

Customer Relationship Management (CRM) System Using Agile Methodology

(Java – Console Based Application)

1. Abstract

Customer Relationship Management (CRM) is an essential business strategy that focuses on building and maintaining long-term relationships with customers. In today's competitive environment, organizations must effectively manage customer information, interactions, and feedback to improve customer satisfaction and business performance. Manual or traditional customer management systems often lead to data inconsistency, redundancy, and inefficiency. To overcome these issues, automated CRM systems are widely adopted.

This project presents the development of a **Customer Relationship Management (CRM) system using Agile methodology**, implemented as a **Java-based console application**. The system is designed to manage customer records such as customer identification details, contact information, and interaction history in an organized and efficient manner. The console-based approach makes the application lightweight, easy to use, and suitable for academic learning as well as small organizations.

Agile methodology is adopted in this project to ensure flexibility, faster development, and continuous improvement. The entire development process is divided into short development cycles called sprints. Each sprint focuses on delivering a specific functionality such as customer data entry, record modification, searching, or report generation. Regular feedback and testing during each sprint help in identifying errors early and improving system quality.

The project is developed using Java programming language due to its platform independence, object-oriented nature, security features, and widespread usage. Testing is carried out using various test cases to ensure that the system performs correctly under different scenarios. The CRM system developed through this project demonstrates the practical application of Agile methodology and provides scope for future enhancements such as database integration, graphical user interface, and web-based deployment. This project serves as a foundation for understanding real-world software development practices using Agile principles.

2. Introduction

2.1 Introduction

Customer Relationship Management (CRM) refers to the combination of processes, strategies, and technologies used by organizations to manage and analyze customer interactions throughout the customer lifecycle. The main goal of CRM is to improve customer satisfaction, retain customers, and increase profitability. A CRM system stores customer information in a centralized manner, making it easy to access and manage.

2.2 Problem Identification

In many organizations, customer information is maintained manually using registers or spreadsheets. Such systems are inefficient, time-consuming, and prone to errors. Searching, updating, or deleting customer records becomes difficult as the volume of data increases. Lack of proper data security and data backup are additional challenges.

2.3 Need of the Project

There is a strong need for an automated CRM system that can efficiently manage customer records and reduce human effort. A Java-based console CRM application provides a simple, cost-effective, and reliable solution for managing customer information, especially for small organizations and educational purposes.

2.4 Project Scheduling

The project is developed using Agile methodology and divided into the following phases:

- Requirement Analysis and Planning
- System Design
- Implementation
- Testing
- Documentation

Each phase is completed in short iterations to ensure timely delivery and flexibility.

2.5 Objectives

- To design and develop a CRM system using Java
- To implement Agile methodology in software development
- To manage customer data efficiently
- To gain practical knowledge of object-oriented programming concepts

3. Software Requirement Specification (SRS)

3.1 Purpose

The purpose of this Software Requirement Specification (SRS) is to describe the functional and non-functional requirements of the CRM system. It serves as a reference for developers and users to understand system behavior and limitations.

3.2 Scope

The scope of this project is limited to a console-based CRM system. The system allows the administrator to add, update, delete, search, and display customer information. Advanced features such as online access and database connectivity are beyond the current scope.

3.3 Hardware / Software Requirements

Hardware Requirements:

- Processor: Intel i3 or higher
- RAM: Minimum 4 GB
- Hard Disk: Minimum 10 GB free space

Software Requirements:

- Operating System: Windows 10 / Windows 11
- Programming Language: Java (JDK 8 or above)
- IDE: Eclipse IDE

3.4 Tools

- Java Development Kit (JDK)
- Eclipse IDE
- GitHub (for version control and report hosting)

3.5 Software Process Model

The Agile Software Development Model is used in this project. Agile emphasizes iterative development, customer collaboration, continuous testing, and quick response to change.

4. System Design

4.1 Data Dictionary

Field Name	Description
------------	-------------

CustomerID	Unique customer identification number
Name	Name of the customer
Phone	Contact number of the customer
Email	Email address
Address	Residential address

4.2 ER Diagram

The Entity Relationship (ER) diagram represents the relationship between different entities in the system. The main entity is Customer, which contains attributes such as CustomerID, Name, Phone, Email, and Address. The Admin entity manages customer records.

4.3 Data Flow Diagram (DFD)

The Data Flow Diagram illustrates the flow of data between the user and the CRM system. It shows processes such as adding customer details, updating records, searching information, and displaying reports.

4.4 Use Case Diagram

The Use Case Diagram identifies the interaction between the Admin and the CRM system. The main use cases include Add Customer, Update Customer, Delete Customer, Search Customer, and View All Customers.

```

-----|-----|
| CustomerID | Unique identification number |
| Name | Customer name |
| Phone | Contact number |
| Email | Email address |
| Address | Customer address |

```

4.2 ER Diagram

Entities include Customer and Admin. Each customer has a unique customer ID and related attributes.

4.3 Data Flow Diagram (DFD)

The DFD shows the flow of data between the user and the CRM system for operations like add, update, and search customer.

4.4 Use Case Diagram

Actors: Admin

Use cases: Add Customer, Update Customer, Delete Customer, Search Customer, View All Customers.

5. Implementation

5.1 Program Code

The CRM system is implemented using Java programming language. Object-oriented concepts such as classes, objects, encapsulation, and collections are used. The application follows a menu-driven approach where the user selects options to perform different operations.

5.2 Output Screens

The output of the application is displayed on the console. The system shows menus, prompts for user input, and displays messages indicating successful or unsuccessful operations.

6. Testing

6.1 Test Data

Sample customer records with different values are used as test data to verify the system functionality. Boundary cases such as empty input and invalid data are also tested.

6.2 Test Result

All test cases were executed successfully. The system produced correct output for valid inputs and displayed appropriate error messages for invalid inputs.

7. User Manual

7.1 How to Use Project Guidelines

1. Start the Java application from the IDE or command prompt
2. Select the required option from the main menu

3. Enter customer details when prompted
4. View results on the console
5. Exit the application safely after use

7.2 Screen Layouts and Description

The application uses a text-based menu displayed on the console. Each menu option is clearly labeled for ease of use.

8. Project Applications and Limitations

Applications

- Small-scale businesses
- Educational institutions
- Training and learning purposes

Limitations

- Console-based user interface
 - No database connectivity
 - Limited scalability
-

9. Conclusion and Future Enhancement

The Customer Relationship Management system developed using Java and Agile methodology successfully meets the objectives of managing customer data efficiently. The project helped in understanding Agile practices, requirement analysis, system design, and implementation. The console-based CRM application is simple, reliable, and easy to use.

Future enhancements of this project may include integration with a database, development of a graphical user interface, role-based access, and deployment as a web or mobile application.

10. Bibliography & References

- Agile Manifesto
- Java Programming Documentation

- Software Engineering Textbooks