

UNIFEOB
CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO
OCTÁVIO BASTOS
ESCOLA DE NEGÓCIOS
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
GESTÃO DE TECNOLOGIA DA INFORMAÇÃO

PROJETO INTEGRADO
SISTEMA EMPRESARIAL

SÃO JOSÉ DOS CAMPOS, SP

ABRIL 2023

UNIFEOB
CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO
OCTÁVIO BASTOS
ESCOLA DE NEGÓCIOS
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
GESTÃO DE TECNOLOGIA DA INFORMAÇÃO

PROJETO INTEGRADO
SISTEMA EMPRESARIAL

MÓDULO DESENVOLVIMENTO DESKTOP

Banco de Dados – Prof. Sidney Gitcoff Telles

Programação Orientada a Objeto – Prof. Sidney Gitcoff Telles

Projeto de Desenvolvimento Desktop – Prof. Sidney Gitcoff Telles

Estudantes:

Júlio César Aguiar, 1012021200357

Sabrina Abreu Azuma, 1012021200358

SÃO JOSÉ DOS CAMPOS, SP
ABRIL, 2023

Sumário

INTRODUÇÃO.....	4
2 DESCRIÇÃO DA EMPRESA	5
3 PROJETO DE CONSULTORIA EMPRESARIAL	6
3.1 BANCO DE DADOS.....	6
3.1.1 MER - MODELAGEM ENTIDADE RELACIONAMENTO	6
3.1.2 DER - DIAGRAMA ENTIDADE RELACIONAMENTO.....	7
3.1.3 FÍSICO	7
3.2 PROGRAMAÇÃO ORIENTADA A OBJETOS.....	8
3.2.1 DIAGRAMA DE CLASSES	8
3.2.2 CÓDIGOS DO SISTEMA.....	9
3.2.3 IMAGENS DO SISTEMA	12
4 CONCLUSÃO	16
REFERÊNCIAS	17
Anexos.....	18

INTRODUÇÃO

O mundo atual é altamente tecnológico e as empresas necessitam acompanhar essa evolução para se manterem no mercado. Uma empresa de barbearia, por exemplo, pode enfrentar desafios significativos se não adquirirem ferramentas tecnológicas para gerenciar seus serviços. É comum que muitas empresas desse tipo mantenham registros manuais de clientes, cortes e agendamentos, o que pode levar a erros, atrasos e até mesmo à perda de informações importantes.

Pensando nisso, esse projeto foi desenvolvido com o objetivo de criar uma solução para uma pequena empresa de barbearia (DR Barber), que possui um sistema de controle de clientes, cortes e agendamentos escrito em papel. A ideia foi criar um aplicativo que pudesse automatizar essas atividades, tornando-as mais eficientes e precisas e visualmente mais organizado.

O resultado foi um aplicativo que permite ao proprietário da barbearia gerenciar os aspectos do seu negócio de forma fácil e intuitiva. Através do aplicativo, é possível cadastrar novos clientes, agendar cortes e manter um registro completo de todos os serviços prestados.

2 DESCRIÇÃO DA EMPRESA

A "DR Barber" é uma empresa de porte pequeno, com opções de serviços voltados para o público masculino. Ela oferece uma variedade de serviços, como cortes de cabelo, barba, sobrancelha e outros tratamentos estéticos. A empresa está localizada na zona leste de São José dos Campos, mais precisamente no Residencial Cambuí.

Embora a empresa seja relativamente nova, ela já conquistou uma base fiel de clientes que valorizam a qualidade dos serviços e a atenção aos detalhes oferecidos pela equipe. No entanto, a empresa ainda não possui dados de pessoa jurídica, o que significa que ainda não tem um CNPJ registrado.

Curiosamente, a "DR Barber" foi a escolha da equipe para esse projeto por um pequeno contratempo. Um dos integrantes da equipe já conhecia o serviço prestado pela "DR Barber" e marcou um corte de cabelo. Porém, o dono da empresa acabou esquecendo de anotar o agendamento em um papel e, com isso, o integrante da equipe teve que esperar mais tempo do que o necessário para ser atendido.

Essa situação fez com que a equipe percebesse que a empresa poderia se beneficiar com a adoção de uma solução tecnológica para gerenciar seus agendamentos e clientes. Foi então que eles decidiram procurar o proprietário da "DR Barber" para criar uma solução personalizada para a empresa. Com a adoção do aplicativo Java desenvolvido pela equipe, a "DR Barber" agora tem um sistema moderno e eficiente para gerenciar seus negócios.

3 PROJETO DE CONSULTORIA EMPRESARIAL

3.1 BANCO DE DADOS

3.1.1 MER - MODELAGEM ENTIDADE RELACIONAMENTO

A imagem mostra o modelo de dados para o sistema de gerenciamento da empresa "DR Barber". Podemos ver três tabelas: "Cliente", "Agendamento" e "Serviço".

A tabela "Cliente" contém os campos "Id", "Nome", "CPF", "Endereço" e "Telefone", que representam os atributos do cliente. Essa tabela é usada para guardar as informações dos clientes.

A tabela "Agendamento" contém os campos "Id", "Nome", "Data", "Hora", "Descrição", "Id_Cliente" e "Id_Serviço", que representam os atributos do agendamento. Essa tabela é usada para armazenar as informações sobre os agendamentos dos clientes, a data e a hora do serviço. Os campos "Id_Cliente" e "Id_Serviço" são usados para estabelecer um relacionamento entre a tabela de agendamento e as tabelas de cliente e serviço.

A tabela "Serviço" contém os campos "Serviço" e "Valor", que representam os atributos do serviço. Essa tabela é usada para armazenar informações sobre os serviços prestados pela empresa, incluindo o nome do serviço e o valor cobrado por ele.

Existe um relacionamento entre a tabela "Cliente" e a tabela "Agendamento", representado por uma linha que conecta o campo "Id" na tabela "Cliente" com o campo "Id_Cliente" na tabela "Agendamento". Da mesma forma, existe um relacionamento entre a tabela "Serviço" e a tabela "Agendamento"

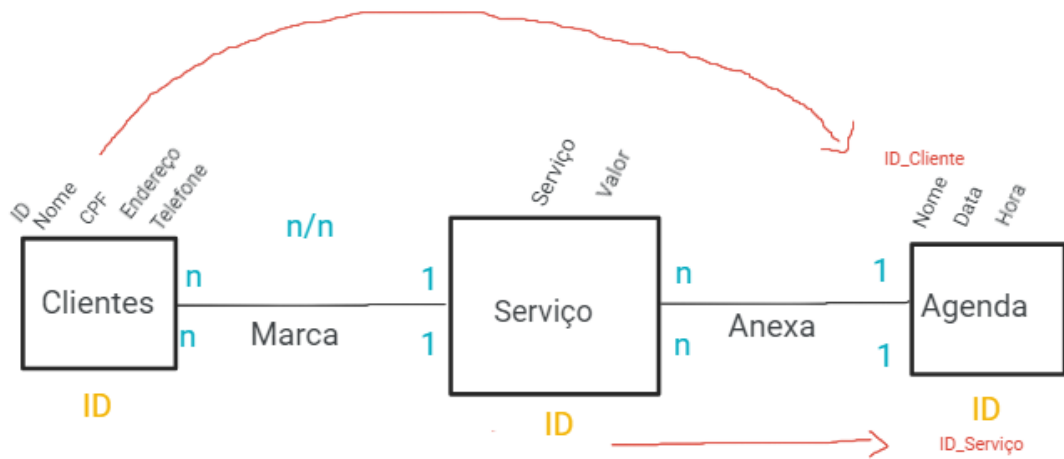


Figura 1

3.1.2 DER - DIAGRAMA ENTIDADE RELACIONAMENTO

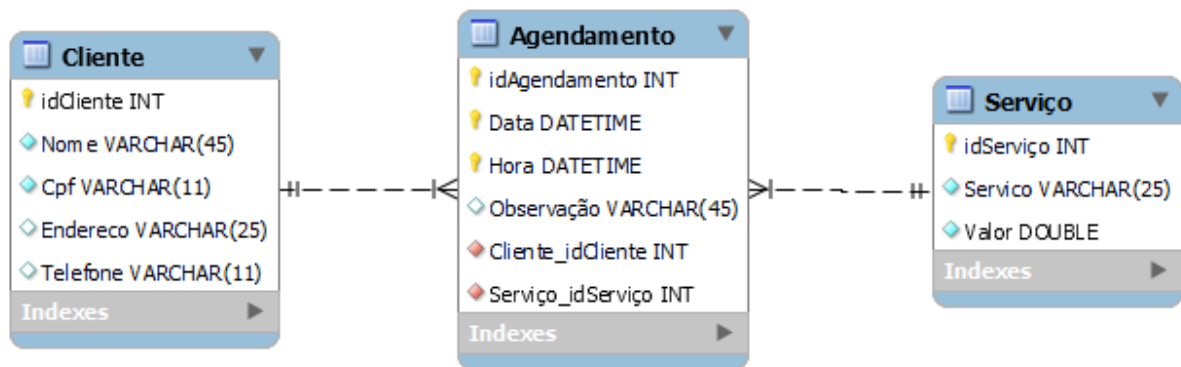


Figura 2

3.1.3 FÍSICO

Durante o desenvolvimento do projeto de gerenciamento da empresa "DR Barber", encontramos dificuldades na conexão do MySQL com o NetBeans, que era a utilizada pelo professor Sidney para programar. Como alternativa, decidimos utilizar o banco de dados PostgreSQL.

Embora tenhamos optado por utilizar o PostgreSQL em vez do MySQL, isso não afetou significativamente o projeto, pois as estruturas e funcionalidades dos dois bancos de dados são bastante semelhantes. Sendo assim, pudemos dar continuidade ao projeto com o novo banco de dados sem maiores problemas e sem precisar alterar significativamente a lógica da aplicação.

- Tabela Serviços:

Id [PK] integer	descricao character varying	valor character varying
---------------------------	---------------------------------------	-----------------------------------

Figura 3

- Tabela Clientes:

Id [PK] integer	nome character varying	cpf character varying	endereco character varying	telefone character varying
---------------------------	----------------------------------	---------------------------------	--------------------------------------	--------------------------------------

Figura 4

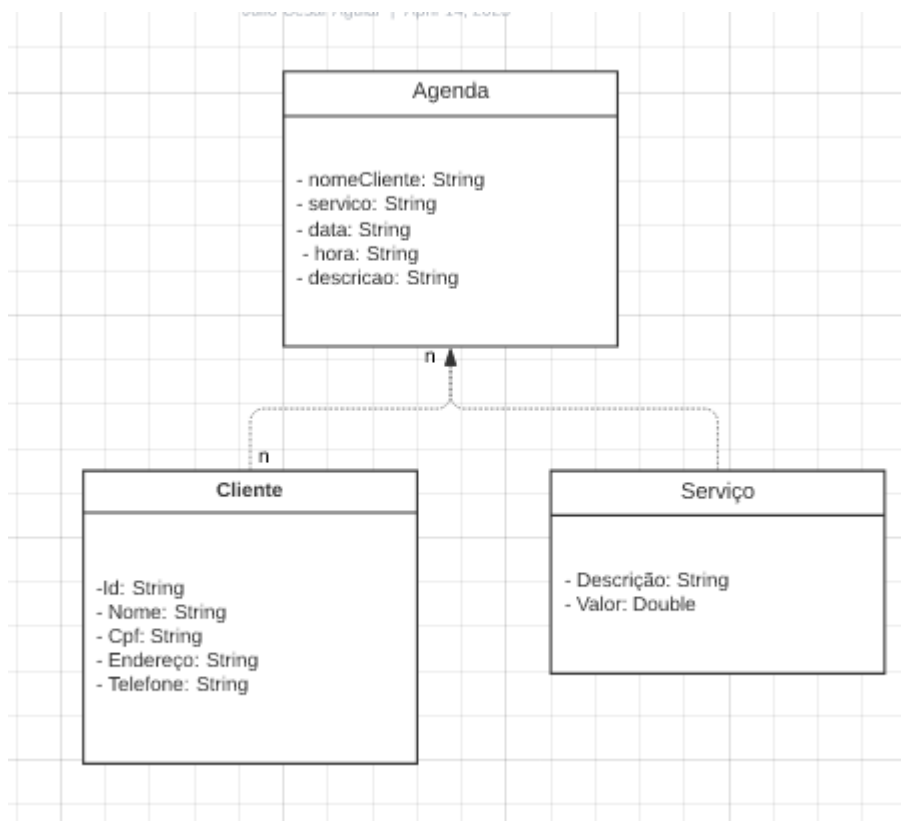
- Tabela Agendamento:

Id integer	cliente character varying	servico character varying	data character varying	hora character varying	descricao character varying
----------------------	-------------------------------------	-------------------------------------	----------------------------------	----------------------------------	---------------------------------------

Figura 5

3.2 PROGRAMAÇÃO ORIENTADA A OBJETOS

3.2.1 DIAGRAMA DE CLASSES



3.2.2 CÓDIGOS DO SISTEMA

O trecho do código a seguir é da view “Agendamento” do projeto, que possui 4 campos obrigatórios e um opcional para ser preenchido:

View Agendamento

```
public class Agenda extends javax.swing.JFrame {  
  
    private final AgendaController controller;  
  
    public Agenda() throws SQLException {  
  
        initComponents();  
  
        controller = new AgendaController(this);  
  
        jComboBoxServico.insertItemAt("Selecione um serviço", 0);  
  
        ServicoFactory servicoFactory = new ServicoFactory(new  
        Conexao().getConnection());  
  
        List<String> descricoes = servicoFactory.getDescricaoServicos();  
  
        for (String descricao : descricoes) {  
  
            jComboBoxServico.addItem(descricao);  
  
        }  
  
        jComboBoxServico.setSelectedIndex(0);  
  
    }  
  
    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
  
        controller.SalvarAgendamento();  
  
    }  
  
}
```

O Código acima foi retirado boa parte, pois é muito extenso, por possuir muitas linhas de código em branco, como linhas que apenas adicionam o título no código, porém o foco é mostrar as classes, e como chamamos no código o controller “AgendaController” que será responsável por pegar os dados dessa view e salvar no banco de dados.

Controller AgendaControlle

```

public class AgendaController {
    private final Agenda view;

    public AgendaController(Agenda view) {
        this.view = view; }

    public void SalvarAgendamento() {
        String servicoSelecionado = (String) view.getjComboBoxServico().getSelectedItem();

        if (servicoSelecionado.equals("Selecione um serviço")) {
            JOptionPane.showMessageDialog(null, "Por favor, selecione um serviço.");
        } else {
            String cliente = view.getjTextFieldCliente().getText();
            String servico = (String) view.getjComboBoxServico().getSelectedItem();
            String data = view.getjTextFieldData().getText();
            String hora = view.getjTextFieldHora().getText();
            String descricao = view.getjTextAreaDescricao().getText();

            Agendamento agendamento = new Agendamento(cliente, data, hora, servico,
            descricao);

            try {
                Connection conexao = new Conexao().getConnection();

                AgendamentoFactory agendafactory = new
                AgendamentoFactory(conexao);

                agendafactory.insert(agendamento);

                JOptionPane.showMessageDialog(null, "Agendamento salvo com sucesso!");

            } catch (SQLException ex) {

                Logger.getLogger(Agenda.class.getName()).log(Level.SEVERE, null, ex);

            }

        }
    }
}

```

Acima temos todo o código do controller da agenda, que é responsável primeiro de validar se o campo “Serviço” foi preenchido com algum serviço, e se caso ele não foi preenchido, o sistema retorna uma mensagem dizendo para selecionar um serviço, caso contrário, ele salva os dados no banco de dados

Model Agendamento

```
public class AgendamentoFactory {
    private final Connection connection;

    public AgendamentoFactory(Connection connection) {
        this.connection = connection;
    }

    public void insert(Agendamento agendamento) throws SQLException{
        String sql = "insert into agendamentos(cliente,servico,data,hora,descricao)
        values("+agendamento.getNomeCliente()+", "+agendamento.getServico()+", "+agend
        damento.getData()+", "+agendamento.getHora()+", "+agendamento.getDescricao()+
        ")";

        PreparedStatement statement = connection.prepareStatement(sql);
        statement.execute();
    }

    public List<Agendamento> listar() throws SQLException {
        List<Agendamento> agendamentos = new ArrayList<>();
        String sql = "SELECT * FROM agendamentos";

        try (PreparedStatement stmt = connection.prepareStatement(sql)) {
            ResultSet rs = stmt.executeQuery();
            while (rs.next()) {
                String cliente = rs.getString("cliente");
                String data = rs.getString("data");
                String hora = rs.getString("hora");
```

```

String servico = rs.getString("servico");

String descricao = rs.getString("descricao");

Agendamento agendamento = new Agendamento(cliente, data, hora, servico,
descricao);

    agendamentos.add(agendamento);

    }

}

return agendamentos;

}

```

Acima, o código que passa os parâmetros que se deve ter para marcar um agendamento no banco de dados

3.2.3 IMAGENS DO SISTEMA

- Tela login com o logo da empresa logo abaixo do botão “Entrar”:



Figura 6

- Tela menu principal



Figura 7

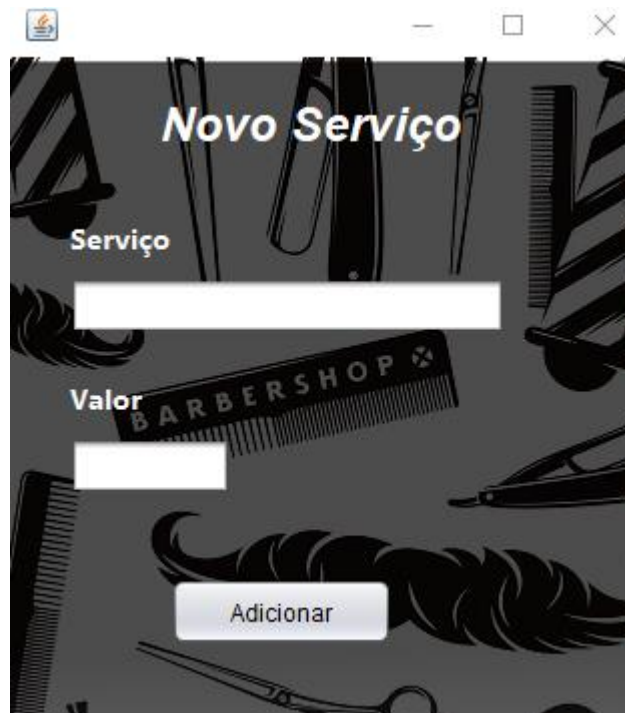
Dentro do menu cadastro temos os submenus: Cliente, Serviço e Usuário. E dentro de processos, temos a agenda.

- Tela cadastro cliente:

A screenshot of a web browser window. The title bar reads "Novo Cliente". The main content area has a dark background with a pattern of white line-art illustrations of barber tools: a straight razor, a pair of scissors, a comb, and a shaving brush. The text "Novo Cliente" is displayed in a large, white, serif font at the top. Below it, there are four white input fields, each with a label to its left: "Nome", "CPF", "Endereço", and "Telefone". At the bottom center, there is a white button with the text "Adicionar".

Figura 8

- Tela cadastro Serviço:



Novo Serviço

Serviço

Valor

Adicionar

Figura 9

- Tela cadastro de usuário do sistema:



Novo Usuário

Usuário

Senha

Adicionar

Figura 10

- Tela de agendamento:



The screenshot shows a mobile application interface for scheduling appointments. The title "Agendamento" is prominently displayed at the top. Below the title, there are four input fields: "Cliente" with a dropdown menu showing "Selecione um cliente", "Serviço" with a dropdown menu showing "Selecione um servi...", "Data" with a text input field, and "Hora" with a text input field. To the right of these fields is a large white text area labeled "OBS:". At the bottom of the form, there are two green buttons: "Agendar" and "Verificar Agenda".

Figura 11

- Consulta de agendamentos:



The screenshot shows a mobile application interface for querying appointments. The title "AGENDAMENTOS" is prominently displayed at the top. Below the title, there is a table with four columns: "Nome", "Serviço", "Data", and "Hora". The table is currently empty, and the background is a light gray color.

Nome	Serviço	Data	Hora
------	---------	------	------

Figura 12

4 CONCLUSÃO

O projeto desenvolvido para a empresa DR Barber foi de grande importância tanto para o proprietário da empresa quanto para o grupo de estudantes envolvidos. Para o proprietário, o projeto trouxe a modernização do controle de clientes, cortes e agendamentos, possibilitando uma melhor organização e agilidade no atendimento. Com a utilização da aplicação desenvolvida em Java, o proprietário agora possui uma ferramenta mais precisa e confiável para gerenciar o seu negócio.

Para o grupo de estudantes envolvidos, o projeto trouxe aprendizados valiosos em diversas áreas, como a criação de um banco de dados utilizando o PostgreSQL, a utilização de programação orientada a objetos em Java, entre outros conhecimentos importantes na área de desenvolvimento de software. Além disso, a experiência de trabalhar em equipe, desenvolver um projeto real e entregar um produto de qualidade, foram aprendizados muito valiosos para a formação acadêmica e profissional dos estudantes.

Em resumo, o projeto trouxe benefícios tanto para o proprietário da empresa quanto para os estudantes envolvidos, sendo uma experiência enriquecedora para todos. Através do desenvolvimento dessa aplicação, foi possível unir a teoria aprendida em sala de aula com a prática, aplicando conceitos e técnicas em um projeto real e gerando resultados concretos.

REFERÊNCIAS

DEV MEDIA. Guia da Linguagem Java. Disponível em: <https://www.devmedia.com.br/guia/linguagem-java/38169>. Acesso em: 19 mar. 2023.

FERREIRA, Marcelo S. Erros de Conexão com o SQL Server. DevMedia, 2014. Disponível em: <https://www.devmedia.com.br/erros-de-conexao-com-o-sql-server/3268>. Acesso em: 02 abr. 2023.

FRESHIP. What is React JS? [Vídeo]. 2023. Disponível em: <https://www.youtube.com/watch?v=XfO3TRvESBo>. Acesso em: 14 abr. 2023.

GEEK UNIVERSITY. Banco de Dados - Aula 1 - Introdução [Vídeo]. 2016. Disponível em: <https://www.youtube.com/watch?v=le8b2tGuJfl>. Acesso em: 16 mar. 2023.

LUZ, José. Java - Como efetuar uma conexão Java JDBC PostgreSQL usando o NetBeans IDE? Arquivo de Códigos, 2019. Disponível em: <https://www.arquivodecodigos.com.br/dicas/3538-java-como-efetuar-uma-conexao-java-jdbc-postgresql-usando-o-netbeans-ide.html>. Acesso em: 23 mar. 2023.

PROGRAMAÇÃO DINÂMICA. Curso de Java Básico [Série de vídeos]. 2022. Disponível em: https://www.youtube.com/watch?v=UtxTG1_AiXk&list=PLJIP7GdByOyuBKB--fIO2DoQaPVXm9ICw. Acesso em: 15 mar. 2023.

PROGRAMAÇÃO DINÂMICA. Java Básico: Aula 8 - Banco de Dados [Vídeo]. 2017. Disponível em: https://www.youtube.com/watch?v=YYPIM7D8oU0&list=PLJIP7GdByOyvpQ7EbzucVHocSAG7_EZQZ. Acesso em: 12 mar. 2023.

TECNOBLOG. Guia ABNT para trabalhos acadêmicos e TCC [Blog]. Disponível em: <https://tecnoblog.net/responde/guia-normas-abnt-trabalho-academico-tcc/>. Acesso em: 10 abr. 2023.

DEV MEDIA. Orientações básicas na elaboração de um diagrama de classes [Blog]. Disponível em: <https://www.devmedia.com.br/orientacoes-basicas-na-elaboracao-de-um-diagrama-de-classes/37224>. Acesso em: 14 abr. 2023.

Anexos

Link vídeo: https://drive.google.com/file/d/1-dKIhslIUy-ISwCsAYkpRqMpinSSoTaQ/view?usp=share_link