**MarketMetrics:** **A Big Market Analysis**

A PROJECT REPORT

*Submitted by*

|  |  |  |  |
| --- | --- | --- | --- |
| **Roll Number** | **Registration Number** | **Student Code** | **Student Name** |
| **23010201076** | **23012005191** | **BWU/MCA/23/081** | **ARSHAD MURTAZA**  **AHMED** |
| **23010201083** | **23012005198** | **BWU/MCA/23/089** | **SUBINOY KHATUA** |
| **23010201066** | **23012005181** | **BWU/MCA/23/071** | **SAYAN MALAKAR** |
| **23010201100** | **23012005216** | **BWU/MCA/23/108** | **PRAVEEN KUMAR** |
| **23010201101** | **23012005217** | **BWU/MCA/23/109** | **RIMI DAS** |

*in partial fulfillment for the award of the degree of*

**Master of Computer Application**

****

**Department of Computational Science BRAINWARE UNIVERSITY**

**398, Ramkrishnapur Road, Barasat, North 24 Parganas, Kolkata - 700 125**

**MarketMetrics:** **A Big Market Analysis**

***Submitted by***

|  |  |  |  |
| --- | --- | --- | --- |
| **Roll Number** | **Registration Number** | **Student Code** | **Student Name** |
| **23010201076** | **23012005191** | **BWU/MCA/23/081** | **ARSHAD MURTAZA**  **AHMED** |
| **23010201083** | **23012005198** | **BWU/MCA/23/089** | **SUBINOY KHATUA** |
| **23010201066** | **23012005181** | **BWU/MCA/23/071** | **SAYAN MALAKAR** |
| **23010201100** | **23012005216** | **BWU/MCA/23/108** | **PRAVEEN KUMAR** |
| **23010201101** | **23012005217** | **BWU/MCA/23/109** | **RIMI DAS** |

***in partial fulfillment for the award of the degree***

***of***

**MASTER OF COMPUTER APPLICATION**

### in

**Department of Computational Science**

****

# BRAINWARE UNIVERSITY

*398, Ramkrishnapur Road, Barasat, North 24 Parganas, Kolkata - 700 125*



**BRAINWARE UNIVERSITY**

*398, Ramkrishnapur Road, Barasat, North 24 Parganas, Kolkata - 700 125*

**Department of Computational Science**

**BONAFIDE CERTIFICATE**

Certified that this project report “MarketMetrics: A Big Market Analysis” is the bonafide work of “**Arshad Murtaza Ahmed, Subinoy khatua, Praveen Kumar, Sayan Malakar, Rimi Das ”** who carried out the project work under my supervision.

**SIGNATURE**  **SIGNATURE**

Dr. Jayanta Aich Dr. Taraknath Paul

**HEAD OF THE DEPARTMENT**   **SUPERVISOR**

Department of Computational Science Assistant Professor Brainware University, Department of Computational Science 398, Ramkrishnapur Road, Barasat,

North 24 Parganas, Kolkata 700125

------------------------------ -----------------------

External Examiner Internal Examiner

**ACKNOWLEDGEMENT**

Firstly, we would like to express our heartfelt gratitude to our project guide Dr. Taraknath Paul, for guiding us throughout the course of the project. We are highly indebted to them for their invaluable guidance and ever-ready support, which was necessary for the successful completion of the project in, stipulates time. Their deep knowledge of computer engineering field made us realize that theoretical knowledge always helps develop efficient operational industrial software, which are a blend of all core subjects of the field. Working under their guidance has been a fruitful experience, which will be very valuable for us, when we enter the corporate world.

.

We also thank our respected Head of Department,

Dr. Jayanta Aich,

for his valuable guidance and encouragement. We also thank all the staff members for their encouragement and support throughout this project and all those who have embedded us with technical knowledge of computer technology. We sincerely thank to all our friends and well-wishers for directly or indirectly helping us during the course of the work.

**Master of Computer Application**

**in**

**Department of Computational Science**

**MCA, Semester 4**

****

**BRAINWARE UNIVERSITY**

*398, Ramkrishnapur Road, Barasat, North 24 Parganas, Kolkata - 700 125*

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Contents** | **Page No:** |
|  | Abstract List of figures | **89** |
| 1. | Introduction:   * 1. Background   2. Objective   3. Purpose   4. Scope of System | **1 - 3** |
| 2. | Requirement And Analysis:   * 1. Problem Definition   2. Requirement Specification   3. Hardware Requirement   4. Software Requirement   5. Planning Scheduling | **4 - 9** |
| 3. | System design:   * 1. System Architecture   2. Preprocessing Model   3. Phases in Model   4. Data Set   5. Input and Output | **10 - 21** |
| 4. | Testing & Implementation:   * 1. Testing approach used   2. Implementation Approaches | **22 - 25** |
| 5. | Conclusion   * 1. Conclusion   2. Limitation of System   3. Future Scope of System   4. References | **26 - 27** |

**ABSTRACT**

Everyone needs fast and easy access to tools and systems that simplify their tasks, whether it's a small query or a large-scale business challenge. As the complexity of businesses grows with the increasing population and market demand, problems related to forecasting sales trends are becoming critical. It takes a lot of time to analyze appropriate data or predictions by consulting different sources and relying on traditional methods. There is no centralized facility that provides easy and convenient access to sales forecasting and prediction systems, which can offer the required insights. Currently, there is no efficient tool that enables businesses to find their nearest trends, patterns, or future sales projections without wasting much time.

MarketMetrics is the solution to this problem. MarketMetrics is a Web Application that provides a platform for businesses to access insightful sales data. This application offers advanced features for analyzing sales trends, market patterns, and potential forecasts based on historical data and specific business parameters. It enables businesses to find their desired insights related to MarketMetrics in a quick and user-friendly manner. To achieve this, a database is designed, which is managed by administrators. The feedback and suggestions provided by users are referred to by authorities for continuous improvement. Users can view MarketMetricss, filter them by various criteria, and access relevant reports easily without any extra hassle. Additionally, the platform includes a detailed overview of the services provided, ensuring users get exceptional support in interpreting and applying sales data.

This application is built to make the interaction between business owners, analysts, and data systems seamless and less time-consuming by managing multiple databases, such as historical sales data and predictive models. It has two primary modules: one at the **Administration Level** and the other for **end-users**, including business owners and analysts. The application maintains authentication to ensure secure access. The administrator's tasks include managing sales data, user information, and predictive model configurations.

The **MarketMetrics System** can be accessed using a username and password. It is accessible primarily by administrators who manage and update the database. The data can be retrieved and analyzed easily. The interface is highly user-friendly, with well-protected data to ensure privacy and efficient processing for accurate and timely insights.

# TABLE OF FIGURES

|  |  |
| --- | --- |
| **Sr. No** | **Figures** |
| **Requirement and Analysis:** | |
| 1 | Spiral model |
| **System Design:** | |
| 2 | System Architecture |
| 3 | Preprocess Model |
| 4 | Phases in Model |
| 5 | Data Set |
| **Input Output Design:**  **User Authority:** | |
| 6 | Home Page |
| 7 | Output Page |
| **Testing:** | |
| 8 | Black Box Testing |
| 9 | White Box Testing |
| 10 | Grey Box Testing |

## Chapter 1: Introduction

In the rapidly evolving landscape of business and commerce, efficient MarketMetrics plays a pivotal role in ensuring informed decision-making and achieving strategic goals. As the demand for accurate forecasting and data-driven insights continues to rise, businesses are increasingly turning to innovative solutions to optimize their operations. In this report, we delve into the realm of MarketMetrics systems and explore their potential to revolutionize the way businesses plan and function.

The objective of this introduction is to present the key features and benefits of the MarketMetricssa system, highlighting its potential to transform traditional business forecasting practices. By embracing advanced technologies and predictive analytics, businesses can empower their teams, streamline planning processes, and ultimately improve their operational efficiency and profitability.

The MarketMetrics system encompasses a wide range of functionalities, addressing every aspect of sales forecasting and trend analysis. It serves as a comprehensive solution that integrates various modules. By consolidating these functions into a single platform, the system eliminates inefficiencies, reduces manual effort, and facilitates real-time access to critical market insights.

Moreover, the MarketMetrics system plays a pivotal role in data management and analysis. By capturing and processing vast amounts of sales data, it provides businesses with valuable insights into market trends, resource allocation, and revenue projections. This data-driven approach enables informed decision-making, facilitates performance monitoring, and supports continuous improvement efforts in sales strategies.

While the implementation of a MarketMetrics tion system presents numerous benefits, it is not without challenges. Issues such as data accuracy and security, staff training, and system integration must be carefully addressed to ensure successful adoption and utilization. By proactively addressing these challenges, businesses can maximize the potential benefits of the system while minimizing risks.

In conclusion, the MarketMetrics system represents a transformative solution that empowers businesses to optimize their forecasting and enhance decision-making. By integrating various analytical functions into a single platform, fostering communication and collaboration among teams, and leveraging advanced data analytics, this system has the potential to revolutionize sales management practices. Through this report, we will explore the features, benefits, challenges, and implementation strategies associated with the MarketMetrics system.

**1.1 Background**

|  |  |
| --- | --- |
| **Project Description** | MarketMetrics:A Big Market Analysis, it is a web-based application.. |
| **Project Duration** | 2 to 3 Month. |
| **Project Guide** | Assistant Professor Dr. Taraknath Paul |
| **Platform** | Windows 11 |
| **Technologies Used** | HTML, CSS and Python (pandas, numpy, seaborn, matplotlib, klib, dtale, scikit-learn, joblib, pandas-profiling) |
| **Tools Used** | Microsoft Visual Studio 2019 WPS Office. |

**1.2 Objective**

The objective of a MarketMetrics system is to streamline and optimize the forecasting and planning processes within a business organization. The system aims to enhance the efficiency, accuracy, and effectiveness of various MarketMetrics functions, ultimately leading to improved decision-making and overall organizational performance. The key objectives of a MarketMetrics system include:

* This application is developed to provide users with easy access to check sales trends, forecast future sales, and analyze market patterns effectively.
* This application can also take user inquiries, suggestions, and feedback to improve communication and provide better solutions to business challenges.
* This application also helps in generating detailed sales reports and scheduling forecasts to assist in strategic planning.

**1.3 Purpose**

This website provides a better way for the user to check sales trends and access detailed insights about market patterns and business forecasts, with great detail to ensure users are confident in making informed decisions.

This system provides a more efficient and modern approach compared to the traditional methods of sales analysis used in previous decades.

**1.4 Scope System**

This application is developed to provide easy access for users to analyze sales trends, forecast future sales, and make data-driven decisions.

This application can also take user inquiries, suggestions, and feedback to improve communication and provide a better way to address business challenges.

This **MarketMetrics System** will be a website whose main programming language will be Python. Its primary aim is to simplify and improve the MarketMetrics process for users, minimizing manual data entry and ensuring data accuracy and security.

**Chapter 2: Requirement and Analysis**

* The existing system is manual system.
* Needs to be converted into automated system.
* As it has a risk of mismanagement of data, less Security, no proper coordination between die rent Applications and Users, fewer users-friendly, accuracy not guaranteed and not in reach of distant users.

**2.2 Requirement Specification**

* ****A system’s requirement analysis is an important component of the system development process.
* This perhaps the most important and essential ingredient of the system analysis phase and its proper completion ensures the success of the entire system.
* It establishes what the new system must do, it involves identifying who needs what information, where, when and how.
* It also identifies the data, process and interface requirements for the users of the new system.
* Errors and omissions in requirement analysis result in user dissatisfaction with the final system and it will force to be highly cost and incur heavily loss.
* The ultimate goal of the requirement analysis is the creation of the requirement specification for the new system.

**Functional Requirement:**

The only requirement is to automate the whole system as a reliable source of providing accurate MarketMetrics information so that users can get the maximum benefit from the insights and data provided by the system.

**Non-functional Requirement:**

The supplementary specification applies to the **MarketMetrics System**. This specification defines the non-functional requirements of the system, such as:

***Functionality:***  
Since it’s a web application, one or more users may access it simultaneously via the web over the internet.

***Usability:***  
Web browser interface  
Any operating system running the latest version of the browser.

***Reliability:***  
The system is available during online mode using the internet.

***Performance:***  
The performance depends on the hardware specifications of the server used.

**2.3 Hardware Requirements**

* Processor Dual core or above
* HDD 500 GB
* RAM 8 GB

**2.4 Software Requirements**

**Frontend Technology:**

HTML, CSS and Browser

**Backend tools:**

Python(Flask Framework)

**2.5 Planning Scheduling**

### Software Project Planning:

### Goal is to establish a pragmatic strategy for controlling, tracking, and monitoring a complex technical project

### Must deal with:

### Project complexity: has a strong effect but is heavily influenced by past practitioner experience

### Project size: as size increases the inter dependency of elements also grows. Watch out for scope creep (when customers change requirements mid-cycle)

### The degree of structural uncertainty: the degree to which requirements are solidified and the ease of functional decomposition.

The purpose of project planning is to ensure that the end result is completed on time, within budget, and exhibits quality!

***Project Planning***

During the development of project, we have used the Spiral model. A brief of Incremental model and how this model is helpful to us for developing this project is given below.

Project planning was done to define the scope of the project, assess risks, and estimate and schedule project activities and thereby lay the foundation for the execution, monitoring and control of the project.

**Spiral Model**

The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. This Spiral model is a combination of iterative development process model and sequential linear development model i.e., the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around the spiral.

**Spiral Model – Design:**

The spiral model has four phases. A software project repeatedly passes through these phases in iterations called Spirals.

**Identification:**

This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase.

This phase also includes understanding the system requirements by continuous communication between the customer and the system analyst. At the end of the spiral, the product is deployed in the identified market.

**Design:**

The Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and the final design in the subsequent spirals.

**Construct or Build:**

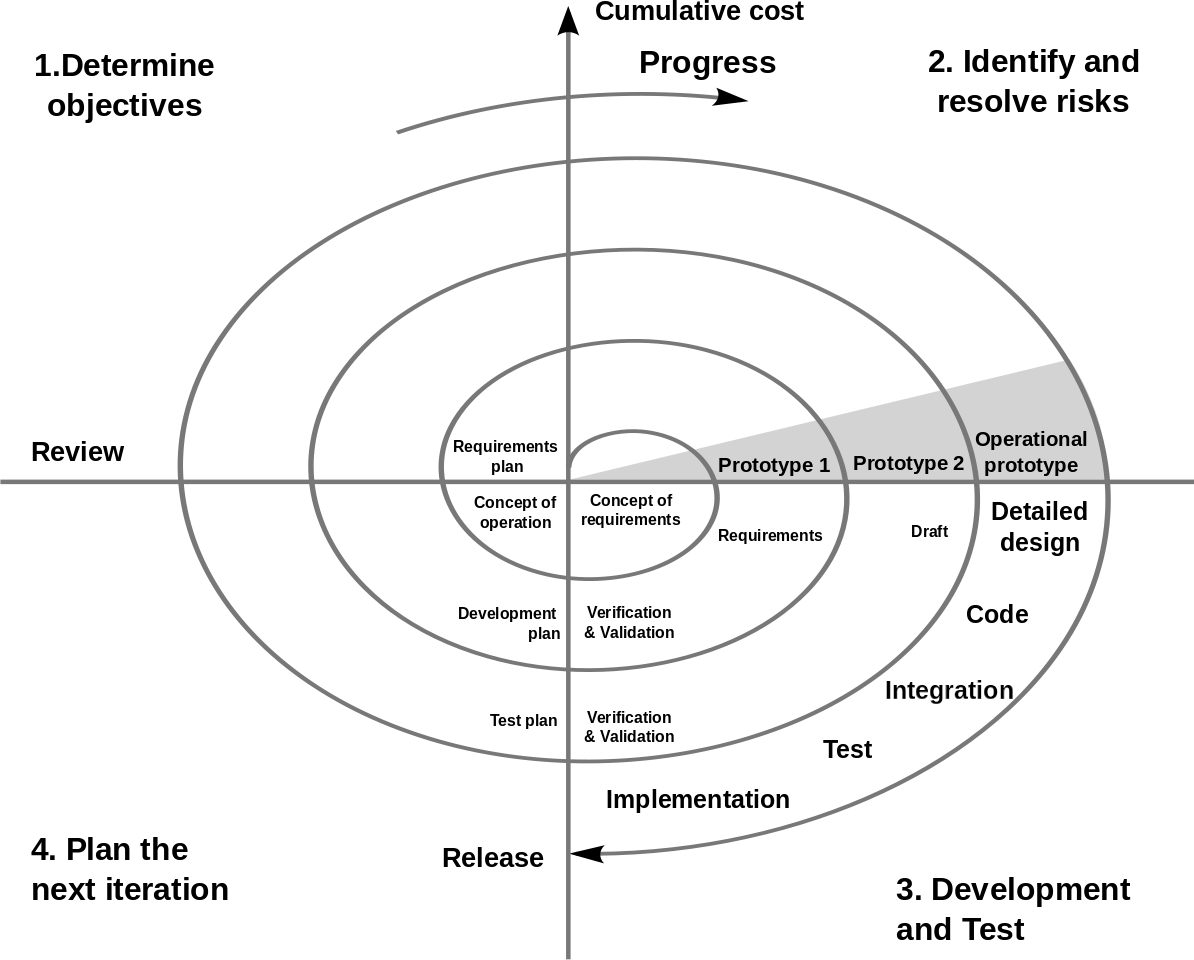
The Construct phase refers to production of the actual software product at every spiral. In the baseline spiral, when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback.

Then in the subsequent spirals with higher clarity on requirements and design details a working model of the software called build is produced with a version number. These builds are sent to the customer for feedback.

**Evaluation and Risk Analysis:**

Risk Analysis includes identifying, estimating and monitoring the technical feasibility and management risks, such as schedule slippage and cost overrun. After testing the build, at the end of first iteration, the customer evaluates the software and provides feedback.

The following illustration is a representation of the Spiral Model, listing the activities in each phase.



Based on the customer evaluation, the software development process enters the next iteration and subsequently follows the linear approach to implement the feedback suggested by the customer. The process of iterations along the spiral continues throughout the life of the software.

**Spiral Model Application:**

The Spiral Model is widely used in the software industry as it is in sync with the natural development process of any product, i.e., learning with maturity which involves minimum risk for the customer as well as the development firms.

**The following pointers explain the typical uses of a Spiral Model –**

* When there is a budget constraint and risk evaluation is important.
* For medium to high-risk projects.
* Long-term project commitment because of potential changes to economic

priorities as the requirements change with time.

* Customer is not sure of their requirements which is usually the case.
* Requirements are complex and need evaluation to get clarity.
* New product line which should be released in phases to get enough customer

feedback.

* Significant changes are expected in the product during the development cycle.

**Spiral Model - Pros and Cons:**

The advantage of spiral life-cycle model is that it allows elements of the product to be added in, when they become available or known. This assures that there is no conflict with previous requirements and design. This method is consistent with approaches that have multiple software builds and releases which allows making an orderly transition to a maintenance activity. Another positive aspect of this method is that the spiral model forces an early user involvement in the system development effort.

On the other side, it takes a very strict management to complete such products and there is a risk of running the spiral in an indefinite loop. So, the discipline of change and the extent of taking change requests is very important to develop and deploy the product successfully.

**Chapter 3: System Architecture**

**3.1 System Architecture**

System Start

Data Set

Preprocessed Data

Training Data

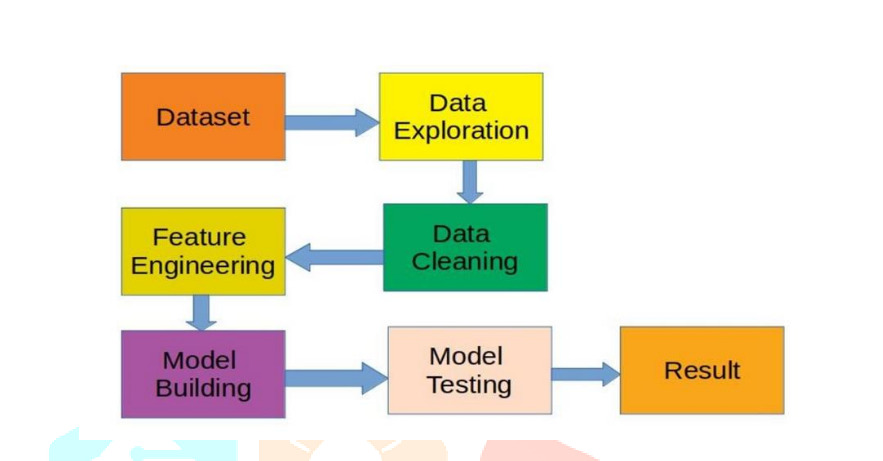
Model building

Testing and choosing best

System End

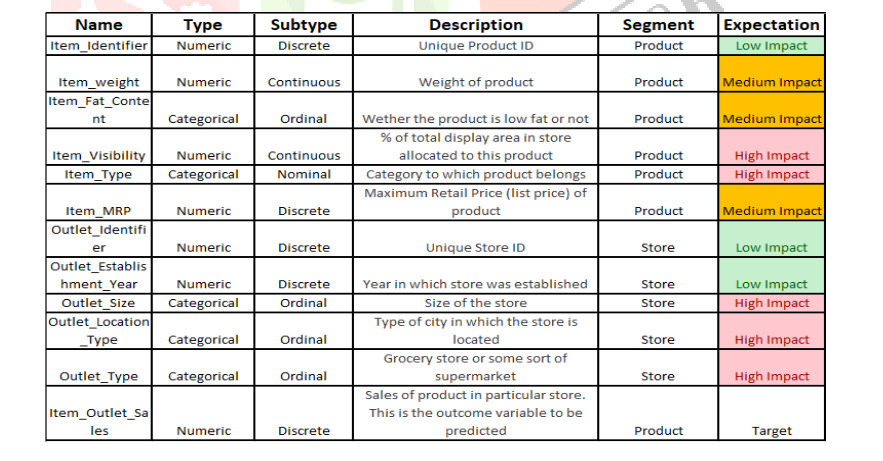
Scikit-learn

**3.2 Proposed Model**

****

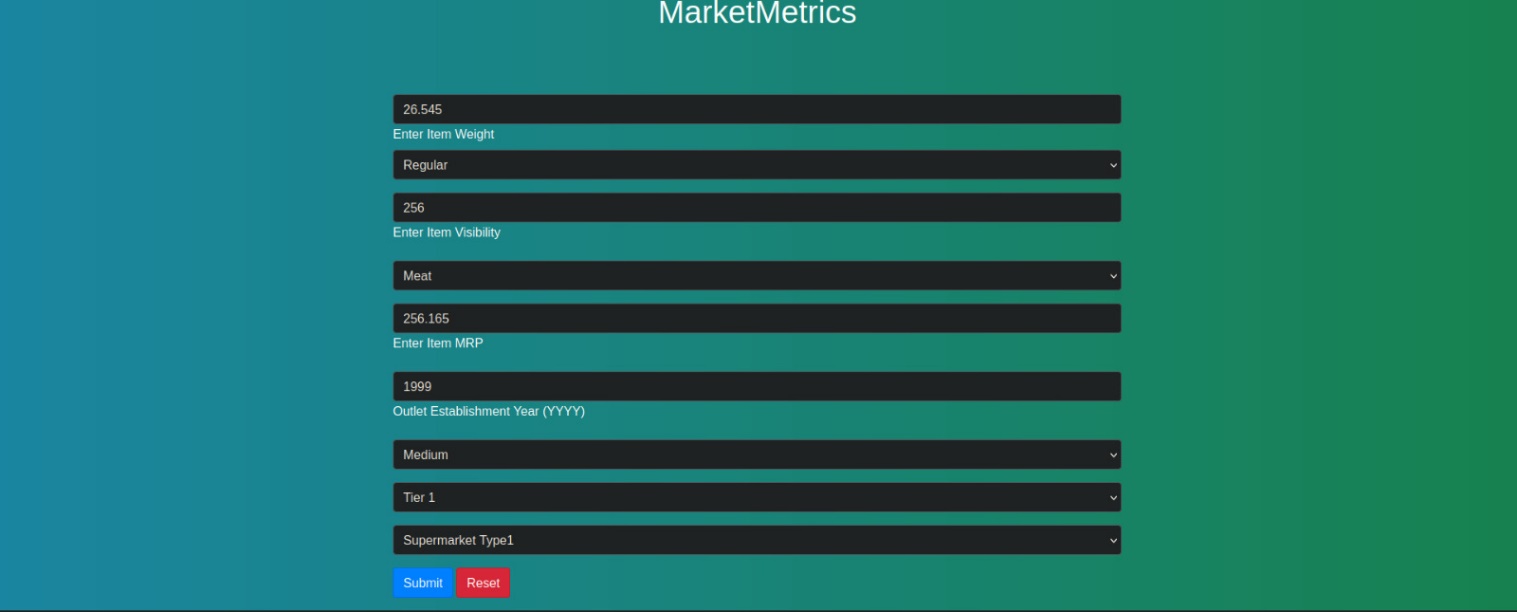
**3.3 Phases in Model**

**Data Set:**

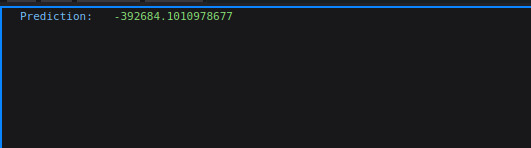
****

**3.4 Input & output design**

**Home Page:**



**Output :**



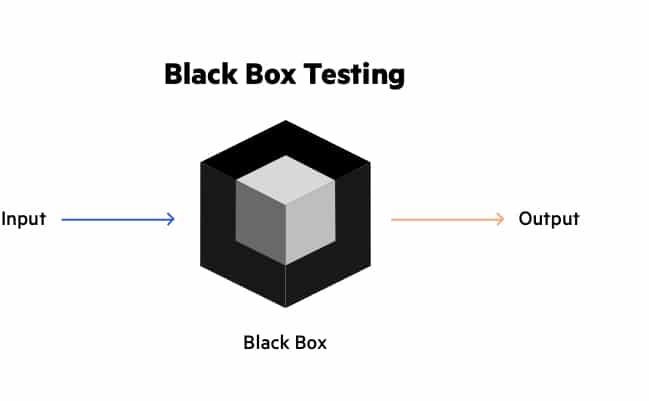
**Chapter 4: Testing & Implementation**

**4.1 Testing approach used**

***Definition:***

***Black box testing:*** Testing, either functional or non-functional, without reference to the internal structure of the component or system.

Black Box, also known as Behavioral Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional.

**[Black Box Testing]**

This method is named so because the software program, in the eyes of the tester, is like a black box, inside which one cannot see.

**This method attempts to find errors in the following categories:**

* ****Incorrect or missing functions
* Interface errors
* Errors in data structures or external database access
* Behavior or performance errors.
* Initialization and termination errors.

**White Box Testing**

White Box Testing (also known as Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing or Structural Testing) is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs. Programming know-how and the implementation knowledge is essential. White box testing is testing beyond the user interface and into the nitty-gritty of a system.

This method is named so because the software program, in the eyes of the tester, is like a white/transparent box; inside which one clearly sees.

**Advantages**

* Testing can be commenced at an earlier stage. One need not wait for the GUI

to be available.

* Testing is more thorough, with the possibility of covering most paths.

**Disadvantages**

* Since tests can be very complex, highly skilled resources are required, with a

thorough knowledge of programming and implementation.

* Test script maintenance can be a burden if the implementation changes too

frequently.

* Since this method of testing is closely tied to the application being tested,

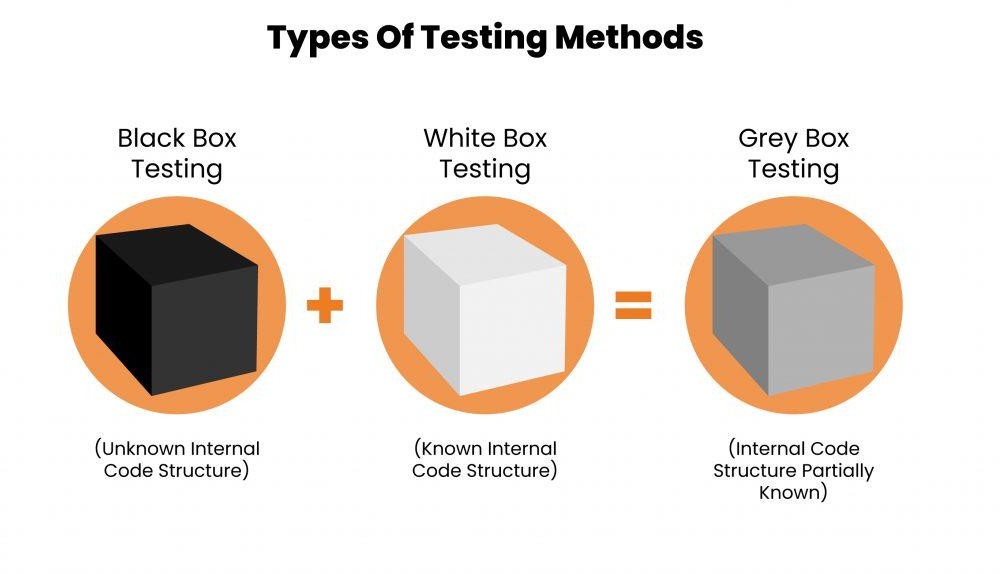
tools to cater to every kind of implementation/platform may not be readily

available.

**Grey Box Testing**

Grey Box testing is testing technique performed with limited information about the internal functionality of the system. Grey Box testers have access to the detailed design documents along with information about requirements.

Grey Box tests are generated based on the state-based models, UML Diagrams or architecture diagrams of the target system.



**[Grey Box Testing]**

**Grey-box testing Techniques:**

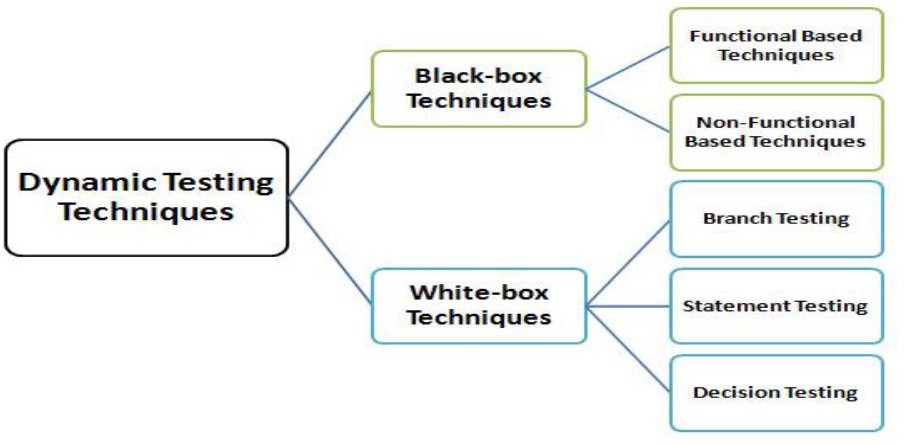
* Regression testing.
* Pattern Testing.
* Orthogonal array testing.
* Matrix testing.

**Benefits:**

* Grey-box testing provides combined benefits of both white-box and black-box testing.
* It is based on functional specification, UML Diagrams, Database Diagrams or architectural view.
* Grey-box tester handles can design complex test scenario more intelligently
* The added advantage of grey-box testing is that it maintains the boundary between independent testers and developers.

**Drawbacks:**

* In grey-box testing, complete white box testing cannot be done due to inaccessible source code/binaries.
* It is difficult to associate defects when we perform Grey-box testing for a distributed system.



**Chapter 5: Conclusion**

**5.1 Conclusion**

In conclusion, the MarketMetrics web application is a comprehensive and efficient solution that streamlines various forecasting and planning processes within a business. By leveraging the power of technology, the application offers a range of features and functionalities to enhance the overall MarketMetrics and decision-making processes.

The web application enables seamless sales trend analysis and forecasting, allowing users to make data-driven predictions conveniently while ensuring optimal utilization of resources and time. It also provides an intuitive interface for managing sales data, ensuring accurate and accessible insights at all times. This improves the efficiency of business strategies and operations, ultimately leading to better decision-making and profitability.

Furthermore, the application enhances communication and collaboration among teams by providing secure messaging and data-sharing features. This promotes efficient information exchange, leading to better coordination and strategic alignment.

In summary, the MarketMetrics web application revolutionizes the way businesses manage sales forecasting, offering a centralized platform for managing various aspects of business operations. With its user-friendly interface and robust features, it optimizes resource utilization, improves decision-making, streamlines workflows, and ultimately contributes to the overall success and effectiveness of the organization.

**5.2 Limitations**

* Online.
* You must have pc/laptop.
* Limited area covered.

**5.3 Future Scope of System**

**Future Scope for MarketMetrics System:**

The future scope for MarketMetrics is very promising. The business and retail industry is growing rapidly, and there is a high demand for accurate sales forecasting and data-driven insights. Some of the factors driving this growth include:

* Market Expansion: As businesses expand into new markets, accurate MarketMetricss are crucial to understanding demand and optimizing resources.
* The increasing reliance on data-driven decision-making: Companies are placing a greater emphasis on using data to drive decisions, which makes MarketMetrics systems even more vital.
* Advancements in technology: As artificial intelligence and machine learning continue to improve, they will enhance the accuracy of MarketMetricss and forecasting models.

**Future Scope for our Website:**

Nothing is perfect in this world, and our system is no exception. Although we have tried our best to present accurate and insightful MarketMetricss, there is always room for further enhancement in the application.

We have taken care of all the critical aspects necessary for developing this project. However, as with any system, there are certain limitations that prevent the system from being 100% accurate. These limitations can be addressed in future updates or improvements, allowing the system to provide even more precise and reliable predictions.

**5.4 Reference**

**. Beaumont, J. R., & Beaumont, C. D. (1988). Applied management information systems — Some personal thoughts on current practice and future potential: I: Strategic positioning. *Futures*, 20(1), 69–77.** [**https://doi.org/10.1016/0016-3287(88)90044-4**](https://doi.org/10.1016/0016-3287(88)90044-4)

**. Harris, M. (1991). Strategic planning for information systems. *Journal of Information Technology*, 6(1), 60.** [**https://doi.org/10.1057/jit.1991.9**](https://doi.org/10.1057/jit.1991.9)

**. Rosenberger, H. R., & Kaiser, K. M. (1985). Strategic planning for health care management information systems. *Health Care Management Review*, 10(1), 7–18.** [**https://doi.org/10.1097/00004010-198501010-00002**](https://doi.org/10.1097/00004010-198501010-00002)

**. King, W. R. (1978). Strategic planning for management information systems. *MIS Quarterly*, 2(1), 27–37.** [**https://doi.org/10.2307/248900**](https://doi.org/10.2307/248900)

**.** **Henderson, J. C., Rockart, J. F., & Sifonis, J. G. (1987). Integrating management support systems into strategic information systems planning. *Journal of Management Information Systems*, 4(1), 5–24.** [**https://doi.org/10.1080/07421222.1987.11517783**](https://doi.org/10.1080/07421222.1987.11517783)