

Solo Suite: A Modern Platform for Freelancers and Clients

Kashyapa Datta

Computer Science Engineer (MAHE)
Manipal Institute of Technology (MAHE)
Hyderabad, India
dhondukashyap@gmail.com

Manas Goel

Data Science Engineer (MAHE)
Manipal Institute of Technology (MAHE)
Jamshedpur, India
manasgoel3011@gmail.com

Abstract—Solo Suite is a web-based AI powered platform designed to connect clients with freelance service providers and solo entrepreneurs. . This document outlines the project's introduction, design, implementation, and future scope. The platform is built using a modern technology stack, including Next.js/React.js for the frontend and Supabase for backend services, with a strong emphasis on user experience AI tools and a rich feature set including portfolio management, a job marketplace, and direct communication channels.

Index Terms—Agentic AI, freelance marketplace, Next.js, Supabase, AI tools, React, full-stack, web application, user experience

I. INTRODUCTION

The gig economy has seen tremendous growth, with platforms like Fiverr becoming central to how freelancers find work and how clients hire talent. However, many existing platforms can be improved in terms of user interface, Productivity tools and modern technologies like Agentic AI, user experience, and feature sets. Solo Suite aims to address these gaps by providing a visually appealing, intuitive, and feature-rich alternative for freelancers and clients. The project was initiated to build a robust platform from the ground up, focusing on modern web development practices. This focuses on strong inter-college/ inter-university community enhancement and the empowerment of solo entrepreneurs.

II. RELATED WORK

Solo Suite builds upon the concepts pioneered by successful freelance marketplaces.

- **Fiverr:** Known for its "gig-based" model where freelancers offer services for a fixed price. Solo Suite adopts a similar model but with more flexibility.

Solo Suite differentiates itself by combining a modern, aesthetically pleasing user interface with powerful features from the start, such as an integrated AI assistant for various tasks and a seamless client-provider workflow.

This work was developed as part of a project to build a modern freelance marketplace.

III. PROPOSED SOLUTION / METHODOLOGY

The platform is architected as a full-stack application using the following technologies:

- **Frontend:** Next.js, React, TypeScript, Tailwind CSS
- **Backend:** Supabase (PostgreSQL, Authentication, Storage)
- **Deployment:** Vercel

The methodology follows an iterative development process, adding features incrementally based on user feedback and project goals. The core components of the proposed solution are:

- **User Authentication:** Secure sign-up and sign-in for clients and providers.
- **Portfolio Management:** A rich portfolio system for providers to showcase their work.
- **Dashboards:** Separate, tailored dashboards for clients and providers to manage their activities.
- **Job Marketplace:** A central hub for clients to post jobs and for providers to apply.

IV. IMPLEMENTATION

This section details the implementation of key features developed so far.

A. Portfolio System

The portfolio system allows providers to create and manage a portfolio of their work. Clients can view these portfolios to assess the provider's skills.

- **UI/UX Enhancements:** The portfolio display on the client dashboard was redesigned for a more modern and clean look, featuring gradient avatars, improved spacing, and hover effects. The provider's name, fetched from a separate 'profiles' table, is now displayed prominently.
- **Detailed Portfolio View:** A "View Full Portfolio" button was added to each portfolio card on the client dashboard. This leads to a dedicated page (`/portfolio/[id]`) that shows a comprehensive view of the provider's profile and all their portfolio items.
- **Provider's Perspective:** A "View Public Profile" feature was implemented on the provider's dashboard, allowing them to see their portfolio as a client would.

B. Job Marketplace (In-Progress)

The foundation for a job marketplace has been laid.

- **Database Schema:** `jobs` and `job_applications` tables were designed and created in Supabase to store job postings and applications.
- **Client-Side:** The "Post a Job" button on the client dashboard is the entry point for this feature (implementation pending).
- **Provider-Side:** A "View Current Jobs" button has been added to the provider dashboard, which will link to the job browsing interface.

V. RESULTS & EVALUATION

While quantitative user testing has not been conducted, the iterative development has yielded significant qualitative improvements:

- The UI redesign of the portfolio section has resulted in a more professional and engaging user experience.
- The separation of a summary card and a detailed portfolio view improves information hierarchy and user flow.
- The codebase has been refactored to be more modular and maintainable, for example, by joining tables in Supabase queries to fetch related data efficiently.

VI. FUTURE SCOPE

The following features are planned for future development:

- **Complete the Job Marketplace:** Build out the UI for posting jobs, browsing jobs, and managing applications.
- **Real-time Notifications & Messaging:** Implement a system for real-time communication between clients and providers.
- **Invoice Management:** A feature for providers to generate and send invoices to clients, and for clients to view and pay them.
- **AI-Powered Assistance:** Further integration of AI to help with tasks like writing job descriptions or summarizing applicant profiles.

VII. CONCLUSION

Solo Suite is on a promising trajectory to become a competitive player in the freelance marketplace domain. By focusing on a superior user experience and a well-thought-out feature set, it aims to provide significant value to both freelancers and clients. The development process has been agile, allowing for rapid implementation and refinement of features.

ACKNOWLEDGMENT

The authors would like to thank the open-source community for the tools and libraries that made this project possible.

REFERENCES

REFERENCES

- [1] Next.js by Vercel - The React Framework for the Web. (n.d.). Retrieved from <https://nextjs.org/>
- [2] Supabase - The Open Source Firebase Alternative. (n.d.). Retrieved from <https://supabase.com/>
- [3] React - A JavaScript library for building user interfaces. (n.d.). Retrieved from <https://reactjs.org/>
- [4] Tailwind CSS - A utility-first CSS framework for rapid UI development. (n.d.). Retrieved from <https://tailwindcss.com/>
- [5] Fiverr - Freelance Services Marketplace. (n.d.). Retrieved from <https://www.fiverr.com/>