e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:02/February-2025 Impact Factor- 8.187 www.irjmets.com

YUPFORM: A DYNAMIC FORM BUILDER

Pathan Faizkhan*1, Prof. Aparajita Biswal*2

*1Student, Department Of Computer Science & Engineering, Parul Institute Of Engineering & Technology, Vadodara, Gujarat, India.

*2Asst. Professor, Department Of Computer Science & Engineering, Parul Institute Of Engineering & Technology, Vadodara, Gujarat, India.

DOI : https://www.doi.org/10.56726/IRJMETS67223

ABSTRACT

YupForm is an online form builder with the simple but robust feature to make web forms in any customized format. No programming skills are needed. An easy-to-use drag-and-drop interface enables rapid building of custom-designed forms that serve one's specific purposes for any surveys, registration forms, or collecting payments. YupForm can connect easily to Stripe and PayPal accounts to provide hassle-free transaction handling. Google reCAPTCHA's anti-spam features, built right into YupForm, prevent malicious spam from invading one's website. YupForm also provides webhook integration. Developers can connect forms with any application for automation from external applications. With such ease of use and flexibility, YupForm makes data collection easier and easier workflows.

Keywords: No-Code Forms, Payment Integration, Workflow Automation, Webhook Support, Data Collection, User-Friendly Interface.

I. INTRODUCTION

In the digital era, businesses and individuals rely heavily on online forms for data collection, user feedback, surveys, and customer interactions. Traditional form builders often come with limitations such as high costs, slow performance, and restricted customization options. YupForm is a modern, scalable, and developer-friendly form-building solution designed to address these challenges.

Built with Next.js for the frontend, Svelte for widget integration, and Node.js with MySQL for backend processing, YupForm ensures fast performance, seamless integration, and efficient data management. The system incorporates Redis caching and a queue-based background processing system, making it capable of handling high-volume submissions without performance bottlenecks.

This paper explores the architecture, implementation, and advantages of YupForm, highlighting how it provides a cost-effective, high-performance, and scalable alternative to traditional form-building platforms.

II. METHODOLOGY

The architecture of YupForm is designed for high performance, scalability, and seamless user experience. The system is composed of multiple components that work together to handle form submissions efficiently while ensuring data integrity and responsiveness.

Frontend (Client-Side)

The frontend of YupForm consists of two major parts:

1. **Main Application (yupform.com):** This is the primary web interface where users can create, manage, and analyze their forms. It is built using Next.js, a React-based framework optimized for performance, server-side rendering (SSR), and SEO benefits. Next.js enhances the overall user experience by reducing load times and making the platform more responsive.
2. **Widget (form.yupform.com):** This widget is embedded on external websites, allowing users to collect form submissions directly from their own platforms. The widget is built using Svelte, a lightweight frontend framework known for its fast performance and minimal runtime overhead. Svelte compiles components into highly efficient JavaScript code, ensuring the widget remains lightweight and does not slow down the host website.

Both components communicate with the backend via RESTful APIs, sending and retrieving data in real time.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:02/February-2025 Impact Factor- 8.187 www.irjmets.com

Backend (Server-Side Processing)

The backend is built using Node.js, a non-blocking, event-driven runtime well-suited for handling a high volume of requests efficiently. It acts as the core of the application, processing form submissions, managing authentication, and facilitating communication between various services.

Key backend responsibilities include:

- Handling user authentication and authorization.
- Processing form submissions and storing data securely.
- Managing payment transactions via Stripe and PayPal integrations.
- Implementing security measures such as reCAPTCHA to prevent spam.

Data Handling & Caching

To optimize data retrieval and reduce database load, Redis is used as a caching layer. Redis stores frequently accessed data in memory, significantly improving response times.

- When a form submission request is received, the backend first checks Redis for existing data to avoid unnecessary database queries.
- If the required data is not found in Redis, it queries the MySQL database, stores the result in Redis for future access, and then returns the response to the client.
- This caching mechanism ensures that form submissions are processed quickly, even under high traffic.

Asynchronous Processing (Queue & Worker System)

For operations that require additional processing time, such as sending email notifications, webhooks, or handling large datasets, a Queue System is implemented.

- When a task is submitted, it is placed into a queue for background processing.
- A Backend Worker continuously processes tasks from the queue, ensuring that time-consuming operations do not delay the user experience.
- This system improves scalability and ensures the platform remains responsive even when handling thousands of submissions.

III. EXISTING SYSTEM AND PROPOSED SYSTEM

Existing Systems:

In the current landscape of online form builders, many platforms offer user-friendly interfaces for creating and managing forms. However, these existing systems often come with limitations such as high subscription costs, limited customization options, and performance issues when handling large volumes of data. Additionally, integrating these forms into diverse web applications can be challenging due to compatibility issues and rigid architectures.

Proposed System:

To address these challenges, we propose YupForm, a modern form-building solution designed for scalability, efficiency, and seamless integration. YupForm leverages a technology stack comprising Next.js for the client interface, Svelte for widget integration, and Node.js with MySQL for backend processing. This architecture ensures fast performance and flexibility. Furthermore, by incorporating Redis caching and a queue-based background processing system, YupForm can handle high-volume form submissions without compromising performance. Its API-driven design and customizable widgets facilitate easy embedding into various web applications, providing a cost-effective and developer-friendly alternative to traditional form builders.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:07/Issue:02/February-2025 Impact Factor- 8.187 www.irjmets.com

Block Diagram:

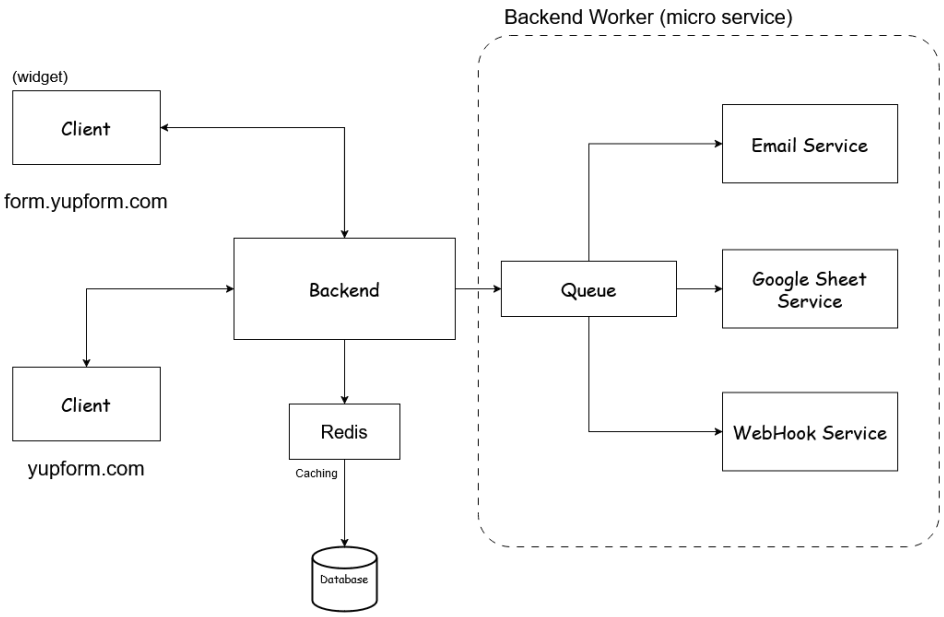

Figure 1: Block Diagram of YupForm

IV. USE CASES AND IMPACT

YupForm serves a wide range of applications, making form creation seamless and efficient. Businesses can use it for contact and lead generation forms, embedding widgets on their websites to collect inquiries and integrate with CRM systems for automatic lead tracking. Event organizers can create registration forms for conferences and webinars, with payment integration for seamless ticket purchases. It also enables surveys and feedback collection, allowing businesses and researchers to design customized surveys with conditional logic and store responses for analysis. E-commerce stores can use it for order forms with built-in payment processing and automated email notifications. In HR and recruitment, YupForm helps collect job applications with file upload support for resumes and portfolios. Internal business workflows such as employee feedback, leave requests, and IT support tickets become more efficient with seamless database integration. Additionally, healthcare providers can facilitate appointment bookings, sending automated reminders to patients, while educational institutions can use it for student registrations, assignment submissions, and multi-step course enrollments.

The impact of YupForm is significant across industries. Performance and efficiency are enhanced through Next.js and Svelte, ensuring faster form loading and interaction, while Redis caching reduces response times. The system is highly scalable, utilizing a queue and worker system to handle large-scale submissions without slowdowns, making it ideal for enterprises with heavy data needs. User experience is modern and mobile-friendly, with customization options that allow businesses to maintain brand consistency. Operational costs are reduced compared to traditional form builders, offering a cost-effective model with self-hosted open-source options. Security and data privacy are prioritized with Google reCAPTCHA to prevent spam and a secure MySQL database with Redis caching to ensure data integrity. Seamless third-party integrations allow businesses to connect with CRMs, email marketing platforms, and payment gateways, with webhook support for real-time data syncing. Finally, development and deployment are faster, as developers can easily embed forms using simple widgets while backend automation streamlines form handling and processing.

V. CONCLUSION

YupForm presents a modern, efficient, and scalable solution for online form creation and data management. By leveraging Next.js for the client interface, Svelte for widget integration, and Node.js with MySQL for backend processing, it ensures optimal performance and flexibility. The inclusion of Redis caching and a queue-based backend worker system enhances responsiveness and scalability, making it capable of handling high-volume submissions without performance bottlenecks.Compared to traditional form builders, YupForm offers a cost-effective, customizable, and developer-friendly alternative that can be integrated seamlessly into various

business applications, from lead generation and surveys to e-commerce orders and appointment scheduling. Its modern tech stack ensures fast load times, enhanced user experience, and improved data security, catering to a wide range of industries.

ACKNOWLEDGEMENTS

I sincerely thank **Prof. Aparajita Biswal** for her valuable guidance and extend my gratitude to the **Ascendtis IT Solutions LLP** team for allowing me to be part of the development team and contribute to this project.

VI. REFERENCES

[1] R. Kumar, S. Sharma, and P. Gupta, "Enhancing Web Form Automation using Next.js and Svelte for Scalable Applications," International Journal of Computer Science and Software Engineering, Vol. 25, Issue 4, 2023, pp. 102-110.

[2] A. Patel and V. Srinivasan, "Optimizing Form Handling with Node.js and MySQL: A Performance Analysis," Journal of Software Engineering and Development, Vol. 18, Issue 3, 2022, pp. 89-97.

[3] J. Lee, H. Kim, and T. Park, "Efficient Data Storage and Retrieval Mechanisms in Web Applications using MySQL and Redis," International Journal of Data Science and Applications, Vol. 20, No. 2, 2021, pp. 134-145.

[4] M. Ramesh and K. Varun, "A Comparative Study on Traditional vs. Modern Web Form Builders: Performance, Security, and Scalability," International Journal of Web Technologies, Vol. 15, Issue 1, 2020, pp. 50-62.

[5] L. Zhang and Y. Chen, "Leveraging Serverless Architectures for High-Performance Web Applications," IEEE Transactions on Cloud Computing, Vol. 19, Issue 6, 2022, pp. 211-220.

[6] S. Thomas and R. Banerjee, "Security and Privacy Concerns in Online Form Submissions: A Case Study," International Journal of Cybersecurity and Networks, Vol. 22, No. 4, 2021, pp. 175-183.