# A

**PROJECT REPORT**

**ON**

**PROJECT TITLE**

***IN FULFILMENT OF***

***MASTER’S DEGREE IN COMPUTER APPLICATIONS (M.C.A.)***

***GUJARAT TECHNOLOGICAL UNIVERSITY (2020-2021)***

******



**GOVERNMENT MCA COLLEGE**

**MANINAGAR, AHMEDABAD**

***(Affiliated to the Gujarat Technological University)***

**SUBMITTED BY: -**

ENROLLMENT NO STUDENT NAME

****



**GOVERNMENT MCA COLLEGE**

**MANINAGAR, AHMEDABAD**

***(Affiliated to the Gujarat Technological University)***

# CERTIFICATE

This is to certify that System Development Project of “**Project Title”** developed and submitted to Gujarat Technological University by **Student Name (Enrollment No)** fulfilmentof the requirement of MCA **SEM-4** for the award of the **“Master Degree in Computer Applications (M.C.A.)”** in the year 2020-2021.

This is the original work and carried out under guidance and supervision. We further certify that to the best of our knowledge and belief the matter presented in this project report is Bonafide certificate.

Date of Submission:

**Guide Name Prof B.B. Prajapati Dr. Chetan B. Bhatt** Designation Head of Department Principal

Government MCA College, Maninagar

Government MCA College, Maninagar

Government MCA College, Maninagar

# TABLE OF CONTENTS

1. **Introduction**
   1. Existing System
   2. Need for the New System……………………………....
   3. Objective of the New System
   4. Problem Definition
   5. Core Components
   6. Project Profile
   7. Assumptions and Constraints…………………………...
   8. Advantages and Limitations of the System………........
2. **Requirement Determination & Analysis**
   1. Requirement Determination
   2. Targeted Users
3. **System Design**
   1. Use Case Diagram
   2. Class Diagram
   3. Interaction Diagram
   4. Activity Diagram
   5. Entity Relationship Diagram
   6. Data Flow Diagram
   7. Data Dictionary
4. **Development**
   1. Coding Standards
   2. Sample Code
   3. Working project screenshot
5. **Project Management** 
   1. Agile Project Charter………………………....................
   2. Agile Roadmap / Schedule
   3. Agile Project Plan
   4. Agile User Story
   5. Agile Release Plan
   6. Agile Sprint Backlog
   7. Agile Test Plan
   8. Agile Burn Charts
6. **Project Finance**
   1. Required Project Resources
      * Human Resources
      * Reusable Software Resources
      * Hardware Resources
   2. Estimate Cost and Efforts
      * Technical Complexity
      * Functionalities
      * Location
      * TimeFrame
      * Reconcile the estimates at each phases of development
7. **Proposed Enhancements**
8. **Conclusion**
9. **Bibliography**
10. Introduction 1.1 Existing System The existing system of the local store involves traditional brick-and-mortar operations with limited online presence, such as a basic website or no online platform at all. Customers have to visit the physical store to make purchases, which may not always be convenient, especially in the era of e-commerce and online shopping.

1.2 Need for the New System The need for a new system arises from the increasing demand for online shopping and the changing consumer preferences towards convenience and ease of use. An online local store web application will provide the store with a digital presence, enabling customers to shop online, browse products, place orders, and make payments from the comfort of their own homes or on the go.

1.3 Objective of the New System The objective of the new system is to create a user-friendly and efficient web application that will enhance the local store's online presence, attract more customers, increase sales, and provide a convenient shopping experience to customers.

1.4 Problem Definition The main problem identified in the existing system is the lack of an online platform for the local store, which limits its reach and potential customer base. Additionally, the manual process of managing orders, payments, and inventory can be time-consuming and error-prone. The new system aims to address these challenges by automating processes and providing an online platform for customers to shop conveniently.

1.5 Core Components The core components of the online local store web application include the frontend built in ReactJS, the backend built in NodeJS, and the database using MongoDB. The frontend is responsible for the user interface and user experience, while the backend handles the logic and communication with the database.

1.6 Project Profile The project profile includes the details of the web application, such as the features, functionalities, and technologies used. It also includes information on the team members involved in the development, project timeline, and milestones.

1.7 Assumptions and Constraints The assumptions and constraints of the project may include limitations in terms of time, budget, and resources. It may also consider assumptions made during the requirement gathering and analysis phase, such as assumptions about user behavior, market trends, and technological feasibility.

1.8 Advantages and Limitations of the System The advantages of the online local store web application include increased customer reach, convenience for customers, automation of processes, and potential for increased sales. However, limitations may include technological limitations, such as internet connectivity, hardware requirements, and user familiarity with online shopping.

2 Requirement Determination & Analysis 2.1 Requirement Determination The requirement determination phase involves gathering and analyzing the requirements of the web application. This includes identifying the features, functionalities, and user expectations. It may involve conducting surveys, interviews, and market research to understand customer needs and preferences.

2.2 Targeted Users The targeted users of the online local store web application include customers who prefer to shop online, local store owners who want to expand their business online, and store administrators who manage orders, payments, and inventory. Understanding the needs and expectations of these users is crucial for designing an effective and user-friendly web application.

Sure, here are the code diagrams for the system design of the online local store web application:

1. Use Case Diagram:

A use case diagram is a visual representation of the functional requirements of a system from the perspective of its users. It illustrates the interactions between actors (users or external systems) and the system itself. Here's an example of a use case diagram for the online local store web application:

```

[Use Case Diagram]

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

| User |

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

| - Browse Products |

| - Search Products |

| - Add Product to Cart |

| - View Cart |

| - Checkout |

| - Place Order |

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

|

| <<include>>

|

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

| Admin |

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

| - Manage Products |

| - Manage Orders |

| - View Sales Report |

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

```

2. Class Diagram:

A class diagram is a visual representation of the classes, objects, and relationships between them in a system. It provides an overview of the structure of the system and the interactions between different classes. Here's an example of a class diagram for the online local store web application:

```

[Class Diagram]

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

| User |

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

| - username: string |

| - password: string |

| - email: string |

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

| + browseProducts() |

| + searchProducts() |

| + addToCart() |

| + viewCart() |

| + checkout() |

| + placeOrder() |

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

|

|

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

| Product |

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

| - productId: int |

| - productName: string |

| - price: float |

| - description: string |

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

| + getProductDetails() |

| + updateProductDetails()|

| + deleteProduct() |

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

```

3. Interaction Diagram:

An interaction diagram is a visual representation of the interactions between objects or components in a system, typically showing the flow of messages or method calls between them. Here's an example of an interaction diagram for the online local store web application:

```

[Interaction Diagram]

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

| User |

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

| - username: string |

| - password: string |

| - email: string |

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

| + browseProducts() |

| + searchProducts() |

| + addToCart() |

| + viewCart() |

| + checkout() |

| + placeOrder() |

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

| |

| |

--------------------------- | ---------------------------

| Product | | | Cart |

--------------------------- | ---------------------------

| - productId: int |<------|--------->| - items: List<Product> |

| - productName: string | | | - total: float |

| - price: float | | ---------------------------

| - description: string | | | + addItem() |

--------------------------- | | + removeItem() |

| + getProductDetails() | | | + updateCart() |

| + updateProductDetails()| | | + viewCartDetails() |

| + deleteProduct() | | ---------------------------

--------------------------- |

```

4. Activity Diagram:

An activity diagram is a visual representation of the flow of activities or actions in a system, typically showing the steps or processes involved in a specific task or use case. Here's an example of an activity diagram for the online local store web application:

```

[Activity Diagram]

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

| User |

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

| - username: string |

| - password: string |