

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO
GRANDE DO NORTE
CAMPUS AVANÇADO LAJES
CURSO TÉCNICO INTEGRADO EM INFORMÁTICA

DÉBORA LAVÍNIA DA SILVA MELO
MARIA RITA FAUSTINO PAIXÃO

***GAME OF CODES: Especificação de banco de dados de um software
pedagógico de introdução à lógica de programação***

LAJES/RN

2021

DÉBORA LAVÍNIA DA SILVA MELO
MARIA RITA FAUSTINO PAIXÃO

***GAME OF CODES: Especificação de banco de dados de um software
pedagógico de introdução à lógica de programação***

Relatório de Prática Profissional apresentado ao Curso Técnico Integrado em Informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Campus Avançado Lajes, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico em Informática.

Orientador (a): Prof. Me. Dannilo Martins Cunha.

LAJES/RN

2021

AGRADECIMENTOS

Agradecemos a todos que nos ajudaram nesse percurso.

RESUMO

O presente trabalho de conclusão de curso foi elaborado pensando na situação atual em que a tecnologia e área da informática são bastante requisitados, principalmente no mercado de trabalho, sendo assim, o conhecimento sobre essa área se torna importante para todos, inclusive para as crianças que podem absorver melhor as informações básicas desde cedo. A partir desse cenário foi colocado como objetivo a montagem de uma base de dados para dar sustentação a um *software* que introduzisse um conhecimento básico sobre programação ao público juvenil de maneira lúdica e assim o jogo intitulado *Game Of Codes* foi planejado. Para a construção do banco de dados foi usado um método constituído por pesquisas acerca das ferramentas precisas para manusear a linguagem SQL e elaborar diagramas e tabelas com êxito. Com essa metodologia, o banco de dados foi elaborado de forma consistente, organizada e interativa em toda sua estrutura, sendo assim, facilitando a disseminação do conhecimento na área da programação para o público infantil .

Palavras-chave: Banco de dados. Banco de dados relacional. Modelagem de dados. Introdução à programação.

LISTA DE ILUSTRAÇÕES

Figura 1 – Modelo Conceitual do <i>Game of Codes</i>	18
Figura 2 – Modelo Lógico do <i>Game of Codes</i>	20
Figura 3 – Criação da tabela Jogadores	21
Figura 4 – Criação da tabela Jogos	22
Figura 5 – Criação da tabela Módulos	22
Figura 6 – Criação da tabela Fases	23
Figura 7 – Criação da tabela Quetões	23
Figura 8 – Criação da tabela Alternativas	24
Figura 9 – Criação da tabela JogosModulos	25

LISTA DE TABELAS

Tabela 1 – Síntese das Atividades do Aluno no Projeto

11

LISTA DE ABREVIATURAS E SIGLAS

BD	Banco de Dados
SQL	<i>Structured Query Language</i>
SGBD	Sistema de Gerenciamento de Banco de Dados
DER	Diagrama de Entidade e Relacionamento
ID	<i>Identity</i>

Sumário

1. INTRODUÇÃO	8
1.1 CONTEXTUALIZAÇÃO	8
1.2 PROBLEMÁTICA	8
1.3 OBJETIVOS	8
1.4 JUSTIFICATIVA	9
1.5 ESTRUTURA DO TRABALHO	9
2. DADOS GERAIS DO TRABALHO DE CONCLUSÃO DO CURSO	10
2.1 SÍNTESE DAS ATIVIDADES DO ALUNO NO PROJETO	10
3. METODOLOGIA	12
4. FUNDAMENTAÇÃO TEÓRICA	13
4.1 BANCO DE DADOS RELACIONAL	13
4.1.1 Tabela	13
4.2 MODELO CONCEITUAL	13
4.2.1 Diagrama de Entidade e Relacionamento	14
4.2.2 Entidade	14
4.2.3 Atributos	14
4.2.4 Relacionamentos	14
4.2.5 Cardinalidade	14
4.3 MODELO LÓGICO	15
4.4 MODELO FÍSICO	15
4.5 SQL	15
4.5.1 Create Table	15
4.5.2 Tipo de texto e numérico	16
5. CARACTERIZAÇÃO DAS ATIVIDADES DESENVOLVIDAS	17
5.1 MODELO CONCEITUAL	17
5.2 MODELO LÓGICO	19
5.3 MODELO FÍSICO	20
6. ANÁLISE E DISCUSSÃO DOS RESULTADOS	25
7. CONSIDERAÇÕES FINAIS	27
REFERÊNCIAS	28
ANEXO A – Formulário de identificação	29

1. INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

A programação vem ganhando gradativamente mais espaço na atualidade em vários setores, se tornando cada vez mais indispensável ter conhecimento sobre esse tema. Sendo o algoritmo a base da programação, é importante que seja feita a propagação para as crianças, tornando-as familiarizadas desde cedo com o assunto.

O *software Game Of Codes* foi pensado para auxiliar o acesso a informações sobre conceitos básicos da programação para o público citado, de forma lúdica, interativa e facilitada, a fim de que haja o desenvolvimento do interesse sobre os fundamentos da lógica da programação.

1.2 PROBLEMÁTICA

Alguns alunos apresentam dificuldades na programação, justamente, por não ter compreendido com eficácia a base, que é formada pela lógica de algoritmos. Essa problemática pode se agravar ao decorrer dos anos, tendo em vista que é preciso dominar a lógica inicial para conseguir compreender outros níveis posteriores.

Diante disso, é necessário que seja criado algo para tratar esse problema de forma preventiva, fortalecendo e focando na base da programação como algo prioritário para todo o resto do processo.

1.3 OBJETIVOS

A partir de todo esse contexto, foi pensado e elaborado a estrutura inicial de um *software* que atendesse ao que o projeto almeja, sendo assim, foi construído a base de dados para dar sustentação ao jogo intitulado *Game Of Codes*. O objetivo foi feito através da linguagem SQL, tendo como base conceitos do Sistema de Gerência de Banco de Dados.

1.4 JUSTIFICATIVA

Tendo em vista o grande avanço na área da programação, é necessário que as crianças tenham familiaridade desde cedo com esse assunto, sendo primordial o enfoque na base, que é os Fundamentos de Lógica de Programação. Isto consiste nos ensinamentos dos Algoritmos, começando com implementações de sequências lógicas de comandos.

Esse foi o motivo pelo qual foi escolhido trabalhar a lógica de programação de forma lúdica e interativa, fazendo com que seja estimulado o interesse desde a infância, fortalecendo a base e conseqüentemente facilitando futuramente o melhor domínio do assunto.

1.5 ESTRUTURA DO TRABALHO

Esse projeto é composto por sete capítulos, o capítulo um é a introdução, a qual traz informações gerais sobre o trabalho como um todo. Logo após, vem o capítulo dois que se trata dos dados gerais do trabalho, onde é feito detalhadamente uma descrição das atividades executadas. O capítulo três é a metodologia, onde é descrito de forma geral as atividades feitas ao longo do trabalho e o capítulo quatro é a fundamentação teórica que apresenta a base de pesquisa feita para a construção do projeto, e posteriormente, o capítulo cinco se trata da caracterização das atividades desenvolvidas com observações sobre tais realizações. A seguir vem o capítulo seis que é a análise e discussão dos resultados e o capítulo sete se trata das considerações finais que abordam uma visão crítica sobre o projeto completo.

2. DADOS GERAIS DO TRABALHO DE CONCLUSÃO DO CURSO

Título do projeto: *Game Of Codes*

Período de realização: de 01/10/2020 a 26/07/2021

Total de horas: 350 horas.

Orientador: Prof. Me. Dannilo Martins Cunha

2.1 SÍNTESE DAS ATIVIDADES DO ALUNO NO PROJETO

Quadro 1 – Síntese das Atividades do Aluno no Projeto.

CARGA HORÁRIA	ATIVIDADES DESENVOLVIDAS	RESULTADOS ALCANÇADOS
40h	Estudo para relembrar conceitos do Modelo Conceitual	Conhecimento para elaborar o modelo conceitual
40h	Estudo para relembrar conceitos do Modelo Lógico	Conhecimento para elaborar o modelo lógico
40h	Estudo para relembrar conceitos do Modelo Físico e da linguagem de manipulação SQL	Conhecimento para elaborar o modelo conceitual
80h	Definição de entidades; definição dos relacionamentos; definição das cardinalidades; definição de atributos; criação do diagrama.	Modelo Conceitual
60h	Definição das chaves primárias; definição das chaves estrangeiras; definição das relações; definição dos atributos dessas relações; criação do diagrama.	Modelo Lógico

15h	Análise, implementação e testes da tabela Jogadores utilizando a linguagem de manipulação SQL.	Implementação da tabela Jogadores
15h	Análise, implementação e testes da tabela Jogos utilizando a linguagem de manipulação SQL.	Implementação da tabela Jogos
15h	Análise, implementação e testes da tabela Módulos utilizando a linguagem de manipulação SQL.	Implementação da tabela Módulos
15h	Análise, implementação e testes da tabela Fases utilizando a linguagem de manipulação SQL.	Implementação da tabela Fases
15h	Análise, implementação e testes da tabela Questões utilizando a linguagem de manipulação SQL.	Implementação da tabela Questões
15h	Análise, implementação e testes da tabela Alternativas utilizando a linguagem de manipulação SQL.	Implementação da tabela Alternativas

Fonte: autoria própria (2021)

3. METODOLOGIA

Neste capítulo apresentaremos as técnicas e os procedimentos que foram utilizados na construção do então projeto, em ordem cronológica, pontuando as principais atividades realizadas.

Foram efetuadas pesquisas através de livros, escritos por especialistas formados na área, sobre a modelagem do banco de dados, especificamente acerca do modelo conceitual, modelo lógico e modelo físico. Para a construção do modelo conceitual usamos a ferramenta Lucidchart, que se trata de um programa que disponibiliza ferramentas para a criação de diagramas, e assim, foi feita análise das partes como um todo.

Já para a elaboração do modelo lógico, para a criação das tabelas, foi usada a ferramenta de criação Paint, onde foram definidas as chaves primária e chaves estrangeiras e suas características são advindas do modelo conceitual. Para o modelo físico, usamos a linguagem SQL, dentro das definições do Sistema de Gerenciamento de Banco de Dados (SGBD), contendo as tabelas, colunas e suas outras funcionalidades.

4. FUNDAMENTAÇÃO TEÓRICA

4.1 BANCO DE DADOS RELACIONAL

Segundo Abrahan Silberschatz (2006) um banco de dados é uma coleção que contém informações importantes, a fim de gerenciá-las, construir sua estrutura para realizar o armazenamento e tornando possível a manipulação desses dados.

Os conceitos de banco de dados relacional e tabelas são definidos a seguir:

Um banco de dados relacional consiste em uma coleção de tabelas, cada uma com um nome unico atribuído. Uma linha em uma tabela representa uma relação entre um conjunto de valores. Informalmente, uma tabela é um conjunto de entidades, e uma linha é uma entidade. (Silberschatz, 2006, p. 25).

Um SGBD é uma sigla referente a Sistema de Gerência de Banco de Dados, é um *software* que permite realizar modificações, recuperações e define os dados pertencentes à base de dados, conforme Carlos Alberto Heuser (2009).

4.1.1 Tabela

As tabelas que são constituídas por linha, que tem como característica valores dos dados relacionais, a nomeação das colunas e tabelas são essenciais para entender o significado dos valores, e os atributos que compõem as entidades, que são caracterizações para a descrição desta entidade, que é uma representação de algo real, segundo Elmasri e Navathe (2011).

4.2 MODELO CONCEITUAL

De acordo com Vinicius Carvalho (2015), o modelo conceitual não traz especificações provindas da implementação do banco, ou seja, é independente da tecnologia da linguagem implementada. Segundo Rosaura Espírito Santo da Silva (2012), esse modelo não mostra como os dados são armazenados e ele é caracterizado através do diagrama entidade-relacionamento.

4.2.1 Diagrama de Entidade e Relacionamento

Esse diagrama é aplicado na construção de *softwares* que efetuam o manuseio dos dados presente no banco, o Diagrama de Entidade e Relacionamento (DER), tem como finalidade melhorar o entendimento do projeto para o usuário. Detalhes como a forma de armazenamento, estão ausentes, conforme Paulo Henrique Cayres (2015).

4.2.2 Entidade

De acordo com Rosaura Espírito Santo da Silva (2012), uma entidade faz a representação de algo real que é aplicado no banco de dados.

4.2.3 Atributos

Já um atributo ajuda na compreensão de uma entidade, que se liga a alguma propriedade, segundo Rosaura Espírito Santo da Silva (2012).

4.2.4 Relacionamentos

Um relacionamento pode ser descrito por Rosaura Espírito Santo da Silva (2012), como um aglomerado de ligações entre as entidades, que é representado por um losango, o qual faz a conexão entre as com linhas aos retângulos, onde estão contidas as entidades.

4.2.5 Cardinalidade

Cardinalidade define a quantidade de cada relacionamento presentes, existe a cardinalidade máxima que mostra a quantidade de instâncias, representada pelo 1 ou N e a mínima significa uma dependência de existência, segundo Paulo Henrique Cayres (2015).

4.3 MODELO LÓGICO

Segundo Rosaura Espírito Santo da Silva (2012), esse modelo lógico faz a exposição das informações acerca dos negócios e não da tecnologia, sem depender do modelo físico. Para Carlos Alberto Heuser (2009), o modelo lógico traz uma descrição detalhada da estrutura do banco a partir da visão do usuário, contendo informações sobre o armazenamento interno.

4.4 MODELO FÍSICO

O modelo físico é descrito a seguir:

Nessa fase, o modelo do banco de dados é enriquecido com detalhes que influenciam no desempenho do banco de dados, mas não interferem na sua funcionalidade. Script do banco de dados em SQL representa os detalhes dos dados internamente ao BD (campo, tipo/domínio, restrições). (Paulo Henrique Cayres, 2015, p.11)

4.5 SQL

Conforme Elmasri e Navathe (2011) “A SQL usa os termos tabela, linha e coluna para os termos do modelo formal relação, tupla e atributo, respectivamente.”. A linguagem SQL, usada na construção do *software*, é utilizada para modificar os dados no banco, construir a estrutura e aplicar restrições de segurança, de acordo com Abraham Silberschatz (2006).

4.5.1 *Create Table*

O comando `create table` nomeia uma nova relação com os atributos, atribuindo nome, valores, tipo de dado e outras restrições, conforme Elmasri e Navathe (2011).

4.5.2 Tipo de texto e numérico

Os tipos de texto são implementados na construção do banco de dados com implementação do SGBD, e definem a caracterização dos dados armazenados. Alguns desses tipos foram utilizados na construção do projeto, como mostra a seguir:

CHAR(tamanho) : guarda um número fixo de caracteres. Pode conter letras, números e caracteres especiais. O tamanho deve ser declarado entre parênteses. Guarda até 255 caracteres. VARCHAR(tamanho): possui as características do tipo CHAR, com a diferença de que, se você criar com mais de 255 caracteres, ele se transforma no tipo TEXT. Ou seja, se for criar algum campo com mais de 255, já crie como TEXT. (Carvalho, 2015, p.26)

Tipos numéricos também são importantes para armazenar informações que contenham números e é definido por Carvalho (2015, p.27), “int (tamanho): guarda números inteiros. Suporta de -2147483648 até 2147483647 caracteres. O número máximo de caracteres pode ser especificado entre parênteses.”.

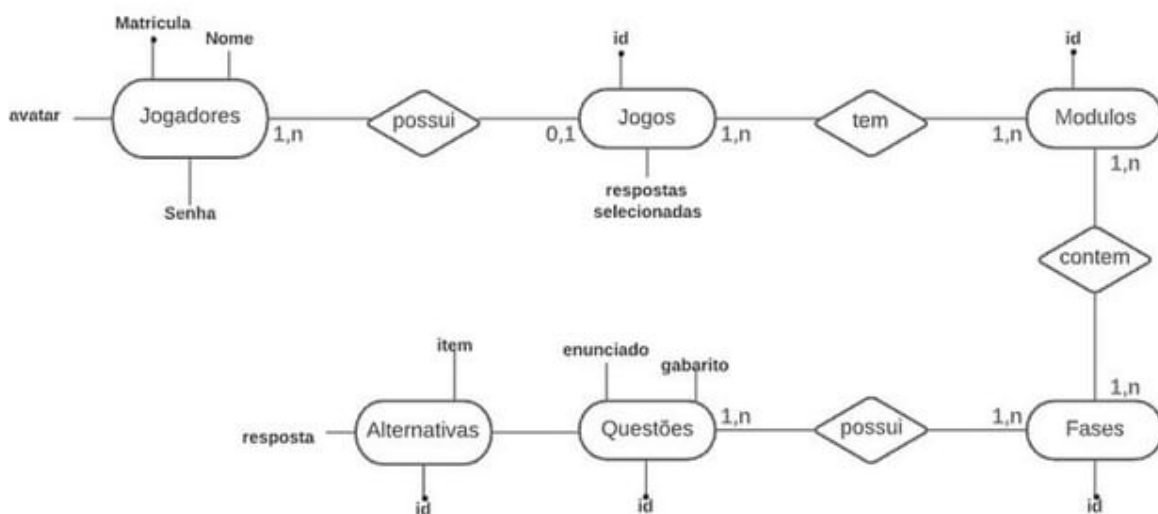
5. CARACTERIZAÇÃO DAS ATIVIDADES DESENVOLVIDAS

5.1 MODELO CONCEITUAL

O modelo conceitual tem como objetivo apresentar os relacionamentos que ocorrem entre as entidades estabelecidas através da análise dos requisitos, onde é mostrado um sistema de objetos, relações e propriedades que servem para situar o desenvolvedor de forma clara, pois ajuda no entendimento do protótipo.

Por fim, tem como função descrever o comportamento e sua finalidade, como é apresentado no modelo conceitual na figura 1; o qual descreve as entidades, suas relações e atributos que fazem parte do nosso projeto.

Figura 1 – Modelo Conceitual do *Game of Codes*.



Fonte: Autoral (2021).

A entidade Jogadores representa os jogadores que vão utilizar o *software* proposto no trabalho, essa entidade é constituída por 4 atributos: matrícula, nome, avatar e senha. Sendo matrícula a identificação de cada jogador, nome o nome do jogador, avatar o *link* da imagem escolhida pelo jogador para ser sua representação

no jogo, e senha a palavra passe escolhida para fazer o *login* em sua conta. A entidade jogadores possui relacionamento com a entidade jogo, podendo ter 1 ou n jogos.

A entidade Jogos representa a partida em que o jogador está, essa entidade é constituída por dois atributos: id, que é a identificação; e as respostas selecionadas que formam uma lista das alternativas selecionadas pelo jogador ao decorrer das partidas. A entidade jogos possui relacionamento com as entidades jogadores e módulos, jogos podem ter 0 ou 1 jogadores e 1 ou n módulos.

A entidade Módulos representa os módulos que serão jogados pelos jogadores, essa entidade é constituída pelo atributo id, que representa a identificação de cada módulo. A entidade possui relacionamento com jogos e fases, no qual módulos tem 1 ou n jogos, e módulos tem 1 ou n fases.

A entidade Fases representa as fases que formam os módulos, essa entidade é constituída pelo atributo id, que é a identificação de cada fase dentro dos módulos. A entidade possui relacionamento com módulo e questões, módulos tem 1 a n fases, e fases tem um a n questões.

A entidade Questões representa as questões que vão formar as fases, essa entidade é constituída por 3 atributos: id, enunciado, gabarito. Sendo id a identificação de cada questão, enunciado o texto referente a questão e gabarito a alternativa correta da questão. A entidade questões possui relacionamento com as entidades fases e alternativas, podendo fazer parte de 1 ou n fases e ter de 1 a n alternativas.

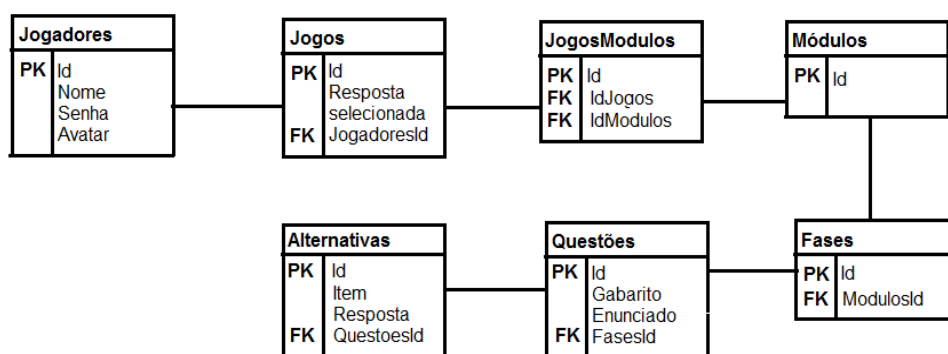
A entidade Alternativas representa as alternativas que formam as questões, essa entidade é constituída por 3 atributos: id, item, e resposta. Sendo id a identificação de cada alternativa, item o texto de cada alternativa, e resposta a resposta escolhida pelo jogador. A entidade alternativas possui relacionamento com a entidade fases, no qual fases podem ter de 1 a n alternativas.

5.2 MODELO LÓGICO

No modelo lógico, é exposto chaves primárias e estrangeiras, nomenclaturas e integridade referencial. Esse modelo, é criado a partir dos dados contidos no modelo conceitual.

Na figura a seguir, contém o modelo lógico do nosso projeto, onde são identificadas as chaves de cada entidade e suas outras características.

Figura 2 – Modelo Lógico do *Game of Codes*.



Fonte: Autoral (2021).

A tabela Jogadores apresenta 4 atributos (matrícula, nome, senha e avatar) como explicado acima, no tópico 5.1, tendo como chave primária a matrícula, essa tabela não possui chave estrangeira.

A tabela Jogos apresenta 3 atributos (id, resposta selecionada e jogadoresId), tendo id como chave primária e jogadoresId como chave estrangeira referente ao atributo matricula na tabela Jogadores.

A tabela JogosModulos é uma tabela auxiliar e apresenta 3 atributos (id, idJogos e idModulos), sendo id como chave primária, idJogos como chave estrangeira referente ao atributo id na tabela Jogos e idModulos como chave estrangeira referente ao atributo id na tabela Módulos.

A tabela Modulos apresenta 1 atributo (id), sendo id a chave primária, essa

tabela não possui chave estrangeira.

A tabela Fases apresenta 2 atributos (id, modulosId), tendo como id a chave primária e modulosId como chave estrangeira referente ao atributo id na tabela Módulos.

A tabela Questões apresenta 4 atributos (id, gabarito, enunciado e fasesId), tendo id como chave primária e fasesId como chave estrangeira referente ao atributo id na tabela Fases.

A tabela Alternativas apresenta 4 atributos (id, item, respostas e questoesId), tendo id como chave primária e questoesId como chave estrangeira referente ao atributo id na tabela Questoes.

5.3 MODELO FÍSICO

Esse modelo físico, apresenta o design do banco de dados, que depende do modelo lógico. Contendo tabelas, colunas, chave primária e estrangeira, podendo existir algumas restrições e além disso, deve ser consideradas as limitações colocadas sobre o SGBD (Sistema de Gerenciamento de Banco de Dados) escolhido.

No modelo físico apresentado nas próximas figuras, mostra a criação de tabelas e colunas do banco de dados do nosso projeto:

Figura 3 – Criação da Tabela Jogadores

```
CREATE TABLE Jogadores(  
  id int AUTO_INCREMENT,  
  nome varchar(255),  
  avatar varchar(255),  
  senha varchar(15),  
  PRIMARY KEY (id)  
);
```

Fonte: Autoral (2021).

Em SQL, foi criada a tabela Jogadores através do comando *create table*, ela tem como colunas: id, nome, avatar e senha. Sendo id do tipo inteiro e usando o recurso *auto_increment*; nome e avatar do tipo varchar permitindo até 255 caracteres, e senha varchar com até 15 caracteres. Por fim, o comando *primary key* indica que a chave primária é a coluna id.

Figura 4 – Criação da Tabela Jogos

```
CREATE TABLE Jogos(  
  id int AUTO_INCREMENT,  
  RespostaSelecionada varchar(255),  
  jogadoresId int,  
  PRIMARY KEY (id),  
  FOREIGN KEY (jogadoresId) REFERENCES Jogadores(id)  
);
```

Fonte: Autoral (2021).

Em SQL, foi criada a tabela Jogos através do comando *create table*, ela tem como colunas: id, RespostaSelecionada, e jogadoresId. Sendo id do tipo inteiro e usando o recurso *auto_increment*; RespostaSelecionada do tipo varchar permitindo até 255 caracteres e jogadoresId também do tipo inteiro. Por fim, o comando *primary key* indica que a chave primária é a coluna id, e o *foreign key* indica que a coluna jogadoresId referencia a coluna id na tabela Jogadores.

Figura 5 – Criação da Tabela Módulos

```
CREATE TABLE Modulos (  
  id int AUTO_INCREMENT,  
  PRIMARY KEY (id)  
);
```

Fonte: Autoral (2021).

Em SQL foi criado a tabela Módulos através do comando *create table* onde tem como coluna o id, que é do tipo inteiro e recebe o recurso de *auto_increment*. Por fim, é usado o comando *primary key* para indicar que a coluna id é a chave primária.

Figura 6 – Criação da Tabela Fases

```
• CREATE TABLE Fases(  
  id int AUTO_INCREMENT,  
  modulosId int,  
  PRIMARY KEY (id),  
  FOREIGN KEY (modulosId) REFERENCES Modulos(id)  
);
```

Fonte: Autoral (2021).

Em SQL, foi criada a tabela Fases através do comando *create table*, ela tem como colunas: id, e modulosId. Sendo as duas colunas do tipo inteiro. A coluna id usa o recurso *auto_increment*. Por fim, o comando *primary key* indica que a chave primária é a coluna id, e o *foreign key* indica que a coluna modulosId referencia a coluna id na tabela Modulos.

Figura 7 – Criação da Tabela Questões

```
• CREATE TABLE Questoes(  
  id int AUTO_INCREMENT,  
  gabarito varchar(255),  
  enunciado text(500),  
  fasesId int,  
  PRIMARY KEY (id),  
  FOREIGN KEY (fasesId) REFERENCES Fases(id)  
);
```

Fonte: Autoral (2021).

Em SQL, foi criada a tabela Questoes através do comando *create table*, ela tem como colunas: id, gabarito, enunciado e fasesId. Sendo id do tipo inteiro e usando o recurso *auto_increment*; gabarito do tipo varchar permitindo até 255 caracteres, enunciado do tipo *text* permitindo textos de até 500 caracteres, e fasesId do tipo inteiro. Por fim, o comando *primary key* indica que a chave primária é a coluna id, e o *foreign key* indica que a coluna fasesId referencia a coluna id na tabela Fases.

Figura 8 – Criação da Tabela Alternativas

```
• CREATE TABLE Alternativas(  
  id int AUTO_INCREMENT,  
  item varchar(255),  
  resposta varchar(255),  
  questoesId int,  
  PRIMARY KEY (id),  
  FOREIGN KEY (questoesId) REFERENCES Questoes(id)  
);
```

Fonte: Autoral (2021).

Em SQL, foi criada a tabela Alternativas através do comando *create table*, ela tem como colunas: id, item, resposta e questoesId. Sendo id do tipo inteiro e usando o recurso *auto_increment*; item e resposta do tipo varchar permitindo até 255 caracteres e questoesId também do tipo inteiro. Por fim, o comando *primary key* indica que a chave primária é a coluna id, e o *foreign key* indica que a coluna questoesId referencia a coluna id na tabela Questoes.

Figura 9– Criação da Tabela JogosModulos

```
• CREATE TABLE JogosModulos(  
  id int AUTO_INCREMENT,  
  idJogos int,  
  idModulos int,  
  PRIMARY KEY (id),  
  FOREIGN KEY (idJogos) REFERENCES Jogos(id),  
  FOREIGN KEY (idModulos) REFERENCES Modulos(id)  
);
```

Fonte: Autorial (2021).

Em SQL, foi criada a tabela JogosModulos através do comando *create table*, ela é uma tabela auxiliar e tem como colunas: id, idJogos, e idModulos. Sendo id do tipo inteiro e usando o recurso *auto_increment*. O comando *primary key* indica que a chave primária é a coluna id, o primeiro *foreign key* indica que a coluna idJogos referencia a coluna id na tabela Jogos, e o segundo *foreign key* indica que a coluna idModulos referencia a coluna id na tabela Modulos

6. ANÁLISE E DISCUSSÃO DOS RESULTADOS

Este capítulo irá analisar e discutir a estrutura do banco de dados do projeto que é composto por tabelas, as quais cada uma tem suas características e funcionalidades distintas, o que as tornam indispensáveis para o projeto.

A tabela Jogadores é primordial para o funcionamento do jogo, pois assim é possível fazer o registro dos perfis cadastrados no banco de dados. Mas, para essa tabela ter melhores resultados outras colunas podem ser futuramente adicionadas, como o email e o telefone do jogador cadastrado, a fim de atingir um melhoramento do projeto.

A tabela Jogos é a essência do *software*, pois sem ela nada pode ser efetuado com êxito. Além dos atributos presentes nesta tabela, outros que podem ser um complemento seriam a duração que o participante obteve para finalizar aquele jogo e a data de início e fim.

A tabela Módulos é precisa para manter a organização, pois ela é indispensável para o controle e manutenção da estrutura do jogo. Outras colunas podem ser acrescentadas para um melhor desenvolvimento, como o nível de dificuldade, temas e vídeo aulas sobre os assuntos tratados.

A tabela Fases também é uma das chaves para manter a organização do *software*, pois ela guarda as fases que estruturam os módulos. Ela pode ser melhorada através da adição de outras colunas como comentários dos jogadores sobre a fase ou dúvidas.

A tabela Questões é necessária para fazer o armazenamento das perguntas em ordem, sem ela, as questões ficariam desordenadas dentro do banco. Podendo ser melhor elaborada, atributos como descrição da solução e dicas para a resolução podem vir a ser novas colunas.

A tabela Alternativas faz um pequeno, porém, não menos importante trabalho dentro do banco de dados, pois ela gerencia as alternativas presentes nas questões. Para ser aperfeiçoada, pode ser agregada uma descrição do erro como novo atributo, onde explicaria o porquê aquela alternativa escolhida estaria equivocada.

A tabela chamada JogosModulos faz se necessária apenas para realizar o relacionamento das tabelas Jogos e Módulos, para haver o funcionamento das duas tabelas com sucesso.

7. CONSIDERAÇÕES FINAIS

O trabalho proposto teve como propósito criar as tabelas, descrevê-las e construir a base de dados do software *Game Of Codes*. A partir disso, foi feito um diagrama do modelo conceitual, apresentando o relacionamento entre as entidades. Também foi realizado o modelo lógico, onde apresenta as chaves primárias e estrangeiras das tabelas. Por fim, o banco de dados foi elaborado.

Com isso, a base de dados obteve uma implementação satisfatória, não apresentando erros no seu funcionamento. É importante que esteja bem organizado para que fique claro como o jogo se estrutura, para que possa haver possíveis novas modificações a fim de obter uma melhora.

REFERÊNCIAS

- CARVALHO, Vinicius. **MySQL**: Comece com o principal banco de dados open source do mercado. São Paulo: Casa do código, 2015.
- CAYRES, Paulo Henrique. **Modelagem de Banco de Dados**. 1 Ed. Rio de Janeiro: Escola Superior de Redes, 2015.
- EIMASRI, Ramez; NAVATHE, Shamkant. **Sistema de banco de dados**. 6 ed. São Paulo: Pearson, 2011.
- HEUSER, Carlos Alberto. **Projeto de banco de dados**. 6 ed. Porto Alegre: Bookman, 2009.
- SILBERSCHARTZ, Abrahan. **Sistema de banco de dados**. 5 ed. Rio de Janeiro: Elsevier, 2006.
- SILVA, Rosaura E. Santo da. **Projeto de Banco de Dados Relacional**. Porto Alegre: Universidade Aberta do Brasil Instituto Federal Sul-rio-grandense, 2012.

ANEXO A – Formulário de identificação

Dados do Relatório Científico	
Título e subtítulo: <i>Game Of Codes</i> : Especificação de banco de dados de um software pedagógico de introdução à lógica de programação	
Tipo de relatório: Prática profissional.	Data: 26/07/2021
Título do projeto/ programa/ plano: <i>Game Of Codes</i> : Especificação de banco de dados de um software pedagógico de introdução à lógica de programação	
Autor(es): Maria Rita Faustino Paixão e Débora Lavínia da Silva Melo	
Instituição e endereço completo: Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte - Campus Avançado Lajes BR-304, Km 120, s/n - Centro, Lajes - RN, 59535-000	
<p>Resumo:</p> <p>O presente trabalho de conclusão de curso foi elaborado pensando na situação atual em que a tecnologia e área da informática são bastante requisitados, principalmente no mercado de trabalho, sendo assim, o conhecimento sobre essa área se torna importante para todos, inclusive para as crianças que podem absorver melhor as informações básicas desde cedo. A partir desse cenário foi colocado como objetivo a montagem de uma base de dados para dar sustentação a um <i>software</i> que introduzisse um conhecimento básico sobre programação ao público juvenil de maneira lúdica e assim o jogo intitulado <i>Game Of Codes</i> foi planejado. Para a construção do banco de dados foi usado um método constituído por pesquisas acerca das ferramentas precisas para manusear a linguagem SQL e elaborar diagramas e tabelas com êxito. Com essa metodologia, o banco de dados foi elaborado de forma consistente, organizada e interativa em toda sua estrutura,</p>	

sendo assim, facilitando a disseminação do conhecimento na área da programação para o público infantil.	
Palavras-chave/descriptores: Banco de dados. Banco de dados relacional. Modelagem de dados. Introdução à programação.	
Nº de páginas: 31	
Jornada de trabalho: 3 horas	Horas semanais: 9 horas
Total de horas: 350 horas.	