**ACKNOWLEDGEMENT**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Topic** | **Page No.** |
| 1 | Company Certificate | 1 |
| 2 | College Certificate | 2 |
| 3 | Progress Report | 3 |
| 4 | Problem Definition |  |
| 5 | Need Of Computerization |  |
| 6 | Scope Of the Proposed System |  |
| 7 | Objective Of the Proposed System - TBD |  |
| 9 | Platform (H/W, S/W, OTE) |  |
| 10 | Design Specification |  |
| 11 | Platform Key Performance Indicators (KPIs |  |
| 12 | Limitations And Drawbacks |  |
| 13 | Future Enhancements |  |
| 14 | Conclusion |  |
| 15 | Reference & Bibliography |  |

Company Certificate

College Certificate

Progress Report

# Problem Definition

1. **Limited Customer Reach:**

* The client's current business model relies solely on physical stores, restricting customer reach to the immediate geographic location.
* This hinders the potential for growth and expansion into new markets.

1. **Inefficient Management of Multiple Stores:**

* Managing separate inventory, customer bases, and sales operations for each physical store is cumbersome and time-consuming.
* This inefficiency can lead to stockouts, discrepancies, and difficulty in tracking overall sales performance.

1. **Lack of Online Presence:**

* The absence of an e-commerce platform prevents the client from capitalizing on the booming online retail market.
* This limits sales opportunities and excludes customers who prefer the convenience of online shopping.

**Need Of Computerization**

* **Limited Scalability:** Manual management of multiple stores with separate inventory and customer bases becomes increasingly difficult as the business grows. A computerized platform like GetiT scales efficiently, handling increased customer traffic and product offerings.
* **Data Management:** Tracking sales, inventory levels, and customer data across physical stores is cumbersome and prone to errors. GetiT provides a centralized database, offering real-time insights and facilitating accurate data analysis for informed decision-making.
* **Inefficient Operations:** Manual communication with customers, order processing, and stock management are time-consuming and error-prone. GetiT automates these tasks, optimizing operations and improving overall efficiency.
* **Limited Customer Reach:** Geographic limitations prevent wider customer acquisition. GetiT expands reach through an online platform, attracting customers beyond the physical stores' locations.
* **Lack of 24/7 Availability:** Traditional stores have limited operating hours. GetiT offers a continuously accessible online platform, allowing customers to browse and shop at their convenience.
* **Data-Driven Marketing:** Without an online platform, understanding customer preferences and behavior is challenging. GetiT provides valuable customer data, enabling targeted marketing campaigns and product offerings based on customer insights.

**Scope** **Of the Proposed System**

This project entails the development and implementation of the GetiT e-commerce marketplace application using a MERN stack architecture. The scope encompasses the following functionalities:

**User Interface:**

* **Product Browsing and Search:** Users can browse products across various categories and utilize search functionality to find specific items.
* **Product Details and Reviews:** Users can view detailed product information, including descriptions, and reviews submitted by other users.
* **Shopping Cart and Checkout:** Users can add items to their carts, manage quantities, and proceed to checkout securely using a Stripe payment gateway integration.
* **Order Tracking and Management:** Users can track order status, view shipment details, and request refunds if necessary.
* **User Profile and Communication:** Users can create profiles, manage account information, and communicate with shop owners through an inbox system.

**Shop Interface:**

* **Shop Setup and Management:** Shop owners can set up their online stores, add product descriptions, images, and manage pricing.
* **Order Management:** Shop owners can view, process, and update order details, including fulfilling orders, managing refunds, and communicating with customers through the inbox system.
* **Inventory Management:** Shop owners can maintain product inventory levels, track stock, and update product availability.
* **Financial Management:** Shop owners can view order history, track earnings, and request withdrawals for admin approval.

**Admin Interface:**

* **User and Shop Management:** Admins can add, remove, and update user and shop information, ensuring platform security and user accountability.
* **Order Management:** Admins can monitor overall order fulfillment, review refund requests, and intervene if necessary.
* **Shop Withdrawal Requests:** Admins can process shop withdrawal requests, ensuring proper financial management within the platform.
* **Data Analytics and Reporting:** Admins can access reports on user activity, shop performance, and sales trends to gain valuable insights for platform optimization.

**Security Considerations:**

* Secure user authentication with JWT tokens.
* Email verification for user registration with a short validity window.
* Payment gateway integration adhering tzo Stripe's security protocols.

**Deliverables:**

* A fully functional e-commerce marketplace application with user, shop, and admin interfaces.
* Comprehensive documentation outlining system functionalities, user guides, and deployment instructions.
* Unit and integration tests to ensure application stability and functionality.

**Objective Of the Proposed System**

1. **Expand Market Reach:** Break geographical limitations by creating an online presence that attracts customers beyond the vicinity of physical stores. This fosters business growth and opens new market opportunities.
2. **Enhance Customer Experience:** Offer a convenient and user-friendly online shopping platform that caters to customers who prefer browsing, purchasing, and managing orders electronically. This improves customer satisfaction and loyalty.
3. **Streamline Business Management:** Centralize inventory management, order processing, and customer interactions into a single platform. This eliminates the inefficiencies associated with managing separate stores and provides a consolidated view of business performance.
4. **Consolidate Customer Base:** Unify customer data and interactions across different product categories, fostering a cohesive brand experience and facilitating targeted marketing campaigns.
5. **Reduce Operational Costs:** The centralized platform potentially reduces operational costs associated with managing individual stores, such as staffing requirements and maintaining separate inventory systems.

By achieving these objectives, GetiT Marketplace platform will create a robust eCommerce marketplace that empowers the client to compete effectively in the online retail landscape.

**Platform (H/W, S/W, OTE)**

**Software Requirement:**

* Operating system: Windows, Mac OS, Linux.
* Web browser: Google Chrome, Mozilla Firefox.
* Front end: React JS (v18.2.0 & later), Material UI.
* Back end: NodeJS (v20.12.2), Mongo Atlas, Cloudinary.

**Hardware requirement:**

* + Hard disk: 500 GB.
  + RAM: 4 GB.
  + Internet Speed: minimum of 200kbps.

**UML Diagrams**

**Activity Diagram-**

**Use Case Diagram-**

**Sequence Diagram-**

**Data Flow Diagram**

**Design Specification**

GetiT E-commerce Marketplace application is based on a Three-Tier Architecture, utilizing a multi-server system built using the MERN stack (MongoDB, Express.js, React.js, Node.js) with Socket.io for real-time communication as below:

**1 Frontend Server**

* Technology Stack: React.js, TailWind CSS, JavaScript
* Functionality:
  + Serves static/dynamic content for UI components using JSX and react hooks.
  + Handles user interactions and routing between pages.
  + Makes API calls to the backend server for data retrieval and manipulation.

**2 Backend Server**

* Technology Stack: Node.js, Express.js, Mongoose (MongoDB ODM), Node Mailer
* Functionality:
  + Implements RESTful APIs for user registration, login, product management, order processing, special events management, shop registration and login, etc.
  + Validates user input and sanitizes data to prevent security vulnerabilities.
  + Leverages Stripe API for seamless payment transaction.
  + Interacts with the MongoDB database using Mongoose models.
  + Sends email notifications using Node Mailer.
  + Handles communication with the Socket.io server for real-time functionalities.

**3 Socket.io Server**

* Technology Stack: Socket.io
* Functionality:
  + Enables real-time communication channels between users and servers.
  + Facilitates features like live chat, order updates, and notifications.
  + Manages connection establishment and message exchange between clients and servers.

**5. Database**

* MongoDB will be used as the primary database for storing application data.
* Mongoose will be used as an object data modeling (ODM) layer for interacting with MongoDB from the backend server.

**6. Security Considerations**

* Enabled email-based user activation and implemented JWT Token for user authentication.
* Sanitized user input to prevent query injection and cross-site scripting (XSS) vulnerabilities.
* Implemented a secure payment gateway using Stripe API.

**7. Scalability**

* Independent horizontal scaling is enabled due to multi-server architecture.
* For the frontend server, Content Delivery Network (CDN) can used to efficiently render static files and improve performance significantly.
* For the backend server, API call traffics can be distributed among multiple backend instances using a load balancer.
* As per business requirement, Mongo Atlas DB can be horizontally scaled for more increased availability or vertically scaled for more storage.

# Platform Key Performance Indicators (KPIs):

**Increased Sales and Revenue:**

* **Expanded Market Reach:** The online platform attracts new customers beyond the limitations of physical stores, leading to a wider customer base and potential for increased sales.
* **24/7 Availability:** Customers can conveniently shop anytime, potentially boosting sales beyond the operational hours of physical stores.

**Enhanced Customer Satisfaction and Loyalty:**

* **User-friendly Shopping Experience:** Easy browsing, purchasing, and order management capabilities improve customer satisfaction.
* **Streamlined Communication:** The platform facilitates clear communication and interaction, potentially fostering stronger customer relationships and loyalty.

**Improved Operational Efficiency and Cost Savings:**

* **Centralized Management:** Inventory, orders, and customer interactions are managed centrally, reducing operational complexity and overhead costs.
* **Consolidated Operations:** Streamlined operations may lead to staffing cost reductions and simplified inventory management.

**Data-Driven Insights and Marketing Strategies:**

* **Real-time Data:** The platform provides valuable customer data and purchasing habits, enabling informed marketing campaigns and data-driven business decisions.
* **Customer Insights:** Understanding customer preferences allows for optimized product selection and promotions, potentially increasing sales.

**Competitive Advantage:**

* **Online Presence:** The platform positions the client as a competitor in the online retail space, attracting customers who primarily shop online.
* **Modern Shopping Experience:** The application offers a modern and convenient shopping experience, keeping pace with current customer expectations.

**Limitations And Drawbacks**

* **Limited Delivery Reach:** Currently, the GetiT platform lacks a dedicated delivery system, restricting door-to-door deliveries to within Mumbai city limits. This significantly hinders the platform's potential to reach customers outside Mumbai, limiting market expansion and sales opportunities.
* **Scalability:** As the user base and product offerings grow, the MERN stack might require scaling considerations. Database optimization, server upgrades, or distributed systems might be necessary to handle increased traffic and data volume.
* **Security Concerns:** Maintaining a secure platform requires ongoing vigilance. Vulnerabilities in user authentication, payment processing, or data storage can lead to security breaches and compromise customer information. Constant security updates and best practices are crucial.
* **Limited Offline Functionality:** The application primarily relies on an internet connection. Limited offline functionality for browsing products or managing orders might be a drawback for users in areas with unreliable internet access

**Future Enhancements**

* **Chatbots and Virtual Assistants:** Introduce chatbots or virtual assistants to provide real-time customer support and answer frequently asked questions.
* **Data Analytics and Business Intelligence:** Develop comprehensive data analytics dashboards to provide valuable insights into customer behaviour, product performance, and overall platform health.
* **Advanced Security Features:** Implement two-factor authentication, multi-level user access control, and fraud detection systems to further strengthen platform security.
* **Hybrid Delivery Model:** GetiT could implement a hybrid approach, utilizing Ekart or other 3PL providers for national deliveries while offering in-house delivery options within Mumbai. This allows for wider reach while maintaining control over local deliveries for a potentially smoother customer experience

**Conclusion**

In Conclusion, the GetiT E-commerce Marketplace application tackles the limitations of the client's physical stores by creating a centralized online platform. This expands reach, improves customer experience, streamlines operations, and offers valuable data insights. The implemented MERN stack architecture with socket integration ensures a secure and scalable foundation.

While considerations like scalability, security, and third-party integrations require attention, the potential benefits are significant. Future enhancements such as chatbot implementation, live support, and advanced security features can further improve user experience. Additionally, integrations with Data / Business analytical dashboards and 3PL for Order fulfilment can empower data-driven decision making and optimize platform performance. By continually innovating, GetiT has the potential to become a thriving online marketplace for the client.

**Reference & Bibliography**

**References:**

* **React 18 Documentation [online] Available at:** [**https://react.dev/reference/react**](https://react.dev/reference/react)
* **Socket.IO Documentation [online] Available at:** **https://socket.io/docs/v4/**
* **Node JS Iron (v20.12.2) Documentation [online] Available at:** [**https://nodejs.org/docs/latest-v20.x/api/index.html**](https://nodejs.org/docs/latest-v20.x/api/index.html)
* **Cloudinary Upload API Documentation [online] Available at:** [**https://cloudinary.com/documentation/image\_upload\_api\_reference**](https://cloudinary.com/documentation/image_upload_api_reference)
* **Mongo Atlas Documentation [online] Available at:** [**https://www.mongodb.com/docs/atlas/**](https://www.mongodb.com/docs/atlas/)
* **Stripe API Documentation [online] Available at:** [**https://docs.stripe.com/api**](https://docs.stripe.com/api)

**Bibliography:**

1. **React 18 Design Patterns and Best Practices - Fourth Edition by Carlos Santana Rold**
2. **Ultimate Full-Stack Web Development with MERN by Nabendu Biswas.**
3. **Mastering MongoDB 7.0: Achieve Data Excellence by Unlocking the Full Potential of MongoDB by Rachelle Palmer.**