**Sales Product Management Web Application**

# 1. Introduction:

This document provides a comprehensive overview of the Sales Product Management web application, which helps manage products and sales records efficiently. This application is built using ReactJS for the frontend and Django for the backend with a PostgreSQL database.

# 2. Features:

* Product listing and CRUD operations
* Responsive design with a modern UI
* Automated startup using systemd on Ubuntu

# 3. Project Structure:

**Frontend Path:** D:\Nandhinisk\Work from home\Django\Django projects\Sample project\frontend\sales-frontend

**Backend Path:** D:\Nandhinisk\Work from home\Django\Django projects\Sample project\testing

# 4. Technologies Used:

* Frontend: ReactJS, HTML, CSS, JavaScript
* Backend: Django (Python)
* Database: PostgreSQL
* Server Management: PM2, systemd

# 5. Setup Instructions:

## 5.1 Frontend (ReactJS)

**1. Navigate to the frontend folder:**

cd /mnt/d/Nandhinisk/Work\ from\ home/Django/Django\ projects/Sample\ project/frontend/sales-frontend

**2. Install dependencies:**

* npm install
* npm install axios react-router-dom
* npm install @mui/material @emotion/react @emotion/styled
* npm install react-slick slick-carousel
* npm install framer-motion

**3. Start using pm2:**

* npm install -g pm2
* npm run build
* npm install -g serve
* pm2 start npm --name "react-app" -- start

**4. Make pm2 start on reboot:**  
 pm2 startup  
 pm2 save

## 5.2 Backend (Django):

**1. Navigate to the backend folder:**  
 cd /home/nandhini/sample\_project/testing

**2. Create a virtual environment:**  
 python3 -m venv env

**3. Activate the environment:**  
 source env/bin/activate

**4. Install dependencies:**

* pip install -r requirements.txt

**5. Apply migrations:**  
 python manage.py migrate

**6. Run the server:**  
 python manage.py runserver

# 6. Systemd Service for Frontend:

[Unit]  
Description=React Frontend Service  
After=network.target  
  
[Service]  
Type=simple  
WorkingDirectory=/mnt/d/Nandhinisk/Work\ from\ home/Django/Django\ projects/Sample\ project/frontend/sales-frontend  
ExecStart=/usr/bin/npm start  
Restart=always  
Environment=PATH=/usr/bin:/usr/local/bin  
Environment=NODE\_ENV=production  
  
[Install]  
WantedBy=multi-user.target

# 7. Systemd Service for Backend:

[Unit]

Description=Django Backend Service

After=network.target

[Service]

Type=simple

WorkingDirectory=/mnt/d/Nandhinisk/Work\ from\ home/Django/Django\ projects/Sample\ project/backend

ExecStart=/usr/bin/python3 manage.py runserver 0.0.0.0:8000

Restart=always

Environment=PATH=/usr/bin:/usr/local/bin

Environment=DJANGO\_SETTINGS\_MODULE=sales\_project.settings

[Install]

WantedBy=multi-user.target

# 8. Monitoring and Status:

**To check the running status of the frontend service:**  
 sudo systemctl status react

**To check the running status of the database service:**

sudo systemctl status postgresql

**To check the running status of the Backend service:**

sudo systemctl status django

**To view pm2 processes:**  
 pm2 status