**SYNOPSIS**

**Report on**

**Customer Relationship Manager**

**Submitted by: -**

**Arnav singh 2100290140036**

**Himanshu saini 2100290140071**

**Harish 2100290140066**

**Session:2022-2023 (4 th Semester)**

**Under the supervision of**

**Mr. Amit Goyal**

****

**Department of Computer Applications,**

**KIET Group of Institutions,**

**Delhi-NCR, Ghaziabad**

**March 2023**

**ABSTRACT**

Customer relationship management (CRM) is a combination of people, processes and technology that seeks to understand a company's customers. It is an integrated approach to managing relationships by focusing on customer retention and relationship development. CRM has evolved from advances in information technology and organizational changes in customer‐centric processes. Companies that successfully implement CRM will reap the rewards in customer loyalty and long run profitability. However, successful implementation is elusive to many companies, mostly because they do not understand that CRM requires company‐wide, cross‐functional, customer‐focused business process re‐engineering. Although a large portion of CRM is technology, viewing CRM as a technology‐only solution is likely to fail. Managing a successful CRM implementation requires an integrated and balanced approach to technology, process, and people.

**TABLE OF CONTENTS**

[INTRODUCTION 1](#_Toc132447729)

[TECHNOLOGIES USED 2](#_Toc132447730)

[HARDWARE AND SOFTWARE REQUIREMENTS 5](#_Toc132447731)

[MODULES 6](#_Toc132447732)

[FUTURE SCOPE 8](#_Toc132447733)

[FUNCTIONING OF THE PROJECT 10](#_Toc132447734)

[GANTT CHART 11](#_Toc132447735)

[CONCLUSION 12](#_Toc132447736)

# INTRODUCTION

Customer Relationship Management (CRM) is a well-known system for managing an organization’s communications with customers, clients, and sales prospects. This system uses technology for organizing, synchronizing, and automating business and sales activities. This is a virtual system, and this system does not require any physical offices or environment. So everybody can use this system easily. Administrators can communicate and provide services through this system, and also users can access and check their status and project opportunities as well as get more information about the projects and update systems.

# TECHNOLOGIES USED

**JAVA**

Java can be used for developing CRM (Customer Relationship Management) systems. Java is a versatile programming language that is widely used for enterprise-level applications due to its scalability, platform independence, and robustness. Several Java frameworks and tools can be used to develop CRM systems, such as Spring, Hibernate, Struts, and Java Server Faces (JSF). These frameworks provide a standardized approach to developing enterprise applications and offer database connectivity, security, and application lifecycle management features. Some of the key features that a CRM system developed using Java can offer to include:

**Scalability:** Java-based CRM systems can handle large amounts of data and users, making them suitable for businesses of all sizes.

**Platform independence:** Java is a platform-independent language, which means that the CRM system developed using Java can run on different platforms without requiring any modifications.

**Security:** Java has built-in security features, such as a sandbox environment and automatic memory management, which can help ensure the security and stability of the CRM system.

**Integration with other systems:** Java-based CRM systems can easily integrate with other enterprise systems, such as ERP and HR systems, to provide a complete view of the customer.

Overall, Java is a suitable programming language for developing CRM systems due to its scalability, platform independence, and robustness. By leveraging Java frameworks and tools, developers can create feature-rich CRM systems that can help businesses manage customer relationships more effectively.

**MySQL**

MySQL is a popular relational database management system that can be used for CRM (Customer Relationship Management) projects. When developing a CRM system using MySQL, developers can leverage several tools and frameworks, such as JDBC (Java Database Connectivity), Spring Data, and Hibernate, to simplify database connectivity and management. Overall, MySQL is a suitable database system for developing CRM systems due to its scalability, performance, security, and open-source nature. By leveraging the various features and tools available in MySQL, developers can create robust and scalable CRM systems that can help businesses manage customer relationships more effectively.

**HTML**

HTML (Hypertext Markup Language) is a standard markup language used for creating web pages and applications. While HTML is not typically used for the backend of a CRM (Customer Relationship Management) system, it's an essential component of the front-end user interface.

**CSS**

CSS (Cascading Style Sheets) is a style sheet language used to describe the presentation of web pages written in HTML. CSS is an essential component of the front-end user interface of a CRM (Customer Relationship Management) system as it's used to define the visual style, layout, and presentation of the user interface.

**Spring Framework**

The Spring Framework is a popular Java-based application framework that can be used for developing various enterprise-level applications, including CRM (Customer Relationship Management) systems. The Spring Framework provides several features and modules that make it an ideal choice for developing scalable, robust, and maintainable CRM systems.

**XML**

XML is a flexible and versatile format that can be easily processed by different programming languages and systems. When using XML in a CRM system, developers can leverage various tools and frameworks, such as JAXB (Java Architecture for XML Binding) and XSLT (Extensible Stylesheet Language Transformations), to simplify data exchange, configuration, reporting, and customization. Overall, XML is a suitable choice for developing CRM systems due to its flexibility, versatility, and ease of processing. By leveraging the various features and tools available in XML, developers can create CRM systems that meet specific business requirements and seamlessly integrate with other enterprise systems.

# HARDWARE AND SOFTWARE REQUIREMENTS

* Processor i5 and above
* 4 GB Ram and above
* Windows 8 and above

# MODULES

**Lead Management**

This module is used to manage leads, including capturing, tracking, and nurturing them through the sales funnel.

**Contact Management**

This module is used to manage customer contact information, including name, address, phone number, and email.

**Sales Management**

This module is used to manage the sales process, including creating and managing sales opportunities, quotes, and orders.

**Marketing Management**

This module is used to manage marketing campaigns, including email marketing, social media marketing, and advertising.

**Customer Service Management**

This module is used to manage customer service requests, including support tickets, inquiries, and complaints.

**Analytics and Reporting**

This module is used to generate reports and analytics, including sales reports, customer behavior, and campaign effectiveness.

**Integration with Third-party systems**

This module is used to integrate with other enterprise systems such as ERP, HR systems, and Marketing automation systems to provide a complete view of the customer.

**User Management**

This module is used to manage users and their roles, permissions, and access rights.

**Dashboard and Notifications**

This module is used to provide a visual summary of the system's performance, including key performance indicators, alerts, and notifications.

# FUTURE SCOPE

The future scope for CRM (Customer Relationship Management) projects developed with Java is promising, driven by the continuous improvements and advancements in Java technology and its ecosystem. Here are some potential future trends and advancements that could shape the future of CRM projects developed with Java:

**Microservices Architecture**

The microservices architecture is gaining popularity as a way to develop scalable and modular CRM systems. Java-based microservices frameworks like Spring Boot, Micronaut, and Quarkus are expected to become more prevalent in CRM projects.

**Cloud Computing**

With the increasing adoption of cloud computing, Java-based CRM projects are expected to leverage cloud platforms like AWS, Azure, and Google Cloud to develop and deploy CRM systems on the cloud.

**Reactive Programming**

Reactive programming is gaining popularity as a way to develop highly responsive and scalable CRM systems. Java-based reactive frameworks like Akka, Reactor, and RxJava are expected to become more prevalent in CRM projects.

**Integration with AI and ML**

Java-based CRM projects are expected to leverage AI and ML technologies to provide personalized recommendations, predictive analytics, and automated workflows.

**Security**

Security and privacy are critical considerations for CRM systems. Java-based CRM projects are expected to incorporate more advanced security features, such as OAuth2, JWT, and SSL, to ensure the security and privacy of customer data.

Overall, the future scope for CRM projects developed with Java is promising, driven by the continuous advancements in Java technology and its ecosystem. By leveraging the various Java-based frameworks, tools, and platforms, developers can create scalable, robust, and maintainable CRM systems that meet specific business requirements.

# FUNCTIONING OF THE PROJECT

The functioning of a CRM (Customer Relationship Management) project can vary depending on its specific requirements and features. However, here are some common functionalities that are typically included in a CRM project:

**Customer data management**: The CRM project allows businesses to store, manage, and update customer data in a centralized database. This includes customer contact information, purchase history, and other relevant data.

**Sales management**: The CRM project allows businesses to manage the sales process by tracking leads, deals, and pipelines. This includes features like lead scoring, deal tracking, and forecasting.

**Marketing automation**: The CRM project allows businesses to automate marketing tasks, such as email campaigns, social media marketing, and lead nurturing. This includes features like email templates, marketing workflows, and analytics.

**Customer support**: The CRM project allows businesses to manage customer support inquiries and tickets. This includes features like ticket routing, automated responses, and reporting.

**Reporting and analytics**: The CRM project allow businesses to generate custom reports and analytics based on customer data. This includes features like dashboard visualization, data segmentation, and custom reporting.

**Integration with other systems**: The CRM project allows businesses to integrate with other enterprise systems, such as ERP (Enterprise Resource Planning) systems, accounting systems, and marketing automation systems. This includes features like APIs, webhooks, and data synchronization.

Overall, the functioning of a CRM project is to provide businesses with a comprehensive platform for managing customer relationships, sales, marketing, and support.

# GANTT CHART

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| – | **WEEK**  **1** | **WEEK 2** | **WEEK 3** | **WEEK 4** | **WEEK 5** | **WEEK 6** | **WEEK 7** | **WEEK 8** | **WEEK 9** | **WEEK 10** | **WEEK 11** | **WEEK 12** |
| Requirement analysis and feasibility check |  |  |  |  |  |  |  |  |  |  |  |  |
| Designing |  |  |  |  |  |  |  |  |  |  |  |  |
| Coding |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing and maintenance |  |  |  |  |  |  |  |  |  |  |  |  |

# CONCLUSION

In conclusion, CRM (Customer Relationship Management) systems and projects have become an essential tools for businesses to manage customer relationships, sales, marketing, and support. The continuous advancements in technology and its ecosystem, particularly in Java and related technologies, offer promising future scope for developing scalable, robust, and maintainable CRM systems. The functionalities of a CRM project can vary depending on specific business requirements, but common features include customer data management, sales management, marketing automation, customer support, reporting, analytics, and integration with other systems. By utilizing a CRM project, businesses can centralize customer data, automate workflows, and provide insights and analytics to improve customer engagement, increase sales, and streamline business processes.