Software Requirement Specification

For

Customer Relationship Management

Version 1.0 approved

Prepared by: Anamika Patel

HCL Training

Date: 15/01/2026

Submitted in partial fulfillment

of the requirements of

Software Engineering (Agile Methodology)

TABLE OF CONTENTS

List Of Figures

Figure No.	Figure Title	Page No.
Fig. 4.2	Entity Relationship Diagram	12
Fig. 4.3	Data Flow Diagram (DFD)	12
Fig. 4.4	Use Case Diagram	13

LIST OF TABLES

1. **Abstract**

A **Customer Relationship Management (CRM) system** is a powerful software solution designed to help organizations manage customer information, interactions, and business processes in an efficient and systematic manner. In today's highly competitive business environment, building and maintaining strong relationships with customers is essential for achieving organizational growth, improving customer retention, and ensuring long-term business success. This project focuses on the design and development of a **CRM system using Agile methodology**, which emphasizes flexibility, iterative development, and continuous customer feedback.

The **Agile development approach** allows the project team to divide the entire system into small and manageable **sprints**, where each sprint delivers a functional and usable component of the system. This iterative model ensures **faster delivery**, **better quality control**, and **reduced project risk**. Agile practices such as **sprint planning, daily stand-up meetings, sprint reviews, and sprint retrospectives** are followed throughout the development lifecycle to ensure transparency, collaboration, and continuous improvement.

The proposed CRM system provides essential features such as **customer data management**, **lead tracking**, **sales pipeline management**, **communication history tracking**, and **report generation**. These features help organizations gain valuable insights into customer behavior, monitor sales performance, and improve decision-making processes. Traditional manual handling of customer records often leads to problems such as **data inconsistency, data duplication, security issues, and operational inefficiency**. The proposed system overcomes these challenges by **automating data storage and retrieval**, ensuring high levels of **accuracy, reliability, and data integrity**.

The CRM system is designed with a **user-friendly interface**, making it easy for employees to access customer information and perform daily operations efficiently. The system also incorporates essential **security mechanisms** such as user authentication and role-based access control to protect sensitive customer data. The **modular architecture** of the system enables easy scalability and future enhancements, allowing organizations to extend functionality as business requirements evolve.

Continuous feedback from stakeholders plays a crucial role in refining system features and ensuring alignment with real-world business needs. By adopting Agile methodology, the project ensures better collaboration between developers and users, resulting in a more reliable and effective product.

Overall, this project demonstrates how the combination of a **robust CRM system** and **Agile software development methodology** can significantly improve operational efficiency, enhance customer satisfaction, and strengthen customer relationships, making it an ideal solution for small and medium-sized organizations.

2. Introduction

2.1 Introduction

In today's fast-paced digital world, Customer Relationship Management (CRM) has become an essential software solution for businesses of all sizes. With customers expecting quick responses, personalized services, and seamless communication, managing customer interactions manually is no longer practical. A CRM system helps organizations store, manage, and analyze customer data in a centralized platform, making it easier to understand customer behavior and preferences. It enables businesses to build stronger relationships by offering timely support, targeted marketing, and customized services. In the era of e-commerce, digital payments, and online customer engagement, CRM systems play a crucial role in improving communication, enhancing service quality, and driving long-term business growth.

2.2. Problem Identification

Many organizations still rely on traditional manual or semi-digital methods to maintain customer records and manage interactions. These methods are time-consuming and require significant human effort for data entry, updating, and retrieval. As customer data continues to grow, managing records manually becomes inefficient and prone to errors. Employees often face difficulties in searching for specific customer information, which delays service delivery and affects overall productivity. The lack of automation also makes it challenging to track follow-ups, complaints, and sales activities in a structured manner.

Another major issue with manual record keeping is the frequent occurrence of data duplication, inconsistency, and data loss. Multiple entries for the same customer lead to inaccurate records and confusion among departments. Since customer information is stored in separate files or registers, tracking complete customer history becomes difficult. This results in poor customer service, missed business opportunities, and reduced customer satisfaction. Therefore, there is a strong need for an automated and centralized Customer Relationship Management system to overcome these challenges.

2.3. Need of the Project

In today's competitive business environment, organizations are required to manage large volumes of customer data efficiently while delivering high-quality services. Manual handling of customer records is slow, error-prone, and difficult to manage as the customer base grows. There is a strong need for automation of customer data to ensure accuracy, consistency, and easy maintenance of records. An automated system helps reduce human errors, saves time, and improves overall operational efficiency by streamlining daily business activities.

Faster access to customer information is essential for providing quick support, personalized services, and effective communication. A centralized CRM system enables employees to retrieve customer details instantly,

track interaction history, and monitor service requests in real time. Additionally, better decision-making support is required for business growth, which can be achieved through data analysis and reporting features. The proposed project fulfills these needs by providing a reliable, secure, and user-friendly CRM system that enhances productivity and customer satisfaction.

2.4 .Project Scheduling

The project is planned and executed using the Agile methodology, where development is divided into multiple sprints. Each sprint focuses on completing specific tasks within a fixed time frame. This approach ensures continuous improvement, regular feedback, and timely delivery of the system.

Sprint No.	Duration	Sprint Goal	Activities	Deliverables
Sprint 1	Week 1	Requirement Analysis	Requirement gathering, stakeholder discussion, backlog creation	Product backlog, requirement document
Sprint 2	Week 2	System Design	Architecture design, database design, UI mockups	Design diagrams, database schema
Sprint 3	Week 3	Module Development	Customer module, login system, data validation	Working modules
Sprint 4	Week 4	Feature Implementation	Lead management, sales tracking, complaint handling	Integrated features
Sprint 5	Week 5	Testing	Unit testing, integration testing, bug fixing	Tested system
Sprint 6	Week 6	Deployment	Final deployment, user training, documentation	Live system, user manual

2.5 Objectives

The main objective of this project is to design and develop an efficient Customer Relationship Management (CRM) system that helps organizations manage customer information in a centralized and organized manner. The system aims to improve customer interaction, enhance service quality, and increase overall customer satisfaction through automation and digital transformation.

The specific objectives of the project are as follows:
• To provide a centralized platform for storing and retrieving customer information securely.

• To automate routine tasks such as follow-ups, notifications, and report generation.
• To improve customer service by tracking customer interactions and resolving issues effectively.
• To analyze customer behavior and generate reports for better decision-making.
• To ensure data security through user authentication and role-based access control.
• To implement the system using Agile methodology for continuous improvement and scalability.

3. Software Requirement Specification

3.1 Purpose

The purpose of this project is to design and develop an effective Customer Relationship Management (CRM) system that helps organizations manage customer information in a centralized and well-organized manner. The system aims to improve customer interactions, enhance service quality, and increase customer satisfaction by automating routine business processes. By providing quick access to accurate customer data, the CRM system supports better decision-making and improves operational efficiency. The project also focuses on building a secure, user-friendly, and scalable solution that can be easily adapted to meet future business requirements.

3.2 Scope

The scope of this project includes the design and development of a Customer Relationship Management (CRM) system that supports efficient management of customer information and interactions. The system provides a centralized database for storing, updating, and retrieving customer details securely. It covers essential functionalities such as customer registration, lead management, sales tracking, complaint handling, and report generation.

The project also includes features like user authentication, role-based access control, and automated notifications to improve security and communication. The system is designed to be scalable, allowing future enhancements such as advanced analytics, cloud integration, and mobile application support. However, third-party integrations and AI-based features are not included in the current phase and can be implemented in future versions.

3.3 Hardware Requirements

The following hardware components are required for the successful development and execution of the Customer Relationship Management (CRM) system.

Component	Specification	Description
Processor	Intel Core i3 or above	Required for smooth application execution and development
RAM	Minimum 4 GB (8 GB recommended)	Ensures efficient multitasking and system performance

Component	Specification	Description
Hard Disk	Minimum 250 GB free space	Required for storing application files and database
System Type	64-bit architecture	Supports modern operating systems and development tools
Input Devices	Keyboard, Mouse	Required for user interaction
Output Devices	Monitor, Printer (optional)	For viewing application output and reports
Internet Connection	Required	Needed for updates, cloud access, and collaboration

3.4 Software Requirements

The following software components are required for the development and execution of the Customer Relationship Management (CRM) system.

Software Component	Specification	Description
Operating System	Windows 10 / Linux / macOS	Platform for running the application and development tools
Programming Language	Java	Used for backend development
Database	MySQL	Used for storing customer data
IDE	Eclipse / IntelliJ IDEA	Development environment for writing and managing code
Web Technologies	HTML, CSS, JavaScript	Used for front-end development
Server	Apache Tomcat	Used to deploy and run the web application
Version Control	Git & GitHub	For source code management and collaboration
Testing Tool	Selenium	For automated testing
Documentation Tool	MS Word / Google Docs	For project documentation

3.5 Tools

The following tools are used for the development, testing, deployment, and documentation of the Customer Relationship Management (CRM) system:

Tool	Purpose
Eclipse IDE / IntelliJ IDEA	For writing, compiling, and managing Java source code
MySQL Workbench	For database design and management
Apache Tomcat Server	For deploying and running the web application
Git & GitHub	For version control and team collaboration
Selenium	For automated testing of the application
Postman	For API testing (if applicable)
MS Word / Google Docs	For documentation
Draw.io	For creating system diagrams and flowcharts
Google Chrome / Firefox	For application testing in different browsers

3.6 Software Process Model

The Agile Software Process Model is used for the development of the Customer Relationship Management (CRM) system. Agile is an iterative and incremental approach that focuses on flexibility, continuous improvement, and regular customer feedback. The entire project is divided into small development cycles called sprints, where each sprint delivers a functional part of the system.

Project requirements are collected in the form of user stories and maintained in a product backlog. These requirements are prioritized based on business needs. Agile practices such as sprint planning, daily stand-up meetings, sprint reviews, and sprint retrospectives are conducted regularly to monitor progress and improve team collaboration.

This model allows quick adaptation to changing requirements, ensures timely delivery, and enhances overall product quality. By following Agile methodology, the project achieves better efficiency, transparency, and customer satisfaction.

4 System Design

4.1 Data Dictionary

The Data Dictionary defines all the data elements used in the CRM system along with their structure and purpose.

Customer Table

Field	Type	Description
customer_id	INT	Customer ID
name	VARCHAR	Customer Name
email	VARCHAR	Email
phone	VARCHAR	Phone

Lead Table

Field	Type	Description
lead_id	INT	Lead ID
customer_id	INT	Linked Customer
status	VARCHAR	Lead Status

Ticket Table

Field	Type	Description
ticket_id	INT	Support Ticket
customer_id	INT	Customer
issue	VARCHAR	Issue

4.2 ER Diagram

The ER Diagram shows the structure of the CRM system by representing the main entities and their

relationships. It illustrates how customer information is connected with leads, sales, support tickets, and feedback records. The diagram helps in understanding data organization and ensures proper data integrity and efficient database design.
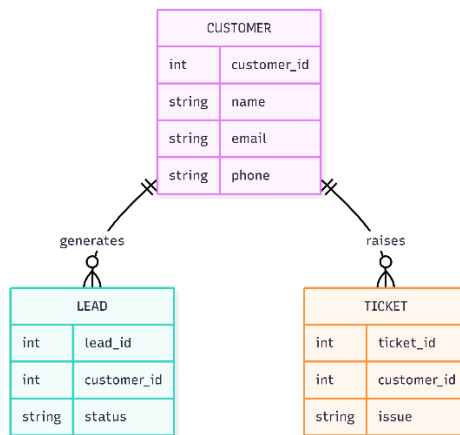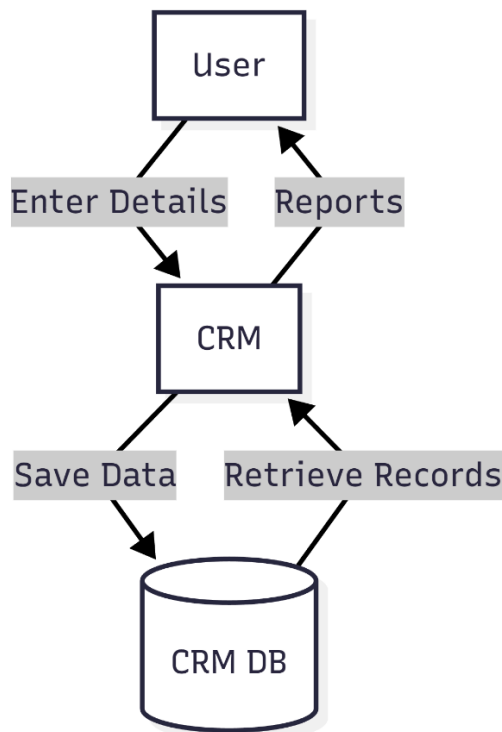

Fig 4.2 ER DIAGRAM

4.3 Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) represents how data moves between users, system processes, and the CRM database. It illustrates the flow of customer information through modules such as customer management, lead management, sales management, and support management

4.4 Use Case Diagram

The Use Case Diagram shows the interaction between different users and the CRM system. It represents how users such as Admin, Sales Executive, and Support Agent perform various operations like managing customers, leads, sales, and support tickets. The diagram helps in understanding system functionality from the user's perspective and defines the roles and responsibilities of each user within the system.
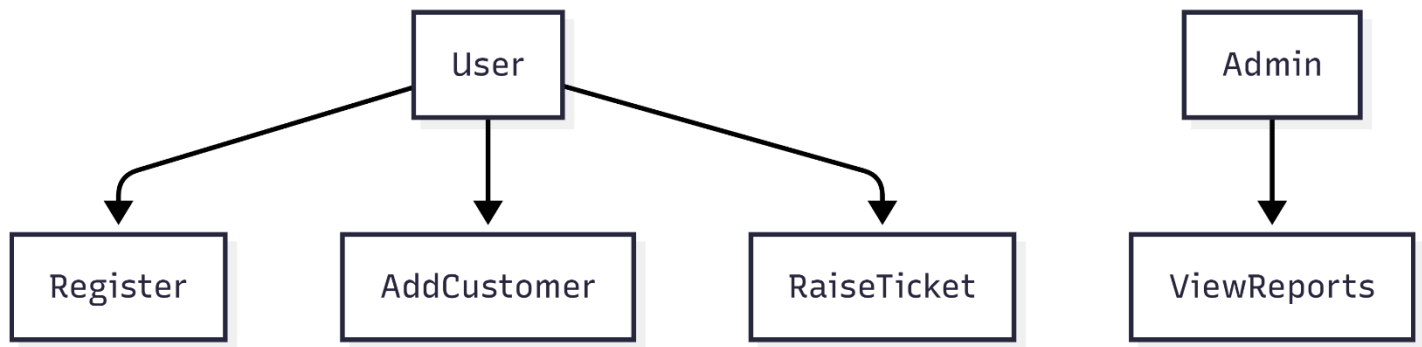

FIG 4.4 USE CASE DIAGRAM

5 Implementation

The implementation phase involves converting the system design into a fully functional Customer Relationship Management (CRM) application. The system is developed using Java for backend processing, MySQL for

database management, and web technologies for the user interface. The application follows a modular architecture to ensure maintainability and scalability.

5.1 Program Code

The program code is written using Java and follows object-oriented programming principles to ensure clarity, reusability, and easy maintenance. Each module such as customer management, lead management, sales tracking, and support handling is implemented as a separate class or package. Database connectivity is handled using JDBC to perform operations such as insert, update, delete, and retrieve customer records.

Input validation is implemented to prevent incorrect or incomplete data entry. Exception handling mechanisms are used to manage runtime errors and ensure system stability. Proper comments are added throughout the code to improve readability and support future enhancements.

Sample code snippet for adding a new customer record:

```
class Customer {

    int id;

    String name, email;


    Customer(int id, String name, String email) {

        this.id = id;

        this.name = name;

        this.email = email;

    }


    void display() {

        System.out.println("Customer ID: " + id);

        System.out.println("Name: " + name);

        System.out.println("Email: " + email);

    }


    public static void main(String[] args) {

        Customer c1 = new Customer(101, "Rahul", "rahul@gmail.com");

        c1.display();

    }}
```

5.2 Output Screens

The output screens provide a user-friendly interface that allows users to interact with the CRM system easily. The main dashboard displays menu options such as Add Customer, View Customer, Update Customer, Delete Customer, Lead Management, Sales Tracking, and Support Management.

The Add Customer screen allows users to enter customer details such as ID, name, email, and contact number. After submitting the form, a confirmation message is displayed showing that the customer record has been added successfully. The View Customer screen displays all stored customer records in a tabular format, making it easy to search and manage data. Each screen is designed with clear labels, buttons, and navigation options to ensure smooth user experience.

6 Testing

Testing is performed to verify that the Customer Relationship Management (CRM) system works correctly and meets all specified requirements. The main objective of testing is to identify defects, ensure proper functionality, and validate that the system produces accurate results for different user inputs.

Different test cases are executed for modules such as login, customer management, lead management, sales tracking, and support handling. Both valid and invalid inputs are tested to ensure correct system behavior. Testing helps in improving system reliability, performance, and security before final deployment.

6.1 Test Data

Test data is used to check the correctness and accuracy of the CRM system. Sample customer records are entered into the system to verify data validation, storage, and retrieval functionality.

TEST DATA TABLE

Test Case ID	Customer ID	Name	Email	Phone	Expected Result
TC_01	101	Rahul	rahul@gmail.com	9876543210	Record added successfully
TC_02	102	Anjali	anjali@gmail.com	8765432109	Record added successfully
TC_03	103	Karan	karan@gmail.com	7654321098	Record added successfully
TC_04	104	Priya	priya@gmail.com	6543210987	Record added successfully
TC_05	105	Amit	amit@gmail.com	9988776655	Record added successfully

6.2 Test Result

The test results show that all test cases were executed successfully and the system behaved as expected. Customer records were added correctly without any data loss or duplication. The search and retrieval functions returned accurate customer details, and input validation prevented incorrect data entry.

The login module authenticated users properly, and role-based access control restricted unauthorized access. The CRM system handled all operations smoothly without errors or crashes. Based on the test results, the system is reliable, secure, and ready for deployment.

TEST RESULT TABLE

Test Case ID	Module	Test Scenario	Expected Result	Actual Result	Status
TC_01	Login	Login with valid credentials	Login successful	Login successful	Pass
TC_02	Customer	Add new customer record	Record added	Record added	Pass
TC_03	Customer	Search customer record	Customer found	Customer found	Pass
TC_04	Customer	Update customer record	Record updated	Record updated	Pass
TC_05	Customer	Delete customer record	Record deleted	Record deleted	Pass

7 User Manual

The User Manual provides step-by-step instructions to help users operate the Customer Relationship Management (CRM) system efficiently. It explains how to use the main features of the system, including

customer management, lead management, sales tracking, and support handling. The manual is designed for users with basic computer knowledge and does not require any technical background. By following the guidelines provided, users can perform daily business operations smoothly and securely.

The CRM system offers a simple and user-friendly interface that allows users to manage customer information, track business activities, and generate reports easily. Proper login authentication ensures data security and prevents unauthorized access.

7.1 How to Use Project Guidelines

Follow the steps below to use the CRM system:

- Start the application and log in using valid username and password.

- After successful login, the dashboard will be displayed with all available modules.

- Select **Customer Management** to add, view, update, or delete customer records.

- Select **Lead Management** to track potential customers and sales opportunities.

- Use the **Sales Module** to record and monitor sales transactions.

- Use the **Support Module** to register and track customer complaints.

- Generate reports from the **Reports Module** for business analysis.

- Log out securely after completing your work.

7.2 Screen Layouts and Description

The CRM system consists of multiple screens designed for easy navigation and efficient operation.

- **Login Screen:** Allows users to enter their credentials and access the system securely.

- **Dashboard Screen:** Displays all system modules such as Customer, Lead, Sales, Support, and Reports.

- **Add Customer Screen:** Allows users to enter customer details such as ID, name, email, and phone number.

- **Customer List Screen:** Displays stored customer records in a tabular format.

- **Reports Screen:** Shows analytical reports and summaries for decision-making.

8 Project Applications and Limitations

8.1 Applications

The Customer Relationship Management (CRM) system can be used in various business environments to manage customer information efficiently. It provides a centralized platform for storing and retrieving customer

data, which helps organizations track customer interactions and history. The system supports quick searching, updating, and deletion of customer records, making daily operations faster and more accurate.

The CRM system is useful for sales teams, customer support departments, and marketing teams. It improves customer service by enabling timely follow-ups and better communication. The system is suitable for small, medium, and large organizations and helps in reducing manual work, improving data accuracy, and enhancing overall business productivity.

8.2 Limitations

The current version of the CRM system supports only basic customer information management. Advanced analytics and reporting features are limited. The system does not include integration with external applications such as email services, payment gateways, or third-party tools.

Multi-user access and cloud-based deployment are not implemented in the current phase. Manual data entry may still lead to human errors. The system also lacks automated backup and recovery features, which can be improved in future versions.

9 Conclusion and Future Enhancement

9.1 Conclusion

The Customer Relationship Management (CRM) project successfully meets its objectives by providing an efficient and user-friendly system for managing customer information. The application simplifies data storage, retrieval, and updating through a centralized platform. Testing confirms that all modules function correctly and deliver accurate results.

The system improves operational efficiency, enhances customer service, and supports better decision-making. Overall, this project demonstrates how an automated CRM solution can transform traditional customer management processes into a modern, digital system.

9.2 Future Enhancement

The following enhancements can be implemented in future versions of the CRM system:

- Add advanced search and filter options
- Implement graphical dashboards and analytics
- Integrate email and SMS notification services
- Enable cloud-based deployment for remote access
- Add mobile application support
- Implement data backup and recovery system
- Add AI-based customer behavior prediction

- Support multi-user access with role-based permissions

10 Bibliography

1. Pressman, R. S., *Software Engineering: A Practitioner's Approach*, McGraw-Hill Education
2. Sommerville, I., *Software Engineering*, Pearson Education
3. Kendall, K. E., & Kendall, J. E., *Systems Analysis and Design*, Pearson Education
4. Laudon, K. C., & Laudon, J. P., *Management Information Systems*, Pearson Education
5. Oracle Java Documentation – Official Website
6. MySQL Reference Manual – MySQL Official Documentation
7. Agile Alliance – Agile Methodology Guide
8. Scrum.org – Scrum Framework Documentation
9. W3Schools – https://www.w3schools.com
10. GeeksforGeeks – https://www.geeksforgeeks.org
11. TutorialsPoint – https://www.tutorialspoint.com