

SISTEMA DE INFORMAÇÕES GERENCIAIS EVOLUTION ARTES MARCIAIS

MAGALHÃES, Felipe Boeira¹

f.maagalhaes@outlook.com

STORNIOLO, Thiago Vaz¹

vaz_vs@hotmail.com

JARDIM, Gabriel Pereira¹

gabriel_pjardim@hotmail.com

PADILHA, Adriano¹

adrianopspadilha@gmail.com

ROMANI, Samuel Etchichury¹

samueel_romani@hotmail.com

BASTOS, Rafael²

rafaelrodriguesbastos@gmail.com

CAMARGO, Salvador L. Tadeu²

salvador@sigasistemas.com.br

FERNANDES, Acauan²

acauanfernandes@gmail.com

PEREIRA, Marcus Vinicius³

marcus_v_sp@hotmail.com

¹ Discentes do Curso de Análise e Desenvolvimento de Sistemas, 2016/2- Faculdade IDEAU – Bagé/RS.

² Docentes do Curso de Análise e Desenvolvimento de Sistemas, 2016/2 - Faculdade IDEAU – Bagé/RS.

³ Coordenador do Curso de Análise e Desenvolvimento de Sistemas - Faculdade IDEAU – Bagé/RS.

RESUMO: O artigo visa à elaboração e concepção de um sistema de informação que permita o registro, consulta e manutenção de informações de uma academia de artes marciais. O sistema terá módulo de login, que permite para cada usuário um login, poderá cadastrar alunos, instrutores, graduações de aluno e data do evento de graduação, poderá fazer consultas, gerar certificados de graduação e relatório dessas informações. Com a implementação desse sistema, pretende-se informatizar o negócio da empresa, suprimindo a necessidade de um melhor cadastro e controle de graduação e acesso a gestão da informação dos alunos e das escolas no estado do Rio Grande do Sul.

Palavras-chave: Sistema de informação; Gestão de academia; Taekwondo; Krav-Maga.

ABSTRACT: The article aims at the development and design of an information system that allows the registration, consultation and maintenance information in a martial arts academy. The system will have a login module that allows each user to a login, can register students, instructors, student grades and date of graduation event, can make queries, generate degree certificates and reports about that information. With the implementation of this system is intended to computerize the company's business, meeting the need for better registration and control of graduation and access to management information of students and schools in the state of Rio Grande do Sul.

Keywords: Information System; Gym management; Taekwondo; Krav-Maga.

1 INTRODUÇÃO

É essencial e indispensável para qualquer empresa um sistema informatizado para o gerenciamento de seus dados. As grandes, médias e pequenas empresas necessitam manter seus dados seguros e atualizados para um melhor controle e gerenciamento.

Os sistemas gerenciais possuem diversas funcionalidades as quais variam de empresa em empresa, e tornam-se essenciais hoje em dia, pois centraliza as informações pertinentes do dia a dia, torna fácil a consulta dessas informações e gera relatórios baseados em filtros personalizados.

A administração da empresa sempre possui suas dificuldades, seja ela grande, média ou pequena, não importa o tamanho de sua empresa, quanto mais fácil for administrá-la menos propensa a problemas será.

O projeto proposto visa aprimorar a administração de uma empresa especializada em artes marciais, a qual possui um sistema onde é realizado o cadastro de alunos e instrutores, só que para realizar consultas é necessário conhecimento em SQL o que deixa a operação do sistema menos eficaz e eficiente.

O sistema a ser desenvolvido visa facilitar e aprimorar o manuseio dos dados dos alunos que estão matriculados na academia. Contendo uma forma mais simples e acessível para efetuar os cadastros de alunos, instrutores, escolas, cidades e eventos.

Também feito de uma forma mais simples e acessível, as consultas por aluno trazem tanto seus dados particulares como seu histórico de graduações dentro da academia.

A consulta por instrutor traz uma lista geral de instrutores com suas respectivas cidades e escolas ou uma consulta que retorna todas as graduações que foram ministradas por ele.

A consulta por graduações que retornara todos os alunos que se graduaram naquele evento e o instrutor que ministrou a graduação, sendo assim a academia possuirá todas as informações necessárias quando quiser e precisar.

2 MATERIAIS E MÉTODOS

Referencial Teórico

Sistemas de informação

Segundo Laudon e Laudon (2011) "O aumento do sistema de informação empresarial gerou uma grande demanda por profissionais que tenham capacidade de projetar e desenvolver novos sistemas de hardware e software para atender as necessidades empresariais."

... A Administração de Sistemas de Informação deve contribuir para o desenvolvimento do processo de planejamento que acompanha tendências da economia, do mercado, das necessidades do cliente, da atuação da concorrência, das tecnologias, do mercado de trabalho e do ambiente legal e social, e transforme as informações levantadas em parâmetros úteis para dirigir os planos. (MEIRELES, 2001, p.10).

Levantamento de requisitos

Segundo Rodriguez e Ferrante (2000) "Uma das condições fundamentais para o total comprometimento dos usuários é que deve-se definir os objetos e formular as especificações do sistema logo no começo do projeto."

De acordo com Mello (2010) "A elicitação de requisitos é necessária para o conhecimento das necessidades do cliente. São realizados encontros entre os analistas e os stakeholders para que os stakeholders informem as suas necessidades e os analistas os orientem. A elicitação busca informações sobre o sistema que deverá ser construído, assim como: suas funcionalidades, restrições, usabilidades, entre outros."

Diagrama Entidade Relacionamento

De acordo com Otti (2010) "O diagrama entidade relacionamento é a forma gráfica do Modelo Entidade Relacionamento. Uma forma mais simples para visualização da estrutura do banco de dados."

Modelo Entidade Relacionamento

De acordo com Heuser (2009) "O modelo entidade relacionamento é um modelo utilizado em Engenharia de Software para o desenvolvimento da entidade relacionamento (ER), onde consiste em descrever os objetos que são as entidades, as suas características que são os atributos de cada entidade e como elas se relacionam."

Entidades: De acordo com Takai, Italiano e Ferreira (2005) "As entidades são partes do domínio. Dependendo de sua existência elas são classificadas como físicas ou lógicas."

Entidades físicas: São aquelas realmente visíveis e existentes no mundo real. Como por exemplo, roupas, carros, clientes, empresas, entre outros. **Entidades lógicas:** São aquelas que são criadas a partir do relacionamento entre entidades físicas, que fazem sentido dentro daquele domínio de negócio.”

Atributos: De acordo com Takai, Italiano e Ferreira (2005) “Atributos descrevem, caracterizam as entidades. Por exemplo, entidade "Carros" terá os atributos: cor, chassi, modelo, ano, entre outras características.”

Relacionamentos: De acordo Takai, Italiano e Ferreira (2005) “Relacionam as entidades, interligando-as. Podemos classificar de três formas os relacionamentos, que são: **Relacionamento 1:1:** as duas entidades que estão se relacionando só podem se envolver com um registro da outra. **Relacionamento 1:n:** uma das duas entidades pode se relacionar com vários registros da outra, mas uma delas só poderá se relacionar com um único registro da outra. **Relacionamento n:m:** ambas as entidades podem se relacionar com muitos registros da outra, mas para isso é gerada uma terceira tabela para manter as chaves das outras duas.”

Dicionário de dados

De acordo com IIBA (2011) “O dicionário de dados que é a documentação que explica todos os objetos que nele se encontram. Ele auxilia os analistas a obter informações sobre todo e qualquer objeto do modelo de forma textual.”

Modelagem de interface

De acordo com Siqueira (2001) “A modelagem de interface representa toda a interação usuário-sistema. Primeiramente é desenvolvido o modelo conceitual, onde apenas têm-se as telas e após desenvolve-se o modelo navegacional o qual já é possível navegar entre telas sendo assim possível identificar os objetos navegacionais que serão utilizados no sistema.”

Object Pascal

De acordo com Fontes (2010) "O Object Pascal é uma linguagem muito bem estruturada que permite ao programador dividir seu programa em partes com suas funcionalidades e torna a linguagem mais vantajosa, pois permite utilizar vários tipos de dados e diversas estruturas de controle."

Segundo David (2007) "A programação orientada a objetos (POO) foi criada e desenvolvida para unir o mais próximo possível o mundo real do virtual. Para realizar isso é necessário fazer uma simulação do mundo real no computador utilizando uma diversa variedade de objetos que existe na plataforma utilizada."

Materiais e Métodos

Microsoft Office Word – O Word é um processador de textos com várias funções que facilitam o trabalho de quem o utiliza. Com o Word foi elaborada a **Elicitação de requisitos** e criou-se o **Dicionário de dados** que consiste em uma lista de todos os elementos que serão necessários no sistema.

Diagram Designer - É uma ferramenta de criação e edição de imagens específico para diagramas e organogramas desenvolvido pela logicnet.dk. O Diagram Designer é open source de licença de uso GPLv2, porta e foi utilizado para criar o **diagrama de entidade e relacionamento (DER)**.

MySQL Workbench - É uma ferramenta para modelagem de bancos de dados de linguagem SQL desenvolvida pela Oracle Corporation. Foi adotada a versão Community, de licença de uso gratuita do tipo BSD para a modelagem de entidade-relacionamento do banco de dados do sistema. A ferramenta workbench foi utilizada para a criação do **modelo entidade relacionamento (MER)**.

O PostgreSQL - É um sistema de gerenciamento de banco de dados objeto-relacional baseado no POSTGRES Versão 4.2 e desenvolvido pela PostgreSQL Global Development Group. O PostgreSQL é open source de licença BSD de uso e foi adotado como ferramenta de gerenciamento de banco de dados. O postgresQL foi utilizado para a execução das sentenças **SQL** assim criando o **banco de dados**.

Lazarus - É um ambiente de desenvolvimento integrado para Object Pascal desenvolvido pela comunidade open source, que é compatível com Delphi e suporta vários sistemas operacionais e arquiteturas. Foi adotado o Lazarus Object Pascal como API de programação do sistema. O Lazarus Object Pascal foi utilizado para a **modelagem de interface** e desenvolvimento do sistema em **object pascal**.

Metodologia

Para o desenvolvimento deste sistema a metodologia aborda foi a elicitação de requisitos, diagrama entidade relacionamento, modelo entidade relacionamento, dicionário de dados, PostgreSQL e modelagem de interface e object pascal.

- **Elicitação de requisitos**

Entre reuniões e encontros com os stakeholders obteve-se as seguintes solicitações do sistema: cadastro de alunos, cadastro de instrutores, cadastro de escolas e suas cidades, cadastro dos eventos de graduações, cadastro dos alunos na graduação, consulta por instrutor, consulta por aluno, consulta por graduação. Também foi informado que todo instrutor é um aluno e que as escolas estão localizadas apenas no estado do Rio Grande do Sul.

- **Diagrama entidade relacionamento (DER)**

O diagrama entidade relacionamento foi desenvolvido com base no levantamento dos requisitos o qual possibilitou uma apresentação gráfica dos possíveis registros do sistema para os stakeholders e auxiliou os analistas no desenvolvimento do modelo entidade relacionamento, pois mostra as devidas relações entre as entidades com sua cardinalidade.

- **Modelo entidade relacionamento (MER)**

Com base no diagrama entidade relacionamento foi desenvolvido o modelo entidade relacionamento que com um grau de abstração inferior possibilitou o desenvolvimento do banco de dados pelos analistas.

- **Dicionário de dados**

Após a realização do modelo entidade relacionamento foi realizado a criação do dicionário de dados o qual facilitará uma atualização futura no sistema, pois constam todas as entidades e atributos do banco de dados descritos e explicados.

- **PostgreSQL**

Foi utilizado o workbench para gerar o código SQL o qual executamos no PostgreSQL. A linguagem SQL é utilizada por muitos SGBD's só que alguns sistemas gerenciais de banco de dados implementam algumas funções, tipos, entre outros a mais. Como o próprio nome diz "Workbench MySQL" ele gera o código mais adaptado para as

mudanças e padrões do MySQL. Então tivemos que fazer algumas alterações. Como mostra a **Figura 1 – Sentenças SQL do WorkBench.**

```

1 CREATE TABLE IF NOT EXISTS `mydb`.`alunos` (
2   `codigo` INT(999) NOT NULL,
3   `nome` VARCHAR(150) NULL,
4   `data_nascimento` DATE NULL,
5   `cpf` VARCHAR(45) NULL,
6   `rg` VARCHAR(45) NULL,
7   `matricula` INT NULL AUTO INCREMENT,
8   `responsavel` VARCHAR(120) NULL,
9   `endereco` VARCHAR(150) NULL,
10  `tel_residencial` VARCHAR(45) NULL,
11  `tel_celular` VARCHAR(45) NULL,
12  `email` VARCHAR(150) NULL,
13  `obsevacao` LONGTEXT NULL,
14  PRIMARY KEY (`codigo`))
15
16 CREATE TABLE IF NOT EXISTS `mydb`.`modalidades` (
17   `codigo` INT(999) NOT NULL,
18   `descricao` VARCHAR(45) NULL,
19   `mod` VARCHAR(1) NULL,
20   PRIMARY KEY (`codigo`))
21 ENGINE = InnoDB;
22
23 CREATE TABLE IF NOT EXISTS `mydb`.`modalidades_has_alunos` (
24   `modalidades_codigo` INT NOT NULL,
25   `alunos_codigo` INT NOT NULL,
26   PRIMARY KEY (`modalidades_codigo`, `alunos_codigo`),
27   INDEX `fk_modalidades_has_alunos_alunos1_idx` (`alunos_codigo` ASC),
28   INDEX `fk_modalidades_has_alunos_modalidades1_idx` (`modalidades_codigo` ASC),
29   CONSTRAINT `fk_modalidades_has_alunos_modalidades1`
30     FOREIGN KEY (`modalidades_codigo`)
31     REFERENCES `mydb`.`modalidades` (`codigo`)
32     ON DELETE NO ACTION
33     ON UPDATE NO ACTION,
34   CONSTRAINT `fk_modalidades_has_alunos_alunos1`
35     FOREIGN KEY (`alunos_codigo`)
36     REFERENCES `mydb`.`alunos` (`codigo`)
37     ON DELETE NO ACTION
38     ON UPDATE NO ACTION)

```

Figura 1 - Sentenças SQL do WorkBench

Após as edições realizadas no código gerado pelo Workbench obtivemos o resultado como mostra a **Figura 2 – Sentenças SQL para PostgreSQL:**

```

1 CREATE TABLE alunos (
2   codigo serial primary key,
3   nome VARCHAR(150) NULL,
4   data_nascimento DATE NULL,
5   cpf VARCHAR(45) NULL,
6   rg VARCHAR(45) NULL,
7   matricula serial,
8   responsavel VARCHAR(120) NULL,
9   endereco VARCHAR(150) NULL,
10  tel_residencial VARCHAR(45) NULL,
11  tel_celular VARCHAR(45) NULL,
12  email VARCHAR(150) NULL,
13  obsevacao LONGTEXT NULL);
14
15 CREATE TABLE modalidades (
16   codigo serial primary key,
17   descricao VARCHAR(45) NULL,
18   mod VARCHAR(1) NULL);
19
20 CREATE TABLE modalidades_has_alunos (
21   modalidades_codigo INT NOT NULL references modalidades(codigo),
22   alunos_codigo INT NOT NULL references alunos(codigo),
23   PRIMARY KEY (modalidades_codigo, alunos_codigo));

```

Figura 2 - Sentenças SQL para PostgreSQL

Como podemos ver no PostgreSQL usamos o tipo “serial” que é basicamente um integer só que associado a uma sequência, ou seja, ele passa a ter incremento automático. Sua estrutura fica da seguinte forma: NomeTabela_NomeCampo_seq. O serial aceita até um limite de 2.147.483.648, mas caso necessite de mais pode-se usar o bigserial que aí faz referência ao bigint.

- **Modelagem de interface e Object Pascal**

Foi utilizado o Object Pascal para a criação do protótipo de tela pela sua facilidade em desenvolver telas, assim ficando mais fácil a comunicação e alteração, se necessária, com os stakeholders. Após a prototipação das telas o desenvolvimento do software foi realizado com a ferramenta Lazarus.

3 RESULTADOS E ANÁLISE

Com nossa análise e desenvolvimento foram obtidos os seguintes resultados de modelagem de dados e interface do sistema:

Modelagem de dados

Com o desenvolvimento do modelo entidade relacionamento foi obtido a seguinte estrutura como mostra a **Figura 3 – Modelo entidade relacionamento.**

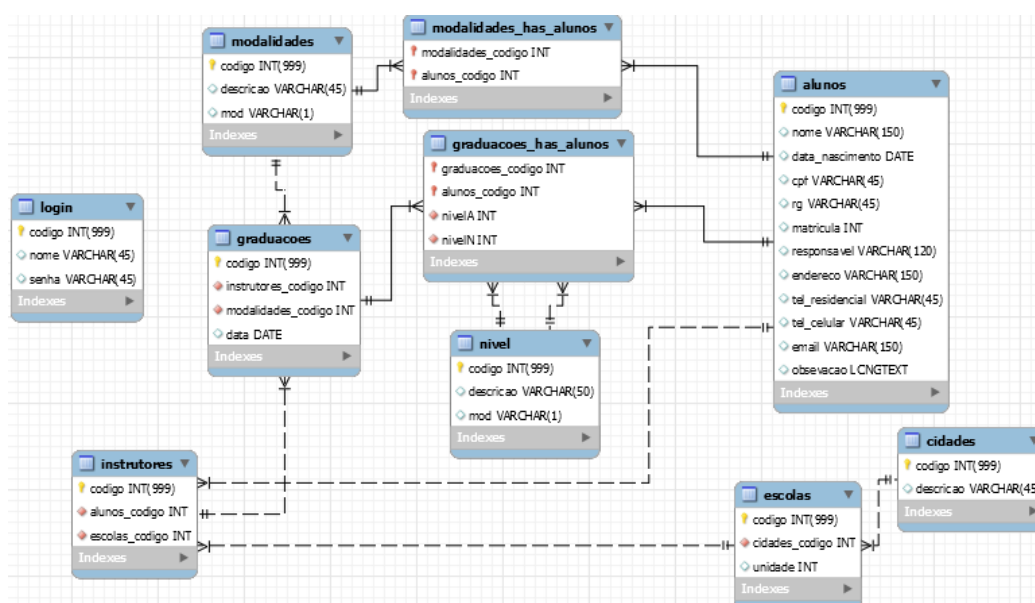


Figura 3 – Modelo entidade relacionamento.

Com a modelagem do banco de dados foram documentados suas entidades e atributos com o desenvolvimento do dicionário de dados.

Tabela alunos: Usada para cadastrar todos os alunos que estão matriculados na academia, com seus dados pessoais e contatos para eventual comunicação.

Nome:	Tipo:	Tamanho:	Descrição:
Codigo (PK)	Serial	-	Número (único) identificador de registro.
Nome	Varchar	45	Nome do aluno a ser cadastrado.
Endereco	Varchar	45	Endereço do aluno a ser cadastrado.
Matricula	Serial	-	Número identificador do aluno na academia
Data_nascimento	Date	-	Data de nascimento do aluno.
CPF	Varchar	45	Cadastro de pessoa física do aluno ou responsável caso haja.
RG	Varchar	45	Registro geral (carteira de identidade) do aluno.
Responsavel	Varchar	120	Nome do responsável do aluno, caso necessário.
Tel_residencial	Varchar	45	Telefone residencial para contato.
Tel_celular	Varchar	45	Telefone celular para contato.
Email	Varchar	150	Email do aluno para contato.
Observacao	Longtext	-	Campo de texto livre para observações.

Tabela 1 - Tabela alunos

Tabela instrutores: Utilizada para cadastro dos instrutores, o associando a uma escola previamente cadastrada. Todo instrutores deve ser um aluno.

Nome:	Tipo:	Tamanho:	Descrição:
Codigo (PK)	Serial	-	Número (único) identificador de registro.
Alunos_codigo (FK)	Integer	-	Número que identifica qual aluno é um instrutor.
Escolas_codigo (FK)	Integer	-	Número que identifica a escola do instrutor.

Tabela 2 - Tabela instrutores

Tabela escolas: Utilizada para cadastrar todas as escolas da Evolution Artes Marciais, como sua unidade e cidade.

Nome:	Tipo:	Tamanho:	Descrição:
Codigo (PK)	Serial	-	Número (único) identificador de registro.
Cidades_codigo (FK)	Integer	-	Número identificador da cidade em relação à escola.
Unidade	Integer	-	Número identificador da unidade. Ex.: 01, 02, 03...

Tabela 3 - Tabela escolas

Tabela cidades: Utilizada para o cadastramento das cidades. Apenas do Rio Grande do Sul.

Nome:	Tipo:	Tamanho:	Descrição:
Codigo (PK)	Serial	-	Número (único) identificador de registro.

Descricao	Varchar	45	Utilizado para registrar o nome da cidade.
------------------	---------	----	--

Tabela 4 - Tabela cidades

Tabela modalidades: Mantém as modalidades da academia armazenadas.

Nome:	Tipo:	Tamanho:	Descrição:
Codigo (PK)	Serial	-	Número (único) identificador de registro.
Descricao	Varchar	45	Utilizado para cadastrar os nomes das modalidades (solicitadas): - Krav-Maga; - Taekwondo;

Tabela 5 - Tabela modalidades

Tabela modalidades_has_alunos: Utilizada para manter uma relação entre os alunos e a(s) modalidade(s) que esta(ão) realizando.

Nome:	Tipo:	Tamanho:	Descrição:
Modalidades_codigo (FK) (PK)	Integer	-	Número identificador da modalidade.
Alunos_codigo (FK) (PK)	Integer	-	Número identificador do aluno.

Tabela 6 - Tabela modalidades_has_alunos

Tabelas graduacoes: Utilizada para cadastrar o evento de graduação referenciando a modalidade e o instrutor.

Nome:	Tipo:	Tamanho:	Descrição:
Codigo (PK)	Serial	-	Número (único) identificador de registro.
Instrutores_codigo (FK)	Integer	-	Número identificador do instrutor.
Modalidades_codigo (FK)	Integer	-	Número identificador da modalidade.
Data	Date	-	Data que o evento será realizado.

Tabela 7 - Tabela graduações

Tabela graduacoes_has_alunos: Utilizada para cadastrar o aluno e uma ou mais graduações com o seu nível.

Nome:	Tipo:	Tamanho:	Descrição:
Graduacoes_codigo (FK) (PK)	Integer	-	Número identificador da graduação.
Alunos_codigo (FK) (PK)	Integer	-	Número identificador do aluno.
Nivela (FK)	Integer	-	Número referente ao antigo nível do aluno.
Niveln (FK)	Integer	-	Número referente ao novo nível do aluno.

Tabela 8 - Tabela graduacoes_has_alunos

Tabela nivel: Utilizada para armazenar os níveis de cada modalidade.

Nome:	Tipo:	Tamanho:	Descrição:
Codigo (PK)	Serial	-	Número (único) identificador do registro.
Descricao	Varchar	50	O nome dos níveis: - Defender – A1;

			- Defender – A2; - Defender – A3; - Defender – A4; - Weapon – A1; - Weapon – A2; - Weapon – A3; - Weapon – A4; - Combat – A1; - Combat – A2; - Combat – A3; - Combat – A4; - Expert nível fácil; - Branca; - Laranja; - Amarela; - Verde; - Azul recomendada; - Azul decidida; - Marrom recomendada; - Marrom decidida; - Vermelha recomendada; - Vermelha decidida; - Vermelha e preta; - Preta 1º DAN; - Preta 2º recomendada; - Preta 2º DAN; - Preta 3º DAN; - Preta 4º DAN; - 1º Parcial; - 2º Parcial; - 3º Parcial; - 4º Parcial; - 5º Parcial; - 6º Parcial; - 7º Parcial; - 8º Parcial; - 9º Parcial; - 10º Parcial;
Mod	Varchar	1	Diferenciar os níveis de Krav-Maga (k) e Taekwondo (t).

Tabela 9 - Tabela nível

Tabela login: Utilizada para manter os usuários com acesso ao sistema.

Nome:	Tipo:	Tamanho:	Descrição:
codigo (PK)	Serial	-	Número identificador do usuário.
Nome	Varchar	45	Nome do usuário (para login).
Senha	Varchar	45	Senha do usuário (para login).

Tabela 10 - Tabela login

Interface

O sistema consta com um modulo de autenticação, ao abrir o software ele solicitará usuário e senha como mostra a **Figura 4 – Tela de login.**



Figura 4 - Tela de login

Caso seja informado o usuário ou a senha que não correspondam com os usuários cadastrados no banco de dados o sistema informará que os dados não correspondem e solicitará que o usuário tente fazer o login novamente.

Com login efetuado com sucesso chegamos ao menu principal do sistema o qual nos oferece varias funções de cadastramento como mostra a **Figura 5 – Menu de cadastros.**

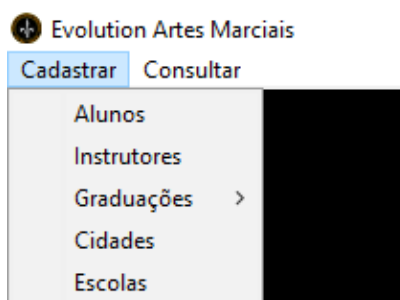


Figura 5 - Menu de cadastros

No menu **cadastrear > alunos** podemos cadastrar os novos alunos com seus dados pessoais, nome, RG, CPF, telefone celular, telefone residencial, responsável (caso haja), data

de nascimento, matrícula (gera automaticamente ao abrir o cadastramento de alunos), endereço, e-mail, observação e a suas modalidades (que pode ser mais de uma).

Na opção **cadastrar** > **instrutores** usamos para cadastrar os instrutores, onde todo aluno é um instrutor e associamos a ele uma escola.

Quando for selecionado o menu **cadastrar** > **graduações** > **aluno** > **krav-maga** nesta parte podemos cadastrar nos alunos que participarão das graduações do krav-maga.

Ao clicar em **cadastrar** > **graduações** > **aluno** > **taekwondo** nesta parte cadastramos os alunos que participarão das graduações do taekwondo.

Ao selecionar **cadastrar** > **graduações** > **evento** neste menu podemos cadastrar os dias das graduações fazendo referencia a modalidade e ao instrutor responsável.

Quando clicar em **cadastrar** > **cidades** podemos cadastrar novas cidades para referenciar as escolas.

No menu **cadastrar** > **escolas** neste menu podemos cadastrar as escolas e referenciar a cidade a qual ela se encontra.

Como mostram as figuras: **Figura 6 – Tela de cadastro de alunos** e **Figura 7 – Tela cadastro de eventos**



Evolution Artes Marciais

Cadastro de Alunos

Matrícula: 7

Nome*:

CPF*:

Endereço*:

Tel. Res.*:

E-mail:

Obs.:

Responsável:

RG:

Data de nascimento*:

Tel. Cel.*:

Modalidades*: Krav-Maga
Taekwondo

Um dos campos, Tel.Res ou Tel.Cel é obrigatório estar preenchido.

Cancelar Salvar

Figura 6 – Tela de cadastro de alunos

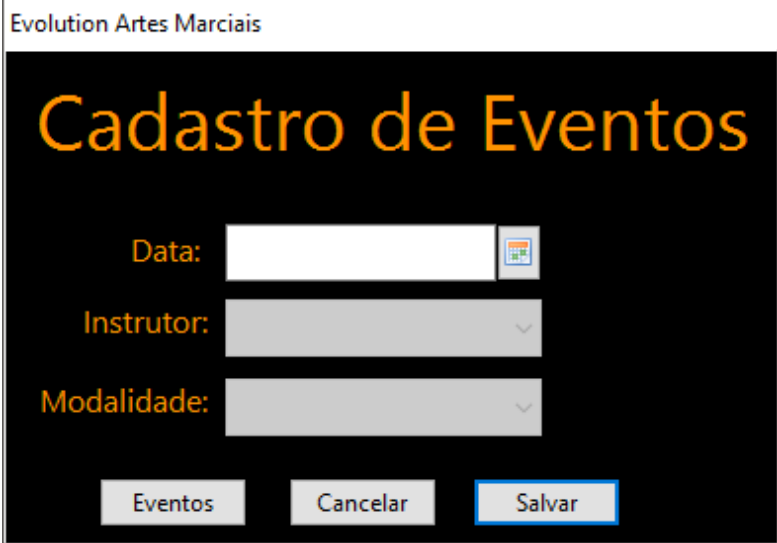


Figura 7 – Tela de cadastro de eventos

Após o menu cadastrar temos o menu consultar que nos possibilita retornar os dados inseridos como mostra a **Figura 8 – Menu consultar**.

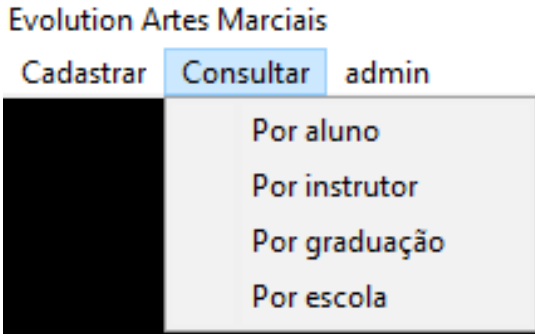


Figura 8 – Menu consultar

Quando selecionar o menu **consultar** > **Por aluno** esta opção te possibilita fazer dois tipos de pesquisa. Uma delas retorna os dados pessoais do aluno cadastro e a outra opção te retorna todo o histórico de graduações do aluno.

Ao clicar em **consultar** > **Por instrutor** possibilita retornar os dados pessoais do instrutor ou todas as graduações a qual ele instruiu.

Após abrir **consultar** > **graduações** pode-se pesquisar todas as graduações que ocorreram entre dois períodos ou pesquisar todos os alunos que participaram de uma graduação específica.

No menu **consultar** > **escolas** traz uma relação de todos os instrutores daquela escola.

Como mostra a **Figura 9 – Telas de consultas por aluno** e **Figura 10 – Tela de consulta por graduações**.

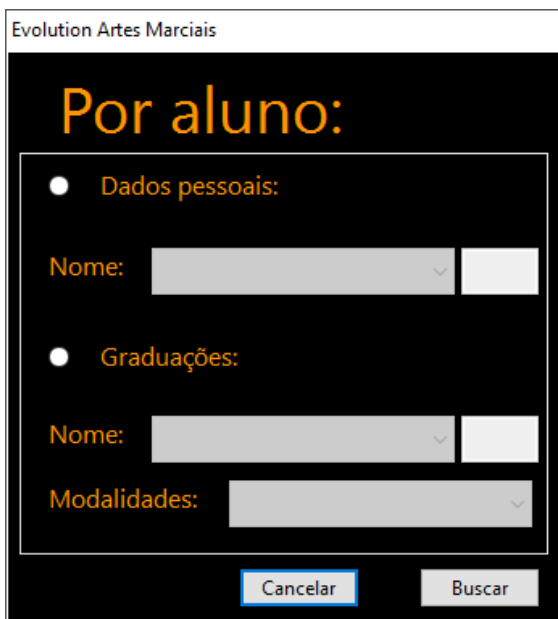


Figura 9 – Tela de consulta por aluno

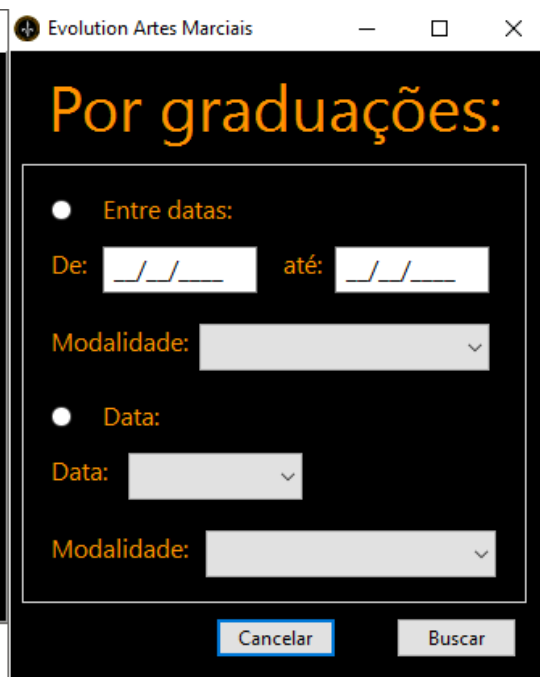


Figura 10 – Tela de consulta por graduações

Após o menu consultar temos o menu é que o nome do usuário logado, como mostra a **Figura 11 – Menu do usuário.**

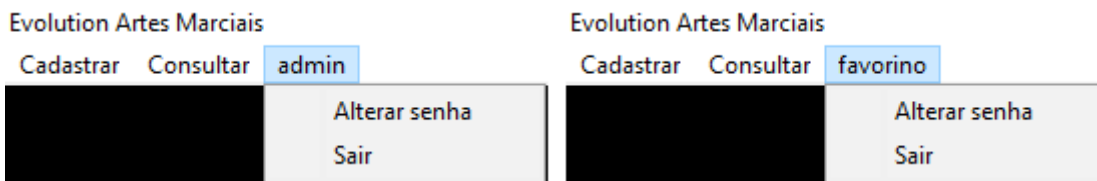


Figura 11 – Menu do usuário

Neste menu temos duas opções alterar senha que possibilita a troca de senha (como mostra a **Figura 12 – Tela de alterar senha**) por parte do usuário e o botão sair o qual encerra o programa.



Figura 12 – Tela de alterar senha

CONCLUSÃO

Foi abordado o gerenciamento de informações de uma empresa especializada em artes marciais, onde o software desenvolvido facilita o gerenciamento dos alunos, graduações, escolas e instrutores da academia.

O software foi planejado e desenvolvido durante o processo do trabalho toma lugar de um sistema defasado onde esse cadastramento era feito apenas por usuários que tinham conhecimento em linguagem SQL. Todos os objetivos propostos foram atingidos.

Com esse trabalho foi possível aperfeiçoar as técnicas de programação em grupo, assim como, ampliar o conhecimento na área de planejamento e desenvolvimento de sistemas na área do domínio do negócio.

REFERÊNCIAS

DAVID, M. F.: **Programação Orientada a Objetos: uma introdução**. Disponível em: <<http://www.hardware.com.br/artigos/programacao-orientada-objetos/>>. Acesso em: 10, Set, 2016.

FONTES, E. A. D. S.: **Desenvolvimento de Software educativo para o ensino de semicondutores**. Disponível em: <http://paginas.fe.up.pt/~ee03095/Tese_Software_ensino_semicondutores_ee03095.pdf>. Acessado em: 20, Set, 2016.

HEUSER, C.A.: **Projeto de banco de dados**, Porto Alegre, Bookman Editora, 2009.

IIBA: **Um guia para o Corpo de conhecimento de análise de negócios**, Toronto, IIBA, 2011.

LAUDON, K. C. e LAUDON, J. P.: **Sistemas de Informação Gerenciais**, São Paulo, PEARSON BRASIL, 2011.

MEIRELES, M.: **Sistemas de Informação**, São Paulo, Arte & Ciência, 2001.

MELLO, L.C.S.: **Levantamento de requisitos**. Disponível em: <http://www.ice.edu.br/TNX/encontrocomputacao/artigos-internos/aluno_leandro_cicero_levantamento_de_requisitos.pdf>. Acesso em: 29, Set, 2016.

OTTI, A.L.C.: **Engenharia de software**. Disponível em: <<http://www.ppgel.net.br/uaisoccer/downloads/1285018147.pdf>> Acesso em: 29, Set, 2016.

RODRIGUEZ, M. V. R. Y. e FERRANTE, A. J.: **Tecnologia de Informação e Gestão Empresarial**, Rio de Janeiro, E-Papers, 2000.

SIQUEIRA, E.G.: **Estratégias e padrões para a modelagem de interface humano-computador de sistemas baseados na arquitetura softboard**. Disponível em: < <http://mtc->



INSTITUTO DE DESENVOLVIMENTO
EDUCACIONAL DO ALTO URUGUAI
FACULDADES IDEAU



m16.sid.inpe.br/col/sid. inpe.br/jeferson/2003/08.08.09.13/doc/publicacao.pdf> Acessado em: 30, Set, 2016.

TAKAI, O.K., ITALIANO, I.C. e FERREIRA, J.E.: **Introdução a banco de dados.** Acessado em: < <https://www.ime.usp.br/~jef/apostila.pdf> > Acessado em: 29, Set, 2016.