IJSRA

International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

(Research Article)

Check for updates

# Exploring full stack development and DevOps Integration: A case study on 'skill-swap' web application

Karan Shah, Nizamuddin Khan, Anupama Jawale * and Ruta Prabhu

*Department of Information Technology, Narsee Monjee College of Commerce and Economics, Vile Parle, Mumba, India.*

International Journal of Science and Research Archive, 2025, 15(01), 045-053

Publication history: Received on 24 February 2025; revised on 01 April 2025; accepted on 03 April 2025

Article DOI: https://doi.org/10.30574/ijsra.2025.15.1.0948

## Abstract

The creation of the "Skill Swap" platform is motivated by the growing need for affordable, accessible, and flexible learning in the digital era. Platforms such as Coursera, Udemy, and LinkedIn Learning, though popular, have some drawbacks that Skill Swap seeks to overcome. One of the main obstacles that learners on these platforms encounter is the cost of accessing courses and learning materials. Most platforms charge subscription fees or one-off payments, and learning is thus out of reach for those with limited budgets. Skill Swap breaks this cost barrier by creating a community-based learning environment where users can swap skills directly with each other, without the requirement of money. A second problem learners face on current platforms is the absence of tailored learning experiences. Coursera, Udemy, and other websites tend to follow set course outlines that do not adjust to individual learning styles, pace, and objectives. Skill Swap solves this problem by using a user-oriented design that enables learners to customize their learning experiences based on their individual needs and interests. The website's sophisticated algorithms pair users according to their skills, experience, and learning goals, promoting more effective and efficient learning relationships.

**Keywords:** Education; Full Stack; Learning; Skills; Web

## 1. Introduction

Most online learning sites have limited space for peer-to-peer interaction and collaborative learning. Students tend to view content in a passive manner without the possibility of actively collaborating with peers. Skill Swap bridges this gap by having real-time communication tools and video calling capabilities, allowing effortless peer interaction and facilitating a collaborative learning environment. The users can participate in discussions, request feedback, and offer assistance to each other, which boosts the learning process.

Trust and responsibility are also issues on such sites as Udemy, where just about anyone can develop and host courses without adequate scrutiny of their competence or the quality of their materials. Such a lack of quality control can result in discrepancies in course content and erode students' trust in the site. Skill Swap resolves this issue by having a full skill registration system in place and skill testing system. The skills of the users are authenticated, and their competence is proven, so that the learners can rely on the skill of their learning peers. Skill Swap's own peer-to-peer learning approach combined with its focus on accessibility, personalization, collaboration, and quality distinguishes it from the platforms such as Coursera, Udemy, and LinkedIn Learning. Utilizing the latest technologies and following a user-oriented approach, Skill Swap addresses a major gap in the online learning space today. The ability of the platform to democratize access to information, creates an active community of learners and experts. In conclusion, Skill Swap's theoretical underpinning is grounded on the realization of the inadequacies and challenges experienced by learners on current online learning platforms. Through eliminating the financial constraints, lack of personalization, no peer interaction, and trust and accountability problems that exist on such platforms as Coursera, Udemy, and LinkedIn

* Corresponding author: Anupama Jawale

Learning, Skill Swap seeks to transform how people gain and transfer skills in today's digital world. Skill-Swap 's main objectives are as follows:

- Facilitate Skill Exchange: Develop a user-friendly platform where individuals can exchange skills in various fields, such as programming, languages, sports, and more, through a peer-to-peer learning model.
- Develop a Matching Algorithm: Build an intelligent algorithm to match users based on the skills they offer and seek, improving the overall effectiveness of the skill exchange process.
- Integrate Real-Time Communication: Implement a messaging system to facilitate real-time communication between users for coordinating skill swaps.
- Adopt DevOps Practices for Deployment: Implement CI/CD pipelines using Jenkins or similar tools to automate testing, integration, and deployment, ensuring the application is always up-to-date and running smoothly.

## 2. Literature Review

### 2.1. Available Technologies

*2.1.1. Frontend*

- JavaScript: The primary programming language for creating interactive and dynamic web experiences.
- React: A popular JavaScript library for building user interfaces, known for its component-based architecture and virtual DOM.
- Angular: A comprehensive JavaScript framework for building large-scale web applications, offering features like dependency injection and two-way data binding.
- Vue.js: A progressive JavaScript framework for building user interfaces, known for its simplicity and flexibility.

*2.1.2. Backend*

- Node.js: A JavaScript runtime built on Chrome's V8 JavaScript engine, allowing developers to use JavaScript for server-side programming.
- Express.js: A minimal and flexible web application framework for Node.js, providing a robust set of features for web and mobile applications.
- Django: A high-level Python web framework that encourages rapid development and clean, pragmatic design.
- Ruby on Rails: A server-side web application framework written in Ruby, known for its convention over configuration approach and developer productivity.

*2.1.3. Database*

- MySQL: A widely used open-source relational database management system known for its reliability, performance, and ease of use.
- PostgreSQL: A powerful, open-source object-relational database system with a strong reputation for reliability, feature robustness, and performance.
- MongoDB: A popular NoSQL database that uses a document-oriented model, providing high scalability and flexibility.
- Redis: An open-source, in-memory data structure store, used as a database, cache, and message broker.

*2.1.4. Chat and Video Call*

- Socket.IO: A JavaScript library that enables real-time, bidirectional, and event-based communication between the browser and the server, commonly used for building chat applications.
- WebRTC: A free and open-source project that provides web browsers and mobile applications with real-time communication capabilities via simple APIs, enabling video and audio communication.
- Twilio: A cloud communications platform that provides APIs for voice, video, and messaging, allowing developers to embed communication features into their applications.

#### *2.1.5. DevOps*

Testing

- Jest: A JavaScript testing framework developed by Facebook, known for its simplicity, performance, and snapshot testing capabilities.
- Mocha: A flexible JavaScript testing framework that runs on Node.js and in the browser, providing a rich set of features for testing asynchronous code.
- Cypress: A modern, open-source testing framework for web applications, offering an intuitive API and powerful debugging capabilities.
- Selenium: A portable framework for testing web applications, allowing developers to write tests in various programming languages and run them across different browsers. **Continuous Integration and Continuous Deployment (CI/CD):**
- Jenkins: An open-source automation server that supports CI/CD, providing hundreds of plugins to support building, deploying, and automating projects.
- GitLab CI/CD: A built-in CI/CD solution within GitLab, allowing developers to define and run pipelines directly from the repository.
- CircleCI: A cloud-based CI/CD platform that automates build, test, and deployment processes, supporting a wide range of languages and frameworks.
- Travis CI: A hosted CI service used to build and test projects hosted on GitHub and Bitbucket, supporting a variety of languages and frameworks.

#### *2.1.6. Monitoring and Logging*

- Prometheus: An open-source systems monitoring and alerting toolkit, collecting metrics from configured targets at given intervals, evaluating rule expressions, and displaying results.
- Grafana: An open-source platform for data visualization, monitoring, and analysis, providing a powerful and flexible way to create, explore, and share dashboards.
- ELK Stack (Elasticsearch, Logstash, Kibana): A collection of open-source tools for collecting, storing, analyzing, and visualizing log data.
- Splunk: A platform for searching, monitoring, and analysing machine-generated big data, offering real- time insights through graphs, reports, and alerts. Table 1 presents comparative analysis of available technologies in chosen area.

**Table 1** Comparative Analysis of Technologies in Chosen Area

| **Next.js (React)** | **Vue.js** |
|---|---|
| Next.js provides built-in server-side rendering (SSR) and static site generation (SSG) out of the box. This enables better performance, SEO, and initial page load times [1]. | Vue.js supports SSR through additional libraries like Nuxt.js, but it requires extra configuration and setup [2]. |
| Next.js offers automatic code splitting, optimized bundling, and efficient client-side navigation. It also supports incremental static regeneration (ISR) for improved performance and faster updates [3]. | Vue.js has a smaller bundle size and faster runtime performance compared to React. However, it lacks some of the advanced performance optimizations provided by Next.js[3]. |
| Next.js benefits from the extensive React ecosystem, which offers a wide range of libraries, tools, and community support. This allows developers to leverage existing React components and resources. | Vue.js has a growing ecosystem with a variety of plugins, tools, and libraries. However, it may not be as extensive as the React ecosystem [4]. |
| Next.js provides a streamlined development experience with features like automatic code splitting, hot module replacement (HMR), and a built-in development server. It also integrates well with popular React development tools [5]. | Vue.js offers a simple and intuitive development experience with its single-file components and a clear separation of concerns. It has its own set of development tools and CLI for easy project setup. |
| Next.js builds on top of React, so developers need to be familiar with React concepts and JSX syntax. However, | Vue.js has a gentler learning curve compared to React. Its template syntax is more familiar to developers |

| | |
|---|---|
| Next.js abstracts away many complexities and provides a structured approach to building applications. | with HTML and CSS background. Vue.js also provides clear documentation and a progressive learning path. |
| MongoDB | MySQL |
| MongoDB uses a flexible, document-oriented data model. It stores data in JSON-like documents called BSON (Binary JSON). This allows for more natural and intuitive representation of complex data structures [6]. | MySQL uses a structured, table-based data model. It enforces a rigid schema with predefined tables, columns, and relationships. This requires careful planning and can be less flexible for evolving data requirements. |
| MongoDB's document-based model allows for flexible and dynamic schemas. Fields can vary from document to document, and the structure can evolve over time without requiring schema migrations. This flexibility is beneficial for agile development and accommodating changing requirements. | MySQL enforces a rigid schema, requiring predefined tables and columns. Altering the schema often involves modifying the table structure and migrating existing data, which can be timeconsuming and require downtime [7]. |
| MongoDB's document-oriented approach and support for embedded documents and arrays can lead to improved performance for certain use cases. It can retrieve related data in a single query, reducing the need for complex joins. MongoDB also supports indexing, including geospatial and text indexes, for fast query performance [7]. | MySQL is known for its strong performance for structured data and complex queries. It uses a mature query optimizer and supports indexes for efficient data retrieval. However, complex joins and transactions can impact performance, especially for large datasets. |
| MongoDB is well-suited for use cases that involve unstructured or semi-structured data, such as content management systems, real-time analytics, mobile apps, and IoT applications. Its flexibility and scalability make it a good fit for rapidly evolving and high-volume data scenarios. | MySQL is commonly used for structured data and traditional relational database use cases, such as e-commerce systems, content management systems, and web applications. It excels in scenarios with complex transactions, strict data consistency requirements, and well-defined schemas. |
| Node.js | Django |
| Built on Chrome's V8 JavaScript engine.<br>High performance and fast execution. Efficiently handles I/O-bound tasks | Utilizes Python, which is generally slower than JavaScript. Suitable for CPU-bound tasks and complex computations |
| High performance and scalability with non-blocking I/O and event-driven architecture [2] | Good performance but can't match Node.js in handling high traffic and real-time applications |
| Built-in asynchronous programming model which allows handling multiple<br>requests efficiently | Asynchronous programming is possible but not as seamlessly integrated as in Node.js |
| Allows using JavaScript for both frontend and backend, making full-stack development easier | Requires using different languages for frontend (HTML/CSS/JS) and backend (Python) |
| Excels in building real-time applications like chat and gaming platforms | Possible to build real-time apps but requires additional tools and setup |
| Highly flexible and allows building everything from scratch based on project needs | Comes with many built-in features and conventions, which can limit flexibility |

Skill-Swap belongs to the EdTech Domain. EdTech or Educational Technology is a fast-growing field that integrates education with information technology to revolutionize learning and teaching. This field targets the development of interactive, inclusive, and customized learning experiences using technology. Contemporary EdTech platforms utilize diverse technologies such as artificial intelligence, machine learning, cloud computing, and interactive multimedia to provide educational resources and learning processes [8].

The EdTech space has a number of its main constituents that determine its boundaries. It basically comprises Learning Management Systems (LMS), delivery of digital content, assessment tools, and student engagement components. Deep learning EdTech solutions usually have adaptive learning routes that modify according to a student's progress, analytics to monitor learning results, and group-focused tools facilitating peer-to-peer learning. It also includes mobile learning, gamification, virtual and augmented reality to deliver experiential learning, and AI-based tutoring tools.

The importance of EdTech has increased exponentially, particularly since the worldwide transition to remote learning. The field now encompasses more than just traditional educational environments to include professional training, skill development, and lifelong learning platforms. Contemporary EdTech solutions prioritize accessibility, scalability, and engagement, frequently integrating social learning features, real- time feedback mechanisms, and customized learning experiences. This has resulted in the development of varied platforms from the massive open online courses (MOOCs) to niche skill-development apps, which have brought education within easier reach and with greater flexibility than ever [9].

The EdTech segment is also typified by having its own peculiar technical demands and challenges that make it distinct from other software verticals. Such demands include demanding user tracking and progress monitoring requirements, detailed analysis of learning accomplishments, secure exam systems, as well as coping with diverse styles of educational materials from text to video to simulations. The domain requires high scalability and availability to support different numbers of concurrent users, particularly during peak learning seasons such as exams or scheduled classes. Moreover, EdTech platforms need to focus on accessibility features to provide inclusive learning experiences, integrate advanced content management systems for educational content, and sustain engagement through functionalities such as gamification, reward systems, and social learning elements. The intricacy of these demands makes EdTech one of the most difficult but rewarding areas in contemporary software development [1].

## 3. Research Methodology

Presently, people looking to acquire new skills or pass on their knowledge to others are faced with various challenges. Most current education platforms focus more on financial exchange, e.g., paid courses or tutoring services, which is limiting for people who have less to spend. Also, the platforms might not give the personalized and interactive learning environment that many people want. Furthermore, there is also a deficiency of focus on the importance of skill-based exchange, where one is able to exchange their skills and knowledge without receiving monetary reward. This non-economic dimension of knowledge acquisition is commonly neglected, precluding people from being able to participate in mutually beneficial skill exchange and learn from others' experience and knowledge.

- The target audience for this problem encompasses a wide range of individuals, including:
- Lifelong learners who are passionate about acquiring new skills and knowledge.
- Experts and professionals who possess valuable skills and are willing to share their expertise with others.
- Individuals who may not have the financial means to access traditional educational resources.
- Community members who value collaboration, reciprocity, and the exchange of knowledge.

The consequences of not addressing this problem are significant. Without a platform that facilitates skill- based exchanges and emphasizes non-financial learning opportunities, individuals may miss out on valuable chances for personal and professional growth. They may struggle to find mentors or learning partners who can provide personalized guidance and support. Furthermore, the lack of a community-driven learning platform can lead to a sense of isolation and disconnection among learners, hindering the potential for collaboration and collective knowledge-sharing. The methodology follows Git Flow Model presented in figure 1.

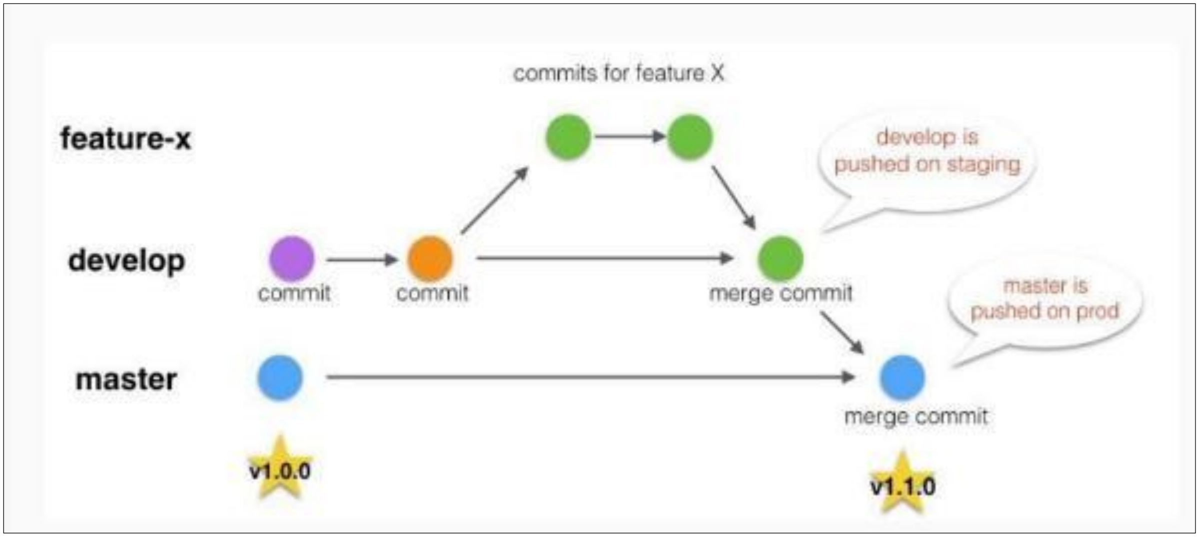


**Figure 1** Git Flow Model

The Git Flow model is a branching model for Git that offers a solid framework for working with big projects. Vincent Driessen first published the model in 2010, and it has since become widely used as a version control workflow for software development teams. According to the model, there are two primary branches: master (or main) and develop. The master branch houses production code, and all of the development code is pushed to this branch and version-tagged. The develop branch houses pre-production code and is an integration branch for the features. The detail methodology is shown in figure 2.

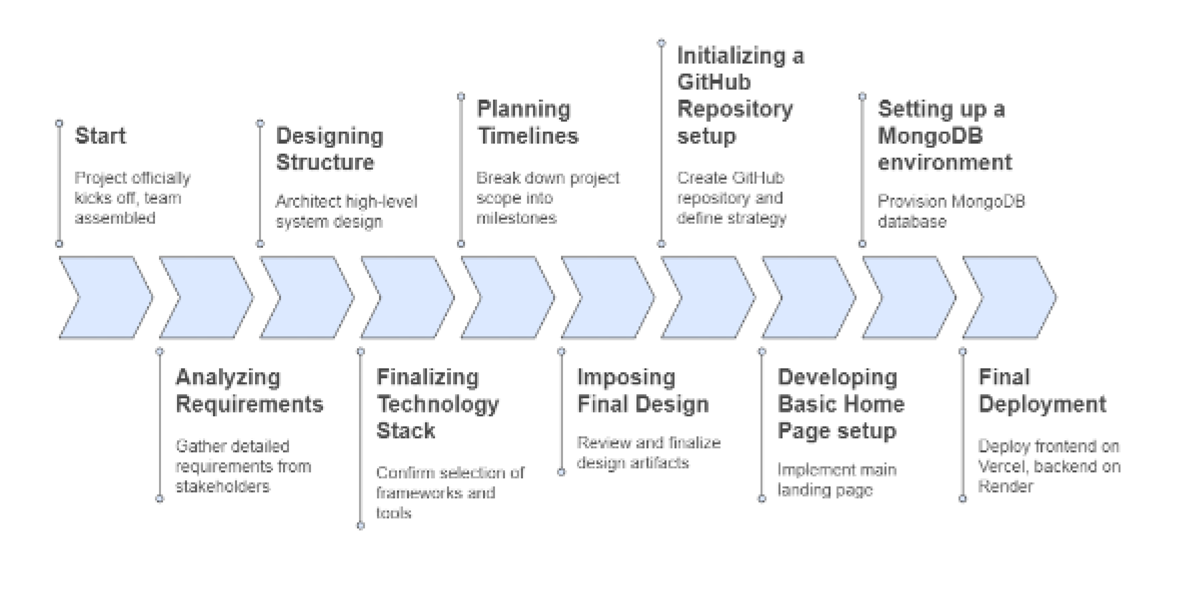
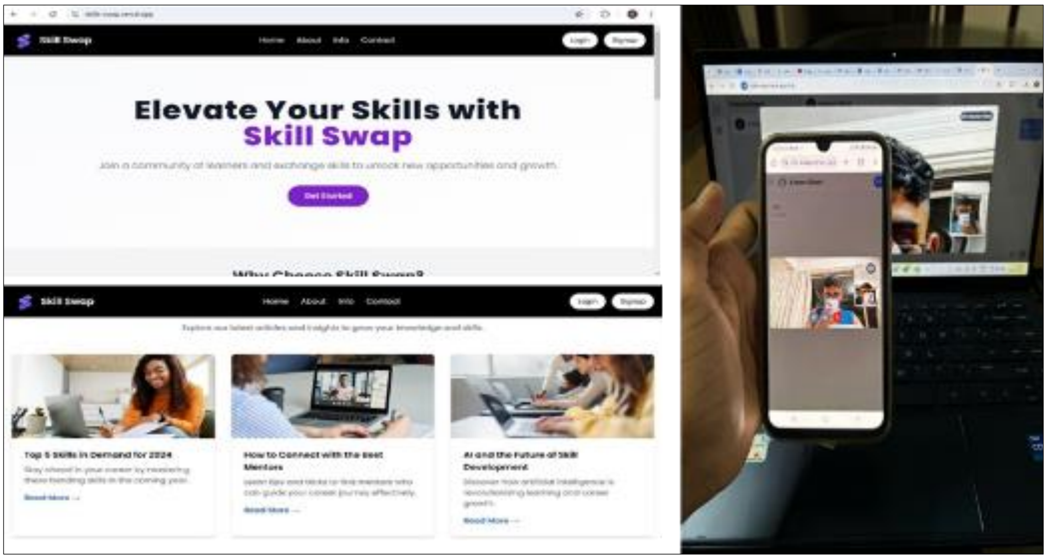

Start
Project officially kicks off, team assembled
Analyzing Requirements
Gather detailed requirements from stakeholders
Designing Structure
Architect high-level system design
Finalizing Technology Stack
Confirm selection of frameworks and tools
Planning Timelines
Break down project scope into milestones
Imposing Final Design
Review and finalize design artifacts
Initializing a GitHub Repository setup
Create GitHub repository and define strategy
Developing Basic Home Page setup
Implement main landing page
Setting up a MongoDB environment
Provision MongoDB database
Final Deployment
Deploy frontend on Vercel, backend on Render

**Figure 2** Flow of methodology of the study

Skill Swap app provides an exciting solution for matching individuals based on their skill sets and providing a collaborative learning environment. Using a robust tech stack of Node.js, MongoDB, and machine learning algorithms, the platform attempts to provide an efficient and customized skill matching experience.

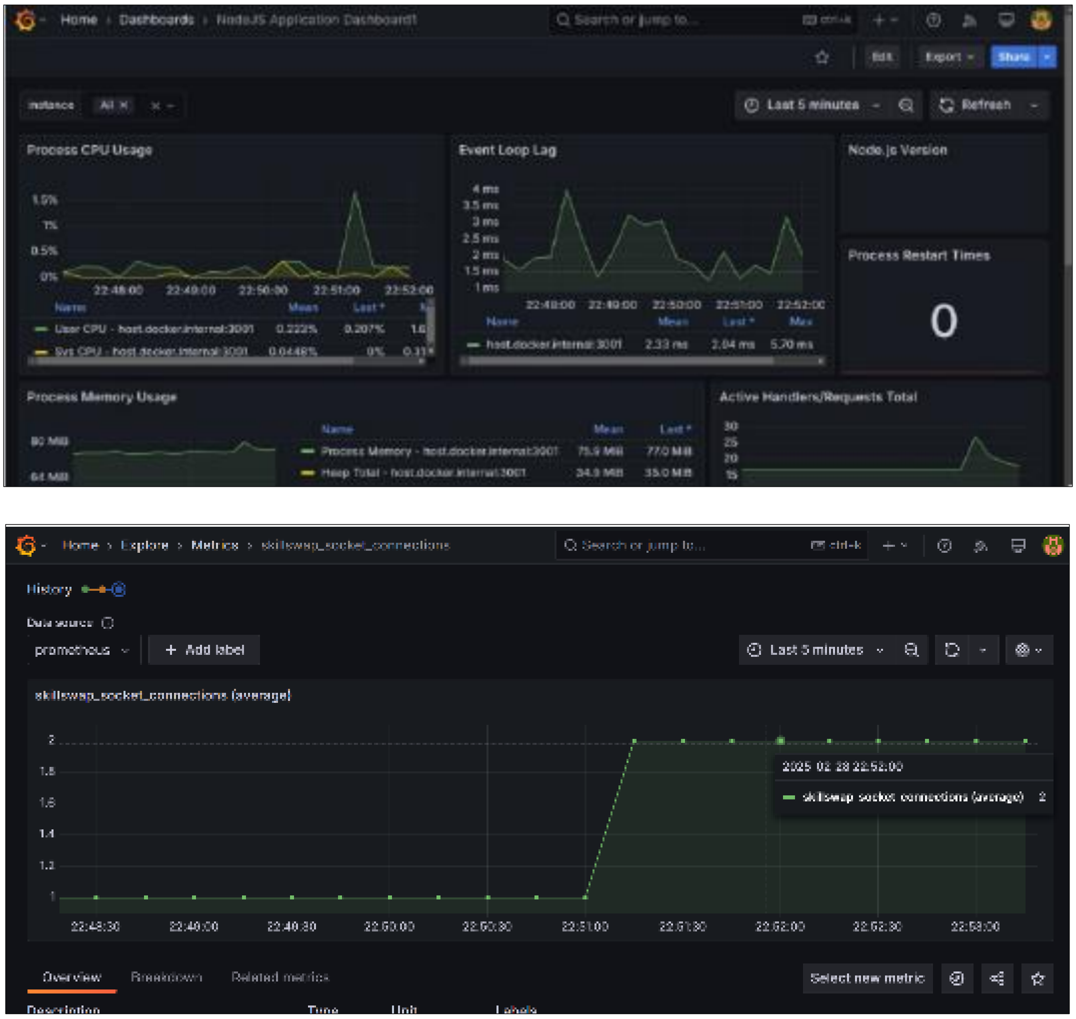
The procedure design process, as evidenced in the diagram, is a systematic process of developing applications. From initial requirement analysis and system design to applying major features such as skill assessment, real-time messaging, and user profiling, each step is systematically planned and executed. Adding a machine learning model for predicting scores gives the platform a smart level, adding an extra layer of accuracy to skill suggestions.

## 4. Results and Discussion



Skill Swap
Home About Info Contact
Login Signup
Elevate Your Skills with
Skill Swap
Join a community of learners and exchange skills to unlock new opportunities and growth.
Get Started
Why Choose Skill Swap?
Skill Swap
Home About Info Contact
Login Signup
Top 5 Skills in Demand for 2024
Read More →
How to Connect with the Best Mentors
Read More →
AI and the Future of Skill Development
Read More →

**Figure 3** Home screen and video calling of Skill Swap



| Name | Mean | Last * |
|---|---|---|
| User CPU - host.docker.internal:3001 | 0.222% | 0.207% |
| Sys CPU - host.docker.internal:3001 | 0.0448% | 0% |

| Name | Mean | Last * | Max |
|---|---|---|---|
| host.docker.internal:3001 | 2.33 ms | 2.04 ms | 5.70 ms |

| Name | Mean | Last * |
|---|---|---|
| Process Memory - host.docker.internal:3001 | 75.9 MiB | 77.0 MiB |
| Heap Total - host.docker.internal:3001 | 34.9 MiB | 35.0 MiB |

**Figure 4** Dashboard of Skill Swap

As Skill Swap undergoes its implementation, it is essential to pay attention to user feedback and continuously iterate on the design in an effort to deliver a seamless and useful user experience. Through agility and adaptability, the platform will be able to navigate challenges and leverage new opportunities within the constantly evolving landscape of skill-exchange and remote working. Figure 3 and 4 shows outcome of this research.

To compute the match characteristics, we initially ascertain the pertinent skill, which is the user's talent that corresponds with the search skill query. Subsequently, we ascertain the matching requirement by identifying the current user's skill that corresponds with the requisite abilities of the pertinent skill. Utilizing this information, we generate the base match score through the `calculateMatchScore` function. The base score is calculated by dividing the pertinent skill's value by 20, hence transforming the scale from 0-100 to 0-5. A mutual match bonus of 1 is awarded if a matching criterion is met, and a experience bonus is provided based on the amount of certificates held by the user, capped at 1. The skill rating denotes the value of the pertinent skill, defaulting to 0 in the absence of a relevant skill, whereas the experience score reflects the quantity of certificates held by the user.

Upon calculating these features, we organize the match data for the machine learning model by compiling a list of match objects that encompass user data and the derived features. A Python child process is initiated to run the `predict.py` script, which receives the match data in JSON format to provide match score predictions. The script subsequently analyzes the input and generates the anticipated match scores. In the event of an error during this operation, the system resorts to the default base match scores.

Upon acquiring the anticipated match scores, they are integrated with the user data to create a compilation of final match objects. The matches are thereafter arranged in descending order according to the match scores. The API response ultimately returns the ordered list of matches.

Auxiliary functions are employed to facilitate this capability. The `calculateMatchScore` function determines the fundamental match score by evaluating the pertinent skill rating, reciprocal skill bonus, and experience bonus. The `getPythonPredictions` function executes the machine learning model by initiating a Python session, transmitting match data in JSON format, obtaining projected match scores, and managing mistakes by reverting to base scores when required.

## 5. Conclusion

With its innovative idea, robust technology basis, and human-cantered design, the Skill Swap app can revolutionize the way individuals acquire, swap, and apply skills in today's fast-paced, knowledge-driven world. The more individuals embrace the platform and contribute to its growth, the more it can be a thriving ecosystem where diverse people across the globe can connect, learn, and thrive together. Moreover, the future enhancements segment is indicative of massive growth and development potential. Stabilizing video calls across different browsers, implementing group calls and messaging, creating Reddit-type skill communities, and incorporating user experience data in matching algorithms are just a few of the manners in which the platform can develop to meet changing demands of users.

## Compliance with ethical standards

### *Disclosure of conflict of interest*

All authors declare that they have no conflicts of interest to disclose.

## References

[1] A. Gurusamy and I. A. Mohamed, “Unlocking Innovation: How Full Stack Development is Reshaping Healthcare Technology,” Journal of Knowledge Learning and Science Technology ISSN: 2959-6386 (online), vol. 1, no. 1, Art. no. 1, Jan. 2021, doi: 10.60087/jklst.vol1.n1.p115.

[2] A. Gurusamy and I. A. Mohamed, “The Evolution of Full Stack Development: Trends and Technologies Shaping the Future,” Journal of Knowledge Learning and Science Technology ISSN: 2959-6386 (online), vol. 1, no. 1, Art. no. 1, Nov. 2020, doi: 10.60087/jklst.vol1.n1.p108.

[3] V. Shrivastava, S. Sharma, N. K. Marwal, and F. S. Khan, “The New Era of Full Stack Development,” SCRS Proceedings of International Conference of Undergraduate Students 2023, pp. 51–60, Oct. 2023.

[4] A. Taivalsaari, T. Mikkonen, C. Pautasso, and K. Systä, “Full Stack Is Not What It Used to Be,” in Web Engineering, M. Brambilla, R. Chbeir, F. Frasincar, and I. Manolescu, Eds., Cham: Springer International Publishing, 2021, pp. 363–371. doi: 10.1007/978-3-030-74296-6_28.

[5] “Development of a Metric for Assessing Full-Stack Developersâ€TM Expertise,” in SciSpace - Paper, Jan. 2018. Accessed: Apr. 01, 2025. [Online]. Available: https://scispace.com/papers/development-of-a-metric-for-assessing-full-stack-developersa-4ts58ymtpv

[6] “A Workshop / Optimist’s Guide to Finding Optimal Infrastructure for a Course in Full-Stack Development | Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 2.” Accessed: Apr. 01, 2025. [Online]. Available: https://dl.acm.org/doi/10.1145/3545947.3569620

[7] “Full Stack Web Development,” Indian Scientific Journal Of Research In Engineering And Management, Oct. 2024, doi: 10.55041/ijsrem37848.

[8] N. Q. Khue, “Location-based web crawler for geotagged images retrieval,” p. 56.

[9] G. Bhumireddy, “Comparison of Machine Learning algorithms on detecting the confusion of students while watching MOOCs”.