|  |  |
| --- | --- |
| Internship Project Title | Operations Management - Inventory Module using MERN Stack |
| Name of the Company | TCS iON |
| Name of the Industry Mentor | Debashis Roy |
| Name of the Institute | Madras Institute of Technology, Anna University |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Start Date | End Date | | Total Effort (hrs.) | | Project Environment | Tools used |
| 12-07-2022 | 10-08-2022 | | 125 | | VS Code, Windows, Firefox | NodeJS with Express, MongoDB Atlas, React, several npm packages |
| Milestone # | 2 | Milestone: | | Complete action items 2, 3 and 4 (Sales, purchases submodules with dashboard and reports) | | |

**TABLE OF CONTENT**

* Acknowledgements
* Objective
* Introduction / Description of Internship
* Internship Activities
* Approach / Methodology
* Flow Diagram
* Reflections on the Internship
* Enhancement Scope
* Link to code and executable file

# Acknowledgements

I thank TCS iON and the industry mentor for their guidance and support as well as for providing necessary information regarding the project.

# Objective

The objective of this project is to develop an automated software for Inventory Management module of Operations Management using MERN Stack.

# Introduction/Description of Internship

The method through which you keep track of your products across the whole supply chain, from purchase to manufacture to final sales, is known as an inventory management system (or inventory system). An inventory management system combines varying software packages to track stock levels and stock movements. The solution can integrate with multichannel sales systems or shipping systems. An inventory management system optimizes inventory levels and ensures product availability across multiple channels.

It controls how you go about managing your company's inventory.

The goal of this project is to create an automated software system utilizing MERN Stack for the Operations Management Inventory Management module. We make use of MongoDB to store information about the required modules such as items, sales orders and credit notes information in the mongo database. We retrieve the information when a request arises from the client with the help of express, which is a package that is used to create the backend for the application. Express handles the requests, manages the routes and fetches and serves information from the database based on the user request. We make use of React to create the front end for the application. React is a JavaScript-based UI development library. Facebook and an open-source developer community run it. Although React is a library rather than a language, it is widely used in web development.

# Internship Activities

Some of the activities to be done during the internship include Pre-project test, activity report, interim project reports, final project report, project test.

# Approach / Methodology

1. Initially we create a backend server to serve and process requests from the client.
2. The backend communicates with the database and retrieves / stores information.
3. For this project I have used MongoDB Atlas which is a MongoDB hosted in the cloud.
4. We then model the data items to be stored in the database. This is done with the mongoose package which helps in data modeling for MongoDB.
5. The models are defined with mongoose schemas.
6. We make models for the items in submodules sales and purchases.
7. The models are made for customers, sales orders, packages, delivery challans, invoices, payments received, sales returns and credit notes in the sales submodule.
8. The models are made for vendors, purchase orders, bills and payments and vendor credit in the purchase’s submodule.
9. Make controllers for the above models to get, post or update information regarding the models.
10. Each controller function is mapped to a specific route. E.g., the route ‘/api/sales/customers’ is used to fetch all customers stored in the system.
11. The frontend is made with React, a java script library for building user interfaces.
12. The frontend communicates with the backend by making get/post and other methods of requests to enable the various functions of inventory management
13. New customers can register, new sales order can be created and so on.
14. The information flow is ensured by gathering details from the database and auto populating with the mongoose populate method wherever possible
15. Similarly, the purchases sub module is designed to create new vendors, purchase orders, bills & payments and vendor credits.
16. The above items have the option to be created as well as display the existing values.
17. The landing page is designed to be a dashboard that includes the links to all sub modules.
18. Various reports such as inventory summary report, inventory aging summary report, product sales report and sales by item/customer are also available to view.

# Flow diagram

# Algorithms

We make use of certain comparisons in the application. For example, displaying the set of sales orders within a specific time period requires calculating the time period from which orders are to be shown and displaying only the valid items.

We also make use of inbuilt search functions in MongoDB which are implemented based on optimized searching/sorting algorithms.

Reflections on the Internship

I'm hoping that working as an intern with TCS ion will allow me to gain greater knowledge of the MERN stack!

# Enhancement scope

There is a lot of scope for improvement of this inventory module.

For instance, the inventory module can be enhanced by connecting with a store database and making real-time changes to the system instead of manually making adjustments.

# Link to code

GitHub repository with documents submitted till now and also the code for the project: <https://github.com/PragadeshBS/tcs-remote-internship>