



**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**

**AGRO PECUÁRIA VALE DO RIO PARDO**

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**

**AGRO PECUÁRIA VALE DO RIO PARDO**

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:

Aline de Souza Fragoso, RA 24000921

Anthony Ferreira, RA 24000150

Grazielly Zambello Fuzato, RA 24000211

João Gabriel Fialho Mazzi de Paiva, RA 24000020

Mary Emanuele Morais, RA 24001844

SÃO JOÃO DA BOA VISTA, SP  
NOVEMBRO 2024

# SUMÁRIO

1. INTRODUÇÃO	5
2. DESCRIÇÃO DA EMPRESA	6
3. PROJETO INTEGRADO	7
3.1 PROGRAMAÇÃO ORIENTADA A OBJETO	7
3.1.1 CLASSES E OBJETOS	7
3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.	8
3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS	8
3.2 LÓGICA DE PROGRAMAÇÃO	9
3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE	9
3.2.2 DESENVOLVIMENTO DE APLICAÇÕES	9
3.2.3 IMPLEMENTAÇÃO E VALIDAÇÃO	10
3.3 MODELAGEM DE DADOS	11
3.3.1 MODELO CONCEITUAL	11
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	13
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	14
3.5.2 ESTUDANTES NA PRÁTICA	15
4. CONCLUSÃO	16
REFERÊNCIAS	17
ANEXOS	18

# 1. INTRODUÇÃO

O nosso projeto integrado será a criação de um Software de Avaliação de Condições de Solo, que terá como objetivo trazer uma informação correta e segura para possíveis compradores que não tem um conhecimento aprofundado sobre o que necessitam para seus respectivos locais de produção agrícola. Inicialmente o nosso software terá um banco de dados com informações dos valores ideais de certos nutrientes para tipos específicos de planta, e com os preços de produtos específicos de correções para problemas futuros. Com esse processo inicial da empresa pronto, o cliente poderá trazer uma amostra de solo para ser utilizada, informando o tamanho do terreno a ser utilizado como base para a amostra entregue. Com os resultados, o sistema irá dizer quais são as deficiências do solo e já dirá quais seriam os produtos ideais para que uma correção seja feita, calculando assim, o valor estimado para que o solo vá para o seu tipo ideal. Com todo esse processo, clientes potenciais terão uma segurança enorme e um alicerce bem fundamentado de confiança, para assim conseguir uma relação saudável de compras.

**Desenvolvimento Rural:** Sendo um grande avanço de tecnologia para pequenos produtores rurais que não teriam condições costumeiras de fazer um teste desse tipo, atribuindo também valor à loja em questão por um serviço de alta qualidade que facilita a tomada de decisão.

## **2. DESCRIÇÃO DA EMPRESA**

A empresa com razão social Agro Pecuária Vale do Rio Pardo registrada sob o CNPJ 50.072.693-0001/04 que exerce suas atividades na Avenida José Gatto, 500 no ramo da Agropecuária, possui um total de 10 funcionários - desconsiderando os donos - e apresenta como ponto problema, a necessidade de aumento de vendas de produtos relacionados a suplementação do solo. Já a motivação para a implantação de nosso protótipo é a ideia dada pela empresa de utilizar um software como potencializador de vendas para clientes da faixa etária de 20 a 80 anos, pois, assim, os clientes terão uma maior confiança na escolha do produto. Tudo isso graças à funcionalidade principal do projeto idealizado: um avaliador de solo e indicador de nutrientes que impulsionará a empresa na conquista de possuir mais um produto principal, além das ferramentas da marca STIHL, produtos para pet e produtos de jardinagem.

Todas as informações citadas acima foram disponibilizadas pelo Dono da empresa Agro Pecuária Vale do Rio Pardo: Alberto José Ferreira.

## 3. PROJETO INTEGRADO

### 3.1 PROGRAMAÇÃO ORIENTADA A OBJETO

A Agro Pecuária Vale do Rio Pardo pode se beneficiar enormemente da Programação Orientada a Objetos (POO) ao desenvolver um software que organize e otimize suas operações, especialmente no que diz respeito à avaliação e correção de nutrientes do solo. Através de classes e objetos, a empresa pode criar uma solução robusta para aumentar suas vendas de produtos relacionados à suplementação de solo, melhorando a confiança dos clientes nas compras.

#### 3.1.1 CLASSES E OBJETOS

No projeto, serão criadas classes que representam entidades relevantes para a Agro Pecuária. Por exemplo:

Classe NutrientEvaluation: Representa a avaliação dos nutrientes do solo, com atributos que definem os níveis ideais de nutrientes e métodos que avaliam as condições do solo e calculam as correções necessárias.

Classe CultureEvaluation: Responsável por gerenciar informações sobre diferentes culturas, como milho e soja, e avaliar suas necessidades específicas de nutrientes.

Classe CostCalculator: Calcula o custo das correções de nutrientes, facilitando a gestão financeira dos investimentos necessários.

A partir dessas classes, serão instanciados objetos que refletem a realidade da empresa, como avaliacao\_nutrientes, para realizar avaliações específicas do solo.

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

Atributos: Os atributos nas classes, como os níveis ideais de nutrientes em NutrientEvaluation, armazenam informações cruciais para a avaliação do solo.

Métodos: Os métodos em cada classe, como avaliar\_nutrientes() e calcular\_custo(), implementam as ações que a Agro Pecuária precisa para gerir seu solo e produtos de maneira eficiente.

Encapsulamento: Segundo a apostila Python e Orientação a Objeto do Curso PY-14 da Caelum: “Encapsular é fundamental para que o sistema seja suscetível a mudanças:

não precisamos mudar uma regra de negócio em vários lugares, mas sim em apenas um único lugar.” Por exemplo, os métodos de avaliação de nutrientes podem ser públicos, enquanto os atributos que armazenam dados sensíveis devem ser privados, garantindo que apenas métodos da classe possam acessá-los e modificá-los.

Herança: Segundo a apostila Python e Orientação a Objeto do Curso PY-14 da Caelum: “herança é uma relação entre classe 'mãe' e classe 'filha’” que herda os atributos da classe mãe. Por exemplo, se houver um tipo específico de cultura que requer cuidados especiais, pode-se criar uma classe `CulturaEspecific` que herde de `CultureEvaluation`.

Polimorfismo: Métodos como `avaliar_nutrientes()` podem ser sobrescritos em subclasses que tratam de diferentes tipos de culturas, permitindo que a Agro Pecuária ajuste suas avaliações com base nas necessidades específicas de cada tipo de solo e cultura.

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

Métodos Estáticos: Métodos estáticos, como `calcular_total_custos()`, podem ser utilizados para realizar operações que não dependem de instâncias, oferecendo funcionalidades utilitárias.

Métodos Públicos: Os métodos públicos podem ser aplicados para métodos como `avaliar_para_cultura()`, permitindo que outras partes do sistema interajam com as avaliações e correções.

Métodos Privados: Os métodos privados garantem que informações sensíveis não sejam expostas diretamente, promovendo segurança e integridade dos dados, como, por exemplo, cálculos intermediários que não precisam ser acessados externamente.

## **3.2 LÓGICA DE PROGRAMAÇÃO**

A lógica de programação é um conjunto de fundamentos e conceitos necessários para a aplicação de códigos em estruturas de programação, sendo constituídas de variáveis, condições, operadores e dados. O python oferece uma enorme gama de possibilidades de criação porém sendo necessário um grande cuidado, tendo em vista que podem ser necessárias extensas operações que são fundamentais para todo o geral, assim um erro pode por acabar quebrando toda a sequência de códigos.

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



Os conceitos básicos são denominados de tal forma por serem alicerces de toda programação. Praticamente em todo código você terá que ter um amplo conhecimento de como se interpretar variáveis, como transformar tipos de dados e como denominar operadores para certas funções. Com uma boa base constituída desses alicerces é possível ir além para algoritmos com grande extensão ou com uso aprofundado dessas variáveis básicas.

### **3.2.2 DESENVOLVIMENTO DE APLICAÇÕES**

Para o desenvolvimento de nosso Software de Avaliação de Condições de Solo, nós iremos precisar de forma básica de um banco de dados com dados de referência com diferentes estatísticas fundamentadas, para que desta forma o sistema possa conferir os valores de cada dado para trazer o resultado de maneira prática e eficiente para saber a qualidade de solo, quais mercadorias para melhoria de solo terão de ser compradas e o preço das mercadorias que terão de ser compradas; já agora para entrada do próprio usuário será necessário uma amostra de solo com no mínimo 500 gramas, entrada de qual o tamanho do solo que será corrigido de acordo com a amostra dada, o tipo de solo e o clima do local. Com tais informações e dados será possível categorizar de forma apropriada e assertiva em uma análise.

### 3.2.3 IMPLEMENTAÇÃO E VALIDAÇÃO

```

def main():
    ..
    culture_evaluation = CultureEvaluation()

    # Escolha a cultura
    cultura = input("Escolha uma cultura entre: milho, soja, sorgo, trigo: ").lower()

    # Coleta dados da área
    area = float(input("Digite a área em metros quadrados: "))

    # Coleta dados do solo
    solo_atual = {
        'n': float(input("Digite o nível de Nitrogênio (N): ")),
        'p': float(input("Digite o nível de Fósforo (P): ")),
        'k': float(input("Digite o nível de Potássio (K): ")),
        'mo': float(input("Digite o nível de Matéria Orgânica (MO): ")),
        'cu': float(input("Digite o nível de Cobre (Cu): ")),
        'fe': float(input("Digite o nível de Ferro (Fe): ")),
        'mn': float(input("Digite o nível de Manganês (Mn): ")),
        'zn': float(input("Digite o nível de Zinco (Zn): ")),
        'ca': float(input("Digite o nível de Cálcio (Ca): ")),
        'mg': float(input("Digite o nível de Magnésio (Mg): ")),
        'ph': float(input("Digite o nível de pH: "))
    }

    # Avaliar e calcular correções para a cultura escolhida
    estados, correcoes, custo = culture_evaluation.avaliar_para_cultura(cultura, solo_atual, area, nutrient_evaluation, cost_calculator)

    if estados is not None:
        # Exibe as condições do solo
        print("\nCondições do solo:")
        for elemento, estado in estados.items():
            print(f"{elemento}: {estado}")

        # Exibe as sugestões de correção
        print("\nSugestões de correção:")
        for elemento, correcao in correcoes.items():
            print(f"Adicionar {correcao:.2f} unidades de {elemento}")

        # Exibe o custo total
        print(f"\nO custo total para corrigir o solo na área de {area:.2f} m² é: R$ {custo:.2f}")

if __name__ == "__main__":
    main()

```

```

class CultureEvaluation:
    def __init__(self):
        self.culturas = {
            'milho': {'N': 20, 'P': 15, 'K': 60, 'MO': 3.5, 'Cu': 1.5, 'Fe': 6, 'Mn': 5, 'Zn': 2, 'Ca': 3, 'Mg': 2},
            'soja': {'N': 30, 'P': 12, 'K': 50, 'MO': 3.0, 'Cu': 1.2, 'Fe': 5.5, 'Mn': 4.5, 'Zn': 1.8, 'Ca': 2.5, 'Mg': 1.5},
            'sorgo': {'N': 25, 'P': 10, 'K': 55, 'MO': 3.2, 'Cu': 1.3, 'Fe': 5.8, 'Mn': 4.8, 'Zn': 1.9, 'Ca': 2.8, 'Mg': 1.8},
            'trigo': {'N': 18, 'P': 14, 'K': 58, 'MO': 3.6, 'Cu': 1.4, 'Fe': 6.1, 'Mn': 5.2, 'Zn': 2.1, 'Ca': 3.2, 'Mg': 2.0}
        }

    def escolher_cultura(self, cultura_nome):
        return self.culturas.get(cultura_nome.lower(), None)

    def avaliar_para_cultura(self, cultura_nome, solo_atual, area, nutrient_evaluation, cost_calculator):
        requisitos_cultura = self.escolher_cultura(cultura_nome)
        if requisitos_cultura is None:
            print("Cultura inválida!")
            return

        estados, correcoes = nutrient_evaluation.avaliar_nutrientes(**solo_atual)
        correcoes.pop('pH', None) # Remover pH das correções

        custo = cost_calculator.calcular_custo(correcoes, area)
        return estados, correcoes, custo

```

```

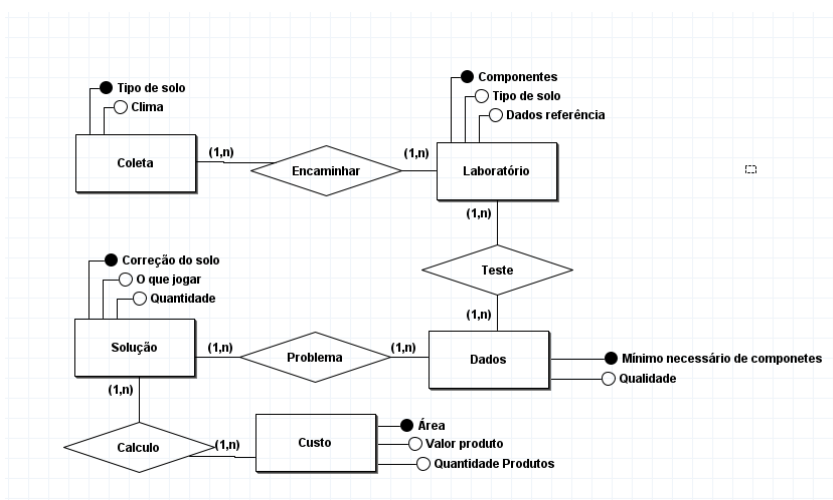
1 class NutrientEvaluation:
2     def __init__(self):
3         self.idealais = {
4             'N': 15.0, 'P': 10.0, 'K': 50.0, 'MO': 3.0,
5             'Cu': 1.0, 'Fe': 5.0, 'Mn': 5.0, 'Zn': 1.0,
6             'Ca': 2.0, 'Mg': 1.0, 'pH': 6.5
7         }
8
9     def avaliar_nutrientes(self, n, p, k, mo, cu, fe, mn, zn, ca, mg, ph):
10        estados = {
11            'N': "baixo" if n < 10.0 else "ideal" if n <= 20.0 else "alto",
12            'P': "baixo" if p < 5.0 else "ideal" if p <= 15.0 else "alto",
13            'K': "baixo" if k < 40.0 else "ideal" if k <= 60.0 else "alto",
14            'MO': "baixo" if mo < 2.0 else "ideal" if mo <= 5.0 else "alto",
15            'Cu': "baixo" if cu < 0.5 else "ideal" if cu <= 2.0 else "alto",
16            'Fe': "baixo" if fe < 3.0 else "ideal" if fe <= 7.0 else "alto",
17            'Mn': "baixo" if mn < 3.0 else "ideal" if mn <= 7.0 else "alto",
18            'Zn': "baixo" if zn < 0.5 else "ideal" if zn <= 2.0 else "alto",
19            'Ca': "baixo" if ca < 1.0 else "ideal" if ca <= 3.0 else "alto",
20            'Mg': "baixo" if mg < 0.5 else "ideal" if mg <= 2.0 else "alto",
21            'pH': "baixo" if ph < 5.5 else "ideal" if ph <= 7.5 else "alto"
22        }
23
24        correcoes = {elemento: max(0.0, self.idealais[elemento] - valor)
25                    for elemento, valor in {'N': n, 'P': p, 'K': k, 'MO': mo, 'Cu': cu,
26                                           'Fe': fe, 'Mn': mn, 'Zn': zn, 'Ca': ca, 'Mg': mg, 'pH': ph}.items()}
27
28        return estados, correcoes
29

```

### 3.3 MODELAGEM DE DADOS

A modelagem de dados é uma técnica fundamental na área de ciência da computação e sistemas de informação, que visa organizar e estruturar os dados de uma organização de forma eficiente e eficaz. Sendo extremamente importante no papel de tomada de decisões dentro das empresas, e contribuindo muito para a melhoria dos processos de gestão e análise de informações.

#### 3.3.1 MODELO CONCEITUAL



### 3.3.2 MODELO LÓGICO E FÍSICO

[WORKBENCH](#) e [MODELO](#)

### 3.3.3 SQL

[BANCO PI](#)

## 3.4 BUSINESS INTELLIGENCE

Business Intelligence refere-se a um conjunto de práticas, tecnologias e ferramentas que visam transformar dados brutos em informações significativas para auxiliar na tomada de decisões empresariais. Este processo envolve a coleta, análise e apresentação de dados, permitindo que as organizações compreendam melhor seu desempenho e identifiquem oportunidades de melhoria. A implementação de B.I. pode incluir painéis, relatórios analíticos e visualizações, possibilitando um acesso mais fácil às informações. Com essa abordagem, empresas podem reduzir custos, aumentar eficiência e impulsionar a competitividade. Assim, o B.I. torna-se uma fundamental ferramenta estratégica no ambiente corporativo contemporâneo.

### 3.4.1 ORGANIZAÇÃO E IDENTIFICAÇÃO DAS INFORMAÇÕES

Para as apresentações de culturas ideais nós teremos como base algumas tabelas de referência com os valores ideais, essa será uma das fundamentações do nosso banco de dados, a partir da coleta de dados sobre as culturas específicas poderemos entender quais são as mais utilizadas e também planejar futuras culturas. Como, por exemplo algumas das culturas são:

Milho | N: 20 | P: 15 | K: 60 | MO: 3.5 | Cu: 1.5 | Fe: 6 | Mn: 5 | Zn: 2 | Ca: 3 | Mg: 2 |

Soja | N: 30 | P: 12 | K: 50 | MO: 3.0 | Cu: 1.2 | Fe: 5.5 | Mn: 4.5 | Zn: 1.8 | Ca: 2.5 | Mg: 1.5 |

Sorgo | N: 25 | P: 10 | K: 55 | MO: 3.2 | Cu: 1.3 | Fe: 5.8 | Mn: 4.8 | Zn: 1.9 | Ca: 2.8 | Mg: 1.8 |

Trigo | N: 18 | P: 14 | K: 58 | MO: 3.6 | Cu: 1.4 | Fe: 6.1 | Mn: 5.2 | Zn: 2.1 | Ca: 3.2 | Mg: 2.0 |

### 3.4.2 MANIPULAÇÃO E ANÁLISE DE DADOS

Banco de Dados está no link a seguir de forma condensada: [x info\\_cliente.xlsx](#)

[x PIBI.xlsx](#) [x PIBI2.xlsx](#)

Nele é possível ser visto os diferentes tipos de cultura de solo, diferentes culturas individuais e as substâncias mais necessárias para o solo. E de acordo com isso o sistema ele irá comparar os níveis dos nutrientes ideais para o cultivo específico com os valores do solo pré-analisado e irá conceder uma tabela de quais são os déficits e quais são os fertilizantes para corrigir os erros.

### 3.4.3 CRIAÇÃO DE MODELOS DE ANÁLISE DE DADOS

Modelo com os dados gerais do projeto [PIBI](#)



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

### 3.5.1 GERENCIANDO FINANÇAS

O tema "Gerenciando Finanças" aborda aspectos essenciais da educação financeira, divididos em quatro tópicos que visam proporcionar aos estudantes uma compreensão mais ampla sobre a administração de recursos pessoais, planejamento financeiro e tomada de decisões econômicas no dia a dia. Abaixo, segue a síntese de cada tópico, com exemplos práticos que podem ser aplicados na vida cotidiana.

**Tópico 1:** Introdução aos conceitos econômicos e financeiros básicos

Este tópico apresenta os princípios fundamentais da economia e das finanças, como a distinção entre receitas e despesas, o conceito de poupança e a compreensão da inflação. A assimilação desses conceitos é essencial para a gestão eficiente do orçamento pessoal.

Exemplo prático: Ao receber um salário mensal de R\$2.000, após o pagamento das despesas fixas (como aluguel, contas e alimentação), um saldo de R\$500 pode ser poupado para emergências ou investido em alternativas financeiras que gerem rentabilidade futura. O conhecimento desses conceitos auxilia no controle financeiro e no planejamento a longo prazo.

**Tópico 2:** Entendendo o ambiente: independência financeira, o valor da riqueza pessoal e o registro diário de finanças

Este tópico discute a importância de atingir a independência financeira, que consiste em ter recursos suficientes para viver sem a necessidade de um trabalho ativo, e enfatiza a relevância do registro diário de receitas e despesas como forma de monitorar e avaliar o crescimento do patrimônio pessoal.

Exemplo prático: Ao realizar um acompanhamento diário das despesas com transporte, alimentação e lazer, é possível identificar padrões de consumo excessivo. Ao reduzir gastos desnecessários, como refeições frequentes fora de casa, os recursos economizados podem ser direcionados para investimentos que contribuirão para a conquista da independência financeira.

**Tópico 3:** Dívidas e juros compostos, opções de empréstimo e alternativas ao endividamento

Neste tópico, são abordados os efeitos dos juros compostos sobre as dívidas, bem como as opções disponíveis para negociação ou obtenção de empréstimos com condições mais vantajosas. Além disso, são discutidas alternativas para pessoas em situação de endividamento.

Exemplo prático: Uma dívida de R\$1.000 em um cartão de crédito, com uma taxa de juros composta de 10% ao mês, pode crescer para cerca de R\$1.771 em seis meses. Para evitar esse aumento exponencial, o indivíduo deve considerar alternativas, como a negociação

da dívida ou a contratação de um empréstimo com taxas de juros mais baixas, a fim de reduzir o montante a ser pago.

**Tópico 4:** Estabelecendo metas para a realização de sonhos e envolvendo o grupo para alcançar objetivos

O quarto tópico foca na definição de metas financeiras claras, como a compra de uma casa, uma viagem ou a realização de um curso, e na importância de envolver grupos ou comunidades no processo de alcance desses objetivos, promovendo colaboração e suporte mútuo.

Exemplo prático: Um grupo de amigos pode formar um fundo coletivo, em que cada integrante contribui com um valor mensal fixo. Esse fundo pode ser utilizado para realizar objetivos em comum, como financiar uma viagem ou apoiar a realização de um curso que beneficie a todos. Esse tipo de colaboração facilita o alcance de metas individuais e coletivas.

### 3.5.2 ESTUDANTES NA PRÁTICA

Logo abaixo temos um vídeo com um banner sobre gerenciamento de finanças dando exemplos práticos de como a falta de conhecimento de tal assunto pode afetar a sua vida e seu cotidiano e também dá pequenas dicas como por exemplo renegociações que por muitas vezes é vista como algo ruim ou uma “última chance” pode ser uma ótima alternativa: [VÍDEO](#)

## 4. CONCLUSÃO

A elaboração do PI visa o desenvolvimento de um software que avalia o solo afim de analisar a composição do mesmo e corrigir os micronutrientes de forma direcionada, tudo isso através de uma venda assertiva de fertilizantes que otimizem o cultivo e ampliem a produção agrícola gerando assim uma relação de confiança entre cliente e vendedor.

Ao longo do projeto nos deparamos com inúmeros problemas, mas principalmente tivemos dificuldades em relação aos valores ideais para os cultivos específicos, como por exemplo achar os valores NPK da soja e do milho. Além disso, ao decorrer das aulas tivemos que nos prontificar a estudar cada vez mais a fundo em diversos pontos como na organização de diferentes classes.

Entretanto agora no fim do projeto podemos ter certeza que abordamos os temas de maneira abrangente porém objetiva e concluímos o nosso avaliador de solo com dados fundamentados.

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

**Caelum Apostila Python e Orientação a Objeto do Curso PY-14**

Acessado no dia 08/11/2024: [LIVRO](#)

## ANEXOS

Cultura	Tipo	Principais Substâncias no Solo	Tipo de Fertilizante
Milho	Grão	Nitrogênio	NPK (Nitrogênio, Fósforo, Potássio)
Arroz	Grão	Fósforo	NPK + Calcário
Feijão	Leguminosa	Potássio	NPK + Fertilizante orgânico
Soja	Leguminosa	Cálcio	NPK + Calcário
Trigo	Grão	Magnésio	NPK + Sulfato de Amônio
Cevada	Grão	Enxofre	NPK + Fertilizante orgânico
Aveia	Grão	Silício	NPK + Calcário
Batata	Tubérculo	Fósforo	NPK + Fosfato natural
Mandioca	Tubérculo	Nitrogênio	NPK + Fertilizante orgânico
Cenoura	Raiz	Cálcio	NPK + Fertilizante de liberação
Beterraba	Raiz	Potássio	NPK + Calcário
Alface	Folhosa	Magnésio	NPK + Fertilizante orgânico
Espinafre	Folhosa	Enxofre	NPK + Fertilizante de liberação
Couve	Folhosa	Silício	NPK + Sulfato de Amônio
Brócolis	Folhosa	Nitrogênio	NPK + Fertilizante orgânico
Repolho	Folhosa	Fósforo	NPK + Calcário
Pimentão	Folhosa	Potássio	NPK + Fertilizante de liberação
Tomate	Fruto	Cálcio	NPK + Sulfato de Amônio
Berinjela	Fruto	Magnésio	NPK + Calcário
Pepino	Fruto	Enxofre	NPK + Fertilizante orgânico
Abobrinha	Fruto	Silício	NPK + Fertilizante de liberação
Abóbora	Fruto	Nitrogênio	NPK + Calcário
Morango	Fruto	Fósforo	NPK + Fertilizante orgânico
Framboesa	Fruta	Potássio	NPK + Sulfato de Amônio
Laranja	Fruta	Cálcio	NPK + Calcário
Limão	Fruta	Magnésio	NPK + Fertilizante orgânico
Maçã	Fruta	Enxofre	NPK + Sulfato de Amônio
Pera	Fruta	Silício	NPK + Calcário
Uva	Fruta	Nitrogênio	NPK + Fertilizante de liberação
Banana	Fruta	Fósforo	NPK + Fertilizante orgânico
Mamão	Fruta	Potássio	NPK + Sulfato de Amônio
Abacaxi	Fruta	Cálcio	NPK + Calcário
Coco	Fruta	Magnésio	NPK + Fertilizante orgânico
Noz	Fruto seco	Enxofre	NPK + Sulfato de Amônio
Castanha	Fruto seco	Silício	NPK + Calcário
Amendoim	Leguminosa	Nitrogênio	NPK + Fertilizante orgânico
Girassol	Oleaginosa	Fósforo	NPK + Sulfato de Amônio
Canola	Oleaginosa	Potássio	NPK + Fertilizante orgânico
Gergelim	Oleaginosa	Cálcio	NPK + Calcário
Linhaça	Oleaginosa	Magnésio	NPK + Fertilizante de liberação
Sorgo	Grão	Enxofre	NPK + Calcário
Milhete	Grão	Silício	NPK + Fertilizante orgânico
Cebola	Bulbo	Nitrogênio	NPK + Fertilizante de liberação
Alho	Bulbo	Fósforo	NPK + Calcário

[PROJETO PARTE 1](#) [PROJETO PARTE 2](#) [PROJETO PARTE 3](#) [PROJETO PARTE 4](#)