

CHAPTER 1

INTRODUCTION

1.1 Client / Organization Profile :

Name : Hotel Surbhi Pure Veg

Location : The Presidency, Old Agra Rd, near Kalika Mandir, Gadkari Chowk, Renuka Nagar, Nashik, Maharashtra 422001

About Organization : Hotel Surbhi Pure Veg, located in Nashik, is a fantastic place for folks who love vegetarian food. It's right in the middle of Nashik, surrounded by beautiful vineyards and a peaceful atmosphere. At Surbhi, we serve only vegetarian dishes made with the freshest ingredients. Whether you're craving Maharashtrian favorites or North Indian delights, we've got something for everyone. Our restaurant has a cozy and welcoming vibe, perfect for spending quality time with family and friends. Whether you're celebrating a special occasion or just want to enjoy a delicious meal, our friendly staff will make sure you have a great experience.

1.2 Abstract :

The restaurant management application, named "Hotel Surbhi," aims to streamline restaurant operations and enhance customer experience by providing a comprehensive solution for both restaurant staff and customers. Developed using the MERN(MongoDB, Express.js, React.js, Node.js) Stack, Hotel Surbhi offers intuitive interfaces for managing orders, inventory, and reservations, while also providing customers with a user-friendly platform for browsing menus, placing orders, and providing feedback.

1.3 Existing System & Need for System :

1.3.1 Existing System :

1.3.1.1 Manual Record-keeping : For Hotel Surbhi Pure Veg, the current record-keeping system heavily relies on manual methods to manage store-related activities such as stock management, accommodation availability, and maintenance records. This manual approach consumes a significant amount of time and effort, making the overall process cumbersome and prone to errors.

1.3.1.2 Lack of Centralized Data : Additionally, the absence of a centralized data system hampers efficient access to crucial information, posing challenges for administrators in managing various aspects of the hotel's operations. From tracking accommodation availability to monitoring stock levels and maintaining maintenance records, the lack of centralized data complicates these tasks.

Administrators encounter difficulties in generating comprehensive reports and analyzing data trends due to the fragmented nature of information storage. This lack of streamlined data management inhibits the ability to make informed, timely decisions crucial for optimizing operational efficiency and enhancing guest satisfaction.

Furthermore, without a centralized system, ensuring data accuracy and consistency becomes a daunting task. Discrepancies in information across different manual records may lead to confusion and errors in decision-making processes.

The implementation of a centralized data management system would not only streamline operations but also facilitate data-driven decision-making, improving overall efficiency and guest experience at Hotel Surbhi Pure Veg.

1.3.1.3 Communication Gap : Moreover, the unavailability of an integrated communication platform creates a communication gap between administrators, clients, and staff members. Important announcements, updates, and notifications may get lost or delayed, resulting in mis-communication and reduced operational transparency.

1.3.1.4 Resource Intensive : Furthermore, the traditional system requires the employment of additional personnel to handle administrative tasks manually. This increases operational costs, adds to the workload, and heightens dependency on manual processes, leading to a higher probability of errors and delays.

1.3.2 Need for the system :

1.3.2.1 Enhanced Operational Efficiency : Implementing a digital system at Hotel Surbhi Pure Veg streamlines various administrative tasks, from staff scheduling and resource management to handling maintenance requests. By digitizing these processes, the system reduces reliance on manual paperwork and human intervention, thereby saving time and effort in day-to-day operations.

Additionally, the digital system at Hotel Surbhi Pure Veg enhances operational efficiency by introducing features such as automated inventory management and online booking systems. With automated inventory management, the system ensures optimal stock levels of ingredients and supplies, minimizing the risk of shortages or overstocking. This not only streamlines procurement processes but also reduces wastage and associated costs.

Moreover, the implementation of an online booking system simplifies the reservation process for guests, allowing them to book accommodations and dining options conveniently from anywhere, at any time. This not only improves customer satisfaction but also reduces the workload on staff members, who can focus on providing personalized service to guests on-site.

1.3.2.2 Centralized Database : The proposed system features a centralized database to store and manage all hotel-related information efficiently. This enables administrators to access real-time data on staff records, room availability, reservation statuses, and maintenance requests, facilitating quicker decision-making and improving overall operational efficiency.

Furthermore, the centralized database implemented in the proposed system for Hotel Surbhi Pure Veg ensures data integrity and security. By consolidating all hotel-related information into a single, centralized repository, the system minimizes the risk of data duplication, inconsistencies, and unauthorized access.

Additionally, the centralized data management system enables seamless integration with other hotel systems and third-party applications, such as online booking platforms and accounting software. This integration enhances interoperability and data sharing, streamlining processes across different departments and external service providers.

1.3.2.3 Seamless Communication : With the digital system in place, Hotel Surbhi Pure Veg can establish seamless communication channels among administrators, guests, and staff members. Through a dedicated platform, announcements, updates, and important notifications can be promptly disseminated, ensuring effective communication and minimizing the risk of miscommunication.

1.3.2.4 Cost-Effectiveness : Transitioning from manual processes to an automated system offers Hotel Surbhi Pure Veg a cost-effective solution. By reducing the need for additional personnel, minimizing paperwork, and optimizing resource allocation, the digital system leads to long-term cost savings and improved financial management for the establishment.

1.4 Scope & Feasibility of work :

1.4.1 Scope of work :

In the context of Hotel Surbhi Pure Veg, the scope of work involves addressing the unique requirements and operational challenges encountered by hotel administrators in efficiently managing various aspects of the establishment. The proposed system will comprise a comprehensive software solution tailored specifically for hotel management purposes.

1.4.1.1 User Authentication and Access Control :

The system will include user authentication features to allow access for clients, administrative staff, and other relevant stakeholders. Each user role will have customized access permissions to ensure secure and controlled information access, safeguarding sensitive data and ensuring operational integrity.

1.4.1.2 Resource Allocation and Management : The system will facilitate the allocation and management of resources. It will provide an interface for administrators to assign customer to specific Products, manage product availability, and handle transfers as needed.

1.4.1.3 Billing and Payment Management : The system will incorporate functionality for managing product fee billing and payment processes. It will generate invoices, track payment records, and send notifications to client regarding payment.

1.4.1.4 Complaints and Maintenance Requests : The system will provide a mechanism for client to submit complaints or maintenance requests related to their services accommodations. Administrators can efficiently manage and track these requests, ensuring timely resolution.

1.4.2 Feasibility of work :

1.4.2.1 Technical Feasibility :

The proposed Hotel Surbhi Pure Veg system will be developed using modern web technologies, specifically MongoDB, Express.js, React.js, and Node.js (MERN stack), along with Tailwind CSS for styling. This tech stack will enable robust front-end and back-end development, ensuring a seamless user experience.

To serve user requests, the system will require a web server such as Express.js, which will handle HTTP requests and responses efficiently. MongoDB will be used as the database to store and manage hotel-related data securely.

The system will be designed to be compatible with popular operating systems, including Android, iOS, Windows, Linux, and macOS, ensuring accessibility across a wide range of devices.

All the necessary hardware and software components required for the development and deployment of the system are readily available in the market, making the implementation technically feasible and ensuring smooth operation of Hotel Surbhi Pure Veg's digital infrastructure.

1.4.2.2 Operational Feasibility :

The proposed system for Hotel Surbhi Pure Veg will present practical solutions to tackle the operational hurdles encountered by administrators. It will feature intuitive and user-friendly interfaces tailored for clients and staff members, ensuring smooth navigation and efficient task execution. By streamlining processes like timetable management, resource tracking, allocation, billing, and complaint handling, the system's operational feasibility rests in its capacity to boost efficiency, simplify operations, and elevate overall management standards at Hotel Surbhi Pure Veg.

1.5 Operational Environment – Hardware & Software :**1.5.1 Client-side System Specification :****1.5.1.1 Hardware :**

Item name	Specification
Laptop / Desktop	Minimum Intel Core i3 or above Minimum RAM: 1 GB or more Minimum Hard disk: 1 GB free space (optional)

1.5.1.2 Software :

Particular	Specification
Operating System	Minimum Windows 7 or above / Minimum Linux 6.1 or above / Minimum Mac OS X 10.1 or above
Browsers	Google chrome 5 or higher / Mozilla Firefox 4 or higher / Microsoft Edge 105.0.1343.27 or higher

1.5.2 Server-Side System Specification :**1.5.2.1 Hardware :**

Item name	Specification
Laptop / Desktop	Intel CORE i7, 8 th Gen RAM: 8 GB SSD: 1 GB Hard disk: 2 TB

1.5.2.2 Software :

Server	Express.JS
Database	MongoDB
Browser	Google Chrome

1.5.3 Developer-Side System Specification :

1.5.3.1 Hardware :

Item name	Specification
Laptop / Desktop	Intel CORE i5, 8th Gen RAM: 4 GB SSD: 128 GB

1.5.3.2 Software :

Particular	Specification
Operating System	Windows 10, Intel CORE i5
Documentation	Microsoft office 2013 or higher
Browser	Google chrome 5 or higher
Text editor	Notepad

1.6 Detail Description of Technology Used :

- **ReactJS(Version: 18)**

React, which was created by Facebook, has quickly emerged as the go-to library for creating smooth, scalable user interfaces. React's component-based architecture and intelligent diffing algorithm make front-end development more efficient as complexity grows.

Today, there are over 12,682,010 live websites using React. Not only that, but industry giants like Apple, Netflix, Paypal, and many others have also already started using React JS in their software productions.

React basically allows developers to utilize individual parts of their application on both the client-side and the server-side, which ultimately boosts the speed of the development process.

In simple terms, different developers can write individual parts and all changes made won't cause the logic of the application.

- **CSS (Version: CSS3)**

It stands for Cascading Style Sheets which is used to make web sites more attractive and beautiful web pages. It is used to design the page and to create animation as well. It used to for describing the presentation of a document written in mark-up language such as HTML or XHTML. It is a corner store technology of the World Wide Web, alongside HTML and JavaScript. CSS was developed by World Wide Web Consortium (W3C). It provides more flexibility and control to the HTML tags and content.

- **Express.JS (4.19.2)**

Express is a node js web application framework that provides broad features for building web and mobile applications. It is used to build a single page, multipage, and hybrid web application.

It's a layer built on the top of the Node js that helps manage servers and routes.

Express was created to make APIs and web applications with ease,

It saves a lot of coding time almost by half and still makes web and mobile applications are efficient.

Another reason for using express is that it is written in javascript as javascript is an easy language even if you don't have a previous

knowledge of any language. Express lets so many new developers enter the field of web development.

- **JavaScript (Version: ES022)**

It is a scripting language. It was developed by Brendan Eich of Netscape. It is one of the most popular languages in 2022. It is used to make web pages more interactive and used to create web application as well as games and mobile application. JavaScript is used to program the behaviour of the web page. All major web browsers have a dedicated JavaScript engine to execute the code on user's device.

- **NodeJS(9.11.2)**

Node.js is composed of Google's V8 JavaScript engine, the libUV platform abstraction layer, and a core library that is written in JavaScript. Additionally, Node.js is based on the open web stack (HTML, CSS, and JS), and operates over the standard port 80.

Node.js provides developers a comprehensive tool for working in the non-blocking, event-driven I/O paradigm. Ryan Dahl, the creator of Node.js was “inspired by applications like Gmail” and—in creating Node.js—aimed to create real-time websites with push capability.

Node.js shines in real-time web applications employing push technology over WebSocket. After over 20 years of stateless-web based on the stateless request-response paradigm, we finally have web applications with real-time, two-way connections, where both the client and server can initiate communication, allowing them to exchange data more freely. This is in stark contrast to the typical web response paradigm, where the client always initiates communication.

- **MongoDB(8.0)**

Most databases force you to use heavy wrappers, like ORMs (Object-Relational Mappers), to get data into Object form for use in programs. MongoDB’s decision to store and represent data in a document format means that you can access it from any language, in data structures that are native to that language (e.g., dictionaries in Python, objects in JavaScript, Maps in Java, etc.).

MongoDB is designed to make data easy to access, and rarely to require joins or transactions, but when you need to do complex querying, it’s more than up to the task.

The MongoDB Query API allows you to query deep into documents, and even perform complex analytics pipelines with just a few lines of declarative code.

MongoDB is designed from the ground up to be a distributed database. Create clusters with real-time replication, and shard large or high-throughput collections across multiple clusters to sustain performance and scale horizontally.

Thanks to the document model used in MongoDB, information can be embedded inside a single document rather than relying on expensive join operations from traditional relational databases. This makes queries much faster, and returns all the necessary information in a single call to the database. If needed, you can perform a left outer join with the \$lookup aggregation pipeline stage, which delivers similar performance to RDBMSs..