**RICARDO TOMPSON FIORAVANTE**

**Analisando Dados com a Linguagem de Programação Python**

Maringá-PR

09/2021

**1. INTRODUÇÃO**

Softwares estão presentes em quase todas as rotinas da nossa vida. Eles são elementos indispensáveis para o progresso humano, o que implica no aumento do interesse por eles, em seus recursos, em suas soluções. Esse interesse tem crescido de maneira significativa recentemente.

Como a humanidade está em constante desenvolvimento, novas demandas surgem a cada dia e um tema atual é a análise de dados. O volume de dados cresce exponencialmente a cada ano. Segundo estimativas, a quantidade de dados produzidos no mundo dobra a cada dois anos. Para utilizar essas informações de forma positiva, é preciso ter conhecimento necessário para lidar efetivamente com elas. Saber operar dados é essencial para o crescimento dos negócios. Conhecer suas características e suas implicações têm grande importância na sociedade atual. Serviços como analytics estão sendo cada vez mais inseridos no portfólio das empresas.

Mas transformar dados brutos em informações úteis tem sido um grande desafio. Muitas ferramentas para essa finalidade são desconhecidas e/ou apresentam dificuldade de utilização pelos usuários. Visando contornar esses obstáculos, novas formas, novas ferramentas para manipulação de dados e elaboração de relatórios surgiram ao longo dos anos, como algumas bibliotecas da linguagem de programação Python, que buscam auxiliar o ser humano nessas tarefas, auxiliando na melhor compreensão dos resultados. Com elas, informações ocultas podem se revelar. Assim, surgem algumas questões, como: Como utilizar Python para explorar um determinado conjunto de dados? Quais bibliotecas do Python podem auxiliar nessas tarefas? Que tipos de gráficos são utilizados com frequência para representar visulalmente os dados? Quais medidas estatísticas são comumente empregadas na análise?

Com o intuito de abordar algumas dessas ferramentas e analisar suas contribuições para a análise de dados, objetivos foram definidos para convergir a esse propósito, sendo: Objetivo geral e objetivos específicos.

**1.2 OBJETIVOS**

Nesta seção, são expostos o objetivo geral e os objetivos especifícos, formulados para resolver o problema levantado neste trabalho.

* **1.2.1 OBJETIVO GERAL:** O objetivo geral deste TCC foi desenvolver um software capaz de manipular dados armazenados em um arquivo de extensão ‘.csv’ (mencionado nos apêndices C e D) com a linguagem de programação Python e extrair computacionalmente insights importantes (como algumas medidas estatísticas) sobre a base de dados, de maneira prática, automática e com agilidade, sem que o usuário precise deduzir expressões lógicas e/ou formulas matemáticas/estatísticas.
* **1.2.2 OBJETIVOS ESPECÍFICOS:** Revisar a literatura pertinente para obter embasamento sobre conceitos-chaves, definições, princípios e implicações, através de pesquisa bibliográfica realizada em livros, artigos científicos e estudos de caso; Levantamento de alguns requisitos de usuários; Escrita e interpretação de um código fonte capaz de sintetizar e satisfazer os requisitos, como por exemplo, exibir algumas medidas estatísticas e alguns tipos de gráficos, de maneira simples e intuitiva.

**2. JUSTIFICATIVA**

A ferramenta mais utilizada para análise de dados ainda é o Microsoft Excel. Mas aprender corretamente muitos de seus recursos demanda tempo e esforço. Geralmente, a elaboração de fórmulas no Excel e uso de seus recursos mais avançados requer conhecimentos que muitos não possuem, dificultando sua utilização. Também pode ocorrer dos usuários usarem incorretamente essa ferramenta, como por exemplo, usando fórmulas incorretas para efetuar determinados cálculos.

Visando contornar essas dificuldades, este trabalho buscou efetuar a análise de dados de forma computacional, com o uso da linguagem de programação Python e algumas de suas bibliotecas. Essas ferramentas práticas facilitam a análise de dados, automatizam muitas tarefas e disponibilizam informações de forma instantânea (quando os usuários precisam). Além disso, gráficos visuais proporcionam compreensão mais direta dos dados quando comparados com conjunto de números. Consequentemente, contribuem para aumentar a performance das pessoas (aspecto muito discutido no âmbito corporativo), além de influenciar diretamente na tomada de decisões/ações.

Exposto a relevância, inicia-se os demais itens relacionados a este trabalho, com detalhamento da situação problema.

**3. METODOLOGIA**

Este trabalho, em se tratando de classificação de pesquisa, é aplicada, pois seu objetivo geral é construir um software.

Quanto aos procedimentos metodológicos, adotou-se uma abordagem quantitativa, já que utiliza métodos e medidas estatísticas com o intuito de explicar os resultados através da quantificação dos dados (ALMEIDA, 2014).

Foi efetuada uma pesquisa sobre as principais medidas da estatística descritiva, bem como sobre os principais gráficos utilizados em análise de dados e sobre algumas bibliotecas da linguagem de programação Python utilizadas para esta finalidade.

Essa pesquisa baseou-se em materiais especializados no tema, principalmente em: McKINNEY(2018), MENEZES (2010) e FACELI (2011).

Efetuou-se algumas tentativas de extração de informações da base de dados descrita no apêndice C e D, com o Microsoft Excel, que é a ferramenta mais popular para esta tarefa.

Em seguida, levantou-se algumas dificuldades em analisar a base de dados em questão com o Microsoft Excel e como contorná-las.

Na sequência, levantou-se alguns requisitos utilizando conceito da metodologia ágil de desenvolvimento Scrum, por meio de histórias de usuários. Após o levantamento, realizou-se priorização dos requisitos para a codificação do código fonte, que foram organizados em artefatos Scrum, como o Backlog do Produto, o Backlog das Sprints, o Scrum Board e o Gráfico de Burndown.

Posteriormente, desenvolveu-se um software capaz de realizar manipulação dos dados e plotagens de gráficos sem a necessidade de trabalho manual e/ou elaboração de fórmulas pelo(s) usuário(s), por meio da escrita e interpretação do código fonte, na linguagem de programação Python.

Por fim, finaliza-se por meio de conclusões sobre a implementação do software, para verificar se ele proporcionou melhoria na análise dos dados.

**3.1 FERRAMENTAS UTILIZADAS**

3.1.1 VS CODE: É editor desenvolvido pela Microsoft. Pode ser utilizado com o Microsoft Windows, MacOS e Linux. Esse editor oferece suporte para depuração do código-fonte escrito em algumas linguagens de programação, como Python. A seguir, segue a configuração da versão utilizada:

* Version: 1.60.2 (user setup)
* Chrome: 91.0.4472.164
* Node.js: 14.16.0
* V8: 9.1.269.39-electron.0
* OS: Windows\_NT x64 10.0.19042

3.1.2 MICROSOFT EXCEL: Popular editor de planilhas, elaborado pela Microsoft. Possui funções para cálculos matemáticos/estáticos e também para construção de gráficos. Foi utilizado para tratar a planilha que serviu de base para este trabalho, comparando os resultados obtidos por meio dele com o software desenvolvido. O produto utilizado foi o Microsoft Excel 365, versão 2108.

3.1.3 Trello: É uma ferramenta online para gerenciamento de projetos e é gratuita. É uma alternativa versátil, que possui interface intuitiva e com vários recursos úteis, como checklists. Organiza os projetos por meio de quadros, similar ao Kanban. Nos quadros são inseridas etiquetas com as tarefas.

**4. REFERENCIAL TEÓRICO**

**4.1 Por que escolher Python?**

Python surgiu em 1991, e atualmente é uma das linguagens de programação interpretadas mais populares do mundo, contando com vastas e ativas comunidades. É uma linguagem de programação considerada simples, sobretudo pela sua sintaxe clara. Apesar dessa característica (simples), ela é uma linguagem com recursos extraordinários. É clara, é objetiva. (Menezes, p. 26)

Ao longo dos anos, as comunidades desenvolveram bibliotecas poderosas para Python, como pandas e matplotlib (que serão tratadas em tópicos posteriores), que tornou Python numa rica alternativa para análise de dados.

Python tem destaque em diversas áreas de TI. Atualmente, é incomum ocorrer disponibilização de alguma biblioteca que não tenha versão em Python (Menezes, p. 26). Dentre as áreas que a utilizam, têm notoriedade: machine learning, ciência de dados e desenvolvimento de produto de software em geral. (McKinney, p. 20)

Python é gratuita, ou seja, é um software livre. Isso é possível devido à colaboração de diversas pessoas e instituições ao redor do mundo, como a Python Foundation. (Menezes, p. 26)

A linguagem de programação Python é utilizada em inúmeros sistemas operacionais, como Microsoft Windows, Mac OS, Linux, entre outros. (Menezes, p. 26)

Outra vantagem é sua legibilidade. Ao contrário de outras linguagens de programação, ela não utiliza ponto e vírgula ou ponto para demarcar o fim de uma instrução. Também não utiliza marcadores (como chaves) no início/fim de um bloco de instruções. Esse aspecto torna os programas mais fácéis de ler. (Menezes, p. 26)

Python é considerada uma linguagem completa, com bibliotecas capazes de se integrar a banco de dados, processar arquivos de extensão XML, desenvolver interfaces gráficas, etc., por meio de código reduzido, o que implica no aumento de produtividade por parte de programadores e redução do números de erros. (Menezes, p. 27)

**4.2 Algumas bibliotecas do Python**

**4.2.1 NumPy**

Numpy é uma biblioteca da linguagem de programação Python para realizar processamento numérico. O termo Numpy é uma abreviação de Numerical Python. Ela oferece um amplo rol de funções e operações que facilitam o processamento de cálculos numéricos, sendo muito utilizada em tarefas matemáticas.

O objeto principal dessa biblioteca é uma forma de Array, um vetor n-dimensional, denominado ndarray, que é um segmento unidimensional contínuo na memória atrelado a um método de indexação que mapeia cada elemento para uma localização na memória do computador. O Array Numpy é uma tabela (comumente números) homogênea, ou seja, os elementos são do mesmo tipo. As dimensões são denominadas eixos.

Seus recursos incluem operações para processar elementos do ndarray, efetuar operações matemáticas entre arrays distintos, gerar números aleatórios, realizar operações de álgebra linear, etc.

**4.2.2 Pandas**

Pandas é uma biblioteca projetada para tornar o trabalho com dados estruturados ou dados tabulares rápido e eficiente. Surgiu em 2010 e desde lá vem auxiliando o Python a ser tornar uma ferramenta robusta para análise de dados.

Seus principais objetos são o Dataframe e as Series. Os Dataframes são estruturas tabulares, possuindo rótulos tanto para linhas quanto para as colunas. As Series também possuem rótulos, e são arrays unidimensionais.

No pandas, existem funcionalidades de indexação que permitem manipular, selecionar e agregar subconjuntos de dados de maneira simples. Devido a essas tarefas serem comuns na análise de dados, pandas é uma biblioteca muito conhecida entre os profissionais dessa área.

**4.2.3 TKinter**

TKinter é uma biblioteca que integra o Python e está embutida na instalação padrão. Possui ferramentas para desenvolvimento de interfaces gráficas do usuário, por meio do pacote tkinter e seus módulos. Qualquer computador que disponha do interpretador Python pode gerar interfaces gráficas com Tkinter.

Um dos principais motivos de sua utilização é a facilidade em acessar seus recursos, além de ser nativa do Python. Além disso, permite inserir widgets (botões, caixas de textos, etc.) na interface com simplicidade.

Outra facilidade do Tkinter é localizar os widgets na interface com comandos simples, para organização dos layouts. Esse módulo oferece os meios: Pack, Grid e Place, todos voltados à geometria e posicionamento dos widgets.

**4.2.4 Matplotlib**

Matplotlib foi criada em John D. Hunter, biólogo e neurocientista. Ela é a mais famosa biblioteca Python para plotagem de dados. Inicialmente foi pensada para fazer plotagens voltadas a publicações, mas atualmente é uma ferramenta amplamente usada, que geralmente se integra bem com demais sistemas.

Essa biblioteca fornece inúmeros recursos para manipulação e visualização de dados. Com ela, gráficos são gerados a partir de conjuntos de dados, facilitando a interpretação.

Os gráficos são customizáveis, permitindo alteração de aspectos visuais, como cores, tamanho das fontes, inserção de legendas, etc. Ela funciona muito bem com os diversos sistemas operacionais e back-ends. Além disso, permite gerar diversos tipos de gráficos, como gráficos de barras, histograma, gráfico de pizza, entre outros.

**4.2.5 Statsmodels**

Statsmodels é uma biblioteca originada a partir dos esforços do professor Jonathan Taylor, do departamento de estatística da Universidade de Stanford. A partir de 2010, o projeto foi expandido.

Ela é elaborada especificamente para cálculos estatísticos. Dentre alguns de seus recursos, estão: modelos de regressão, cálculo de variância, análise sobre dados temporais, visualização de resultados estatísticos etc.

**5. Arquivos CSV**

Dados do tipo planilha (tabulares) são a grande maioria dos dados armazenados em banco de dados relacionais ou arquivos do tipo CSV(da língua inglesa, “Character-separated values”), que são usados para armazenar dados tabulares, em que pode ocorrer de cada coluna ser de um tipo distinto (numérico, string, etc). Além disso, uma grande quantidade de conjunto de dados não-estruturados podem ser convertidos para o formato estruturado (McKinney, p. 20).

Os arquivos CSV são sequências de caracteres. Eles armazenam diversos registros dissociados por quebras de linhas, em que cada registro ocupa uma linha. Os registros possuem campos (um ou mais deles), separados por um delimitador (Character-separated), que podem ser a vírgula, o ponto-vírgula e o tab. Os arquivos separados por vírgula ou ponto-vírgula geralmente têm a extensão CSV, enquanto os separados por tab, a extensão TSV.

**6. Estatística**

A estatística descritiva é uma forma de explorar dados e extrair informações úteis do conjunto de dados. Ela resume de maneira quantitativa as características relevantes de grupos de elementos.

**6.1 Medida de Frequência**

É uma medida que mensura a proporção de vezes que uma característica (atributo) admite um certo valor num conjunto de dados. Aplica-se tanto a valores numéricos, quanto a simbólicos. Como exemplo: Em um conjunto de dados de estudantes, 30% são mulheres.

**6.2 Medidas de Localidade**

São medidas que estabelecem pontos referenciais nos dados e são muito utilizadas para verificar a simetria/assimetria deles. Distinguem-se para valores numéricos e valores simbólicos. No caso de dados simbólicos, a moda é o valor que surge com maior frequência. Por exemplo, a moda do conjunto F = {‘maçã’, ‘pera’, ‘uva´, maçã} é maçã, que tem frequência igual a 2, enquanto ‘pera’ e ‘uva’ tem frequência (cada uma delas) igual a 1.

Já para atributos numéricos, são utilizadas comumente a média, o percentil e a mediana. Para um conjunto com n valores (numéricos): VJ = {v1, v2, ..., vn}, o valor da média desse conjunto é obtido pela equação 4.2.1.

=

(4.2.1)

Com relação a mediana, ela é uma medida menos sensível a presença de outliers no conjunto de dados se comparada com a média. Ela determina o valor central. Para utilizar essa medida, primeiramente é necessário ordenar de maneira crescente o conjunto de dados. Após a ordenação, a equação 4.2.2 é utilizada para calculá-la:

(4.2.2)

Como exemplo, seja o conjunto de valores {5,11,7,9,32}. Ordenando esses valores de forma crescente, tem-se {5,7,9,11,32}. Uma observação faz-se necessária: em caso de ter valores repetidos, eles são todos mantidos na ordenação. A mediana é igual a 9. Já para o conjunto de valores {5,11,7,9,32,40}, após ordená-lo de forma crescente, tem-se {5,7,9,11,32,40}. A mediana = (9+11)/2 = 10.

Outras medidas utilizadas com frequência são os quartis e os percentis. Da mesma forma que a mediana, antes de serem calculados, os valores são ordenados em forma crescente. Tanto os quartis quanto os percentis utilizam pontos de divisão distintos da mediana. Os quartis dividem o conjunto ordenado em quartos (Q1, Q2 e Q3), em que cada parte tem 25% dos dados; já os percentis dividem a amostra em cem partes (P1, P2, ..., P99) iguais, cada parte contendo 1% dos dados.

Para o cálculo do quartil, inicialmente verifica-se a posição deles pela tabela 4.2.3, com n denotando o número de elementos e k a posição:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Para n par | | | Para n ímpar | | |
| Q1 | Q2 | Q3 | Q1 | Q2 | Q3 |
|  |  |  |  |  |  |

Tabela 4.2.3: Equações para cálculo de posição dos quartis

Após obter a posição dos quartis, é preciso encontrar o elemento que está localizado nela (os dados precisam estar ordenados). Como convenção, caso o valor da posição k seja fracionário, faz-se a média entre os valores das posições anterior e posterior à posição encontrada. Caso os dados estejam organizados em distribuição de frequências, o procedimento é o mesmo, considerando as frequências relacionadas (valor discreto) a cada valor ou, se for o caso, o ponto médio da classe.

Para os percentis, o procedimento é semelhante aos quartis. Para calculá-los, utiliza-se a equação 4.2.4:

P1 (1ª Percentil) = n/100

(4.2.4)

P2 (2ª Percentil) = 2n/100

Pi (iª Percentil) = in/100

.

.

P99 (99ª Percentil) = 99n/100

**6.3 Medidas de Espalhamento**

Medidas de Espalhamento mensuram a dispersão presente em um conjunto de valores. Auxiliam na observação da concentração ao redor de um valor, como a mediana, por exemplo, ou se estão espalhados. As medidas mais usadas são a variância e o desvio padrão. O cálculo da variância para um conjunto com n valores é dado pela equação 4.3.1, em que  j é a média. Já o desvio padrão é calculado através da raiz quadrada da variância.

Variância (xJ) = 2

(4.3.1)

**6.4 Correlação**

A covariância permite analisar a relação entre atributos distintos (dois ou mais). Ela mensura o grau de variação conjunta entre eles. Covariância próxima de zero indica que não há relacionamento linear entre os atributos; se apresentar um valor positivo, isso caracteriza que os atributos possuem relação direta, ou seja, havendo aumento de um atributo, frequentemente o outro também aumenta. Em caso de covariância negativa, o contrário também ocorre, ou seja, ocorre a diminuição.

Porém, a covariância sofre influência da dimensão dos atributos. Para eliminar essa questão, usa-se a medida de correlação, que corrige a influência da dimensão, sendo mais clara com relação à indicação da relação linear entre os atributos.

A covariância pode ser calculada pela equação 4.3.2. Nela, diz respeito ao valor médio do i-ésimo atributo; já representa o valor do i-ésimo atributo do objeto k-ésimo:

(4.3.2)

A matriz de correlação contém a correlação entre cada par de atributos. Na matriz, cada elemento pode ser definido através da equação 4.3.3, em que diz respeito ao i-ésimo atributo e ser refere ao desvio padrão desse atributo em questão.

(4.3.3)

Na matriz de correlação, os elementos assumem valores que variam entre -1 e +1, sendo -1 o máximo da correlação negativa e +1 o máximo da correlação positiva. Quanto os atributos têm correlação positiva, geralmente o aumento do valor de um dos atributos causa aumento no valor do outro. Da mesma maneira, isso ocorre com a correlação negativa, ou seja, a diminuição do valor de um acarreta a diminuição do outro.

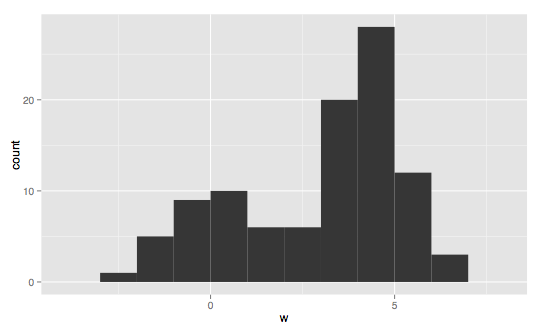
**7. Gráficos**

Gráficos são recursos visuais que facilitam a análise e são muito utilizados principalmente na elaboração de dashboards. Os mais comumente usados nas análises são: histograma, boxplots, gráficos de pizza e scatter plot.

**7.1 Histograma**

Histograma é um tipo de gráfico de barras que engloba distribuição de frequências. Nele, cada barra diz respeito a uma classe e a altura da barra denota a frequência absoluta. Ele é muito utilizado para comparar resultados e sintetizar grandes conjuntos de dados de maneira visual. Basicamente, serve para ilustrar a distribuição dos dados. A seguir (Figura 5.1), a imagem de um histograma:

Figura 5.1 – Exemplo de Histograma

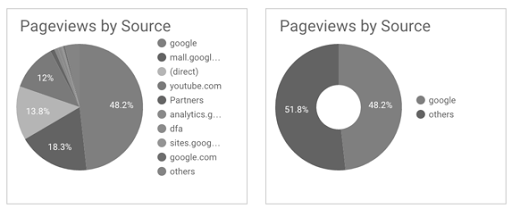


Fonte: Wikipedia[[1]](#footnote-1)

**7.2 Gráfico de Pizza**

Gráfico de Pizza é um gráfico circular em que cem por cento dos dados corresponde a trezentos e sessenta graus. Cada setor dele é proporcional ao valor que representa e diz respeito a uma categoria em que o ângulo diz respeito a uma porcentagem dos dados. É bastante utilizado quanto envolve poucas categorias, no entanto, a leitura fica comprometida quando há muitas classes. Como ilustração, segue um exemplo de gráfico de pizza:

Figura 5.2 – Exemplo de Gráfico de Pizza



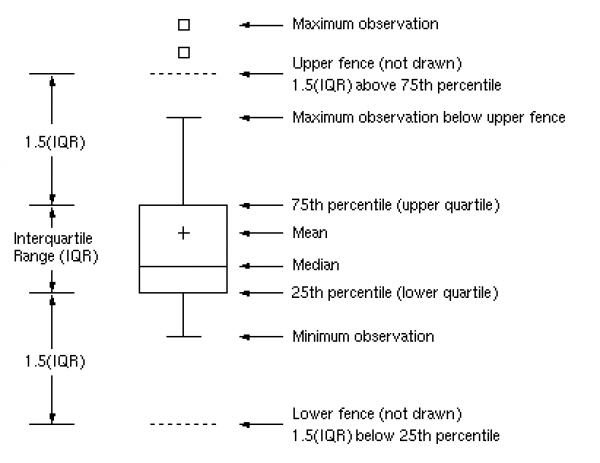
Fonte: Suporte Google[[2]](#footnote-2)

**7.3 Boxplot**

Boxplot é uma das formas de se visualizar a distribuição dos dados. Ele mostra um resumo dos quartis (Q1, Q2 e Q3), bem como os limites inferiores e superiores. Nesse tipo de gráfico, a linha horizontal inferior indica o valor mínimo e a linha horizontal superior, o valor máximo dos dados analisados. No retângulo, o lado inferior diz respeito ao primeiro quartil; já o lado superior, ao terceiro quartil. A linha horizontal no interior do retângulo denota a mediana.

Uma variação desse tipo de gráfico é o boxplot modificado. Nele, são estabelecidos limites inferiores e superiores. No limite superior, a linha tracejada alcança geralmente valores não muito distantes do terceiro quartil. Uma medida comum para estabelecer esse limite é: terceiro quartil + 1,5 x (terceiro quartil – primeiro quartil). De forma análoga, para estabelecer o limite inferior: primeiro quartil – 1,5 x (terceiro quartil – primeiro quartil). Os valores abaixo (ou acima) do limite inferior(superior) são considerados outliers.

Figura 5.3 – Boxplot modificado[[3]](#footnote-3)



Fonte: Laboratório de Ecologia de Florestas da USP3

**7.4 Scatter Plot**

Gráfico muito utilizado para visualizar informações de dados multivariados. Ele demonstra a correlação linear entre duas características de um objeto. Levando-se em conta apenas dois dos atributos de um objeto, associa-se um ponto num plano de duas dimensões. Os valores dos atributos (números inteiros ou reais) estabelecem as coordenadas do ponto no plano bidimensional.

Figura 5.4 – Scatter Plot[[4]](#footnote-4)

Gráfico, Gráfico de dispersão

Descrição gerada automaticamente

Fonte: Tutorial Python4

**8. Desenvolvimento do Software**

* 1. **OBJETIVOS**

Desenvolver um software para que os usuários possam extrair insights e algumas informações estatísticas tendo como base de dados um arquivo com extensão ‘.csv’, armazenado em: https://github.com/RichardTompson/EngenhariaSoftware-/blob/main/planilha\_tcc.csv

* 1. **ESCOPO GERAL DO PRODUTO**

Efetuou-se levantamento de requisitos para o software desenvolvido no trabalho de conclusão do curso de Engenharia de Software, fornecendo informações imprescindíveis para o desenvolvimento.

O software foi desenvolvido com o intuito de possibilitar que os usuários possam efetuar consultas e visualizar as informações com facilidade. O programa foi desenvolvido para atender essa finalidade específica. Os requisitos foram levantados por meio de histórias de usuários, que é um dos procedimentos adotados pela metodologia ágil Scrum.

* 1. **STAKEHOLDERS**

Usuários em geral, sem público específico.

Classificação dos stakeholders: Dentre as classificações possíveis (favoráveis, neutros, contrários e sabotadores), os stakeholders acima citados são classificados como *favoráveis* ao desenvolvimento do software.

* 1. **SISTEMA OPERACIONAL**

Microsoft Windows

* 1. **LINGUAGEM DE PROGRAMAÇÃO**

Linguagem de Programação Python, versão 3.9.5.

**9. Levantamento de Requisitos**

O desenvolvimento de softwares não se inicia pela escrita do código, mas pelo levantamento dos requisitos. Mesmo nas metodologias ágeis, que apresentam documentação reduzida, os requisitos continuam sendo fundamentais. Segundo Sutherland, é preciso identificar o usuário final do produto, ou seja, para quem o produto será destinado? As perspectivas mudam conforme o usuário final.

Os requisitos são especificados de forma mais simples no Scrum, por isso, essa foi a abordagem utilizada. Histórias de Usuário são descrições suscintas das funcionalidades, sob a perspectiva dos usuários. Elas não são escritas em linguagem técnica, mas em linguagem de negócio. São simples, claras e objetivas. Sobre isso, temos que “as pessoas pensam em narrativas, histórias. É desse modo que compreendemos o mundo.” (Sutherland, p. 91).

**9.1 CONVENÇÕES, TERMOS E ABREVIAÇOES**

Os requisitos foram levantados considerando as praticas da metodologia ágil Scrum. Assim, elas foram elicitadas por meio de Histórias de Usuário, com a seguinte convenção:

**[HU00X][Nome]**<identificador><Nome da história de usuário>

SENDO<usuário>

POSSO<funcionalidade>

POIS ASSIM<vantagem>

Para ilustrar, segue um exemplo:

**[HU001] Consulta vendas por ano**

SENDO um gerente comercial

POSSO consulta as vendas do ultimo ano

POIS ASSIM posso visualizar informações sobre o faturamento por período.

Nessa história de usuário utilizada como exemplo, [HU001] quer dizer que ela está localizada numa seção denominada ‘Histórias de Usuário’, cujo identificador é [HU001]. HU quer dizer ‘História de Usuário’, e 001, que é a primeira levantada.

Para evitar repetição desnecessária em requisitos de um mesmo grupo de usuários, criou-se um contexto, conforme descrito a seguir:

**CONTEXTO**

SENDO<pré-condição que se repete em algumas histórias de usuário>

Como exemplo, segue abaixo a convenção adotada:

**[HU001] Consulta vendas por ano**

SENDO um gerente comercial

POSSO consulta as vendas do ultimo ano

POIS ASSIM posso visualizar informações sobre o faturamento por período.

**[HU002] Consulta vendas por funcionário**

SENDO um gerente comercial

POSSO consulta as vendas de cada funcionário

POIS ASSIM posso visualizar informações sobre a performance deles

Aplicação do contexto:

**CONTEXTO**

SENDO um gerente commercial

**[HU001] Consulta vendas por ano**

POSSO consulta as vendas do ultimo ano

POIS ASSIM posso visualizar informações sobre o faturamento por período.

**[HU002] Consulta vendas por funcionário**

POSSO consulta as vendas de cada funcionário

POIS ASSIM posso visualizar informações sobre a performance deles

**Resultados e Conclusões**

Apesar de Python ser uma linguagem interpretada, que executa o código de maneira mais lenta quando comparada às linguagens compiladas, quanto se trata de análise de dados, essa diferença no tempo de execução acaba sendo compensada (em grande parte) pelo menor tempo gasto pelos programadores de Python em detrimento a linguagens de mais baixo nível (McKinney, p. 22). Além disso, ela é capaz de sintetizar rapidamente dados brutos em informações valiosas.

Python possui muitas funções matemáticas e estatísticas disponíveis em algumas de suas bibliotecas, que tornam os cálculos relativamente simples, de maneira automatizada. Além disso, esses recursos são procedimentos amplamente testados antes de serem disponibilizados para uso. Isso contribui para evitar erros de cálculos, já que eles foram previamente avaliados por pessoas competentes para tal finalidade. Muitos conceitos matemáticos/estatísticos são complexos de implementar e de serem realizados manualmente e/ou usando determinados recursos, mas com a linguagem Python eles são realizados com extrema facilidade e mostraram-se simples de implementar.

As bibliotecas Matplotlib e Seaborn do Python mostraram-se poderosos recursos que permitiram plotagens de gráficos, tornando a visualização das informações bastante agradáveis para o usuário e de forma sintetizada, facilitando a compreensão das informações disponibilizadas. Os elementos gráficos mostraram-se um excelente recurso visual sobre os dados.

O software proporcionou um ganho de tempo considerável, já que disponibiliza diversas consultas como apenas a seleção da opção desejada pelos usuários, sem a necessidade de realizar cálculos manuais ou deduzir fórmulas complexas necessárias em muitas outras ferramentas de análise de dados. Pode-se notar uma grande melhoria na otimização do tempo.

O software também contribui para a tomada de decisões, por exemplo, pode-se constatar que a ocupação que mais toma empréstimos é a de aposentado. Dessa forma, caso o foco seja concessão de empréstimos, esse grupo pode ser abordado prioritariamente, já tem que aceitação melhor do produto. Técnicas estatísticas para visualização viabilizam o entendimento da distribuição dos dados e podem auxiliar na melhor forma de abordar a questão (Faceli, p. 28).

Portanto, conclui-se que Python e suas ferramentas são recursos que tornam mais simples muitos problemas relacionados à análise de dados, contribuindo sobremaneira para diversas demandas, economizando tempo e sendo muito eficiente.

**REFERÊNCIAS**

ALMEIDA, Mario de Souza. **Elaboração de projeto, TCC, dissertação e tese: uma abordagem simples, pratica e objetiva.** 2. Ed. São Paulo: Atlas, 2014.

CARVALHO, B. V e MELLO, C. H. P. Aplicação do método ágil Scrum no desenvolvimento de produtos de software emu ma pequena empresa de base tecnológica. Disponível em: https://www.scielo.br/j/gp/a/34xH953TFwLPYDB9BYdJghL/?lang=pt. Acesso em: 7 de jan. de 2021.

COHN, M. Desenvolvimento de software com Scrum: aplicando métodos ágeis com sucesso. São Paulo: Bookman, 2011

FACELI, Katti et. al Inteligência Artificial: **Uma Abordagem de Aprendizagem de Máquina.** Rio de Janeiro: LTC, 2011.

FREITAS, Janaina Aparecida de. **Gerenciamento de Software.** Reimpressão, Maringá-PR: Unicesumar, 2017.

Guia para o conhecimento Scrum (Guia SBOK™) by ScrumStudy

PERSEGUINE, V. R. **Engenharia de Requisitos.** Reimpressão, Maringá-PR.: Unicesumar, 2016.

PRESSMAN*, R. S.; MAXIM, B. R.* ***Engenharia de software: uma abordagem profissional.*** *8. ed. Porto Alegre: AMGH, 2016.*

*MARTINELLI, M.* ***Gráficos e Mapas: Construa-os Você Mesmo.*** *São Paulo: Moderna, 1998.*

*McKINNEY W.* ***Python para Análise de Dados****, 2.ed, Ed.O´Reilly Media, Inc, Wes McKinney 2018*

*MENEZES, Nilo Ney Coutinho.* ***Introdução à programação com python****. 3.ed. Editora Novatec, 2010.*

Sbrocco, J. H. T. C. e Macedo, P. C. 2012 **Metodologias ágeis engenharia de software sob medida.** Editora Érica Ltda. 1º ed. 2012.

SOMMERVILLE, I. **Engenharia de Software**. 9.ed. São Paulo: Pearson Prentice Hall, 2011.

SOMMERVILLE, I. **Engenharia de Software**. 8. ed. São Paulo: Pearson Addison-Wesley, 2007.

SUTHERLAND, Jeff. **Scrum - a arte de fazer o dobro de trabalho na metade do tempo**. São Paulo: LEYA BRASIL, 2016.

**APÊNDICES**

**APÊNDICE A**

**A.1 INTERFACE/MENU**

A.1.1 História de usuário: [HU001] Interface/Menu.

Figura A1 - Interface principal

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

**Subopções do menu principal:**

Figura A2 – Menu Clientes

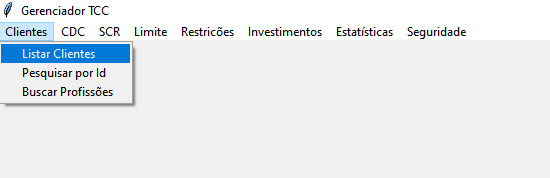


Figura A3 – Menu SCR

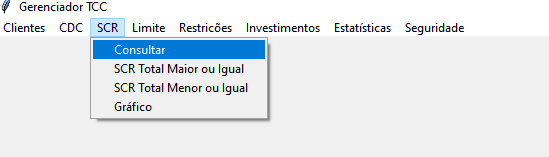
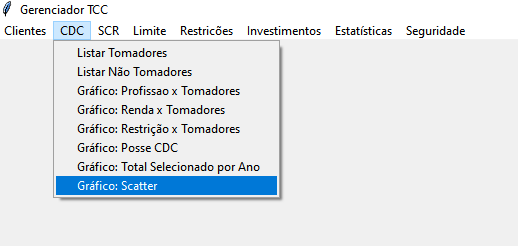


Figura A4 – Menu CDC



**APÊNDICE B**

**B.1 GRÁFICOS PLOTADOS NO SOFTWARE**

B.1.1 História de usuário: [HU028] Gráfico Seguridade

Figura A5 – Gráfico referente a posse de produtos

Gráfico, Gráfico de cascata

Descrição gerada automaticamente

B.1.2 História de usuário: [HU010] Gráfico posse de empréstimos

Figura A6 – Gráfico de pizza posse de empréstimos

Gráfico, Gráfico de pizza

Descrição gerada automaticamente

B.1.3 História de usuário: [HU026] Correlação

Figura A7 – Matriz de CorrelaçãoGráfico

Descrição gerada automaticamente

B.1.4 História de usuário: [HU028] Gráfico Seguridade

Figura A8 – Gráfico de Pizza Posse de Produtos

Gráfico, Gráfico de pizza

Descrição gerada automaticamente

B.1.5 História de usuário: [HU011] Gráfico empréstimos por ano

Figura A9 – Total de Empréstimo por mês e ano

Interface gráfica do usuário

Descrição gerada automaticamente

B.1.6 História de usuário: [HU015] Gráfico SCR

Figura A10 – Gráfico SCR

Gráfico, Gráfico de barras

Descrição gerada automaticamente

B.1.7 História de usuário: [HU025] Estatísticas

Figura A11 – Tela com as estatísticasInterface gráfica do usuário, Texto, Tabela

Descrição gerada automaticamente

B.1.8 História de usuário: [HU008] Gráfico renda dos tomadores

Figura A12 – Faixa de renda dos tomadores

Gráfico, Gráfico de barras

Descrição gerada automaticamente

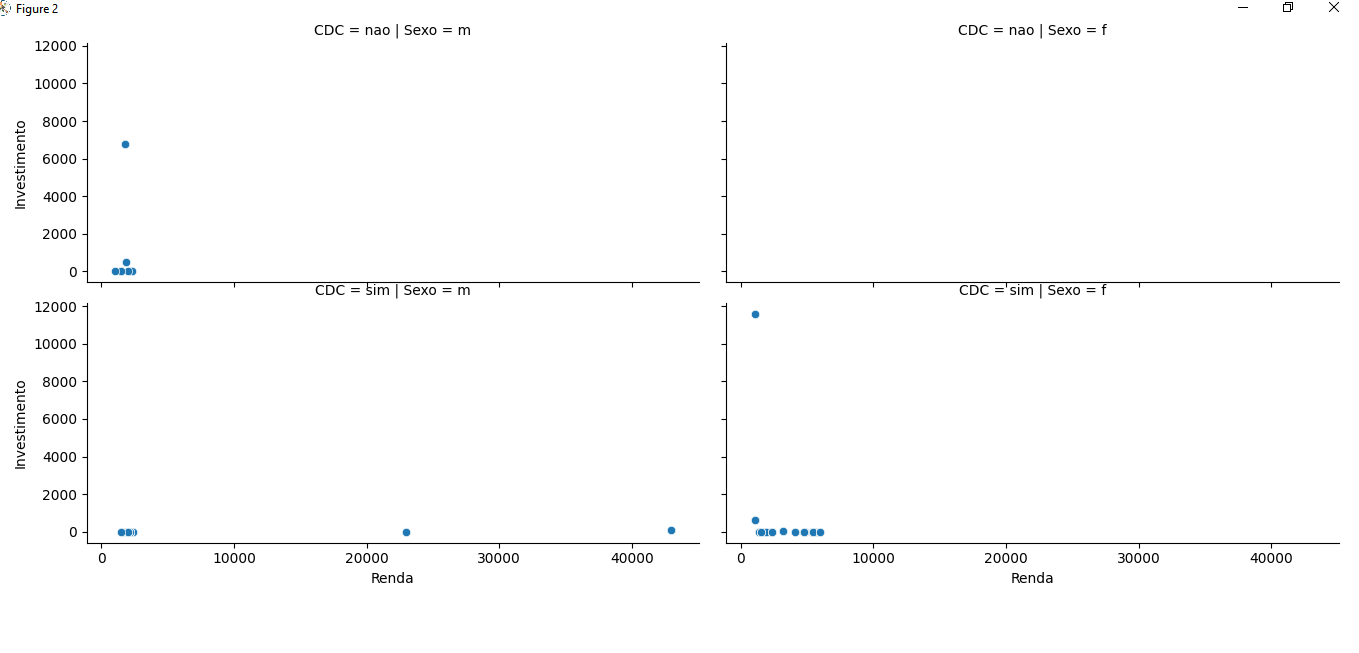
B.1.9 História de usuário: [HU007] Gráfico profissão dos tomadores

Figura A13 – Profissão dos tomadoresGráfico, Histograma

Descrição gerada automaticamente

B.1.10 [HU012] Gráfico relação empréstimo e outros atributos

Figura A14 – Relação alguns atributos

****

B.1.11 [HU025] Estatísticas

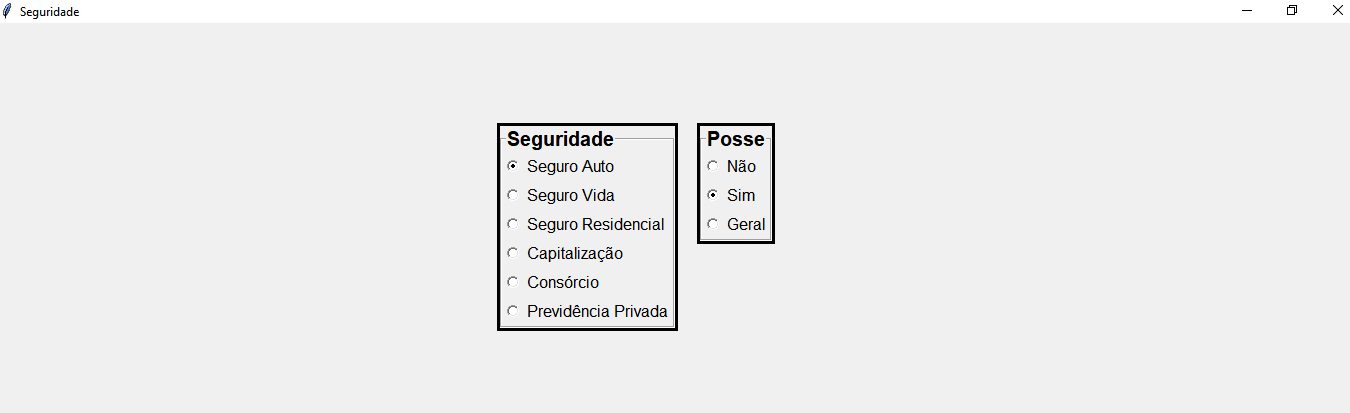
Figura A15 – Boxplot SCR

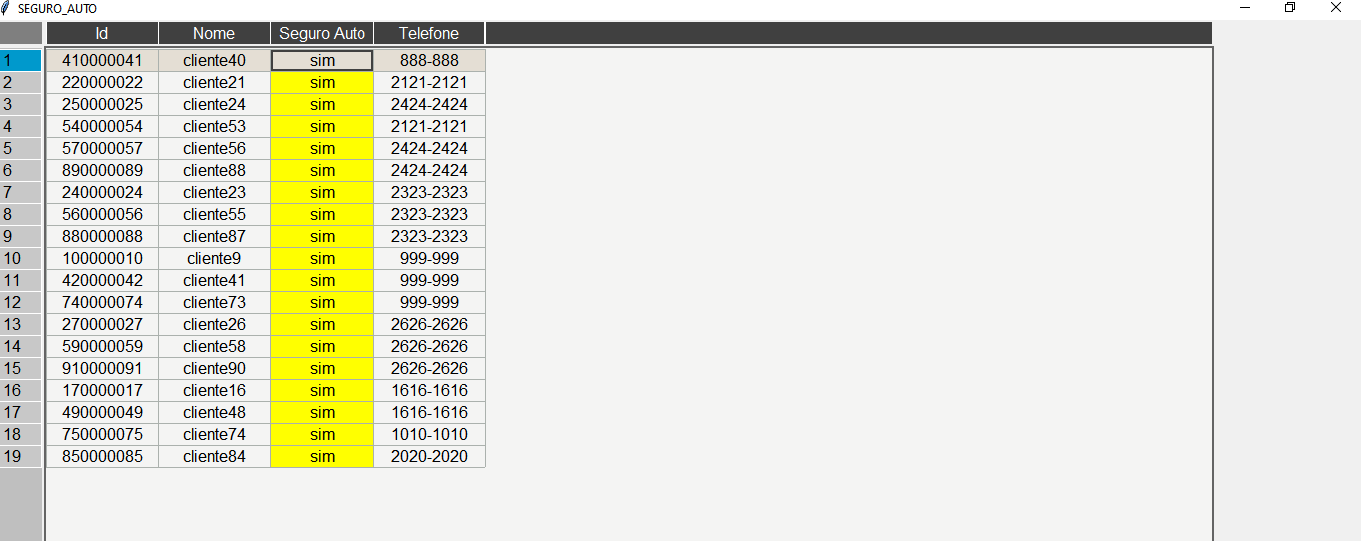
Gráfico, Gráfico de caixa estreita

Descrição gerada automaticamente

B.1.12 [HU027] Posse de Seguridade

Figura A16 – Posse seguro auto

****

****

**APÊNDICE C**

**BASE DE DADOS**

A seguir, segue um registro da base, a título de ilustração:

Figura A17 – Registro da base de dados





A base de dados utilizada trata-se de um arquivo com extensão .csv, contendo 29 colunas, em que cada uma representa um atributo do registro. A seguir, são discriminados os atributos de cada registro da base:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Tipo** | **Delimitador** | **Significado** |
| **id** | Numérico(int) | Ponto e vírgula | Identificador único de cada cliente |
| **nome** | Nominal | Ponto e vírgula | Nome do cliente |
| **investimento** | Numérico(float) | Ponto e vírgula | Investimento total do cliente |
| **cdc\_antes** | Nominal | Ponto e vírgula | Informa se o cliente já tomou empréstimo anteriormente |
| **scr\_nosso** | Numérico(float) | Ponto e vírgula | Indica o valor do endividamento total do cliente na própria instituicao |
| **scr\_outros** | Numérico(float) | Ponto e vírgula | Indica o endividamento total do cliente em concorrentes |
| **scr\_total** | Numérico(float) | Ponto e vírgula | Valor que reflete o endividamento total do cliente (somando-se o scr\_nosso e scr\_outros) |
| **usando\_limite** | Nominal | Ponto e vírgula | Indica se o cliente está usando o limite da conta (sim) ou não |
| **restricao\_hist** | Nominal | Ponto e vírgula | Indica se o cliente tem algum tipo de restrição no histórico de cadastro. |
| **pgto\_parcial\_fatura** | Nominal | Ponto e vírgula | Indica se a pessoa pagou parcialmente a fatura do cartão de crédito nos últimos 03 meses |
| **ano\_nasc** | Nominal | Ponto e vírgula | Ano de Nascimento do cliente |
| **profissao** | Nominal | Ponto e vírgula | Indica a profissão |
| **aposentado** | Nominal | Ponto e vírgula | Indica se a pessoa é aposentado (sim) ou não |
| **sexo** | Nominal | Ponto e vírgula | Indica o sexo |
| **valor\_empr** | Numérico(float) | Ponto e vírgula | Valor dos total dos empréstimos |
| **tempo\_conta** | Numérico(int) | Ponto e vírgula | Indica o tempo de conta , desde a abertura, até a data de levantamento, em meses |
| **linha** | Nominal | Ponto e vírgula | Diz sobre a linha de crédito tomada na ligação |
| **data** | Nominal | Ponto e vírgula | A data da contração do empréstimo |
| **contatofone** | Nominal | Ponto e vírgula | Indica se houve contato telefônico junto ao cliente nos três meses anteriores |
| **valor\_fatura\_mes** | Numérico(float) | Ponto e vírgula | Indica o valor total da fatura do cartão |
| **renda** | Numérico(float) | Ponto e vírgula | Renda total bruta |
| **telefone** | Nominal | Ponto e vírgula | Número de telefone |
| **seguro\_auto** | Nominal | Ponto e vírgula | Indica se possui (sim) ou não seguro auto |
| **seguro\_residencial** | Nominal | Ponto e vírgula | Indica se possui (sim) ou não seguro residencial |
| **seguro\_vida** | Nominal | Ponto e vírgula | Indica se possui (sim) ou não seguro vida |
| **capitalizacao** | Nominal | Ponto e vírgula | Indica se possui (sim) ou não capitalização |
| **previdencia\_privada** | Nominal | Ponto e vírgula | Indica se possui (sim) ou não previdência privada |
| **consorcio** | Nominal | Ponto e vírgula | Indica se possui (sim) ou não consórcio |
| **contratou** | Nominal | Ponto e vírgula | Indica se o cliente contratou(sim) algum produto no momento da oferta. |

Convém destacar que partiu-se da premissa de que os dados foram exportados para uma planilha no formato .csv por um banco de dados robusto, experimentado e confiável, que adota os aspectos ACID (atomicidade, consistência, isolamento e durabilidade). Portanto, não foram considerados aspectos de pré-processamento dos dados, como dados incompletos, inconsistentes, redundantes ou com ruídos.

Para nortear o leitor menos acostumado com esses termos, Faceli(2011, p. 34) dá uma definição sobre eles, que é: dados incompletos “é a ausência de valores para alguns atributos de alguns objetos.” ; inconsistentes seriam “aqueles que possuem valores conflitantes em seus atributos”. Dados redudantes são quando um objeto “é muito semelhante a um outro objeto do mesmo conjunto de dados”. “Dados com ruídos são dados que contêm objetos que, aparentemente, não pertencem à distribuição que gerou os dados analisados.”.

**APÊNDICE D**

**DISPONIBILIDADE DOS ARQUIVOS**

**D.1. Base de Dados:**

https://github.com/RichardTompson/EngenhariaSoftware-/blob/main/planilha\_tcc.csv

**D.2. Código Fonte:**

https://github.com/RichardTompson/EngenhariaSoftware-/blob/main/software\_tcc.py

**D.3 Trabalho de conclusão de curso:**

https://github.com/RichardTompson/EngenhariaSoftware-/blob/main/meu\_pre\_projeto\_ad.docx

**APÊNDICE E**

**HISTÓRIAS DE USUÁRIOS**

**CONTEXTO**

SENDO Usuário em geral

**[HU001] Interface/Menu**

POSSO acessar os dados por meio de uma interface

POIS ASSIM visualiza-se informações a respeito dos clientes com facilidade por meio de menus e opções de consultas, ao invés de trabalhar com planilhas menos acessíveis e intuitivas.

**[HU002] Lista Geral de Clientes**

POSSO consultar todos os clientes cadastrados

POIS ASSIM visualiza-se informações gerais dos clientes.

**[HU003] Consultar por ID**

POSSO consultar clientes pelo id.

POIS ASSIM seleciona-se um(a) determinado(a) cliente E obtem-se informações cadastrais especificamente sobre ele(a) E SE não lembrar o id de interesse, ENTÂO posso fazer uma consulta aos id’s da base.

**[HU004] Consultar profissões**

POSSO consultar todos os clientes por determinada profissão

POIS ASSIM filtra-se profissões específicas para focar em determinadas profissões de interesse.

**[HU005] Empréstimo Posse**

POSSO consultar clientes com posse de empréstimo (CDC)

POIS ASSIM visualiza-se clientes que costumam tomar empréstimos rotineiramente.

**[HU006] Empréstimo Sem Posse**

POSSO consultar clientes sem posse de empréstimo (CDC)

POIS ASSIM visualiza-se os clientes que não costumam tomar empréstimos e ofertar a eles.

**[HU007] Gráfico profissão dos tomadores**

POSSO consultar a profissão dos tomadores de empréstimo

POIS ASSIM identifica-se aquelas que têm maior aceitação do produto. E compara elas entre si.

**[HU008] Gráfico renda dos tomadores**

POSSO consultar a faixa de renda dos tomadores de empréstimo

POIS ASSIM identifica-se as faixas que têm maior aceitação do produto E compara elas entre si.

**[HU009] Gráfico tomadores**

POSSO consultar a quantidade de tomadores E não tomadores de empréstimo

POIS ASSIM visualiza-se uma característica essencial para o relacionamento.

**[HU010] Gráfico posse de empréstimos**

POSSO visualizar o percentual de clientes que tomaram OU não CDC em um gráfico

POIS ASSIM compara-se a quantidade de clientes tomadores E não tomadores.

**[HU011] Gráfico empréstimos por ano**

POSSO visualizar a quantidade de empréstimo efetuado por ano

POIS ASSIM compara-se a quantidade de empréstimos realizados por período de tempo.

**[HU012] Gráfico relação empréstimo e outros atributos**

POSSO visualizar se existente uma relação entre a renda E o sexo E os valores de investimento E a posse de CDC

POIS ASSIM visualiza-se se existe algum vínculo entre essas características.

**[HU013] Consultar SCR**

POSSO fazer uma consulta sobre o saldo de SCR dos clientes

POIS ASSIM visualiza-se o endividamento deles conosco E com a concorrência, comparando-as.

**[HU014] Filtrar endividamento no SCR**

POSSO selecionar os clientes com saldo de SCR maior E menor que determinado valor

POIS ASSIM visualiza-se aqueles que possuem maior E menor endividamento

**[HU015] Gráfico SCR**

POSSO visualizar o endividamento dos clientes no SCR em um gráfico

POIS ASSIM posso efetuar uma comparação entre os endividamentos em nossa base E na concorrência.

**[HU016] Uso de Limite**

POSSO consultar clientes que não estejam OU estejam utilizando limite da conta

POIS ASSIM visualiza-se aqueles(as) clientes que estão OU não estão pagando juros pela utilização do limite.

**[HU017] Gráfico Uso de Limite**

POSSO visualizar o percentual de clientes usando OU não o limite em um gráfico

POIS ASSIM posso efetuar uma comparação entre a quantidade de clientes utilizando E não utilizando limite

**[HU018] Histórico de Restrições**

POSSO consultar clientes com restrições vigentes OU sem restrições

POIS ASSIM verifica-se um característica importante sobre o perfil do clientes.

**[HU019] Sem restrições e uso Limite**

POSSO consultar clientes sem restrições vigentes E que usam o limite

POIS ASSIM filtra-se cliente sem restrições E que estejam usando limite, que são públicos propensos a tomada de crédito.

**[HU020] Sem restrições e parcial cartão**

POSSO consultar clientes sem restrições vigentes E que pagaram a fatura do cartão de crédito parcialmente

POIS ASSIM verifica-se os clientes sem restrições e que talvez precisem de crédito para pagar integralmente a fatura do cartão.

**[HU021] Sem restrições e aposentados**

POSSO consultar clientes sem restrições vigentes E que são aposentados

POIS ASSIM verifica-se os clientes sem restrições e que talvez precisem de crédito com taxa diferenciada para esse público.

**[HU022] Gráfico Restrições**

POSSO visualizar o percentual de clientes com restrições OU não em um gráfico

POIS ASSIM posso efetuar uma comparação entre a quantidade de clientes com restrições E sem restrições.

**[HU023] Investimentos ordenados**

POSSO consultar os valores investidos em ordem crescente OU decrescente

POIS ASSIM consulta-se os clientes com menores ou maiores investimentos que são perfis de interesse para os negócios.

**[HU024] Gráfico Investimentos**

POSSO consultar os valores investidos em ordem crescente OU decrescente

POIS ASSIM consulta-se os clientes com menores ou maiores investimentos que são perfis de interesse para os negócios.

**[HU025] Estatísticas**

POSSO consultar estatísticas sobre os dados

POIS ASSIM tem-se uma percepção maior sobre a base de dados, como: qual é o maior valor de investimento? Qual é o menor? A média? É necessário consultar essas informações para várias caracteristicas dos clientes E também analisar a quantidade de posse de produtos, por sexo**.**

**[HU026] Correlação**

POSSO consultar alguma medida estatística, algo do tipo

POIS ASSIM verifica-se se algum acréscimo OU decréscimo nos valores de uma característica (como a renda, o sexo, etc.) influenciam nos valores do(s) outro(s).

**[HU027] Posse de Seguridade**

POSSO consultar clientes que tenham posse OU não de alguma seguridade (seguro auto, residencial, vida, capitalização, previdência privada e consórcio)

POIS ASSIM verifica-se quem já têm o hábito de adquirí-los OU não.

**[HU028] Gráfico Seguridade**

POSSO visualizar a quantidade de clientes que tenham posse OU não de alguma seguridade (seguro auto, residencial, vida, capitalização, previdência privada e consórcio)

POIS ASSIM compara-se a quantidade de pessoas que ja adquiram OU não algum desses produtos.

**APÊNDICE F**

**ARTEFATOS SCRUM**

F.1 Backlog do Produto





****

****

****

**F.2 Gráfico de Burndown do Produto**

**F.2 Backlog da Sprint**

**Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo, Tabela

Descrição gerada automaticamente**

**F.3 Burndown da Sprint**





F.4 Sprint Retrospective



1. Disponível em: <https://pt.wikipedia.org/wiki/Histograma> Acesso em ago. 2021. [↑](#footnote-ref-1)
2. https://support.google.com/datastudio/answer/7187587?hl=pt-BR#zippy=%2Cneste-artigo [↑](#footnote-ref-2)
3. http://labtrop.ib.usp.br/doku.php?id=cursos:planeco:roteiro:05-descr\_base [↑](#footnote-ref-3)
4. https://pythonspot.com/matplotlib-scatterplot/ [↑](#footnote-ref-4)