CUSTOMER SEGMENTATION USING K-MEANS CLUSTERING

A Mini Project Report submitted to
JNTU Hyderabad in partial fulfillment
of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted by

PATLOLLA PRATHUSHA	**21RG1A05B2**
POPULLA VASUNDHARA	**21RG1A05B3**
SHEERLA NEHA	**21RG1A05C0**

Under the Guidance of

Mrs.V.SUNEETHA RANI
B.Tech,M.Tech
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MALLA REDDY COLLEGE OF ENGINEERING FOR WOMEN
An UGC Autonomous Institution
Approved by AICTE New Delhi and Affiliated to JNTUH
Maisammaguda , Medchal (Dist), Hyderabad -500100, Telangana.

OCTOBER 2024

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MALLA REDDY COLLEGE OF ENGINEERING FOR WOMEN

An UGC Autonomous Institution

Approved by AICTE New Delhi and Affiliated to JNTUH

Maisammaguda , Medchal (Dist), Hyderabad -500100, Telangana.

OCTOBER 2024

CERTIFICATE

This is to certify that the Mini project entitled **"CUSTOMER SEGMENTATION USING K-MEANS CLUSTERING"** has been submitted by **PATLOLLA PRATHUSHA (HT-No.21RG1A05B2), POPULLA VASUNDHARA (HT-No.21RG1A05B3), SHEERLA NEHA(HT-No. 21RG1A05C0)** in partial fulfillment of the requirements for the award of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE & ENGINEERING**. This record of bonafide work carried out by them under my guidance and supervision. ***The result embodied in this mini project report has not been submitted to any other University or Institute for the award of any degree.***

Mrs.V.SUNEETHA RANI
Assistant professor
Project Guide

Mrs. K. SHEETAL
Head of the Department

External Examiner

ACKNOWLEDGEMENT

The Mini Project work carried out by our team in the Department of Computer Science and Engineering, Malla Reddy College of Engineering for Women, Hyderabad. ***This work is original and has not been submitted in part or full for any degree or diploma of any other university.***

We wish to acknowledge our sincere thanks to our project guide **Mrs.V.Suneetha Rani,** Assistant Professor, Computer Science & Engineering for formulation of the problem, analysis, guidance and her continuous supervision during the course of work.

We acknowledge our sincere thanks to **Dr.Kanaka Durga Returi**, Principal & Professor of Department of Computer Science and Engineering and **Mrs. Sheetal Kulkarni,** Head of the Department, MRCEW and all our faculties of CSE Department for their kind cooperation in making this Mini Project work a success.

We extend our gratitude to **Sri. Ch. Malla Reddy**, Founder Chairman and **Sri. Ch. Mahender Reddy,** Secretary, **Dr. Vaka Murali Mohan,** Director for their kind cooperation in providing the infrastructure for completion of our Mini Project.

We acknowledge our special thanks to the entire teaching faculty and non-teaching staff members of the Computer Science & Engineering Department for their support in making this project work a success.

PATLOLLA PRATHUSHA	**21RG1A05B2**	________________
POPULLA VASUNDHARA	**21RG1A05B3**	________________
SHEERLA NEHA	**21RG1A05C0**	________________

INDEX

Chapter **Page No.**

ABSTRACT **vii**

LIST OF FIGURES **viii**

1. SYSTEM ANALYSIS **1**

1.1 Existing System 1

1.1.1 Disadvantages of Existing System 1

1.2 Proposed System 1

1.2.1 Advantages of Proposed System 2

1.3 Introduction 3

2. LITERATURE SURVEY **4**

3. SYSTEM DESIGN **7**

3.1 System modules **7**

3.1.1 Server 7

3.1.2 User 7

3.2 System Architecture 8

3.3 System Requirements 9

3.3.1 Hardware Requirements 9

3.3.2 Software Requirements 9

3.4 UML Diagrams 10

3.4.1 Use Case Diagram 10

3.4.2 Class Diagram 11

3.4.3 Sequence Diagram 12

3.4.4 Flow Chart Diagram 13

3.4.5 Date Flow Diagram 15

4. INPUT & OUTPUT DESIGN 16
4.1 Input Design 16
4.2 Output Design 17
5. SOFTWARE ENVIRONMENT 18
5.1 Python Technology 18
5.2 Advantages 18
5.3 Advantages of python over other languages 20
5.4 Disadvantages 20
5.5 History of python 21
5.6 Python Development Steps 22
5.7 Purpose 22
5.8 TensorFlow 23
5.9 Numpy 23
5.10 Pandas 24
5.11 Matplotlib 24
5.12 Scikit-learn 24
5. 13 Install python step-by-step in windows and mac 25
6. SYSTEM STUDY 32
6.1 Economic Feasibility 33
6.2 Technical Feasibility 33
6.3 Social Feasibility 34
7. SYSTEM TESTING 34
7.1 Types of Tests 34
7.1.1 Unit testing 34
7.1.2 Integration testing 35

7.1.3 Functional testing 35

7.1.4 System testing 36

7.1.5 Acceptance testing 36

8. RESULTS 37

9. CONCLUSION & FUTURE ENHANCEMENT 47

10. BIBLIOGRAPHY 48

ABSTRACT

The spirit of the modern age is characterized by innovation, with everyone engaged in a race to outshine the competition.Today's businesses thrive on such innovation, captivating customers with their products. However, the vast array of products available often leaves customers bewildered about what to purchase, while companies struggle to identify which customer segments to target. This is where machine learning becomes crucial. Various algorithms are applied to uncover hidden patterns in data, facilitating better decision-making for the future.This elusive concept of which segment to target becomes clear through segmentation. The Customer segmentation involves grouping customers with similar behaviours into the same segment and those with different patterns into separate segmentsA Python program was developed and trained using a standard scales on a dataset with two features from 200 training samples from a local retail shop.The features include the average amount spent by customers and the average number of visits to the shop annually. By applying clustering, five customer segments were identified: Careless, Careful, Standard, Target, and Sensible customers.

LIST OF FIGURES

Fig. No.	Fig. Name	Page.no
3.1	System Architecture	9
3.2	Use Case Diagram	10
3.3	Class Diagram	11
3.4	Sequence Diagram	12
3.5	Flow Chart Diagram	13
3.6	Flow Chart Diagram	14
3.7	Data Flow Diagram	15
8.1	Screenshot of Home Page	37
8.2	Screenshot of Upload dataset	38
8.3	Screenshot of Gender Distribution	39
8.4	Screenshot of Age Distribution	40
8.5	Screenshot of Age and Gender Distribution	41
8.6	Screenshot of Annual income count	41
8.7	Screenshot of Income per age by Gender	42
8.8	Screenshot of Average Annual income	43
8.9	Screenshot of Spending score distribution	43
8.10	Screenshot of Elbow method	44
8.11	Screenshot of Different types of clusters	44
8.12	Screenshot of Prediction by Annual Income	45
8.13	Screenshot of Prediction by Spending score	46
8.14	Screenshot of Prediction of Final Output	46

CHAPTER-1: SYSTEM ANALYSIS

1.1 EXISTING SYSTEM

Customer segmentation is a critical practice in marketing and business strategy, aiming to divide a customer base into groups with similar characteristics, behaviours, or needs. Two common methods for customer segmentation are earlier methods presumably traditional or heuristic-based.The existing base paper uses “Elbow method” to find out The minimum optimal clusters for the K-means clustering. But elbow method does not work effectively in a few cases.

From the scatter plot of X1 vs X2 we may be able to tell that the data originates from five different clusters, but we struggle to perceive high-Dimensional data.The Elbow Method, as seen on the left, would most likely Lead us to k = 4.The Elbow Method causes us to conclude that two of the Clusters are one since they are so close together. This is due to the fact that establishing a centroid in the centre of both Clusters reduces the relative distance between data points.Calculating the appropriate number of clusters for our clustering job requires an approach that is more accurate, rigorous, and dependable. At this moment, the Silhouette score is used.

1.1.1 Disadvantages of Existing System

Elbow Method and heuristic algorithms has several disadvantages. It is sensitive to the initial centroid selection, which can lead to suboptimal clustering. The Elbow Method for choosing the number of clusters is subjective and may not always yield a clear result. K-Means assumes clusters are spherical, which is often unrealistic in real-world data, and it struggles with outliers and noisy data. Additionally, it has difficulty handling complex cluster shapes and can become computationally expensive and inefficient when applied to large datasets.

1.2 PROPOSED SYSTEM

In the proposed system, the system adopted a comprehensive approach to the pickpocket detection problem. The overall framework of solution is illustrated in this system. The system first partitioned the city area into regions with functional categories. Then, the mobility characteristics of passengers were extracted from transit records dynamically over time.A core component of the system was a two-step passenger classification process, the first step being regular passenger filtering, and the second step being suspect detection. Finally, system user feedback information, such as newly confirmed thieves, was entered as ground truth for future model training based on a utility function that strikes a tradeoff between effectiveness (i.e., performance) and relevance (i.e., recency). A more detailed description of this system may be found in this system.

The contribution of study can be summarized as follows. Firstly, we identified a number of features that may be extracted from AFC records and are potentially useful for distinguishing thieves from regular passengers.Secondly, a two-step approach was proposed to make the suspect detection problem practical in a large-scale data environment where the positive and negative samples are extremely imbalanced.Thirdly, a dynamic filtering enhancement significantly reduced the everyday computation costs and maintained superior accuracy. Most importantly, a real system for the end user was designed and tested using real-world, large-scale data. As an applied data science study, this solution is the first to address an important social issue identifying Pick pockets by using big data.

1.2.1 Advantages of Proposed System

The proposed system for customer segmentation improves on K-Means by offering optimized centroid initialization, reducing the risk of poor clustering. It automates cluster selection, eliminating the subjective nature of the Elbow Method. By handling complex, non-spherical clusters, the system is better suited for real-world customer data. It also incorporates techniques for outlier detection, minimizing the impact of noise and improving the quality of segmentation.

1.3 INTRODUCTION

Customer segmentation is a critical process in marketing and business analytics that involves dividing a customer base into distinct groups based on shared characteristics, behaviors, or needs. This helps businesses better understand their customers, tailor marketing efforts, and improve product offerings to meet the specific demands of different customer segments. One of the most popular techniques used for customer segmentation is K-Means clustering, a machine learning algorithm that groups data points into a predefined number of clusters based on their similarity.

K-Means clustering works by partitioning a dataset into K clusters, where each data point belongs to the cluster with the nearest mean, called the centroid. The algorithm iteratively assigns data points to clusters and updates the centroids until convergence is achieved. The simplicity, speed, and scalability of K-Means make it a widely-used tool in customer segmentation for identifying patterns and trends within customer data.

A key step in K-Means is determining the optimal number of clusters (K), as this can significantly impact the quality of the segmentation. One commonly used technique to address this is the **Elbow Method**, which plots the sum of squared distances from each point to its assigned centroid. The “elbow” point on the curve, where the rate of decrease slows down, is used as an indication of the optimal number of clusters. However, this method can be subjective and may not always clearly indicate the best value for K.

K-means clustering has a wide array of applications in customer segmentation. For instance, behavioral segmentation allows businesses to analyze purchasing habits, frequency of transactions, and preferences for certain product categories. This insight can drive targeted promotions, loyalty programs, and personalized marketing campaigns. On the demographic front, clustering based on variables such as age, gender, income, and location helps companies tailor their messaging and product offerings. Geographic segmentation, on the other hand, enables businesses to understand regional trends and preferences, allowing for localized marketing strategies.

CHAPTER-2: LITERATURE SURVEY

1. Khan et al. (2021): He explored the integration of K-means clustering with deep learning techniques to improve segmentation accuracy. This highlights a growing trend towards hybrid methodologies that leverage the strengths of multiple analytical approaches.Despite the advancements in applying K-means clustering for customer segmentation, several knowledge gaps persist. Primarily, there is limited research addressing the challenges associated with high-dimensional data, which can complicate the clustering process. Future studies could explore the application of dimensionality reduction techniques in conjunction with K-means to enhance segmentation outcomes.Additionally, the integration of real-time data analytics with K-means clustering presents an exciting avenue for exploration. As customer behaviors increasingly shift in response to market dynamics and technological advancements, developing adaptive segmentation models that can accommodate real-time data is vital.Furthermore, while existing studies have predominantly focused on the retail and airline sectors, there is a need for research investigating the application of K-means clustering in emerging industries such as e-commerce and telecommunication. These sectors may exhibit unique customer behaviors that require tailored segmentation strategies.

2. Abhijit J. Patil and Prashant R.Nair(2021):states that the integration of machine learning techniques, including clustering algorithms, for customer . It emphasizes the benefits of utilizing advanced algorithms to uncover complex patterns in customer data. The study showcases real-world examples of businesses that have successfully improved sales by adopting machine learning-driven segmentation strategies.

3.Wu et al (2020): He conducted an empirical study that combined the RFM (Recency, Frequency, Monetary) model with K-means clustering to segment customers based on purchase behaviors. Their findings revealed that this combined approach significantly enhances the precision of customer segmentation efforts.

4.V. Kumar and Rohit Aggarwal(2019):states an in-depth analysis of customer segmentation techniques, focusing on both traditional methods and modern data-driven approaches. It emphasizes the role of clustering in creating meaningful customer segments and discusses the impact of segmentation on sales and business outcomes. The paper highlights the importance of considering various data sources, such as transaction data and customer behavior, in the segmentation process.

5. Riaz et al. (2019): He proposed using eight validity indexes to facilitate this process, enhancing the robustness of segmentation outcomes. This methodological advancement is crucial, as improper clustering can lead to misinterpretation of customer segments and ineffective marketing strategies.

6.Elham Fadalyet al(2018): states that the different customer segmentation strategies using clustering techniques and evaluates impact on sales and customer retention. The paper compares K-means clustering with other methods and discusses the advantages and limitations of each approach. It provides valuable insights into the practical implementation of clustering for sales improvement.

7.Syakur et al(2018): He emphasized the integration of the elbow method with K-means to improve its efficiency and effectiveness in processing large datasets for customer profiling. The elbow method aids in determining the optimal number of clusters, thereby enhancing segmentation accuracy.

8. Rappoport and Shamir (2018):they highlighted the algorithm's potential in multi-omic data analysis, indicating its broader applicability in complex data environments. This flexibility is also noted in the work by Peker et al. (2017), who applied K-means in conjunction with the SPSS tool to develop a real-time sales prediction system in the grocery retail industry.

9.M.A.Hossain and Mohammad Shorif Uddin(2016): states that a practical application of clustering, specifically K-means clustering, for customer segmentation based on purchasing behavior. The study groups and proposes strategies to target these segments. The findings suggest that demonstrates the effectiveness of clustering in identifying distinct customer personalized marketing approaches derived from clustering analysis can significantly improve sales.

10.Liau and Tan (2014): demonstrated the application of K-means clustering within the low-cost airline industry to gain insights into customer preferences and behaviors through text mining. This application underscores the versatility of K-means in various sectors where customer experience is paramount.

11.Michel Wedel and Wagner Kamakura(2000): states that the concept of market segmentation and highlights its importance in marketing strategy. It covers various segmentation techniques, including clustering, and discusses their benefits in improving sales and customer satisfaction. The paper provides a comprehensive overview of segmentation methods and their applications.

CHAPTER-3: SYSTEMDESIGN

3.1 System Modules

3.1.1 Server:

In this module, the Web Server has to login by using valid user name and password. After login successful he can do some operations such as List of All Users and Authorize, Add Route Details, View Route Details, View Smart Card Details, view All Passenger Travelled Detail, View Detecting Pickpocket Suspects, View Passenger Trips and Transit Records Results

3.1.2 User:

In this module, there are n numbers of users are present. User should register before doing some operations. After registration successful he has to login by using authorized user name and password. Login successful he will do some operations like View Your Profile, Add Smart Card, View Your Smart Card Details, Add Boarding Station Details, View and Add Exiting Station Details, View Your Travelled Details.

3.2 System Architecture

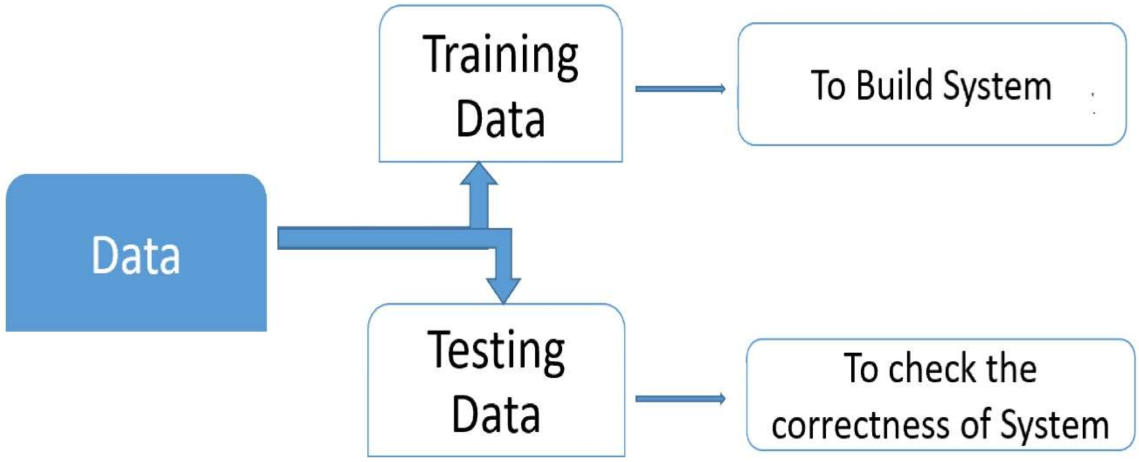


Fig.3.1: System Architecture

3.3 System Requirements

3.3.1 Hardware Requirements

System : Intell I-3, 5, 7 Processor.

Hard Disk : 500 GB.

Floppy Drive : 1.44 Mb.

Monitor : 14’ Colour Monitor.

Mouse : Optical Mouse.

Ram : 2Gb.

3.3.2 Software Requirements:

Operating system : Windows 7,8,10 Ultimate, Linux, Mac.

Front-End : Python.

Coding Language : Python.

Software Environment : Anaconda (jupyter or spyder).

3.3 UML Diagrams

3.4.1 **Use Case Diagram**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

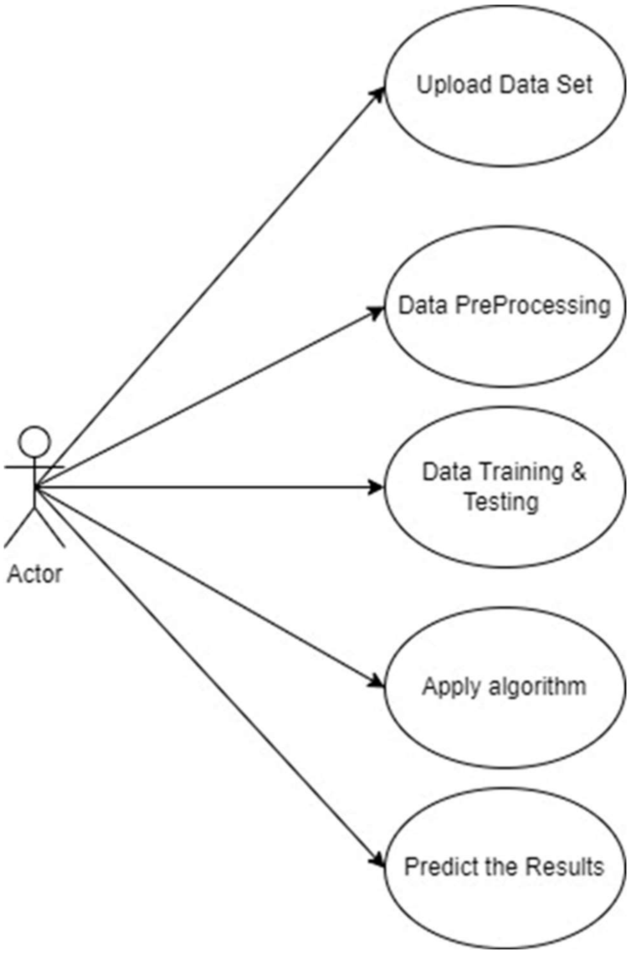
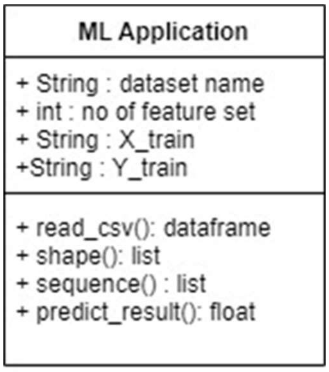


Fig. 3.2 : Use Case Diagrams

3.4.2 **Class Diagram**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information



ML Application

+ String : dataset name
+ int : no of feature set
+ String : X_train
+String : Y_train

+ read_csv(): dataframe
+ shape(): list
+ sequence() : list
+ predict_result(): float

Fig. 3.3: Class Diagram

3.4.3 **Sequence Diagram**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams

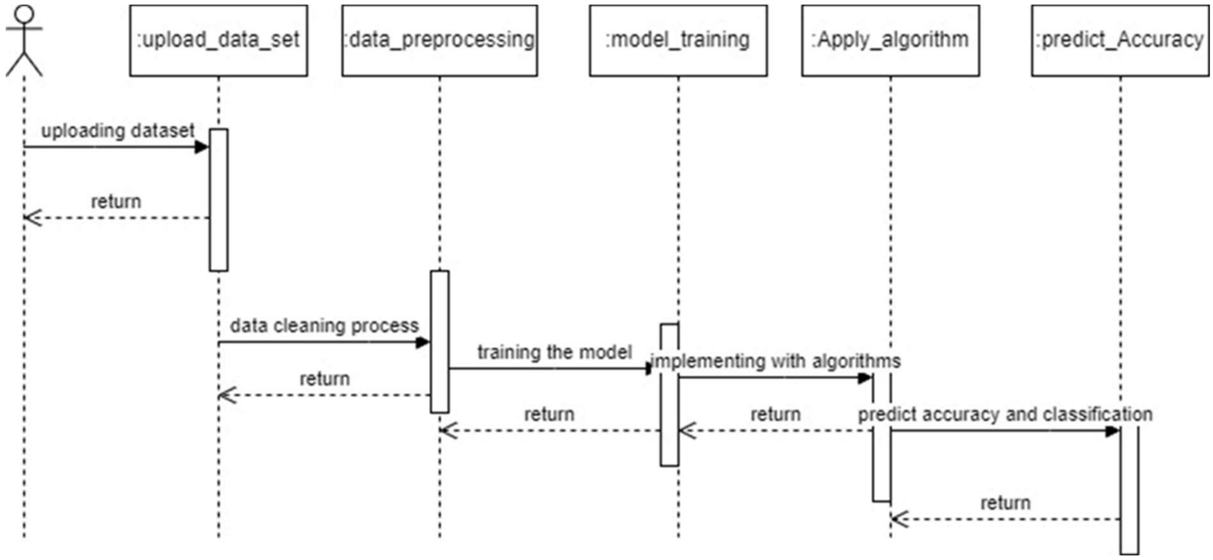


Fig. 3.4 : Sequence Diagram

3.4.4 **Flow Chart Diagram**

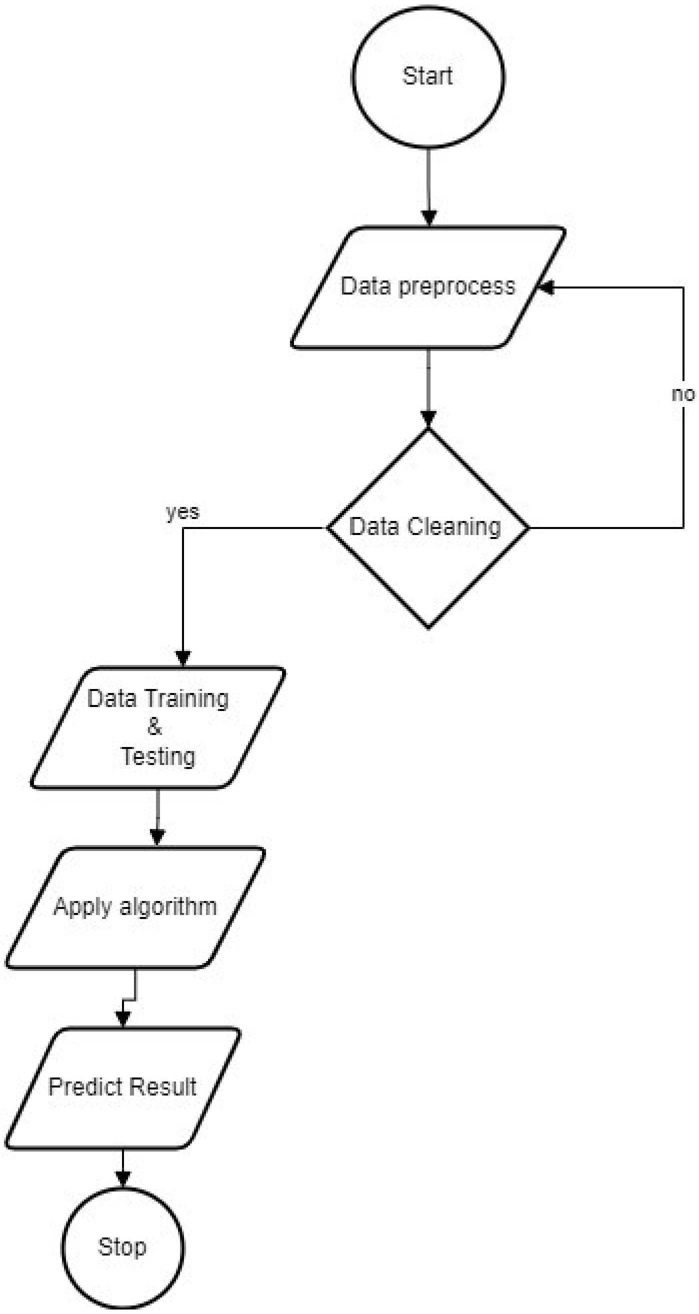


Fig.3.5 : Flow Chart Diagram

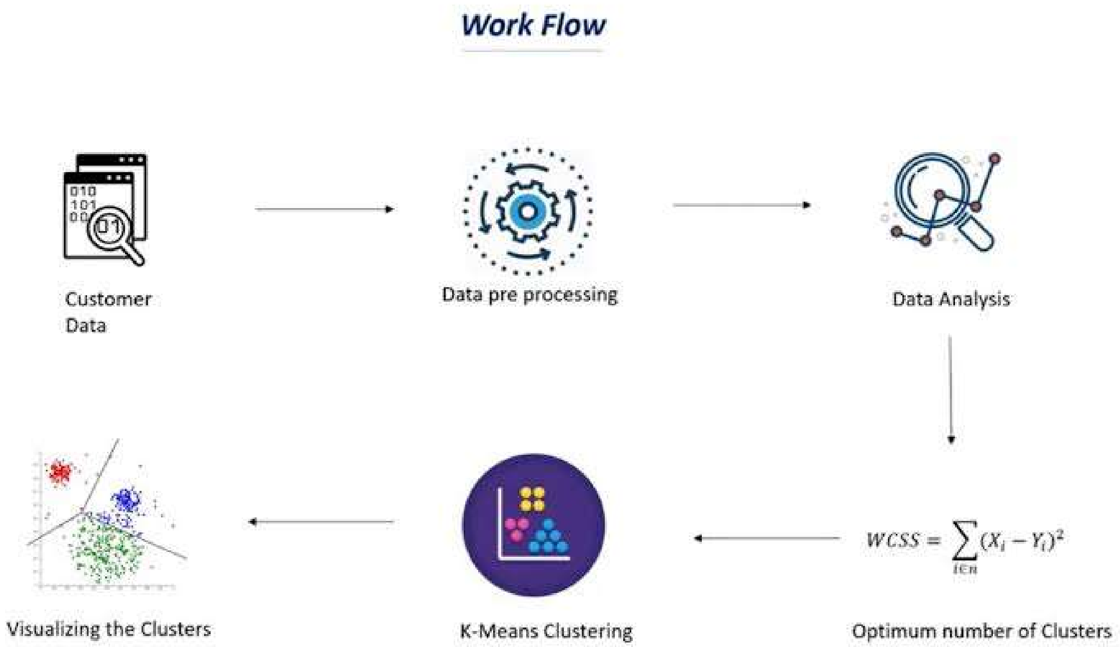


Fig.3.6: Flow Chart Diagram

3.4.5 **Data Flow Diagram**

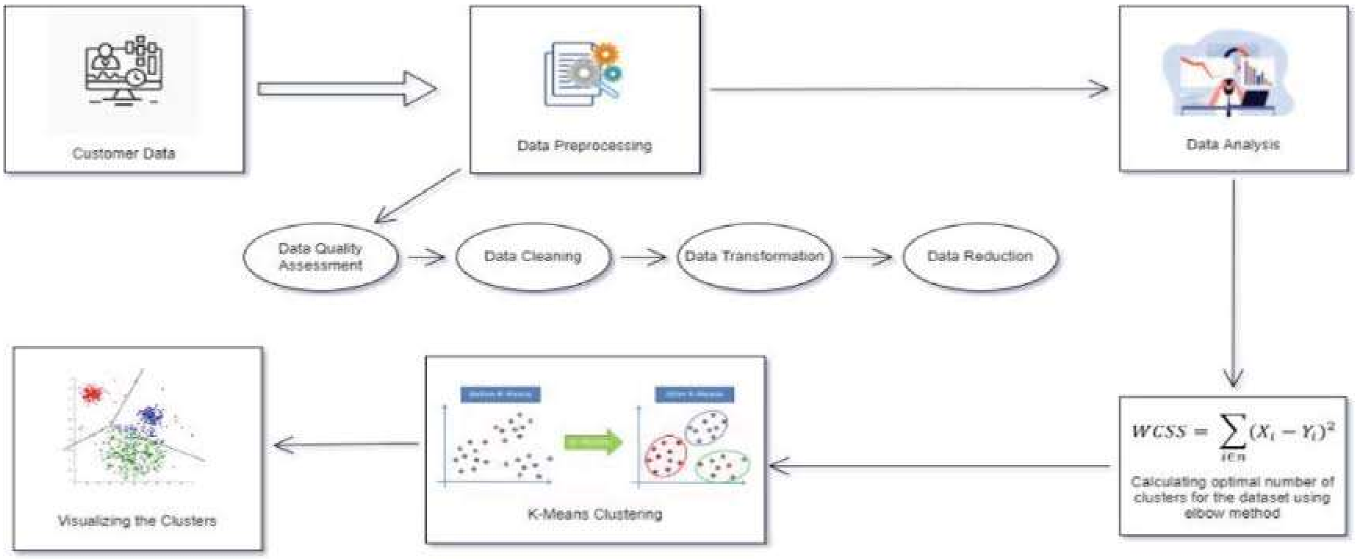


Fig.3.7: Data Flow Diagram

CHAPTER-4: INPUT AND OUTPUT DESIGN

4.1 INPUT DESIGN

Input design is a part of overall system design. The main objective during the input design is as given below:

- To produce a cost-effective method of input.
- To achieve the highest possible level of accuracy.
- To ensure that the input is acceptable and understood by the user.

INPUT STAGES

The main input stages can be listed as below:

- Data recording
- Data transcription
- Data conversion
- Data verification
- Data control
- Data transmission
- Data validation
- Data correction

INPUT TYPES

It is necessary to determine the various types of inputs. Inputs can be categorized as follows:

- External inputs, which are prime inputs for the system.
- Internal inputs, which are user communications with the system.
- Operational, which are computer department's communications to the system?
- Interactive, which are inputs entered during a dialogue.

INPUT MEDIA

At this stage choice has to be made about the input media. To conclude about the input media consideration has to be given to;

- Type of input
- Flexibility of format
- Speed
- Accuracy

- Verification methods
- Rejection rates
- Ease of correction
- Storage and handling requirements
- Security
- Easy to use
- Portability

4.2 OUTPUT DESIGN

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provides a permanent copy of the results for later consultation. The various types of outputs in general are:

- External Outputs, whose destination is outside the organization
- Internal Outputs whose destination is within organization and they are the
- User's main interface with the computer.
- Operational outputs whose use is purely within the computer department.
- Interface outputs, which involve the user in communicating directly.

OUTPUT DEFINITION

The outputs should be defined in terms of the following points:

- Type of the output
- Content of the output
- Format of the output
- Location of the output
- Frequency of the output
- Volume of the output
- Sequence of the output

It is not always desirable to print or display data as it is held on a computer. It should be decided as which form of the output is the most suitable.

CHAPTER-5: SOFTWARE ENVIRONMENT

5.1 **Python Technology**

Below are some facts about Python.

- Python is currently the most widely used multi-purpose, high-level programming language.
- Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.
- Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.
- Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.

The biggest strength of Python is huge collection of standard library which can be used for the following –

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like Opencv, Pillow)
- Web scraping (like Scrapy, BeautifulSoup, Selenium)
- Test frameworks
- Multimedia

5.2 Advantages Let's see how Python dominates over other languages.

1. Extensive Libraries

Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don't have to write the complete code for that manually.

2. Extensible

As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add scripting capabilities to our code in the other language.

4. Improved Productivity

The language's simplicity and extensive libraries render programmers more productive than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

5. IOT Opportunities

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet Of Things. This is a way to connect the language with the real world.

6. Simple and Easy

When working with Java, you may have to create a class to print 'Hello World'. But in Python, just a print statement will do. It is also quite easy to learn, understand, and code. This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.

7. Readable

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and indentation is mandatory. This further aids the readability of the code.

8. Object-Oriented

This language supports both the procedural and object-oriented programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the encapsulation of data and functions into one.

9. Free and Open-Source

Like we said earlier, Python is freely available. But not only can you download Python for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

10. Portable

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn't the same with Python. Here, you need to code only once, and you can run it anywhere. This is called Write Once Run Anywhere (WORA). However, you need to be careful enough not to include any system-dependent features.

11. Interpreted

Lastly, we will say that it is an interpreted language. Since statements are executed one by one, debugging is easier than in compiled languages.

Any doubts till now in the advantages of Python? Mention in the comment section.

5.3 Advantages of Python Over Other Languages

1. Less Coding

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don't have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

2. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

The 2019 Github annual survey showed us that Python has overtaken Java in the most popular programming language category.

3. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and machine learning, automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

5.4 Disadvantages of Python

So far, we've seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let's now see the downsides of choosing Python over another language.

1. Speed Limitations

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in slow execution. This, however, isn't a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

2. Weak in Mobile Computing and Browsers

While it serves as an excellent server-side language, Python is much rarely seen on the client-side. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called Carbonnelle.

The reason it is not so famous despite the existence of Brython is that it isn't that secure.

3. Design Restrictions

As you know, Python is dynamically-typed. This means that you don't need to declare the type of variable while writing the code. It uses duck-typing. But wait, what's that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can raise run-time errors.

4. Underdeveloped Database Access Layers

Compared to more widely used technologies like JDBC (Java DataBase Connectivity) and ODBC (Open DataBase Connectivity), Python's database access layers are a bit underdeveloped. Consequently, it is less often applied in huge enterprises.

5. Simple

No, we're not kidding. Python's simplicity can indeed be a problem. Take my example. I don't do Java, I'm more of a Python person. To me, its syntax is so simple that the verbosity of Java code seems unnecessary.

This was all about the Advantages and Disadvantages of Python Programming Language.

5.5 History of Python

What do the alphabet and the programming language Python have in common? Right, both start with ABC. If we are talking about ABC in the Python context, it's clear that the programming language ABC is meant. ABC is a general-purpose programming language and programming environment, which had been developed in the Netherlands, Amsterdam, at the CWI (Centrum Wiskunde &Informatica). The greatest achievement of ABC was to influence the design of Python. Python was conceptualized in the late 1980s. Guido van Rossum worked that time in a project at the CWI, called Amoeba, a distributed operating system. In an interview with Bill Venners1, Guido van Rossum said: "In the early 1980s, I worked as an implementer on a team building a language called ABC at Centrum voor Wiskunde en Informatica (CWI). I don't know how well people know ABC's influence on Python. I try to mention ABC's influence because I'm indebted to everything I learned during that project and to the people who worked on it. "Later on in the same Interview, Guido van Rossum continued: "I remembered all my experience and some of my frustration with ABC. I decided to try to design a simple scripting language that possessed some of ABC's better properties, but without its problems. So I started typing. I created a simple virtual machine, a simple parser, and a simple runtime. I made my own version of the various ABC parts that I liked. I created a basic syntax, used indentation for statement grouping instead of curly braces or begin-end blocks, and developed a small number of powerful data types: a hash table (or dictionary, as we call it), a list, strings, and numbers."

5.6 Python Development Steps

Guido Van Rossum published the first version of Python code (version 0.9.0) at alt.sources in February 1991. This release included already exception handling, functions, and the core data types of list, dict, str and others. It was also object oriented and had a module system.

Python version 1.0 was released in January 1994. The major new features included in this release were the functional programming tools lambda, map, filter and reduce, which Guido Van Rossum never liked. Six and a half years later in October 2000, Python 2.0 was introduced. This release included list comprehensions, a full garbage collector and it was supporting unicode. Python flourished for another 8 years in the versions 2.x before the next major release as Python 3.0 (also known as "Python 3000" and "Py3K") was released. Python 3 is not backwards compatible with Python 2.x. The emphasis in Python 3 had been on the removal of duplicate programming constructs and modules, thus fulfilling or coming close to fulfilling the 13th law of the Zen of Python: "There should be one -- and preferably only one -- obvious way to do it."Some changes in Python 7.3:

- Print is now a function.
- Views and iterators instead of lists
- The rules for ordering comparisons have been simplified. E.g., a heterogeneous list cannot be sorted, because all the elements of a list must be comparable to each other.
- There is only one integer type left, i.e., int. long is int as well.
- The division of two integers returns a float instead of an integer. "//" can be used to have the "old" behaviour.
- Text Vs. Data Instead of Unicode Vs. 8-bit

5.7 Purpose

We demonstrated that our approach enables successful segmentation of intra-retinal layers—even with low-quality images containing speckle noise, low contrast, and different intensity ranges throughout—with the assistance of the ANIS feature.

Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

- Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

Modules Used in Project

5.8 TensorFlow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

5.9 NumPy

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

- A powerful N-dimensional array object
- Sophisticated (broadcasting) functions
- Tools for integrating C/C++ and Fortran code
- Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary datatypes can be defined using NumPy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

5.10 Pandas

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

5.11 Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

5.12 Scikit – learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

- Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

5.13 Install Python Step-by-Step in Windows and Mac

Python a versatile programming language doesn't come pre-installed on your computer devices. Python was first released in the year 1991 and until today it is a very popular high-level programming language. Its style philosophy emphasizes code readability with its notable use of great whitespace.

The object-oriented approach and language construct provided by Python enables programmers to write both clear and logical code for projects. This software does not come pre-packaged with Windows.

How to Install Python on Windows and Mac

There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

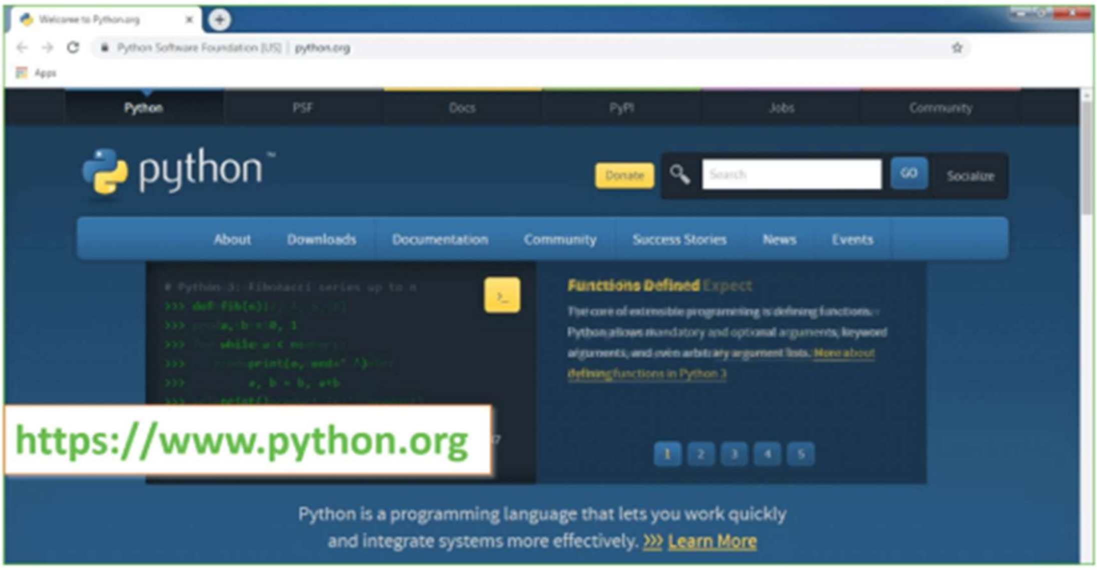
Note: The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your System Requirements. Based on your system type i.e. operating system and based processor, you must download the python version. My system type is a Windows 64-

bit operating system. So the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheatsheet here.The steps on how to install Python on Windows 10, 8 and 7 are divided into 4 parts to help understand better.

Download the Correct version into the system

Step 1: Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: https://www.python.org



```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
```

Functions Defined Expect

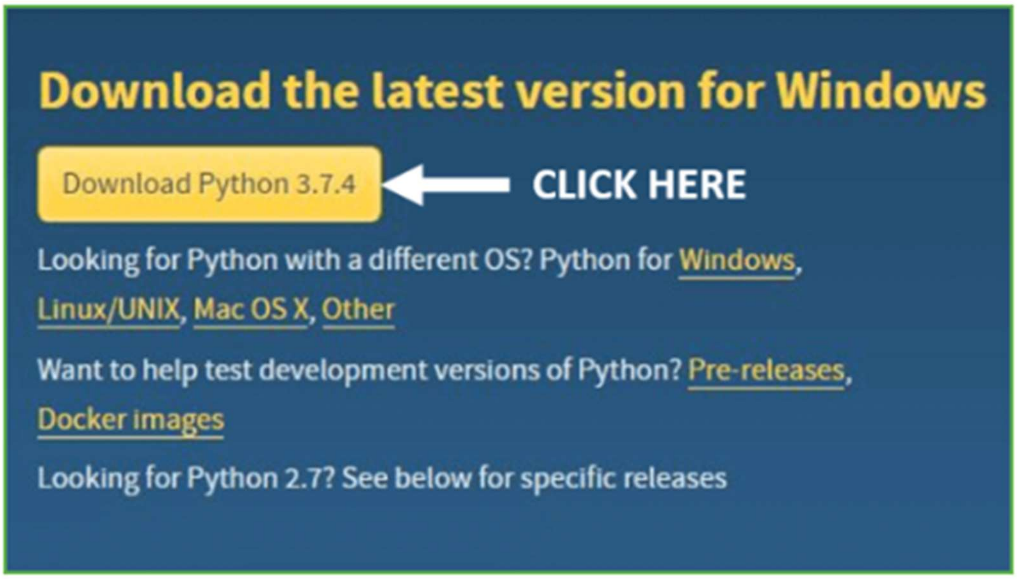
The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. More about defining functions in Python 3

https://www.python.org

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> Learn More

Now, check for the latest and the correct version for your operating system.

Step 2: Click on the Download Tab.



Download the latest version for Windows

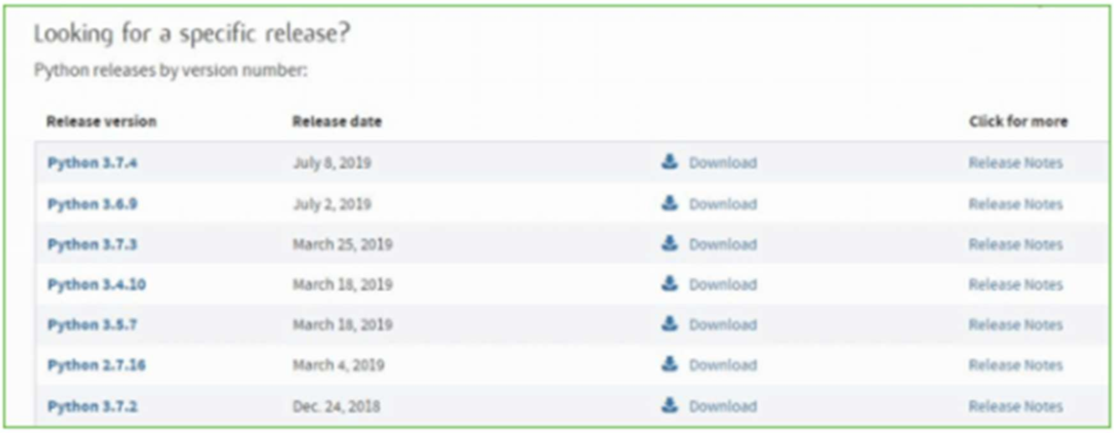
Download Python 3.7.4 ← CLICK HERE

Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other

Want to help test development versions of Python? Pre-releases, Docker images

Looking for Python 2.7? See below for specific releases

Step 3: You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4



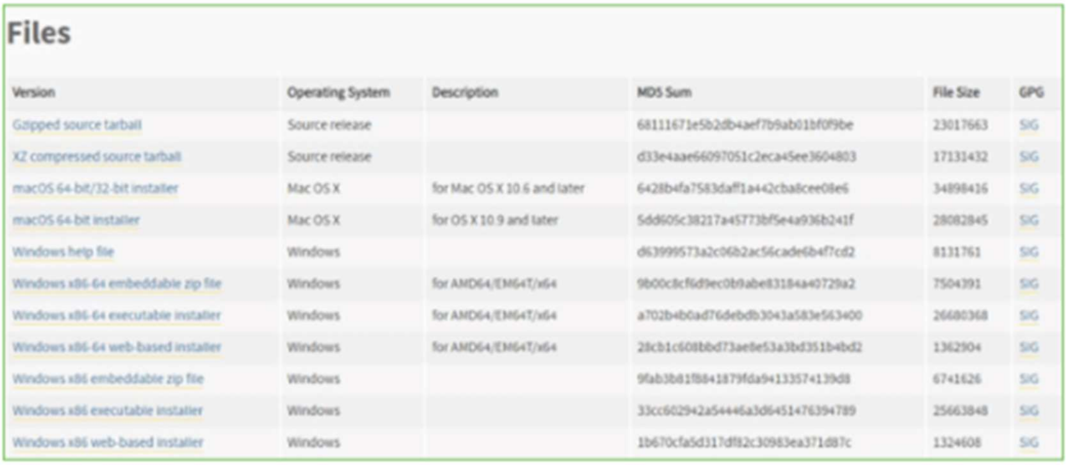
Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
Python 3.7.4	July 8, 2019	Download	Release Notes
Python 3.6.9	July 2, 2019	Download	Release Notes
Python 3.7.3	March 25, 2019	Download	Release Notes
Python 3.4.10	March 18, 2019	Download	Release Notes
Python 3.5.7	March 18, 2019	Download	Release Notes
Python 2.7.16	March 4, 2019	Download	Release Notes
Python 3.7.2	Dec. 24, 2018	Download	Release Notes

Step 4: Scroll down the page until you find the Files option.

Step 5: Here you see a different version of python along with the operating system.



Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		68111671e5b2db4aef7b9ab01bf0f9be	23017663	SIG
XZ compressed source tarball	Source release		d33e4aae66097051c2eca45ee3604803	17131432	SIG
macOS 64-bit/32-bit installer	Mac OS X	for Mac OS X 10.6 and later	6428b4fa7583daff1a442cba8cee08e6	34898416	SIG
macOS 64-bit installer	Mac OS X	for OS X 10.9 and later	5dd605c38217a45773bf5e4a936b241f	28082845	SIG
Windows help file	Windows		d63999573a2c06b2ac56cade6b4f7cd2	8131761	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	9b00c8cf6d9ec0b9abe83184a40729a2	7504391	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	a702b4b0ad76debdb3043a583e563400	26680368	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	28cb1c608bbd73ae8e53a3bd351b4bd2	1362904	SIG
Windows x86 embeddable zip file	Windows		9fab3b81f8841879fda94133574139d8	6741626	SIG
Windows x86 executable installer	Windows		33cc602942a54446a3d6451476394789	25663848	SIG
Windows x86 web-based installer	Windows		1b670cfa5d317df82c30983ea371d87c	1324608	SIG

•To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.

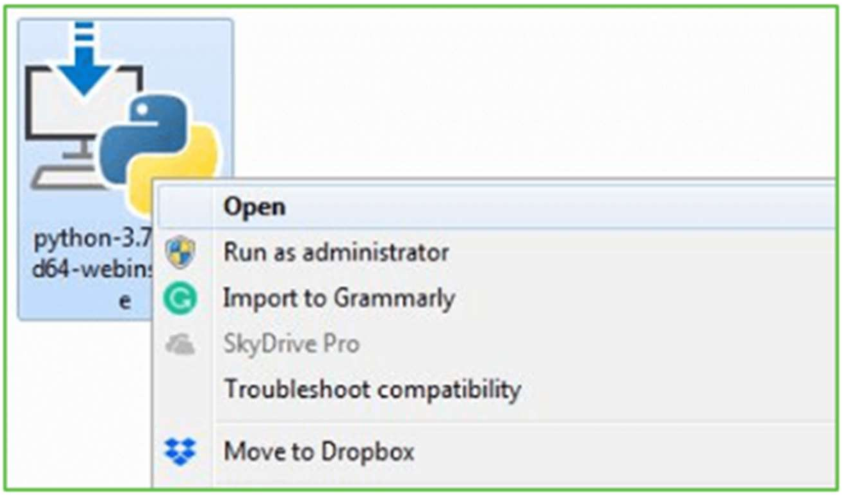
•To download Windows 64-bit python, you can select any one from the three options: Windows x86-64 embeddable zip file, Windows x86-64 executable installer or Windows x86-64 web-based installer.

Here we will install Windows x86-64 web-based installer. Here your first part regarding which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e. Installation

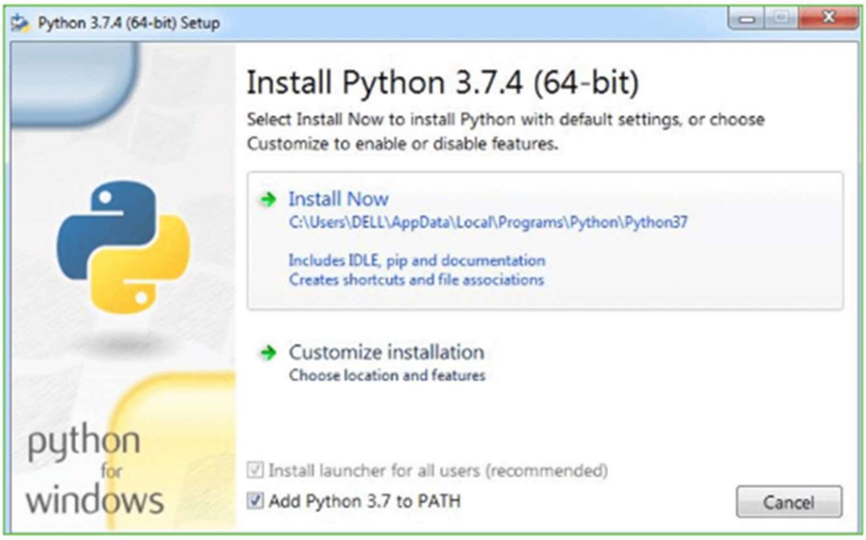
Note: To know the changes or updates that are made in the version you can click on the Release Note Option.

Installation of Python

Step 1: Go to Download and Open the downloaded python version to carry out the installation process.



Step 2: Before you click on Install Now, Make sure to put a tick on Add Python 3.7 to PATH.



Python 3.7.4 (64-bit) Setup

Install Python 3.7.4 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\DELL\AppData\Local\Programs\Python\Python37

Includes IDLE, pip and documentation
Creates shortcuts and file associations

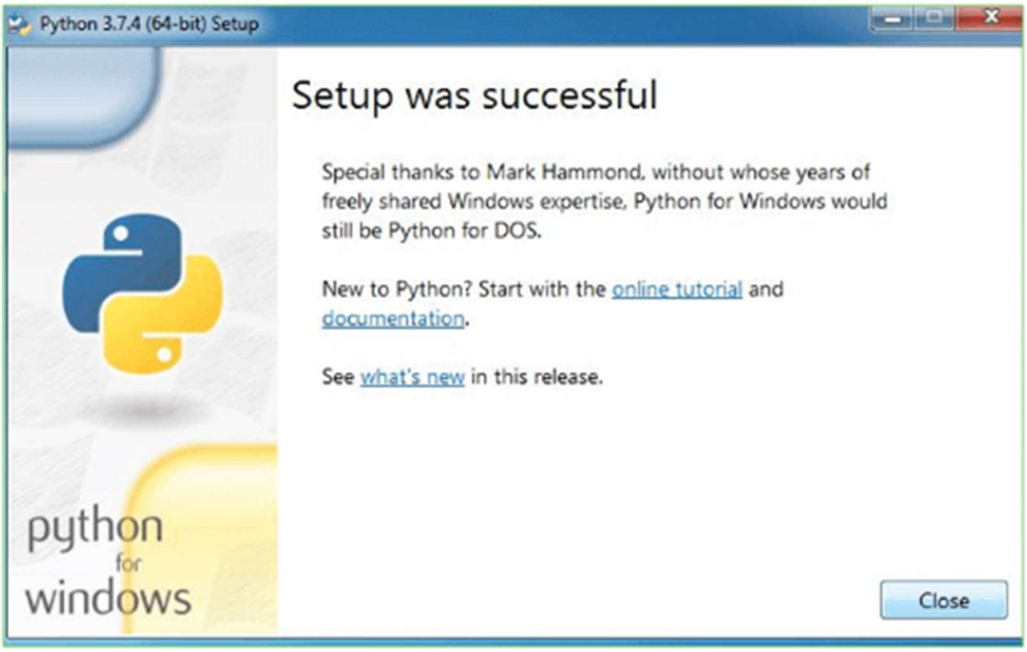
Customize installation
Choose location and features

☑ Install launcher for all users (recommended)

☑ Add Python 3.7 to PATH

Cancel

Step 3: Click on Install NOW After the installation is successful. Click on Close.



Setup was successful

Special thanks to Mark Hammond, without whose years of freely shared Windows expertise, Python for Windows would still be Python for DOS.

New to Python? Start with the online tutorial and documentation.

See what's new in this release.

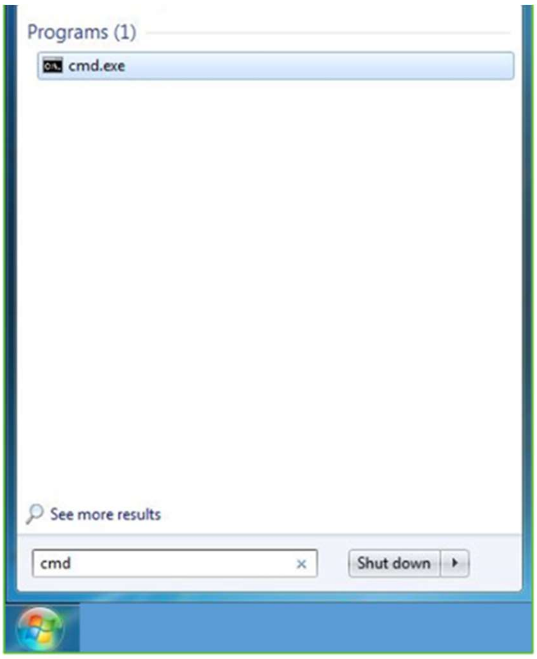
With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

Note: The installation process might take a couple of minutes.

Verify the Python Installation

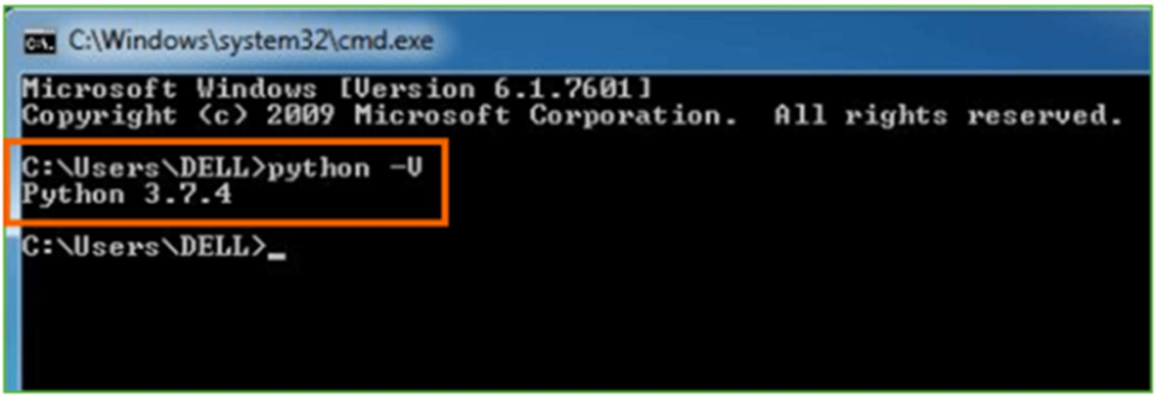
Step 1: Click on Start

Step 2: In the Windows Run Command, type “cmd”.



Step 3: Open the Command prompt option.

Step 4: Let us test whether the python is correctly installed. Type python –V and press Enter.



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\DELL>python -V
Python 3.7.4

C:\Users\DELL>_
```

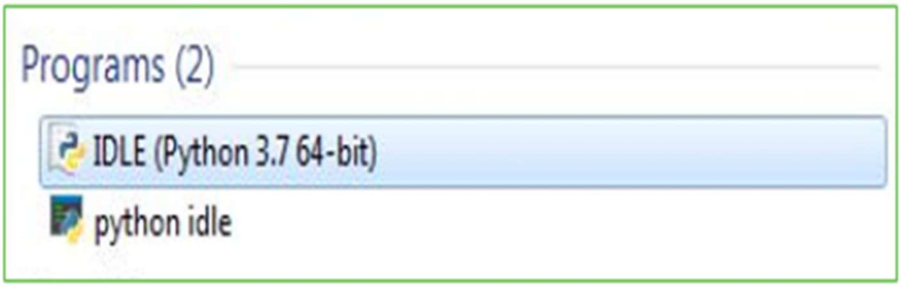
Step 5: You will get the answer as 3.7.4

Note: If you have any of the earlier versions of Python already installed. You must first uninstall the earlier version and then install the new one.

Check how the Python IDLE works

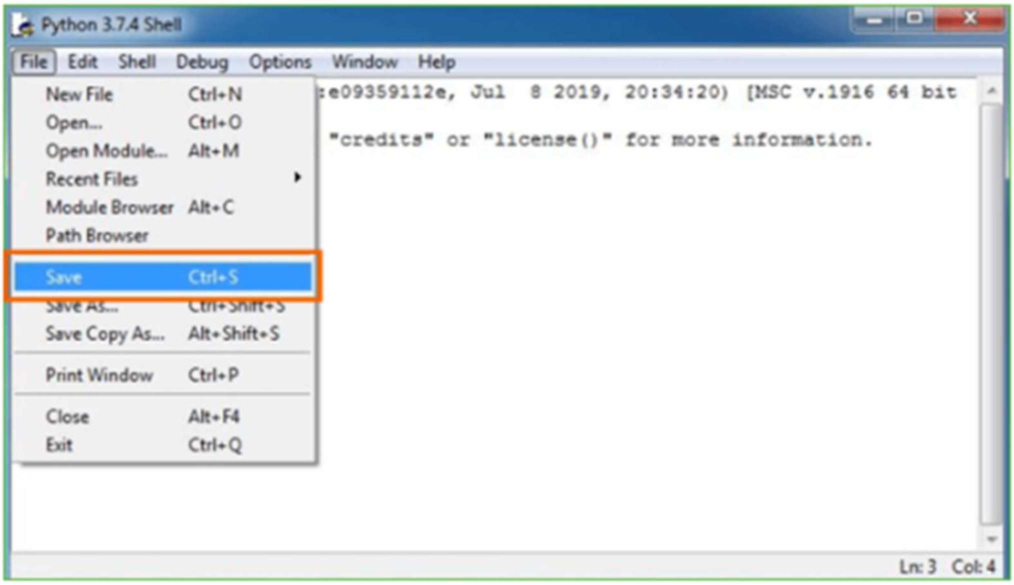
Step 1: Click on Start

Step 2: In the Windows Run command, type “python idle”.



Step 3: Click on IDLE (Python 3.7 64-bit) and launch the program

Step 4: To go ahead with working in IDLE you must first save the file. Click on File > Click on Save

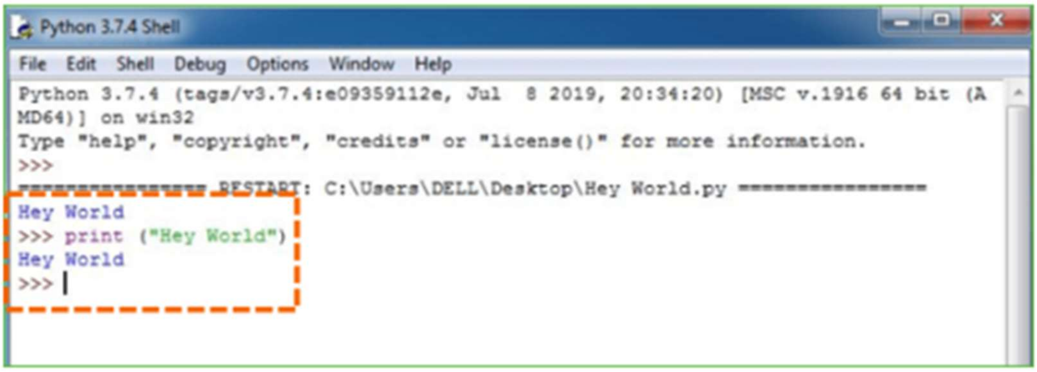


```
:e09359112e, Jul  8 2019, 20:34:20) [MSC v.1916 64 bit

 "credits" or "license()" for more information.
```

Step 5: Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

Step 6: Now for e.g. enter print ("Hey World") and Press Enter.



```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 20:34:20) [MSC v.1916 64 bit (A
MD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=============== RESTART: C:\Users\DELL\Desktop\Hey World.py ===============
Hey World
>>> print ("Hey World")
Hey World
>>> 
```

You will see that the command given is launched. With this, we end our tutorial on how to install Python. You have learned how to download python for windows into your respective operating system.

Note: Unlike Java, Python does not need semicolons at the end of the statements otherwise it won't work.

CHAPTER-6: SYSTEM STUDY

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

1. ECONOMICAL FEASIBILITY
2. TECHNICAL FEASIBILITY
3. SOCIAL FEASIBILITY

6.1 ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased**.**

6.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

6.3 SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

7. TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

7.1 TYPES OF TESTS

7.1.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.1.2. Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

7.1.3. Functional testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items
Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised
Systems/ Procedures : interfacing systems or procedures must be invoked.
Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

7.1.4 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

7.1.5 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results:

All the test cases mentioned above passed successfully. No defects encountered

8.1 **Home Page**

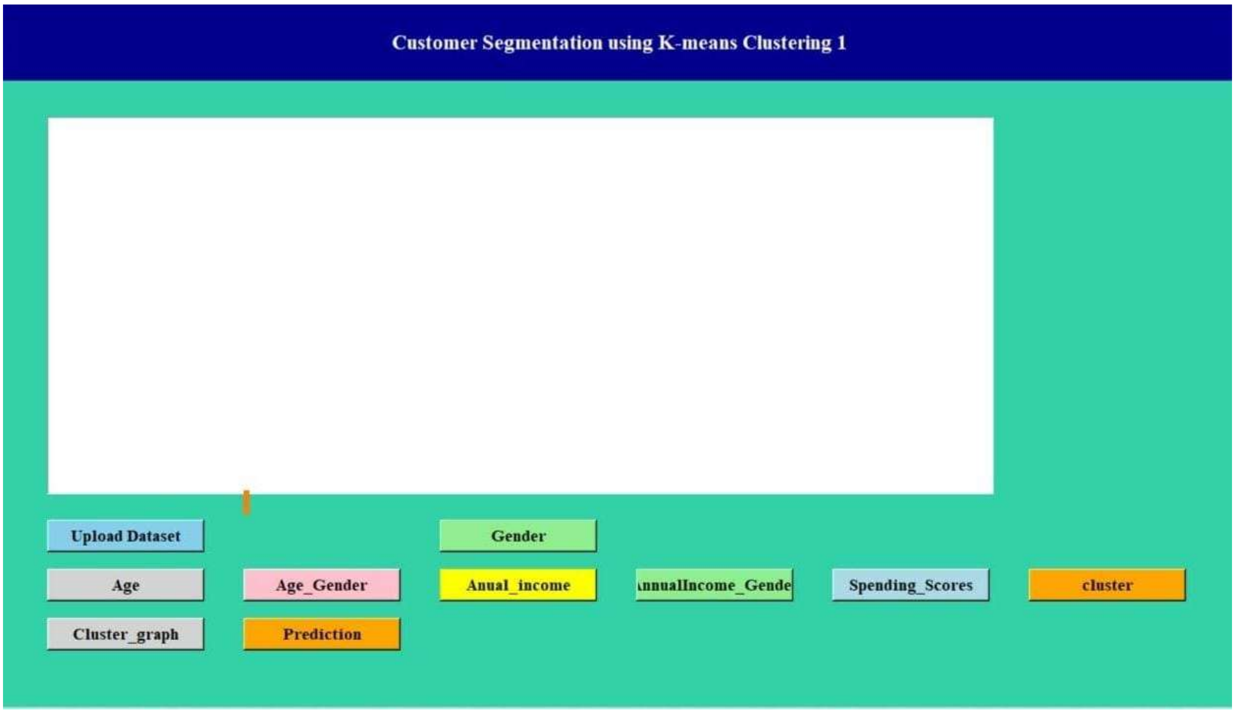
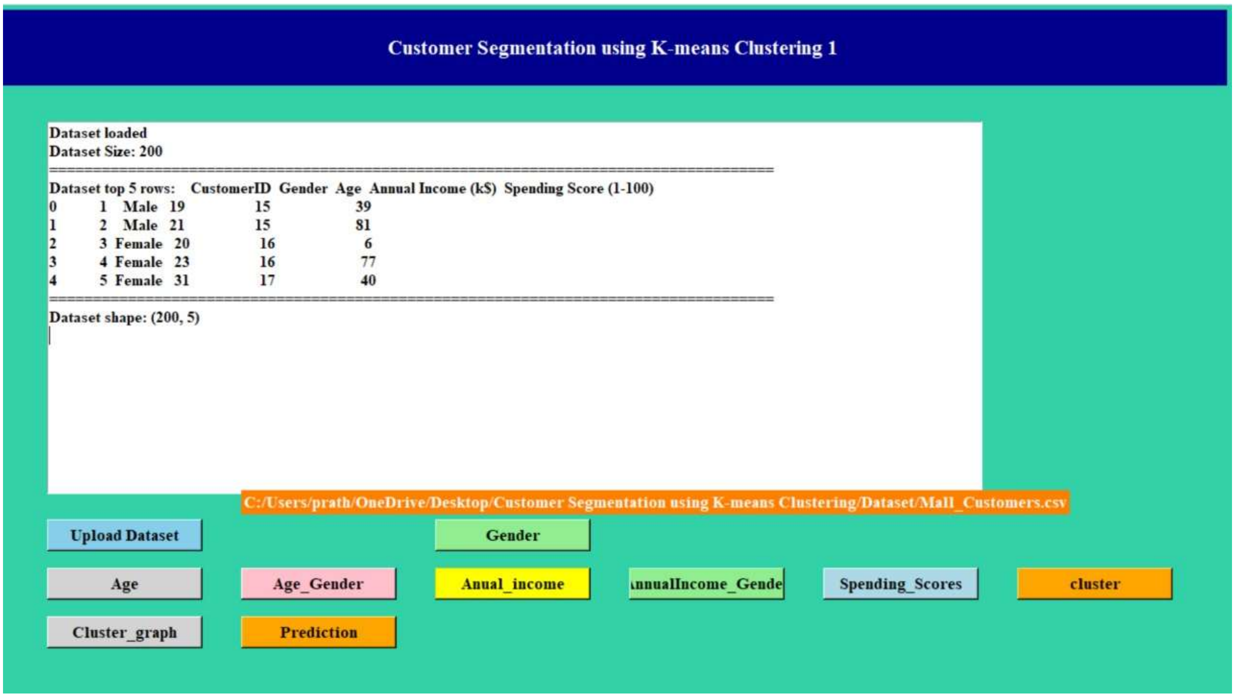


Fig.8.1 : Screenshot of Home Page

8.2 **Upload Dataset**



```
Dataset loaded
Dataset Size: 200
================================================================================
Dataset top 5 rows:    CustomerID  Gender  Age  Annual Income (k$)  Spending Score (1-100)
0         1   Male   19            15                39
1         2   Male   21            15                81
2         3  Female  20            16                 6
3         4  Female  23            16                77
4         5  Female  31            17                40
================================================================================
Dataset shape: (200, 5)
```

C:/Users/prath/OneDrive/Desktop/Customer Segmentation using K-means Clustering/Dataset/Mall_Customers.csv

Fig.8.2 : Screenshot of Uploading Dataset

8.3 **Gender Distribution**

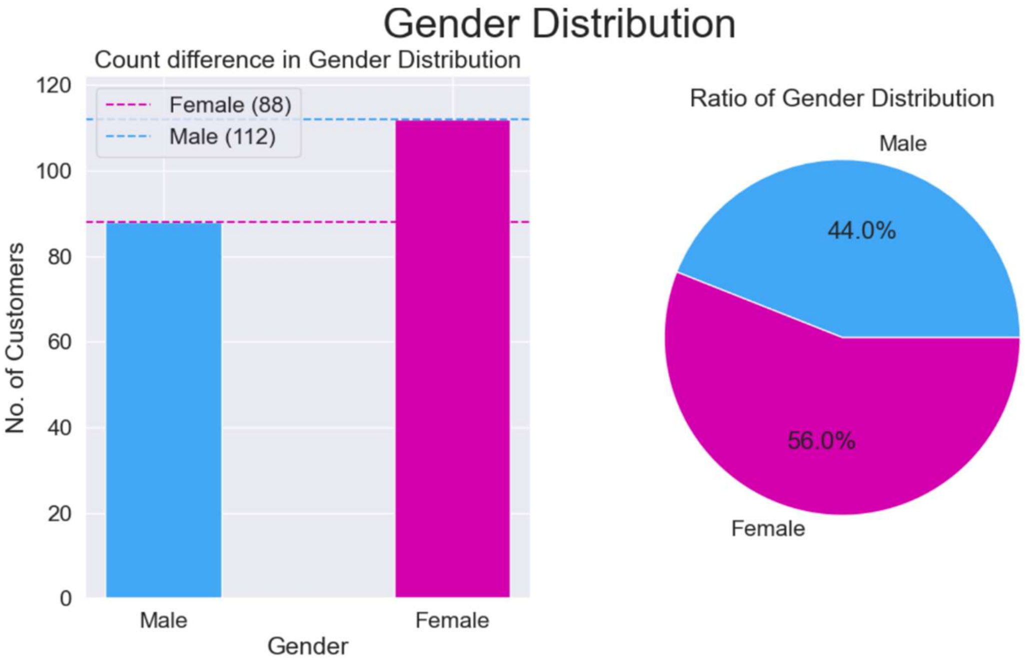


Fig.8.3: Screenshot of Gender Distribution

8.4 **Age Distribution**

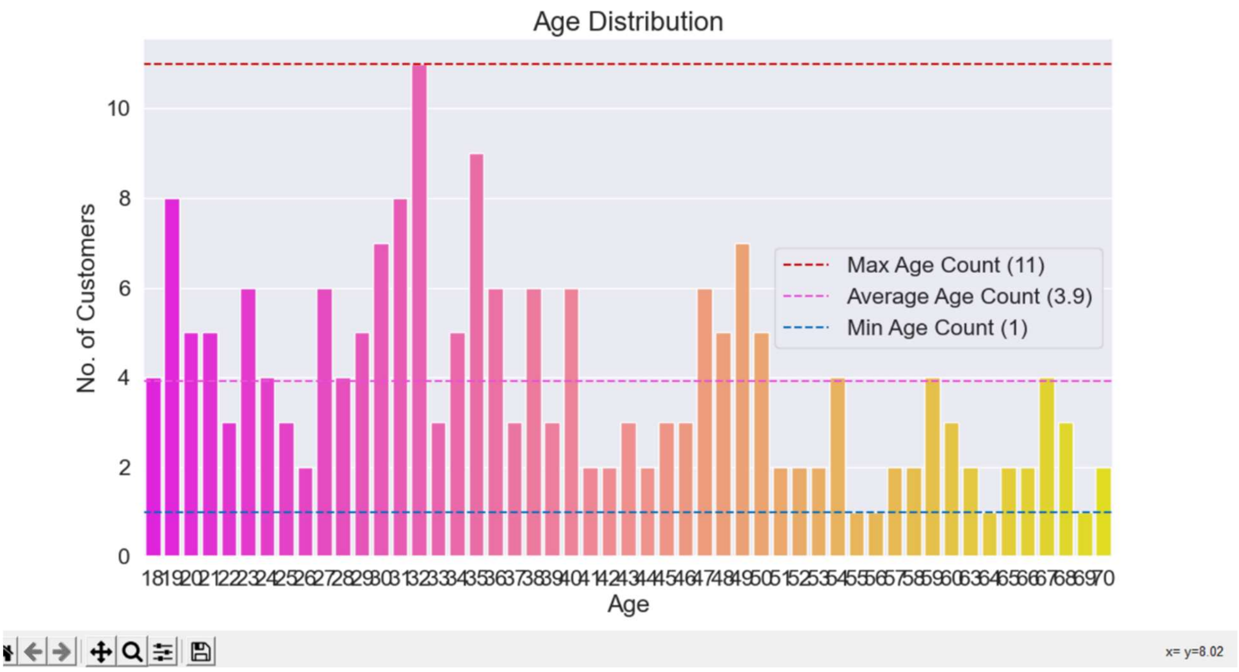


Fig.8.4 : Screenshot of Age Distribution

8.5 **Age and Gender Distribution**

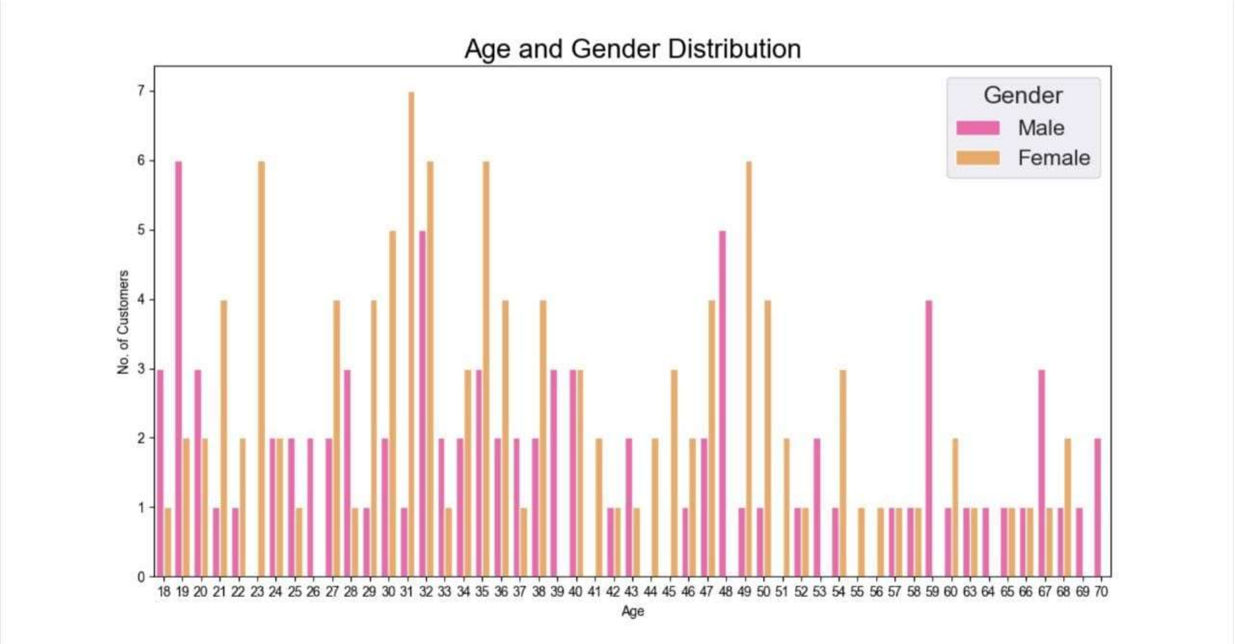


Fig.8.5: Screenshot of Age and Gender Distribution

8.6 **Annual Income count Distribution of Customers**

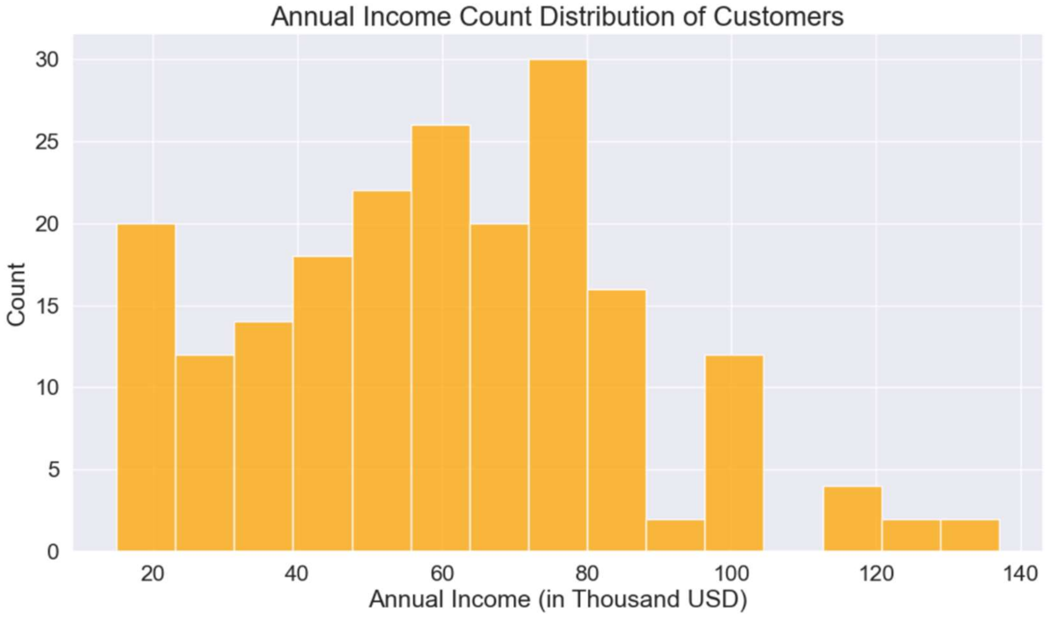


Fig.8.6 : Screenshot of Annual Income count Distribution of Customers

8.7 **Annual Income per Age by Gender**

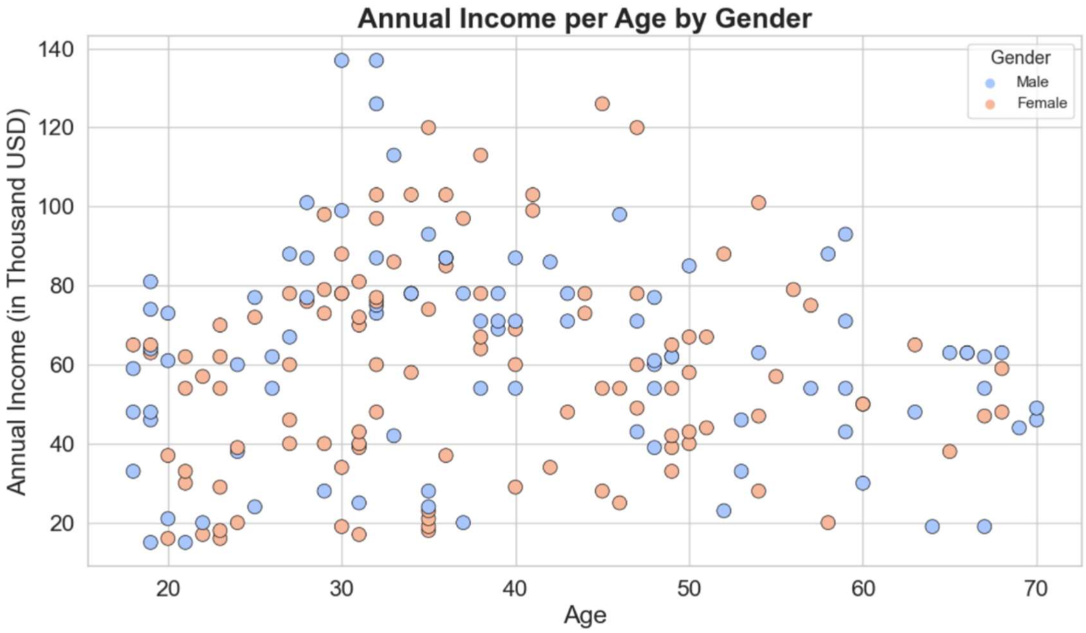


Fig.8.7 : Screenshot of Income per age by Gender

8.8 **Average Annual Income by Gender**

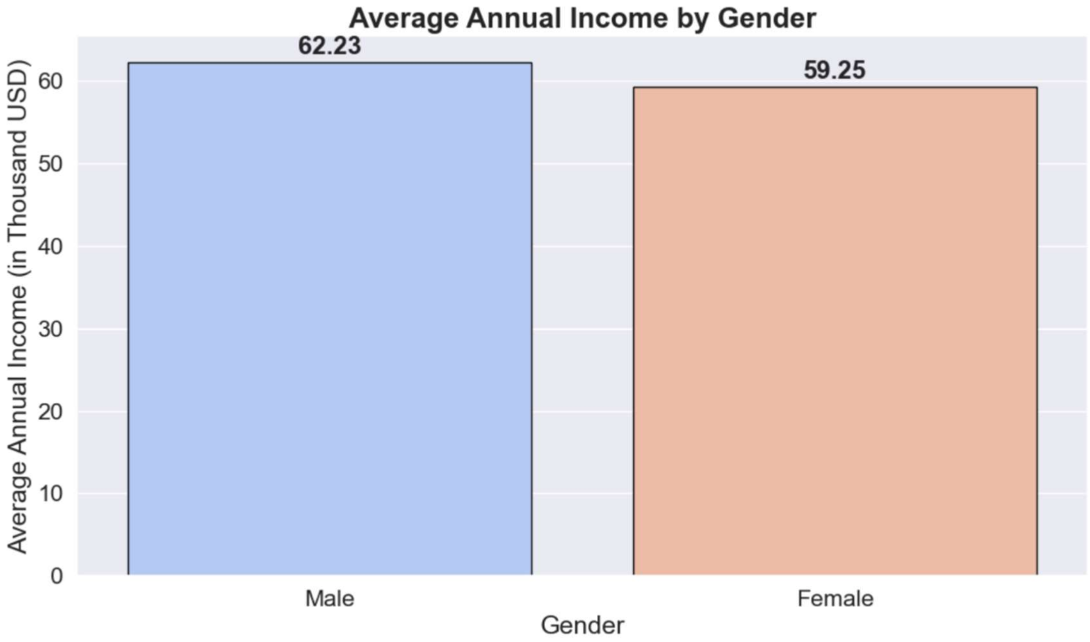


Fig.8.8: Screenshot of Average Annual Income by Gender

8.9 **Spending Scores Distribution by Age**

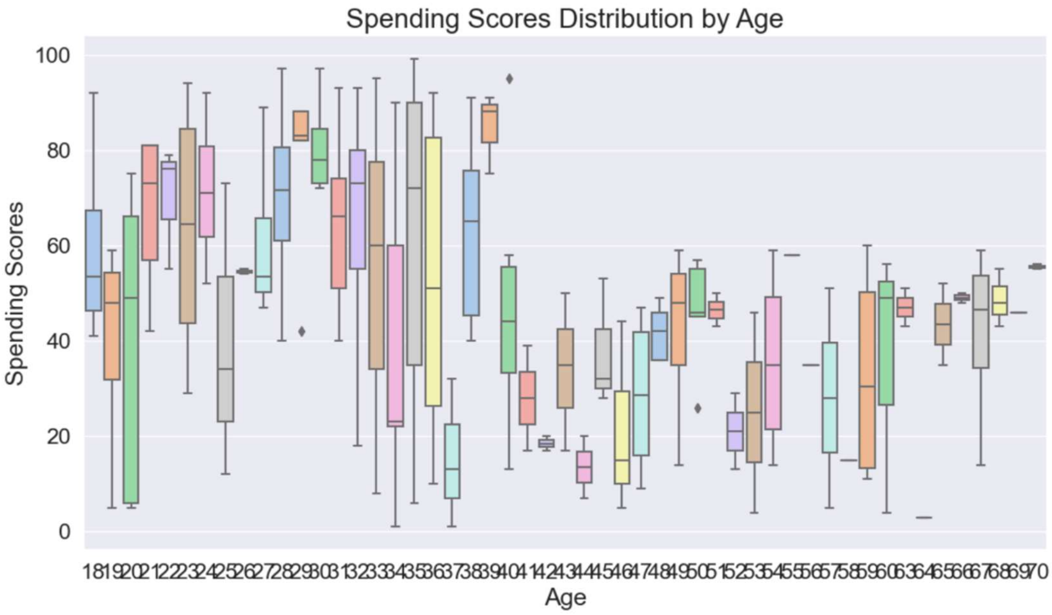


Fig.8.9: Screenshot of Spending Scores Distribution by Age

8.10 **Elbow Method**

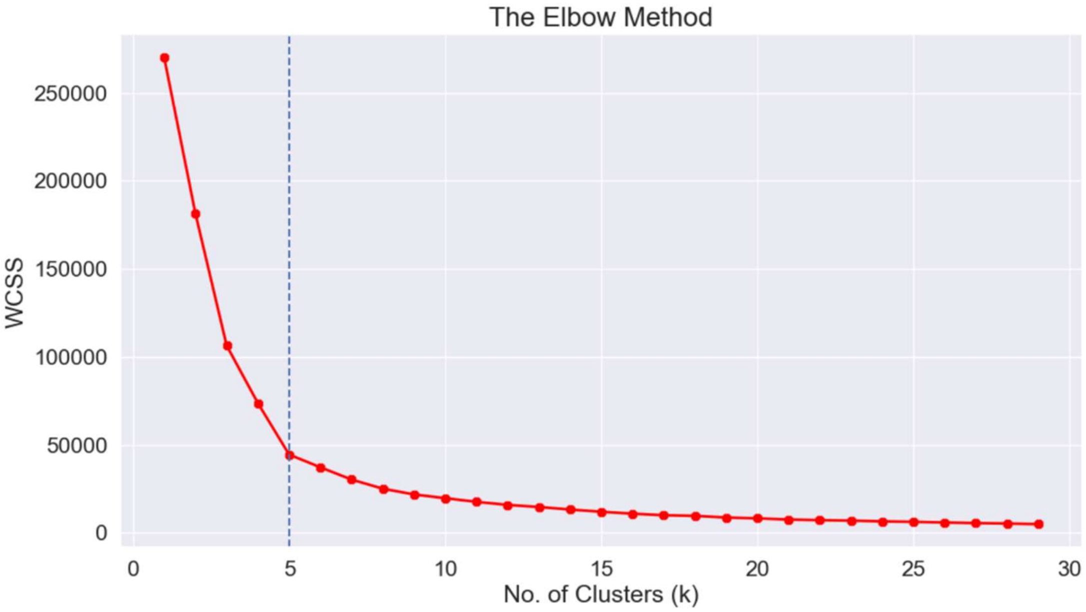


Fig.8.10 : Screenshot of a Graph using Elbow Method

8.11 **Cluster**

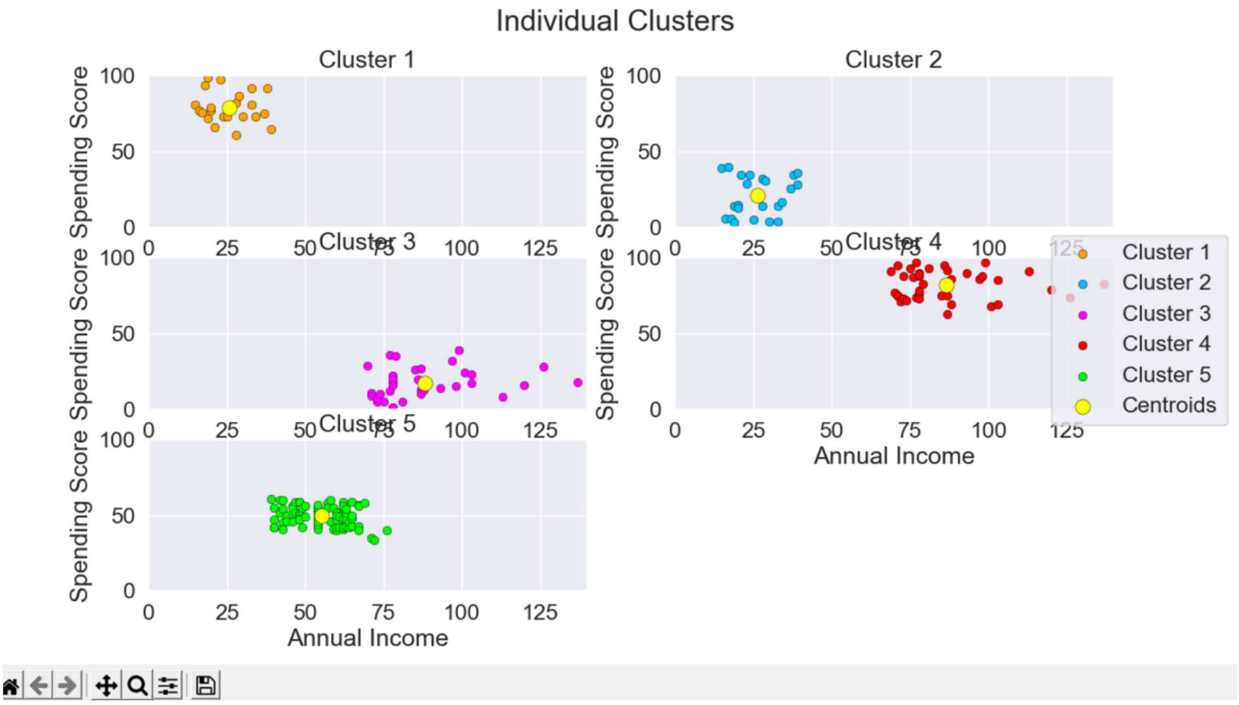
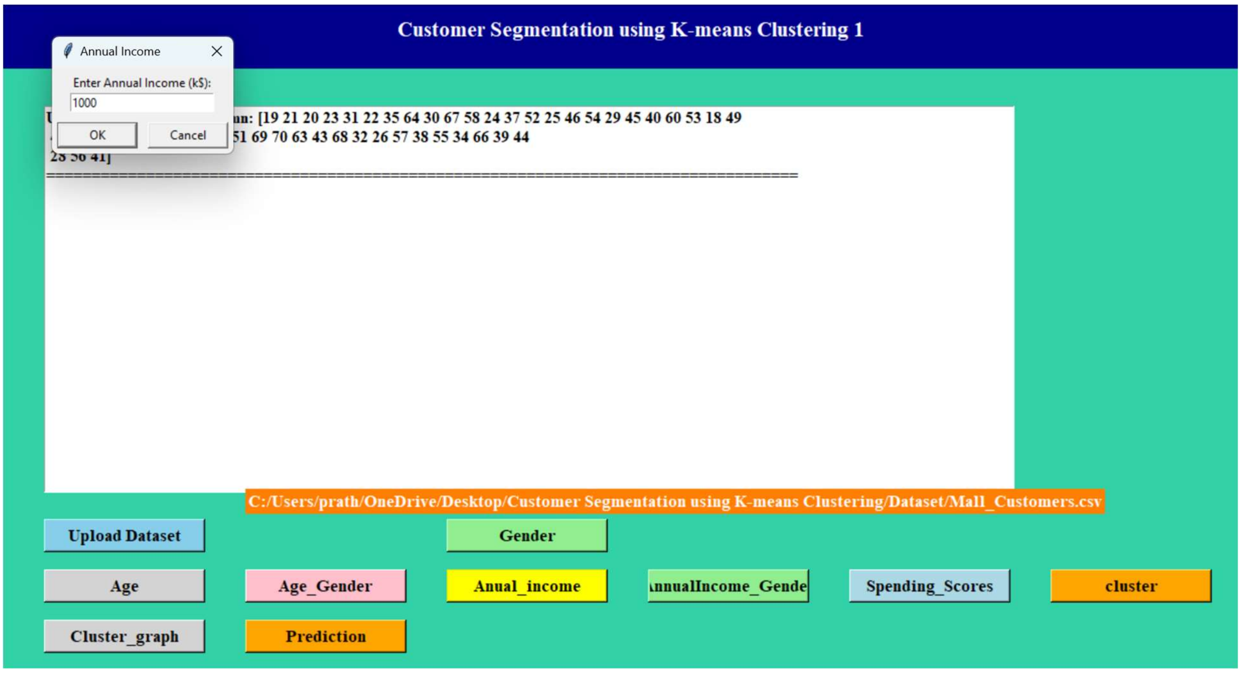


Fig.8.11 : Screenshot of Different types of Clusters

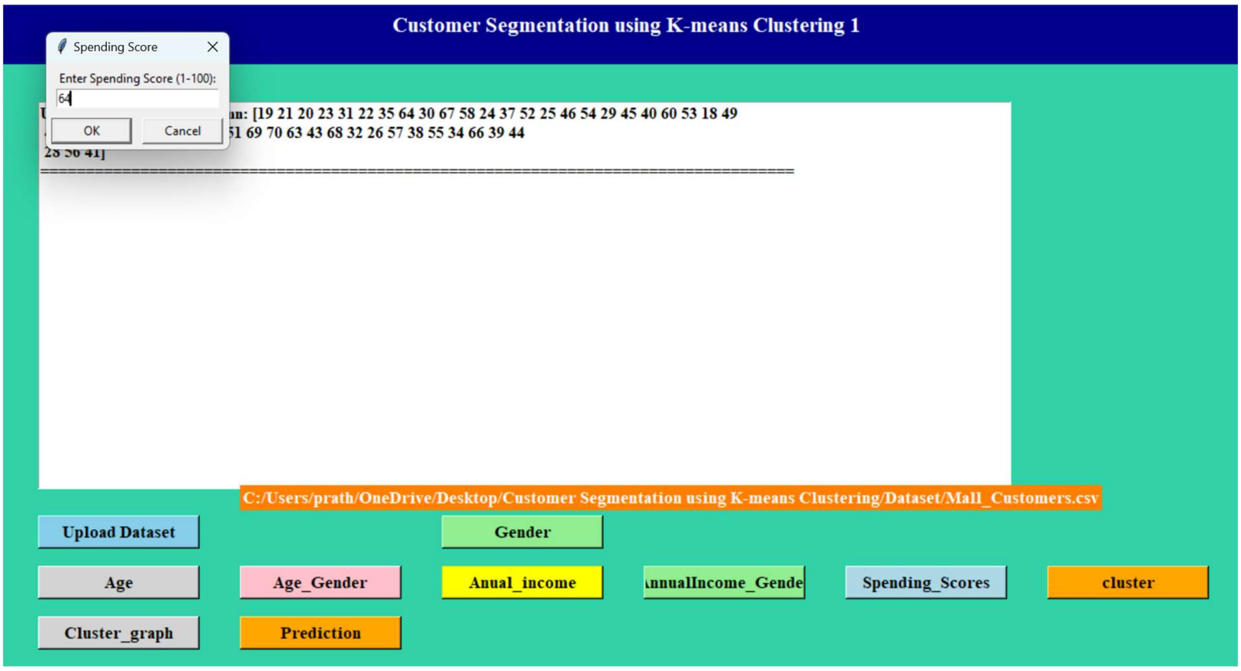
8.12 **Prediction by Annual Income**


Customer Segmentation using K-means Clustering 1
Annual Income
Enter Annual Income (k$):
1000
OK
Cancel
C:/Users/prath/OneDrive/Desktop/Customer Segmentation using K-means Clustering/Dataset/Mall_Customers.csv
Upload Dataset
Gender
Age
Age_Gender
Anual_income
nnualIncome_Gende
Spending_Scores
cluster
Cluster_graph
Prediction

```
U                 nn: [19 21 20 23 31 22 35 64 30 67 58 24 37 52 25 46 54 29 45 40 60 53 18 49
                  51 69 70 63 43 68 32 26 57 38 55 34 66 39 44
28 56 41]
=============================================================================
```

Fig.8.12 : Screenshot of Prediction by Annual Income

8.13 **Prediction by Spending Score**



Customer Segmentation using K-means Clustering 1

Spending Score

Enter Spending Score (1-100):

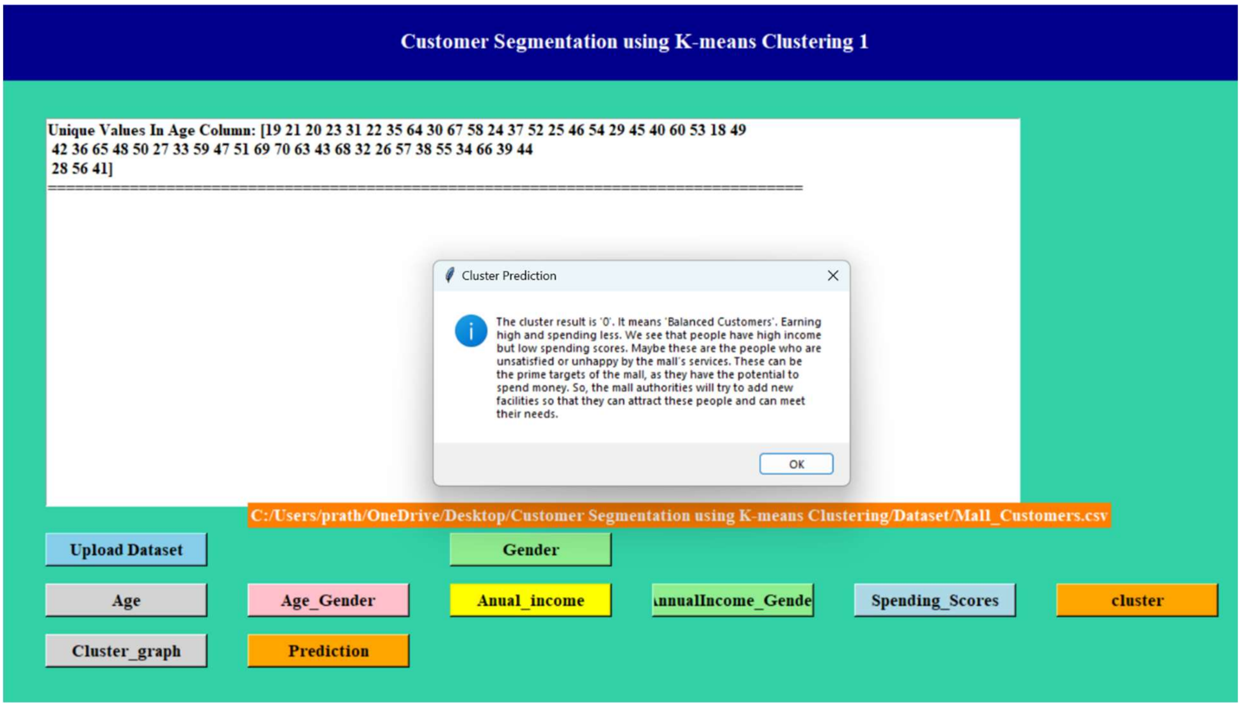
64

```
U                  m: [19 21 20 23 31 22 35 64 30 67 58 24 37 52 25 46 54 29 45 40 60 53 18 49
                  51 69 70 63 43 68 32 26 57 38 55 34 66 39 44
 28 56 41]
================================================================================
```

C:/Users/prath/OneDrive/Desktop/Customer Segmentation using K-means Clustering/Dataset/Mall_Customers.csv

Fig.8.13: Screenshot of Prediction by Spending Score

8.14 **Prediction of Final Output**



Customer Segmentation using K-means Clustering 1

```
Unique Values In Age Column: [19 21 20 23 31 22 35 64 30 67 58 24 37 52 25 46 54 29 45 40 60 53 18 49
 42 36 65 48 50 27 33 59 47 51 69 70 63 43 68 32 26 57 38 55 34 66 39 44
 28 56 41]
================================================================================
```

Cluster Prediction

The cluster result is '0'. It means 'Balanced Customers'. Earning high and spending less. We see that people have high income but low spending scores. Maybe these are the people who are unsatisfied or unhappy by the mall's services. These can be the prime targets of the mall, as they have the potential to spend money. So, the mall authorities will try to add new facilities so that they can attract these people and can meet their needs.

C:/Users/prath/OneDrive/Desktop/Customer Segmentation using K-means Clustering/Dataset/Mall_Customers.csv

Fig.8.14 : Screenshot of Prediction of Final Output

CHAPTER 9:CONCLUSION & FUTURE ENHANCEMENT

This documentation consists of a developed suspect detection and tracking system by mining large-scale transit records. The system assists in identifying pickpocket suspects and enables active surveillance in high-risk areas such as Metro stations. Specifically, firstly construct a feature representation for profiling passengers. Then, established a novel two-step framework to distinguish regular passengers from pickpocket suspects. Finally, real-world datasets from multiple sources for model training and validation are used and implemented a prototype system for end users. Experimental results on real-world data showed the effectiveness of our proposed approach.

FUTURE ENHANCEMENT:

Various activity patterns are studied and analyzed to detect anomaly behaviors. Anomaly detection from the travel log or trajectory data is a more promising process nowadays. This is very challenging due to several reasons such as mobility patterns are unique in every user; the trajectories are dynamic and need frequent updating. Detecting outlier/ anomaly from those dynamic and updatable datasets need more concentration. The techniques should be developed carefully. Surveyed various applications related to the public transit records such as identifying pickpocket suspects, daily activity pattern detection, bus route planning, traffic abnormality detection, etc. From this summary; a new technique can be developed and integrated into the recent application

CHAPTER10:BIBLIOGRAPHY

1. Jing Zhang,Ting Fan1, Ding Lang , Yuguang Xu , Hong-an Li and XueweLi, "Intelligent crowd sensing pickpocketing group identification using remote sensing data for secure smart cities", Mathematical Biosciences and Engineering, Vol.20, Issue 8,PP. 13777–13797,2023.

2. Satya Hari Prasad, Sri.S.K.Alisha, Sri.V.Bhaskara Murthy, "Detecting Pickpocket Suspects From Large-Scale Public Transit Records", jespublication, ISSN:0377-9254, Vol 13 Issue 07,PP.234-239,2022.

3. P Naveen Kumar and B. N. Srinivasa Gupta, "Analyzing Large-Scale Public Transit Records to Identify Pickpocket Suspects", International Journal for Modern Trends in Science and Technology, ISSN: 2455-3778, Vol.8(09), pp. 146-149,2022

4. S. Sajini,T. Madhan, E. Kathiravan, J. Mervin Kumar, "Recognizing Pickpocket Suspects from Enormous Scale Open Travel Records", International Journal of Research in Engineering, Science and Management, ISSN: 2581-5792, Vol:3, PP: 975-977,2020.

5. B. Du, C. Liu, W. Zhou, Z. Hou and H. Xiong, "Detecting Pickpocket Suspects from Large-Scale Public Transit Records," in IEEE Transactions on Knowledge and Data Engineering, ISSN: 1041-4347, vol. 31, no. 3, pp.465-478,2019.

6.Cang-Hong Jin, Dong-Kai Chen, Fan-Wei Zhu, Ming-Hui Wu, "Detecting Suspects by Large-Scale Trajectory Patterns in the City", Mobile Information Systems, ISSN:1837594, Vol.19, PP.11, 2019.

7. Stephanie Sharp, Richard Timothy Coupe, "Pickpocketing on Railways" , Crime Solvability Factors,ISBN 978-3-030-17160-5,Vol: 24 , PP 89-106,2019.

8.Haishuo Gu; Yaqi Guo; Huigen Yang; Peng Chen; Minggang Yao; Jiaqi Hou, "Detecting Pickpocketing Offenders by Analyzing Beijing Metro Subway Data", IEEE 4th International Conference on Big Data Analytics (ICBDA), pp. 62-66,2019.

9.Cindy Verleysen ,“Pickpocketing committed by mobile organised crime groups”, EUCPN Theoretical Paper ,Vol :23, PP:234-238,2017.

10.YAN Mi-qiao, GUO Zhong-yang, REN Zhe-hao, Spatio-temporal analysisof bus pickpocketing using association rules based on clustering of EastJanez Mekinc, Rob Mawby & Anita Trnavcevic, “Security and Tourism in European Cities - TheMultiple Case Study”, Lex Localis - Journal Of Local Self-Government, Vol. 15,PP: 359 -385,2017.

12.Bowen Du ,Chuanren Liu,Wenjun Zhou,Zhenshan Hou, Hui Xiong, “Catch Me IfYou Can: Detecting Pickpocket Suspects from Large-Scale Transit Records”,KDD

13.Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining ISBN:9781450342322,Vol.7,pp.2176,2016.

14.PETER K. ANDERSSON, “Bustling, crowding, and pushing’: pickpockets and the nineteenth-century street crowd”, Urban History, Vol:41,PP:291-310,2014.

15.Henri Bouma, Jan Baan, Gertjan J. Burghouts, Pieter T. Eendebak, Jasper R. van Huis, Judith Dijk, Jeroen H. C. van Rest, “Automatic detection of suspicious behavior of pickpockets with track-based features in a shopping mall”, spiedigitallibrary,Vol:13,PP:20,2014.

16.Xia Zhao, Yong Zhang, Hao Liu, Shaofan Wang, Zhen Qian, Yongli Hu, Baocai Yin, “Detecting Pickpocketing Gangs on Buses with Smart Card Data”, IEEE Intelligent Transportation Systems Magazine, vol.11,pp. 181-199, 2019.

17.Hao Zhang, Ole Lund, Morten Nielsen , “The Pick Pocket method for predicting binding specificities for receptors based on receptor pocket similarities: application to MHC-peptide binding”, Bioinformatics, Vol :

18. Andrew Newton, Henry Partridge & Andy Gill, “In and Around: Identifying Predictors of Theft within and Near to Major Mass Underground Transit Systems” Safety and Security in Transit Environments, ISBN:978-1-349-57179,Vol:6,PP:99-115,2015.

19. Bambi Vincent, Workshop Teaches Pickpocket Recognition, IEEE, Vol: 35.PP:116-119,2005.

20. Timothy J. Gilfoyle , “Street-Rats and Gutter-Snipes: Child Pickpockets and Street Culture in New York City”, Journal of Social History, vol. 37 ,PP: 853-862.2004.

MoE's
INNOVATION CELL
(GOVERNMENT OF INDIA)

INSTITUTION'S INNOVATION COUNCIL
MOE'S INNOVATION CELL

INSTITUTION'S
INNOVATION
COUNCIL
(Ministry of Education Initiative)

Institute Name:
Malla Reddy College of Engineering for women

Title of the Innovation/Prototype:
CUSTOMER SEGMENTATION USING K-MEANS CLUSTERING

Team Lead Name: SHEERLA NEHA	**Team Lead Email:** nehasheerla@gmail.com	**Team Lead Phone:** 8247696483	**Team Lead Gender:** Female
FY of Development: 2023-24	**Developed as part of:** Academic Requirement/Study Project	**Innovation Type:** Market Place	**TRL LEVEL:** 1

Theme:
Consumer Goods and Retail,

Define the problem and its relevance to today's market / sociaty / industry need:
Consumers expect personalized experiences. Segmentation allows businesses to tailor marketing messages, product recommendations, and services to specific customer groups, improving engagement and satisfaction. By understanding different customer segments, companies can create targeted marketing campaigns that resonate more effectively with each group, leading to higher conversion rates and better return on investment (ROI).Segmentation helps businesses allocate their resources more efficiently by focusing efforts on the most profitable or promising customer segments. Insights from customer segmentation can guide product development and innovation, ensuring that new products or features meet the needs of specific segments.

Describe the Solution / Proposed / Developed:
customer segmentation using K-means clustering offers businesses a robust method to uncover actionable insights, enhance customer understanding, and drive targeted marketing efforts When comparing current segmentation using K-means clustering to earlier methods such as heuristic or traditional segmentation approaches, several significant differences and advantages can be identified.

Explain the uniqueness and distinctive features of the (product / process / service) solution:
Utilizes mathematical algorithms to identify clusters based on data patterns, allowing for more precise segmentation. It can uncover subtle variations and group customers into distinct clusters based on multiple dimensions Analyses multiple variables simultaneously, incorporating complex interactions between variables to form clusters. It can handle large datasets and various types of data (numeric, categorical) effectively.

How your proposed / developed (product / process / service) solution is different from similiar kind of product by the competitors if any:
Customer segmentation using k-means clustering can provide several unique advantages over competitor solutions, depending on how it is implemented. Customization, Flexibility, Scalability, Data Integration ,Algorithm Transparency, Cost-Effectiveness.

Is there any IP or Patentable Component associated with the Solution?:
No

Copy of IP/Patent Applied or Obtained: View File

Has the Solution Received any Innovation Grant/Seefund Support?:
Yes

Total grant fund amount (Rs.) Received from various sources: 7500

Grant fund amount (Rs.) Received From Institute/Incubation Unit: 7000

Are there any Recognitions (National/International) Obtained by the Solution?:
No

***Is the Solution Commercialized either through Technology Transfer or Enterprise Development/Startup?:**
No

Had the Solution Received any Pre-Incubation/Incubation Support?:
Yes

Pre-Incubation / Incubation Unit Name:
MRCEW-Incubation

Video URL:
https://drive.google.com/file/d/15QjAO7W39kH_LVMdwcA4slZ9WLJAG0yg/view?usp=drivesdk

Innovation Photograph:
View File

Downloaded on: 20-07-2024

This report is electronically generated against Yukti - National Innovation Repository Portal.