

Zaprint – A Remote Printing Management System

**Soham Joshi¹, Rushil Ekhande², Ibrahim Kazi³, Tanmay Naik⁴,
Shubhangi Chintawar⁵**

^{1,2,3,4}Student, Department of Computer Engineering, V.E.S. Polytechnic, Mumbai

⁵Professor, Department of Computer Engineering, V.E.S. Polytechnic, Mumbai

Abstract

The development of web-based services has also contributed to the modernization of document handling and printing processes. The traditional printing process is often limited by manual coordination, the need for physical presence, and the tightly coupled backend infrastructure. This paper introduces Zaprint, a remote printing management system that allows users to submit print jobs via a web application while giving local print shops the ability to manage and process these print jobs via a desktop application. The proposed system does not require the use of a backend server, as it directly connects both client applications to Supabase, which handles authentication, database management, file storage, and communication services. A cloud-based data layer is used to connect users and print shops in real-time. The system workflow, architecture, and approach are fully discussed. The findings show that the system is more efficient and easier to use compared to traditional print management systems.

Keywords: Remote Printing, Supabase, Web Application, Desktop Application, Realtime Systems

Introduction

The printing service industry has been dependent on traditional interactions between users and printing service centers. The transfer of documents has been done through removable storage media or email, followed by manual confirmation and status updates. These processes have been prone to delays, a lack of transparency, and increased operational complexity. With the rapid pace of digital transformation, the need for remote and automated printing services has increased in the educational, office, and local service sectors.

Traditional printing services are restricted by fragmented communication channels, the unavailability of real-time status updates, and the need for centralized backend servers that require constant maintenance. These factors have often led to increased deployment costs and lower scalability. The driving force behind digital remote printing services comes from the need to simplify the process of ordering, provide transparency into status, and facilitate efficient coordination between users and printing service providers. The main aim of the proposed system is to design and test a remote printing management system that works without the use of a backend server. The system will focus on using cloud technology to offer authentication, data storage, document storage, and real-time updates in a single platform.

Literature Review

The traditional systems used in print shops are mostly manual or semi-automatic, involving human interaction and simple record-keeping systems. Such systems are not integrated with web technology and therefore do not allow real-time tracking and remote access.

The existing online printing services mostly use backend servers to handle user authentication, file uploading, and order processing. Although these systems make it easier for users to access printing services, they also make it complicated to set up, scale, and maintain. The software used in print management in an enterprise setup is mostly concerned with controlling access and tracking expenses, rather than user-friendly remote ordering.

Some research works have pointed out the shortcomings of the existing systems, including high infrastructure costs, lack of flexibility, and delays in synchronizing the system components.

Proposed System

Zaprint is envisioned as a remote printing management system that aims to make printing easier. The system will have a web application for users and a desktop application for print shops. Both applications will communicate directly with Supabase, which will serve as the single cloud service provider for all applications.

The idea of a backend-less architecture is implemented by delegating common server-side tasks to cloud services. Authentication, database management, file storage, and realtime messaging will be handled by Supabase, removing the need for backend development.

The employment of Supabase in the system brings many benefits, including simplified development, inherent security features, and native support for realtime data synchronization.

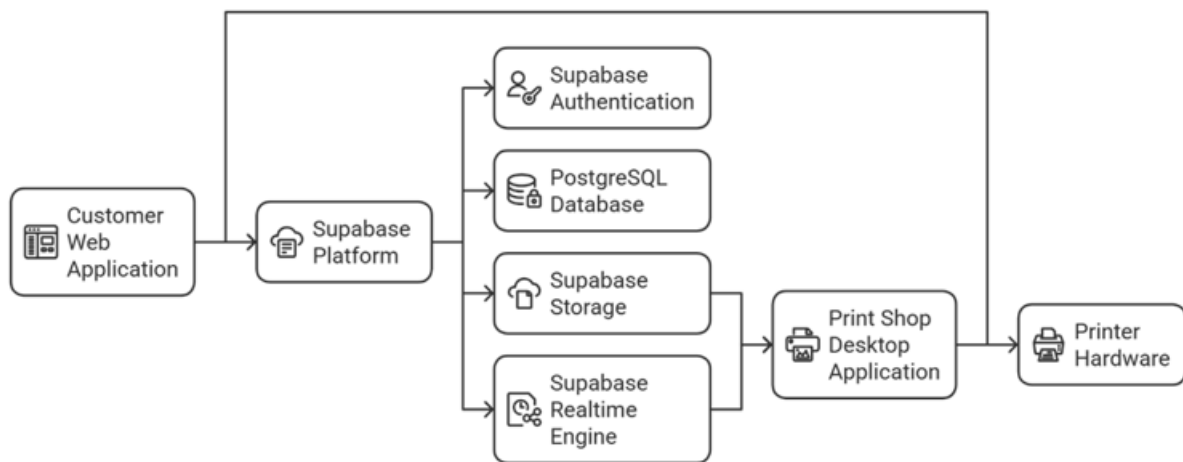


Figure 1: System Architecture Diagram

System Architecture

The system architecture consists of three main components: the web application module, the desktop application module, and the Supabase services.

The web application module handles user authentication, document upload, and order submission. The users access this module via a browser-based interface, and all actions are carried out directly on the Supabase services.

The desktop application module is intended for local print shops and is developed as a standalone application. This module subscribes to the Supabase realtime updates, enabling print shops to receive new orders, update order status, and control print jobs effectively.

The Supabase services are the backbone of the system architecture. PostgreSQL is utilized for storing structured data, cloud storage is utilized for document files, authentication services handle user and shop identities, and Supabase realtime channels facilitate the immediate dissemination of data among applications.

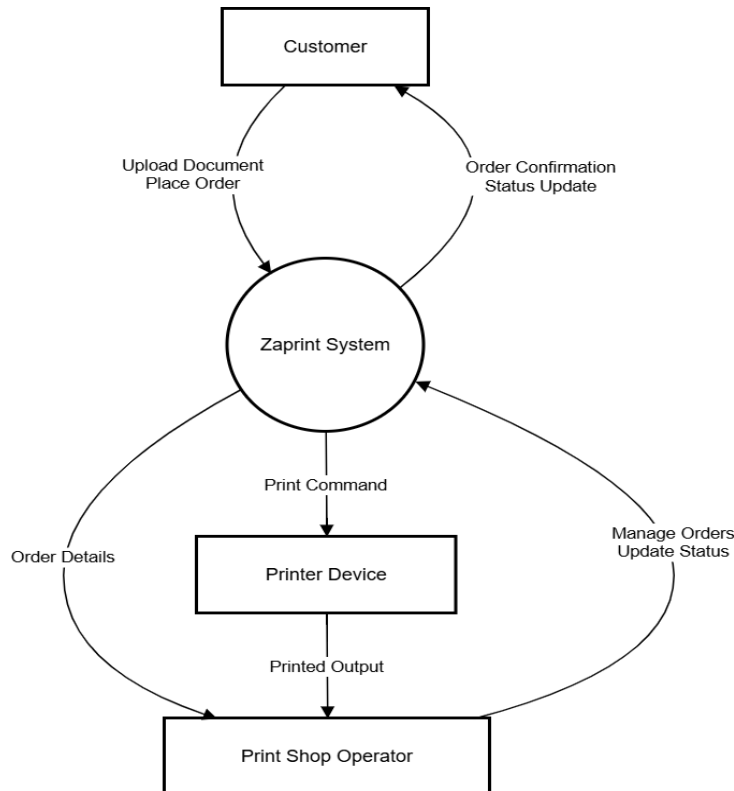


Figure 2: Component Diagram

Methodology

The workflow of the system starts with a user uploading a document via the web app. The document file is saved in cloud storage, and the order metadata is saved in the database. After a successful upload, the realtime notifications are sent.

The desktop app is always listening for realtime updates about new or changed orders. When a print shop gets a new order, it gets processed, and the order status is updated in the database. This change is immediately reflected back to the user via the realtime channels.

The data flow between the components is handled completely via Supabase services. This helps in maintaining consistency and reducing latency. This eliminates the need for servers in between.

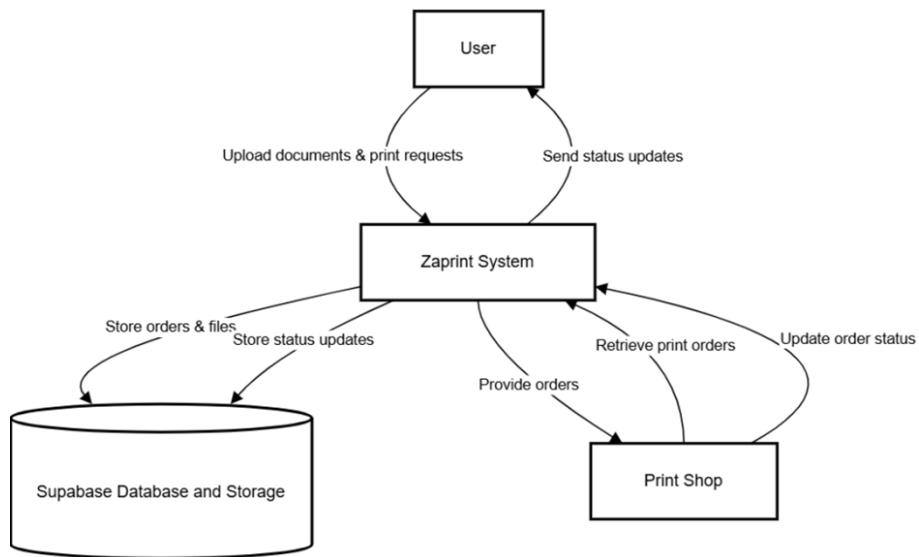


Figure 3: Data Flow Diagram (DFD) – Level 0

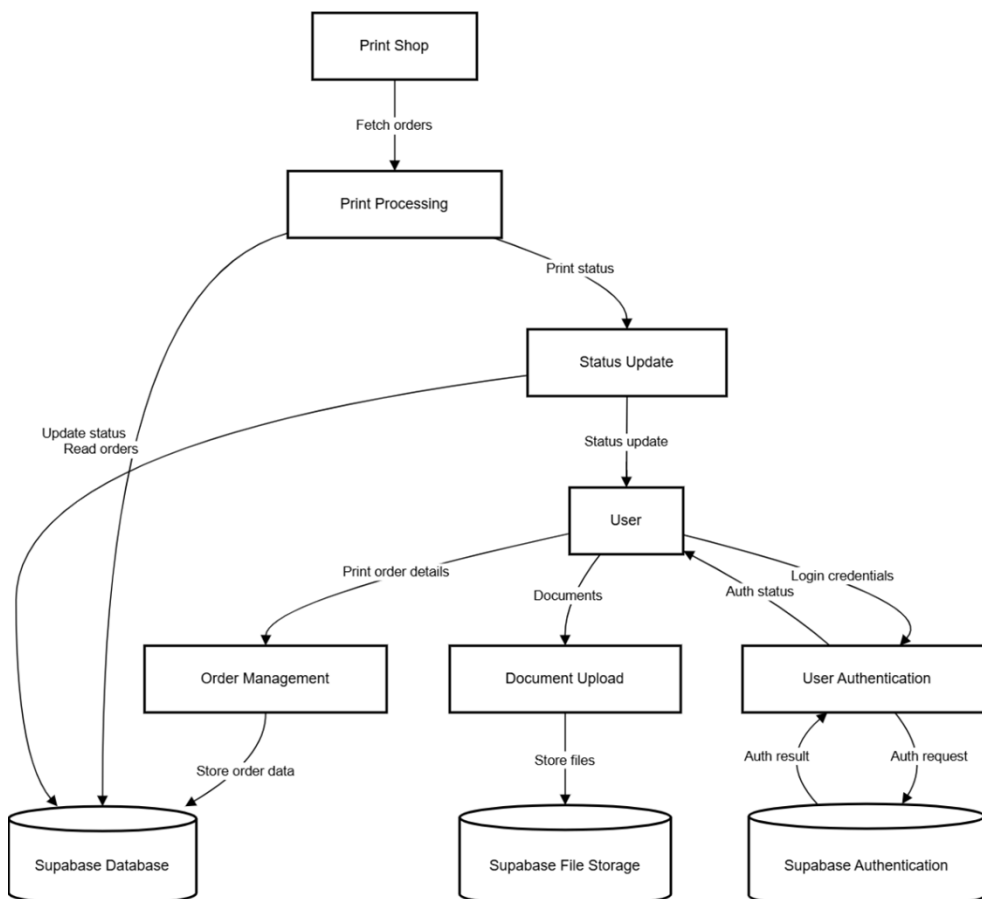


Figure 4: Data Flow Diagram (DFD) – Level 1

Implementation Details

Authentication is done through the use of managed identity services, which provide a secure way for both users and print shops to access the system. Role-based access control is also implemented to limit the operations performed by users depending on their roles.

Storage of documents is done using cloud object storage, which provides a secure way for uploading and downloading documents. Access controls are also implemented to restrict unauthorized downloads. Real-time communication is done through database change subscriptions, which provide real-time updates of order status changes. Security measures include encrypting data during transmission, controlling access to storage, and implementing strict authentication policies.

Results and Discussion

The observed system performance demonstrates low latency in order synchronization and status updates. Realtime communication ensures that users and print shops remain consistently informed without manual refresh mechanisms.

Usability outcomes indicate improved user satisfaction due to simplified order placement and transparent tracking. Print shops benefit from centralized order management and reduced administrative effort.

When compared to traditional methods, the proposed system reduces infrastructure complexity, eliminates backend maintenance overhead, and enhances operational efficiency.

Applications

The system is applicable in educational institutions where students require quick and remote access to printing services. Local print shops can adopt the system to streamline order management and expand digital reach. Offices and organizations can utilize the platform to manage internal printing workflows efficiently.

Limitations and Future Enhancements

The current system relies on continuous internet connectivity, which may limit usability in low-connectivity environments. File size constraints imposed by cloud storage services may also affect large document handling.

Future enhancements may include offline job queuing, advanced analytics for print shops, and integration with payment gateways to support end-to-end transaction management.

Results and Application

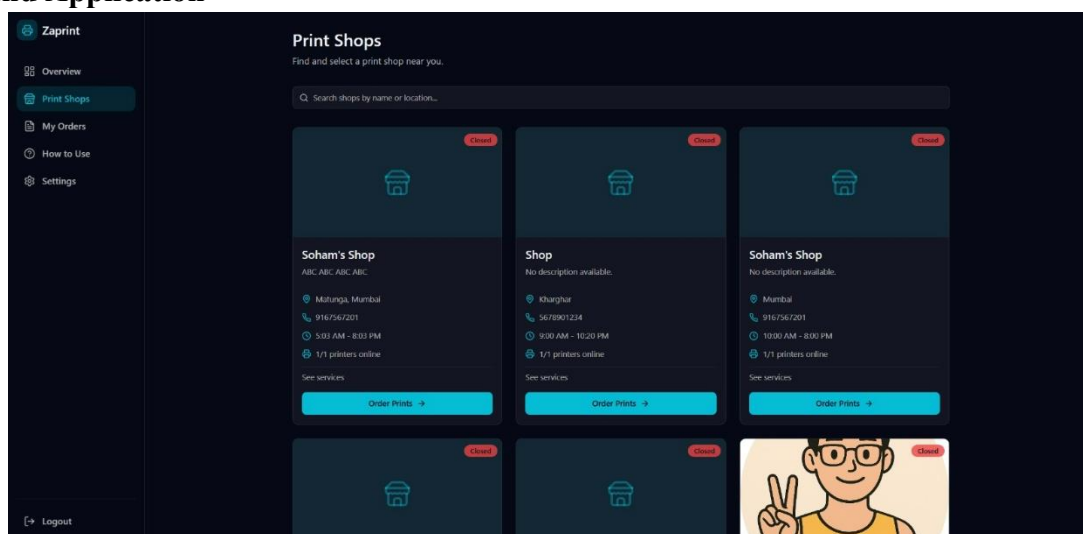


Figure 5: Print Shop Selection Interface in Web Application

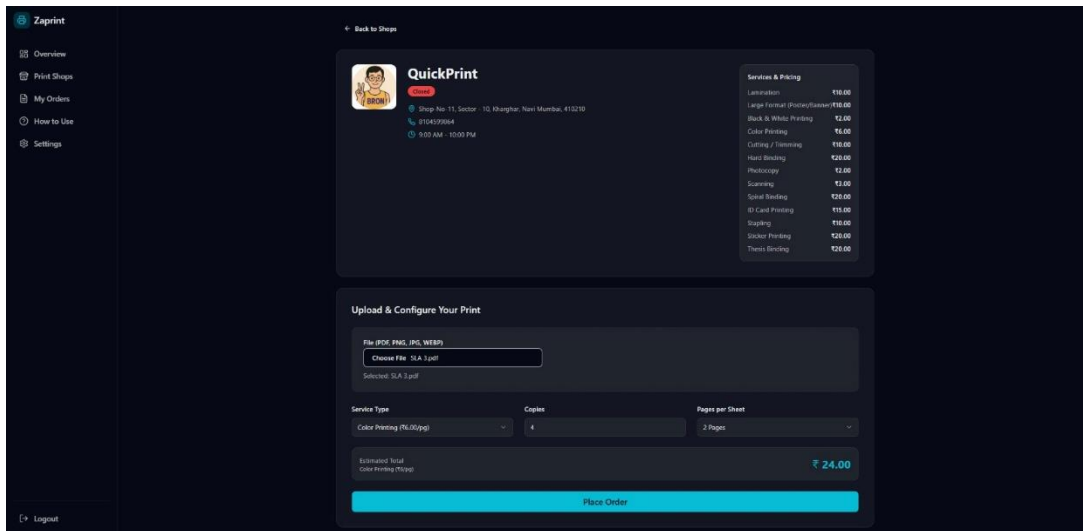


Figure 6: Document Upload and Print Configuration Interface

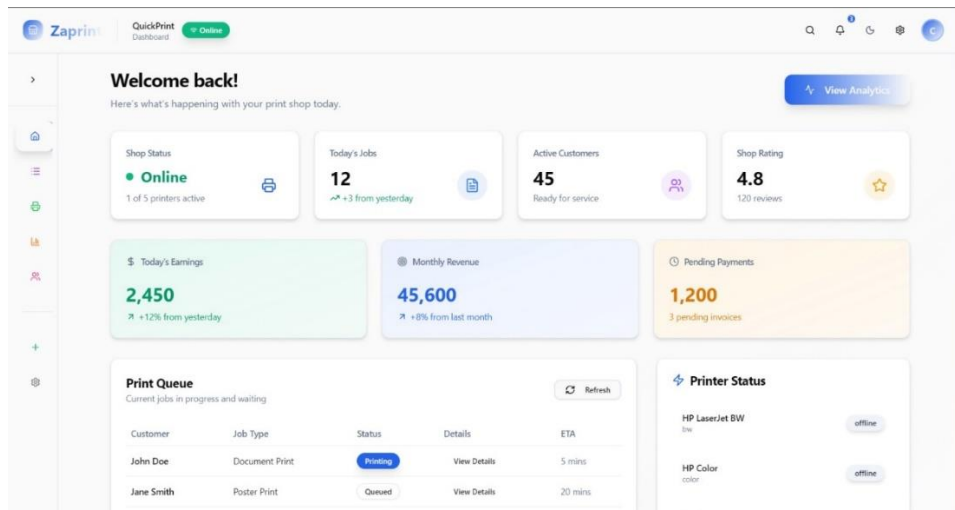


Figure 7: Print Shop Desktop Dashboard Overview

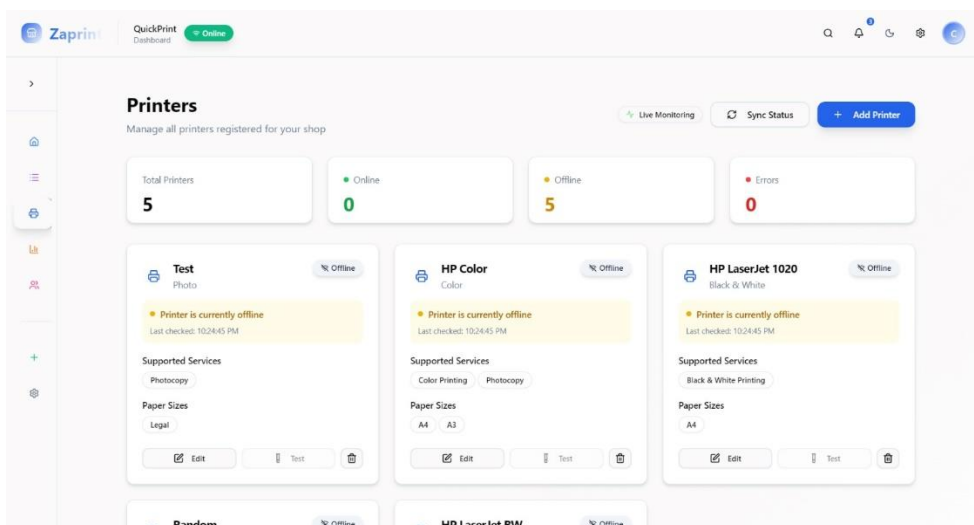


Figure 8: Registered and Connected Printers Interface

Conclusion

This paper presented Zaprint, a backend-less remote printing management system that leverages cloud-based services to eliminate traditional server dependencies. The proposed architecture demonstrates how modern managed platforms can be used to build scalable and efficient systems with minimal infrastructure overhead. The results confirm that the system improves usability, transparency, and operational efficiency. The study concludes that backend-less architectures are a viable and effective approach for modern print management solutions.

Acknowledgement

The authors would like to acknowledge the support and guidance received during the development and evaluation of this research work.

References

1. Srinivas Vallabhaneni, "Serverless Architecture in Modern Distributed Systems: From FaaS to Production Patterns," *International Journal of Information Technology and Management Information Systems*, vol. 16, no. 2, pp. 683–695, 2025.
2. Swati Karni, "Serverless & Event-Driven Architectures: Redefining Distributed System Design," *Emerging Frontiers Library for The American Journal of Interdisciplinary Innovations and Research*, vol. 7, no. 10, pp. 13–22, 2025.
3. Joel Lopes and Ceres Dbritto, "Implementing Serverless Computing Architectures for Expandable and Cost-Effective Cloud Applications," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 72, no. 8, pp. 171–179, 2024.
4. Kumar Avinash, "Architectural Approaches to Scaling Distributed Microservice Systems in the Cloud," *Emerging Frontiers Library for The American Journal of Engineering and Technology*, vol. 7, no. 10, pp. 90–98, 2025.
5. Dileep Kumar Siripurapu, "Real-Time Data Synchronization in Distributed Systems: A Comprehensive Analysis of Architectures, Strategies, and Implementation Technologies," *International Journal of Research in Computer Applications and Information Technology*, vol. 8, no. 1, pp. 3412–3424, 2025.
6. Naresh Vurukonda and B. Thirumala Rao, "A Study on Data Storage Security Issues in Cloud Computing," *Procedia Computer Science*, vol. 79, pp. 335–343, 2016.
7. M. B. Jayalekshmi and S. H. Krishnaveni, "A Study of Data Storage Security Issues in Cloud Computing," *Indian Journal of Science and Technology*, vol. 8, no. 24, pp. 1–5, 2015.
8. Vaibhav Vudayagiri, "Demystifying Serverless Architecture for Scalable Web Applications," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 10, no. 6, pp. 254–263, 2024.
9. Real-Time Database Synchronization Techniques in Firebase for Mobile App Development, *International Journal of Scientific Research in Engineering and Management*, vol. 07, no. 12, pp. 1–10, 2025.
10. Dileep Kumar Siripurapu, "Real-Time Synchronization in Distributed Systems: Analysis and Implementation Technologies," *International Journal of Research in Computer Applications and IT*, vol. 8, no. 1, pp. 3412–3424, 2025.