D.K.T.E. Society's Textile and Engineering Institute, Ichalkaranji

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)
Accredited with 'A+' Grade by NAAC

Department of Computer Science & Engineering
2024-2025

Promoting excellence in
Teaching, Learning & Research

Mini Project I Report

on

"Parag Refrigeration and Engineering Works Website"

Under the guidance

Of

Prof. Mrs. P.S.More

Submitted By:

Name of Student

Karan Mahendra Nagure [22UAD046]

Harshvardhan Pratap Londhe [22UAD036]

Harshal Makarand Mali [22UAD038]

D.K.T.E. Society's Textile and Engineering Institute,
Ichalkaranji

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)
Accredited with 'A+' Grade by NAAC

Department of Computer Science & Engineering
2024-2025

CERTIFICATE

This is to certify that

Name of Student

Karan Mahendra Nagure [22UAD046]

Harshvardhan Pratap Londhe [22UAD036]

Harshal Makarand Mali [22UAD038]

Have successfully completed the Mini Project I work entitled,

"Parag Refrigeration and Engineering Works Website**"**

For Third Year Vth Semester in Computer Science and Engineering department (AI and DS). This is the record of their work carried out during academic year 2024-2025.

Date:- 22 – Sep - 2024 Place:-Ichalkaranji

Prof. Mrs. P.S.More
Guide

Prof. Dr. T. I. Bagban
HOD CSE AIDS

DECLARATION

We undersigned hereby declare that the project report entitled "Parag Refrigeration and Engineering Works Website" is an work carried out by us during Third Year Vth Semester course under the guidance of Mr. S. R. Shinge. I have not copied from any project report previously submitted for the award of any degree or diploma of this university. Any such copying is liable to be punished in a way the university authorities may deem fit.

Date: - 22 – Sep - 2024

Place: - Ichalkaranji

INDEX

Software Requirements Specification (SRS) for Parag Refrigeration and Engineering Works Website

1. Introduction

1.1 Problem Description

Currently, Parag Refrigeration and Engineering Works lacks an integrated, user-friendly web platform to display products, services, and company information, limiting customer outreach and operational efficiency.

1.2 Problem Statement

The absence of a digital interface hinders both customer engagement and service accessibility. A dedicated website will bridge this gap by showcasing offerings and enabling users to request services online.

1.3 Objectives

- Create an interactive, responsive website with product and service details.
- Include a user-friendly backend for easy content management.
- Integrate customer interaction features like service requests and inquiries.

1.4 Scope

Product Name: Parag Refrigeration Website
Key Features: Product catalog, inquiry forms, service booking.
Exclusions: No e-commerce functionality in the initial phase.
Application: Primarily for showcasing business offerings and allowing service booking.

2. Overall Description

2.1 Product Perspective

System Components:

1. Frontend: User Interface for customers.
2. Backend: Content Management System (CMS) for admins.
3. Database: Stores customer inquiries, service bookings, and product data.

2.1.1 Block Diagram

(Include a block diagram showcasing the frontend-backend-database flow.)

2.1.2 Hardware Requirements

- Web server with minimum 4 GB RAM and 50 GB storage
- Internet connection for remote access

2.1.3 Software Requirements

- Frontend: HTML, CSS, JavaScript
- Backend: FLASK (Python)
- Database: MySQL

2.2 Product Functions

- Display product catalog
- Allow users to submit inquiries and service requests
- Admin panel for updating content dynamically

3. Specific Requirements

3.1 External Interfaces

- User Input: Service requests, inquiries
- System Output: Confirmation messages, notifications to admin

3.2 Functional Requirements

- The system shall allow users to submit inquiries through an online form.
- The system shall notify administrators via email for each service request.
- The admin panel shall enable modification of product and service details.

3.3 Design Constraints

- Must follow modern UI/UX guidelines.
- Compatible with all major browsers.

3.4 Logical Database Requirements

- Tables: Users, Products, Services, Inquiries
- Data Relationships: One-to-Many (Service Bookings to Users)
- Data Retention: Maintain inquiry records for 1 year.

3.5 Software System Attributes

3.5.1 Availability

Available 24/7, with routine maintenance windows on weekends.

3.5.2 Security

- Use SSH encryption for secure communication.
- Role-based access for admin users.

3.5.3 Maintainability

Modular backend code for easy updates.

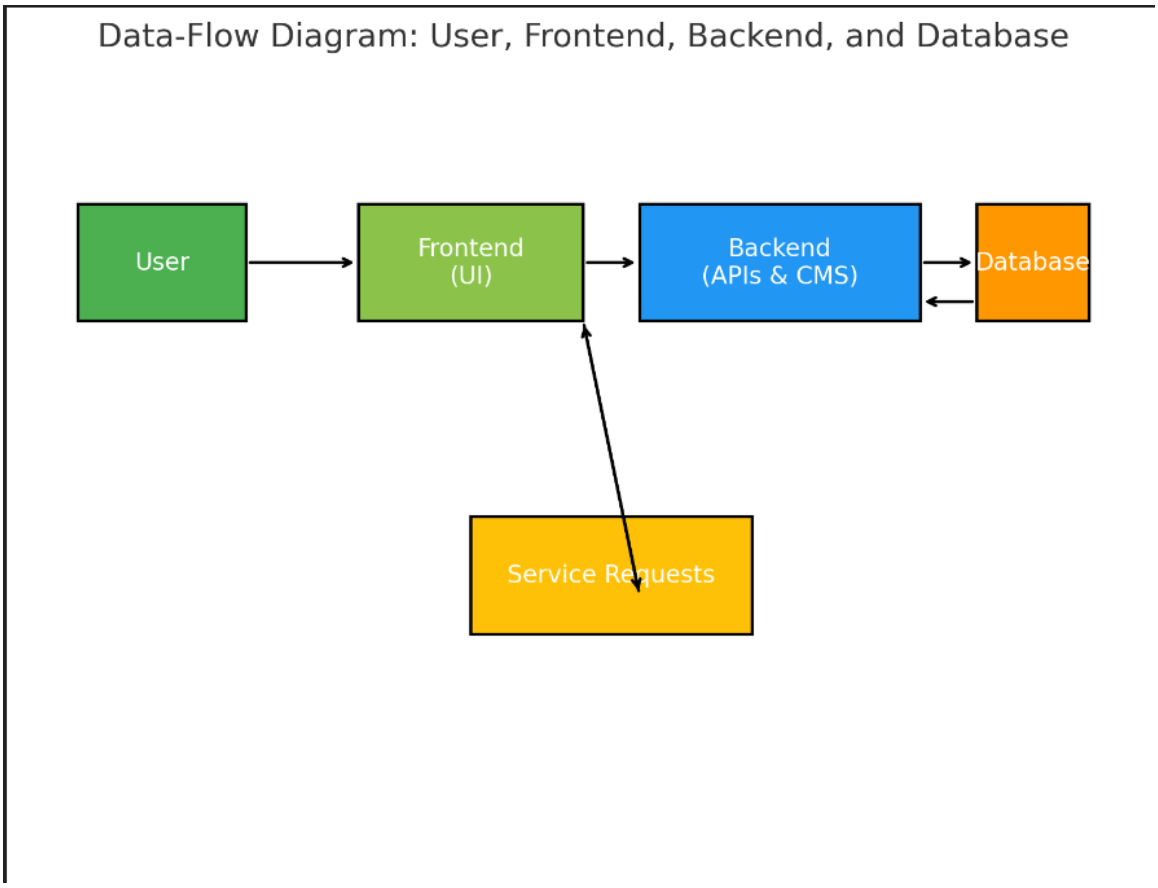
3.5.4 Portability

Compatible with AWS cloud-based hosting platforms.

4. Software Design Document

4.1 Structural Design

- Class Diagrams: Define structure of product, user, and service entities.



5. Algorithms for Functional Requirements

```python
from flask import Flask, render_template, request, redirect
import MySQLdb
import my_sql
app = Flask(__name__)
@app.route('/')
def index():
    return render_template('index.html')

@app.route('/form_k')
def form_k():
    return render_template('form_k.html')

@app.route('/submit', methods=['POST'])
def submit():
    # Get form data
    Name = request.form['name']
    Email = request.form['email']
    Number = request.form['Number']
    PinCode = request.form['PinCode']
    Category = request.form['Category']
    State = request.form['State']
    City = request.form['City']
    Address = request.form['Address']

    # Create a cursor
    # cursor = mydb.cursor()

    # Execute SQL query to insert data
    my_sql.my_cursor.execute("INSERT INTO Form_Details.form (Name, Email, Number, PinCode, Category, State, City, Address)
    VALUES (%s, %s, %s, %s, %s, %s, %s, %s)", (Name, Email, Number, PinCode, Category, State, City, Address))
    # Commit to the database
    my_sql.mydb.commit()
    # Redirect to a thank you page or back to the form
    return redirect('/ThankYou')

@app.route('/ThankYou')
def ThankYou():
    return render_template('ThankYou.html')
```

6. References

- Web Development Standards and Guidelines
- MySQL Documentation for Database Integration
- UI/UX Best Practices