

## BANCO DE DADOS

Gestão de conhecimentos em projetos de extensão e equipes  
de competições da Faculdade de Tecnologia

Eric do Vale Castro, 15/0124236

Gabriel Guimarães Almeida de Castro, 15/0126425

Pedro Galieta, 15/0144938

Rafael Augusto Torres, 15/0145365

Brasília, Julho de 2019



**ENGENHARIA  
MECATRÔNICA**  
UNIVERSIDADE DE BRASÍLIA

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>1</b>
1.1	CONTEXTUALIZAÇÃO .....	1
1.2	OBJETIVOS .....	1
1.3	METODOLOGIA DE TRABALHO .....	2
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>4</b>
2.1	BANCOS DE DADOS.....	4
2.2	SISTEMAS GERENCIADORES DE BANCOS DE DADOS(SGBD) .....	4
2.3	MODELAGEM DE BANCOS DE DADOS .....	5
2.4	NORMALIZAÇÃO .....	6
<b>3</b>	<b>DESENVOLVIMENTO .....</b>	<b>7</b>
3.1	MODELO ENTIDADE RELACIONAMENTO .....	7
3.2	MODELO LÓGICO .....	8
3.3	MODELO FÍSICO.....	10
<b>4</b>	<b>CONCLUSÕES .....</b>	<b>13</b>
4.1	UTILIZAÇÃO PRÁTICA .....	13
4.2	ADAPTAÇÕES FUTURAS .....	13
	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>14</b>

# Capítulo 1

## Introdução

Este projeto tem como objeto o desenvolvimento de um banco de dados que possibilite fazer a gestão de conhecimentos em projetos de extensão da Universidade de Brasília, com foco em equipes de competição ligadas à Faculdade de Tecnologia.

O banco de dado tornará possível que membros das equipes de competição pesquisem projetos já desenvolvidos por outros grupos de alunos, sendo possível analisar as áreas de conhecimento envolvidas no projeto e para qual fim este foi idealizado. Assim, para todos os projetos, participantes e experiências estarão registradas e o acesso de membros a esses conhecimentos servirá de ponto de partida para o desenvolvimento de futuras atividades.

### 1.1 Contextualização

Dentro da Universidade de Brasília há várias equipes formadas majoritariamente por graduandos e pós-graduandos que buscam melhorar a formação dos integrantes propiciando atividades práticas em grupo para o desenvolvimento de projetos que serão apresentados em eventos nacionais e/ou internacionais, com foco em eventos de competição, como competições de robótica, de fórmula SAE, BAJA SAE, aeromodelismo e afins. O fato destas equipes serem formadas por alunos acaba fazendo com que as mesmas possuam uma grande rotatividade de pessoas, dificultando o repasse de conhecimento entre os membros mais antigos e membros mais novos, podendo resultar em retrabalho através de projetos repetidos e perda de conhecimento. Portanto, há uma certa necessidade de conseguir guardar informações acerca do que já foi desenvolvido por outros membros de equipes, possibilitando obter informações sobre quem trabalhou em determinado tópico e facilitando o alcance a conhecimentos por parte de membros interessados.

### 1.2 Objetivos

Dado as informações apresentadas, o banco de dados resultante deste projeto tem dois objetivos principais:

- Salvar dados sobre os integrantes de equipes de competição, facilitando a obtenção de contatos e trocas de informações;
- Salvar dados sobre os projetos, com foco nos participantes, finalidade do projeto e áreas de conhecimento envolvidas.

Estes dois objetivos permitiram um acesso mais simples a conhecimentos desenvolvidos no passado, facilitar a obtenção de conhecimento e evitariam a perda de conhecimento ao longo do tempo e retrabalho nas equipes de competição.

Além dos objetivos principais, o banco deverá guardar dados acerca das equipes e das competições que estas participaram, para melhor contextualizar o projeto.

### 1.3 Metodologia de Trabalho

Para desenvolver o banco de dados em questão, a metodologia de trabalho empregada foi bastante sequencial, sendo a principal sequência de passos a seguinte:

1. Definição de objetivos.
2. Descrição do banco proposto através de um modelo entidade-relacionamento.
3. Criação do modelo lógico.
4. Idealização do modelo físico e implementação.
5. Otimização final.

A gestão de conhecimento do projeto foi desenvolvida, primeiramente, realizando uma definição de projeto, esta etapa envolveu conversas e acompanhamentos por parte de membros de equipes de competição, o que permitiu desenvolver um produto com maior aderência com a realidade das equipes. Dessa forma, foram descobertos os principais pontos negativos acerca da deficiência dos projetos da Universidade em guardar e gerenciar conhecimento e o que deveria ser realizado para evitar esta perda no futuro.

A partir disso, criamos uma sequência cronológica e uma organização do projeto. Para a organização acabamos por utilizar o *software Trello*, que agrupa o que é relacionado com o projeto, tanto o que é preciso ser feito ou o que já foi feito. Os códigos desenvolvidos serão atrelados ao *O Github*, para acesso futuro e desenvolvimento paralelo de códigos. Foram utilizados também o *app Whatsapp* para comunicação e o *Overleaf* para a realização do relatório do projeto.

O seguinte passo consistiu então na criação de um modelo entidade relacional, com a idealização das entidades e seus relacionamentos. Seguindo com a ordem cronológica, verificou-se o modelo lógico com sua devida normalização, assim retirando qualquer tipo de ambiguidade do sistema. Por fim, foi implementado o modelo físico, definindo assim o sistema de SGBD do *MySQL* por sua praticidade e a linguagem *Python* para fornecer uma camada de interface do usuário.

Foi idealizado um mapa mental que resume as atividades desenvolvidas ao longo do processo de desenvolvimento, este pode ser visto na figura 1.1

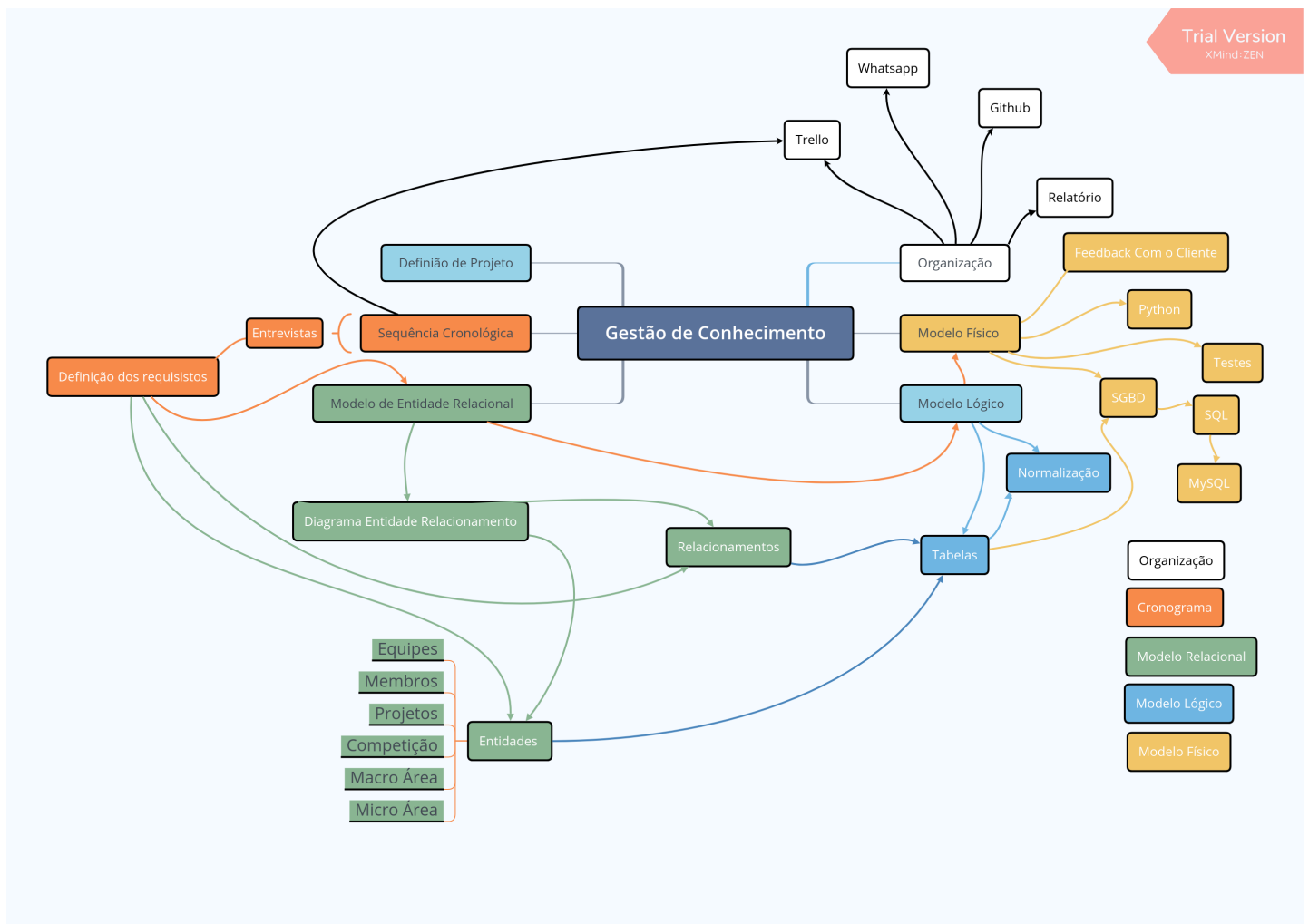


Figura 1.1: Mapa Mental

Nota-se pela figura 1.1 as diferentes colorações para cada etapa de desenvolvimento, sendo que a cor laranja identifica a linha do tempo e os passos seguidos na metodologia empregada.

## Capítulo 2

# Fundamentação Teórica

### 2.1 Bancos de Dados

Os bancos de dados são conjuntos de dados salvos de modo que a pesquisa, inserção e remoção de dados seja facilitada buscando oferecer métodos de acesso e modificação simples e protegidos para o usuário. Os bancos de dados relacionais fornecem métodos de relacionar dados de forma que as informações contidas nele se relacionem entre si e o usuário possa então realizar pesquisas e estudos com eficiência.

### 2.2 Sistemas Gerenciadores de Bancos de Dados(SGBD)

Um sistema de gerenciamento de banco de dados é um conjunto de programas de *software* que permite aos usuários criar, editar, atualizar, armazenar e recuperar dados em tabelas de banco de dados. As características desejáveis em um SGBD são: ter um controle de redundância, compartilhamento de dados, controle de acesso de dados, múltiplas interfaces, representações de associações complexas, garantia de restrições de integridade e recuperação de falhas.

Existem inúmeras vantagens de se usar um SGBD sendo as principais delas é a disponibilidade entre utilizadores diferentes, ou seja, a informação pode ser compartilhada mais facilmente; os dados são mais concisos, e isto gera que uma informação aparece apenas uma vez e assim reduzindo significativamente o custo de armazenamento; a precisão de dados torna o banco de dados mais integro, pois a atualização e alterações dos dados só é permitido a realização em um só lugar; os dados são mais fáceis de acessar e/ou manipular com um SGBD; algumas fontes de informações são protegidas ou garantidas por níveis de seguranças que podem ser vistas por certos usuários através de senhas ou qualquer outro sistema que restringe o acesso aos dados.

## 2.3 Modelagem de Bancos de Dados

Para a criação de um simples modelos de Banco de Dados(BD) são necessários alguns conhecimentos de seu desenvolvimento e suas etapas de criação.

Primeiramente, ao descobrir o seu devido problema que possa ser solucionado com um BD, um mapa mental é indispensável para adicionar um cronograma do trabalho, uma organização de como trabalhar e de como será trabalhado até a chegada do de seu modelo físico.

Em seguida, é interessante realizar um Modelo Entidade e Relacionamento(MER) que será nele que florescerá os temas principais, os dados que serão guardados e os relacionamentos observados. Alguns exemplos de diagramas MERs podem ser vistos a seguir.

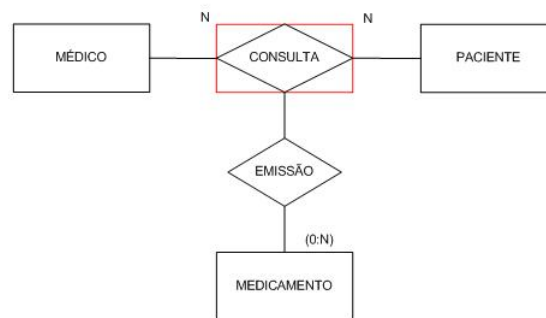


Figura 2.1: Exemplo 1 Diagrama Modelo Entidade Relacional

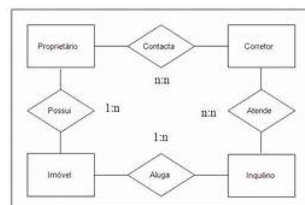


Figura 2.2: Exemplo 2 Diagrama Modelo Entidade Relacional

Partindo do MER, obtém-se o Modelo Lógico(ML), que otimizará o MER em formas de tabelas e que definirá que atributo será chave, quem está se relacionando e etc. Como por exemplo:

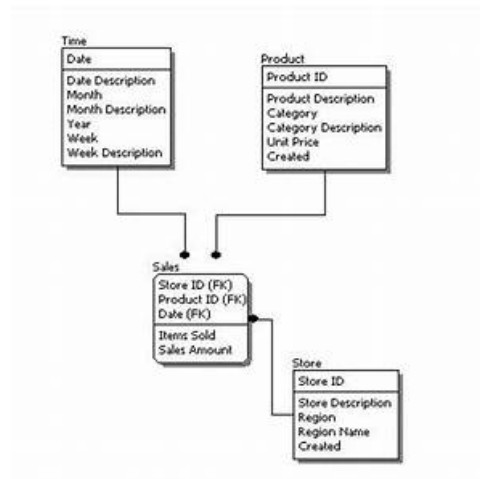


Figura 2.3: Exemplo de Diagrama Modelo Lógico

E por fim, temos o desenvolvimento físico do BD, onde é possível escolher em qual tipo de servidor serão armazenados estes dados, como *MySQL*, *Oracle*, *SQLite* e etc. A linguagem SQL - *Structured Query Language* é a mais utilizada em aplicações de bancos de dados, fornecendo meios de comunicação com o servidor gerenciador de banco de dados e facilitando a implementação por parte do usuário.

## 2.4 Normalização

Durante a etapa de idealização do modelo lógico busca-se reduzir a quantidade de dados repetidos no banco de dados, visando uma otimização por parte da memória gasta por este, para tal é realizado um processo denominado normalização. Este processo foi proposto em 1972 por Edgar F. Codd [1] e permite criar bancos de dados mais simples e mais otimizados.

A normalização é composta por três etapas, onde cada etapa gera um modelo lógico na respectiva forma normal. Na primeira etapa busca-se retirar dados multivalorados dos dados simples e simplificar campos compostos. A segunda forma normal garante que os atributos dependam da chave primária da tabela relacionada em sua totalidade, já o último passo faz com que os dados sejam obtidos dependam só da chave primária e não de outros dados da tabela.



## Capítulo 3

# Desenvolvimento

### 3.1 Modelo Entidade Relacionamento

Após a definição de objetivos junto ao cliente, chegou-se a uma conclusão acerca das entidades e relacionamentos que devem compor o banco de dados em questão, em relação às entidades observadas, têm-se:

- Integrante -> Agrupa os meios de comunicação para se comunicar com os integrantes das equipes.
- Equipe -> Guarda informações básicas das equipes, como nome e curso predominante.
- Competição -> Informações sobre as competições que equipes participaram no passado.
- Projeto -> Dados suficientes sobre os projetos desenvolvidos para saber o que foi realizado e meios de acessar documentos referentes ao projeto.
- Área de conhecimento -> Salva as diversas áreas de conhecimento utilizadas ao longo dos projetos.

Já em relação às relações, observa-se o seguinte:

- Uma equipe possui vários integrantes e uma pessoa deve participar de no mínimo uma equipe.
- Um integrantes possui um determinado nível de conhecimento em certa área.
- Um projeto envolveu a participação de integrantes e uma pessoa participa de vários projetos.
- Uma competição deve ter a participação de alguma equipe e uma equipe participa de várias competições.
- Um projeto deve ser ligado a exatamente uma competição.
- Vários projetos podem ser desenvolvidos para a mesma competição.

- Para o desenvolvimento de um projeto, são utilizadas várias áreas de conhecimento, cada uma com um nível de relevância para o projeto.

Um diagrama que resume essas interações pode ser visto na figura 3.1.

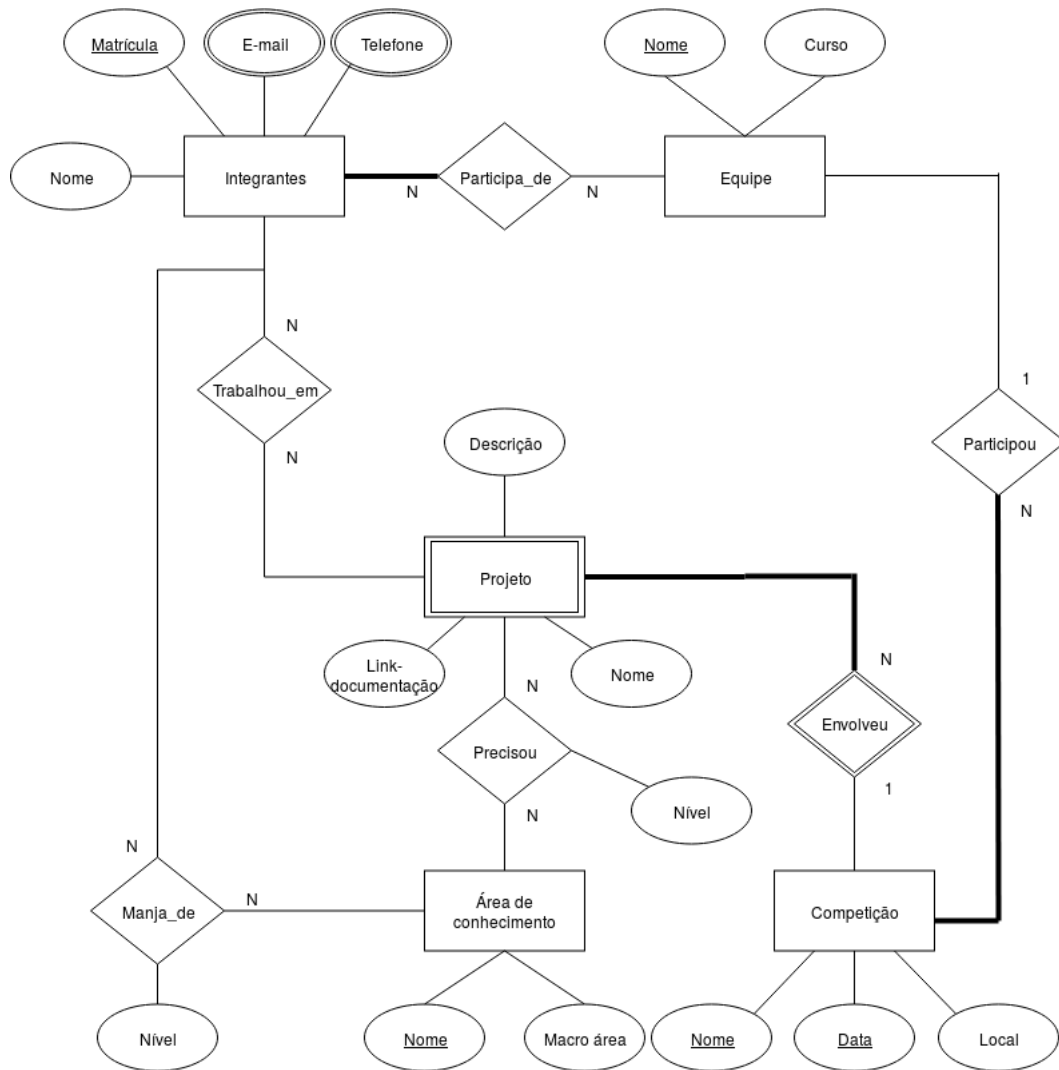


Figura 3.1: Diagrama Modelo Entidade Relacional

A figura resume bem o que é necessário pelas equipes para se ter o acesso a conhecimentos gerais sobre os projetos das equipes, mostrando o grau de relacionamento entre as entidades e a participação destas nos relacionamentos.

## 3.2 Modelo Lógico

Partindo do Modelo de Entidades e Relacionamentos, foi construído o Modelo Lógico do sistema, organizando os dados em tabelas. A ideia inicial de representação pode ser vista na figura 3.2.

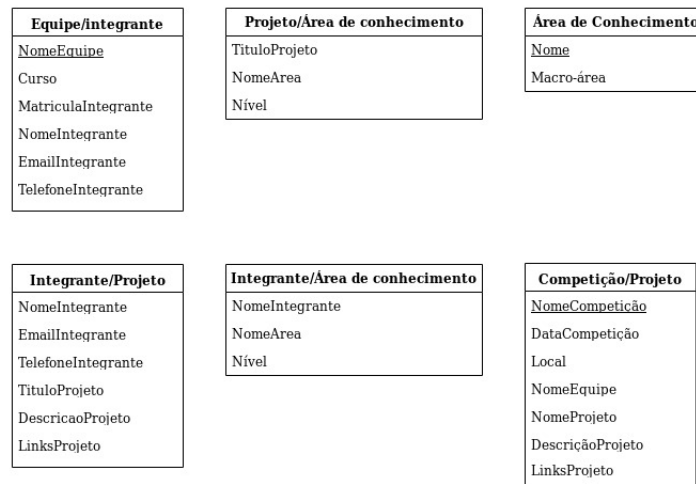


Figura 3.2: Modelo Lógico inicial do sistema

Essa representação apresentaria uma grande repetibilidade de dados, algo indesejável em relação à utilização de memória, portanto, foi realizada uma normalização do sistema e obteve-se então um modelo mais otimizado e de melhor entendimento, disposto na figura 3.3.

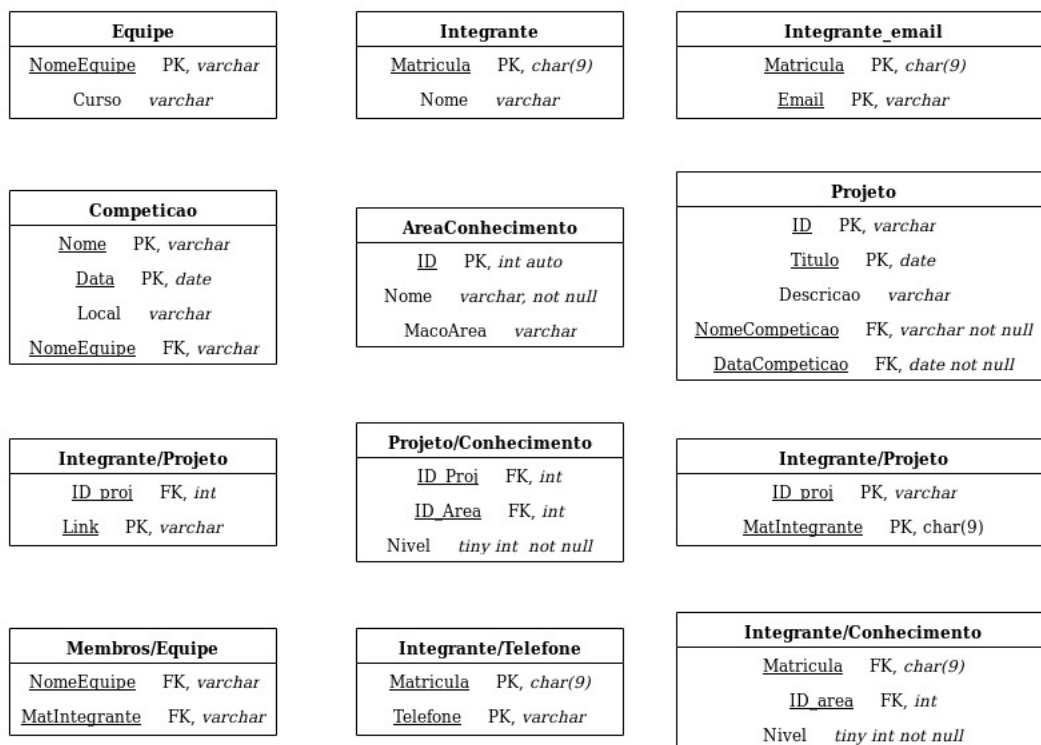


Figura 3.3: Modelo Lógico do sistema

Uma das regras utilizadas para fazer a conversão do MER para o modelo lógico, observada no modelo lógico normalizado, foi a transição de relações com cardinalidade NxN para uma nova tabela, separada das tabelas envolvidas na relação. Atributos multi-valorados também são repre-

sentados em tabelas separadas.

Esse modelo já é capaz de identificar os principais atributos, chaves e tipos de atributos que vão vir a compor as tabelas do modelo físico do sistema, facilitando a futura implementação do sistema.

O nível de conhecimento do integrante acerca de determinada área e o nível de relevância desta área em algum projeto serão dados em uma escala de 1 a 5, sendo 1 pouco relevante e 5 muito relevante.

### 3.3 Modelo Físico

O modelo físico foi dividido em duas camadas, uma camada de acesso mais direto ao banco de dados e outra camada que permite uma interface de acesso aos usuários do sistema, facilitando e limitando *queries* para direcionar as pesquisas sobre os projetos desenvolvidos.

O SGBD escolhido foi o MySQL, por apresentar diversas funcionalidades que facilitam a implementação do banco de dados nele, além de garantir segurança ao acesso por parte do usuário e realizar algumas atualizações automáticas no banco de dados. Foram utilizadas na criação do banco de dados chaves primárias, chaves estrangeiras e diversos tipos de atributos, buscando utilizar ao máximo as propriedades disponíveis no MySQL para facilitar a implementação.

Dentre as funcionalidades utilizadas encontram-se as ‘*CONSTRAINTS*’ do tipo ‘*PRIMARY KEY*’, ‘*FOREIGN KEY*’, ‘*UNIQUE*’ e ‘*CHECK*’. Para representar as diversas áreas de conhecimento e os projetos foram utilizadas variáveis do tipo INT, com modificador ‘*UNSIGNED*’ e ‘*AUTO\_INCREMENT*’. Para os níveis de conhecimento utilizados por projetos e de integrantes foram utilizados ‘*TINYINT*’, pois estes valores são limitados entre 1 e 5. Um exemplo de conversão entre as tabelas do modelo lógico e o código físico para implementação do sistema pode ser visto na figura 3.4, onde são demonstradas as criações das tabelas referentes aos projetos e competições.

```
69 # Competition event table
70 query = """
71     CREATE TABLE IF NOT EXISTS Competicao (\
72         nome_comp VARCHAR(40) NOT NULL,\
73         data DATE NOT NULL,\
74         local CHAR(30),\
75         nome_equipe VARCHAR(30) NOT NULL,\
76         CONSTRAINT unique_comp UNIQUE (nome_comp, data),\
77         CONSTRAINT pk_comp PRIMARY KEY (nome_comp, data, nome_equipe),\
78         CONSTRAINT fk_comp_equipe FOREIGN KEY (nome_equipe) REFERENCES Equipe (nome_equipe) ON DELETE CASCADE\
79     );
80 """
81 db_cursor.execute(query)
82
83 # Projects tables
84 query = """
85     CREATE TABLE IF NOT EXISTS Projeto (\
86         id_proj INT NOT NULL AUTO_INCREMENT,\
87         titulo TEXT NOT NULL,\
88         descricao TEXT NOT NULL,\
89         nome_comp VARCHAR(40) NOT NULL,\
90         data_comp DATE NOT NULL,\
91         CONSTRAINT pk_projeto PRIMARY KEY (id_proj, nome_comp, data_comp),\
92         CONSTRAINT fk_projeto_comp FOREIGN KEY (nome_comp, data_comp) REFERENCES Competicao (nome_comp, data) ON DELETE NO ACTION\
93     )
94 """
95 db_cursor.execute(query)
```

Figura 3.4: Código parcial para criação do modelo físico

Nota-se pela figura 3.4 a junção entre a linguagem *python* e a SQL, onde as *queries* são englobadas em *strings* e a comunicação com o MySQL é realizada através de objetos disponíveis na biblioteca ‘MySQLdb’.

A interface gráfica desenvolvida busca facilitar a vida do usuário e limitar as ações sobre o banco de dados, a página principal desenvolvida pode ser vista na figura 3.5.

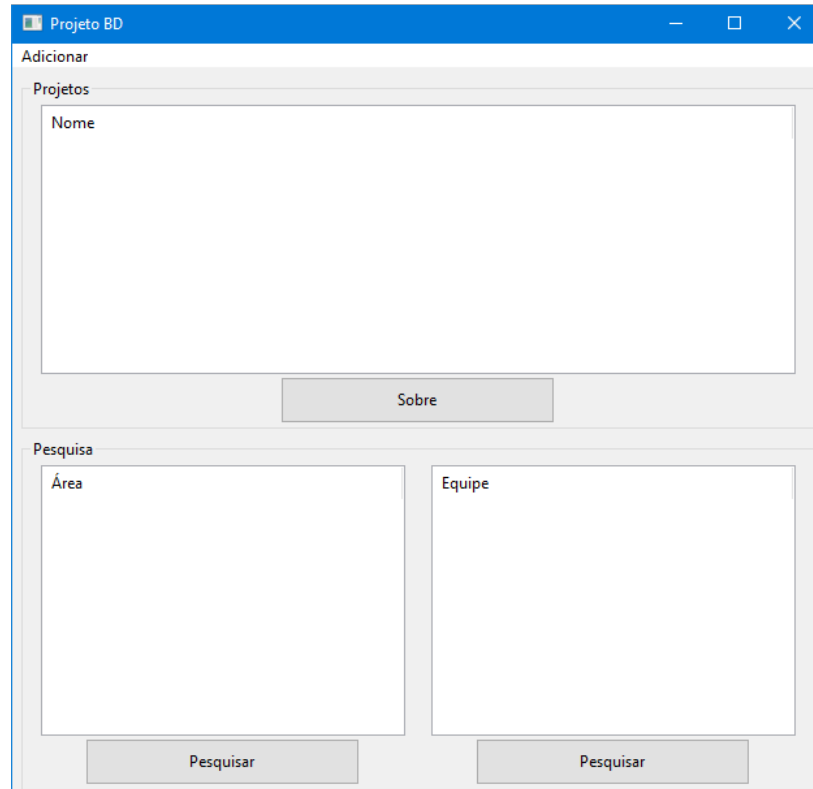


Figura 3.5: Interface gráfica base do programa

Para a comunicação entre a interface e a camada de acesso ao SGBD MySQL, foram utilizadas *queries* pré definidas que realizariam de fato alterações e pesquisas no banco de dados. Essas *queries* são responsáveis por criar o banco de dados e as tabelas deste caso não existam e permitir a inserção de novos dados na maneira ‘INSERT INTO nome\_tabela VALUES (...)’, e também a remoção e pesquisa por dados. Com um foco na pesquisa de projetos e contatos dos participantes envolvidos, é importante também a utilização de *queries* que tornem possível essas pesquisas, alguns exemplos de *queries* desenvolvidas para este fim estão listadas abaixo:

- SELECT id\_proj FROM Proj\_conhecimento WHERE id\_con = x AND nivel > 2;
- SELECT titulo, descricao FROM Projeto WHERE id\_proj = y;
- SELECT nome\_comp, data FROM Competicao WHERE nome\_equipe = z;
- SELECT titulo, descricao FROM Projeto WHERE nome\_comp = w;

As *queries* acima foram utilizadas também em conjunto por meio de *sub-queries* e garantindo um manuseio mais fácil do banco de dados.

Algumas instruções em SQL utilizadas para inserir e remover os dados no banco de dados podem ser vistas abaixo, sendo estas utilizadas sempre de modo a obedecer a ordem e as restrições impostas no momento de criação das tabelas.

- INSERT INTO Equipes (nome\_equipe, curso) VALUES('A', 'Mecânica');
- INSERT INTO Integrante (matricula, nome\_int) VALUES ('150111111', 'P');
- INSERT INTO Int\_emails (matricula, email) VALUES ('150111111', 'p@gmail.com');
- DELETE FROM Equipes WHERE nome\_equipe = 'A';
- DELETE FROM Integrante WHERE nome\_int = 'P';
- DELETE FROM Int\_emails WHERE matricula = '150111111';

A interface que mostra dados sobre o projeto pode ser vista na figura 3.6, que permite obter os contatos dos participantes e o nível de conhecimento das diversas áreas envolvidas.

Figura 3.6: Pagina com informações acerca de um projeto

Por fim, para garantir a participação total de alguma entidade em algum relacionamento, foram utilizadas *CONSTRAINTS* do tipo *FOREIGN KEY* e o método de remoção por cascata do SGBD MySQL, garantindo assim o funcionamento do modelo físico de acordo com as definições iniciais do modelo entidade-relacionamento.

## Capítulo 4

# Conclusões

### 4.1 Utilização Prática

Somente com o passar do tempo e a inserção de projetos no banco de dados poderá ser verificado o seu uso prático real, no entanto, é possível afirmar que o banco de dados cumpre o que foi proposto, que seria propiciar um meio para centralizar e facilitar o acesso aos diversos conhecimentos desenvolvidos pelas equipes de competição da Faculdade de Tecnologia da UnB.

O MySQL forneceu ferramentas suficientes para a implementação do banco de dados na maneira desejada, garantindo restrições pedidas pelo cliente de maneira satisfatória e automatizada, facilitando o trabalho por parte do time de implementação e garantindo bons resultados com o projeto.

### 4.2 Adaptações Futuras

Com o objetivo de descentralizar o acesso ao Banco de Dados pode ser desenvolvida uma página WEB para acesso a este, garantindo que os usuários não precisem estar fisicamente próximos ao computador onde está o SGBD para conseguir acessar o banco de dados.

Seria interessante também permitir que diferentes equipes participem da mesma categoria, visto que o número de equipes e projetos de extensão cresce com o tempo, identificando os projetos também pelas equipes responsáveis. Isto permitira incluir equipes também com sede no campus Gama da Universidade de Brasília, garantindo não só um maior entrosamento entre as equipes de competição mas também entre os campus da Instituição.

# REFERÊNCIAS BIBLIOGRÁFICAS

- [1] PAPAIZ, F. *Slides Prof. Fabiano Papaiz*.
- [2] SGBD. Disponível em: <<http://ehgomes.com.br/disciplinas/bdd/sghd.php>>.
- [3] BANCO de Dados. Disponível em: <[https://pt.wikipedia.org/wiki/Banco\\_de\\_dados](https://pt.wikipedia.org/wiki/Banco_de_dados)>.