RESOURCE MANAGEMENT SYSTEM FOR FMCG INDUSTRY

PROJECT SYNOPSIS

OF MINOR PROJECT

BACHELOR OF TECHNOLOGY
Computer Science & Engineering

SUBMITTED BY

DILRAJ SINGH	EKUSPREET SINGH	HARSIMRAN SINGH
2104092	2104097	2104117

GURU NANAK DEV ENGINEERING COLLEGE,
LUDHIANA

INDEX

S.No	Title	Page No.
1	Introduction	1
2	Rationale	2
3	Objectives	3
4	Literature Review	4
5	Feasibility Study	5
6	Methodology/Planning	6
7	Facilities Required	8
8	Expected Outcomes	9
9	References	10

INTRODUCTION

FMCG or Fast Moving Consumer Goods are goods with quick turnover and high-volume sales. These goods include everyday items such as foods, beverages, household and personal care objects. The FMCG Industry is currently the fourth largest sector in the Indian Economy. The sector that this project deals with is Animal Fodder. Due to its rapid rates of production, it is often a challenge for producers to track raw materials efficiently. Swift cycles demand precise inventory oversight and improper tracking risks operational errors.

In collaboration with Natural Animal Diets, our proposal to their problem is to digitize production operations monitoring and simulate Multi-Way Handshaking Protocol in real life to keep track of the raw material stock.

Views : The project will include multiple views but the project can also split/combine the roles if needed. Some of the views are:
Admin View : The person with deepest access to the application. This role can invite employees, read/write raw material data and upload production recipes.

Employee View : The person in charge of viewing the jobs, and updating the completion status. This role will not have access to the company's recipes.

The technology the project utilizes is ReactJs for frontend and Flask for backend because React can leverage reusable components that can maintain a standard interface and Flask, being written in python, is highly flexible and easily debuggable.

RATIONALE

The primary idea behind building this project was to solve a real-life problem. The challenges faced by NAD were minimizable by digitization. The solution was to make a software automating repetitive tasks such as Monitoring and Evaluating stocks which can be implemented using standard web technologies and some scripts written in python.

Some flaws in the current system that motivated the design of the project :

1. Error prone monitoring causing supply/distribution issues.
2. Lack of effective communication.
3. No Alert(s) in case of lack of Raw Materials.
4. Human involvement in excessive calculation problems.

OBJECTIVES

Following are the objectives of the RESOURCE MANAGEMENT SYSTEM FOR FMCG INDUSTRY for Natural Animal Diets :

1. To ensure convenient management of inventory at NAD.
2. To improve communication and mitigate shortages through a robust alert system.
3. To manage and store employee data and monitor the workload.
4. To infer potential repeat customers using sales data.

LITERATURE REVIEW

1. PARALAX (Existing System) : This is a resource management system that mainly implements scheduling. This software implements a handshake algorithm. to manage the workload of employees. It is user friendly.

Area of Improvement : No tracking of raw material usage, nothing about production and sales.

2. ODOO(Existing System) : Odoo is one of the useful resource management systems available in the market. It lets you store your orders, schedule manufacturing based on sales and orders and organize the supply chain. It is a single app for multiple tasks and free to use.

Areas Of Improvement : No in-built alert system.

3. RAW MATERIAL INVENTORY MANAGEMENT SOFTWARE : This article highlights the general required features in a raw material inventory management system. Some key aspects are to maintain accurate inventories, minimize lot sizes, and increase production availability. Some improvements that the article highlights are to maintain supply chain integrity, reduce recall lot sizes, eliminating rework and reruns.

4. As mentioned by Sunitha, K. V. (2012) in her thesis, inventory management is vital for keeping costs down, when meeting regulations. It is difficult to balance demand and supply and inventory management to make sure that the balance is untouched. The trained inventory Management and good quality software will help make inventory management a victory. The ROI of Inventory management has seen better revenue and profits, positive employee ambiance and increase in customer satisfaction.

FEASIBILITY STUDY

1. TECHNICAL FEASIBILITY : Technologies like ReactJs for the frontend and Flask for the backend, known for their stability and flexibility. Plethora of libraries in python to add convenience to the process of development.

2. OPERATIONAL FEASIBILITY : The pre-defined division of roles by the company into Admin and Employee views ensures a clear picture for the app users, promoting efficient workflow management. The App will have standard UI features to make it easy to operate by the end users.

3. ECONOMIC FEASIBILITY : In our collaboration with the industry, the company will assume financial liabilities. We are utilizing Free and Open Source Softwares (FOSS) for this project as much as possible, eliminating the need for large monetary investments in software to achieve our goals.

4. SCHEDULE FEASIBILITY : Incremental and Iterative development cycles will ensure a well defined schedule for each feature and will help in proper time estimation.

5. LEGAL FEASIBILITY : This project focuses on digitizing operations and monitoring, adhering to legal standards for data privacy and security. The implementation of a Multi-Way Handshaking Protocol aligns with ethical considerations by providing a transparent and traceable system for raw material tracking.

METHODOLOGY/PLANNING

1. REQUIREMENTS GATHERING : The beginning of the project includes a clear and concise collection of requirements. Proper communication was maintained while gathering requirements in order to understand the problem completely. The requirements will be noted down for future reference.
2. SWOT ANALYSIS :
   - Strengths: Collaboration improves creative thinking, problem solving capacity and effectiveness in general. Proven, reliable frameworks can assist to fasten development. A consistent workflow ensures homogeneity and efficiency.
   - Weaknesses: Software developed may be susceptible to unexpected changes in the industry. Initial Setup and Implementation costs.
   - Opportunities: The possibility of creating software solutions that can be rather easily adapted across contexts and industries. Integrating context specific requirements.
   - Threats: A lack of proper training or misuse by staff may result in operational inefficiencies, security threats, or data breaches from the software.
3. SOFTWARE DESIGN : A design of the target software will help in visualizing the end product and will act as a guide for UI design, also will eliminate any design inconsistencies beforehand. The design will be created in Canva or Figma.
4. ENVIRONMENT SETTING : A boilerplate project will be created and a standard will be initialized for various mnemonics. All the dependencies will be installed inside a python virtual environment.

5. DATABASE SETUP & FRONT END : All the views and designs will be coded in React using JSX. The components will be isolated for reusability. A database schema will be designed and set up to store the data as required.
6. SOFTWARE CODING : Coding tasks will be organized into separate files, following best practices for code structure and readability. Each module or feature will have its own file or directory, promoting code modularity and ease of maintenance. The integration of React with Flask will be carefully implemented to establish seamless communication between the front end and the back end.
7. UNIT TESTING : Testing of API files individually to determine if each endpoint works correctly in isolation.
8. INTEGRATION TESTING : Testing of combined software to rule out any integration flaws.This is to determine if all components work seamlessly together.
9. CONSTANT FEEDBACK : After every process, feedback will be taken to improve the software/ adjust any new changes/requirements.
10. BUILDING & DEPLOYMENT : The project will initially be hosted in a local network environment for development and testing purposes. Once the software reaches a stable state, it will be deployed to the web or a standalone PC, ensuring accessibility for the NAD Admin and end-users.
11. SOFTWARE DEVELOPMENT MODEL : The development approach outlined aligns with aspects of both the Iterative and Incremental development models.

FACILITIES REQUIRED

Effective software development involves visits to the NAD office for meetings and feedback. To facilitate the process of development, systems with 8 GB RAM and 5 GB storage should suffice for development and testing. Utilizing pre-existing data jumpstarts production. Reliable hosting solutions are crucial for seamless web deployment. Subscribing to relevant paid tools like Canva enhances design aspects and visually appealing final product. These facilities collectively contribute to a successful software development cycle.

EXPECTED OUTCOMES

The software will feature a responsive and user-friendly interface, ensuring a seamless experience for both administrators and employees. Integration of the FrontEnd and BackEnd components will be a key outcome, promoting efficient communication within the system. The resolution of all user needs, such as monitoring raw material stocks, alert system, recipe uploads, will be a priority. The ultimate software outcome will be successful deployment for end users, providing Natural Animal Diets with a digitized and streamlined resource management solution.

REFERENCES

[1] React Documentation [Online]. Available: https://react.dev/ .

[2] Flask's Documentation [Online]. Available: https://flask.palletsprojects.com/en/3.0.x/

[3] Flask's Tutorials by Tech With Tim [Online]. Available: https://www.youtube.com/watch?v=mqhxxeeTbu0&list=PLzMcBGfZo4-n4vJJybUVV3Un_NFS5EOgX.

[4] GeeksForGeeks. How to connect ReactJs with Flask API [Online]. Available: https://www.geeksforgeeks.org/how-to-connect-reactjs-with-flask-api/

[5] Towards Data Science. How To Build & Deploy a React + Flask App [Online]. Available: https://towardsdatascience.com/build-deploy-a-react-flask-app-47a89a5d17d9