UNIVERSIDADE FEDERAL DO PARANÁ

Sistema de gerenciamento de maratona

CURITIBA

2016

arion assao kamitani

joão pedro medeiros de farias

joão victor alves da maia

Sistema de gerenciamento de maratona

Trabalho apresentado como requisito parcial para aprovação na disciplina de Implementação de Aplicação para Computador do curso de Tecnologia em Análise e Desenvolvimento de Sistemas da Universidade Federal do Paraná.

Orientador: Prof. Dr. Luiz Antônio Pereira Neves.

CURITIBA

2016

Sumário

LISTA DE ILUSTRAÇÕES v

1 Introdução 1

2 prOTOTIPAÇÃO DE TELAS 3

2.1 introdução 3

2.2 página inicial 3

2.3 Login 4

2.4 cadastrar 5

2.5 inscrever 6

2.6 resultados 8

2.7 administrador 8

2.8 fluxo de telas 9

2.9 conclusão 10

3 modelagem de dados 11

3.1 Modelo entidade-Relacionamento 11

3.2 dicionário de dados 12

3.3 Conclusão 13

4 script banco de dados 15

4.1 SCRIPT PARA criação das TABELAS 15

4.2 script para inSERÇÃO DE DADOS 17

4.3 Conclusão 18

5 DIAGRAMA de FLUXO DE DADOS 20

5.1 Diagrama de contexto 20

5.2 DFD NÍVEL 0 20

5.3 conclusão 21

6 conclusão 22

LISTA DE ILUSTRAÇÕES

FIGURA 1 - PAGINA INICIAL 4

FIGURA 2 - lOGIN 5

FIGURA 3 - CADASTRAR 6

FIGURA 4 - INSCREVER 7

FIGURA 5 - RESULTADOS 8

FIGURA 6 - Administrador 9

FIGURA 7 - FLUXO DE TELAS 10

FIGURA 8 - modelo entidade-relacionamento 11

FIGURA 9 - Diagrama de contexto 20

FIGURA 10 - DFD nível 0 21

# Introdução

C om o passar do tempo, os sistemas de gerenciamento tem se tornado cada vez mais recorrentes na vida das empresas e das pessoas. O avanço na capacidade de processamento dos computadores permite a construção de interfaces ergonomicas e intuitivas as quais são capazes de atrair usuários cada vez mais leigos no que diz respeito à aspectos técnicos da computação.

Hoje convivemos normalmente com sistemas bancários que possibilitam todas as operações bancárias que antes eram feitas apenas em agências. Tudo isso em uma série de plataformas como dispositivos móveis e browsers. Além disso, oferecem recursos de segurança extremamente avançados como tokens e algoritmos complexos de criptografia assimétrica através de pares de chaves públicas e privadas que garantem a integridade dos dados compartilhados pela rede e a autenticação dos envolvidos (remetente e destinatário). Além dos softwares rodarem em diferentes plataformas, também apresentam funcionalidades como processamento de imagens para pagamento de boletos, reconhecimento facial, processamento automático de documentos fiscais entre outras.

Com tantos recursos oferecidos em diferentes plataformas com funcionalidades que envolvem alto nível de processamento e complexidade nos seus algoritmos, é necessário que surjam tecnologias e protocolos de comunicação que possibilitem esse tipo de desenvolvimento sem perda de desempenho e agilidade no processo da construção do software.

Durante a matéria de Implementação de Aplicação para Computadores do curso de Análise e Desenvolvimento de Sistemas da Universidade Federal do Paraná, foi sugerido a construção de um software para Web com a finalidade de cadastro e acompanhamento de maratonas de corridas. O sistema apresenta registro em uma maratona específica, sistema de login, cadastro dos tempos de cada competidor, seleção de kits de participação, pagamento dos mesmos e acompanhamento de desempenho. Considerando a tendência das tecnologias de desenvolvimento de sistemas no mundo todo, escolheram-se, para a construção desse software, arquiteturas e frameworks compatíveis com o que é sugerido pelo mercado de Web.

Entre as tecnologias escolhidas está o Bootstrap para construção do layout do site, e a arquitetura Restful para a comunicação entre cliente/servidor, permitindo, assim, que a mesma implementação de regra de negócio seja utilizada para diferentes plataformas. A linguagem escolhida para o back-end foi o PHP versão 5, e, para o front-end, HTML5 e JavaScript, os quais são tendências no mercado devido a sua praticidade de migração para dispositivos móveis através de frameworks open-source como o Ionic por exemplo.

Além disso, são abordadas características de documentação e modelagem de sistemas estruturada como diagrama entidade-relacionamento (DER) dicionário de tabelas, modelo entidade-relacionamento (MER), fluxo de telas e diagrama de fluxo de dados (DFD).

# prOTOTIPAÇÃO DE TELAS

## introdução

A prototipação é utilizada como base para o desenvolvimento de softwares, pois assim pode-se ter uma visão geral de como será o resultado final do sistema. No caso da prototipação realizada neste sistema, foi utilizado o software Balsamiq devido ao fato deste possuir bons atributos para produção de diagramas e telas.

## página inicial

A tela inicial do site (FIGURA 1) o usuário pode verificar informações básicas da maratona, como os kits disponíveis, o percurso e os patrocinadores. E poderá acessar as paginas de login ou cadastro, resultados e de inscrição. O menu funcionará de forma a que caso o usuário clique nas opções percurso, kits, informações ou galeria, será direcionado através do scroll da pagina inicial.

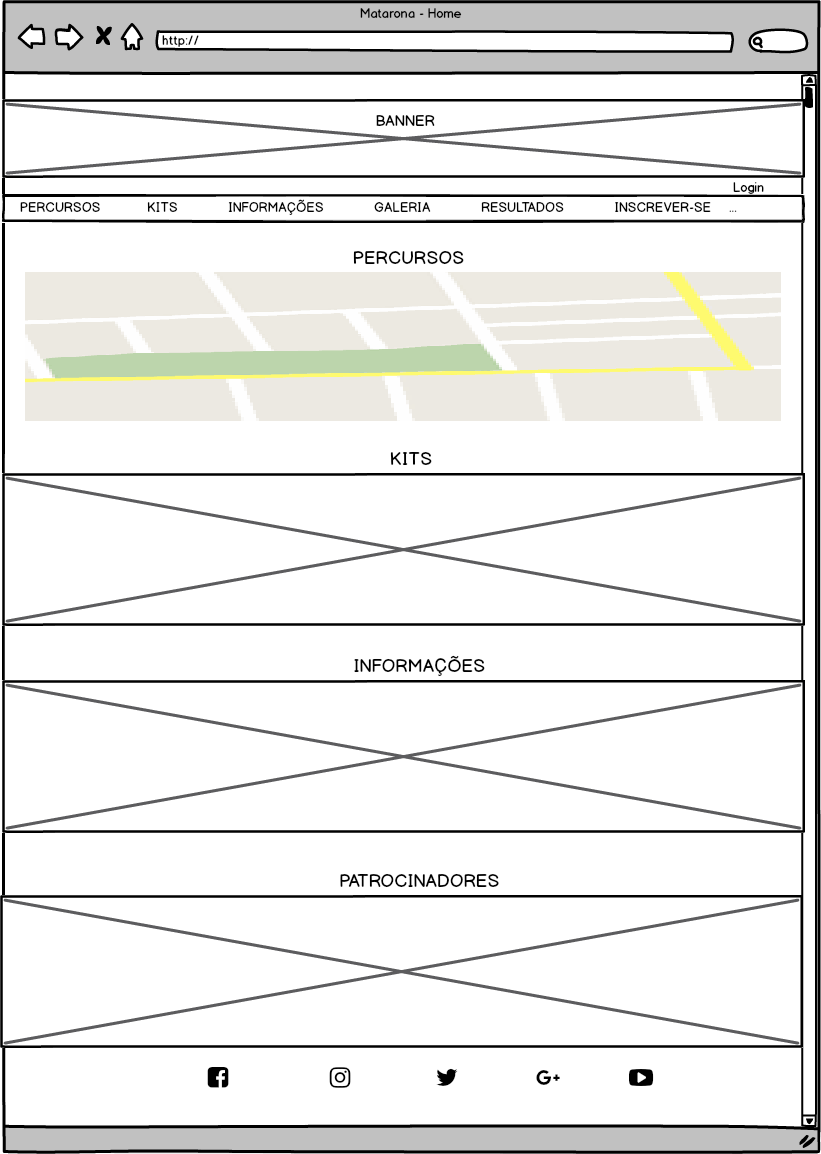


Figura - pagina inicial

## Login

Caso o usuário já possua cadastro no site, ele poderá ir direto para a tela de login, nesta pagina o usuário poderá visualizar a imagem de marketing da empresa de maratonas, bem como o campo com os dados necessários para acessar sua conta com sucesso. Ainda conta com a opção de lembrar a senha, caso ele a tenha esquecido, ou ainda, de criar uma nova conta (FIGURA 2).

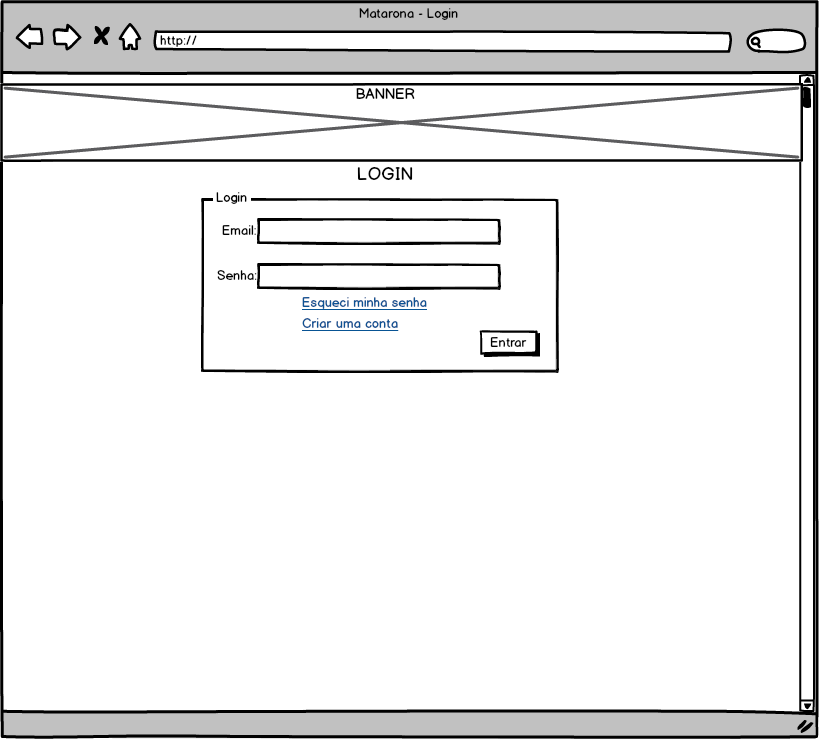


Figura – login

## cadastrar

Se for a primeira vez que o usuário está entrando no site, ele poderá fazer um cadastro acessando o link “criar nova conta” na pagina de Login. Para que isso aconteça, deverá inserir nos campos os dados solicitados, cujo os mesmos serão salvos em seu cadastro assim que o usuário clicar no botão “enviar” (FIGURA 3).

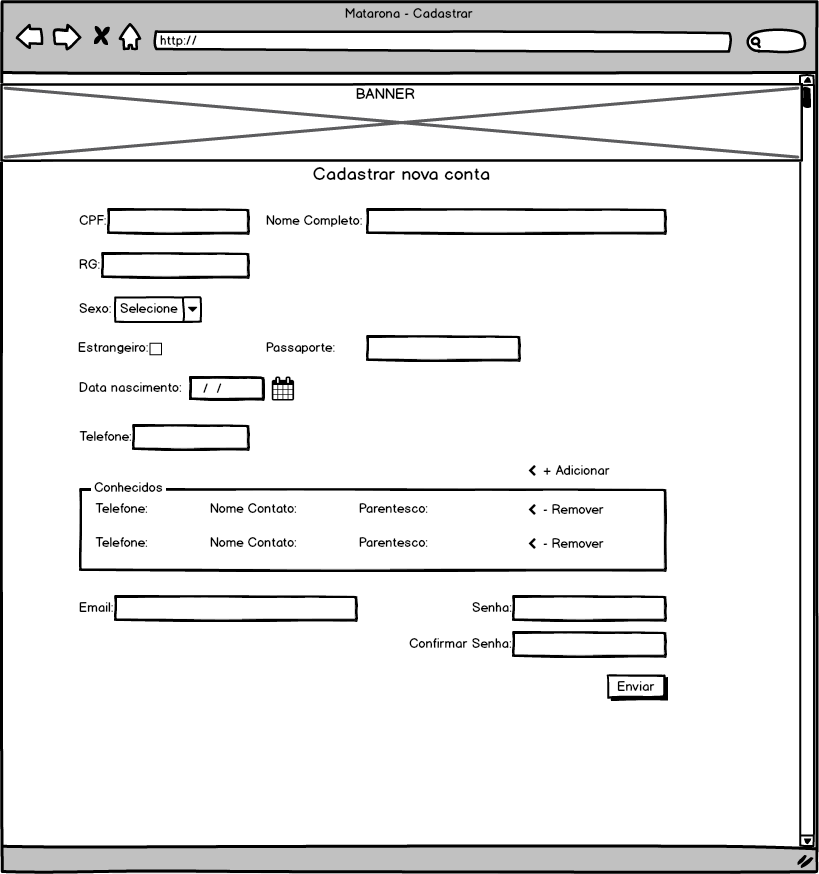


Figura – cadastrar

## inscrever

Após o login, o usuário poderá acessar a pagina de inscrição através do menu da pagina inicial, nesta seção deverá escolher o kit e a modalidade que deseja competir (apenas uma opção de cada pode ser escolhida) e logo abaixo realizar o pagamento para que seja incrito na maratona e receba seu número serial de inscrição. Nesta página poderá visualizar a descrição de cada kit e visualizar o percurso para cada modalidade (FIGURA 4).

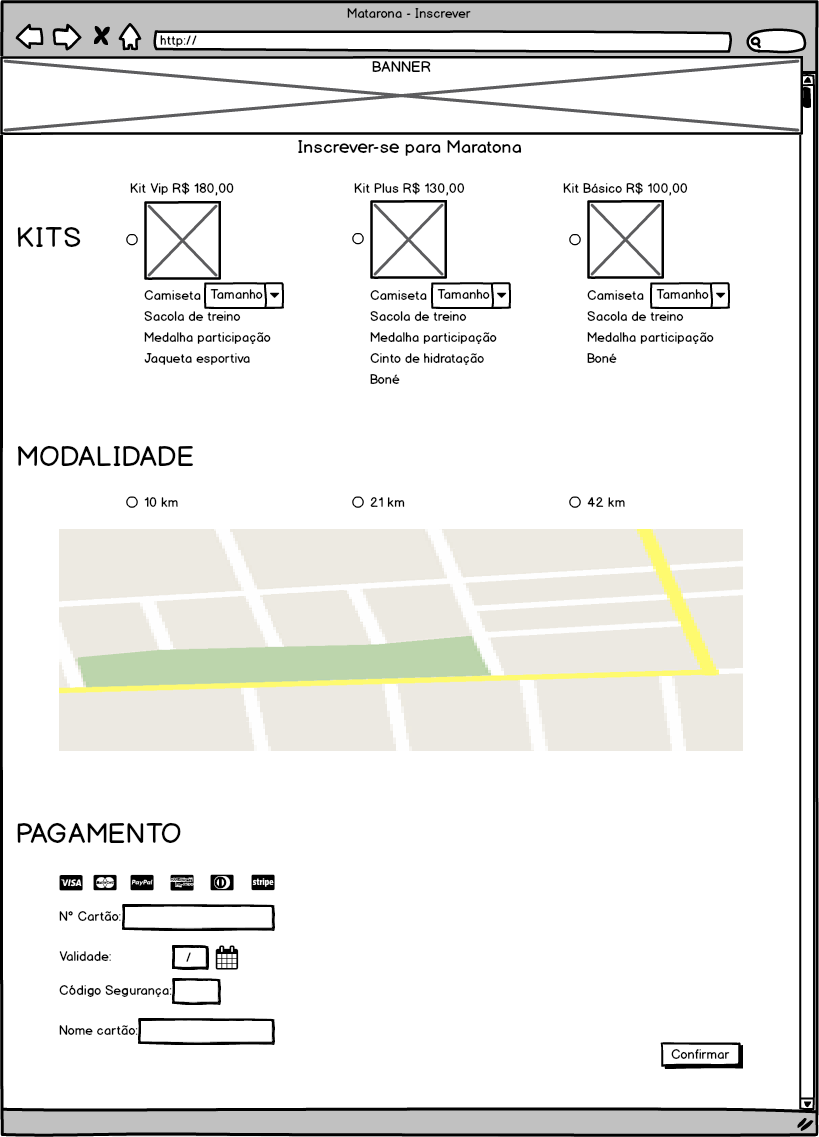


Figura – inscrever

## resultados

Após o termino da corrida, a página de resultados pode ser acessada no menu da página inicial, nela o usuário poderá ver seus resultados individuais, bem como buscar outros resultados por nome, número de inscrição, categoria e modalidade. Após clicar no botão “buscar”, uma tabela será criada de forma a mostrar os resultados obtidos através da busca feita pelo usuário. (FIGURA 5).

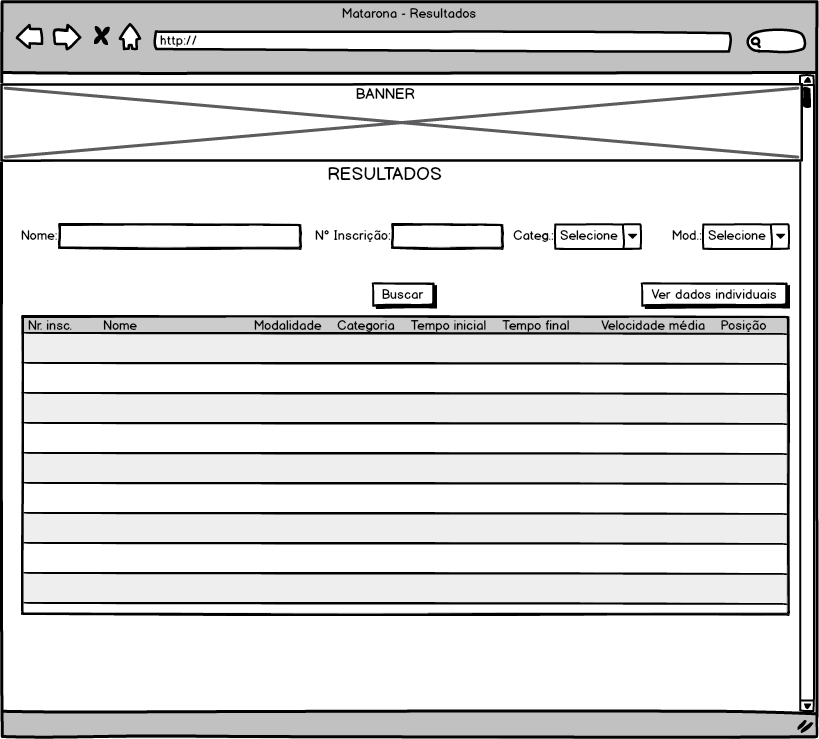


Figura - resultados

## administrador

O Administrador do sistema pode ter acessoa a esta pagina através de um login cadastrado no sistema. Nesta pagina o administrador insere os tempos de cada corredor dando inicio á maratona e após o término da maratona, esses dados inseridos podem ser visualizados na página de Resultados.

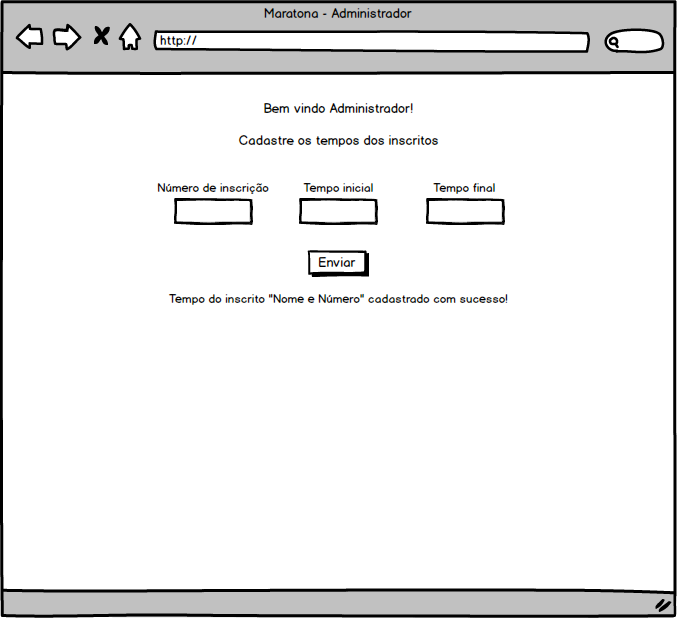


Figura 6 - Administrador

## fluxo de telas

Para facilitar o entendimento de como o sistema funciona, mostra-se abaixo o fluxo de telas. Foi utilizado a ferramenta online Draw.io para o desenvolvimento do diagrama, o qual mostra como as telas disponíveis no sistema interagem, ou seja, quais caminhos o usuário ou administrador deve seguir para desempenhar seus objetivos com o sistema.



Figura - FLUXO DE TELAS

## conclusão

Fazer a prototipação de tela mostra-se importante para fazer a modelagem de dados e o banco de dados, pois facilita o desenvolvimento e evita ambiguidades que podem ser geradas pela subjetividade do desenvolvedor. Já o fluxo de telas tem como principal objetivo a visualização da interação que o usuário tem na aplicação, isso permite que o fluxo completo seja visualizado de uma maneira simples, assim permitindo que seja intuitivo para quem acessa.

# modelagem de dados

Após a prototipação das telas, é feito a modelagem de dados, que compreende quais dados são necessários para o funcionamento do sistema e quais dados são gerados a partir dele.

Para ser possível o desenvolvimento do projeto, utilizamos o software DB Manager para a construção do Modelo Entidade-Relacionamento, o qual é um modelo de dados que descreve as entidades envolvidas com suas características e como elas se relacionam, em um aspecto geral, a modelagem de dados representa a estrutura que o banco de dados terá na aplicação.

## Modelo entidade-Relacionamento

O Modelo Entidade-Relacionamento (FIGURA 6) caracteriza-se por mostrar de forma simplificada e direta de como as entidades e seus atributos relacionam-se dentro do sistema. Para o desenvolvimento do diagrama foi utilizado o software DBdesign e no modelo apresenta-se todas as tabelas do banco de dados do sistema e como elas se relacionam.

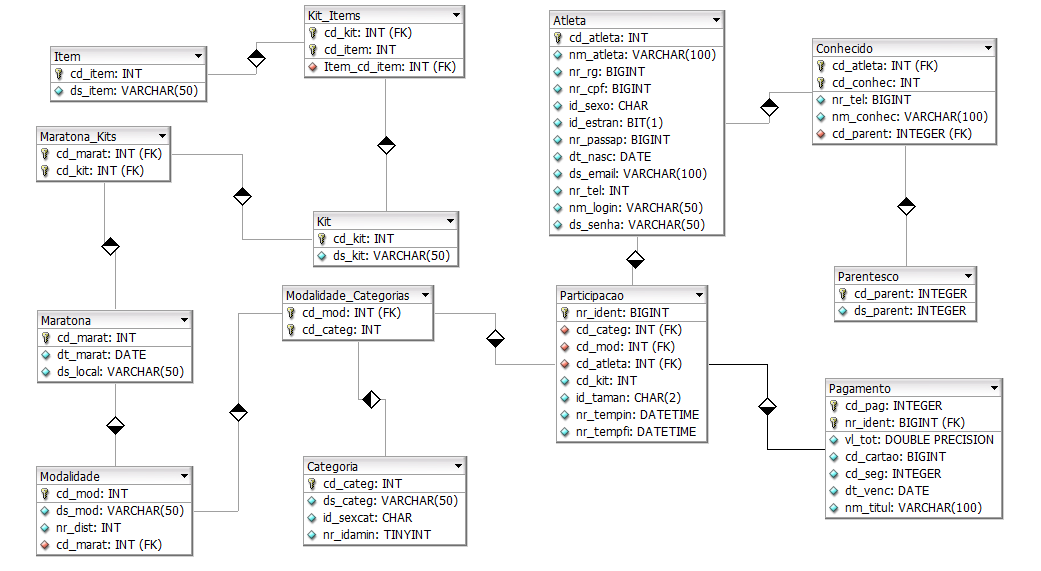
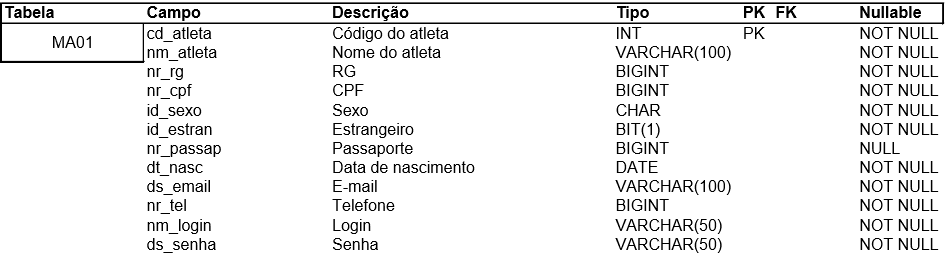


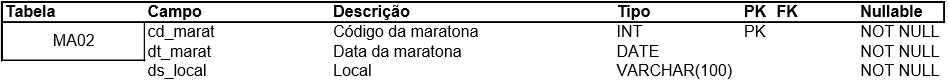
Figura - Modelo entidade-relacionamento

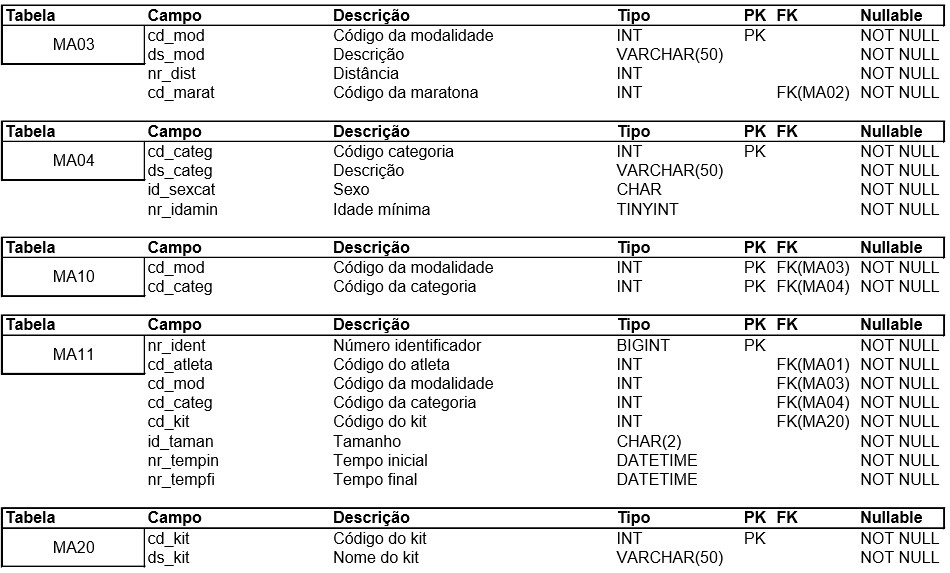
## dicionário de dados

O dicionário de dados é responsável por armazenar as informações de relacionamento e chaves primárias das tabelas. Contém todas as descrições necessárias para que o DBA possa criar os scripts de banco de dados corretamente.

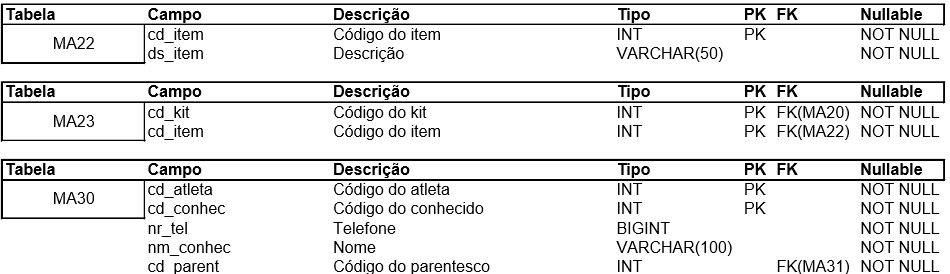














## Conclusão

O Modelo Entidade-Relacionamento facilita o desenvolvimento de modo a apresentar em detalhes cada campo da tabela como o nome, a descrição e tipo dos atributos. Com isso facilita a geração dos scripts para o banco de dados e auxilia o desenvolvedor na integração da aplicação com a base de dados. Já o dicionário de dados facilita a visualização e entendimento da relação de atributos presentes em cada tabela.

# script banco de dados

Scripts de banco de dados são arquivos que contêm instruções em linguagem SQL que manipulam os dados contidos no sistema. Os scripts abaixo foram criados com base na modelagem de dados que geraram o Diagrama Entidade-Relacionamento, e, a partir dele, foi gerado o modelo físico através do software BRModelo, sendo feito apenas algumas pequenas alterações para o seu perfeito funcionamento no MySQL.

## SCRIPT PARA criação das TABELAS

/\*============================================================\*/

/\* Script de criação das tabelas

/\* Banco de dados: MySQL

/\* Versão 1

/\*============================================================\*/

CREATE TABLE Atleta (

cd\_atleta INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

nm\_atleta VARCHAR(100) NOT NULL ,

nr\_rg BIGINT NOT NULL ,

nr\_cpf BIGINT NOT NULL ,

id\_sexo CHAR NOT NULL ,

id\_estran BIT(1) NOT NULL ,

nr\_passap BIGINT NULL ,

dt\_nasc DATE NOT NULL ,

ds\_email VARCHAR(100) NOT NULL ,

nr\_tel BIGINT NOT NULL ,

nm\_login VARCHAR(50) NOT NULL ,

ds\_senha VARCHAR(50) NOT NULL

);

CREATE TABLE Maratona (

cd\_marat INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

dt\_marat DATE NOT NULL ,

ds\_local VARCHAR(100) NOT NULL

);

CREATE TABLE Modalidade (

cd\_mod INT NOT NULL PRIMARY KEY,

ds\_mod VARCHAR(50) NOT NULL ,

nr\_dist INT NOT NULL ,

cd\_marat INT NOT NULL ,

FOREIGN KEY (cd\_marat) REFERENCES Maratona (cd\_marat) ON DELETE RESTRICT

);

CREATE TABLE Categoria (

cd\_categ INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

ds\_categ VARCHAR(50) NOT NULL ,

id\_sexcat CHAR NOT NULL ,

nr\_idamin TINYINT NOT NULL

);

CREATE TABLE Modalidade\_Categorias (

cd\_mod INT NOT NULL ,

cd\_categ INT NOT NULL ,

PRIMARY KEY (cd\_mod, cd\_categ) ,

FOREIGN KEY (cd\_mod ) REFERENCES Modalidade (cd\_mod ) ON DELETE RESTRICT,

FOREIGN KEY (cd\_categ) REFERENCES Categoria (cd\_categ) ON DELETE RESTRICT

);

CREATE TABLE Kit (

cd\_kit INT NOT NULL PRIMARY KEY,

ds\_kit VARCHAR(50) NOT NULL

);

CREATE TABLE Participacao (

nr\_ident BIGINT NOT NULL PRIMARY KEY,

cd\_atleta INT NOT NULL ,

cd\_mod INT NOT NULL ,

cd\_categ INT NOT NULL ,

cd\_kit INT NOT NULL ,

id\_taman CHAR(2) NOT NULL ,

nr\_tempin DATETIME NOT NULL ,

nr\_tempfi DATETIME NOT NULL ,

FOREIGN KEY (cd\_atleta) REFERENCES Atleta (cd\_atleta) ON DELETE RESTRICT,

FOREIGN KEY (cd\_mod ) REFERENCES Modalidade(cd\_mod) ON DELETE RESTRICT,

FOREIGN KEY (cd\_categ ) REFERENCES Categoria (cd\_categ) ON DELETE RESTRICT,

FOREIGN KEY (cd\_kit ) REFERENCES Kit (cd\_kit) ON DELETE RESTRICT

);

CREATE TABLE Maratona\_Kits (

cd\_marat INT NOT NULL ,

cd\_kit INT NOT NULL ,

PRIMARY KEY (cd\_marat, cd\_kit) ,

FOREIGN KEY (cd\_marat) REFERENCES Maratona(cd\_marat) ON DELETE RESTRICT,

FOREIGN KEY (cd\_kit ) REFERENCES Kit (cd\_kit) ON DELETE RESTRICT

);

CREATE TABLE Item (

cd\_item INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

ds\_item VARCHAR(50) NOT NULL

);

CREATE TABLE Kit\_Items (

cd\_kit INT NOT NULL ,

cd\_item INT NOT NULL ,

PRIMARY KEY (cd\_kit, cd\_item) ,

FOREIGN KEY (cd\_kit ) REFERENCES Kit (cd\_kit) ON DELETE RESTRICT,

FOREIGN KEY (cd\_item) REFERENCES Item(cd\_item) ON DELETE RESTRICT

);

CREATE TABLE Parentesco (

cd\_parent INT NOT NULL PRIMARY KEY AUTO\_INCREMENT,

ds\_parent VARCHAR(50) NOT NULL

);

CREATE TABLE Conhecido (

cd\_atleta INT NOT NULL ,

cd\_conhec INT NOT NULL ,

nr\_tel BIGINT NOT NULL ,

nm\_conhec VARCHAR(100) NOT NULL ,

cd\_parent INT NOT NULL ,

PRIMARY KEY (cd\_atleta, cd\_conhec) ,

FOREIGN KEY (cd\_parent) REFERENCES Parentesco(cd\_parent) ON DELETE RESTRICT

);

## script para inSERÇÃO DE DADOS

/\*=========================================================\*/

/\* parentesco \*/

/\*=========================================================\*/

INSERT INTO parentesco (cd\_parent, ds\_parent) VALUES (1, 'Pai');

INSERT INTO parentesco (cd\_parent, ds\_parent) VALUES (2, 'MÃ£e');

INSERT INTO parentesco (cd\_parent, ds\_parent) VALUES (3, 'Tio(a)');

INSERT INTO parentesco (cd\_parent, ds\_parent) VALUES (4, 'AvÃ´(Ã³)');

INSERT INTO parentesco (cd\_parent, ds\_parent) VALUES (5, 'Namorado(a)');

INSERT INTO parentesco (cd\_parent, ds\_parent) VALUES (6, 'Outro');

/\*=========================================================\*/

/\* Item \*/

/\*=========================================================\*/

INSERT INTO item (cd\_item, ds\_item) VALUES (1, "Camiseta");

INSERT INTO item (cd\_item, ds\_item) VALUES (2, "Sacola de treino");

INSERT INTO item (cd\_item, ds\_item) VALUES (3, "Medalha de participaÃ§Ã£o");

INSERT INTO item (cd\_item, ds\_item) VALUES (4, "Jaqueta esportiva");

INSERT INTO item (cd\_item, ds\_item) VALUES (5, "Cinto de hidrataÃ§Ã£o");

INSERT INTO item (cd\_item, ds\_item) VALUES (6, "BonÃ©");

/\*=========================================================\*/

/\* kit \*/

/\*=========================================================\*/

INSERT INTO kit (cd\_kit, ds\_kit) VALUES(1, "Kit VIP");

INSERT INTO kit (cd\_kit, ds\_kit) VALUES(2, "Kit Plus");

INSERT INTO kit (cd\_kit, ds\_kit) VALUES(3, "Kit BÃ¡sico");

/\*=========================================================\*/

/\* kit\_itens \*/

/\*=========================================================\*/

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (1, 1);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (1, 2);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (1, 3);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (1, 4);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (2, 1);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (2, 2);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (2, 3);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (2, 5);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (2, 6);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (3, 1);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (3, 2);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (3, 3);

INSERT INTO kit\_items(cd\_kit, cd\_item) VALUES (3, 6);

/\*=========================================================\*/

/\* maratona \*/

/\*=========================================================\*/

INSERT INTO maratona(cd\_marat, dt\_marat, ds\_local) VALUES (1, STR\_TO\_DATE('15-11-2015','%d-%m-%Y'), 'PraÃ§a OsÃ³rio, Curitiba');

/\*=========================================================\*/

/\* maratona\_kits \*/

/\*=========================================================\*/

INSERT INTO maratona\_kits(cd\_marat, cd\_kit) VALUES (1, 1);

INSERT INTO maratona\_kits(cd\_marat, cd\_kit) VALUES (1, 2);

INSERT INTO maratona\_kits(cd\_marat, cd\_kit) VALUES (1, 3);

/\*=========================================================\*/

/\* modalidade \*/

/\*=========================================================\*/

INSERT INTO modalidade(cd\_mod, ds\_mod, nr\_dist, cd\_marat) VALUES (1, "Iniciante" , 10, 1);

INSERT INTO modalidade(cd\_mod, ds\_mod, nr\_dist, cd\_marat) VALUES (2, "IntermediÃ¡rio", 21, 1);

INSERT INTO modalidade(cd\_mod, ds\_mod, nr\_dist, cd\_marat) VALUES (3, "AvanÃ§ado" , 42, 1);

## Conclusão

O script do banco de dados é a concretização da modelagem das tabelas, ele é aplicado no sistema gerenciador de banco de dados MySQL e utilizando o padrão de consulta a modelos estruturados, ou seja, SQL. Os scripts desenvolvidos apresentam a criação das tabelas, bem como o relacionamento entre elas e as chaves primárias, também é apresentado a inserção da massa de dados inicial para o bom funcionamento do sistema.

# DIAGRAMA de FLUXO DE DADOS

O Diagrama de Fluxo de Dados serve para mostrar como os dados são trafegados pelo sistema, de maneira bem rasa como é mostrado no Diagrama de Contexto e de maneira mais específica no DFD de nível 0.

## Diagrama de contexto

Na FIGURA 7, aparecem duas entidades externas, uma delas é o Usuário que vai enviar os dados pessoais ao sistema de maratona, e o sistema retorna o número de inscrição. Já a outra entidade externa, chama de Admin, vai entrar o tempo dos corredores e o sistema retorna o resultado da maratona.



Figura – diagrama de contexto

## DFD NÍVEL 0

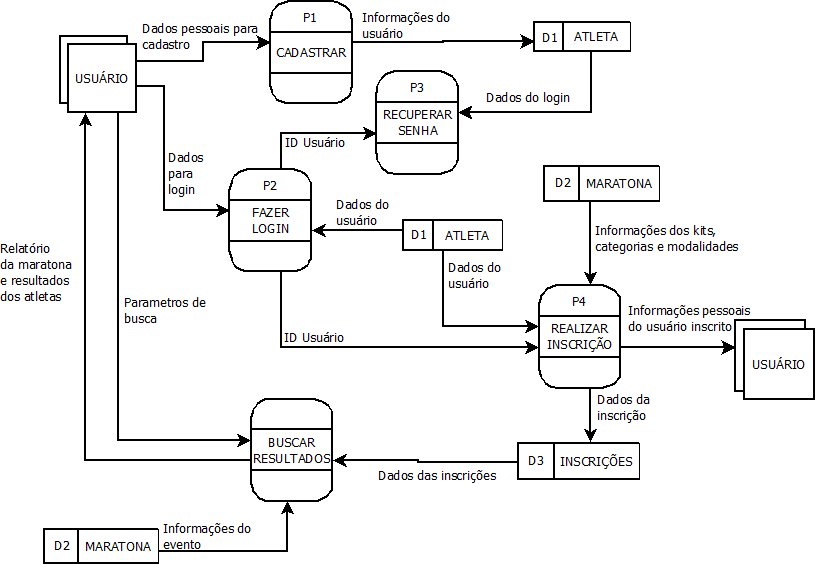
No Diagrama de Fluxo de Dados no nível 0 (FIGURA 8) mostrando em um contexto geral como o usuário interage com o sistema, passando informações, como por exemplo, seus dados pessoais, e o sistema retorna informações também, como o número de inscrição. 

Figura - dFD NÍVEL 0

## conclusão

Com o diagrama conceitual se tem uma ampla visão de como é a interação do sistema com o usuário, ou seja, quais dados o usuário precisa inserir e o que é retornado a ele após a execução do sistema. O diagrama de fluxo de dados mostra quais e como os dados percorrem pelas diversas etapas do sistema, nele podemos ter uma visão mais detalhada da entrada e saída de dados e dos processos envolvidos em cada etapa.

# fonte

Arial 12

# conclusão