International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

A Multi-Vendor Saas website builder with Role-Based CRM

[1]Ajay Varande, [2]Harshad Waghmare, [3]Omkar Londhe, [4]Rohit Rathod, [5]Prof. Sarita Khedikar

[1,2,3]Student, Smt. Indira Gandhi College of Engineering, Ghansoli, New Mumbai, Maharashtra, India
[4]Professor, Dept. of AI & ML, Smt. Indira Gandhi College of Engineering, Ghansoli, New Mumbai, Maharashtra, India

Abstract **-In the modern digital economy, businesses require scalable, customizable, and cost-effective solutions to establish their online presence and manage operations efficiently. Traditional website development and CRM systems often lack flexibility, automation, and integration capabilities. This research proposes a Multi-Vendor SaaS Website Builder and Funnel Management System integrated with Stripe Connect, enabling agencies and businesses to build, manage, and monetize websites and sales funnels efficiently.**

The system supports role-based access, multi-tenancy, real-time dashboards, and subscription-based monetization. It integrates project management tools such as Kanban boards and analytics dashboards, allowing organizations to streamline workflows, track performance, and enhance customer engagement. Built using modern technologies like Next.js 14, Prisma, MySQL, Tailwind CSS, and Bun, the platform ensures scalability, performance, and security.

Keywords: SaaS, Website Builder, CRM, Stripe Connect, Multi-Vendor, Funnel Builder, Role-Based Access

I. INTRODUCTION

With the rapid growth of digital businesses and the increasing shift toward online presence, there is a rising demand for platforms that simplify and streamline the process of building and managing digital products and services. Traditional approaches often require businesses to rely on multiple tools for website development, sales funnel creation, customer relationship management (CRM), and payment processing. This fragmentation leads to inefficiencies, higher operational costs, and challenges in maintaining data consistency across systems. As a result, businesses—especially small to medium-sized enterprises and digital agencies—struggle to manage their workflows effectively.

To address these challenges, this project introduces a unified Software-as-a-Service (SaaS) platform that integrates key business functionalities into a single, cohesive ecosystem. The platform combines a website builder for creating and customizing professional websites, a funnel builder for designing and optimizing sales journeys, a robust CRM system for managing customer interactions and data, and seamless payment integration using Stripe for secure and efficient transactions. By consolidating these features, the platform eliminates the need for multiple third-party tools, thereby improving productivity, reducing costs, and enhancing user experience.

1.1 Project Aims and Objectives

The development of a modern SaaS-based website builder and CRM system requires a clear set of objectives to ensure scalability, usability, and business effectiveness. This project aims to address the limitations of traditional tools by integrating website creation, funnel management, payment processing, and customer relationship management into a single unified platform.

Objectives and Aims:

1. To develop a scalable SaaS platform for website and funnel creation
2. To enable multi-vendor support with agency and sub-account hierarchy
3. To integrate Stripe Connect for automated payment distribution
4. To provide role-based access control for secure operations.
5. To implement real-time analytics and dashboards.
6. To streamline project management using Kanban boards
7. To support subscription-based and add-on monetization models
8. To ensure high performance and scalability using modern frameworks like Next.js and optimized backend architecture
9. To enable seamless media management and storage for websites and funnels
10. To provide customize dashboard for agencies and sub-account with real time insights.
11. To implement a secure and extensible secure architecture that support third party .

1.2 System Objectives

The system is designed with a strong focus on scalability, modularity, performance, security, and usability to meet the evolving demands of modern digital businesses. It ensures high scalability by supporting thousands of users and organizations simultaneously without performance degradation. The modular architecture allows easy extension and integration of new features such as funnels, CRM

functionalities, and payment systems. Performance optimization is achieved through modern technologies like Next.js 14, enabling fast rendering and efficient data handling. Security is a key priority, with robust authentication mechanisms and secure payment processing to protect user data and transactions. Additionally, the system emphasizes usability by providing a clean, intuitive, and responsive user interface built using Tailwind CSS and ShadCN, ensuring a seamless user experience across all devices.

Furthermore, the system is designed to maintain high availability and reliability by ensuring minimal downtime and consistent service delivery. It incorporates efficient database management practices using Prisma and MySQL to handle large volumes of data with integrity and speed. The architecture also supports real-time data updates and synchronization, enabling users to access the latest information instantly across dashboards and modules.

In addition, the platform is built with extensibility and integration capabilities in mind, allowing seamless connectivity with third-party services such as payment gateways, analytics tools, and marketing platforms. This flexibility ensures that the system can adapt to future technological advancements and business requirements. Overall, the system objectives aim to deliver a robust, scalable, and future-ready SaaS solution that enhances productivity, simplifies operations, and improves the overall user experience.

1.3 Background of Project

In recent years, the rapid growth of digital businesses and online services has significantly increased the demand for efficient website development, customer management, and payment processing systems. Traditional approaches often require multiple separate tools for website building, marketing, CRM, and payment handling, which leads to increased complexity, higher costs, and inefficient workflows. Businesses, especially small agencies and startups, struggle to manage these fragmented systems while maintaining scalability and performance.

Existing solutions such as standalone website builders, CRM platforms, and payment gateways lack deep integration and flexibility. These systems often do not support multi-vendor architectures or role-based access, making it difficult for agencies to manage multiple clients under a single platform. Additionally, the absence of centralized dashboards and real-time analytics limits decision-making capabilities and reduces operational efficiency. As a result, there is a growing need for a unified platform that can integrate all essential business functionalities into one system.

To address these challenges, this project proposes the development of a Multi-Vendor SaaS Website Builder integrated with funnel management, CRM, and Stripe-based payment systems. The platform aims to provide a centralized, scalable, and user-friendly solution that enables agencies and businesses to create websites, manage customers, process payments, and analyze performance within a single ecosystem. By leveraging modern technologies and modular architecture, the system enhances productivity, reduces operational overhead, and supports future scalability and innovation.

II. COMPONENTS

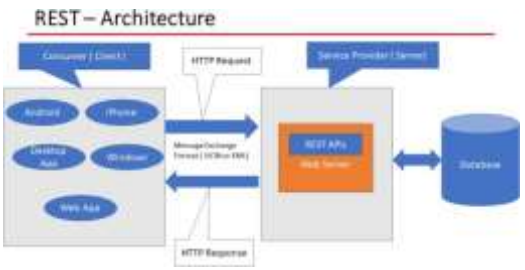
The proposed Multi-Vendor SaaS Website Builder system is composed of several software and system components that work together to deliver a scalable, secure, and high-performance platform. Each component plays a critical role in ensuring efficient system functionality, data management, and user experience.

2.1 Software components for processing the system

i) Frontend Layer**:**

The frontend layer is responsible for delivering an interactive and user-friendly interface for the SaaS platform. It is developed using modern technologies such as Next.js 14, which enables fast rendering and efficient routing through the App Router architecture. The user interface is styled using Tailwind CSS and ShadCN UI, ensuring a clean and responsive design across various devices.

ii) Backend Layer

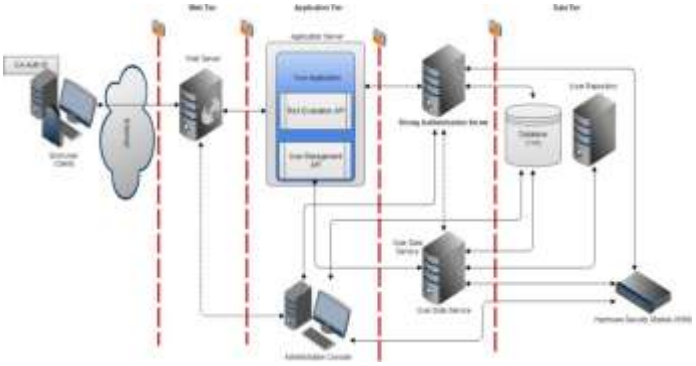


The backend layer acts as the core processing unit of the system, managing all business logic and server-side operations. It is built using the Bun runtime, which provides high performance and efficient execution. This layer handles authentication, authorization, API requests, and communication between the frontend and the database. It ensures secure processing of user data and system operations while maintaining scalability. Additionally, it supports modular development, making it easier to integrate new features such as CRM functionalities, funnel management, and subscription systems without affecting existing components.

iii) Database Layer

The database system is responsible for storing and managing all application data in a structured and efficient manner. It utilizes MySQL as the relational database, combined with Prisma ORM for simplified database interactions and schema management. The system stores critical data such as user profiles, agencies, sub-accounts, funnels, transactions, and subscriptions. It ensures data integrity, consistency, and fast query performance, even with large datasets. The database is designed to support scalability and real-time data access, enabling seamless synchronization across different modules of the application.

iv) Authentication and Authorization Module



The authentication and authorization module ensures secure access control within the system. It implements role-based access control (RBAC), allowing different levels of access for Admins, Agencies, and Sub-account users. This module manages user login, session handling, and permission validation to prevent unauthorized access. It protects sensitive data and ensures that users can only perform actions permitted by their roles. By incorporating secure authentication mechanisms, the system maintains data privacy and enhances overall platform security.

v) CRM and Project Management Module

The CRM and project management module is designed to manage customer relationships and streamline workflow processes. It provides tools for lead tracking, customer data management, and communication handling within the platform. The inclusion of a Kanban board allows users to organize tasks, monitor progress, and improve team collaboration. This module centralizes all customer-related data, making it easier to manage interactions and improve service delivery. It enhances productivity by integrating project management and CRM functionalities into a single unified system.

III. METHODOLOGY

The development of the proposed Multi-Vendor SaaS Website Builder and CRM system follows a structured and systematic methodology to ensure efficiency, scalability, and reliability. The methodology includes multiple stages such as requirement analysis, system design, development, testing, deployment, and maintenance. Each phase is carefully planned to ensure that the final system meets user requirements and performs optimally in real-world scenarios.

1. Requirement Analysis: The first step involves gathering and analyzing system requirements from users, businesses, and stakeholders. This includes identifying

key features such as website building, funnel management, CRM functionalities, and payment integration. User roles such as Admin, Agency, and Sub-account users are defined along with their permissions. Functional and non-functional requirements such as scalability, security, and performance are also analyzed. This phase ensures a clear understanding of system expectations before development begins..

2. Design: In this phase, the overall system architecture is designed, including frontend, backend, and database structures. Database schemas are created using Prisma ORM to define relationships between entities such as users, agencies, funnels, and transactions. UI/UX wireframes are designed for dashboards, website builders, and CRM modules. API structures and data flow between components are also planned. This stage provides a blueprint for the development process.
3. Development: The development phase involves implementing the system using modern technologies such as Next.js 14 for frontend, Bun for backend, and MySQL with Prisma for database management. Features such as multi-vendor support, role-based access, funnel builder, CRM system, and Stripe integration are developed. Modular coding practices are followed to ensure scalability and maintainability. This phase converts design concepts into a fully functional application..
4. Database Integration: This phase focuses on integrating the database with the backend system to enable data storage and retrieval. MySQL is used to manage structured data, while Prisma ORM simplifies database operations. Data models are implemented for users, subscriptions, funnels, and transactions. Proper indexing and optimization techniques are applied to ensure efficient query performance. This stage ensures seamless data flow across the application.
5. Testing: Testing is conducted to ensure the system functions correctly and meets quality standards. Various testing methods such as unit testing, integration testing, and system testing are performed. Payment flows using Stripe are tested to ensure secure and accurate transactions. Performance testing is also conducted to evaluate system scalability and responsiveness. This phase helps identify and fix bugs before deployment.
6. Deployment: In this stage, the system is deployed to a production environment where users can access it. Cloud platforms such as Vercel or VPS hosting are used for deployment. Continuous Integration and Continuous Deployment (CI/CD) pipelines may be implemented for smooth updates. The database is hosted securely, and the application is configured for real-time access. This phase ensures that the system is available and operational for end users.
7. Maintenance: The final phase involves maintaining and updating the system to ensure long-term performance and reliability. Regular bug fixes, feature enhancements, and security updates are implemented. System performance is continuously monitored to identify potential issues. User feedback is also considered to improve functionality and usability. This phase ensures that the platform remains up-to-date and adaptable to future requirements.

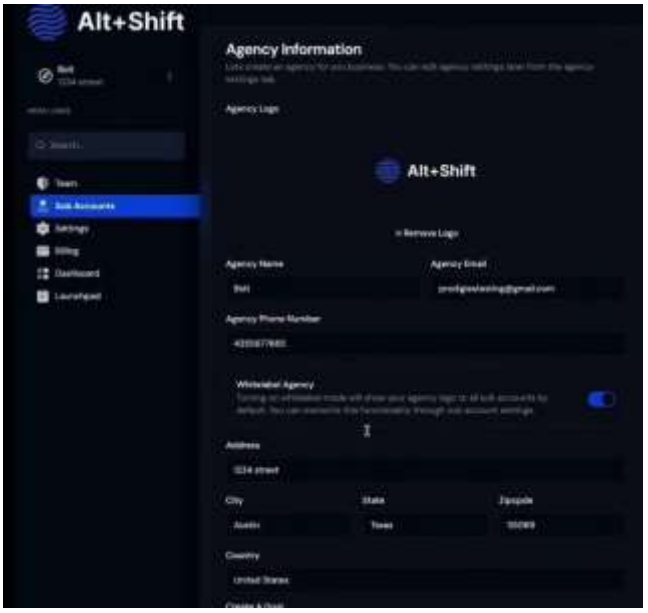
IV. RESULT

The result of a parking management system can include optimized space utilization, reduced congestion, improved revenue collection, enhanced security, and better overall user experience for both drivers and parking facility operators.

4.1 User, Agency, and Sub-Account Structure

The platform follows a hierarchical structure consisting of users, agencies, and sub-accounts. A user refers to any individual who accesses the system, with permissions varying based on their role. At the top level, an agency represents the main organizational entity responsible for managing operations and overseeing multiple projects or clients.

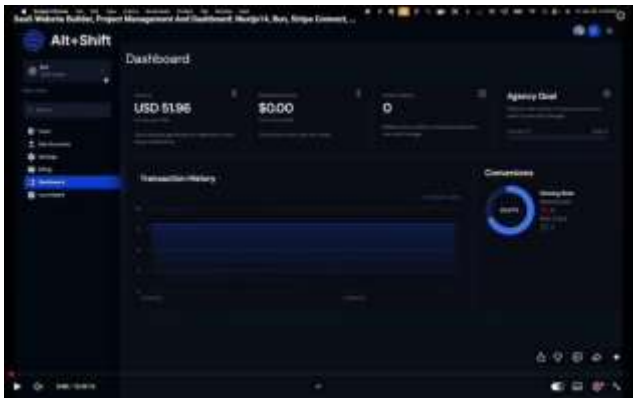
Sub-accounts exist under the agency and function as dedicated workspaces for specific clients or projects. Each sub-account operates independently, maintaining its own funnels, websites, and contact data. This structure enables efficient organization, clear separation of work, and scalable management across multiple clients.



4.2 Login & Dashboard Access

Users access the platform by logging in with their registered email and password. Upon successful authentication, they are directed to the Agency Dashboard, which serves as the central interface for managing activities and navigating the system. From this dashboard, users can easily switch between existing sub-accounts or create new ones based on their requirements.

Access and control within the platform are governed by role-based permissions. Different roles such as admin, editor, and viewer determine the level of authority a user has, including the ability to modify settings, manage content, or simply view information. This ensures secure and structured access management across the platform.



4.3 Website Builder (Drag and Drop Editor)

The Website Builder is a visual drag-and-drop editor designed to enable users to create complete websites either from pre-built templates or from scratch. It provides an intuitive interface where users can easily design pages such as Home, About, Services, and Contact without requiring coding skills.

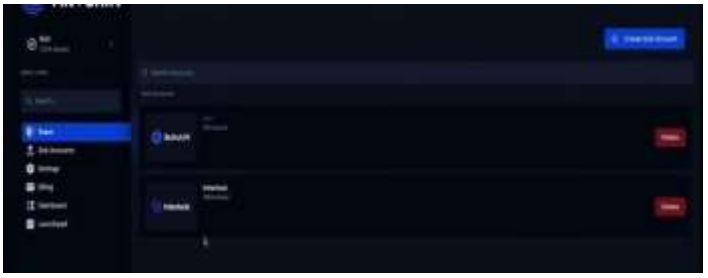
The builder also supports style customization, SEO field configuration, and asset uploads, allowing users to create visually appealing and optimized websites.

The typical workflow involves navigating to Funnels & Websites > Websites and selecting the “Create Website” option. Users can then add pages and incorporate various components such as images, text, and videos through the drag-and-drop interface. Additionally, buttons and forms can be connected to funnels or CRM systems for enhanced functionality. Once completed, the website can be published to a subdomain, enabling immediate access and deployment.

4.4 CRM (Customer Relationship Management)

The CRM (Customer Relationship Management) system is designed to store, organize, and manage leads, prospects, and customers in a centralized manner. It provides detailed contact profiles that include essential information such as email addresses, phone numbers, and tags for segmentation. Additionally, it maintains a complete communication history, covering emails, SMS, and calls, along with features for managing tasks and scheduling appointments.

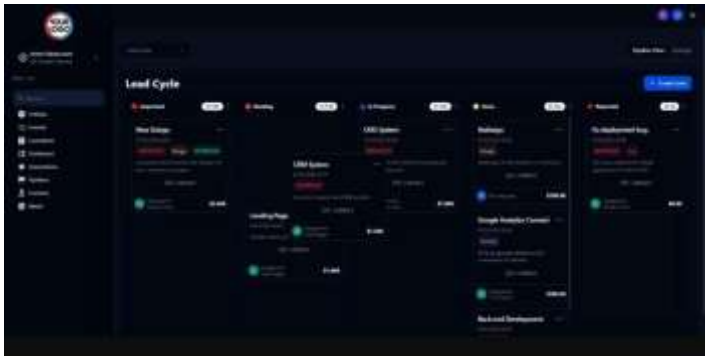
The workflow begins when a contact is added to the system, either through form submissions, manual entry, or data import. Users can then access and update contact details, assign tags, and add notes for better tracking. Furthermore, automated workflows can be triggered based on specific actions, enabling seamless communication through email or SMS and improving overall customer engagement.



4.5 Kanban Board

Pipelines and the Kanban Board are used to visually manage the progression of leads or deals through different stages of the sales process. Each pipeline represents a structured sales funnel, where stages such as New, Contacted, Demo Booked, and Closed are displayed as columns. Within these columns, individual cards represent specific contacts or deals, providing a clear and organized view of ongoing activities.

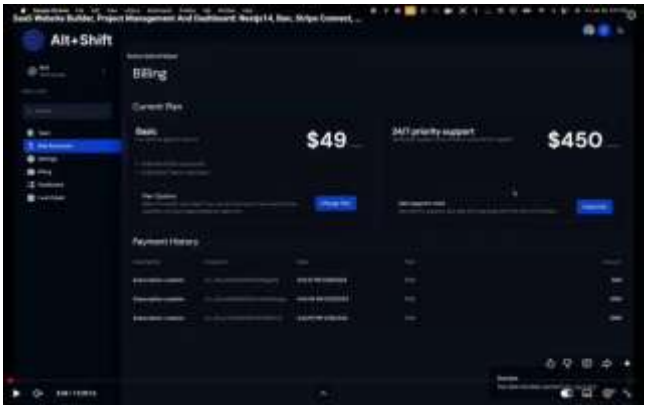
The workflow begins by creating a deal either manually or directly from the CRM and assigning it to a pipeline. Users can then move deals across stages by simply dragging and dropping cards between columns, reflecting real-time progress. Additionally, automated actions such as sending emails or updating records can be triggered when a deal transitions from one stage to another, enhancing efficiency and ensuring timely follow-ups.



4.6 Stripe Integration & Payments

The Stripe integration enables the platform to accept payments seamlessly through funnels and websites. It allows businesses to securely process transactions while linking payment data directly to their system. By integrating with Stripe, users can manage products, pricing, and customer payments within a unified workflow, enhancing both efficiency and user experience.

The process begins with an admin connecting their Stripe account using an OAuth-based authentication flow. Once connected, users can add a Checkout component within funnels or websites using the drag-and-drop builder. They can then configure product details, pricing, and success or failure URLs. When a visitor completes a payment, the transaction is processed via Stripe, and the corresponding customer data is automatically captured and stored in the CRM for further tracking and engagement.



V. Conclusion

In conclusion, the system successfully leverages Python's versatility and its extensive ecosystem of libraries to build a comprehensive, efficient, and user-friendly platform. By integrating multiple functionalities such as website building, funnel creation, CRM management, pipeline tracking, and payment processing into a single interface, it significantly reduces operational complexity for both administrators and users.

The intuitive design ensures that even non-technical users can navigate and utilize the system effectively, thereby improving productivity and workflow management.

Moreover, the modular architecture of the platform plays a crucial role in ensuring scalability and adaptability. It allows organizations to customize features according to their specific business requirements while also supporting future enhancements without disrupting existing functionalities. This flexibility makes the system suitable for a wide range of applications, from small businesses to large enterprises.

Overall, the platform provides a complete and integrated solution that supports the entire business lifecycle, from lead generation and customer engagement to sales execution and growth, making it a reliable and future-ready technological framework.

V. FUTURE SCOPE

1. **Advanced AI Integration**: Incorporating AI-driven features such as automated funnel optimization, smart lead scoring, and personalized user experiences to improve conversion rates and decision-making.

2. **Enhanced Analytics & Insights**: Expanding dashboards with predictive analytics, deeper funnel performance tracking, and real-time business intelligence for agencies and sub-accounts.

3. **Third-Party Integrations Expansion**: Supporting integrations with more platforms such as marketing tools, email services, CRM systems, and payment gateways beyond Stripe to increase flexibility.

4. Automation & Workflow Builder: Introducing advanced automation workflows for lead nurturing, email/SMS campaigns, task assignments, and pipeline updates to reduce manual effort.

ACKNOWLEDGEMENT

As every project is ever complete with the guidance of experts. So we would like to take this opportunity to thank all those individuals who have contributed in visualizing this project. We express our deepest gratitude to our project guide Prof. Sarita Khdeikar (CSE(AIML) Department, Smt. Indira Gandhi College of Engineering, University of Mumbai) for her valuable guidance, moral support and devotion bestowed on us throughout our work. We would also take this opportunity to thank our project coordinator Prof. Tularam Bansod Sir for his guidance in selecting this project and also for providing us all the details on proper presentation of this project. We extend our sincere appreciation to our HOD ma'am and entire professors from Smt. Indira Gandhi College of Engineering for their valuable inside and tip during the designing the project. Their contributions have been valuable in many ways that we find it difficult to acknowledge them individually.

REFERENCES

[1] Armbrust, Michael, et al. "A view of cloud computing." Communications of the ACM 53.4 (2010): 50-58.

[2] Chong, Frederick, and Gianpaolo Carraro. "Architecture strategies for catching the long tail." Microsoft Developer Network (MSDN) (2006).

[3] Bezemer, Cor-Paul, and Arie Zaidman. "Multi-tenant SaaS applications: maintenance dream or nightmare?" Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE). ACM, 2010.

[4] Pautasso, Cesare, Olaf Zimmermann, and Frank Leymann. "RESTful web services vs. big web services: making the right architectural decision." Proceedings of the 17th international conference on World Wide Web. 2008.

[5] Fielding, Roy T. "Architectural styles and the design of network-based software architectures." Doctoral dissertation, University of California, Irvine, 2000.

[6] Zhang, Qi, Mohamed Chen, Li Li, and Zhen Xiao. "An analysis of multi-tenant SaaS architecture." IEEE International Conference on Cloud Computing. IEEE, 2010.

[7] Li, Zhen, and Minyi Zhou. "Cloud-based SaaS platform for business applications." 2013 IEEE International Conference on Service Operations and Logistics, and Informatics. IEEE, 2013.

[8] Subramanian, Nachiappan, et al. "Customer relationship management systems: a study." International Journal of Information Management 34.3 (2014): 344-349.

[9] Kumar, V., and Werner Reinartz. "Customer relationship management: concept, strategy, and tools." Springer, 2018.

[10] Stripe Inc. "Stripe Connect documentation and payment integration guide." Stripe Official Documentation, 2023.

[11] Vercel Inc. "Next.js deployment and subdomain architecture." Vercel Documentation, 2023.

[12] React Team. "Next.js 14 and modern web development practices." Official React and Next.js Documentation, 2026.

[13] Anderson, David J. "Kanban: successful evolutionary change for your technology business." Blue Hole Press, 2010.

AUTHORS BIOGRAPHY

Rohit Rathod,
Pursuing Final year in B.E. CSE (AI&ML) at Smt. Indira Gandhi College of Engineering, Ghansoli, New Mumbai, Maharashtra, India.

Omkar Londhe,
Pursuing Final year in B.E. CSE (AI&ML) at Smt. Indira Gandhi College of Engineering, Ghansoli, New Mumbai, Maharashtra, India.

Ajay Varande ,
Pursuing Final year in B.E. CSE (AI&ML) at Smt. Indira Gandhi College of Engineering, Ghansoli, New Mumbai, Maharashtra, India.

Harshad Waghmare,
Pursuing Final year in B.E. CSE (AI&ML) at Smt. Indira Gandhi College of Engineering, Ghansoli, New Mumbai, Maharashtra, India

Citation of this Article:

Rohit Rathod, Omkar Londhe, Ajay Varande, Harshad Waghmare, Prof. Sarita Khedikar, "A Multi-Vendor Saas website builder with Role-Based CRM", Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 8, Issue 4, pp 87-91, APRIL 2026 . Article DOI https://doi.org/10.47001/IRJIET/2026.804012

****