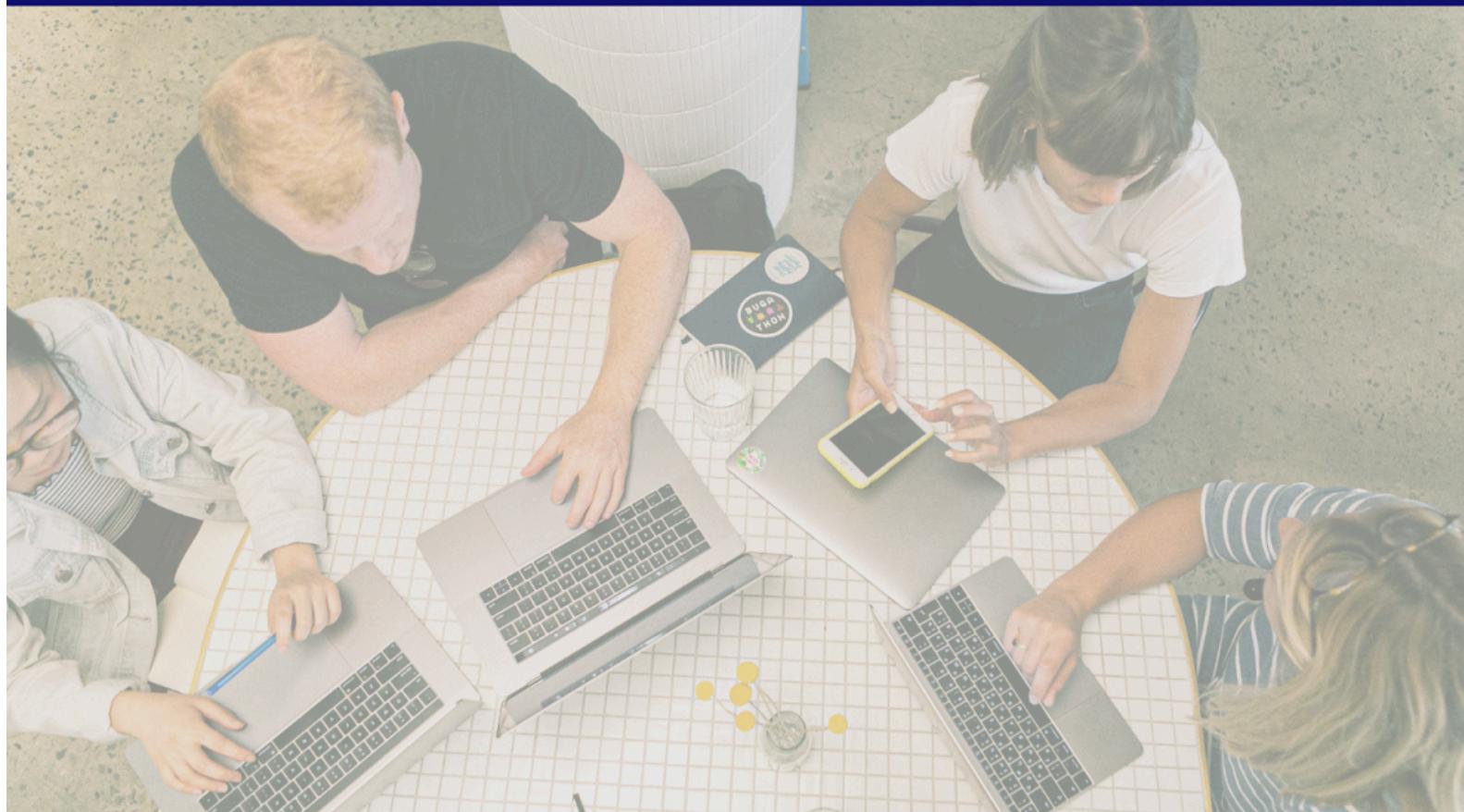


**UNifeob**  
| ESCOLA DE NEGÓCIOS



2024

# PROJETO INTEGRADO



UNIFEOB

CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO  
OCTÁVIO BASTOS

ESCOLA DE NEGÓCIOS

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS  
CIÊNCIA DA COMPUTAÇÃO**

**PROJETO INTEGRADO**

**DESENVOLVIMENTO DE SOLUÇÕES CONSOLE  
INTEGRADAS PARA EDUCAÇÃO,  
SUSTENTABILIDADE, INCLUSÃO SOCIAL E  
EMPREENDEDORISMO**

**<EMPRESA>**

SÃO JOÃO DA BOA VISTA, SP

NOVEMBRO 2024

UNIFEOB  
CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO  
OCTÁVIO BASTOS  
ESCOLA DE NEGÓCIOS  
**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS  
CIÊNCIA DA COMPUTAÇÃO**

**PROJETO INTEGRADO**  
**DESENVOLVIMENTO DE SOLUÇÕES CONSOLE  
INTEGRADAS PARA EDUCAÇÃO,  
SUSTENTABILIDADE, INCLUSÃO SOCIAL E  
EMPREENDEDORISMO**

**<EMPRESA>**

MÓDULO MODELAGEM E DESENVOLVIMENTO DE SISTEMAS

Business Intelligence – Profª. Mariângela Martimbianco Santos

Programação Orientada a Objeto – Prof. Nivaldo de Andrade

Lógica de Programação – Prof. Marcelo Ciacco Almeida

Modelagem de Dados – Prof. Max Streicher Vallim

Projeto de Modelagem e Desenvolvimento de Sistemas – Profª. Mariângela M. Santos

Estudantes:

Euller Sodré Terciano, RA 24000526

Gustavo Oliveira Luna Nascimento, RA 24000461

José Eduardo de Paula Sorg, RA 24000304

Miguel Turatti Quirino, RA 24000776

Rafael Modena Emygdio, RA 24000463

SÃO JOÃO DA BOA VISTA, SP  
NOVEMBRO 2024

# SUMÁRIO

1. INTRODUÇÃO	4
2. DESCRIÇÃO DA EMPRESA	7
3. PROJETO INTEGRADO	8
3.1 PROGRAMAÇÃO ORIENTADA A OBJETO	8
3.1.1 CLASSES E OBJETOS	8
3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.	9
3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS	10
3.2 LÓGICA DE PROGRAMAÇÃO	10
3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE	11
3.2.2 DESENVOLVIMENTO DE APLICAÇÕES	11
3.2.3 IMPLEMENTAÇÃO E VALIDAÇÃO	11
3.3 MODELAGEM DE DADOS	11
3.3.1 MODELO CONCEITUAL	12
3.3.2 MODELO LÓGICO E FÍSICO	12
3.3.3 SQL	12
3.4 BUSINESS INTELLIGENCE	12
3.4.1 ORGANIZAÇÃO E IDENTIFICAÇÃO DAS INFORMAÇÕES	12
3.4.2 MANIPULAÇÃO E ANÁLISE DE DADOS	12
3.4.3 CRIAÇÃO DE MODELOS DE ANÁLISE DE DADOS	13
3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS	13
3.5.1 GERENCIANDO FINANÇAS	13
3.5.2 ESTUDANTES NA PRÁTICA	14
4. CONCLUSÃO	16
REFERÊNCIAS	17
ANEXOS	18

# 1. INTRODUÇÃO

O objetivo do nosso Projeto Integrado é desenvolver uma aplicação console que integre conceitos avançados de tecnologia para atender às necessidades do Supermercado Paraíso, promovendo educação, sustentabilidade, inclusão social e empreendedorismo. A proposta visa criar soluções práticas e inovadoras que contribuam para o desenvolvimento sustentável e inclusivo dentro do contexto do supermercado.

## Áreas de Atuação e Impacto

### Educação e Capacitação:

Diferença: Implementação de programas de capacitação para funcionários, incluindo cursos sobre atendimento ao cliente, gestão de estoque e uso de tecnologias modernas. Isso aumentará a eficiência e a satisfação no ambiente de trabalho.

### Meio Ambiente e Sustentabilidade:

Diferença: Desenvolvimento de um sistema de gestão de resíduos que minimize o desperdício e maximizar a reciclagem, além de campanhas de conscientização ambiental para clientes e funcionários, destacando práticas sustentáveis.

### Inclusão Social:

Diferença: Criação de iniciativas que promovam a inclusão de pessoas com deficiência e outras minorias, ajustando a infraestrutura do supermercado e oferecendo programas de emprego e treinamento específicos.

### Tecnologia e Inovação:

Diferença: Aplicação de novas tecnologias para otimizar o gerenciamento de estoque e melhorar a experiência do cliente, como sistemas de autoatendimento e análise de dados para personalizar ofertas e promoções.

### Desenvolvimento Rural:

Diferença: Estabelecimento de parcerias com agricultores locais para promover a agricultura familiar, garantindo produtos frescos e sustentáveis, além de apoiar a economia local.

### Empreendedorismo e Economia Solidária:

Diferença: Fomento ao empreendedorismo local por meio de parcerias com pequenos produtores e cooperativas, criando um espaço para a venda de produtos locais e sustentáveis, fortalecendo a economia solidária.

## Proposta de Valor

O projeto proporcionará ao Supermercado Paraíso uma vantagem competitiva ao aliar tecnologia e práticas sustentáveis, melhorando a eficiência operacional e a imagem da marca. Além disso, contribuirá para a formação de uma comunidade mais consciente e integrada, aumentando a fidelidade dos clientes e o impacto social positivo.

#### Conclusão

Esta aplicação representa uma oportunidade única de integrar tecnologia e responsabilidade social, posicionando o Supermercado Paraíso como líder em inovação e sustentabilidade no setor, enquanto capacita seus colaboradores e apoia a comunidade local.

## **2. DESCRIÇÃO DA EMPRESA**

Em 14 de abril de 2001, os irmãos Carlos e Claudia Pulieri abriram o primeiro supermercado Paraíso. Com o passar dos anos e o sucesso do negócio, surgiu a oportunidade de expansão, e, em 20 de novembro de 2010, inauguraram a segunda unidade.

O projeto Paraíso Store foi uma iniciativa para expandir o negócio para fora da cidade de Aguai, com a abertura de uma loja em São João da Boa Vista. No entanto, devido à falta de mão de obra qualificada, a unidade precisou ser fechada.

Atualmente, os irmãos planejam abrir um terceiro supermercado em 2026. A empresa começou sob a administração de seu pai, com o nome Armando Pulieri ME. Com o crescimento das operações, foi transformada em uma sociedade LTDA, na qual Carlos e Claudia dividem as participações em 50% cada. Essa parceria, baseada em perspectivas complementares, fortalece tanto as decisões estratégicas quanto a evolução do negócio.

### **3. PROJETO INTEGRADO**

Nesta etapa do PI serão apresentados os conteúdos que cada unidade de estudo utilizará para realizar o projeto, assim como a forma que serão aplicados na empresa escolhida para a realização do projeto.

#### **3.1 PROGRAMAÇÃO ORIENTADA A OBJETO**

A Programação Orientada a Objetos (POO) desempenha um papel crucial na formação de estudantes de tecnologia, permitindo-lhes estruturar o código de maneira organizada e eficiente. Conceitos fundamentais como classes, objetos, herança e polimorfismo são não apenas pilares da POO, mas também ferramentas essenciais para a construção de soluções robustas que atendam às demandas do mercado de trabalho.

Com a crescente necessidade por software bem estruturado e de fácil manutenção, os estudantes que dominam a POO se destacam em um cenário competitivo. Essa abordagem permite que os desenvolvedores encapsulam dados e funcionalidades, promovendo uma manipulação segura e eficiente das informações. Além disso, a POO facilita a integração com outras áreas de projetos, como Business Intelligence e Modelagem de Dados, o que é vital para o desenvolvimento de sistemas complexos.

A aplicação prática dos conceitos de POO é especialmente relevante em áreas como educação, sustentabilidade, inclusão social, tecnologia e inovação, desenvolvimento rural e empreendedorismo solidário. Ao utilizar a POO para resolver problemas complexos, os estudantes não apenas aprimoram suas habilidades técnicas, mas também contribuem para um desenvolvimento sustentável e inclusivo.

Em suma, a Programação Orientada a Objetos é uma ferramenta poderosa que, além de preparar os estudantes para o mercado de trabalho, permite que eles se tornem agentes de mudança em suas comunidades, utilizando a tecnologia para promover soluções que atendam às necessidades sociais e ambientais. Assim, a formação em POO é um investimento fundamental para o futuro dos profissionais da área de tecnologia.

### **3.1.1 CLASSES E OBJETOS**

Neste projeto "Desenvolvimento de Soluções Console Integradas para Educação, Sustentabilidade, Inclusão Social e Empreendedorismo", nós, como estudantes, somos responsáveis por aplicar o conhecimento adquirido para desenvolver uma aplicação utilizando conceitos de programação. A ideia central é representar entidades do mundo real, relacionadas aos temas do projeto, e estruturar dados e comportamentos de maneira eficiente. Através dessa prática, temos a oportunidade de entender como organizar sistemas que promovam a modularidade e a reutilização de código, elementos essenciais para construir aplicações robustas e fáceis de manter. Além disso, o projeto nos prepara para enfrentar desafios reais, desenvolvendo soluções tecnológicas que abordam questões fundamentais, como a sustentabilidade e a inclusão social, ao mesmo tempo que aprimoramos habilidades cruciais para o mercado. Esse trabalho proporciona uma compreensão mais profunda de como a programação pode ser utilizada para criar impacto positivo, promovendo tanto o aprendizado técnico quanto a reflexão sobre a aplicação prática de soluções inovadoras.

### **3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.**

No contexto do projeto "Desenvolvimento de Soluções Console Integradas para Educação, Sustentabilidade, Inclusão Social e Empreendedorismo", a Programação Orientada a Objetos (POO) se destaca como uma abordagem essencial para a criação de soluções robustas e eficientes. O uso de conceitos fundamentais da POO, como atributos, métodos, encapsulamento, herança e polimorfismo, possibilitará o desenvolvimento de uma aplicação bem estruturada e facilmente mantida.

Os atributos representam as características ou propriedades das classes. Por exemplo, na classe curso, serão definidos atributos como nome, duração e categoria que armazenam informações específicas sobre cada curso oferecido. Essa definição clara dos atributos ajuda a organizar os dados e a estabelecer uma base sólida para a manipulação das informações.

Os métodos, por sua vez, são as ações ou comportamentos que as classes podem executar. Na classe curso, métodos como matricularAluno() e emitirCertificado() desempenham funções cruciais para a gestão dos cursos e a interação com os alunos. A

implementação adequada desses métodos permitirá uma gestão eficiente das operações relacionadas aos cursos.

O encapsulamento é outra prática fundamental, que envolve ocultar os detalhes internos de uma classe e expor apenas o necessário. No projeto, isso será alcançado ao definir atributos como privados e fornecer métodos públicos para acessá-los e modificá-los. Essa abordagem garante a integridade dos dados, aumenta a segurança do código e facilita a manutenção futura.

A herança permitirá a criação de classes que derivam características e comportamentos de outras, promovendo a reutilização de código. Por exemplo, uma classe `CursoOnline` poderá herdar de `Curso`, adicionando atributos e métodos específicos, como `plataforma` e `adicionarMaterial`. Essa característica não apenas simplifica o código, mas também promove uma estrutura mais lógica e organizada.

Por fim, o polimorfismo permite que métodos com o mesmo nome se comportem de maneira diferente em classes distintas. No projeto, será possível definir o método `matricularAluno()` tanto na classe `Curso` quanto na classe `CursoPresencial`, com implementações específicas que atendem às necessidades de cada tipo de curso. Essa flexibilidade é crucial para adaptar o comportamento do software às diferentes situações encontradas durante o uso.

A aplicação desses conceitos de POO resultará em uma aplicação mais organizada, modular e reutilizável, facilitando a gestão e a expansão do código. Dessa forma, o projeto não só atenderá aos objetivos de educação, sustentabilidade, inclusão social e empreendedorismo, mas também se mostrará eficiente e inovador, promovendo um impacto positivo nas comunidades envolvidas. Assim, a POO se confirma como uma ferramenta poderosa para a construção de soluções que buscam transformar realidades e contribuir para um futuro mais sustentável e inclusivo.

### **3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS**

No projeto "Desenvolvimento de Soluções Console Integradas para Educação, Sustentabilidade, Inclusão Social e Empreendedorismo", você utilizará diferentes tipos de métodos para garantir que a aplicação seja eficiente e bem estruturada. Métodos Estáticos são

aqueles que pertencem à classe em si, e não a uma instância específica da classe. Eles são úteis para operações que não dependem de dados de instância, como métodos utilitários ou funções auxiliares. Por exemplo, você pode criar um método estático `calcularTotalCursos()` na classe `GerenciadorDeCursos` para calcular o total de cursos disponíveis, que pode ser chamado diretamente na classe sem a necessidade de criar um objeto.

Métodos Públicos são aqueles que podem ser acessados de fora da classe e são utilizados para interagir com objetos. Eles definem as ações que podem ser realizadas sobre e com a instância da classe. No seu projeto, métodos públicos como `matricularAluno()` na classe `Curso` permitem que outras partes do código interajam com os objetos de curso, realizando operações como matricular um aluno ou atualizar informações do curso.

Métodos Privados, por outro lado, são acessíveis apenas dentro da própria classe e são usados para encapsular a lógica interna que não deve ser exposta ao mundo externo. Por exemplo, um método privado `validarDados()` na classe `Curso` pode ser utilizado para verificar se as informações fornecidas são corretas antes de realizar uma matrícula. Isso ajuda a manter a integridade dos dados e a proteger a classe contra modificações indesejadas de fora.

Utilizando métodos estáticos, públicos e privados de forma adequada, você garantirá que a aplicação seja modular, segura e fácil de manter, facilitando a gestão de suas funcionalidades e a interação com os dados e comportamentos encapsulados.

## 3.2 LÓGICA DE PROGRAMAÇÃO

A lógica de programação é uma parte fundamental no início do desenvolvimento de sistemas, pois fornece a base para entender como os computadores processam informações e como podemos usar essa lógica para resolver problemas. Neste tópico, exploraremos os conceitos essenciais que compõem a lógica de programação, com foco na aplicação prática através da linguagem Python.

Os estudantes aprenderão sobre algoritmos, que são sequências de instruções claras e finitas para resolver um problema específico. A criação de algoritmos é uma habilidade crucial, pois permite que os programadores planejem a solução antes de codificá-la.

Outro conceito importante são as variáveis, que são usadas para armazenar dados que podem ser manipulados durante a execução do programa. Junto a isso, entender os tipos de

dados é fundamental, uma vez que eles definem a natureza das informações que as variáveis podem conter, como inteiros, flutuantes, strings (textos) e booleanos.

O estudo das funções também é essencial, pois elas permitem que o código seja organizado em blocos reutilizáveis, facilitando a manutenção e a legibilidade. Além disso, as estruturas condicionais são abordadas para ensinar como os programas podem tomar decisões com base em condições específicas, usando comandos como `if`, `else` e `elif`.

Por fim, os operadores lógicos e operadores de comparação serão discutidos, pois eles são utilizados para avaliar expressões e criar condições complexas que afetam o fluxo do programa.

Com a linguagem Python como ferramenta de aprendizado, os estudantes poderão aplicar todos esses conceitos de forma prática, desenvolvendo habilidades que serão indispensáveis no futuro desenvolvimento de sistemas. A lógica de programação, portanto, não apenas fundamenta o aprendizado técnico, mas também promove um raciocínio crítico e analítico necessário para o sucesso na área de tecnologia.

### **3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE**

Variáveis são valores que podem ser modificados durante a execução de um programa. Elas têm um tipo de dado, que se refere à categoria de valores que podem ser armazenados. Por exemplo, uma variável de tipo inteiro guarda valores de números inteiros, enquanto um `int` ou uma `string` são tipos de dados. Também existem tipos compostos, permitindo guardar vários valores ou mesmo vários tipos em uma única estrutura, como uma lista ou `array`, que pode ajudar a organizar e a manipular os dados em código.

Para receber funções que te ajudam a realizar garantir um determinado conjunto de atividade, você pode reutilizar esse código em vários locais diferentes do seu programa; Isto leva a uma grande disposição e clareza do código. As funções possuem entradas, que são geralmente chamadas de parâmetros ou argumentos. Eles não só alteram a maneira como sua função deve ser executada mas, eles também retornam um valor sempre que chamados podendo logo depois ser usado de alguma forma, por sua vez, caminha para a criação de uma programação mais modular, que se torna mais dinâmica e reutilizável, ao mesmo tempo que

também fomenta o desempenho decente e nivela a quantidade de código que deve ser mantido.

Estruturas condicionais são fundamentais para que um programa tome decisões conforme o fluxo de execução. A mais comum é a estrutura "se" (if), que permite ao programa, se necessário, verificar se uma condição está correta. Com isso o código executa diferentes ações. De acordo com o resultado elas se tornam o programa em si, permitindo que ele responda entradas ou variáveis cujo estado é controlado (dependendo de uma condição previamente especificada).

Operadores em programação são símbolos que realizam ações específicas para manipular variáveis e valores, essenciais na criação de algoritmos e controle do fluxo do programa. Operadores aritméticos, como soma e multiplicação, permitem cálculos numéricos; operadores lógicos, como "e" e "ou", auxiliam em decisões condicionais; e operadores de comparação, como "igual a" ou "maior que", verificam relações entre valores, retornando verdadeiro ou falso. Esses operadores tornam possível definir a lógica e o comportamento das operações que o sistema deve executar.

### **3.2.2 DESENVOLVIMENTO DE APLICAÇÕES**

As regras de negócios definem comportamentos e operações essenciais que o aplicativo deve seguir para atender aos requisitos do sistema. Deve ser claro, conciso e refletir as necessidades do usuário ou do domínio comercial. Exemplos de regras de negócios:

Cadastro de usuários: O sistema deve permitir o cadastro de novos usuários com os campos nome, email, senha e data de nascimento. O e-mail deve ser único e verificado no momento do cadastro. Autenticação: O usuário deverá fazer login com seu e-mail e senha. Se os dados não corresponderem, uma mensagem de erro será exibida. Relatório de vendas: O sistema deverá gerar um relatório mensal de vendas com filtros por categoria de produto, data e valor. Essas regras servem de base para a construção da lógica de negócios da aplicação.

#### **2. Estrutura do algoritmo**

Após definir as regras de negócio, deve-se implementar os algoritmos que irão realizar as operações esperadas pela aplicação. A implementação deve ser feita de forma que cada

trecho de código tenha uma responsabilidade única, o que ajuda a manter o código limpo e modular. Exemplo:

Algoritmo de registro de usuário:

Obtenha dados de entrada (nome, email, senha, data de nascimento). Verifique o e-mail (deve ser exclusivo). Verifique se a senha atende aos critérios de segurança (mínimo 8 caracteres). Salve os dados no banco de dados. Retorna uma mensagem de sucesso ou erro. Este algoritmo pode ser implementado em uma função específica de cadastro de usuário, que será reutilizável em diferentes partes do código.

### 3. Divisão em funções e módulos

Para garantir que o código seja modular e de fácil manutenção, é importante dividir a aplicação em pequenas funções que resolvam problemas específicos. Cada módulo deverá ser responsável por uma tarefa específica e bem definida.

Exemplo de estrutura de módulo:

Módulo do usuário:

Função `Cadastrar_usuario()`: para cadastrar novos usuários. Função `valid_email()`: para verificar a exclusividade do email. Função `Validate_password()`: para garantir que a senha é segura. Função `Login_usuario()`: para realizar autenticação. Formulário de relatório:

Função `Generate_relatorio_vendas()`: para gerar relatórios de vendas com base em filtros. Função `Filter_sales()`: para filtrar dados com base nas preferências do usuário (data, categoria, valor, etc.). Estes módulos deverão ser independentes e reutilizáveis, facilitando assim a manutenção e expansão do sistema no futuro.

### 3.2.3 IMPLEMENTAÇÃO E VALIDAÇÃO

A etapa de implementação e validação envolve a união de todos os componentes desenvolvidos, garantindo que a aplicação funcione de maneira integrada e eficiente. Primeiro, os módulos criados são conectados e testados, um de cada vez, para garantir que cada parte do sistema esteja funcionando corretamente antes de avançar para a próxima.

Em seguida, são feitos testes para verificar se a aplicação está atendendo às necessidades do projeto. Isso inclui checar se cada funcionalidade funciona como esperado e se o sistema, como um todo, opera sem falhas. A validação dos requisitos é a última etapa,

onde se compara o que foi entregue com os requisitos originais para garantir que o sistema atenda a todas as expectativas.

Essa fase assegura que o produto final esteja pronto para uso, com todos os requisitos atendidos e funcionando corretamente.

### **3.3 MODELAGEM DE DADOS**

#### **Importância da Modelagem de Dados**

A modelagem de dados é crucial para garantir que o banco de dados seja eficiente e atenda corretamente às necessidades do sistema. O modelo lógico, ao mapear as entidades e seus relacionamentos de maneira abstrata, permite uma visão clara de como os dados serão utilizados, enquanto o modelo físico garante que esses dados sejam armazenados de forma otimizada para consultas rápidas e para manutenção de integridade e consistência.

Durante o desenvolvimento de sistemas, a escolha e implementação do modelo físico dependem de diversos fatores, como o volume de dados, a frequência de acesso e as características do ambiente de execução. Um modelo físico bem estruturado pode melhorar significativamente o desempenho do sistema, evitando problemas como lentidão nas consultas ou redundância desnecessária de dados. Portanto, a transição do modelo lógico para o físico é uma fase crítica para garantir que o banco de dados seja não só funcional, mas também eficiente e escalável.

Em resumo, a modelagem de dados, ao combinar o modelo lógico e o modelo físico, proporciona uma base sólida e bem estruturada para o desenvolvimento de qualquer sistema que envolva a manipulação de dados. Ao realizar essas etapas com cuidado, os estudantes garantem que o banco de dados será capaz de suportar as operações do sistema de maneira eficaz e sem comprometer o desempenho, o que é fundamental para o sucesso a longo prazo do sistema em produção.

Sobre o nosso PI:

**RH (Recursos Humanos)**

O setor de RH está conectado a várias funções, indicando que ele gerencia aspectos relacionados à gestão e funcionários. O relacionamento entre RH e outros elementos é de (1,n), o que sugere que o setor de RH gerencia vários funcionários e operações.

### **Dados**

Este componente está associado à gestão e alimentação, possivelmente indicando que o mercado coleta e organiza dados referentes à alimentação e gestão do local. A presença de “Dados” pode representar o controle de informações para tomadas de decisão.

### **Funcionários e Empregado**

Funcionários representam a força de trabalho do mercado, e está ligado a empregado e empacotador, com uma relação de (1,n), indicando que um funcionário pode estar alocado em diferentes funções.

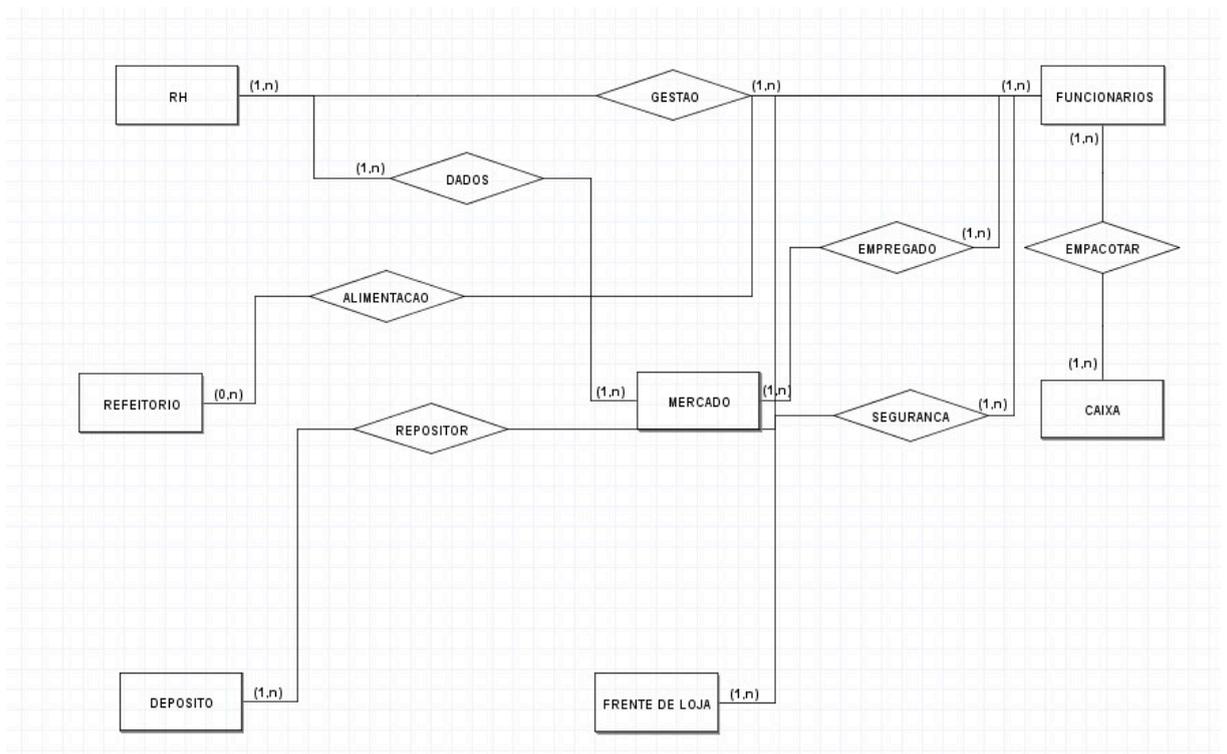
Empregado está relacionado à gestão, indicando que há uma relação direta entre a gestão e o quadro de empregados.

#### **3.3.1 MODELO CONCEITUAL**

O modelo conceitual é uma etapa fundamental no desenvolvimento de qualquer aplicação, pois nos permite entender e mapear as informações de forma abstrata, antes mesmo de começarmos a programar. Nesse contexto, trabalhamos com a criação de entidades que representam elementos do mundo real, como pessoas, recursos ou ações, relacionadas aos temas do projeto, como educação, sustentabilidade e inclusão social. Além de identificar essas entidades, é necessário definir seus atributos e como elas se conectam entre si, o que nos ajuda a construir uma estrutura sólida de dados que servirá de base para o desenvolvimento do sistema. Essa etapa é essencial para garantir que a aplicação seja eficiente, organizada e fácil de manter, promovendo uma visão clara e modular de como as informações serão manipuladas e processadas. O modelo conceitual não apenas organiza o pensamento, mas também prepara o terreno para a implementação técnica, permitindo que o desenvolvimento seja mais ágil e assertivo. Por meio dessa prática, adquirimos habilidades importantes, como a capacidade de planejar e estruturar soluções tecnológicas que resolvem problemas reais, ao mesmo tempo que contribuimos para desafios globais, como a inclusão e a sustentabilidade, aplicando o conhecimento de forma estratégica e inovadora.

De acordo com Richardson (2012), um modelo conceitual é uma “representação simplificada e abstrata de uma realidade complexa”, que ajuda a visualizar e a interpretar relações entre conceitos. Na prática, isso significa que o modelo conceitual define quais elementos são relevantes para o estudo e como eles se interconectam. Na pesquisa científica, esses modelos são fundamentais para construir hipóteses e determinar o que será investigado, orientando o desenvolvimento da metodologia e das perguntas de pesquisa (Yin, 2016).

Os modelos conceituais geralmente consistem em variáveis, conceitos e suas relações. Eles podem ser diagramas, gráficos ou mesmo descrições textuais, dependendo do campo de estudo e do objetivo da pesquisa. Miles e Huberman (1994), em sua obra sobre análise de dados qualitativos, argumentam que modelos conceituais ajudam a organizar o raciocínio lógico, facilitando a compreensão e a interpretação dos dados. Para os autores, esses modelos são essenciais para estruturar as primeiras etapas de uma pesquisa e são frequentemente representados por diagramas, que auxiliam na visualização das interações entre variáveis.



### 3.3.2 MODELO LÓGICO E FÍSICO

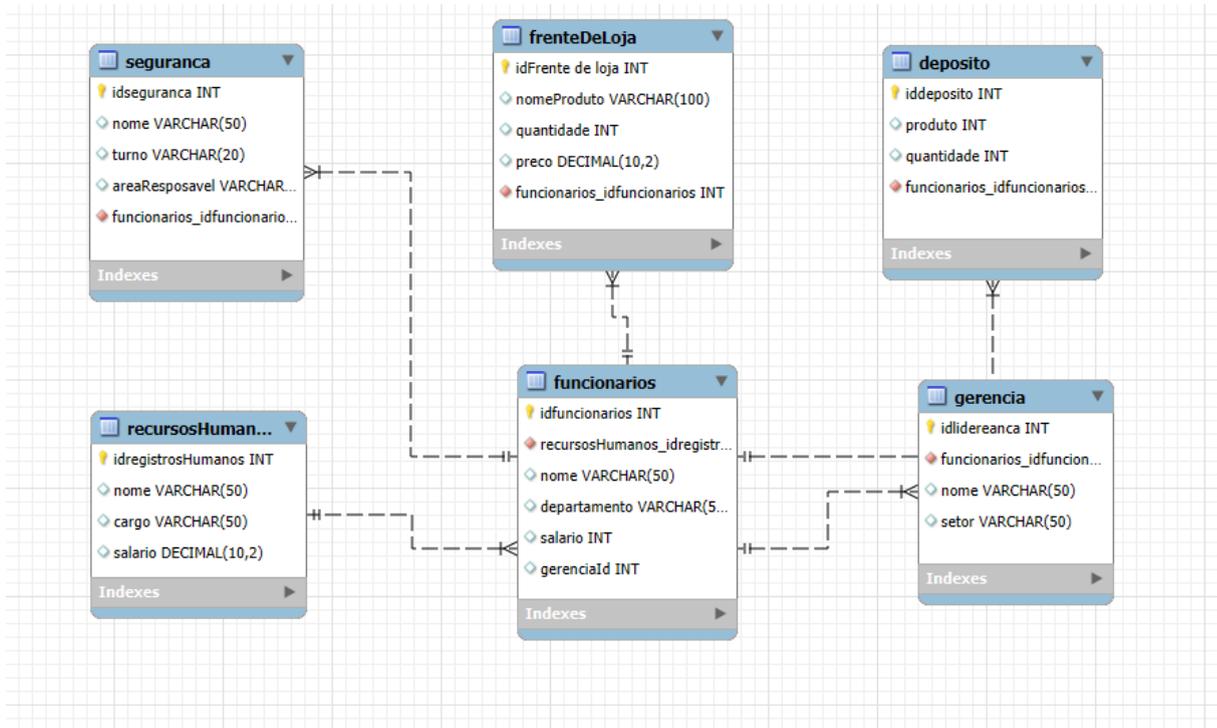
## Sistema de Gerenciamento de Banco de Dados (SGBD)

Um Sistema de Gerenciamento de Banco de Dados (SGBD) é um software que permite criar, manipular e gerenciar bancos de dados de forma eficiente e segura. Ele fornece uma interface para realizar operações fundamentais, conhecidas como CRUD (Create, Read, Update, Delete), que são indispensáveis para a gestão de dados de qualquer sistema (DATE, 2004). No caso do projeto em questão, o SGBD **MySQL** foi escolhido por ser um sistema gratuito, de código aberto e amplamente utilizado na indústria e na academia. Segundo Silberschatz, Korth e Sudarshan (2011), o MySQL é um dos SGBDs mais populares no mundo devido à sua capacidade de lidar com grandes volumes de dados e múltiplas conexões simultâneas, oferecendo alto desempenho e confiabilidade.

### Linguagem SQL e Comandos Básicos

A SQL (Structured Query Language) é a linguagem padrão para manipulação de dados em bancos de dados relacionais. Ela permite definir a estrutura do banco de dados, inserir dados e realizar consultas complexas. Conforme descrito por Coronel, Morris e Rob (2017), a SQL é dividida em sub-linguagens.

A construção e manipulação de um banco de dados relacional no MySQL permite aos estudantes consolidar os conceitos fundamentais de banco de dados e explorar ferramentas e comandos práticos de SQL. Essa experiência oferece uma base sólida para a criação de sistemas que dependem de dados organizados e acessíveis, preparando-os para desafios mais complexos em desenvolvimento de software e ciência de dados.



### 3.3.3 SQL

#### Importância dos Comandos Básicos no Desenvolvimento de Sistemas

O uso dos comandos básicos de SQL é essencial para a criação de sistemas robustos que dependem de bancos de dados. Em um projeto prático, esses comandos permitem popularizar o banco com dados iniciais, modificar registros conforme necessário e criar consultas que auxiliam na construção das funcionalidades do sistema. Durante as etapas de teste, os comandos **INSERT**, **UPDATE**, **DELETE** e **SELECT** permitem que os estudantes simulem operações reais, testando o comportamento do sistema diante de dados novos, alterados e excluídos. Além disso, essas operações serão frequentemente utilizadas nas unidades de programação do sistema, integrando o banco de dados ao código da aplicação e permitindo a comunicação com a interface do usuário.

A compreensão desses comandos básicos e o domínio da SQL formam a base para a construção de sistemas de informação que dependem de um banco de dados eficiente e bem estruturado. No ambiente acadêmico, essa prática também ajuda os alunos a desenvolver uma compreensão sólida sobre como dados podem ser organizados e gerenciados para servir a diferentes necessidades de um sistema, reforçando habilidades essenciais para futuros projetos e desafios no campo de desenvolvimento de software.

- ```
-- Inserção de Dados de Exemplo para Testar as Consultas
```
- **INSERT INTO** `seguranca` (`nome`, `turno`, `areaResponsavel`) **VALUES**  
 ('Carlos Silva', 'Noturno', 'Entrada'),  
 ('Ana Santos', 'Diurno', 'Caixa'),  
 ('João Pereira', 'Noturno', 'Estoque'),  
 ('Marina Lopes', 'Diurno', 'Corredores'),  
 ('Felipe Mendes', 'Noturno', 'Saída');
  - **INSERT INTO** `frenteDeLoja` (`nomeProduto`, `quantidade`, `preco`) **VALUES**  
 ('Produto A', 100, 10.00),  
 ('Produto B', 50, 20.00);
  - **INSERT INTO** `deposito` (`produtoId`, `quantidade`) **VALUES**  
 (1, 200),

### 3.4 BUSINESS INTELLIGENCE

***POWER BI:***



Power BI é uma ferramenta de análise e visualização de dados da Microsoft que permite criar relatórios interativos e dashboards dinâmicos. Ele facilita a transformação de dados brutos em insights visuais claros, auxiliando empresas na tomada de decisões informadas e ágeis.

#### 3.4.1 ORGANIZAÇÃO E IDENTIFICAÇÃO DAS INFORMAÇÕES

A maior vantagem que o POWERBI consegue te trazer é a organização de informações, com ela você consegue desenvolver tudo aquilo que seu cliente que tenha solicitado um banco foi sugerido.

E com isso você tem uma amplitude de serviço sobre o caso, no caso do nosso PI, uma das grandes partes que fizemos utilização dele foi pelo fato do controle e análise de dados do Supermercado Paraíso.

Pois o POWERBI, nos possibilita, que consultemos entrada e saída e até mesmo quantidade de carga e descarga de produto, e com isso temos uma grande base de dados gerado pelos números requeridos, pela curetagem durante a empresa, e com isso iniciamos as atividades dentro do “BI”.

Onde nos colocamos à disposição de fazer um dashboard no qual fosse útil e que ficasse agradável para o cliente que estaríamos entregando este produto, com os dados requeridos e utilizados, de acordo com as normas da empresa.

### **3.4.2 MANIPULAÇÃO E ANÁLISE DE DADOS**

A manipulação e análise de dados são processos fundamentais no contexto de Business Intelligence (BI), onde o objetivo principal é transformar dados brutos em informações estratégicas (Kimball & Ross, 2013). No Power BI, ferramenta de análise de dados da Microsoft, esses processos são essenciais para extrair insights que sustentem a tomada de decisão nas organizações.

#### **Manipulação de Dados**

A manipulação de dados, segundo Inmon (2005), refere-se ao processo de extrair, transformar e carregar (ETL) dados de diferentes fontes, ajustando-os para uma análise coerente. No Power BI, o Power Query atua como ferramenta de ETL, permitindo a limpeza, transformação e enriquecimento de dados, que são essenciais para a análise correta (Marco, 2000). Esse processo inclui operações como remoção de dados irrelevantes, tratamento de valores nulos e padronização de tipos de dados (Kimball & Ross, 2013).

#### **Modelagem de Dados**

A modelagem de dados, conforme apresentado por Silberschatz, Korth e Sudarshan (2011), envolve a organização dos dados em um formato que facilite a análise e consulta. No Power BI, essa etapa ocorre por meio da criação de relacionamentos entre tabelas e da utilização de DAX (Data Analysis Expressions), uma linguagem desenvolvida para cálculos complexos. DAX permite a criação de medidas e colunas calculadas, habilitando análises

detalhadas, como cálculos temporais e condicionais, essenciais para o processo decisório (Ferrari & Russo, 2015).

### Visualização e Análise dos Dados

A visualização é uma etapa crucial, conforme defendido por Few (2009), pois facilita a compreensão de dados complexos e auxilia na interpretação de insights. No Power BI, a transformação de dados em gráficos e painéis interativos promove uma análise intuitiva e rápida. Tufte (2001) também destaca que uma boa visualização é fundamental para comunicar insights e facilitar a tomada de decisões.

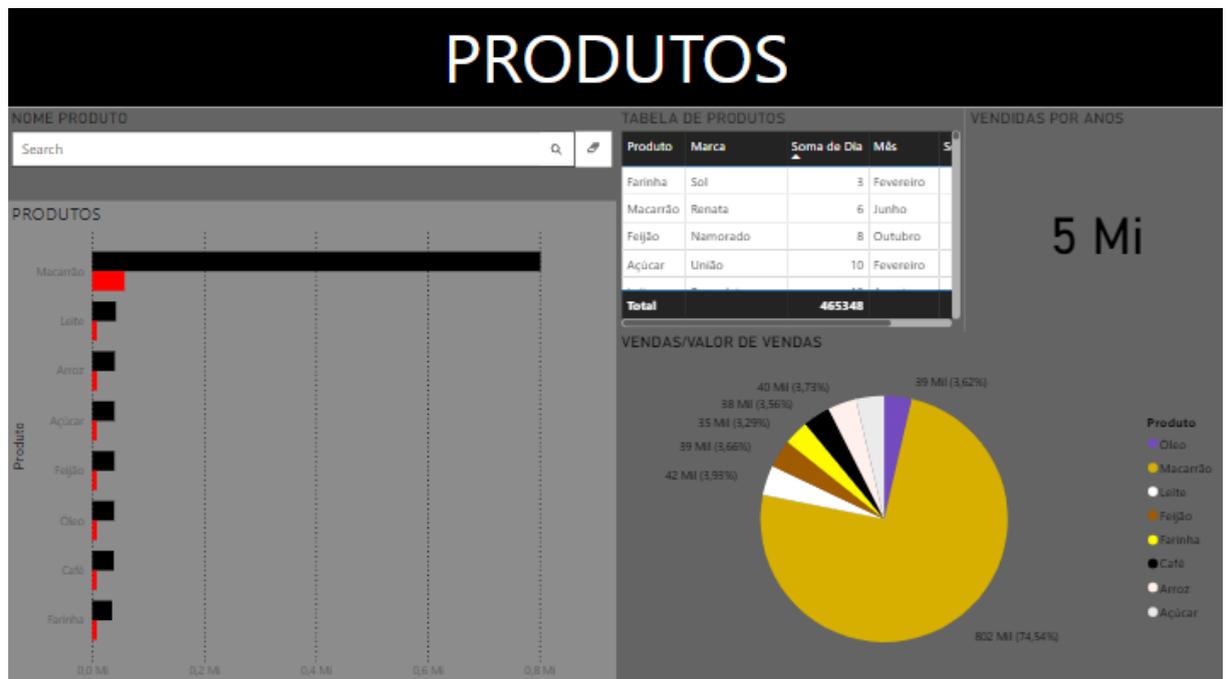
### Atualização e Automação

A atualização e automação de dados, mencionadas por Kimball e Ross (2013), são diferenciais importantes no contexto de BI. O Power BI permite a automação da atualização de relatórios, integrando fontes de dados em tempo real, o que assegura que as análises estejam sempre atualizadas e refletem o cenário mais recente da organização (Inmon, 2005).

### Conclusão

A manipulação e análise de dados no Power BI, embasadas nas referências teóricas, permitem um fluxo completo de ETL, modelagem e visualização, conforme Kimball e Ross (2013). Esse processo garante que dados sejam convertidos em insights estratégicos, apoiando as organizações na tomada de decisões informadas e ágeis.

### 3.4.3 CRIAÇÃO DE MODELOS DE ANÁLISE DE DADOS



## 3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS

A Formação para a Vida, como parte do Projeto Pedagógico de Formação por Competências da UNIFEOB, inclui o tema "Gerenciando Finanças", com o objetivo de capacitar os alunos em habilidades práticas de gestão financeira. Essa formação é voltada para o desenvolvimento de competências que têm aplicação direta na vida cotidiana e no mercado de trabalho. A proposta busca ensinar aos participantes como gerenciar suas finanças pessoais de forma eficiente, abordando temas como controle de orçamento, planejamento financeiro, redução de gastos e investimentos. O foco é proporcionar aos alunos o conhecimento necessário para tomar decisões financeiras conscientes e responsáveis, tanto para o seu bem-estar individual quanto para o benefício da sociedade. Ao integrar a extensão universitária, o conteúdo busca ser útil e relevante para a comunidade, contribuindo para a formação de cidadãos mais preparados financeiramente.

### 3.5.1 GERENCIANDO FINANÇAS

A educação financeira vai além de apenas saber gastar; inclui compreender como funciona a economia, como gerir os recursos de forma consciente e como tomar decisões que tenham um impacto positivo no futuro financeiro.

Para começar é fundamental compreender os conceitos básicos, como o conceito de receitas, despesas, poupanças, investimentos e sobretudo a importância de um orçamento. O orçamento é uma ferramenta essencial para quem quer controlar os gastos e evitar surpresas no final do mês. É a base de todo planejamento financeiro porque permite visualizar com clareza suas fontes de insumos (como salários e outras receitas) e saídas (como contas, dívidas e outras despesas). Quando essas informações estão bem organizadas fica muito mais fácil identificar onde você pode cortar custos ou, pelo contrário, onde você pode investir mais para atingir seus objetivos. Depois de ter um orçamento, você começa a entender como seu dinheiro funciona ao longo do tempo e, como resultado, fica mais fácil tomar decisões de economia. Por exemplo, você pode perceber que gasta muito em algo que não traz nenhum benefício real, como certos tipos de hobbies ou compras por impulso. Ao perceber essas tendências, você poderá ajustar seus hábitos e priorizar o que realmente importa para seu bem-estar e objetivos financeiros.

Além disso, um orçamento bem estruturado também ajuda no planejamento do futuro. Isto serve não só para controlar o presente, mas também para pensar a longo prazo. Quer você sonhe em viajar, comprar um imóvel ou até mesmo garantir sua aposentadoria, um orçamento bem planejado lhe dará o direcionamento e as ferramentas necessárias para atingir esses objetivos. Isto inclui aprender a poupar regularmente, compreender a importância de ter um fundo de emergência e, finalmente, aprender como investir para multiplicar o seu dinheiro.

Dívidas são obrigações financeiras contraídas por meio de empréstimos ou parcelamentos de compras, e os juros compostos fazem com que o valor devido cresça rapidamente porque os juros são calculados sobre o valor original e os juros acumulados. Para evitar o acúmulo de dívidas, é importante tomar cuidados como evitar pagamentos desnecessários, pagar à vista sempre que possível e quitar integralmente o cartão de crédito, pois as taxas de juros do crédito rotativo são muito altas. Ao recorrer ao crédito, é fundamental avaliar a sua capacidade de pagamento antes de contrair novas dívidas.

Existem diferentes tipos de empréstimos, onde o crédito consignado é uma opção útil, com taxas de juros mais baixas, principalmente para quem está oficialmente empregado ou aposentado. Para quem já está endividado, uma alternativa é negociar suas dívidas com os credores ou buscar a portabilidade de crédito para transferir a dívida para uma instituição que

ofereça melhores condições. Se você possui um fundo de emergência, é recomendável usar parte dele para quitar dívidas com juros elevados, o que pode resultar em uma economia significativa ao longo do tempo.

Os empréstimos pessoais são comuns, mas podem ter taxas de juros mais altas, enquanto os empréstimos consignados, mais úteis para quem está formalmente empregado ou aposentado, oferecem taxas de juros mais baixas porque o valor é deduzido diretamente do seu pagamento. Por exemplo, um empréstimo consignado pode ter juros de 1,5% ao mês, enquanto um empréstimo pessoal pode ser de até 3%, o que gera uma economia significativa na hora de escolher um empréstimo consignado. O texto apresenta alternativas para quem está endividado. Uma opção é negociar diretamente com o credor melhores condições, como taxas de juros mais baixas ou prazos de pagamento mais longos. Outra solução é a portabilidade do empréstimo, que permite transferir a dívida para uma instituição que ofereça melhores taxas. Além disso, é aconselhável usar seu fundo de emergência para saldar dívidas com juros elevados. Por exemplo, transferir uma dívida de cartão de crédito de R\$10 mil com juros de 15% para um empréstimo com juros de 3% pode ser uma grande economia.

A importância de estabelecer metas financeiras claras para realizar sonhos. Ao estabelecer metas claras, com prazos e valores mensais, você pode se planejar e evitar dívidas inesperadas. Sugere-se o método SMART para tornar as metas mais eficazes: elas devem ser específicas, mensuráveis, alcançáveis, relevantes e oportunas. Quando o objetivo envolve mais pessoas, como numa família, é essencial que todos participem ativamente e contribuam para o objetivo. A comunicação aberta e as recompensas por atingir a meta ajudam a manter a motivação. O exemplo de uma poupança familiar para criar um fundo de emergência mostra como pequenos esforços conjuntos podem fazer a diferença.

### **3.5.2 ESTUDANTES NA PRÁTICA**

## *FORMAÇÃO PRA VIDA: (finanças)*

1.

REGULARMENTE REVISAR  
E ATUALIZAR SEU PLANO  
FINANCEIRO.

2.

ESTABELECEER METAS  
FINANCEIRAS  
SIGNIFICATIVAS

3.

CRIAR UM  
ORÇAMENTO E  
SEGUI-LO

4.

ECONOMIZAR PARA  
EMERGÊNCIAS

5.

VIVER DENTRO DE SUAS  
POSSIBILIDADES E EVITAR  
DÍVIDAS DESNECESSÁRIAS

## **4. CONCLUSÃO**

Nesta parte deve ser feita uma conclusão do PI, descrevendo os principais pontos abordados, as dificuldades encontradas e outras informações que se julgarem relevantes.

Não se esqueça de revisar os textos, corrigir os erros de digitação/ortografia, fazer uma última conferência na formatação.

Também é preciso atualizar o sumário, isso pode ser feito automaticamente, basta clicar sobre ele e em seguida no botão que aparecerá à esquerda: “Atualizar sumário”.

## REFERÊNCIAS

CODD, E. F. A relational model of data for large shared data banks. *Communications of the ACM*, v. 13, n. 6, p. 377-387, 1970.

CORONEL, C.; MORRIS, S.; ROB, P. *Database Systems: Design, Implementation, and Management*. 13. ed. Boston: Cengage Learning, 2017.

DATE, C. J. *An Introduction to Database Systems*. 8. ed. Boston: Addison Wesley, 2004.

MILES, M. B.; HUBERMAN, A. M. *Qualitative data analysis: an expanded sourcebook*. 2nd ed. Thousand Oaks, CA: Sage Publications, 1994.

RICHARDSON, R. J. *Pesquisa social: métodos e técnicas*. 3. ed. São Paulo: Atlas, 2012.

YIN, R. K. *Estudo de caso: planejamento e métodos*. 5. ed. Porto Alegre: Bookman, 2016.

Essa parte está reservada para as referências, as quais devem estar metodologicamente discriminadas em ordem alfabética e corresponder às citações realizadas ao longo dos textos.

A utilização da metodologia científica é obrigatória e deve ser utilizado o Manual UNIFEOP para Trabalhos Acadêmicos ou as Normas da ABNT.

## **ANEXOS**

Essa parte está reservada para os anexos, caso houver, como figuras, organogramas, fotos etc.

O estudante também deve enviar pela Intranet o Relatório Final das Atividades de Extensão, conforme modelo a seguir.



## RELATÓRIO FINAL DAS ATIVIDADES DE EXTENSÃO

### 1. IDENTIDADE DA ATIVIDADE

**RELATÓRIO:** Desenvolvimento de soluções console integradas para educação, sustentabilidade, inclusão social e empreendedorismo

**CURSO:** Análise e Desenvolvimento de Sistemas

**MÓDULO:** Modelagem e Desenvolvimento de Sistemas

**PROFESSOR RESPONSÁVEL:** Mariângela Martimbianco Santos

**ESTUDANTE:** José Eduardo de Paula Sorg

**PERÍODO DE REALIZAÇÃO:** 08/2024 a 12/2024

### 2. DESENVOLVIMENTO

**Contextualização:** A proposta do projeto é desenvolver uma aplicação via console que integre conceitos de Business Intelligence, Programação Orientada a Objeto, Lógica de Programação e Modelagem de Dados, visando criar soluções inovadoras para promover a educação e capacitação, sustentabilidade ambiental, inclusão social, tecnologia e inovação, desenvolvimento rural e empreendedorismo e economia solidária. O projeto busca proporcionar uma experiência prática e interdisciplinar, para capacitar a utilização da tecnologia de forma eficiente e criativa para resolver problemas reais e contribuir para um desenvolvimento sustentável e inclusivo. A empresa beneficiada pelo projeto foi: Supermercado Paraíso

**Desafio:** No projeto, enfrentamos alguns desafios como alinhar a equipe, lidar com atrasos, gerenciar o que cada um iria fazer, manter a qualidade da entrega, adaptar-se a mudanças e superar limitações técnicas, exigindo muita cooperação e flexibilidade ao decorrer do desenvolvimento do projeto.

**Cronograma das Ações:** Os encontros aconteceram de sexta-feira, nos seguintes dias: 09 de agosto de 2024; 16 de agosto de 2024; 23 de agosto de 2024; 30 de agosto de 2024; 06 de setembro de 2024; 13 de setembro de 2024; 20 de setembro de 2024; 27 de setembro de 2024; 04 de outubro de 2024; 11 de outubro de 2024; 18 de outubro de 2024; 25 de outubro de 2024; 01 de novembro de 2024; 08 de novembro de 2024; 15 de novembro de 2024; 22 de novembro de 2024; 29 de novembro de 2024; 06 de dezembro de 2024; 13 de dezembro de 2024; 20 de dezembro de 2024; 27 de dezembro de 2024.

**Síntese das Ações:** Live de apresentação do cronograma dos Projetos Integrados; Organização e definição das equipes junto aos professores orientadores; Definição e apresentação das informações sobre a empresa a ser utilizada no projeto; Instruções sobre as questões metodológicas de referências, citações bibliográficas e formatação utilizadas no projeto de acordo com as Normas da ABNT; Período destinado ao desenvolvimento do projeto; Instruções sobre a atividade relacionada ao eixo de Formação para a Vida; Entrega da Formação para a vida (1ª entrega). Tópico 3.5 do Projeto Integrado; Entrega parcial para verificação da formatação do trabalho seguindo as normas da ABNT. (2ª entrega); Preenchimento do formulário sobre trabalho em equipe; Instruções sobre a elaboração e critérios de avaliação das apresentações dos projetos; Live sobre o preenchimento do relatório de extensão para entrega via sistema e início das entregas dos Projetos em PDF pelo Classroom; Entregar parte escrita completa com citações, referências e textos nas normas da ABNT; Preenchimento do segundo

formulário sobre trabalho em equipe; Apresentações dos Projetos Integrados e período de retificações dos projetos reprovados.

**a. Aspectos positivos:** Aprender com este projeto foi uma experiência muito enriquecedora. Além de aprimorar habilidades técnicas, pudemos melhorar a forma de trabalhar em equipe e a gestão de nosso tempo. Enfrentar desafios inesperados nos ensinou a pensar de forma mais criativa e a nos adaptarmos com mais facilidade. No fim, a sensação de ter concluído algo importante nos deu mais confiança e motivação para continuar aprendendo e buscando novos desafios.

**b. Dificuldades encontradas:** Durante o projeto, tivemos dificuldades para organizar o tempo e alinhar a equipe, já que cada um tinha sua própria rotina, o que dificultou a conciliação de agendas e causou alguns atrasos. Além disso, enfrentamos desafios técnicos inesperados que demoraram mais para serem resolvidos do que o previsto.

**c. Resultados atingidos:** Foi feita uma identidade na empresa com Power BI, para planilhar todos os produtos com a relação de entrada e saída. Buscando trazer melhoras tanto para o proprietário quanto para os clientes para que nunca falte um produto e ambos consigam o que desejam.

**d. Sugestões / Outras observações:** Incluir sugestões e/ou observações sobre o projeto (opcional).

### 3. EQUIPE DOS ESTUDANTES NO PROJETO

|              |                                        |
|--------------|----------------------------------------|
| RA: 24000526 | NOME: Euller Sodré Terciano            |
| RA: 24000461 | NOME: Gustavo Oliveira Luna Nascimento |
| RA: 24000304 | NOME: José Eduardo de Paula Sorg       |
| RA: 24000776 | NOME: Miguel Turatti Quirino           |
| RA: 24000463 | NOME: Rafael Modena Emygdio            |