



GUSTAVO RODRIGUES SOUSA

**CRIAÇÃO DO SISTEMA DE BANCO DE DADOS COMO APOIO
PARA CARDÁPIOS AUTOMATIZADOS DE RESTAURANTES
UNIVERSITÁRIOS**

LAVRAS - MG

2022

GUSTAVO RODRIGUES SOUSA

**CRIAÇÃO DO SISTEMA DE BANCO DE DADOS COMO APOIO PARA CARDÁPIOS
AUTOMATIZADOS DE RESTAURANTES UNIVERSITÁRIOS**

Projeto Acadêmico apresentado à Universidade Federal de Lavras como parte das exigências do curso de Ciência da Computação, para a obtenção do título de Bacharel.

Prof. Dr. Ramon Gomes Costa

Orientador

Prof. Dra. Andreza Cristina Beezão Moreira

Coorientadora

LAVRAS - MG

2022

LISTA DE FIGURAS

Figura 3.1 – Logotipo Gerauca	10
Figura 4.1 – Diagrama de Entidade Relacionamento	16
Figura 4.2 – Exemplo Tabela Pivô	17
Figura 4.3 – Diagrama Entidade Relacionamento de nível lógico	19
Figura 4.4 – Tabela de ingredientes	21
Figura 4.5 – Tabela de preparações	22
Figura 4.6 – Código da tela de boas vindas	27
Figura 4.7 – Código da tela de listagem de ingredientes	28
Figura 5.1 – Tela de boas vindas do sistema	32
Figura 5.2 – Tela de login	33
Figura 5.3 – Menu de acesso	33
Figura 5.4 – Tela de listagem de ingredientes	34
Figura 5.5 – Tela de cadastro de ingrediente	35
Figura 5.6 – Tela de listagem de preparações	36
Figura 5.7 – Tela de informações extras da preparação	37
Figura 5.8 – Tela de edição da preparação	38
Figura 5.9 – Tela de confirmação de exclusão da preparação	39
Figura 5.10 – Tela de cadastro de excludentes da preparação	40

LISTA DE CÓDIGOS

Código 4.1	Busca por todos ingredientes	23
Código 4.2	Selecionar todas as guarnições	24
Código 4.3	Selecionar preparação excludente	25

SUMÁRIO

1	Introdução	5
2	Tecnologias utilizadas	6
2.1	TerraER	6
2.2	SQL	6
2.3	MariaDB	6
2.4	MySQL Workbench	7
2.5	HTML	7
2.6	CSS	7
2.7	Bootstrap	7
2.8	JavaScript	8
2.9	PHP	8
3	Descrição geral do tema	9
4	Implementação	14
4.1	Diagrama entidade relacionamento conceitual de alto-nível	14
4.2	Diagrama de entidade relacionamento de nível lógico	17
4.3	Dicionário de dados	20
4.3.1	Tabela de Ingredientes	20
4.3.2	Tabela de Preparações	20
4.4	Consultas criadas para a utilização do sistema	23
4.4.1	Busca por todos os ingredientes	23
4.4.2	Busca por preparações	23
4.4.3	Buscar por preparações excludentes	25
4.5	Códigos das telas	26
5	Estudo de caso	29
5.1	Atividades realizadas	29

5.1.1	Outubro de 2019 a Novembro de 2019	29
5.1.2	Dezembro de 2019	29
5.1.3	Janeiro de 2020	29
5.1.4	Fevereiro de 2020	30
5.1.5	Março de 2020	30
5.1.6	Abril de 2020	30
5.1.7	Maio de 2020	30
5.1.8	Junho de 2020	30
5.1.9	Julho de 2020	31
5.1.10	Agosto de 2020	31
5.1.11	Setembro de 2020	31
5.1.12	Outubro de 2020	31
5.1.13	Novembro de 2020	31
5.1.14	Dezembro de 2020	32
5.2	Sistema na prática	32
6	Conclusão	41
	REFERÊNCIAS	43

1 INTRODUÇÃO

Este trabalho tem como objetivo apresentar os resultados obtidos durante a realização do projeto do sistema de banco de dados como apoio para o cardápio automatizado do restaurante universitário.

Este sistema foi construído para servir de apoio ao projeto de geração automática de cardápios do restaurante universitário da UFLA. O objetivo consiste em auxiliar a equipe de nutricionistas da instituição no planejamento de refeições para o restaurante, visando a otimização do uso de ingredientes e das refeições servidas, prezando pela nutrição balanceada dos cardápios e evitando desperdícios desnecessários.

Para que a geração automática de cardápios funcionasse, necessitava-se dados sobre os alimentos, uma vez que os cardápios são baseados nos possíveis ingredientes e preparações, para tanto, foi solicitado o desenvolvimento de uma plataforma *web* e um sistema de banco de dados para a realização do cadastro e manipulação de informações sobre os cardápios. Estes dados, então, são utilizados como entrada para o algoritmo de geração automática de cardápios.

O projeto foi realizado como bolsa institucional ofertada pela UFLA pelo edital do Programa de Aprendizado Técnico (PROAT) - N°25/2019/PIB/PRAEC/UFLA.

2 TECNOLOGIAS UTILIZADAS

Foram utilizadas as seguintes tecnologias para implementação do sistema:

2.1 TerraER

TerraER¹ é um software gratuito e de código aberto, desenvolvido com o intuito de auxiliar a comunidade universitária a desenvolver diagramas relacionais que representem a estrutura de um banco de dados. Este software é muito popular entre alunos que estão aprendendo sobre banco de dados, e foi utilizado para a geração dos diagramas apresentados neste trabalho.

2.2 SQL

SQL² é um acrônimo que significa *Structured Query Language*, ou Linguagem de Consulta Estruturada, ou seja, é uma linguagem de programação que surgiu para padronizar o modo que se comunica com os Sistemas Gerenciadores de Banco de Dados (SGBDs). Devido à sua facilidade de uso, ela foi amplamente adotada por empresas desenvolvedoras de SGBDs, e é a linguagem mais utilizada para comunicação com banco de dados (DB-ENGINES. . . , 2022). Com SQL, é possível implementar bancos de dados inteiros, realizar funcionalidades como criar tabelas, consultar dados, modificar a estrutura dos dados, definir relacionamentos entre os dados, entre outras. Neste trabalho, a linguagem foi utilizada para criação da base de dados do sistema, bem como para consulta, edição e manipulação destes dados.

2.3 MariaDB

MariaDB³ é um SGBD que surgiu como uma opção gratuita ao seu antecessor: o MySQL. Na prática, a maioria dos comandos estruturados em SQL desenvolvidos para serem executados no MySQL, funcionarão no MariaDB. Esta tecnologia foi utilizada neste trabalho para a implementação do banco de dados desenvolvido. (MARIADB, 2021)

¹ <http://www.terraer.com.br/>

² <https://www.w3schools.com/sql/>

³ <https://mariadb.org/>

2.4 MySQL Workbench

MySQL Workbench⁴ é uma ferramenta visual para a administração de banco de dados, gratuita e desenvolvida pela Oracle Corporation (ORACLE, 2022). Com ela é possível implementar bancos de dados, produzir diagramas de nível lógico, realizar manutenção em bancos de dados, dentre outras. Esta ferramenta foi utilizada principalmente para a produção de diagramas relacionais de nível lógico sobre o banco de dados construído.

2.5 HTML

HyperText Markup Language⁵ (HTML) é a linguagem de definição de documentos mais utilizada na internet para a criação de páginas *web*. Estas páginas em formato HTML são interpretadas pela vasta maioria dos navegadores *web*, que então as formatam e apresentam ao usuário final.

2.6 CSS

Cascading Style Sheets⁶ (CSS) se trata de uma linguagem de definição de estilos, geralmente utilizada conjuntamente com o HTML. Ela serve para definir estilos, cores, *layouts*, e outras características pertinentes ao documento HTML.

2.7 Bootstrap

Bootstrap⁷ é um *framework web* de código-fonte aberto para desenvolvimento de interfaces gráficas de aplicações *web* que utilizam HTML, CSS e JavaScript. Este *framework* possui componentes gráficos prontos para uso em diversas aplicações, que facilitam e agilizam o desenvolvimento dos *softwares*.

⁴ <https://www.mysql.com/products/workbench/>

⁵ <https://www.w3schools.com/html/>

⁶ <https://www.w3schools.com/css/>

⁷ <https://getbootstrap.com/>

2.8 JavaScript

JavaScript⁸ é uma linguagem de programação interpretada, de alto nível e com tipagem dinâmica. A sua maior utilização se dá em páginas *web* feitas em HTML e CSS, servindo para alterar elementos da página em tempo real, sendo executada na máquina do usuário que acessa a *web*. Com ela, é possível criar páginas *web* dinâmicas, que se atualizam em tempo de execução, permitindo operações muito mais complexas do que em páginas estáticas.

2.9 PHP

PHP⁹ é uma linguagem de programação interpretada, usada principalmente para o desenvolvimento de aplicações *web*. Com o PHP, é possível gerar documentos HTML dinamicamente, que são modificados em tempo de execução pelo servidor que hospeda a aplicação. Estas páginas dinâmicas, então são enviadas ao usuário que pode interagir com a aplicação e fazer novas requisições ao servidor, conforme a utiliza. Neste trabalho, o código produzido em PHP é capaz de interagir com o SGBD e gerar as páginas do sistema dinamicamente, dependendo dos dados que estavam armazenados.

⁸ <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>

⁹ <https://www.php.net/>

3 DESCRIÇÃO GERAL DO TEMA

A ideia do desenvolvimento de um sistema de banco de dados para armazenar informações nutricionais sobre o restaurante universitário se deu a partir do momento em que detectou-se uma oportunidade de melhoria que poderia ser feita na geração de cardápios diários para serem servidos no restaurante universitário da UFLA. Este processo de elaboração dos cardápios é feito manualmente por uma das nutricionistas, de tempos em tempos, a fim de definir os cardápios que serão servidos em um horizonte de planejamento, considerado curto. Tendo em vista que é um processo trabalhoso e demorado, o professor Dr. Mayron César de Oliveira Moreira, do Departamento de Ciência da Computação, lecionava Programação Matemática, na ocasião, e vislumbrou uma oportunidade de utilizar este problema como exemplo em sua matéria. Porém, devido à complexidade do problema, o professor Mayron juntamente com a professora Dra. Andreza Cristina Beezão Moreira e o professor Dr. Ramon Gomes Costa, transformaram-no em um projeto acadêmico, criando bolsas estudantis para que alunos pudessem colaborar. Agradecimentos especiais para: Dra. Ana Paula Piovesan Melchiori e Dra. Emília Cristina Mões pelo apoio oferecido ao projeto. Foram abertos dois editais de bolsas acadêmicas: um para a criação do método de otimização, onde foi proposto um modelo matemático, e outro para a implementação do sistema de banco de dados que auxilia o modelo matemático, etapa da qual o autor deste trabalho fez parte. Visando assim, tornar o trabalho de planejamento de refeições do restaurante universitário célere e descomplicado.

A proposta então foi estudada e dada como possível de se construir, uma vez que todas as regras de criação dos cardápios foram fornecidas pelas nutricionistas do restaurante universitário, resultando na representação matemática do problema, através da modelagem. Tendo em vista que todas as regras para a construção dos cardápios puderam ser identificadas e postas no papel, abriu-se um edital para seleção de estudantes bolsistas que colaborariam com a construção do sistema. Assim, iniciou-se o desenvolvimento de um *software* que fosse capaz de analisar dados sobre os alimentos, e montar o cardápio automaticamente, utilizando a solução advinda do modelo matemático que, por sua vez, obedece as restrições e regras propostas pela equipe de nutrição.

Em face da construção deste *software*, viu-se a necessidade de possuir os dados sobre os alimentos que seriam utilizados no cardápio, uma vez que a alimentação deve prezar pelas indi-

cações nutricionais para os alimentados, respeitando-se todas as condições dietéticas, nutricionais e também levando-se em conta a limitação de tempo de produção, quantidade de equipamentos e mão de obra.

Com a impossibilidade de obter estes dados de forma organizada e estruturada para o uso no *software* de geração automática de cardápios, foi proposta a criação de um outro projeto, que seria um banco de dados, contemplando a capacidade de conter todos os dados estipulados pelas nutricionistas, centralizando-os em um único lugar, de forma a facilitar o uso do *software*.

Assim surgiu esse projeto. Com a criação de um sistema de banco de dados personalizado para o uso específico do corpo nutricional da UFLA, seria possível fazer com que o *software* gerador de cardápios, tivesse à sua disposição, dados atuais e condizentes com a realidade do problema. Estes dados seriam então cadastrados e atualizados conforme demanda dos nutricionistas envolvidos.

Analisando-se a iminente construção do banco de dados, viu-se outra necessidade: a construção de uma interface gráfica agradável para uso dos nutricionistas, uma vez que não seria viável ter um administrador de banco de dados disponível a todo momento. Então, a criação do sistema *web* aconteceu de forma concomitante ao banco de dados.

O início deste projeto se deu em 2018, com dois ex-bolsistas responsáveis pela criação do sistema *web* e do banco de dados. Foram realizados estudos da identidade visual que o *website* assumiria, *layout* do sistema e a construção inicial do banco de dados, contemplando as demandas solicitadas na época. Além disso, eles nomearam o sistema *web* carinhosamente como Gerauca, um acrônimo para Geração Automática de Cardápios. Um nome que teve impacto na geração da logotipo do sistema, que remete à uma senhora acolhedora, como pode ser visto na Figura 3.1.

Figura 3.1 – Logotipo Gerauca



Conforme o andamento do projeto ocorria, e com o encerramento da vigência da bolsa de estudos dos ex-integrantes do projeto, encontravam-se novas demandas e funcionalidades que precisavam existir no sistema, para que o algoritmo de geração automática de cardápios tivesse suas necessidades cumpridas. Foi então que o autor deste trabalho iniciou suas atividades como bolsista do programa PROAT¹ em 2019 e encerrou suas atividades em 2020.

A descrição matemática do problema nutricional abordado não é foco principal deste trabalho, mas por estar intrinsecamente ligada às atividades exercidas pelo autor, será brevemente explicitada aqui.

O objetivo dos trabalhos descritos é o de otimizar a montagem de cardápios servidos no restaurante universitário, prezando pela otimização de ingredientes, variedade das preparações durante a semana, minimizando o esforço de confecção do cardápio e agilizando a obtenção de resultados.

Diante disso, as condições e restrições que nortearam a criação do modelo matemático que representa o contexto do restaurante universitário da UFLA, são:

Critério de qualidade do algoritmo: maximização do número de preparações selecionadas.

- Restrição 1: Garante pelo menos 70% de utilização das preparações.
- Restrição 2: Uma mesma preparação não pode ser servida no almoço e no jantar.
- Restrição 3: Deve-se ter no mínimo uma e no máximo duas preparações proteicas por almoço e por jantar.
- Restrição 4: Deve-se ter duas saladas por almoço e por jantar.
- Restrição 5: Deve-se ter no máximo uma preparação (somente) vegetariana por almoço e por jantar.
- Restrição 6: Deve-se ter uma preparação vegana por almoço e por jantar.
- Restrição 7: Deve-se ter uma guarnição por almoço e por jantar. Há preparações proteicas que dispensam guarnição (por exemplo, escondidinho de carne seca).

¹ Programa de Aprendizado Técnico

- Restrição 8: Deve-se evitar, no mesmo cardápio, preparações que sejam excludentes em quesitos como uso de infraestrutura (espaço no forno; fritadeiras; número de formas), baixa aceitação pelos comensais, cor e textura.
- Restrição 9: Há preparações que aparecem sempre juntas em um cardápio (por exemplo, coxa/sobrecoxa de frango e creme de milho). Para uma preparação pode haver uma lista de possíveis opções para aparecerem juntas a esta.
- Restrição 10: 4 vezes carne de aves por semana.
- Restrição 11: 5 vezes carne bovina por semana.
- Restrição 12: 2 vezes carne suína por semana.
- Restrição 13: carne suína sempre com uma carne bovina ou de ave, como alternativa extra (caráter religioso).
- Restrição 14: 1 vez pescado ou prato de alto custo ou preparação de difícil elaboração (panqueca, lasanha, galinhada, feijoadada, etc).
- Restrição 15: 1 vez guarnição do tipo massa por semana.
- Restrição 16: 2 vezes caldos (polenta, canjiquinha, etc) por semana.
- Restrição 17: 1 vez guarnição seca (batata palha ou farofa) por semana.
- Restrição 18: 6 vezes guarnições do tipo refogado com legumes por semana.
- Restrição 19: 1 vez preparação do tipo fritura por semana.
- Restrição 20: As opções de salada podem ser compostas por 1 folha e 1 legume ou 2 folhas, mas nunca dois legumes, ou seja, proíbe dois legumes.
- Restrição 21: Preparações do mesmo grupo não devem aparecer no mesmo dia.
- Restrição 22: Não há refeições na janta do sábado e domingo.

- Restrição 23: Frutas não devem aparecer no almoço dos dias de semana.
- Restrição 24: Entre proteína e guarnição, não é permitido seco-seco.
- Restrição 25: Entre proteína e guarnição, não é permitido caldo-caldo.
- Restrição 26: Entre proteína e guarnição, não é permitido seca-massa.
- Restrição 27: Entre proteína e guarnição, não é permitido caldo-refogado.
- Restrição 28: Como forma de "forçar" uma boa variabilidade de cardápios, impôs-se que cada preparação proteica, guarnição e vegana apareçam no máximo uma vez por semana.
- Restrição 29: 14 preparações proteicas por semana.
- Restrição 30: 10 guarnições por semana.
- Restrição 31: no máximo 24 saladas por semana.
- Restrição 32: no máximo 1 fruta por semana.
- Restrição 33: 12 preparações veganas por semana.
- Restrição 34: no máximo 12 preparações vegetarianas por semana.

O sistema então foi construído obedecendo todas estas características e restrições descritas.

4 IMPLEMENTAÇÃO

Neste capítulo são apresentadas as características de implementação do banco de dados, diagrama de entidade-relacionamento (em nível conceitual e lógico), dicionário de dados, bem como a construção da aplicação *web* que é a interface gráfica do usuário final.

4.1 Diagrama entidade relacionamento conceitual de alto-nível

O diagrama entidade relacionamento conceitual de alto-nível é uma representação gráfica construída nas fases iniciais do planejamento do banco de dados, e serve para visualizar como as entidades se comportam e se relacionam entre si.

Proposto por Peter Chen (PETER. . . , 1976), o modelo de entidade relacionamento (MER) tem as seguintes características: As entidades devem ser representadas por retângulos, seus atributos por elipses e os relacionamentos por losangos. As entidades podem ser ligadas por linhas, que devem representar também sua cardinalidade, podendo ser de 1 para 1, 1 para vários ou vários para vários.

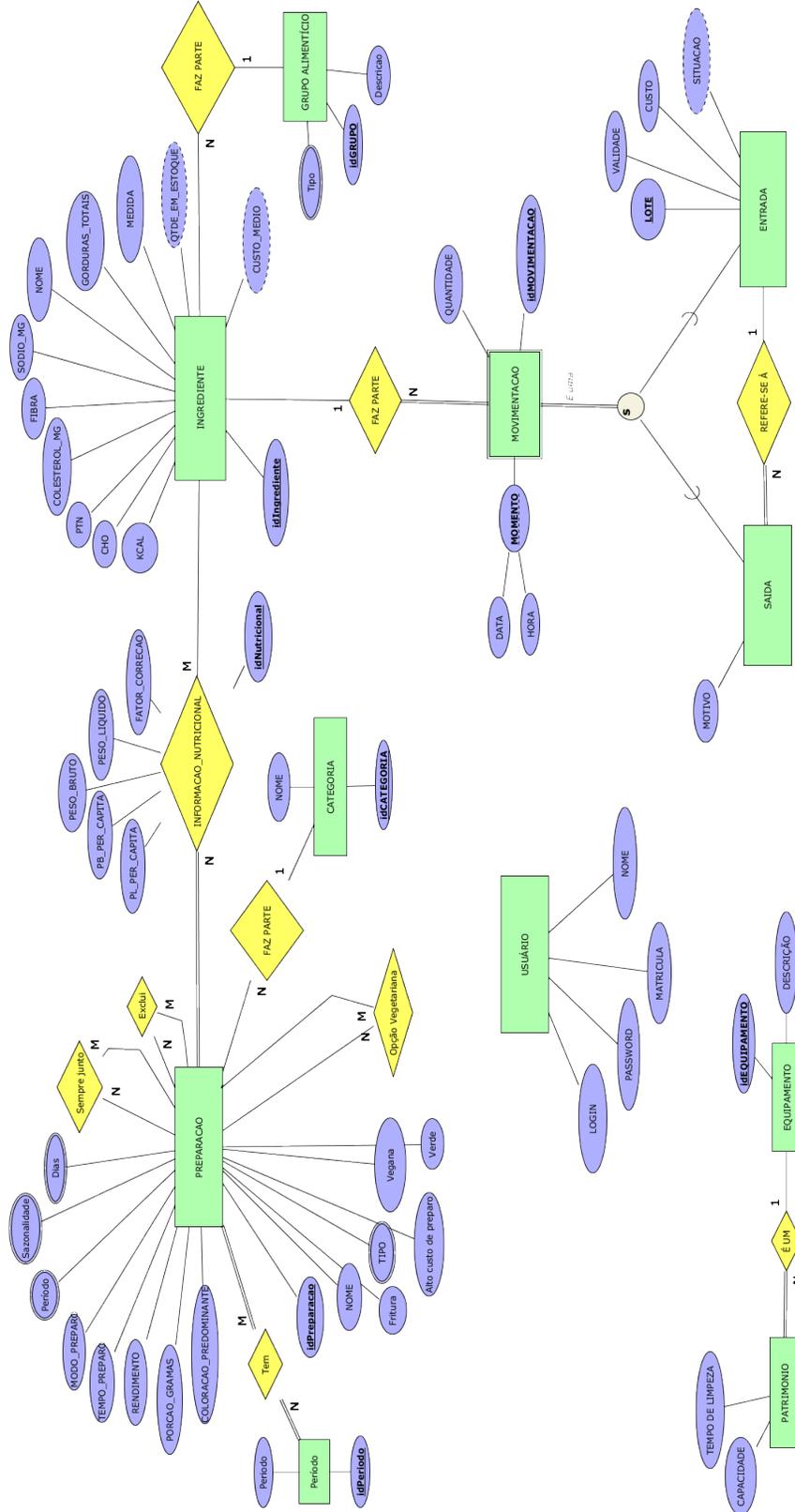
Para o desenvolvimento desse trabalho, foi criado o diagrama utilizando o TerraER que pode ser visto na Figura 4.1. No domínio representado pelo diagrama ER¹ conceitual de alto nível, temos as seguintes entidades e relacionamentos:

- Preparação faz parte de uma Categoria;
- Categoria faz parte de uma ou várias preparações;
- Preparação está Sempre Junta de nenhuma ou várias Preparações;
- Preparação Exclui nenhuma ou várias Preparações;
- Preparação é uma Opção Vegetariana de nenhuma ou de várias Preparações;
- Preparação tem nenhum ou vários Períodos;
- Períodos tem nenhum ou várias Preparações;

¹ Entidade Relacionamento

- Preparação possui nenhum ou vários Ingredientes;
- Ingrediente pertence a nenhuma ou várias preparações;
- Ingrediente faz parte de um Grupo Alimentício;
- Grupo Alimentício pertence a nenhum ou vários ingredientes;
- Ingrediente faz parte de nenhuma ou várias Movimentações;
- Movimentação pertence a um Ingrediente;
- Movimentação é uma Entrada ou Saída;
- Entidade Usuário;
- Equipamento é um Patrimônio;
- Patrimônio é nenhum ou vários Equipamentos.

Figura 4.1 – Diagrama de Entidade Relacionamento



4.2 Diagrama de entidade relacionamento de nível lógico

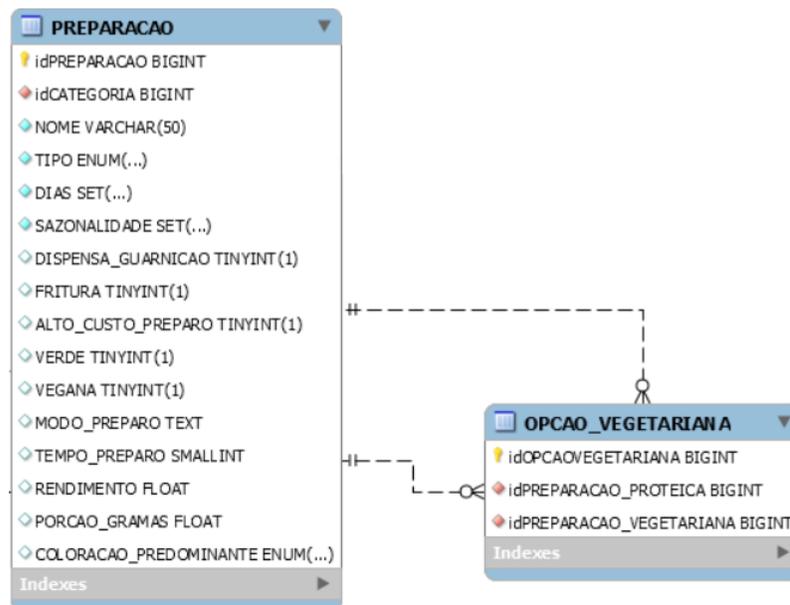
Outra maneira de se representar o diagrama do (MER)² é em nível lógico. Este formato apresenta mais informações pertinentes à implementação do banco de dados em si, tais como, os tipos dos dados, as tabelas que servirão de pivô a relacionamentos, se algum dado pode ou não ser vazio e quais atributos são chaves primárias e chaves estrangeiras.

Tipo de dado se refere a como o computador trata o dado armazenado, podendo ser um dado textual ou numérico, por exemplo.

Tabelas pivô ocorrem quando uma entidade se relaciona com ela mesma, ou quando possui um relacionamento de N para N com outra entidade, havendo a necessidade de criar-se outra tabela na implementação, para que este relacionamento fique armazenado no banco de dados.

Um exemplo disso pode ser visto na Figura 4.2, em que a entidade Preparação se relaciona com ela mesma para definir um relacionamento do tipo Opção Vegetariana, ou seja, uma preparação é uma opção vegetariana à outra preparação.

Figura 4.2 – Exemplo Tabela Pivô



² Modelo Entidade Relacionamento

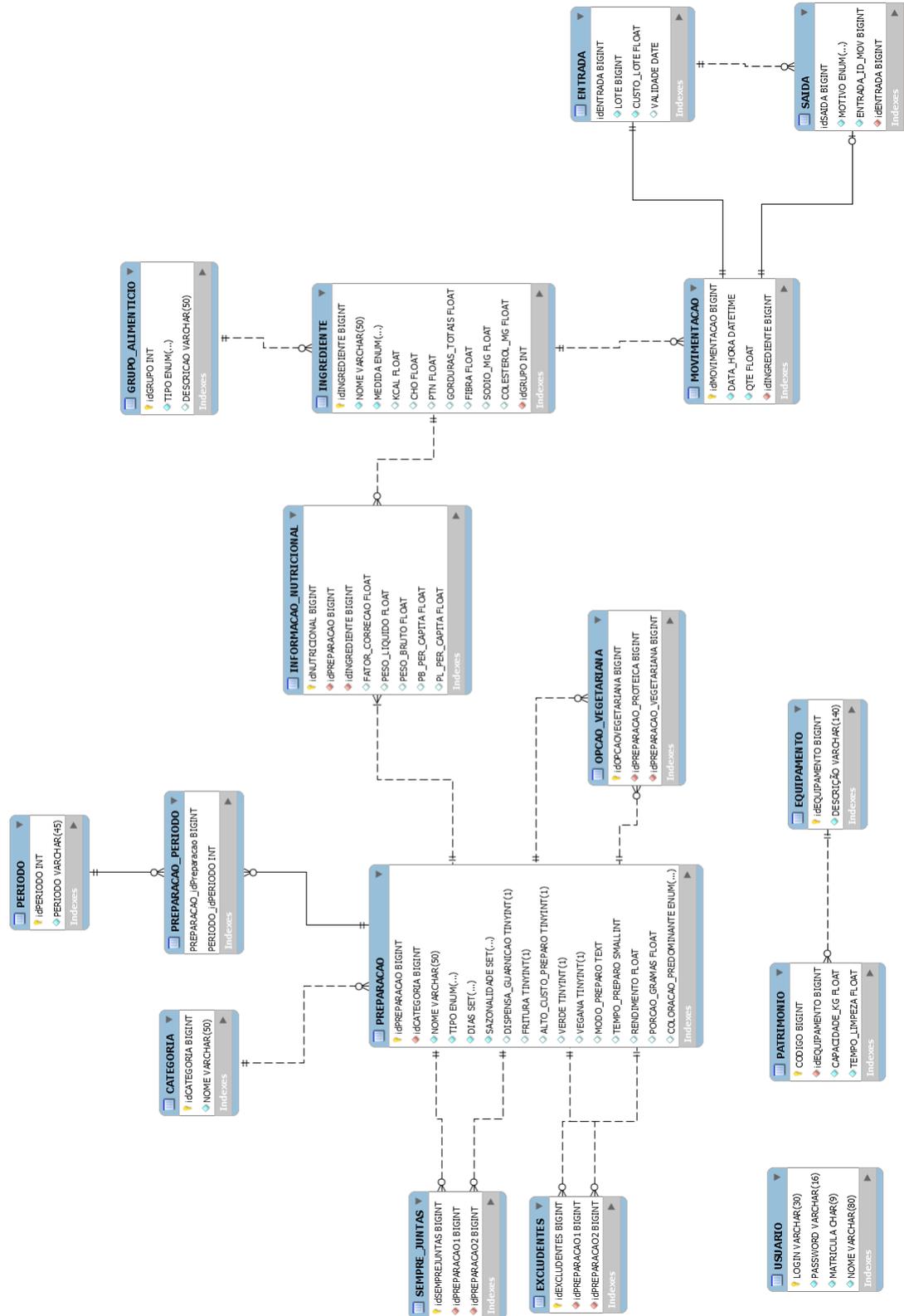
Pela Figura 4.2 observa-se que um losango sem preenchimento azul-claro, representa um atributo que pode ser nulo, ou seja, para a existência de tal entidade, este atributo não é necessário. O losango azul-claro preenchido representa um atributo que deve ser preenchido.

O símbolo de uma chave amarela representa uma chave primária daquela entidade, o que significa que é um atributo único para representação daquela entidade, podendo ser um número, por exemplo. A chave primária é um valor único daquela entidade, é a identificação principal daquele registro para o banco de dados.

O losango vermelho representa uma chave estrangeira, o que significa, no caso da Figura 4.2, que esta entidade possui um relacionamento N para 1 com esta outra entidade. No exemplo da Figura 4.2, a entidade PREPARAÇÃO deve possuir uma e somente uma CATEGORIA, representada pelo atributo idCATEGORIA.

Na Figura 4.3 está o diagrama entidade-relacionamento de nível lógico do sistema desenvolvido.

Figura 4.3 – Diagrama Entidade Relacionamento de nível lógico



4.3 Dicionário de dados

Apresentamos os dicionários de dados de duas tabelas do sistema, consideradas as principais, que descrevem a aplicação. Elas são consideradas principais por interagirem ou possuírem relacionamento com todas as outras tabelas, sendo, assim, o cerne do problema que o sistema se propôs solucionar.

Um dicionário de dados é um documento que descreve de forma estruturada o significado, origem, relacionamento e uso dos dados. Neste trabalho, cada tabela do dicionário do banco de dados possui 4 colunas, que descrevem seus atributos, sendo elas: nome, descrição, domínio e "permite nulo". Domínio diz respeito ao formato que o dado assumirá ao ser armazenado no computador, e "permite nulo" é uma informação que diz se tal atributo pode deixar de ser armazenado (ser NULL) ou não.

4.3.1 Tabela de Ingredientes

Tabela responsável por armazenar os ingredientes cadastrados no sistema. Um ingrediente possui um nome, quantidade em estoque, e outros valores que descrevem atributos nutricionais do mesmo (Ver Figura 4.4).

4.3.2 Tabela de Preparações

Tabela responsável por armazenar as preparações que são servidas no restaurante, e serão a base para integrar o cardápio de um dia no Restaurante Universitário (RU) (Ver Figura 4.5).

Figura 4.4 – Tabela de ingredientes

Tipo Entidade	INGREDIENTE		
Descrição	Alimento comum encontrado em uma receita do restaurante universitário		
Atributos			
Nome	Descrição	Domínio	Permite Nulo? (S/N)
Nome	Atributo Chave para o nome do ingrediente	Cadeia de até 50 caracteres	N
Quantidade em Estoque	Atributo Derivado para a quantidade em estoque do ingrediente	Um número real	-
Índice Kcal	Atributo Simples para armazenar o índice de calorias do ingrediente	Um número real	S
Índice Cho	Atributo Simples para armazenar o índice de carboidratos do ingrediente	Um número Real	S
Índice PTN	Atributo Simples para armazenar o índice de proteínas do ingrediente	Um número Real	S
Índice LIP ou Gorduras Totais	Atributo Simples para armazenar o índice de lipídeos do ingrediente (GORDURAS TOTAIS)	Um número Real	S
Índice Fibras	Atributo Simples para armazenar o índice de Fibras do ingrediente	Um número Real	S
Índice Sódio	Atributo Simples para armazenar o índice de Sódio do ingrediente	Um número Real	S
Índice Colesterol	Atributo Simples para armazenar o índice de Colesterol do ingrediente	Um número Real	S
Custo Médio	Atributo Derivado para armazenar o preço do ingrediente por unidade de medida	Um número Real	-

Figura 4.5 – Tabela de preparações

Tipo Entidade	PREPARAÇÃO		
Descrição	Receita utilizada no restaurante universitário		
Atributos			
Nome	Descrição	Domínio	Permite Nulo? (S/N)
Nome	Atributo Chave para o nome da Preparação	Cadeia de até 50 caracteres	N
Modo Preparo	Atributo Simples para armazenar um texto descritivo explicando o passo a passo da receita	Texto	N
Tempo Preparo	Atributo Simples para armazenar o tempo em minutos gasto para preparar a receita	Um número inteiro	N
Rendimento	Atributo Simples para armazenar o rendimento em gramas da preparação	Um número real	N
Porção em gramas	Atributo Simples para armazenar o peso de uma porção em gramas	Um número Real	N
Coloração predominante	Atributo enumerado para armazenar a coloração da preparação	LARANJA-AMARELO, LARANJA, VERMELHO, VERDE-AMARELO, VERDE, MARROM, VERDE-BRANCO ou VERMELHO-ROXO	N
Sazonalidade	Atributo Simples para armazenar os meses em que dada preparação não pode ser feita. Sendo cada número referente a um mês	Um conjunto de números entre 1 e 12 (inclusos), sem repetição	N
Dias	Atributo para descrever os dias da semana em que dada preparação pode ser feita. Sendo cada número referente a um dia, começando por Domingo.	Um conjunto de números entre 1 e 7 (inclusos), sem repetição	N

4.4 Consultas criadas para a utilização do sistema

Apresentamos aqui alguns trechos de código, da linguagem SQL, para demonstrar como são realizadas as consultas ao Sistema Gerenciador de Banco de Dados (SGBD).

4.4.1 Busca por todos os ingredientes

Esta busca tem a função de listar todos os ingredientes cadastrados. O Código 4.1 mostra sua implementação. A cláusula **SELECT** do SQL descreve quais atributos devem ser exibidos na consulta. Neste caso, como logo após a cláusula temos um * (asterisco), significa que queremos que todos os campos daquela tabela sejam selecionados e exibidos na consulta. Logo após isso, temos a cláusula **FROM** que diz o nome da tabela em que esta consulta será realizada, neste caso, será na tabela "ingrediente".

Código 4.1 – Busca por todos ingredientes

```
1  SELECT * FROM ingrediente;
```

Fonte: Gustavo Sousa

4.4.2 Busca por preparações

As buscas por preparações são bem mais complexas de serem feitas, pois elas são divididas em 4 categorias, sendo elas: preparações proteicas, preparações vegetarianas, saladas e guarnições.

Para cada uma das 4 categorias de preparações foi necessário criar uma *Stored Procedure* no SGBD. Uma *Stored Procedure*, ou procedimento armazenado, é um trecho de código em SQL que pode ser armazenado para futuro uso. Este trecho de código pode possuir parâmetros, como uma função computacional ou matemática, por exemplo, que podem modificar valores específicos do código e assim gerar resultados diferentes para cada chamada do procedimento.

O Código 4.2 mostra o trecho de uma *Stored Procedure* para selecionar as preparações que pertencem ao grupo de guarnições. Algumas cláusulas a serem notadas: a cláusula **CASE WHEN**, que aparece pela primeira vez na linha 8, serve para criar uma condicional, em que se o resultado for verdadeiro, o valor depois da cláusula **THEN** é retornado, caso contrário o valor da cláusula

ELSE é retornado; A cláusula **END** quando posta depois de um **CASE WHEN** serve para indicar o término da verificação condicional.

As cláusulas **BEGIN** na linha 1, e **END** na linha 34 servem para delimitar o trecho de código que consiste na *Stored Procedure*.

A cláusula **WHERE** serve para criar restrições (ou filtros) à consulta, em que só serão exibidos os resultados que cumprirem estas condições. Na linha 29, por exemplo, nota-se que o tipo da preparação deve ser igual a "Guarnicao".

Na linha 33 do Código 4.2, nota-se a cláusula **LIMIT**, seguida de "índice" e "qtd". Esta cláusula serve para criar a paginação dos resultados. Tanto "índice" quanto "qtd" são parâmetros que são passados à *Stored Procedure* a cada invocação. O "índice" indica qual o índice do primeiro resultado que deve ser mostrado no resultado final, e "qtd" é um número que define quantos resultados serão mostrados a partir da contagem do índice.

Código 4.2 – Selecionar todas as guarnições

```

1  BEGIN
2    SELECT
3      preparacao.idPREPARACAO,
4      preparacao.nome, categoria.nome AS categoria,
5      GROUP_CONCAT(periodo.PERIODO SEPARATOR ', ')
6      AS Período,
7      preparacao.DIAS, preparacao.sazonalidade,
8      CASE WHEN preparacao.FRITURA <> 0
9        THEN 'Sim'
10       ELSE 'Nao'
11     END AS Fritura,
12     CASE WHEN preparacao.ALTO_CUSTO_PREPARO <> 0
13       THEN 'Sim'
14       ELSE 'Nao'
15     END AS ALTO_CUSTO_PREPARO ,

```

```

16     CASE WHEN preparacao.VERDE <> 0
17         THEN 'Sim'
18         ELSE 'Nao'
19     END AS Verde,
20     preparacao.MODO_PREPARO, preparacao.TEMPO_PREPARO,
21     preparacao.COLORACAO_PREDOMINANTE,
22     preparacao.RENDIMENTO, preparacao.PORCAO_GRAMAS
23 FROM
24     preparacao, categoria, preparacao_periodo, periodo
25 WHERE
26     preparacao.idCATEGORIA=categoria.idCATEGORIA
27     AND preparacao_periodo.PREPARACAO_idPreparacao =
28         preparacao.idPREPARACAO
29     AND preparacao_periodo.PERIODO_idPERIODO=periodo.idPERIODO
30     AND tipo='Guarnicao'
31 GROUP BY preparacao.nome
32 ORDER BY preparacao.nome
33 LIMIT indice, qtd;
34 END

```

Fonte: Gustavo Sousa

4.4.3 Buscar por preparações excludentes

Uma das regras do sistema, como pôde ser visto na definição dos diagramas de banco de dados, é que cada preparação pode possuir um relacionamento com outra preparação, sendo este relacionamento classificado como "excludente", "sempre juntas" e "opção vegetariana".

Foi criada um *Stored Procedure* para cada uma das 3 categorias citadas anteriormente. Para exemplificação, o Código 4.3 mostra a busca por uma delas: as preparações excludentes.

Código 4.3 – Selecionar preparação excludente

```
1 BEGIN
2   SELECT p2.NOME FROM excludentes
3   JOIN preparacao p1 ON p1.idPREPARACAO = excludentes.idPREPARACAO1
4   JOIN preparacao p2 ON p2.idPREPARACAO = excludentes.idPREPARACAO2
5   WHERE excludentes.idPREPARACAO1 = idPrep;
6 END
```

Fonte: Gustavo Sousa

No Código 4.3, o parâmetro passado ao procedimento é "idPrep", e este código retornará todas as preparações que possuem um relacionamento de "excludentes" com a preparação passada por parâmetro. Por ser uma consulta que usa a mesma tabela 2 vezes (Preparação), vemos a aparição de *aliases* ou apelidos, que é uma maneira de identificar a qual das duas tabelas o código refere-se naquele momento, em caso de haver repetição da mesma. Neste caso, a tabela "preparacao" toma forma dos apelidos "p1" e "p2" (linhas 3 e 4).

4.5 Códigos das telas

A tela de boas vindas, que poderá ser vista na seção Estudo de Caso, foi implementada com o código descrito na Figura 4.6. O código em questão foi produzido em HTML e CSS e define apenas a tela inicial do sistema. Podemos ver *tags* que são delimitadores no HTML, escritas com as notações < e >. A tag <title> por exemplo, serve para dizer qual o título da página em questão. Na Figura 4.6 podemos ver na linha 4 que o título da página é "Gerauca", nome dado ao sistema.

Para exemplificar o uso do PHP, a linguagem de programação que foi utilizada, e que é executada no servidor da aplicação, pode-se ver o exemplo do código da tela de listar ingredientes do sistema, que poderá ser vista mais à frente no capítulo de Estudo de Caso, deste trabalho.

Neste trecho de código, Figura 4.7, nota-se o uso de condicionais como o **if** na linha 133, muito comum em linguagens de programação. Uma peculiaridade do PHP é que as *variáveis* são

Figura 4.6 – Código da tela de boas vindas

```

boasVindas.php X
boasVindas.php > ...
1 <!DOCTYPE html>
2 <html lang="pt">
3 <head>
4 <title>Gerauca</title>
5 <meta charset="utf-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1">
7
8 <!-- Compiled and minified JS/CSS Bootstrap-->
9 <style>
10 .image {
11     position: absolute;
12     top: 50%;
13     left: 50%;
14     width: 120px;
15     height: 120px;
16     margin:-60px 0 0 -60px;
17     -webkit-animation:spin 4s linear infinite;
18     -moz-animation:spin 4s linear infinite;
19     animation:spin 4s linear infinite;
20 }
21 @-moz-keyframes spin { 100% { -moz-transform: rotate(360deg); } }
22 @-webkit-keyframes spin { 100% { -webkit-transform: rotate(360deg); } }
23 @keyframes spin { 100% { -webkit-transform: rotate(360deg); transform:rotate(360deg); } }
24 </style>
25
26 </head>
27
28 <body>
29 
30 </body>
31 </html>
32

```

sempre acompanhadas do pré-fixo \$ (cifrão). Comandos como *echo* fazem com que algo seja impresso na tela.

Dentro das aspas em algumas linhas há um código similar ao HTML e isso acontece porque o PHP faz com que as páginas HTML sejam geradas dinamicamente baseadas nos dados presentes no banco de dados naquele momento.

Na linha 138 há o uso de *funções*, outro termo muito comum em linguagens de programação. Neste caso, a função *mysqli_fetch_assoc* faz conexão com o banco de dados e executa a consulta SQL que está armazenada na variável *result*.

Outras funcionalidades, como laços de repetições, também podem ser observadas. Na linha 138 há o uso de palavra-chave *while*.

Figura 4.7 – Código da tela de listagem de ingredientes

```
133 if (!$result) {
134     echo "<style>td{text-align: center}</style><tr><td colspan=\"9\">(sem resultados)</tr></td>";
135 } else if ($result->num_rows === 0) {
136     echo "<style>td{text-align: center}</style><tr><td colspan=\"9\">(sem resultados)</tr></td>";
137 } else if ($result) {
138     while ($row = mysqli_fetch_assoc($result)) {
139         echo '<tr id="tr' . $row['idINGREDIENTE'] . '">';
140         echo "<td>" . $row['NOME'] . "</td>";
141         echo "<td>" . $row['MEDIDA'] . "</td>";
142         echo "<td>" . $row['KCAL'] . "</td>";
143         echo "<td>" . $row['CHO'] . "</td>";
144         echo "<td>" . $row['PTN'] . "</td>";
145         echo "<td>" . $row['GORDURAS_TOTAIS'] . "</td>";
146         echo "<td>" . $row['FIBRA'] . "</td>";
147         echo "<td>" . $row['SODIO_MG'] . "</td>";
148         echo "<td>" . $row['COLESTEROL_MG'] . "</td>";
149         echo "<td>" . $row['TIPO'] . "</td>";
150         echo "</tr>";
151     }
152 }
```

5 ESTUDO DE CASO

O estudo de caso consiste na implementação de um sistema *web* que forneça uma interface gráfica para o usuário interagir com os dados, podendo visualizar, criar, editar e excluir nas diversas telas que serão mostradas a seguir, certificando-se que o sistema cumpra com os requisitos acordados, até então, entre as partes envolvidas.

5.1 Atividades realizadas

As atividades realizadas durante a vigência da bolsa compreenderam o período de 01/10/2019 à 31/12/2020, totalizando um ano e três meses. Como exigência do programa, foram realizadas 12 horas de atividades por semana, resultando em 48 horas de atividades por mês e 720 horas no total.

5.1.1 Outubro de 2019 a Novembro de 2019

Os dois primeiros meses foram utilizados para reuniões com os bolsistas anteriores e com o orientador para entendimento do problema, reuniões com a nutricionista encarregada para estabelecer novas metas para o projeto, reuniões com a equipe de desenvolvimento do programa de geração de cardápios para determinação de compatibilidade entre os projetos.

5.1.2 Dezembro de 2019

O terceiro mês foi utilizado para estudo das tecnologias que foram usadas.

5.1.3 Janeiro de 2020

No quarto mês foi feita a migração da hospedagem do sistema, do GitLab (GITLAB, 2022) para o GitHub (GITHUB, 2022), e a execução do servidor *web* em máquina local para entendimento do que já havia sido feito até então.

5.1.4 Fevereiro de 2020

Foi refeita a modelagem conceitual de alto nível do banco de dados, produzindo diagramas de entidade-relacionamento.

5.1.5 Março de 2020

Foi refeita a modelagem lógica do banco de dados, definindo tipos e atributos das entidades, produzindo várias versões do diagrama lógico que foram alteradas ao longo do tempo, a pedido do orientador em reuniões realizadas.

5.1.6 Abril de 2020

Foi desenvolvido o *script* SQL de geração do banco de dados para o MariaDB, corrigindo erros de incompatibilidade do MySQL (tecnologia que estava sendo usada) com o MariaDB (nova tecnologia adotada). No processo houve dificuldades em adaptar o *script* gerado automaticamente pelo MySQL Workbench ao MariaDB.

5.1.7 Maio de 2020

Atualizado o documento de Dicionário de Dados referente às novas alterações feitas no banco de dados e a criação de procedimentos SQL (*stored procedures*) para listagem de diferentes tipos de refeições.

5.1.8 Junho de 2020

Foi feita a refatoração das páginas de cadastro das preparações, adequando as interfaces aos novos modelos do banco de dados, atualizando o código PHP do *back-end* e HTML do *front-end*. Melhorias na segurança do sistema, bloqueando algumas rotas que podiam ser acessadas sem ter feito *login*.

5.1.9 Julho de 2020

Foram adequadas as interfaces de cadastro de preparações para exibição dinâmica do conteúdo do banco de dados, mostrando na tela automaticamente as mudanças feitas no banco de dados do sistema.

5.1.10 Agosto de 2020

Foram preparadas novas telas para cadastro de excludentes, sempre juntas e opções vegetarianas. Estas categorias são restrições que foram estabelecidas para o problema nas etapas de reuniões feitas anteriormente com os orientadores.

5.1.11 Setembro de 2020

Neste período, o andamento do projeto estava atrasado, e faltavam muitas atividades que deveriam ser realizadas pela definição da fase de planejamento. Entretanto, noticiou-se a prorrogação da bolsa que teria fim em outubro, para término em dezembro. Com isso, ganhou-se um tempo extra para a execução do projeto e foi produzido um cronograma de atividades a serem realizadas até a data final. Foram feitas reuniões com toda a equipe para apresentação do que foi feito, e definição de novas metas finais.

5.1.12 Outubro de 2020

Finalização da regra de negócio de cadastro das restrições feitas em agosto. Foram implementadas telas para a edição de preparações, que até então só haviam telas para cadastro e exclusão das mesmas.

5.1.13 Novembro de 2020

Foi atualizada a página de cadastro de ingredientes contemplando as novas mudanças realizadas no banco de dados. Foi criada paginação nas telas de listagem de preparações.

5.1.14 Dezembro de 2020

Foi criada a tela e regra de negócio para edição de ingredientes. Criado também o código de exportação do banco de dados do sistema, que é o arquivo de entrada para o algoritmo de geração automática de cardápios.

5.2 Sistema na prática

Para criar um sistema *web* que cumprisse as necessidades do projeto realizado, foi necessário implementar alguns requisitos funcionais e não-funcionais. Requisitos funcionais são aqueles que definem o cerne da aplicação, o que vai resolver de fato o problema do usuário final do sistema. Já os requisitos não-funcionais são aqueles implícitos na criação da aplicação, ou seja, eles não são ditos ou requisitados pelo cliente mas precisam ser implementados para o bom funcionamento do sistema.

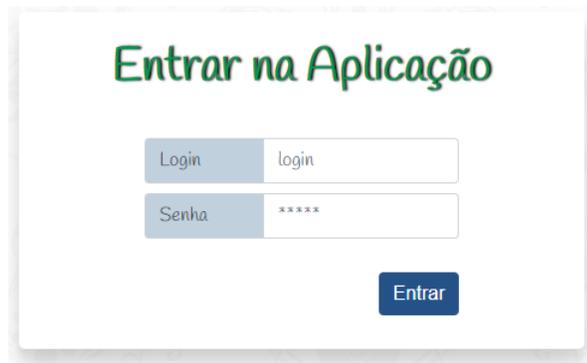
Para a tela de boas vindas, é apresentada uma interface simples onde o usuário pode apenas efetuar *login* ou *logout* do sistema (Figura 5.1).

Figura 5.1 – Tela de boas vindas do sistema



Na tela de *login* são solicitados, ao usuário, um nome e senha para autenticar-se na plataforma (Figura 5.2).

Figura 5.2 – Tela de login

A imagem mostra a tela de login da aplicação. No topo, o título "Entrar na Aplicação" está escrito em uma fonte verde arredondada. Abaixo dele, há dois campos de entrada: "Login" com o texto "login" e "Senha" com caracteres ocultos por pontos. Um botão azul "Entrar" está posicionado à direita dos campos.

O menu no canto superior esquerdo (Figura 5.3) do sistema dá acesso a algumas páginas: *homepage*, página de contatos, ingredientes, preparações e exportar.

Figura 5.3 – Menu de acesso



A página de ingredientes fornece uma listagem com todos os ingredientes cadastrados na plataforma. Logo abaixo do título há uma barra de busca para facilitar ao usuário encontrar algum ingrediente específico. Acima da listagem há um botão para cadastrar novos ingredientes. Ao lado de cada ingrediente há um botão para editá-lo e outro para excluí-lo do sistema. (Figura 5.4)

Figura 5.4 – Tela de listagem de ingredientes

The screenshot shows a web application interface for consulting ingredients. At the top, there is a navigation bar with a menu icon, the logo 'Gerauca', and a 'Sair' button. The main heading is 'Consultar Ingredientes'. Below the heading is a search bar with the placeholder text 'Busca Ingrediente' and a search icon. A blue button labeled 'Cadastrar' is positioned above the table. The table itself has 11 columns: 'Nome', 'Medida', 'KCal', 'CHO', 'PTN', 'Lipideos', 'Fibra', 'Sódio', 'Colesterol', and 'Grupo'. Each row represents an ingredient, and the 'Grupo' column for all items is 'REGULADORES'. To the right of each row are two icons: a blue pencil for editing and a red 'X' for deletion. At the bottom of the table, there are pagination controls: 'Primeira', '1', '2', '3', and 'Última'.

Nome	Medida	KCal	CHO	PTN	Lipideos	Fibra	Sódio	Colesterol	Grupo		
Abóbora	KG	310	622	793	51	565	486	17	REGULADORES		
Abobrinha	KG	827	729	512	710	835	867	124	REGULADORES		
Acelga	KG	23	296	410	46	359	652	982	REGULADORES		
Agrião	KG	186	308	762	963	990	951	936	REGULADORES		
Alcachofra	KG	277	247	119	300	495	551	694	REGULADORES		
Alface	KG	237	466	605	215	313	775	811	REGULADORES		
Alho	KG	246	599	476	894	891	103	254	REGULADORES		
Alho-poró	KG	544	14	779	416	348	250	165	REGULADORES		
Almeirão	KG	714	108	993	639	786	515	909	REGULADORES		
Berinjela	KG	992	864	154	597	994	278	947	REGULADORES		
Bertalha	KG	500	111	387	360	312	406	418	REGULADORES		
Beterraba branca (acelga)	KG	377	258	960	20	946	152	586	REGULADORES		
Bredo	KG	114	605	669	785	284	747	2	REGULADORES		
Brócolis	KG	872	919	949	32	559	535	869	REGULADORES		
Broto de alfafa	KG	367	272	428	440	681	181	85	REGULADORES		
Broto de feijão	KG	386	198	145	99	958	898	930	REGULADORES		
Cariru	KG	4	716	611	645	769	526	138	REGULADORES		
Caruru	KG	688	833	906	729	181	981	46	REGULADORES		
Catalonha	KG	641	45	738	489	447	777	50	REGULADORES		
Cebola	KG	974	472	232	103	758	276	679	REGULADORES		

Ao clicar em "cadastrar" na tela de listagem de ingredientes, ou pelo menu de acesso, o usuário é redirecionado à tela de cadastro de um novo ingrediente (Figura 5.5), podendo preencher todos os campos necessários. Caso clique no ícone de editar o ingrediente (símbolo de uma caneta azul), o usuário é levado a uma tela de cadastro com os dados daquele ingrediente já preenchidos, de forma que é possível alterá-los e depois salvá-los.

Figura 5.5 – Tela de cadastro de ingrediente

The screenshot shows a web interface for registering an ingredient. At the top, there is a dark blue header with a menu icon, the logo 'Gerauca', and a 'Sair' (Logout) button. The main content area has a light green background with a decorative floral pattern. The title 'Cadastrar Ingrediente' is centered in a large, green, sans-serif font. Below the title, there are several input fields and buttons:

- Nome:** A text input field containing the word 'Ingrediente'.
- Medida:** A dropdown menu currently set to 'L'.
- Grupo:** A dropdown menu currently set to 'ENERGÉTICOS'.
- Nutritional Values:** A grid of input fields for various nutrients, all currently set to '0,00':
 - Kcal/100g
 - Proteína/100g
 - Lípidos/100g
 - Fibras/100g
 - CHO/100g
 - Sódio/100g
 - Colesterol/100g
- Buttons:** At the bottom right, there are two buttons: 'Consultar Registro' (with a magnifying glass icon) and 'Cadastrar'.

Na tela de Preparações (Figura 5.6), o usuário pode consultar todas as preparações cadastradas no sistema, por tipo. Existem abas na parte superior da tabela para exibição destes 4 tipos, sendo eles: Proteica, Guarnição, Salada e Vegetariana. Ao clicar sobre cada uma delas, a listagem muda para exibir as preparações de acordo. A tabela dispõe de algumas informações mais pertinentes ao problema em questão, como *Categoria*, *Período*, *Dias*, *Sazonalidade*, *Dispensa Guarnição*, *Fritura*, e *Alto Custo de Preparo*. *Período* diz respeito a qual período do dia dada preparação poderá ser feita, podendo ser almoço, janta ou ambos. *Dias* mostra quais dias da semana que aquela preparação pode ser feita. *Sazonalidade* diz respeito aos meses em que a preparação pode ser feita. As 3 últimas colunas exibem informações extras sobre o preparo, que podem ser "sim" ou "não".

Figura 5.6 – Tela de listagem de preparações

Geravca Sair

Listagem de Preparações

[+ Cadastrar](#)

Proteicas			Guarnições		Saladas			Vegetarianas			
Nome	Categoria	Período	Dias	Sazonalidade	Dispensa Guarnição?	Fritura?	Alto Custo de Preparo?				
Bife à milanesa	Seca	Janta	DOM, TER, SEX	JAN, FEV, MAR, ABR, MAI, SET, OUT, NOV	Sim	Sim	Não				
Bife a rolê	Refogado	Almoço	DOM, TER, QUI, SAB	ABR, MAI, JUN, AGO, SET, NOV	Não	Não	Não				
Bife a rolê de frango ao molho	Massa	Almoço	DOM, SAB	FEV, ABR, JUN, SET, NOV, DEZ	Não	Sim	Sim				
Bife de panela	Bovina	Almoço	DOM, SEG, QUA, QUI, SAB	JAN, FEV, MAR, ABR, JUN, JUL, AGO, NOV	Não	Não	Sim				
Bife de panela com ervilha	Bovina	Almoço	TER, QUA, QUI, SAB	FEV, MAR, MAI, NOV	Sim	Não	Sim				
Bife de pernil	Bovina	Almoço	QUA	JAN, MAR, JUN, JUL, AGO, OUT, NOV	Sim	Não	Sim				
Bife na Chapa	Legume	Almoço	DOM, QUI, SEX	FEV, MAR, ABR, MAI, JUN, JUL, OUT, NOV, DEZ	Não	Sim	Sim				
Bobó de frango	Bovina	Janta	SEG, TER, QUI	JAN, ABR, MAI, JUL, AGO, NOV, DEZ	Não	Não	Não				
Carne com batata	Bovina	Almoço	DOM, QUI	JAN, MAI, SET, OUT, DEZ	Não	Não	Não				
carne em cubos ao molho	Massa	Almoço	DOM, QUI, SEX, SAB	MAR, ABR, MAI, JUL, NOV, DEZ	Não	Não	Sim				
Carne em Cubos com Azeitona	Porco	Almoço	DOM, SEG, TER, QUI	MAR, ABR, JUN, JUL, AGO, OUT, DEZ	Não	Não	Não				
Carne em iscas com batata fúza e brócolis	Porco	Almoço	TER, QUI	JAN, MAR, ABR, JUN, JUL, AGO, SET, DEZ	Sim	Não	Sim				
Carne louca	Caldo	Almoço	DOM, SEG, SEX	JAN, FEV, MAR, ABR, MAI, JUN, JUL, SET, DEZ	Sim	Sim	Não				

[Primeira](#)
[1](#)
[2](#)
[3](#)
[Última](#)

Ao lado de cada preparação há 4 ícones. O primeiro deles serve para exibir mais informações sobre aquela preparação (Figura 5.7). O segundo ícone leva o usuário à tela de edição daquela preparação (Figura 5.8), parecida com a edição de ingredientes, em que a tela é idêntica à tela de cadastro, porém com os dados já preenchidos e que podem ser alterados. O terceiro leva à tela de cadastro de restrições daquela preparação (mais sobre isto à frente). E o quarto botão serve para excluir a preparação, de modo que, ao clicar nele, é exibida uma janela de confirmação da exclusão (Figura 5.9), para melhor usabilidade do sistema.

Figura 5.7 – Tela de informações extras da preparação

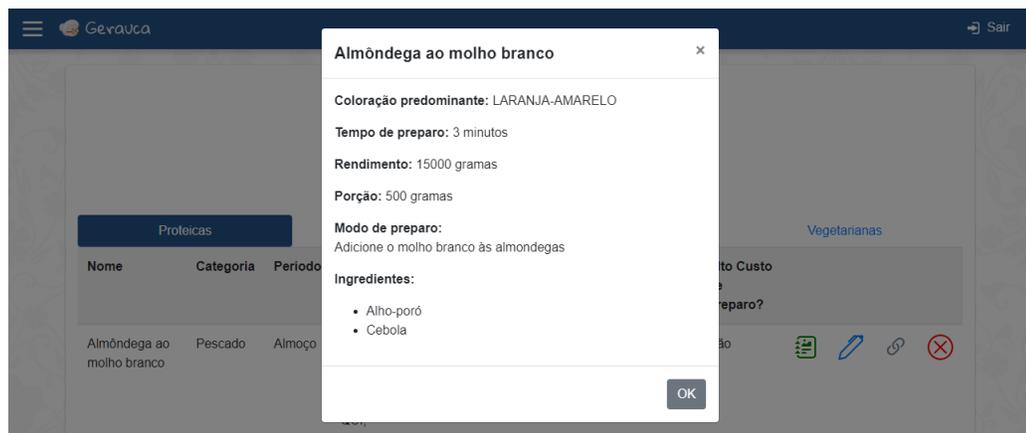


Figura 5.8 – Tela de edição da preparação

Editar Preparação

Nome: Almondega ao molho branco | Tipo: Proteica

Coloração Predominante: LARANJA-AMARELO | Categoria: Pescado

Fritura | Alto custo de preparo | Verde
 Exige opção vegana | Dispensa Guarnição

Período: Almoço | Janta

Dia: domingo | segunda-feira | terça-feira
 quarta-feira | quinta-feira | sexta-feira
 sábado

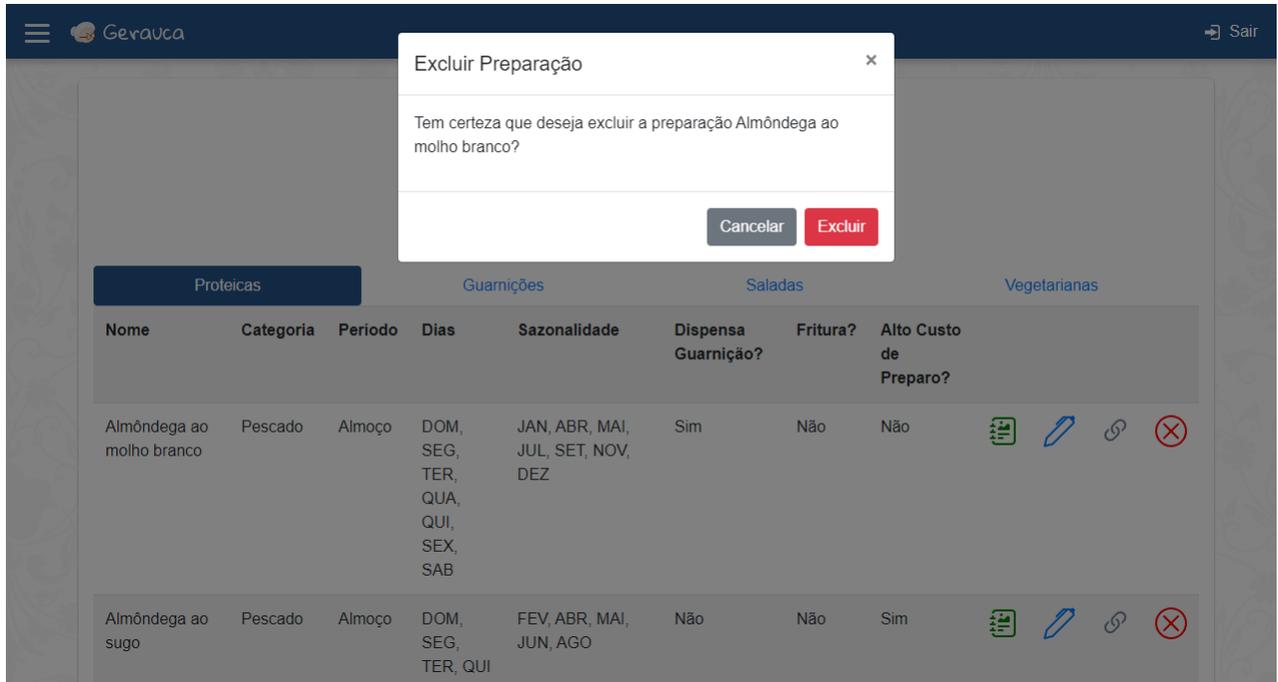
Mês: Janeiro | Fevereiro | Março
 Abril | Maio | Junho
 Julho | Agosto | Setembro
 Outubro | Novembro | Dezembro

Tempo de Preparo: 3 | Rendimento: 15000 | Porção/Gramas: 500

Modo de Preparo: Adicione o molho branco às almondegas

[Consultar Registro](#) [Atualizar](#)

Figura 5.9 – Tela de confirmação de exclusão da preparação



Ao clicar sobre um símbolo de correntes cinzas, o usuário é levado à tela de cadastro de excludentes (Figura 5.10) daquela preparação. Nesta tela, existem 3 abas, em que o usuário pode alternar entre os tipos de restrições que quer cadastrar para aquela preparação. Sendo elas: "Excludentes", "Sempre Juntas" e "Opções Prot-Vege". Como explicado anteriormente, estas restrições são pertinentes à geração automática de cardápios, e são consideradas diretamente no modelo matemático proposto.

Figura 5.10 – Tela de cadastro de excludentes da preparação

Cadastro de Excludentes

Excludentes Sempre juntas Opções Prot-Vege

Almôndega ao molho branco

↓

carne em cubos ao molho

+ X

Voltar Atualizar

SIGA A PRAEC

A PRAEC
 Sobre
 Equipe
 Regimento Interno
 Organização Administrativa
 Comunicação PRAEC

COORDENADORIAS
 Acessibilidade
 Alimentação
 Diversidade e Diferenças
 Esporte e Lazer
 Moradia
 Programas Sociais
 Saúde

CONTATO
 Pró-Reitoria de Assuntos Estudantis e Comunitários
 Caixa Postal 3037 - CEP 37200-000
 Lavras, Minas Gerais, Brasil
 (35) 3829-1132
 praec@praec.ufa.br

PRAEC
 PROGRAMA DE ACESSIBILIDADE
 ESTUDANTIL E COMUNITÁRIO

UFPA
 UNIVERSIDADE FEDERAL DO PARÁ

As 3 telas são iguais em seu *layout*, contendo um botão com símbolo de + (mais) para adicionar a restrição em questão, e outro com símbolo de - (menos) para excluir aquela restrição. A tela conta também com um botão de "Atualizar" para persistir as alterações feitas, e um botão de "Voltar" que leva o usuário de volta à tela de listagem das preparações.

6 CONCLUSÃO

O projeto realizado juntamente com o professor orientador e toda a equipe envolvida foi um momento de grande importância na vida acadêmica e profissional do discente. As atividades realizadas já se provam ter grande impacto na carreira do estudante que aplica muito do que viu e aprendeu, neste projeto, em suas empreitadas atuais como programador. Muitos assuntos estudados no curso de graduação não teriam surtido o mesmo efeito de aprendizado caso o estudante não pudesse ter uma chance de aplicá-los em um projeto real e importante, para si e para os docentes envolvidos.

Destaca-se que o foco do projeto foi a construção de um banco de dados, juntamente com um sistema *web*, habilidades das quais o mercado de trabalho em Tecnologia da Informação (TI) urge por novos profissionais capacitados com estas competências.

Apesar das vantagens, vale ressaltar que muitas dificuldades foram enfrentadas ao longo do caminho, das quais deixa-se sugestões de possíveis melhorias ao projeto. No início do projeto o estudante sentiu dificuldade muito grande para o entendimento do intuito do projeto e obtenção de familiaridade com a estrutura do sistema feito até então, isto se reflete na baixa produtividade do período inicial. Houve também certa confusão por parte dos orientadores para definir a divisão do projeto aos bolsistas, de forma que inicialmente pensou-se até que também seria desenvolvido o algoritmo. Além disso, a mudança de bolsista durante o projeto da produção do algoritmo dificultou o andamento do desenvolvimento, pois exigiu-se novas reuniões para esclarecimento do problema.

As mudanças repentinas dos requisitos, fosse por inabilidade comunicativa por parte dos encarregados de sua coleta, do entendimento do discente sobre o problema, ou dos pedidos do requerente também fizeram com que uma porção do trabalho fosse descartado, acarretando tempo perdido de produtividade.

Algumas atividades que poderão ser realizadas no futuro, são:

- modificar a forma de acesso (*login*) ao sistema, utilizando o *login* de acesso unificado da UFLA;

- modificar a tela de cadastro de preparações para corrigir o defeito visual do preenchimento do formulário de ingredientes, em que o nome do ingrediente continua aparecendo mesmo depois do usuário pressionar enter ou clicar no nome sugerido;
- modificar a tela de cadastro de preparações para que ela seja dinâmica, de forma que na hora de cadastrar cada uma das 4 categorias de preparação (proteica, guarnição, vegetariana, salada), o formulário mude automaticamente para exibir apenas os campos pertinentes a cada uma delas;
- refatoração do código, analisando possíveis códigos que podem ser descartados e que permanecem na pasta do projeto, necessitando uma varredura minuciosa para tal; e
- modificar as telas de cadastro de excludentes, sempre juntas, e opções vegetarianas para que seja possível excluir uma linha preenchida anteriormente sem que se perca a informação contida nas linhas abaixo. Atualmente é necessário excluir de baixo para cima para que o formulário funcione. Entretanto, o sistema não permite o cadastro com linhas vazias, por isso o risco é apenas de um trabalho desnecessário para o usuário, e não perda de informações.

O resultado do projeto é um sistema funcional para cadastro e manipulação dos dados necessários para a geração automática de cardápios, satisfatório para a solução do problema proposto. O sistema atual já pode ser utilizado e cumpre bem o papel de auxílio à manipulação dos dados que, até então, é realizada de forma dificultosa e custosa através de planilhas pela equipe de nutrição do restaurante universitário.

Por fim, volta-se a destacar que foi extremamente satisfatório ao discente participar do projeto, mesmo diante dos problemas enfrentados. O empenho de todos os participantes tornou a experiência importante e agradável para a formação acadêmica e profissional do autor deste trabalho.

REFERÊNCIAS

DB-ENGINES Raking. 2022. (Acessado em 12/03/2022). Disponível em: <<https://db-engines.com/en/ranking>>.

GITHUB. 2022. (Acessado em 12/03/2022). Disponível em: <<https://github.com/>>.

GITLAB. 2022. (Acessado em 12/03/2022). Disponível em: <<https://about.gitlab.com/>>.

MARIADB. 2021. (Acessado em 12/03/2022). Disponível em: <<https://mariadb.org/>>.

ORACLE. 2022. (Acessado em 12/03/2022). Disponível em: <<https://www.oracle.com/br/>>.

PETER Chen. 1976. (Acessado em 12/03/2022). Disponível em: <https://edisciplinas.usp.br/pluginfile.php/1769466/mod_resource/content/1/p9-chen.pdf>.