SUMMER INTERNSHIP

REPORT

WOODBERRY PROJECT MANAGEMENT SYSTEM

RASHMI PRABHU

201712009

MSc-IT

Table of Contents

INTERNSHIP DETAILS

1. **Internship Start Date**: 14th May 2018
2. **Internship End Date**: 14th July 2018
3. **Mentor:**

   Mr. Jaidev Prabhu

   Admin & HR Manager,

   Woodberry,Ahmedabad.

4. **Company Details:**

   B 95-96, Pariseema Complex,

   Above Jade Blue,

   C.G Road,Ahmedabad-380009,

   Gujarat.

5. **Company Website**: https://woodberry.in/
6. **Submitted To:** Dr. Manish Khare
7. **Submitted By:**

   a. Rashmi Prabhu(201712009)
   b. Tulika (201712036)

INTRODUCTION

In any field you rarely have solo projects where the only person working on the project is you, instead the teams that you are required to work in for a project can range from 3 to 10. Of course for a team to be successful communication is the key. But it's not easy to make sure that every person in the team has been told any and all information related to their work. So many applications started providing ways to organize team efforts for better results. But the majority of these applications are designed in a generalized manner so as to cover a wider customer base. But every company will have some processes that are unique to its own processes. Also there are very few applications that focus on providing this service for the manufacturing sector. Builder Trend is one such application which focuses on the manufacturing sector but the drawbacks are that it's paid and costly to use and as it is also a generalized application, the customized processes have to be represented in a roundabout manner. Our client Woodberry was incurring a cost of more than 35,000 per month by using this application just to organize their processes.

We proposed to our client that we will build an application for them which is customized according to their needs and built after studying their business processes. To do this we spent time studying and understanding not only their business processes but how they use the application. By using our application they won't have to pay a monthly charge and will get an easy to use interface without any unnecessary features. We also wanted to see if we could identify something to add to the current features being used. As we had roughly three months for the application we decided to focus on the main and most required features that would be needed by them.

PROJECT DETAILED DESCRIPTION

The idea for this project came from the previous application being used, not only was the UI bulky and cumbersome it was hard to understand and learn. After talking with the employees we came to know that many of the features which were being offered by that application were not required and just made the dashboard and interface more cluttered.

After talking with the employees at length and understanding that application from them, we got the basic understanding of their processes and what they require from such an application. They work according to orders; these orders are referred to them as jobsites as the orders generally pertain to a location .e.g. house, office building or complex. These jobsites would then be further divided into zones which correspond to the different areas or rooms of that place. Now they need to keep track of the different services and furniture that the client wants according to each zone. Each of these pieces of furniture is an item that is added to that zone by the employees. The employees use the application to keep track of the orders according to each zone and this list should be sharable and viewable by all the employees working for that jobsite. But the problem was that the application did not provide such things and they had found a very roundabout and complex way of achieving this in the application.

So in our application we made separate menus for Jobsite, Zones, Categories, Choices, and Items as was required by the customer. We included admin and employee as the two types of users who would have access to this system. They would also be given access rights accordingly. Admin has the permission to add, update and delete anything from any of the menus he can also add employees and remove them. Employee mostly has view access but has the access to add items because as per the client's request the employees should be able to add items without admin permission. While making this application we tried to focus on the client's main requirements and needs were being met, we kept most of the additional features as future scope. We wanted to build an application that would be simple, user friendly and easy to learn. For this we decided to build the application using Node.js for the backend, Vue Js for front end and we used NoSQL database MongoDB.

TECHONOLGIES LEARNT

1. Node.js:

a. We used Node.js to build a RESTful API for our application.
b. It is an open source cross-platform runtime environment that has been written in JavaScript, which made it a good choice for our application.
c. Because of the effective use of JavaScript in Node.js it is easy to create servers.
d. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices.
e. Additionally, its applications can be run on a variety of servers including Microsoft Windows, Mac OS X, and Unix.
f. Node.js uses V8 engine by Google. It has been written in JavaScript and has an exceptional running speed.
g. Another good thing about Node.js is that it makes the running of the coding faster. Thus, Node.js can increase any framework's speed.

2. Vue.js

a. We used Vue.js to create the frontend of our application. We also came to know while researching that Node.js and Vue.js are compatible.
b. Vue.js is one of those new software technologies that are being widely used across the world for web development.
c. Vue.js is actually a JavaScript framework with various optional tools for building user interfaces.
d. With Vue.js, you can write an app very quickly and run it straight from the browser.
e. It is also flexible enough that it accommodates for different ways of writing code, for example, you can write a template in HTML, or in JavaScript, or use JSX.

f. It facilitates you to integrate with the existing applications. This is because it is based on JavaScript framework and can be integrated into other applications built on JavaScript.
g. In addition Vue.js documentation is very detailed, easy to understand, and includes various code examples.

3. MongoDB:

a. We used MongoDB for our database. It is a NoSQL database.
b. MongoDB is an open source database that uses a document-oriented data model.
c. Instead of using tables and rows as in relational databases, MongoDB is built on architecture of collections and documents.
d. Documents comprise sets of key-value pairs and are the basic unit of data in MongoDB.
e. Collections contain sets of documents and function as the equivalent of relational database tables.
f. Like other NoSQL databases, MongoDB supports dynamic schema design, allowing the documents in a collection to have different fields and structures.

WHAT WE LEARNT?

1. **New Skills:**

   We gained skills that are actually in demand in the industry right now. We learnt new technologies and new languages like Node.js, Vue.js and MongoDB which will prove useful to us in the future. As challenging as it was to work with new technologies it was also rewarding.

2. **Time Management:**

   Time management is what many times makes or breaks a project. So we had to be careful that we followed deadlines, completed our work on time, so that the project would stay on track.

3. **Team Work:**

   In this project the team work required was not only amongst us but also with the company. We not only had to ensure proper communication between our team but also the employees of the company. This was necessary to ensure that we gather proper requirements and make a product which matches our client's needs.