Evolving Web Application Development Frameworks: A Survey of Ruby on Rails, Python, and Cloud-Based Architectures

Kiran Nittur[1*], Srinivas Chippagiri[2], Mikhail Zhidko[3]

[1,3]Software Engineer, RackWare Inc, San Jose, USA
[2]Software Engineer, Tableau Software Inc, Seattle USA

***Corresponding Author,** kiran.nittur@rackwareinc.com

ABSTRACT

The popularity of web apps and other forms of online application development has skyrocketed in the last few years. It is crucial to choose the optimal technology or framework, especially given the proliferation of options. This survey explores three major paradigms: Ruby on Rails, Python-based frameworks and cloud-based architectures. Ruby on Rails is focused on the MVC model and consists of an ideal structure of working and the philosophy of "convention over configuration," increasing productivity along with focusing on such values as standardization of work. Two widely used Python-based frameworks are Django and Flask; these have the options for extensive flexibility, scalability and libraries to support virtually any kind of application that one might want to build. Cloud computing, with its layers and service models, brings great scalability, cost efficiency, and a rich resource pool to deployment. Additionally, some research space has been addressed that needs to be tackled related to the improvement of frameworks, flexibility and their scalability as well as long-term sustainment of the systems; these areas pose a need for applying inter-disciplinary solutions incorporating cutting-edge technologies for building more reliable and efficient web applications.

Keywords—Web application development, frameworks, ruby on rails, Django, cherrypie, cloud computing, web2py

INTRODUCTION

There were a lot of mistakes and human labor involved in the early days of web development when all apps were developed manually. The introduction of web frameworks in the early 2000s served as a response to these advancements. A web framework is essentially an instrument that aids in the creation of websites, preventing errors and saving time. Web pages may employ frameworks for both static and dynamic content. A framework may be selected based on the job at hand. Frameworks fall into one of two categories: client-side or server-side. Client-side frameworks, like vue.js, angular.js, and ember.js, are in charge of creating and enhancing user interfaces, which include animated elements and sophisticated layouts[1]. The principles and design of server-side frameworks, on the other hand, let you build a wide variety of pages. The web pages may benefit from these frameworks' safety features. Examples of server-side programming include Ruby on Rails, Zend, Django, and others [2]. The introduction of web frameworks not only addressed the inefficiencies of manual coding but also introduced a structured approach to web development, enabling the reuse of software components and improving the overall quality of web applications.

An advancement above basic library-based reuse, a framework is a high-level solution for software component reuse that permits sharing generic logic and common functionalities of a domain application. The fact that a critical component of the application is already present in the framework and has undergone testing means that the final output will be of higher quality. There are a plethora of web frameworks available today, built on several programming languages: Web frameworks written in Java include JSF and Struts; Ruby-based frameworks include Ruby on Rails and Grails; and PHP-based frameworks include CakePHP[3]

Selecting the best architecture for application development is a highly challenging issue in today's internet-driven era, when a wide variety of technologies are accessible for designing web applications. Among the various web frameworks, Ruby and its Rails framework have stood out for their efficiency in rapid application development, showcasing the power of frameworks to enhance both development speed and functionality [4]. Development of web-based services and applications is best accomplished using the Ruby language and the Rails framework. Ruby is a great language for quick prototyping and integrating different web services because of its current and dynamic characteristics. The rapid web app creation, database access, and AJAX support offered by Rails make it a great option for creating the front end and back end of next-gen online applications and services [5]

Python's versatile frameworks have revolutionized web development by simplifying complex tasks and supporting the creation of robust applications[6]. Traditional web programming approaches, such as CGI and WSGI, required building web applications from scratch using Python's standard libraries. However, modern Python web frameworks, packaged as modules, have streamlined this process by automating tasks like request interpretation, response generation, and data storage. These frameworks significantly reduce developers' workload while supporting both server-side and client-side programming[7]. Additionally, the rise of cloud-based architectures has further transformed web application development, enabling scalable, resilient, and globally distributed applications. Technologies like serverless computing, microservices, and containerization allow developers to focus more on application logic than infrastructure management[8]. As traditional frameworks converge with cloud-based architectures, developers must adopt sustainable and progressive methodologies to meet the demands of global interconnectivity and evolving digital ecosystems.

A. Organization of paper

Here is how the remainder of the paper is organized: A summary of Ruby on Rails, its design, and its benefits is given in Section II. Section III discusses Python-based web application frameworks, their merits. Section IV discusses how cloud technologies are changing web development and includes information on its architecture, services model, and benefits. The literature review and research deficit are covered in the concluding section, Section V, which also offers the review's conclusion and prospects for further research.

Overview Of Ruby On Rails

RubyonRails is one example of a framework for creating web applications. This framework is popular among developers for backend development, and it is written in Ruby programming language. Software engineers find online application development more laborious; to make it less laborious and more adaptable, web development specialists use ROR technology. One of the most effective and well-liked frameworks for creating dynamic web applications is Ruby on Rails.The Ruby on Rails framework is a free and open-source set of tools for building web applications. The course is structured on the Model-Controller-View paradigm. With ROR, building a database-backed web application using the Ruby programming language is a breeze [9].

Architecture Model for Ruby on Rails

ROR is an application framework that lets you concentrate on the app's other features while it takes care of the repetitious code that most applications have. Developers may use the tools provided to create a web application. There are a few components that are used by almost every Ruby program. Ruby code libraries make up this framework. You may study the framework's inner workings, make contributions, and evaluate the complete codebase since it is open source. The fact that it is an MVC framework, however, is the primary factor in its widespread adoption [10].

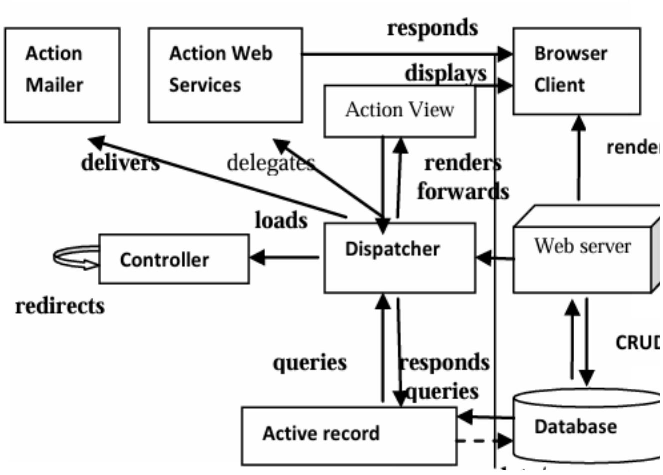

Fig. 1. Architecture Model for Ruby on Rails (RoR) Web Application Implementation

Figure 1 shows the Architecture of Rails framework. Active Record and Action Pack are two important components of the Rails framework. The Model component of an MVC application is handled by the Action Record layer, which is an ORM component. To the remainder of the program, this layer represents the rows of the data table as model objects. The view and controller components of an MVC application are handled by Action Pack. The action gets the answer ready while collaborating with the relevant model objects. The Action Web services, or the Action Mailer Component transmitan appropriate answer, unless the browser made the request, in which case the Action View Component renders the response. Below is a breakdown of each architectural feature seen in the above image:

- **User component–** Browsers may access Rails web applications over many protocol interfaces, including HTTP, FTP, or SOAP for web service expenditure.
- **Web Server–** A web server hosts the built Rails app, processes requests from the user-facing client component, and passes them on to the dispatcher.
- **Dispatcher–** Depending on the user's current request, the dispatcher will call the appropriate controller.
- **Controller–** A controller takes in a request from a user and then triggers the appropriate action.

Ruby on Rails Architecture Features

There are some architecture features for Ruby on Rails.

- Ruby on Rails follows Model-View-Controller architecture.
- Web services using Representational State Transfer (REST).
- MySQL,Oracle,Microsoft SQL Server,PostgreSQL, IBM DB2, and every other major database is compatible with ROR architecture.
- Open-source server-side scripting language is used by Rails architecture.
- ROR architecture follows convention over configuration principle.
- It has generators of scripts for task automation.
- Rails uses the human-readable data serialization format known as YAML machine [11].

Advantages of Ruby on Rails

There are some advantages of Ruby on Rails, so provide some important advantages given below:

- The robust support of web standards for the whole application, including the user interface and data transport, is one of the most noticeable features of Ruby on Rails.
- Principles of program design, such as "coding by convention," "don't repeat yourself," and the "active record pattern" are expected to be followed by ROR applications.
- ROR is one of the three most popular web development technologies, and it is utilized by the most popular websites in the world[12]. This indicates that a sizable number of individuals use this framework to develop applications worldwide.
- Ruby on Rails makes use of a large variety of useful tools and libraries.
- When it comes to the tools they utilize for their projects, large organizations with qualified analysts often make wise choices. For this reason, such equipment often acquires an unquestionable reputation for being sturdy and dependable[14].

PYTHON BASED WEB APPLICATION FRAMEWORKS

The Python Web Framework is a set of modules or packages that let programmers create Web services or apps. It relieves developers of the burden of managing low-level aspects like sockets, protocols, and process/thread management [15]. CherryPy, Django, Bottle, Flask, and other web frameworks have all been developed using Python. Content management, database interface, and internet protocol (HTTP, SMTP, XML-RPC, FTP, and POP) interfacing are all made easier using the standard libraries and modules provided by these frameworks [16].

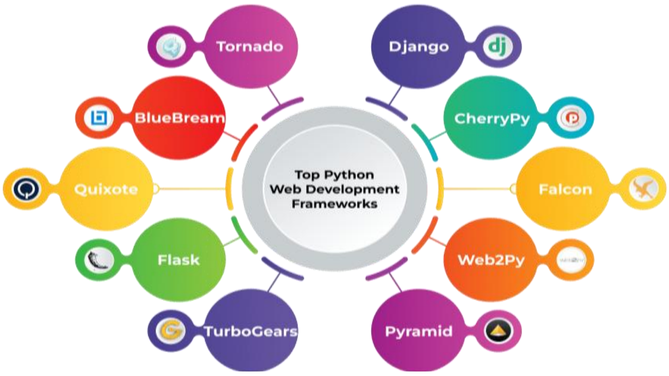

Fig. 2. **Python-based web development frameworks**

B. Popular Python frameworks

They discuss about the popular python frameworks for using web application development.

- **Django:** Python is the language's foundation for this web platform. Its insanely high speed was intentionally intended to facilitate the rapid development of apps from developers' concepts [17]. It takes care of most of the problems with making web apps, so instead of starting from square one, developers can focus on making the app's main business logic[18]. It is very adaptable.
- **Flask:**It is a Python-based web framework. Websites may be quickly and easily built using Flask's library and scripts, eliminating the need to start from zero. The fundamental features of flask will reduce its need on several external libraries that require maintenance and make it lighter[19].
- **Web2Py:**DAL and rule-based access control capabilities are provided by a collection of robust libraries that are built on the Python language foundation. Not only is it compatible with HTML, XML, and JSON, but it also mandates the use of the model-view-controller paradigm[20],[21].
- **Bottle:**The Bottle is a genuine Python framework, which is a simple microframework[22]. The framework is lightweight and quick to implement since it just a single file. Unlike other frameworks, the Bottle framework is feature-limited and does not rely on any libraries outside of the Python standard library.
- **CherryPy:**It is Python's first programming language foundation. Because of its flexible nature, this microframework may be expanded. Tools like sessions, authorization, caching, routing, JSON processing, and database support are all built in. Out of the box, it has the WSGI thread pool server, which is compatible with HTTP/1.1[23]. It can concurrently execute numerous HTTP servers on various ports.

Benefits of Python- based framework

Python-based web frameworks, such as Django and Flask, offer numerous advantages that make them highly popular among developers. These benefits include:

- Python offers extensive libraries for tasks like string operations, web services, scientific programming, and computation.
- Its libraries enable scalability, reduce coding time, and ensure reusability for future applications.
- Python's versatility and clean syntax make it adaptable and easy to use for various development needs.
- Its dynamic typing simplifies data analysis and enhances its appeal to data scientists.
- Python's readable syntax resembles English, reducing coding complexity and aiding understanding.
- It supports easy data visualization through charts and graphs, simplifying reporting and insights.
- Asynchronous coding in Python allows efficient processing, avoiding deadlocks and simplifying development.
- Python supports multi-paradigm programming: object-oriented, procedural, and functional.
- As an open-source language, Python is free and frequently updated with community-driven enhancements.
- Python has a vast developer community offering quick support, project assistance, and issue resolution.

II. Envolve Transforming Web Development With Cloud Technologies

Web applications have evolved from traditional websites or web systems, enhancing their capabilities by adding essential business functionality. However, they face significant challenges, particularly in terms of scalability and high availability. To address these challenges, cloud data stores are utilized, as they offer the necessary properties to ensure both high availability and scalability[25]. Moreover, cloud computing presents a vision of an almost limitless pool of storage, computing, and networking resources, which allows applications to scale effectively and be deployed with ease. By leveraging these cloud technologies, businesses can better meet their operational needs and adapt to changing demands [26].

A. Architectural layers of cloud computing

The two primary components of a cloud computing system are the front end and the back end. The net or other network connects each area unit to every other area unit. In contrast to what the user (consumer) perceives, the system's cloud constitutes the back end. The client's laptop and the necessary application to access the cloud make up the front, while the cloud computing services, including several computers, servers, and information storage, make up the back [27],[28]. Figure 3 shows that there are four distinct levels to the cloud computing architecture: physical, infrastructure, platform, and application [29].

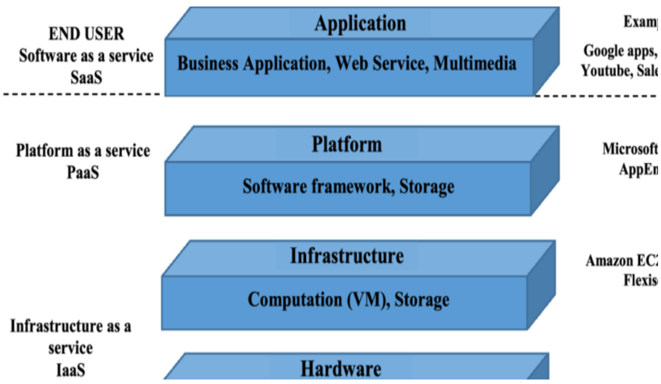

Fig. 3. Architecture layers of cloud computing

- **Hardware layer:** Things like electricity, cooling systems, servers, switches, and routers are handled by the hardware layer, which is also responsible for the cloud's physical assets.
- **Infrastructure layer:** The virtualization layer is another name for the infrastructure layer. As a result of the infrastructure layer's use of virtualization technologies such as VMware and KVM, a shared pool of storage and processing power is created.
- **Platform layer:** The foundational layer, which includes OSes and requisition structures, forms the basis of the platform layer.
- **Application layer:** The real cloud offerings, including business apps, multimedia, and web services, are housed on the application layer [4].

Service Models of Cloud Computing

Figure 4 shows the several service models available in cloud computing, including Infrastructure as a Service (IAAS), Platform as a Service (PAAS), and Software as a Service (SAAS).

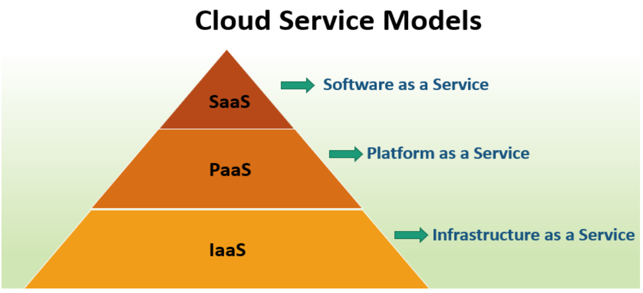

Fig. 4. Cloud Service Models

Infrastructure as a Service (IAAS)

One kind of cloudcomputing that offers virtualized computer resources across an internet is IaaS. Customers of the cloud service have direct access to the IaaS cloud's processing power, storage space, networks, and other essential computer resources. IaaS clouds enable "Virtualization" to connect and disassemble physical resources on demand, allowing them to adapt to fluctuating resource demands. A prime example of IaaS is Amazon EC2[32].

1) Platform as a Service (PAAS)

PaaSis an all-inclusive cloud-based environment for developing and deploying applications, including tools to build simple cloud-based apps to sophisticated corporate programs with cloud capability [32][33]. By using PaaS, customers are able to construct their own cloud services and apps, similar to SaaS, directly on top of the PaaS cloud. This platform encompasses the whole of the "Software Lifecycle." Google App Engine is a Paa) example.

Software as a Service (SAAS)

Instead of buying software outright, SaaS users pay to access and use it over the Internet using a cloud-hosted approach. Customers of cloud services are able to publish their apps to a hosting environment, which application customers may access over the internet using a variety of devices and browsers. Google Docs, Google Mail, and SalesForce.com are all examples of SaaS[34].

Benefits of cloud computing in web development processes

Web development has been revolutionized by cloud computing, which provides solutions that are scalable, efficient, and cost-effective. The key benefits include:

Cloud Hosting
Benefits for
Web App
Development
Optimum Use of
Resources
Scalability
Cost savings

Fig. 5. Cloud Hosting Benefits for Web App Development

- **Scalability:** Cloud computing enables seamless scalability by increasing server capacity or upgrading hardware, eliminating traditional concerns like server load balancing, and supporting strategic growth effortlessly.
- **The abundance of cloud services & resources:** Cloud services simplify web application development, testing, and deployment with minimal downtime, real-time performance metrics, and automated tools, reducing profit loss from maintenance issues.
- **Cost-effective:** Cloud computing reduces upfront and operational costs through pay-as-you-go pricing. It eliminates the need for physical servers and enables businesses to scale affordably as demand grows[35].

Literature Review

In this section, some previous studies of the researchers are explained on Evolving Web Application Development Frameworks of Ruby on Rails, Python, and Cloud-Based Architectures.

In,Tachibana, Kon and Yamaguchi (2017)showcase a Docker-consolidated performance assessment of Ruby on Rails (RoR) web apps. To start, Theylooked at how the performance of containers changes as a function of their quantity. In the second place, highlight the problems with heavily consolidating servers. Finally, go over a way to fix the problems. Technologies for virtualization, which allow for flexible allocation of computing resources, are extensively used for many different applications. Because of its minimal virtualizing overhead via kernel sharing, the most widely used container-based virtualization technologies are becoming more and moresignificant[32].

In, Liawatimenaet al. (2018)adopted an approach for evaluating the code quality of the Django Web Frameworks with the help of Radon and Pylint. The development of online apps that provide a plethora of possibilities, including many designers and sophisticated tools. The combination of using a content management system and building an application from the ground up is made wonderful by web frameworks. The focus of this piece is Django, an open-source web development framework[33].

In,Eugenio and Miguel (2018) included the period from 1998 to 2017. Result of this development is a useful tool for authorities and, in general, for university community. he webs service runs at Google Cloud Platform which, among others, offers Google App Engine development platform. The web service demands very few resources. It works under Google free quota, made available for web developers. All this software is equipped with the ability to coordinate communication between different parts of the web service[34].

In,Takatori et al. (2015)recommend an innovative home network system that uses the cloud to accomplish resource, security, and fault isolation. Our first step is to utilize IaaS to set up a virtual home server for each family. A single service or application is housed in each of the supplementary virtual machines that are built on top of each virtual home server. Our three-tiered design, which includes HNS Lite, House Cloud, and Service Provider, is based on this concept[35].

In,Munetoh and Yoshioka (2013) suggest a strategy and build a static verification tool to ensure that all security elements are fully implemented in web applications. Theyvalidated the authentication and authorization capabilities of several Ruby on Rails web applications by applying the suggested technique to their source codes. The program proved to be a useful tool for enhancing code-centric and iterative web application development processes with security features[36].

In,Shuaibu and Ibrahim (2017) suggests a paradigm for developing online applications that incorporates security concerns throughout the online Application Development Life-cycle. Applications that are crucial to security, such as e-commerce sites, may adopt this approach. Methods for developing secure online applications have shown a 96% success rate, with just around 4% of attempts failing. The information severity level of the failures means that they are usually not significant to the underlying application's security[37].

This Table Ioffers a structured overview of an objectives, methods, key findings, advantages, and limitations of the referenced works, along with potential areas for future research and improvement.

TABLE I. RELATED WORK SUMMARY

References	Focus/Objective	Key findings	Technology/Frameworks	Future Work/Limitations
[32]	Performance evaluation of Ruby on Rails (RoR) consolidated with Docker.	Explored the relation between container count and performance. Identified issues with high server consolidation and proposed mitigation strategies.	Ruby on Rails, Docker	Addressing kernel-sharing challenges in containerised environments for large-scale applications.
[33]	Measurement of Django Web Framework code quality using Radon and Pylint.	Delivered insights into code quality metrics for Django, emphasising tools and methods for improved software design and maintenance.	Django, Radon, Pylint	Focus on scalability challenges and advanced debugging for large-scale Django applications.
[34]	Evolution of cloud-based web service hosted on Google Cloud Platform.	Developed a web service demanding minimal resources, leveraging Google's free quota.	Google Cloud Platform, Google App Engine	Limited focus on high-resource applications; suggested expansion for complex enterprise solutions.
[35]	Novel cloud-based architecture for secure home network systems using IaaS.	Proposed architecture achieves security, fault, and resource isolation through layered virtualised home servers.	IaaS, Virtual Machines	Requires practical implementation for large-scale adoption; further exploration of real-world failure scenarios is needed.
[36]	Development of a static verification tool for web applications to verify security features.	Effective in checking authentication and authorisation features in Ruby on Rails applications, aiding iterative development.	Ruby on Rails	Limited to security features; future work includes broader scope for verifying performance or usability features.
[37]	Security-focused web application development model for critical applications like e-commerce sites.	Achieved a 96% security level in development; failures are minimal and non-critical to application security.	Generic Security Models	Further testing is required in diverse application domains; enhance integration of AI-based threat detection in the lifecycle.

CONCLUSION AND FUTURE WORK

The need for web application frameworks has surged in tandem with the growing necessity for various sorts of online applications. One way to streamline the process of creating web applications is via a framework, which is essentially a collection of tools and libraries. In this study, they have discussed various frameworks and technologies that contribute to the evolution of web application development, especially focusing on Ruby on Rails, python-based frameworks, and cloud computing architectures, each offering unique benefits such as enhanced productivity, scalability and cost efficiency. Despite their advancements, challenges like resource optimization, adaptability to dynamic environments, and integration with emerging technologies persist.Future research should endeavor to fill these gaps to achieve new and effective solutions for web applications with improved and innovative technological standards. Rapid web application development has never been easier than with Ruby on Rails, an open-source framework that focuses on the Ruby programming languages and is widely utilized by agile programming teams. Programmers describe uncommon configurations and employ unique, non-duplicate information, which is an intriguing feature of RoR. For Agile development methods, Rails is the ideal platform.

REFERENCES

[1] V. Kumar, V. V. Kumar, N. Mishra, F. T. S. Chan, and B. Gnanasekar, “Warranty failure analysis in service supply Chain a multi-agent framework,” in SCMIS 2010 - Proceedings of 2010 8th International Conference on Supply Chain Management and Information Systems: Logistics Systems and Engineering, 2010.

[2] D. H. Curie, J. Jaison, J. Yadav, and J. R. Fiona, “Analysis on webframeworks,” in Journal of Physics: Conference Series, 2019. doi: 10.1088/1742-6596/1362/1/012114.

[3] M. D. P. Salas-Zárate, G. Alor-Hernández, R. Valencia-García, L. Rodríguez-Mazahua, A. Rodríguez-González, and J. L. López Cuadrado, “Analyzing best practices on Web development frameworks: The lift approach,” Science of Computer Programming. 2015. doi: 10.1016/j.scico.2014.12.004.

[4] V. Kumar, M. Tripathi, S. Tyagi, and M. K. Tiwari, “An integrated real time optimization approach (IRTO) for physical programming based redundancy allocation problem,” Proc. 3rd Int. Conf. Reliab. Saf. …, 2007.

[5] E. M. Maximilien, “Web services on rails: Using ruby and rails for Web services development and mashups,” in Proceedings - ICWS 2006: 2006 IEEE International Conference on Web Services, 2006. doi: 10.1109/ICWS.2006.139.

[6] V. V. Kumar, F. T. S. Chan, N. Mishra, and V. Kumar, “Environmental integrated closed loop logistics model: An artificial bee colony approach,” in SCMIS 2010 - Proceedings of 2010 8th International Conference on Supply Chain Management and Information Systems: Logistics Systems and Engineering, 2010.

[7] S. Taneja and P. R. Gupta, “Python as a Tool for Web Server Application Development,” Int. J. Information, Commun. Comput. Technol., 2014.

[8] V. V Kumar, “An interactive product development model in remanufacturing environment : a chaos-based artificial bee colony approach,” 2014.

[9] D. V. Waghmare and P. P. Adkar, “Agile Development using Ruby on Rails Framework,” IRE Journals, 2019.

[10] J. Nijjar and T. Bultan, “Bounded verification of Ruby on rails data models,” in 2011 International Symposium on Software Testing and Analysis, ISSTA 2011 - Proceedings, 2011. doi: 10.1145/2001420.2001429.

[11] V. V. Kumar, S. R. Yadav, F. W. Liou, and S. N. Balakrishnan, “A digital interface for the part designers and the fixture designers for a reconfigurable assembly system,” Math. Probl. Eng., 2013, doi: 10.1155/2013/943702.

[12] S. Meenakshi, “Ruby on Rails – An Agile Developer’s Framework,” Int. J. Comput. Appl., vol. 112, no. 1, 2015.

[13] M. A. Shajahan, N. Richardson, N. Dhameliya, B. Patel, S. K. R. Anumandla, and V. K. Yarlagadda, “AUTOSAR Classic vs. AUTOSAR Adaptive: A Comparative Analysis in Stack Development,” Eng. Int., vol. 7, no. 2, pp. 161–178, Dec. 2019, doi: 10.18034/ei.v7i2.711.

[14] T. Vinothraj, M. Fareez, and A. . Sayeth Saabith, “Python Current Trend Applications- an Overview,” Int. J. Adv. Eng. Res. Dev., 2017.

[15] V. V. Kumar, F. W. Liou, S. N. Balakrishnan, and V. Kumar, “Economical impact of RFID implementation in remanufacturing: a Chaos-based Interactive Artificial Bee Colony approach,” J. Intell. Manuf., 2015, doi: 10.1007/s10845-013-0836-9.

[16] V. V Kumar, M. Tripathi, M. K. Pandey, and M. K. Tiwari, “Physical programming and conjoint analysis-based redundancy allocation in multistate systems: A Taguchi embedded algorithm selection and control (TAS&amp;C) approach,” Proc. Inst. Mech. Eng. Part O J. Risk Reliab., vol. 223, no. 3, pp. 215–232, Sep. 2009, doi: 10.1243/1748006XJRR210.

[17] M. R. Mufid, A. Basofi, M. U. H. Al Rasyid, I. F. Rochimansyah, and A. Rokhim, “Design an MVC Model using Python for Flask Framework Development,” in IES 2019 - International Electronics Symposium: The Role of Techno-Intelligence in Creating an Open Energy System Towards Energy Democracy, Proceedings, 2019. doi: 10.1109/ELECSYM.2019.8901656.

[18] M. Di Pierro, “Web2py for scientific applications,” Comput. Sci. Eng., 2011, doi: 10.1109/MCSE.2010.97.

[19] M. Grisostomi, L. Ciabattoni, M. Prist, G. Ippoliti, and S. Longhi, “Application of a wireless sensor networks and Web2Py architecture for factory line production monitoring,” in 2014 IEEE 11th International Multi-Conference on Systems, Signals and Devices, SSD 2014, 2014. doi: 10.1109/SSD.2014.6808882.

[20] M. Z. Hasan, R. Fink, M. R. Suyambu, and M. K. Baskaran, “Assessment and improvement of intelligent controllers for elevator energy efficiency,” in IEEE International Conference on Electro Information Technology, 2012. doi: 10.1109/EIT.2012.6220727.

[21] S. Hellegouarch, R. Delon, R. Brewer, and Safari Tech Books Online, CherryPyessentials : rapid Python web application development : design, develop, test, and deploy your Python web applications easily. 2007.

[22] M. Z. Hasan, R. Fink, M. R. Suyambu, M. K. Baskaran, D. James, and J. Gamboa, “Performance evaluation of energy efficient intelligent elevator controllers,” in IEEE International Conference on Electro Information Technology, 2015. doi: 10.1109/EIT.2015.7293320.

[23] M. Arora and S. Choudhary, “Building Web Application Using Cloud Computing,” no. January, pp. 88–92, 2013.

[24] M. Sharma, “Cloud Computing Architecture& Services,” IOSR J. Comput. Eng., 2017, doi: 10.9790/0661-1902031318.

[25] A. Jain, P. Singh, and E. A. Jain, “Survey Paper on Cloud Computing,” Artic. Int. J. Innov. Eng. Technol., 2014.

[26] V. V. Kumar, M. K. Pandey, M. K. Tiwari, and D. Ben-Arieh, “Simultaneous optimization of parts and operations sequences in SSMS: A chaos embedded Taguchi particle swarm optimization approach,” J. Intell. Manuf., 2010,doi: 10.1007/s10845-008-0175-4.

[27]. S. S. Manvi and G. Krishna Shyam, “Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey,” Journal of Network and Computer Applications. 2014. doi: 10.1016/j.jnca.2013.10.004.

[28] D. Beimborn, T. Miletzki, and S. Wenzel, “Platform as a Service (PaaS),” WIRTSCHAFTSINFORMATIK, 2011, doi: 10.1007/s11576-011-0294-y.

[29] V. Kumar and F. T. S. Chan, “A superiority search and optimisation algorithm to solve RFID and an environmental factor embedded closed loop logistics model,” Int. J. Prod. Res., vol. 49, no. 16, 2011, doi: 10.1080/00207543.2010.503201.

[30] W. T. Tsai, X. Y. Bai, and Y. Huang, “Software-as-a-service (SaaS): Perspectives and challenges,” Sci. China Inf. Sci., 2014, doi: 10.1007/s11432-013-5050-z.

[31] D. Wu, D. W. Rosen, L. Wang, and D. Schaefer, “Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation,” CAD Comput. Aided Des., 2015, doi: 10.1016/j.cad.2014.07.006.

[32] Y. Tachibana, J. Kon, and S. Yamaguchi, “A study on the performance of web applications based on RoR in a highly consolidated server with container-based virtualization,” in

Proceedings - 2017 5th International Symposium on Computing and Networking, CANDAR 2017, 2017. doi: 10.1109/CANDAR.2017.84.

[33] S. Liawatimenaet al., “Django Web Framework Software Metrics Measurement Using Radon and Pylint,” in 1st 2018 Indonesian Association for Pattern Recognition International Conference, INAPR 2018 - Proceedings, 2018. doi: 10.1109/INAPR.2018.8627009.

[34] M. C. C. Eugenio and C. L. T. Miguel, “Web service for monitoring electricity billing at a university campus,” in Proceedings of the 2018 IEEE 38th Central America and Panama Convention, CONCAPAN 2018, 2018. doi: 10.1109/CONCAPAN.2018.8596449.

[35] S. Takatori, S. Matsumoto, S. Saiki, S. Tokunaga, J. Lee, and M. Nakamura, “A cloud-based architecture for home network system,” in Proceedings of the International Conference on Cloud Computing Technology and Science, CloudCom, 2015. doi: 10.1109/CloudCom.2014.125.

[36] S. Munetoh and N. Yoshioka, “Model-assisted access control implementation for code-centric ruby-on-rails web application development,” in Proceedings - 2013 International Conference on Availability, Reliability and Security, ARES 2013, 2013. doi: 10.1109/ARES.2013.47.

[37] M. B. Shuaibu and R. A. Ibrahim, “Web application development model with security concern in the entire life-cycle,” in 4th IEEE International Conference on Engineering Technologies and Applied Sciences, ICETAS 2017, 2017. doi: 10.1109/ICETAS.2017.8277849.