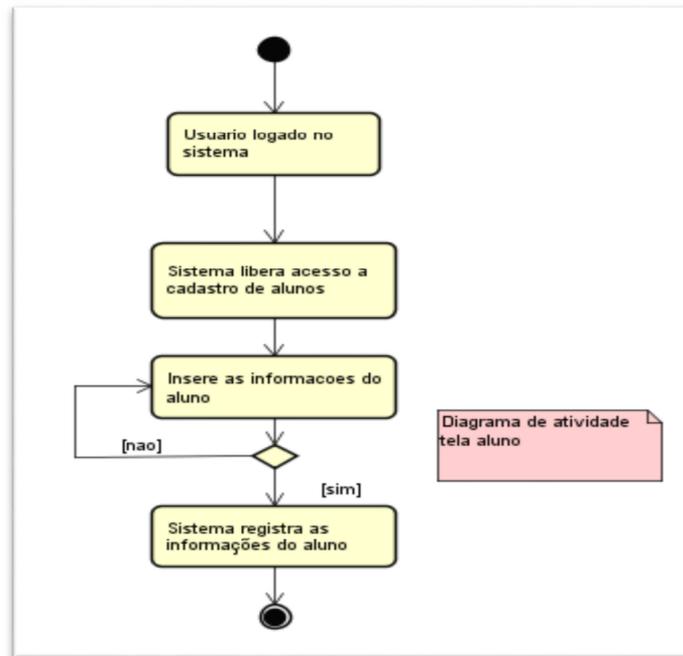
Apresenta-se como diagrama responsável por demonstra o fluxo de tarefas que serão executadas por um ator ou não.

Figura 21: Diagramas de Atividade tela login

Fonte: Elaborado pelo autor do trabalho.

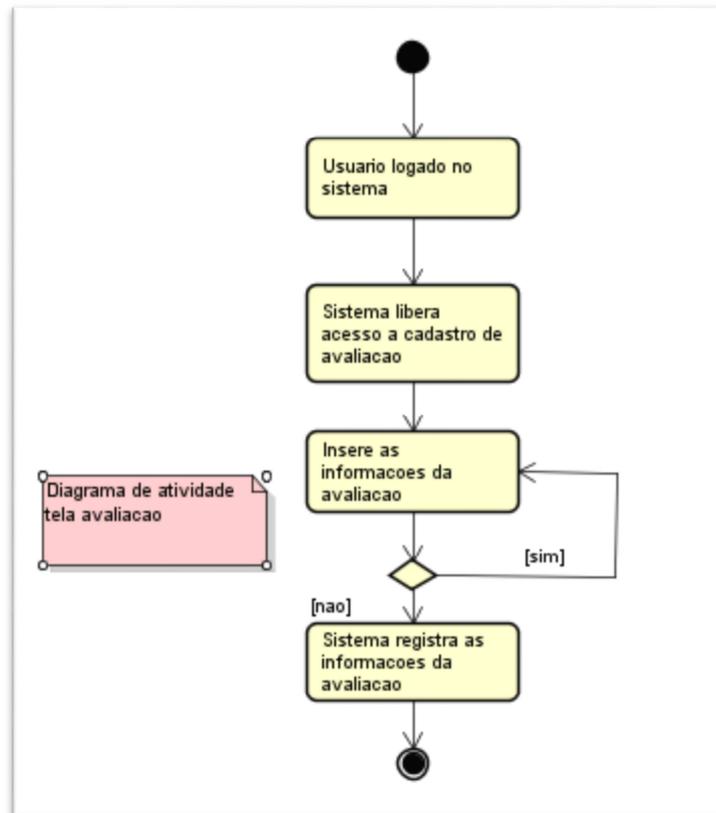
A imagem demonstra as atividades que o usuário precisa realizar para que o mesmo seja autenticado pelo sistema e tenha a permissão para realizar inserções, alterações e pesquisas no sistema.

Figura 22: Diagrama de atividade tela cadastro aluno



Fonte: Elaborado pelo autor do trabalho.

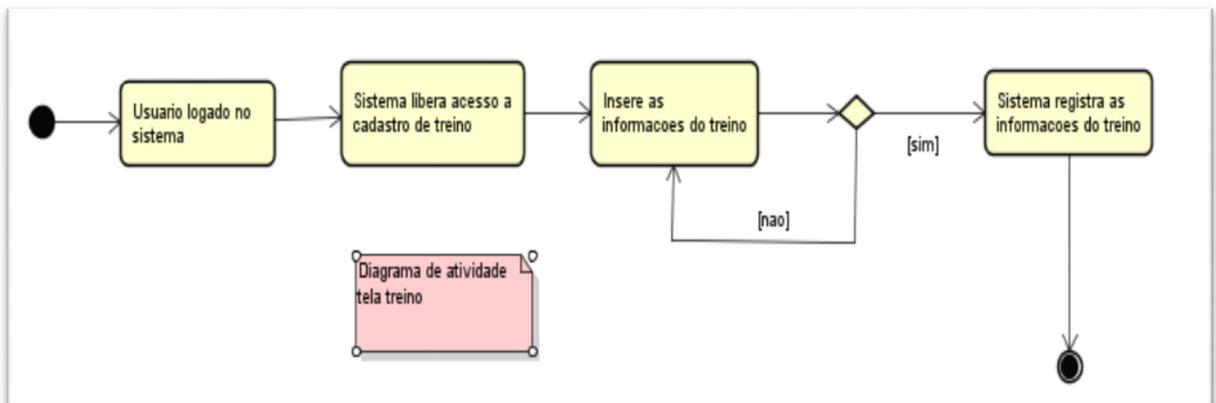
Após a validação do usuário realizado pelo sistema o mesmo tem acesso as demais funções do sistema.

Figura 23: Diagrama de atividade tela avaliação

Fonte: Elaborado pelo autor do trabalho.

Conforme mostra a imagem o administrador consegue realizar de forma computadorizada a avaliação física do aluno ao modo que após a finalização ela fique acessível no sistema para que em uma futura avaliação consiga avaliar os objetivos alcançados pelo aluno.

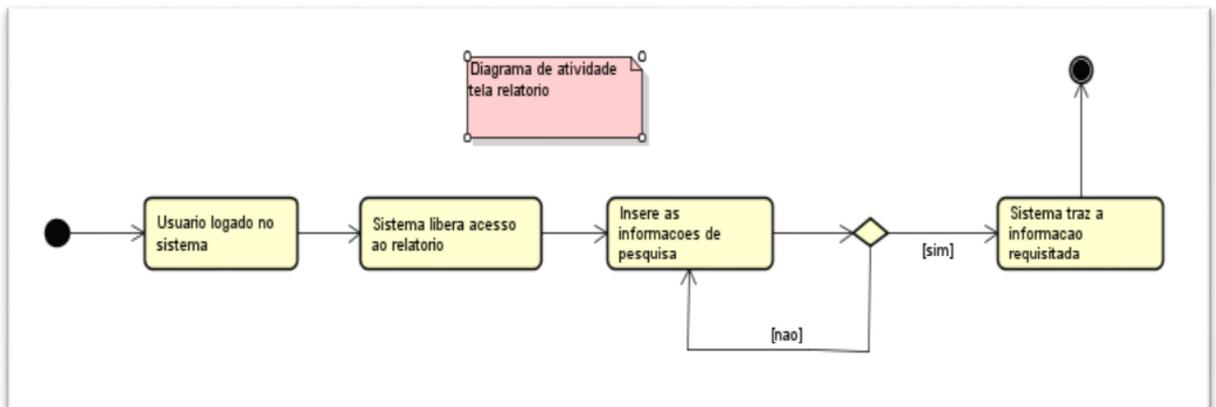
Figura 24: Diagrama de atividade tela treino



Fonte: Elaborado pelo autor do trabalho.

Mediante as atividades desempenhadas, é possível analisar de acordo com a imagem a relação e o passo a passo de todas as atividades desempenhadas para se montar um treino físico.

Figura 25: Diagrama de atividade tela relatório



Fonte: Elaborado pelo autor do trabalho.

É claro de acordo com a imagem a ação tomada pelo administrador quando o mesmo deseja ter acesso a todas as informações de um aluno específico.

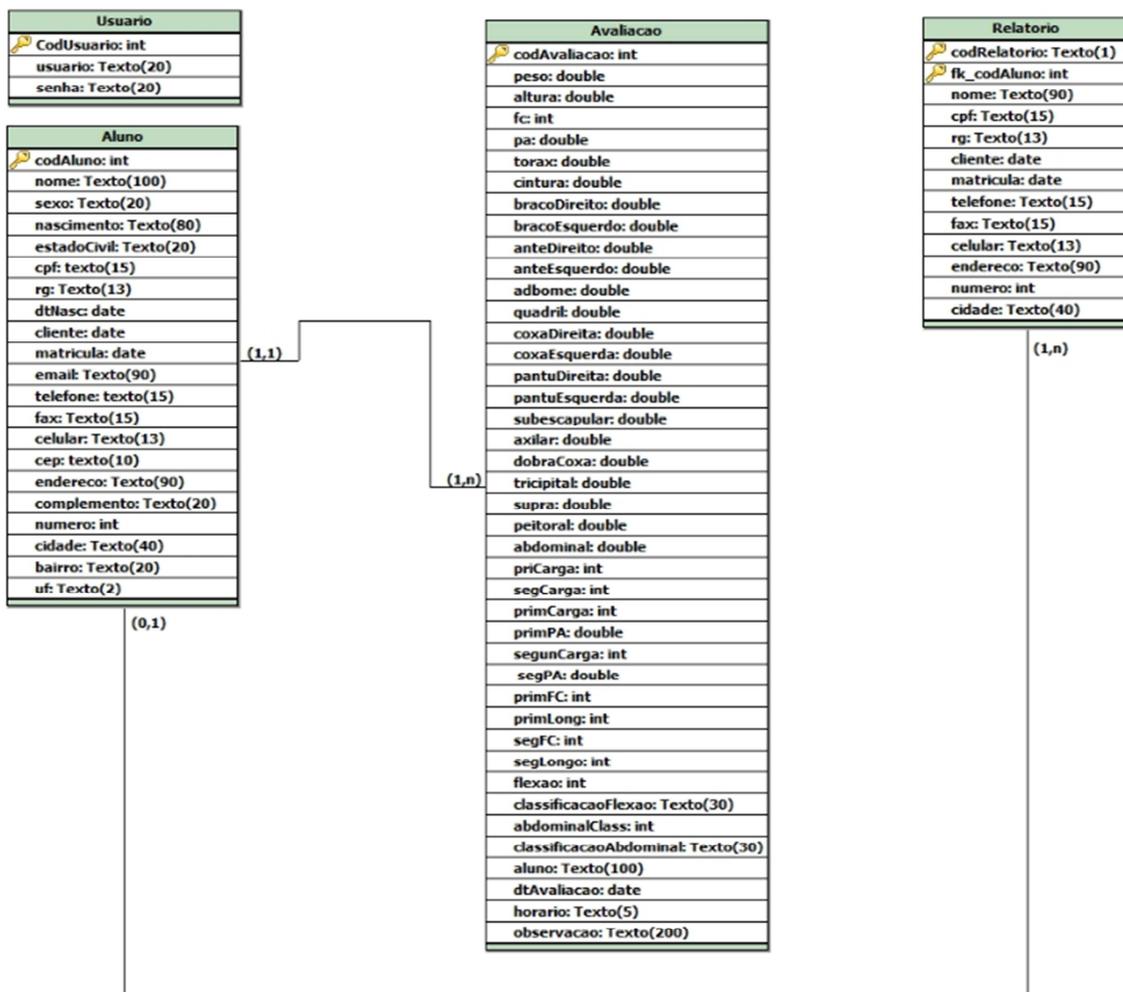
6.5. BANCO DE DADOS MYSQL

O MySQL teve sua origem na década de 90, desenvolvido por David Axmark Allan Larsson e Michael Widenius a fim de criar interfaces compatíveis com o sistema de rotinas o ISAM que utilizavam em suas aplicações e tabelas.

Para Milani (2006, p. 22) “O MySQL é um servidor e gerenciador de banco de dados (SGBD) relacional [...]”.

Desta maneira, pode-se definir o MySQL como sendo um sistema gerenciador de banco de dados relacional que utiliza a linguagem SQL (*Structure Query Language*), usado principalmente nas aplicações *web* gratuitas devido a facilidade de manipulação de dados.

Figura 26: MER do sistema SIGAM

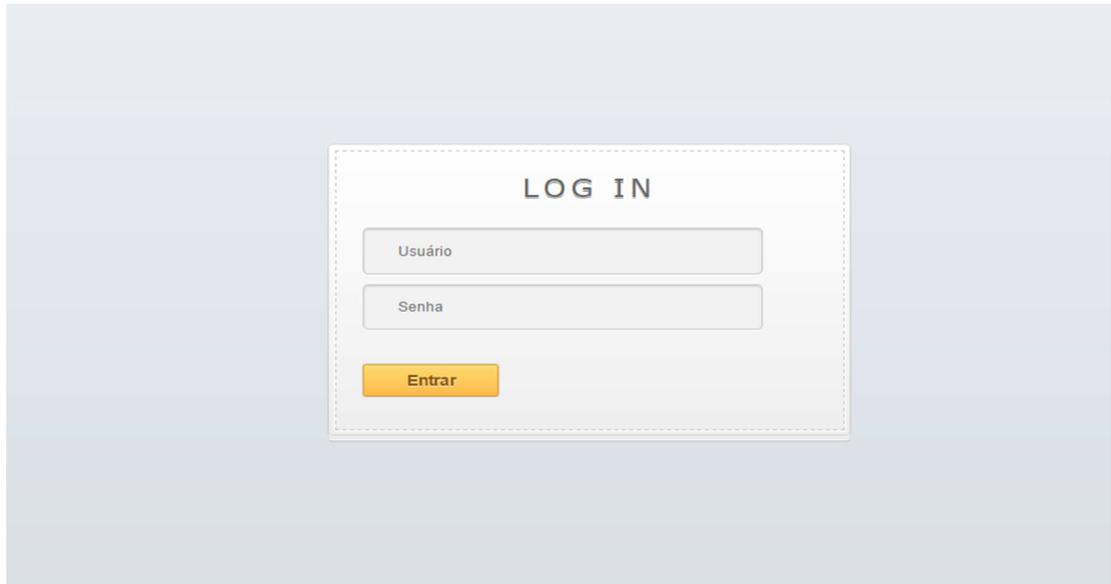


Fonte: Elaborado pelo autor do trabalho.

Conforme mostra a imagem acima, estas são as tabelas tem o objetivo de mostrar a interação entre as mesmas além de ser a essência da aplicação (SIGAM).

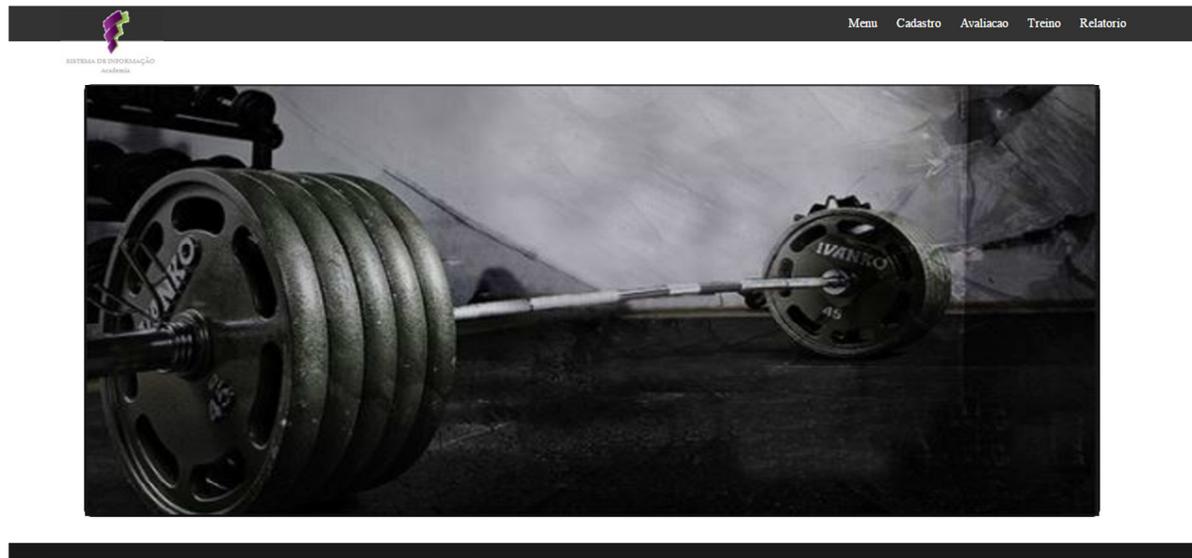
6.6. INTERFACE SIGAM

Figura 27: Tela login SIGAM

A imagem mostra a tela de login do sistema SIGAM. O formulário é centralizado e contém o título "LOG IN" no topo. Abaixo do título, há dois campos de entrada: "Usuário" e "Senha". Abaixo dos campos, há um botão amarelo com o texto "Entrar".

Fonte: Elaborado pelo autor do trabalho.

A figura 27 é a tela de login do sistema de informação gerencial para academia de musculação (SIGAM) no qual irá receber os dados do usuário que posteriormente serão validados e caso esteja correto permite que o funcionário possa utilizar os recursos disponibilizados pelo sistema.

Figura 28: Tela menu do sistema (SIGAM)

Fonte: Elaborado pelo autor do trabalho.

Quando o usuário estiver logado no sistema, o mesmo terá acesso ao menu de opções que o aplicativo disponibiliza para que ele realize determinadas funções conforme mostra a imagem anexada.

Figura 29: Tela cadastro Cliente

Nome Sexo

Natural De Estado Civil

CPF Identidade

Data Nascimento Cliente desde Data Matricula E-mail

Telefone Recado Celular

CEP Endereço Complemento Número

Cidade Bairro UF

Fonte: Elaborado pelo autor do trabalho.

Uma das funcionalidades disponibilizada pelo sistema (figura 29) ao usuário é a parte de cadastro dos alunos que frequenta a academia.

Figura 30: Tela de avaliação

Ficha de Avaliação Física

Peso Altura FC Repouso PA Repouso

PERÍMETROS (PROTOCOLO: POLLOCK)

Torax Cintura Antebraço D E Braço D E

Abdome Quadril Coxa Panturrilha

DOBRAS CUTÂNEAS (PROTOCOLO: POLLOCK, 2DC)

Subescapular Axilar-média Coxa Tricipital Supra-iliaca Peitoral Abdominal

AVALIAÇÃO CARDIORESPIRATÓRIA (PROTOCOLO: ASTRAND - SUB-MÁXIMO)

Carga 1: kp Carga 2: kp

2 min: FC PA 2 min: FC PA

4 min: FC PA 4 min: FC PA

RESISTÊNCIA

Flexões Classificação: Abdominal Classificação:

Aluno Dt. Avaliação Horário

Observação

Fonte: Elaborado pelo autor do trabalho.

Conforme mostra a imagem acima, o usuário tem a opção de realizar a avaliação física do aluno via sistema.

Figura 31: Ficha treino aluno

Ficha treino aluno

Aluno Objetivo

PERNA	COSTA	PEITO
EXTENSORA	PUXADOR COSTA	SUPINO RETO
ADUTORA	PUXADOR FRENTE	SUPINO INCLINADO
LEG PRESS	REMO BAIXO	CRUCIFIXO
AGACHAMENTO	REMADA CURVADA	FLY
QUADRICEPS	REMADA UNILATERAL	
AVANÇO/AFUNDO	PULLDOWN	
HACK		
FLEXORA	BICEPS	TRICEPS
STIFF	ROSCA DIRETA	PULLEY BARRA
LEVANTAMENTO TERRA	ROSCA ALTERNADA	PULLEY CORDA
PANTURRILHA HACK	ROSCA CONCENTRADA	TESTA
PANTURRILHA SENTADO	MAQUINA SCOTT	FRANCES
GLUTEO POLIA BAIXA		UNILATERAL DEITADO
GLUTEO POLIA ALTA	OMBRO	TREINO
GLUTEO CANELEIRA EXTENDIDO	DESENVOLVIMENTO COSTA	A
GLUTEO CANELEIRA FLEXIONADO	DESENVOLVIMENTO FRENTE	B
GLUTEO ABDUTORA	ELEVAÇÃO FRONTAL	C
GLUTEO MAQUINA	ELEVAÇÃO LATERAL	
	REMADA ALTA	

Fonte: Elaborado pelo autor do trabalho.

A figura 31 tem o objetivo de mostrar as atividades de forma separada cada um com um tipo, fornecendo assim ao professor da academia a realizar uma ficha de treino físico ao aluno.

Figura 32: Tela relatório

SISTEMA DE INFORMAÇÃO academia

Menu Cadastro Avaliação Treino Relatório

Nome

Fonte: Elaborado pelo autor do trabalho.

A tela de relatório (figura 32) tem por objetivo demonstrar as informações cadastrais do aluno.

7. CONSIDERAÇÕES FINAIS

O presente trabalho trouxe como proposta o desenvolvimento de um software *web* para gestão de academias podendo ser acessado de forma local ou até por dispositivos móveis, buscando este entendimento foi realizado pesquisas teóricas de forma abrangente sobre sistemas de informação, engenharia de software, linguagem de programação *web* e sistemas gerenciador de banco de dados (SGBD) MySQL.

No desenvolvimento prático deste trabalho foram utilizadas ferramentas *free*, como notepad++ em sua versão 7.4.1 para desenvolver a parte de front-end e back-end do sistema, astah para desenvolver diagramas UML, brModelo em sua versão 3.0 para desenvolver o MER e o SGBD HeidiSql versão 9.4 uma ferramenta com foco em MySQL, para manipular dados do software.

Vale ressaltar que o desenvolvimento do Sistema de Informação Gerencial para Academia de Musculação – “Objeto de Estudo”, deste trabalho, contempla todos os requisitos propostos e explicados no decorrer desta documentação. Ressaltando que o sistema não está em sua totalidade, uma vez que solicitante do sistema não autorizou que os demais módulos fossem divulgados, tendo assim total exclusividade do produto final.

No que se diz respeito à parte teórica foram utilizadas múltiplas referências aos temas abordados, a fim de permitir que o leitor possa ter uma visão do projeto como um todo, além de conhecer as primícias sobre desenvolvimento e aplicações ou softwares *web* que atendam a necessidade do cliente.

Conclui-se que em virtude dos fatos mencionados, desde a introdução até as considerações finais do trabalho, é possível afirmar que o software tem a necessidade de atender academias de musculação, além de fornecer aos proprietários e professores a melhor solução em controle das informações, permitindo ao mesmo, cliente, proporcionar um tempo maior para inserção de novas atividades que se julgue interessante e lucrativa para seus negócios.

REFERÊNCIAS

BARBOSA, Simone Diniz Junqueira; SILVA, Bruno Santana da. **Interação Humano-Computador**. Rio de Janeiro: Elsevier, 2010. 408 p.

BIO, Sérgio Rodrigues. **Sistema de informação Um enfoque Gerencial**. São Paulo: Atlas, 1991. 180p.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML: guia do usuário**. Rio de Janeiro: Elsevier, 2006. 474 p.

BRUNO, Odemir M.; ESTROZI, Leandro F.; NETO, João E. S. Batista. **Programando para a internet com PHP**. Rio de Janeiro: Brasport, 2010. 332 p.

CARRIL, Marly. **HTML - Passo a Passo**. E-book, 2012. 80 p.

CONVERSE, Tim; PARK, Joyce. **PHP a bíblia**. 2. ed. Rio de Janeiro: Elsevier, 2003. 868 p.

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistema de Banco de Dados**. 6. ed. São Paulo: Pearson, 2011. 790 p.

FLANAGAN, David. **JavaScript: O Guia Definitivo**. 6. ed. Porto Alegre: Bookman, 2012 . 1080 p.

GUEDES, Gilleanes Thorwald Araujo. **Uml 2: Uma Abordagem Prática**. 2. ed. São Paulo: Novatec, 2011. 488 p.

JOBSTRAIBIZER, Flávia. **Criação de sites com o CSS**. São Paulo: Digerati Books, 2009. 144 p.

KROENKE, David M. **Sistemas de informação Gerenciais**. São Paulo: Saraiva, 2013. 312 p.

LAUDON, Kenneth C.; LAUNDON, Jane P. **Sistemas de informação gerenciais**. Trad. Arlete Simille Marques. 7. ed. São Paulo: Pearson, 2007. 480 p.

LAUDON, Kenneth; LAUDON, Jane. **Sistemas de Informação Gerenciais**. 9. ed. São Paulo: Pearson, 2011. 448 p.

LOBO, Edson Junio Rodrigues. **Curso prático de MySQL**. São Paulo: Digerati Books, 2008. 128 p.

MILANI, André. **MySQL - Guia do Programador**. São Paulo: Novatec, 2006. 400p.

O'BRIEN, James A.; MARAKAS, George M. **Administração de Sistemas de informação**. 15. ed. Porto Alegre: Mcgraw Hill, 2013. 620p.

PRESSMAN, Roger. S. **Engenharia de software: uma abordagem profissional**. Tradução de Ariovaldo Griesi e Mario Moro Fecchio. 7. ed. Porto Alegre: Mc Graw Hill, bookman, AMGH Editora Ltda., 2011. 780 p.

PRESSMAN, Roger; MAXIM, Bruce. **Engenharia de Software**. 8. ed. Porto Alegre: Mc Graw Hill Education, 2016. 968 p.

RAMOS, Ricardo; SILVA, Joel da; ÁLVARO, Alexandre; AFONSO, Ricardo. **PHP para profissionais**. São Paulo: Universo dos Livros, 2007 p. 138.

REMOALDO, Pedro. **O guia prático do dreamweaver CS3 com PHP, JavaScript e Ajax**. Lisboa, Centro Atlântico, 2008. 676 p.

REVISTABW. As 10 heurísticas de usabilidade de Nielsen. **Revista Brasileira de Web: Tecnologia**. Disponível em <<http://www.revistabw.com.br/revistabw/as-10-heuristicas-de-usabilidade-de-nielsen/>> Acesso em 01 de julho de 2017.

SCHEIDT, Felipe Alex. **Fundamentos de CSS: criando design para sistemas web**. Foz do Iguaçu: Outbox Livros Digitais, 2015. 126 p.

SILVA, Maurício Samy. **Fundamentos de HTML5 e CSS3**. São Paulo: Novatec, 2015. 304 p.