Secure and Efficient Cloud Service Ranking Using MCDO Approach

Mrs.S.Gayathri[1],
[1]Assistant Professor,
Department of Information Technology,
Karpagam College of Engineering,

Divya.P[2],
Department of Information Technology,
Karpagam College of Engineering,

Jayakumar.A[3],
Department of Information Technology,
Karpagam College of Engineering,

Karthikeyan.P[4],
Department of Information Technology,
Karpagam College of Engineering,

Likeeth.L[5]
Department of Information Technology,
Karpagam College of Engineering,

Abstract.An increase in cloud services makes service selection challenging for cloud users. It is critical to identify the best service that can meet user needs. To that end, this paper introduces a hybrid multiple-attribute decision-making (MADM) model. The proposed method uses the service measurement index cloud (SMI Cloud) structure for qualitative attributes of cloud services and fuzzy values to account for vague user requirements: analytical hierarchy process AHP and fuzzy logic rank cloud services. In addition, an undefined Delphi filtering method is proposed to reduce the execution time of ranking cloud services. Various aspects of the experiments are investigated, including accuracy, execution time, scalability, and sensitivity analysis. The findings suggest that the proposed technique overperforms existing methods in terms of execution time and scalability. Furthermore, the experiments demonstrate that the proposed method achieved an accuracy of 74%.

INTRODUCTION

The project focuses on cloud computing, a crucial aspect of modern media technology. Cloud computing is the process of data storage on Internet servers and its temporary caching on many client devices. Service Level Agreements (SLAs) are acted upon as contracts between cloud consumers and providers, outlining responsibilities, definitions, and parameters for auditing services. The rapid expansion of cloud computing has resulted in pay-as-you-go approaches for IaaS, PaaS, and SaaS, but the competitive nature of the cloud service market makes it challenging for Cloud Users CUs to be selected by the best service providers. Trusted cloud service selection becomes essential for user satisfaction and further development. The cloud is a parallel and distributed system delivering IT capabilities through services, including Anything as a Service. Software as a Service, or SaaS; Platform as a Service, or PaaS; and Infrastructure as a Service (IaaS) are the three cloud delivery methodologies. Due to numerous functionally equivalent cloud services, optimal service selection is crucial. The CSMIC and CQM provide standardized methods for measuring and analyzing services, focusing on key performance indicators and quality criteria, with QoS impacting client-side performance, especially with unpredictable internet connections. To address the difficulty of optimum service selection, the proposed CloudRank system predicts QoS rankings without the need for real-world service invocations, providing a practical solution for cloud application designers. The ultimate goal is to build high-quality cloud applications by leveraging the QoS ranking prediction framework. The research helps to understand and improve cloud service choices by stressing the relevance of security and efficiency in the cloud computing ecosystem.

RELATED WORK

An Efficient Hybrid Ranking Method for Cloud Computing Services Based on User Requirements In this paper, the proposal of a hybrid multiple-attribute decision-making (MADM) model is aimed at resolving the issue of service selection for cloud users, with the goal of identifying the most appropriate service that meets user needs. The proposed method incorporates a service measurement index cloud (SMICloud) structure for the qualitative characteristics of cloud services and factors in vague user needs through fuzzy values. The ranking of cloud services utilizes analytical hierarchy process (AHP) and fuzzy logic, and execution time is reduced by a fuzzy Delphi filtering method. Experimental results suggest the method outperforms existing ones in execution time and scalability, attaining a 96% accuracy rate. Furthermore, the PFDR algorithm, which eliminates unnecessary attributes and exhibits efficiency, is introduced for quick and precise cloud service ranking, yielding a robustness rate of 93.6%. Future studies will center on bettering service selection through interval code encoding of user demands and cloud services, and the exploration of metaheuristic optimization methods such as ant-colony optimization for enhanced energy and SLA metrics performance.

Evaluating and Ranking Cloud IaaS, PaaS and SaaS Models Based on Functional and Non-Functional Key Performance Indicators With the recent maturity of Cloud computing technology and the flexibility of Cloud services, the offering of Cloud services has been exponentially spread by different Cloud service providers, each highlighting various performance indicators. Due to the diversity of Cloud services and their highlighted performance indicators, the selection of the most suitable Cloud service is deemed a complex problem for Cloud users, leading to financial losses and time delays when an unsuitable Cloud service is chosen. In this study, an efficient three-layered framework is proposed for evaluating and ranking IaaS, PaaS, and SaaS Cloud services. Functional and non-functional key performance indicators (KPIs) for Cloud services were identified from six KPI classes, classified with respect to their types and criticality to facilitate easy selection by the Cloud user. The relative importance of the KPIs was determined using the CRITIC method, and KPI values were combined with their relative importance for an overall evaluation and ranking of the Cloud services using the Vikor method. Additionally, an efficient three-layered framework is presented to address the need for effective comparative evaluation and ranking of Cloud services. Various KPIs are highlighted for comprehensive evaluation of Cloud IaaS, PaaS, and SaaS service models, classified as functional or non-functional and static or dynamic, allowing Cloud users to choose KPIs based on their evaluation needs. The second layer of the framework determines KPI weights based on variations in KPI values and their conflicting behaviors, while the third layer effectively ranks Cloud services considering KPI weights. A step-by-step demonstration of the proposed framework is provided through a case study.

MCDM and Various Prioritization Methods in AHP for CSS: A Comprehensive Review The text is as follows: "Confusion in selecting appropriate cloud service providers (CSPs) has been caused by their increasing availability and diversity, as some excel in certain services while others specialize differently. The critical multi-criteria decision analysis (MCDA) problem of selecting appropriate cloud services in the uncertain cloud industry has led decision-making researchers to consider various multi-criteria decision-making (MCDM) methods for optimal cloud rank determination. This paper presents a review of essential decision criteria and their sub-criteria for CSP evaluation, and an overview of different MCDM methods for decision-making, including the strengths and weaknesses of these techniques to inform on current trends. Additionally, MCDM techniques used for Cloud service selection (CSS), and several methods deriving priority vectors from a Pairwise Consistency Matrix (PCM) in the Analytic Hierarchy Process (AHP) are detailed, highlighting their increased utilization in recent years. A growing emphasis on accurate and timely decision-making within given timeframes in CSS underscores the reliance on mathematical-based statistical models, data mining, machine learning, and MCDM methods by business analysts. The research provides insights into various MCDM techniques for alternative set-ranking, with AHP being the most extensively used. The discussion includes various PCM deriving methods in AHP, their advantages, and limitations, contributing to understanding current trends in decision-making methodologies."

UniDRM: Unified Data and Resource Management for Federated Vehicular Cloud Computing The burgeoning demand for computational resources in vehicular environments, triggered by the rise of intelligent transportation systems, has given rise to the federated vehicular cloud as an alternative to traditional cloud platforms for running high-intensity, delay-sensitive applications. Yet, capacity-constrained communication channels and limited resource capacities within vehicles pose challenges. To address these challenges, UniDRM, a unified data and resource management framework specifically tailored for the federated vehicular cloud is proposed. UniDRM classifies vehicles on the road based on their mobility and resource traits, then it segments rigorous task data through an analytical model and assigns them to vehicles in clusters for parallel execution. A trio of partitioning and scheduling schemes - time-aware, cost-aware, and reliability-aware - are also proposed to accommodate various task requirements. Realistic simulations are undertaken to measure the effectiveness of these schemes.

Conditional Preference Networks for Cloud Service Selection and Ranking With Many Irrelevant Attributes In the realm of cloud computing, cloud service selection and ranking are tasks that are increasingly challenging to be managed by end customers due to the rapid growth in the number of available services offered by cloud providers with varying quality of service. Furthermore, many services in the cloud are defined over a large set of attributes, with only a small subset of them being of interest to potential customers, while others are seen as irrelevant to customer selection and ranking strategies. In this work, the utilization of conditional preferential dependencies in the domain of cloud service selection and ranking is demonstrated. Specifically, Conditional Preference Networks (CP-nets) are utilized as a compact model for representing and reasoning with customer preferences and criteria interdependencies. It is shown how the best service can be selected and how a set of services can be ranked, in non-increasing order, given possibly incomplete information in the customer's CP-net. The feasibility of CP-nets as a mechanism for selecting and ranking hundreds of services defined on hundreds of attributes with complex interdependencies is proven through experimental results. It is demonstrated that CP-nets, a prominent preference model in the AI community, are an adequate model for customer preferences in cloud service selection and ranking. Furthermore, it is demonstrated that even in the case of incomplete CP-nets, it is still possible to reason about the best available service and rank an arbitrary set of services. Experimental results prove that CP-nets, and preferential dependencies in general, are indeed valuable tools for capturing preferences in domains with a large number of attributes. It is demonstrated that it is possible to compute attribute importance weights in an efficient way from purely comparative statements and complex interactions as represented by the CP-net graph. In the near future, plans are made to extend this work into learning relevant and irrelevant attributes in an interactive manner, using preference languages that are more expressive than CP-nets, and testing the usage of CP-nets with real-world data. The technical support of the University of Jeddah is gratefully acknowledged by the authors.

An Integrated MCDM Approach for Cloud Service Selection Based on TOPSIS and BWM Cloud Computing (CC) has been increasingly popularized as it provides a wide variety of customized and reliable computational services. With the rapid growth of this technology, more and more IT services providers are competed to offer high-quality and cost-effective cloud services that best fulfill their customers' needs. Given the vast diversity of these offers, the choice of the most appropriate Cloud Service Provider (CSP) has become a dilemma that confuses most cloud customers. Many diverged criteria have to be considered to precisely evaluate services offered by several CSPs, some of these criteria cannot be easily quantified such as usability and security. The selection of the best CSP is thus regarded as a complex Multi-Criteria Decision Making (MCDM) problem that needs to be addressed efficiently. Previous studies of this problem have employed MCDM methods that are either unfeasible when it is difficult or meaningless to quantify alternatives over criteria or computationally expensive and inconsistent when relative preferences of alternatives and criteria are used instead. In this paper, a novel MCDM approach is proposed that is feasible, efficient, and consistent using relative preferences of criteria and alternatives. The proposed approach incorporates Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) and the Best Worst Method (BWM) to rank CSPs using evaluation criteria characterizing their services. The integrated approach has been tested and validated through a use-case scenario which demonstrates its effectiveness and correctness. The proposed approach has also been compared to the most commonly used MCDM approach, Analytical Hierarchical Process (AHP). The results clearly show that the proposed approach outperforms AHP in terms of computational complexity and consistency; hence, it is concluded to be more efficient and reliable than AHP.

Evaluating and Ranking Cloud-Based E-Learning Critical Success Factors (CSFs) Using Combinatorial Approach Cloud computing has been regarded as one of the significant Information Technology (IT) tools, with its services increasingly adopted by many sectors for business support. The E-Learning system has been transformed to become more user-friendly, thanks to cloud computing, which has become a new IT paradigm. As a result, E-Learning usage is rapidly growing and being preferred over the conventional teaching-learning process on a large scale. This revolutionary change is attributed to advancements in digital technology, which have made the teaching-learning process flexible, easy, and convenient for effective knowledge transfer. Many factors of different dimensions that are of significant importance for cloud-based E-Learning success depend upon, hence they must be studied to analyze their level of importance and fulfill Cloud-based E-Learning positive effectiveness. The current research provides a detailed literature review for cloud-based E-Learning Critical Success Factors (CSFs) of the teaching-learning process. Further, the research employs the combinatorial approach to evaluate the diversified dimensions and CSFs of cloud-based E-Learning, helping in quantifying and comparing the influence of various dimensions and CSFs of cloud-based E-Learning. Four dimensions and fourteen factors have been identified through in-depth literature review and later evaluated for prioritization using a combinatorial approach. The influence of such dimensions and factors will help various stakeholders to plan their strategy and allocate resources for the betterment of knowledge transfer through cloud-based E-Learning. The dimensions and CSFs of the cloud-based E-Learning system play a very significant role in the successful teaching-learning process. Hence, it is very pertinent to investigate the influence of the dimensions and CSFs of the cloud-based E-Learning system to know their influence on the teaching-learning process. Once the influence of each CSF is ascertained, the stakeholders like university management, instructors, and students will be able to control the effect of each such dimension and CSF. The combinatorial approach of AHP-GDM and FAHP methodology may prove to be successful in classifying the CSFs in various grades of influence like High Influence, Moderate Influence, and Low Influence. This particular CSFs classification will further help stakeholders to deploy resources like time, money, service infrastructure enhancement, etc.

PROBLEM DEFINITION

Cloud computing is a kind of computing model on subscription basis. In cloud computing environment, there are a lot of cloud providers that present variety kind of services with different quality of services. Users have various kinds of applications that should be carried out on suitable cloud services. Consequently the users might encounter problems in choosing the best service. Hence selection of a method to compare services and to choose
best service has been regarded as a challenge.
At present, trusted cloud service selection involves the following challenges: accurate measurements of different types of QoS , objective descriptions of CUs preference demands of different QoS attributes , and rapid ranking of the cloud services according to the demands of CUs.

METHODOLOGY

The system frontend is developed using HTML5, CSS3, and Javascript. HTML5 ensures cross-platform support and organized CSS, while Javascript facilitates user interaction with the frontend and components, enabling communication with Ethereum nodes via HTTP connection.The backend will be developed with Python Flask, offering flexibility and a structured approach for project expansion. Jinja2 is utilized as a web template engine, and @app.route converts Python functions into Flask view functions, handling incoming web requests and generating HTTP responses. The framework assists cloud customers in selecting the best service based on QoS criteria, user experience, and service performance. It searches and ranks cloud services, providing detailed specifications and utilizing the MCDO method to track user requirement priorities and historical data.

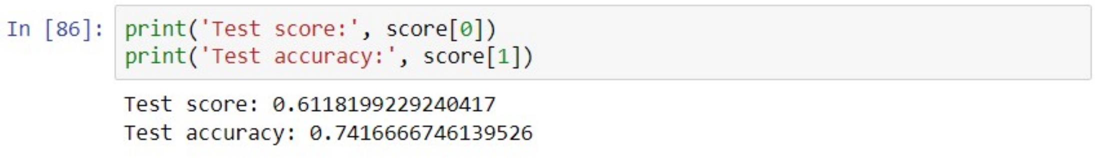
DATASET COLLECTION

This module is dedicated to collecting datasets essential for the project's various components. It gathers data related to cloud service providers, such as their functional and nonfunctional specifications, as well as Quality of Service (QoS) criteria values. Additionally, it compiles historical information on user requirements priorities to aid in the calculation of transition matrices. Furthermore, this module collaborates with a trusted third party to collect performance metrics of cloud services, including CPU performance, memory performance, storage performance, network latency, and bandwidth. These datasets are crucial for benchmarking cloud services and facilitating informed decision-making for cloud users.
The backend will be developed using Python Flask. Flask is flexible. It doesn’t require you to use any particular project or code layout. However, when first starting, its helpful to use a more structured approach. This means that the tutorial will require a bit of boilerplate up front, but its done to avoid many common pitfalls that new developers encounter, and it creates a project thats easy to expand on. Once you become more comfortable with Flask, you can step out of this structure and take full advantage of Flasks flexibility. Jinja2 is a web template engine which combines a template with a certain data source to render the dynamic web pages. Once you create the app instance, you use it to handle incoming web requests and send responses to the user. @app.route is a decorator that turns a regular Python function into a Flask view function, which converts the functions return value into an HTTP response to be displayed by an HTTP client, such as a web browser. You pass the value / to @app.route() to signify that this function will respond to web requests for the URL /, which is the main URL.
.

The framework aids cloud customers in selecting the optimal service based on QoS criteria, user experience, and performance. It searches and ranks cloud services, providing detailed specifications and leveraging the MCDO method to track user requirement priorities and historical data.Implementation of a recommendation system to suggest personalized cloud service options tailored to individual user preferences and workload requirements.

Cloud Services Benchmarking

This module is managed by a trustworthy third party, analysing cloud service performance metrics such as CPU, memory, storage performance, network latency, and bandwidth. It provides public access to benchmark results, influencing cloud service ranking based on repository information.



```
In [86]: print('Test score:', score[0])
         print('Test accuracy:', score[1])

         Test score: 0.6118199229240417
         Test accuracy: 0.7416666746139526
```

Figure 2. Test score and accuracy

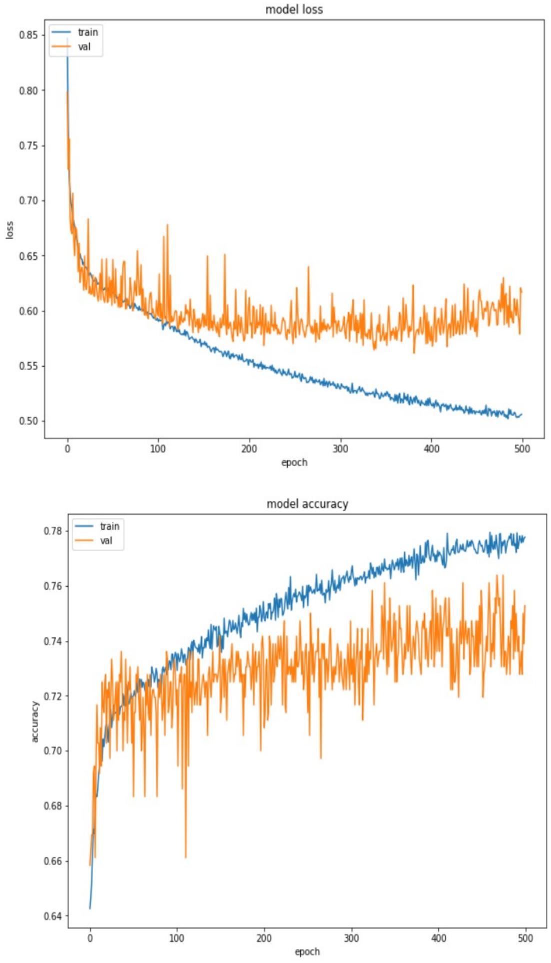
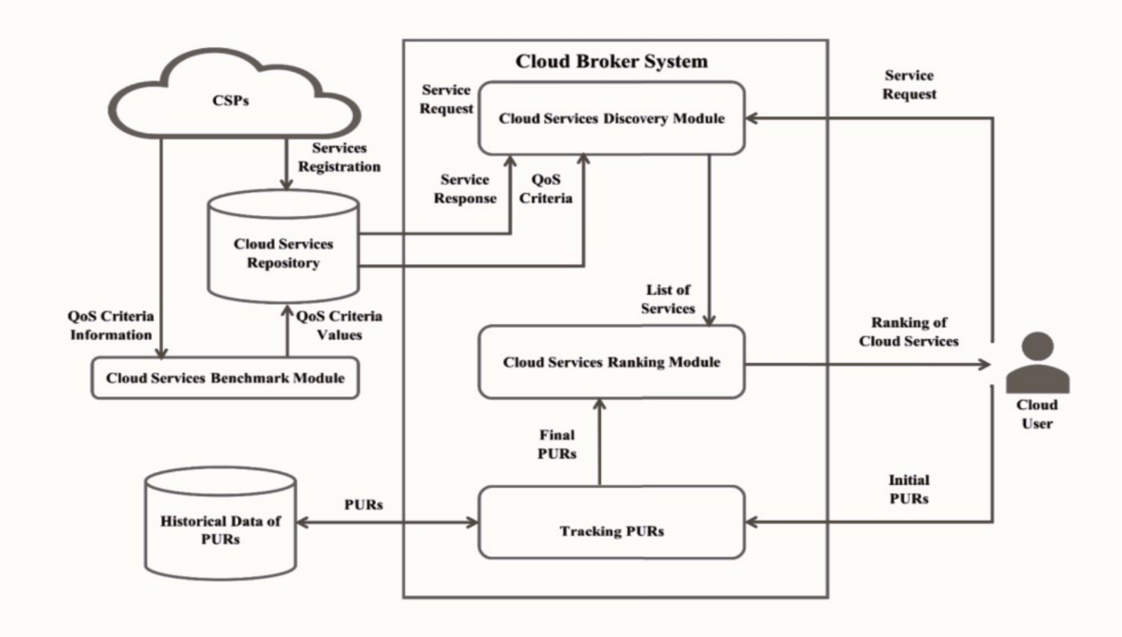


Figure 3. Model Loss and Accuracy of MCDO approach

FLOW DIAGRAM



IMPLEMENTATION

The project's implementation phase aims to address ranking fraud in the mobile app industry by transforming data exploration into a functional system. The study uncovers intricate patterns within the data using a range of methods, including database management systems, statistical analysis software, and data mining techniques. The Visages platform seamlessly integrates data from shared databases into application user interfaces, advocating for a data-centric approach.The preprocessing stage prepares the dataset for sentiment analysis using the Stanford CoreNLP library, Named Entity Recognition, coreference resolution, and dependency parsing. Standardized markers replace victim references, and irrelevant elements are eliminated. News articles are compiled for past events, and symbols, ASCII code, and punctuation are removed for accuracy.

Feature selection is crucial in the language processing phase, using the Term Frequency-Inverse Document Frequency (TF-IDF) method to optimize the feature set. Feature selection methods are categorized into filters, wrappers, and embedded/hybrid methods.The TF-IDF vectorizer transforms textual data into numerical vectors, streamlining the algorithm's ability to analyze and classify sentiments accurately. This systematic process promises valuable insights into user sentiments and contributes to a more reliable and secure mobile app ecosystem.

RESULTSAND DISCUSSION

The backend development of the system is anchored on Python Flask, offering a flexible framework for project layout and expansion. Flask's structured approach, though initially requiring boilerplate code, proves advantageous by steering clear of common developmental pitfalls and facilitating scalability. Leveraging Jinja2 for dynamic web page rendering enhances user experience. The Cloud Service Selection Module guides customers towards optimal cloud service providers by assessing user experience, service performance, and Quality of Service standards. This module systematically searches and ranks cloud services, incorporating detailed provider specifications and monitored QoS criteria. Through the MCDO method, it dynamically tracks user requirement priorities and calculates final priorities via a transition matrix, storing historical data for benchmarking. Managed by a trusted third party, the benchmark module ensures credible performance metrics, including low-level indicators like CPU performance and network latency. This integration of Flask, MCDO method, and third-party benchmarking reflects a comprehensive approach to backend development, decision-making, and benchmarking, ultimately facilitating efficient cloud service selection.

CONCLUSION

In today's world, several cloud services are provided by different providers, varying in their quality of service. Selecting a cloud service that meets the consumer's requirements becomes challenging. In this paper, we formulated the problem of finding the best Cloud service among a given set of services into a multi-attribute decision-making problem, where the Cloud services are alternatives and the QoS attributes are the criteria. We proposed using multi-attribute decision-making to rate Cloud services based on a set of standard quality service qualities.

FUTURE SCOPE

In our upcoming research, we intend to use different types of nonlinear membership functions for the Quality of Service (QoS) attributes of Cloud services. Membership functions are important in fuzzy logic because they help describe the link between input and output variables. By employing nonlinear membership functions, we aim to capture more complex and nuanced patterns in the QoS characteristics of Cloud services. This approach allows us better to represent the diverse and dynamic nature of QoS attributes, offering a more accurate and flexible framework for evaluating and understanding the performance and reliability of Cloud services. We seek to enhance our methodology by incorporating advanced mathematical functions that can better capture the intricacies of QoS in the cloud computing environment.

REFERENCES

[1] A. Bagherzandi, B. Hore, and S. Mehrotra, Search over Encrypted Data. Boston, MA, USA: Springer, 2011, pp. 1088–1093.

[2] H. Pham, J. Woodworth, and M. A. Salehi, ‘‘Survey on secure search over encrypted data on the cloud,’’ Concurrency Comput. Pract. Exper., vol. 31, p. 1– 15, Apr. 2019.

[3] C.Nagarajan and M.Madheswaran - ‘Experimental verification and stability state space analysis of CLL-T Series Parallel Resonant Converter’ - Journal of ELECTRICAL ENGINEERING, Vol.63 (6), pp.365-372, Dec.2012.

[4] C.Nagarajan and M.Madheswaran - ‘Performance Analysis of LCL-T Resonant Converter with Fuzzy/PID Using State Space Analysis’- Springer, Electrical Engineering, Vol.93 (3), pp.167-178, September 2011.

[5] C.Nagarajan and M.Madheswaran - ‘Stability Analysis of Series Parallel Resonant Converter with Fuzzy Logic Controller Using State Space Techniques’- Taylor & Francis, Electric Power Components and Systems, Vol.39 (8), pp.780-793, May 2011.

[6] C.Nagarajan and M.Madheswaran - ‘Experimental Study and steady state stability analysis of CLL-T Series Parallel Resonant Converter with Fuzzy controller using State Space Analysis’- Iranian Journal of Electrical & Electronic Engineering, Vol.8 (3), pp.259-267, September 2012.

[7] Nagarajan C., Neelakrishnan G., Akila P., Fathima U., Sneha S. “Performance Analysis and Implementation of 89C51 Controller Based Solar Tracking System with Boost Converter” Journal of VLSI Design Tools &amp; Technology. 2022; 12(2): 34–41p.

[8] C. Nagarajan, G.Neelakrishnan, R. Janani, S.Maithili, G. Ramya “Investigation on Fault Analysis for Power Transformers Using Adaptive Differential Relay” Asian Journal of Electrical Science, Vol.11 No.1, pp: 1-8, 2022.

[9] G.Neelakrishnan, K.Anandhakumar, A.Prathap, S.Prakash “Performance Estimation of cascaded h-bridge MLI for HEV using SVPWM” Suraj Punj Journal for Multidisciplinary Research, 2021, Volume 11, Issue 4, pp:750-756

[10] G.Neelakrishnan, S.N.Pruthika, P.T.Shalini, S.Soniya, “Perfromance Investigation of T-Source Inverter fed with Solar Cell” Suraj Punj Journal for Multidisciplinary Research, 2021, Volume 11, Issue 4, pp:744-749

[11] C.Nagarajan and M.Madheswaran, “Analysis and Simulation of LCL Series Resonant Full Bridge Converter Using PWM Technique with Load Independent Operation” has been presented in ICTES’08, a IEEE / IET International Conference organized by M.G.R.University, Chennai.Vol.no.1, pp.190-195, Dec.2007

[12] M Suganthi, N Ramesh, “Treatment of water using natural zeolite as membrane filter”, Journal of Environmental Protection and Ecology, Volume 23, Issue 2, pp: 520-530,2022

[13] M Suganthi, N Ramesh, CT Sivakumar, K Vidhya, “Physiochemical Analysis of Ground Water used for Domestic needs in the Areabof Perundurai in Erode District”, International Research Journal of Multidisciplinary Technovation, pp: 630-635, 2019

[14] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, ‘‘Searchable symmetric encryption: Improved definitions and efficient constructions,’’ in Proc. 13th ACM Conf. Comput. Commun. Secur., New York, NY, USA, 2011, pp. 79–88, 2006.

[15] M. Zeng, H.-F. Qian, J. Chen, and K. Zhang, ‘‘Forward secure public key encryption with keyword search for outsourced cloud storage,’’ IEEE Trans. Cloud Comput., early access, Sep. 27, 2019, doi: 10.1109/TCC.2019.2944367.

[16] S. Kamara, C. Papamanthou, and T. Roeder, ‘‘Cs2: A searchable cryptographic cloud storage system,’’ Microsoft Res., Redmond, WA, USA, Tech. Rep. MSR-TR-2011-58, May 2011.

[17] W. Song, B. Wang, Q. Wang, Z. Peng, W. Lou, and Y. Cui, ‘‘A privacypreserved full-text retrieval algorithm over encrypted data for cloud storage applications,’’ J. Parallel Distrib. Comput., vol. 99, pp. 14–27, Jan. 2017.

[18] A. G. Kumbhare, Y. Simmhan, and V. Prasanna, ‘‘Designing a secure storage repository for sharing scientific datasets using public clouds,’’ in Proc. 2nd Int. workshop Data Intensive Comput. Clouds, 2011, pp. 31–40.

[19] Z. Yang, J. Tang, and H. Liu, ‘‘Cloud information retrieval: Model description and scheme design,’’ IEEE Access, vol. 6, pp. 15420–15430, 2018.

[20] H. Yin, J. Zhang, Y. Xiong, L. Ou, F. Li, S. Liao, and K. Li, ‘‘CP-ABSE: A ciphertext-policy attribute-based searchable encryption scheme,’’ IEEE Access, vol. 7, pp. 5682–5694, 2019.