



# Customer Relationship Management System

<sup>1</sup>Arpitha V, <sup>2</sup>Dr. Shashidhar Kini K

<sup>1</sup>Student, <sup>2</sup>Professor & Head

<sup>1</sup>Department of Computer Application,

<sup>1</sup>Srinivas Institute of Technology, Valachil Mangaluru, Karnataka, India

**Abstract:** A Customer Relationship Management (CRM) system using the MERN stack (MongoDB, Express, React, Node.js) is a platform designed to manage customer information, interactions, and activities efficiently. It consists of a React-based frontend for user interaction, an Express and Node.js backend for business logic and API management, and a MongoDB database for storing customer data and interaction history. The system enables functionalities such as adding, editing, deleting, and viewing customer records, managing interactions, and searching or filtering customer information. User requests are handled by the frontend, processed by the backend, and stored or retrieved from the database to ensure seamless data management. This architecture provides a scalable and user-friendly solution for centralized customer relationship management.

*Index Terms – CRMS, MERN Stack, Responsive UI, MongoDB, Express.js, React.js, Node.js*

## I. INTRODUCTION

A Customer Relationship Management (CRM) system is a technology solution designed to help businesses manage and analyze customer interactions throughout the customer lifecycle. The primary goal of a CRM system is to improve business relationships with customers, enhance customer satisfaction, and drive sales growth. It provides a centralized platform where businesses can store customer data, track communications, manage leads, and automate various sales and marketing processes. By organizing and integrating customer information, a CRM system enables companies to respond more effectively to customer needs, personalize interactions, and build long-term loyalty. As a vital tool in today's competitive market, CRM systems not only streamline internal operations but also offer valuable insights that support strategic decision-making and improved service delivery.

A Customer Relationship Management (CRM) system is a comprehensive software solution designed to manage a company's interactions with current and potential customers. It serves as a central hub for storing customer information, tracking communication history, managing leads and opportunities, and automating various business processes related to sales, marketing, and customer service. By organizing all customer-related data in one place, a CRM system enables businesses to better understand customer behavior, improve customer satisfaction, and build stronger, more personalized relationships.

### Benefits of the System:

- Real-Time and Dynamic User Interface
- Flexible and Scalable Database
- Modular and Maintainable Architecture
- Fast Development and Strong Community Support
- Cost-Effective Solution
- Scalable and Efficient Backend Node.js and Express.js

### EASE OF USE

A well-designed Customer Relationship Management (CRM) system prioritizes user-friendliness that ensure employees across different departments can quickly adopt and effectively use the system. Ease of use is achieved through a clean, intuitive interface that simplifies tasks such as adding new customers, updating information, tracking interactions, and managing sales pipelines.

### Abbreviations and Acronyms

Define each abbreviation or acronym the first time it appears in the text, even if it was already defined in the abstract.

- CRUD – Create, Read, Update, Delete
- UI – User Interface
- B2B – Business to Business
- B2C – Business to Customer
- MERN – MongoDB, Express.js, React.js, Node.js
- ODM – Object Data Modelling
- CSS – Cascading Style Sheets
- API – Application Programming Interface
- KPI – Key Performance

## II. RESEARCH METHODOLOGY

The methodology section outlines the approach and methods used to conduct the study. This includes the universe of the study, sample of the study, data and sources of data, the system's variables, and analytical framework. The details are as follows:

### 2.1 Requirement analysis

Requirement analysis is a crucial phase in the development of a Customer Relationship Management (CRM) system, as it helps identify and understand the needs and expectations of the end-users and stakeholders. In this phase, both functional and non-functional requirements are gathered through various methods such as surveys, interviews, questionnaires, and direct observations of business processes.

### 2.2 System design

The system design phase is a critical step in the research methodology of a Customer Relationship Management (CRM) system, as it translates the gathered requirements into a structured blueprint for development. This phase involves creating architectural diagrams, data flow models, and wireframes that define how different components of the system will interact with each other. The design outlines the user interface (UI) layout, backend architecture, database schema, and API structure, ensuring a seamless flow of data between modules such as customer profiles, lead management, communications, and analytics.

### 2.3 Testing and Validation

The Testing and validation are essential stages in the development of a Customer Relationship Management (CRM) system, ensuring that the software functions as intended and meets user requirements. This phase involves multiple levels of testing, including unit testing to verify individual components like customer data input forms or lead tracking modules, and integration testing to ensure different parts of the system work together seamlessly—such as the interaction between the frontend (React.js) and backend APIs (Node.js/Express.js). User Acceptance Testing (UAT) is also conducted to gather feedback from actual users, helping confirm that the system is intuitive, efficient, and aligned with business workflows.

### Independent Variables:

- **Customization Capability:**

The Customization capability in a (CRM) system refers to the system's flexibility to adapt to specific business needs, workflows, and user preferences. A customizable CRM allows users to modify data fields, create custom reports, set personalized dashboards, automate unique workflows, and integrate third-party tools relevant to their operations.

- **System Functionality:**

This System functionality in a Customer Relationship Management (CRM) system refers to the range and effectiveness of features that support business processes related to managing customer relationships. Core functionalities typically include contact and lead management, sales tracking, task automation, communication history logging, and reporting tools.

- **System Usability:**

The System functionality in a Customer Relationship Management (CRM) system refers to the range and effectiveness of features that support business processes related to managing customer relationships. A functional CRM system ensures that users can efficiently collect, organize, and analyze customer data to improve service delivery and business outcomes. functionality directly contributes to improved customer engagement, streamlined workflows, and better decision-making.

- **Data Integration:**

Data integration in a Customer Relationship Management (CRM) system is the process of connecting various data sources and applications to ensure that all customer-related information is unified and accessible in one platform. This allows businesses to gather data from emails, websites, social media, sales platforms, and support systems, creating a

comprehensive view of each customer.

### Dependent Variable:

In Dependent variables in a Customer Relationship Management (CRM) system refer to the outcomes or results that are influenced by factors such as system usability, functionality, customization, and integration. These variables reflect the effectiveness and impact of the CRM system on business operations. Common dependent variables include user satisfaction, employee productivity, customer retention, sales performance, and data accuracy.

For example, if the system is easy to use and well-integrated with other tools, it can lead to higher user satisfaction and better customer service. Similarly, effective CRM features can improve employee efficiency and help increase sales by tracking leads and managing customer interactions more effectively. These dependent variables are essential in evaluating the success and value of a CRM system within an organization.

### 2.4 Statistical tools and econometric models

This section elaborates on the proper statistical/econometric/technical models used to advance the study from data collection to inference. The detailed methodology is provided as follows:

#### 2.4.1 Statistical Tools

- **JavaScript:**  
The primary programming language used for both frontend and backend development, enabling seamless integration across the MERN stack and supporting the dynamic nature of the platform.
- **MongoDB:**  
A NoSQL, document-oriented database used to store user profiles, travel journal entries, media (photos/videos), and location data. MongoDB is ideal for the flexible and scalable structure required to handle the diverse nature of travel data.
- **Express.js:**  
A minimal and flexible Node.js web application framework that builds APIs, manages server-side routing, and handles middleware for authentication, data handling, and user role validation (admin, traveller, content creator).
- **React.js:**  
A powerful frontend library used to build dynamic and responsive user interfaces, enabling smooth user interactions and real-time state management across the platform. React.js supports component-based development to create reusable and efficient UI elements.
- **Node.js:**  
A runtime environment that executes JavaScript on the server side. Node.js facilitates asynchronous operations, making it ideal for handling concurrent user requests efficiently and maintaining performance during high traffic.
- **Mongoose:**  
An Object Data Modelling (ODM) library used to structure and validate MongoDB data, simplifying interactions between the application and the database. Mongoose helps ensure data consistency, especially for user-generated content and media uploads.
- **NPM (Node Package Manager):**  
A package manager used to manage and install third-party libraries and tools needed throughout the project, ensuring smooth integration and version control for dependencies.

#### 2.4.2 Econometric Models

##### 2.4.2.1 Linear Regression Model:

In A linear regression model can analyze the relationship between system features (usability, functionality, etc.) and customer satisfaction or employee productivity. With a CRM developed using MERN, data can be collected via the frontend in React.js (user interactions) and stored in MongoDB. The backend Node.js and Express.js can serve as the API layer to process the data and run the regression analysis.

The linear regression model is a fundamental econometric tool used to quantify the relationship between one or more independent variables and a dependent variable. In the context of a Customer Relationship Management (CRM) system, linear regression can be employed to understand how different system features—such as system usability, training quality, or CRM functionality—influence key outcomes like customer satisfaction, employee productivity, or sales performance.

##### 2.4.2.2 Time –Series Analysis:

Time series analysis is a statistical technique used to analyze data points collected or recorded at specific time intervals to identify patterns, trends, and forecasts over time. In the context of a Customer Relationship Management (CRM) system, time series analysis can be applied to study metrics such as monthly sales, customer interactions, lead conversions, or customer retention rates. By examining historical data stored in the system, businesses can detect seasonal trends, recurring patterns, or growth trajectories that help in strategic planning and forecasting future performance. For example, a CRM built with the MERN stack can collect and store chronological sales and interaction data in MongoDB, while the backend (Node.js/Express.js) processes it for time-series modeling.

### 2.4.2.3 Structural Equation Modeling (SEM):

The Structural Equation Modeling (SEM) is an advanced statistical technique that allows researchers and developers to analyze complex relationships between observed and latent variables within a system. In a Customer Relationship Management (CRM) system, SEM can be used to explore how various components—such as system usability, training effectiveness, and feature utilization—indirectly and directly affect outcomes like employee productivity, customer satisfaction, and sales growth. Unlike simpler regression models, SEM can test multiple dependent relationships simultaneously and account for hidden factors that influence user behavior and performance.

### 2.4.3 Model Evaluation Metrics

The performance of the Customer Relationship Management System platform is evaluated using several statistical metrics to ensure the application runs efficiently and delivers a positive user experience. Key metrics include:

- **Mean Absolute Error(MAE):**  
Measures the average magnitude of errors between predicted and actual values. It is simple and interpretable—lower values indicate better accuracy.
- **Area Under the ROC Curve(AUC-ROC):**  
The Area Under the Receiver Operating Characteristic (ROC) Curve (AUC-ROC) is a powerful evaluation metric used to assess the performance of classification models, particularly in binary classification tasks such as predicting customer churn or lead conversion in a Customer Relationship Management (CRM) system.
- **Accuracy:**  
Reflects how accurately the application processes and displays data, such as the relevance of travel recommendations, search results, or content categorization. Higher accuracy ensures that users find useful and relevant content based on their travel interests.
- **Error Rate:**  
Measures the percentage of failed or incorrect actions, such as unsuccessful media uploads or broken links in journal entries. A lower error rate indicates better system reliability and fewer issues for users.
- **Precision,Recall,and F1 Score:**  
Tracks These are particularly useful in imbalanced classification problems (e.g., predicting rare customer churn events).

## III. RESULTS AND DISCUSSION

**Table 4.1: Descriptive Statistics**

Variable	Mean
System Usability Score	4.2
Customer Satisfaction	4.0
Employee Efficiency	75.6%
CRM Feature Usage(avg)	6.8
Training Hours Completed	12.3

**Table 4.1:** These metrics provide a snapshot of the platform's usage, performance, and engagement. They demonstrate the active participation of users and the efficiency of your system in handling media uploads and API responses. You can further analyze these metrics to understand user behavior, optimize performance, and make improvements to the platform.

## IV. ACKNOWLEDGMENT

The author wishes to express sincere gratitude to the Project Guide and Head of the Department of MCA, Dr. Shashidhar Kini K, for his invaluable guidance, constant encouragement, and kind support throughout this research work. Appreciation is also extended to the Principal, Dr. Shrinivasa Mayya D, for fostering an environment conducive to completing this project within the institution. The author thanks the management of Srinivas Institute of Technology for their direct and indirect support. Gratitude is also due to all the faculty members and non-teaching staff of the MCA department for their constant help and support. Finally, the author is indebted to parents and friends for their unwavering support and belief throughout this endeavor.

## REFERENCES

- [1] Manning, C. (2023). Full-Stack React, TypeScript, and Node: Build Modern Web Applications with React, Node.js, and TypeScript. Manning Publications.
- [2] Koller, M. (2023). Building Full-Stack Applications with the MERN Stack: A Guide for Developers. O'Reilly Media.
- [3] Haverbeke, M. (2022). Eloquent JavaScript: A Modern Introduction to Programming (3rd ed.). No Starch Press.
- [4] Kumar, A. (2023). Learning MongoDB: A Hands-On Guide to Building NoSQL Databases. O'Reilly Media.
- [5] Schultz, D., & Paul, A. (2022). Express.js Guide: The Comprehensive Guide to Building Web Applications with Node.js and Express. Apress.

