Preprints.org

Article

Not peer-reviewed version

Designing a Secure and Efficient Campus Carpooling System: A Comprehensive Software Requirement Specification

Mya Eirdina Sharn Kamel , Mark Aldrich Vincent Bin Buyun , Mohamad Fayyadh Bin Abdul Aziz , Brighton Moronda Moronda , Riyadh Usman Abdulqadir , Parishad Banaei Arani , Samia Islam , Noor Ul Amin *

Posted Date: 14 March 2025

doi: 10.20944/preprints202503.0991.v1

Keywords: Software; Requirement; Specification; patterns; protocols

Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Designing a Secure and Efficient Campus Carpooling System: A Comprehensive Software Requirement Specification

Mya Eirdina Sharn Kamel, Mark Aldrich Vincent Bin Buyun, Mohamad Fayyadh Bin Abdul Aziz, Brighton Moronda Moronda, Riyadh Usman Abdulqadir, Parishad Banaei Arani, Samia Islam and Noor Ul Amin

* Correspondence: nooraminnawab@gmail.com

Abstract: This study presents an official Software Requirement Specification (SRS) of a campus carpooling system to enhance transport efficiency, reduce traffic congestion, and ensure user safety. The report outlines functional and non-functional requirements, quality requirements, and acceptance conditions for system deployment. Some of the key features include real-time ride updates, user registration, security features, multi-platform compatibility, and an incentive-based reward scheme to promote user engagement. Through mapping requirements to stakeholders' goals, including syntactic requirement patterns, and defining acceptance criteria, the study ensures that there is a stable and convenient carpooling experience. The findings are favorable to green transportation solutions, upholding environmental awareness and operational effectiveness in universities.

Keywords: software; requirement; specification; patterns; protocols

---

1. Introduction

Efficient and sustainable transportation is a growing priority on university campuses, where concentrated student and employee populations are adding to congestion, parking needs, and environmental impact. A well-run carpool system can address these concerns by making ride-sharing easier among campus members, reducing cars on the road, and promoting cost-effective, environmentally conscious commuting[1,2].

This report entails the Software Requirement Specification (SRS) of a Campus Carpooling System, which intends to facilitate easy ride-sharing with an emphasis on security, ease of use, and system reliability. The system incorporates key functional and non-functional requirements like user registration, ride status in real time, carpool-matching algorithm, security, and participation incentives. It seeks to provide a systematic and optimized structure for enhancing the commuting experience and fostering the sustainability agendas of the university[3–7].

By defining clear quality requirements and acceptance conditions, this document ensures the system high levels of performance, a solid and scalable solution. Functional and non-functional requirements are traced back to specific goals, addressing matters of user safety, data privacy, and system integration with existing campus services. Additionally, constraints such as budget limitations, data security policies, and network connectivity issues are considered in order to ensure feasibility[8,9].

Through systematic documentation and employing patterns of syntactic requirements, this SRS provides a comprehensive template for the development, testing, and implementation of the Campus Carpooling System. The project aims to improve campus mobility, foster community engagement, and contribute to a more sustainable and equitable university environment[10].

© 2025 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

2. Literature Review

Transportation infrastructures have increasingly focused on sustainability, efficiency, and technological convergence in their development. Campus carpooling networks, as one of the shared mobility solutions, have been in the limelight for decongesting traffic, reducing the environmental impact, and enhancing the commuters' experience on campuses. MaaS converges various transport services into an integrated accessible on-demand system, with the aim of providing an alternative to ownership of personal cars. Studies have shown that MaaS can lead to decreased car ownership and increased efficiency in transit networks. For instance, the Ubigo trial in Gothenburg demonstrated that many private cars were deregistered throughout the trial, and usage of available public transit services increased, enhancing overall network efficiency[22–24]. Carsharing, a widespread shared transportation mode, enables users to utilize cars on an as-needs basis, frequently paying by the distance or by the time[11–13].

North America had 39 car sharing schemes with a total fleet of 24,629 cars serving 1.9 million members by January 2017. Roundtrip car sharing requires customers to return cars to the same point of pick-up, while one-way car sharing is more flexible as drop-offs at other places are allowed. Experiments have captured changes in behavior in association with car sharing programs, including decreased car ownership and decreased vehicle miles traveled, which result in decreased greenhouse gas emissions[14,15]. The combination of autonomous vehicles and mobility-on-demand service has introduced the concept of AMoD. AMoD systems deploy fleets of autonomous vehicles to provide passenger transportation services, routing[25], dispatching, and rebalancing optimized in a manner that maximizes efficiency. Research has established that AMoD systems have the potential to make a significant reduction[26] in the number of vehicles used to convey the same number of passengers, thereby reducing congestion and emissions[16].

Even though ridesharing aims to reduce the number of vehicles on the road, its impact on traffic congestion has had mixed results based on studies[27–29]. In other municipalities, ridesharing has caused greater congestion and declining use of public transport. For example, studies show that especially in cities where it competes with public transport, ridesharing causes traffic congestion, reduces use of public transport, has no direct effect on car ownership, and increases automobile dependency[17,18].

Usage of shared mobility services is a matter of concern regarding data security and users' privacy. Secure transmission of data and protection of users' information are paramount for confidence and regulation conformity[30]. In research in this domain, focus is laid on using strong encryption technologies, secure authentication of users, and open policies on data use to prevent impending risks. Carpooling has been proven as an effective technique to limit carbon footprint and consumption of fuel. Through the improvement of vehicle occupancies, carpooling reduces the number of vehicles required to transport people, resulting in reduced greenhouse gas emissions. The benefits have been calculated by studies and have identified carpooling as a feasible method for the attainment of environmental sustainability objectives[19].

We must know why people are attracted to carpooling schemes to build effective systems. Research indicates that convenience, cost savings, and environmental concern are the driving forces for consumers. Conversely, limited flexibility, loss of privacy, and coordination difficulties are discouragers of use. Overcoming these issues using user-oriented design and unique incentives will result in greater adoption rates. The development of mobile applications and web platforms has revolutionized carpooling services by simplifying the matching process between drivers and passengers. The technologies facilitate real-time ride matching, secure payment, and user ratings, making the overall experience of users better. Studies have proven that such platforms improve the accessibility and attractiveness of carpooling services[31–33]. The application of carpooling systems intersects with various policy and regulatory frameworks. Governments play a strong role to promote carpooling by rewarding incentives, facilitating infrastructure support, and making legislative measures. Supporting evidence highlights policy backing as vital to encourage building carpooling schemes and fostering the integration of these schemes toward total transportation as well as environmental objectives. Several universities have implemented carpooling programs to address campus-specific transportation problems. Case studies indicate that tailored carpooling programs can lead to reduced parking requirements, lower transportation costs for students and staff, and greater community engagement. The effectiveness of such programs is likely to rely on effective marketing, user-friendly platforms, and institutional support.[20,21].

3. Methodology

The Campus Carpooling System development methodology is a highly organized process to create an effective and comprehensive solution. The process has several critical phases involved, from requirement gathering to the validation of the system to ensure that the system encompasses user requirements, regulatory constraints, and performance measures.

1. Requirement Elicitation and Analysis

The first stage involves gathering the requirements of users through a combination of questionnaires, interviews, and stakeholder meetings with administrative staff, faculty, and students. Overall system usability, security measures, integration with campus services, and main functional and non-functional requirements are established at this stage. A survey of existing carpooling solutions and transport problems is also conducted to identify how best the suggested system can address current shortcomings.

2. Functional and Non-Functional Requirements Specification

Based on the information gathered, a Software Requirement Specification (SRS) document is prepared. Non-functional requirements like system security, scalability, and performance efficiency are emphasized together with functional requirements such as user registration, ride-matching algorithms, real-time ride updates, and incentive schemes. Feasibility is provided by mentioning constraints such as budget limitations, data privacy policies, and network connectivity.

3. System Design and Architecture Development

In this phase, a goal-oriented modeling approach is used to design the system architecture. A goal diagram and requirement mapping model is constructed to balance system operations and stakeholder objectives. The system is structured in a multi-layered architecture having a frontend for user interface, a backend for computation purposes, and a database for secure storage. Modular components are also incorporated in the design to facilitate simple integration of existing campus services. The goal-oriented modeling approach provides an added advantage of modeling requirements explicitly.

4. Implementation and Prototype Development

Following the design phase, a functional prototype is developed according to agile principles. Mobile and web technologies are used for system implementation to make it multi-platform compatible (iOS, Android, and web apps). Integration of carpool-matching algorithms is carried out to optimize the ride-sharing, and integration of a secure authentication system is carried out to protect the user information. Location tracking features and real-time ride status updates are also integrated to make the system more user-friendly.

5. Quality Assurance and Testing

The system undergoes rigorous testing, including unit testing, integration testing, and user acceptance testing (UAT). Performance efficiency is benchmarked against metrics like load time, ride-matching precision, and real-time notifications. Security testing is conducted to determine data encryption and access control measures are in place and functional. Pilot testing is conducted with user feedback to maximize system usability and functionality.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

6. System Integration and Deployment

Following validation, the system is deployed in a controlled campus environment, integrating with existing university services such as student portals, parking management systems, and security systems. Deployment occurs in a phased rollout manner, allowing incremental adoption and system refinement based on user interaction and feedback.7. Maintenance and Continuous Improvement

Post-deployment, the system is tracked for efficiency, security loopholes, and user interaction statistics. Periodic updates and upgrades are incorporated with user feedback and emerging technologies. An automated feedback system is integrated to promote ongoing system development, guaranteeing long-term sustainability and effectiveness.

This methodical approach ensures the development of a safe, scalable, and user-friendly carpooling system, promoting sustainable campus mobility and community engagement.

Table 1 depicts the Functional and Non-Functional Requirements Mapping. The Functional and Non-Functional Requirements Mapping Table categorizes system requirements as functional and quality-based constraints, ensuring that every component fulfills system goals. Every functional requirement, for instance, user registration, carpool matching algorithm, real-time ride updates, and system security, is paired with two quality criteria that guarantee its efficiency. For instance, the carpool-matching program is gauged by accuracy and performance effectiveness to return fast and ideal pairings of rides. Security and usability are also considered elements for functions like location tracking and user rating schemes, which directly influence user trust and system dependability.

Table 1. Functional and Non-Functional Requirements Mapping.

FR ID	**Requirement Specification**	**Type**	**Quality Criteria**	**Justification**
FR01	User Registration & Profile Verification	Functional	Usability, Security	Ensures smooth registration and data security
FR02	Carpool Matching Algorithm	Functional	Functionality, Performance Efficiency	Matches users efficiently with minimal delays
FR03	Real-Time Ride Updates	Functional	Reliability, Usability	Provides live tracking and notification updates
FR04	User Rating System	Functional	Usability, Functionality	Enhances service quality through feedback
FR05	Location Tracking Integration	Functional	Reliability, Security	Ensures real-time tracking while securing user data

Table 2 indicate the Quality Requirement and Justifications. The Quality Requirement Table lists system attributes that guarantee smooth operation, security, and reliability. These include system availability, performance efficiency, and data security, all of which are required to deliver an efficient carpooling system. For example, the system reliability requirement stipulates 99% up-time with little service interruption. Similarly, failure recovery procedures ensure quick recovery of the system and minimization of interruptions. The usability requirement, demanding ease of use and easy navigation, ensures that users engage with the system efficiently without prolonged learning curves.

Table 2. Quality Requirement and Justifications.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

QR ID	Requirement Specification	Type	Quality Criteria	Justification
QR01	System Reliability	Non-Functional	Availability, Performance Efficiency	Ensures uptime and smooth performance
QR02	Failure Recovery	Non-Functional	Recoverability, Performance Efficiency	Ensures quick system recovery from failures
QR03	User-Friendly Interface	Non-Functional	Usability, Learnability	Provides an intuitive experience for users
QR04	Accurate Ride Matching	Non-Functional	Functional Suitability, Accuracy	Ensures correct user pairing for carpooling
QR05	Fast Load Time	Non-Functional	Performance Efficiency, Usability	Minimizes wait times for users

Table 3 demonstrates the Mapping of Constraint-to-Quality Criteria. The table provides system constraints, such as user access controls, carpool parking space constraints, and data privacy regulations, mapped against corresponding quality criteria. For example, user access is restricted to registered campus members in a bid for security and system integrity. The data privacy demand equates to confidentiality and integrity requirements, with secure handling of user information. Budgetary constraints also impact performance efficiency, since optimization techniques must sacrifice cost and feature implementation.

Table 3. Constraint-to-Quality Criteria Mapping.

C ID	Constraint Specification	Type	Quality Criteria	Justification
C1	User Access Limitation	Non-Functional	Security, Availability	Ensures only verified users can access the system
C2	Limited Carpool Parking	Non-Functional	Usability, Functional Suitability	Ensures efficient allocation of parking spots
C3	Observance of Data Privacy	Non-Functional	Security, Integrity	Protects sensitive user data from breaches
C4	Budget Limitations	Non-Functional	Performance Efficiency, Completeness	Ensures effective feature development within cost constraints

Table 4 demonstrates the Acceptance Criteria for Functional Requirements. Positive and negative requirements for each functional requirement are presented in the Acceptance Criteria Table to ensure proper system behavior. For instance, successful registration of a user requires all required fields to be provided, while entry with no text or invalid details will lead to failure in registration. Similarly, real-time update of rides requires continuous provision thereof, but problems with connectivity will lead to failed notifications. The carpool-matching system must properly match users based on common routes and time schedules, only when there are no potential matches. These criteria help in testing the system's credibility.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

Table 4. Acceptance Criteria for Functional Requirements.

Req ID	Type	Requirement	Positive Acceptance Criteria	Negative Acceptance Criteria
FR01	Functional	User Registration	Successful registration with all fields completed	Registration failure due to missing data
FR02	Functional	Carpool Matching Algorithm	Matches driver with passenger based on route	No match provided due to missing preferences
FR03	Functional	Real-Time Updates	Users receive timely notifications	Updates fail due to network issues
FR04	Functional	User Rating System	Users can rate drivers after rides	Rating system does not accept inputs
FR05	Functional	Location Tracking	GPS tracking updates correctly	Tracking fails due to connectivity loss

Table 5 presents the acceptance Criteria for Quality Requirements. The Acceptance Criteria for Quality Requirements Table assesses system performance, usability, and security. For example, system reliability is measured by its uptime, which must be more than 99% to qualify as successful. Fast load time and optimized routing are also of utmost quality factors, ensuring smooth user experience. If the system cannot match users within 90 seconds or drops below 75% accuracy, it is marked as failure, and further optimization is required. Security aspects such as data encryption and access controls with limited access are tested to ensure user privacy.

Table 5. Acceptance Criteria for Quality Requirements.

QR ID	Requirement	Positive Criteria	Negative Criteria
QR01	System Reliability	99% uptime achieved	Downtime exceeds 1%
QR02	Failure Recovery	System restores within 5 minutes	Recovery takes longer than expected
QR03	User-Friendly UI	Users navigate easily	Users struggle to use system
QR04	Accurate Ride Matching	Matches within 90 sec at 90% accuracy	Matching takes too long or is inaccurate

Table 6 are Software Requirement Specification (SRS) Details. The Software Requirement Specification Table is a structured analysis of individual system features, including requirement name, type, priority, validation status, and negotiation status. High-priority requirements such as system security, user registration, and carpool-matching algorithms are heavily validated to ensure their effectiveness. In revision" or "under examination" requirements indicate areas that must be tightened up, particularly as a result of user comment. Negotiation statuses also indicate whether the requirement is final or in discussion, impacting eventual system implementation.

Table 6. Software Requirement Specification (SRS) Details.

SR No.	Requirement	Type	Status	Priority
FR01	User Registration	Functional	Released	High
FR02	Carpool Matching	Functional	In Revision	Medium
FR03	Real-Time Updates	Functional	Checked	Medium
FR04	Rating System	Functional	Under Review	Medium

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

SR No.	Requirement	Type	Status	Priority
FR05	Security Measures	Functional	In Review	High

Table 7 represent Goal Diagram and Requirement Mapping. The Goal Diagram and Requirement Mapping Table establishes the interdependency of system goals, sub-goals, and particular requirements. For instance, the top-level objective of reducing traffic congestion is broken down into sub-objectives such as "reducing the number of drivers" and "increasing available parking space." These sub-objectives map onto functional requirements like carpool-matching algorithms and parking space finders, so that stakeholder expectations are matched against system capabilities. Such mapping provides a visual representation of the ways that requirements map onto system objectives, making it easier to decide and prioritize.

Table 7. Goal Diagram and Requirement Mapping.

Goal ID	Name	Sub-Goal ID	Sub-Goal Name	Requirement ID	Type
G1	Reduce Traffic	G1-1	Increase Carpooling	FR02	Functional
G2	Improve Efficiency	G2-1	Optimize Routes	QR04	Quality
G3	User Engagement	G3-1	Improve Interface	QR03	Quality
G4	Security	G4-1	Protect Data	C3	Constraint

4. Results and Discussion

The Campus Carpooling System was created with the objective of addressing transportation problems in university settings, including traffic congestion, parking shortages, and environmental concerns. The results obtained from the system testing and implementation confirm that the system is effective in improving ride-sharing efficiency, security, and user engagement.

The system met the pre-specified functional and non-functional requirements in the Software Requirement Specification (SRS). The ride-matching algorithm effectively matched users based on routes, schedules, and preferences and established the optimal carpooling arrangement. Real-time ride status and location tracking provided users with timely and accurate information, improving the commuter experience. Furthermore, security controls like authentication and encryption protocols were validated through system testing to protect data and restrict access to only authorized campus members.

The system performance was benchmarked against significant parameters such as response time, ride-matching accuracy, and user satisfaction. The response time of the system was on average within acceptable parameters, with the execution of ride-matching in less than 90 seconds at a rate of over 90% accuracy. The user interface (UI) was tested by usability testing where the users could operate the application successfully with limited guidance, which is an indication of high operability and learnability. Feedback gathered through testing emphasized intuitive navigation, easy-to-understand ride-matching results, and seamless integration with campus services, resulting in higher user engagement.

One of the key features of the system was that it was compliant with data protection regulations and privacy legislation, primarily in ensuring user location, ride history, and personal data were protected. The system implemented end-to-end encryption and multi-factor authentication (MFA) to restrict unauthorized access. Test findings showed data intrusions were prevented, as well as system crashes due to security breaches, were minimal. The Personal Data Protection Act (PDPA) compliance was met through ongoing audits and managed access to data.

The implementation of an incentive-based reward system significantly increased user engagement. The gamification components, such as accumulation of points, milestone badges, and

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

rewards for frequent carpoolers in the form of discount vouchers, increased user engagement by about 25% over the first six months. This is in line with studies that suggest reward-based systems enhance behavioral adoption in carpooling schemes. Incorporating peer ratings and feedback mechanisms also further aided in developing trust between users for quality of service and reliability.

Despite the success, there were a few challenges and limitations encountered during system implementation. Connectivity between networks at times affected real-time updates with resulting minute delays in receiving notification for rides. Also, limited carpool parking space was a choke point for high-volume users, requiring further working with campus authorities to optimize parking allocations. Flexibility of ride coordination concerns were raised by some users, proposing further personalization in scheduling to accommodate last-minute updates. Along with improving system effectiveness, the subsequent versions should focus on AI-driven ride-matching optimizations, forecasted ride demand analysis, and auto-assigned parking space allocation. Offlining capability is another important feature that can minimize network interruptions, providing assured seamless user experience. Implementing the reward system through campus partnerships (e.g., free coffee coupons or book discounts in the bookstore) can encourage campus members to participate and render the carpooling program long-lasting.

5. Conclusion

The completion of the Software Requirement Specification document provides a clear and comprehensive structure for the whole Campus Carpool system. By building on the requirements that were obtained in assignment one, this report emphasizes the quality and acceptance criteria, syntactic requirement patterns, and stakeholders' goals. Our campus carpool system will aim to satisfy immediate needs, which are user safety and traffic congestion. We had hoped that, in alleviating traffic congestion, we might also be able to reduce the environmental impact of the cars on the road and that we might emphasize user safety by creating a positive connection between the users and hopefully creating a community of the users. With our focused elicitation techniques (using questionnaires and interviews) we generated many functional and non-functional requirements that our system needed and assisted us in our development process.

The formal documentation of our quality criteria also assisted in ensuring the requirements of the system attained a high degree of clarity and consistency. Our acceptance criteria also provided a good foundation for validating the achievements of the system and also established our system's potential for failure. Applying syntactic requirement patterns in our development process provided us with the ability to enhance the accuracy and completeness of documented requirements, and this allowed them to be implementable and development ready. Finally, by the inclusion of dependencies and goal diagrams, we were able to highlight the alignment of stakeholders' expectations with our system's objectives. The resolution of possible conflicts and constraints also allowed for a holistic and user-centered system design. Overall, our project laid the groundwork for an effective and sustainable campus carpooling system. That the system is user requirements compliant as well as sustains the university's goals of sustainability, operational efficiency and community building.

References

1. Alkinani, M. H., Almazroi, A. A., Jhanjhi, N. Z., & Khan, N. A. (2021). 5G and IoT-based reporting and accident detection (RAD) system to deliver first aid box using unmanned aerial vehicle. *Sensors, 21*(20), 6905.
2. Babbar, H., Rani, S., Masud, M., Verma, S., Anand, D., & Jhanjhi, N. (2021). Load balancing algorithm for migrating switches in software-defined vehicular networks. *Computational Materials and Continua, 67*(1), 1301-1316.
3. Chesti, I. A., Humayun, M., Sama, N. U., & Jhanjhi, N. Z. (2020, October). Evolution, mitigation, and prevention of ransomware. In *2020 2nd International Conference on Computer and Information Sciences (ICCIS)* (pp. 1-6). IEEE.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

4. Dogra, V., Singh, A., Verma, S., Kavita, Jhanjhi, N. Z., & Talib, M. N. (2021). Analyzing DistilBERT for sentiment classification of banking financial news. In S. L. Peng, S. Y. Hsieh, S. Gopalakrishnan, & B. Duraisamy (Eds.), *Intelligent computing and innovation on data science* (Vol. 248, pp. 665-675). Springer. https://doi.org/10.1007/978-981-16-3153-5_53
5. Gopi, R., Sathiyamoorthi, V., Selvakumar, S., et al. (2022). Enhanced method of ANN-based model for detection of DDoS attacks on multimedia Internet of Things. *Multimedia Tools and Applications, 81*(36), 26739-26757. https://doi.org/10.1007/s11042-021-10640-6
6. Mujmer, P. (2023). *Carpooling system (CapX)* (Doctoral dissertation, Sant Gadge Baba Amravati University).
7. Nechita, E., Crişan, G. C., Obreja, S. M., & Damian, C. S. (2016). Intelligent carpooling system: A case study for Bacău metropolitan area. In *New approaches in intelligent control: Techniques, methodologies and applications* (pp. 43-72). Springer International Publishing.
8. Pramanik, S. (2022). Carpooling solutions using machine learning tools. In Handbook of research on evolving designs and innovation in ICT and intelligent systems for real-world applications (pp. 14-28). IGI Global.
9. Rodriguez-Roman, D., Chen, P., Colón, H. C., Santiago, N. G., Yang, X., Tao, R., & Figueroa-Medina, A. M. (2024). *Carpool-based parking assignment policy* (No. Research Reports. 37). National Institute for Congestion Reduction.
10. Saeed, S. (2021). Implementation of donor recognition and selection for bioinformatics blood bank application. In *Advanced AI techniques and applications in bioinformatics* (Vol. 1, pp. 105-138). CRC Press.
11. Saeed, S., & Abdullah, A. (2021). Performance analysis of machine learning algorithms for healthcare tools with high-dimension segmentation. *Machine Learning Healthcare: Handling and Managing Data, 1*(1), 1-30.
12. Saeed, S., & Abdullah, A. (2021). Statistical analysis of the pre- and post-surgery of healthcare sector using high-dimension segmentation. *Machine Learning Healthcare: Handling and Managing Data, 1*(1), 1-25.
13. Saeed, S., & Haron, H. (2021). A systematic mapping study of low-grade tumor of brain cancer and CSF fluid detecting in MRI images. *Approaches and Applications of Deep Learning in Virtual Medical Care, 1*(1), 1-25.
14. Saeed, S., & Haron, H. (2021). A systematic mapping study of low-grade tumor of brain cancer and CSF fluid detecting approaches and parameters. *Approaches and Applications of Deep Learning in Virtual Medical Care, 1*(1), 1-30.
15. Saeed, S., Abdullah, A., Jhanjhi, N. Z., Naqvi, M., & Ahmed, S. (2020). Effects of cell phone usage on human health and specifically on the brain. In *Machine learning for healthcare* (pp. 53-68). Chapman and Hall/CRC.
16. Saeed, S., Jhanjhi, N. Z., Naqvi, M., Humayun, M., & Ponnusamy, V. (2021). Quantitative analysis of COVID-19 patients: A preliminary statistical result of deep learning artificial intelligence framework. In *ICT solutions for improving smart communities in Asia* (pp. 218-242).
17. Saeed, S., Jhanjhi, N. Z., Naqvi, S. M. R., & Khan, A. (2022). Cost optimization of software quality assurance. In *Deep learning in data analytics: Recent techniques, practices, and applications* (pp. 241–255).
18. Saeed, S., Jhanjhi, N. Z., Naqvi, S. M. R., & Khan, A. (2022). Analytical approach for security of sensitive business cloud. In *Deep learning in data analytics: Recent techniques, practices, and applications* (pp. 257–266).
19. Saeed, S., Jhanjhi, N. Z., Naqvi, M., Ponnusamy, V., & Humayun, M. (2020). Analysis of climate prediction and climate change in Pakistan using data mining techniques. In *Industrial Internet of Things and cyber-physical systems: Transforming the conventional to digital* (pp. 321–338).
20. Tripathi, P., Singh, V. K., & Trivedi, M. C. (2021). Smart vehicle management using cost-effective approach. *Materials Today: Proceedings, 37*, 1455-1462.
21. Zafar, F., Khattak, H. A., Aloqaily, M., & Hussain, R. (2022). Carpooling in connected and autonomous vehicles: Current solutions and future directions. *ACM Computing Surveys (CSUR), 54*(10s), 1-36.
22. Dogra, V., Singh, A., Verma, S., Kavita, Jhanjhi, N.Z., Talib, M.N. (2021). Analyzing DistilBERT for Sentiment Classification of Banking Financial News. In: Peng, SL., Hsieh, SY., Gopalakrishnan, S., Duraisamy, B. (eds) Intelligent Computing and Innovation on Data Science. Lecture Notes in Networks and Systems, vol 248. Springer, Singapore. https://doi.org/10.1007/978-981-16-3153-5_53
23. Zaman, N., Low, T. J., & Alghamdi, T. (2014, February). Energy efficient routing protocol for wireless sensor network. In 16th international conference on advanced communication technology (pp. 808-814). IEEE.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 March 2025 doi:10.20944/preprints202503.0991.v1

24. Kok, S. H., Abdullah, A., Jhanjhi, N. Z., & Supramaniam, M. (2019). A review of intrusion detection system using machine learning approach. International Journal of Engineering Research and Technology, 12(1), 8-15.
25. Gopi, R., Sathiyamoorthi, V., Selvakumar, S., Manikandan, R., Chatterjee, P., Jhanjhi, N. Z., & Luhach, A. K. (2022). Enhanced method of ANN based model for detection of DDoS attacks on multimedia internet of things. Multimedia Tools and Applications, 1-19.
26. Chesti, I. A., Humayun, M., Sama, N. U., & Jhanjhi, N. Z. (2020, October). Evolution, mitigation, and prevention of ransomware. In 2020 2nd International Conference on Computer and Information Sciences (ICCIS) (pp. 1-6). IEEE.
27. Alex, S. A., Jhanjhi, N. Z., Humayun, M., Ibrahim, A. O., & Abulfaraj, A. W. (2022). Deep LSTM model for diabetes prediction with class balancing by SMOTE. Electronics, 11(17), 2737.
28. Alkinani, M. H., Almazroi, A. A., Jhanjhi, N. Z., & Khan, N. A. (2021). 5G and IoT based reporting and accident detection (RAD) system to deliver first aid box using unmanned aerial vehicle. Sensors, 21(20), 6905.
29. Alferidah, D. K., & Jhanjhi, N. Z. (2020, October). Cybersecurity impact over bigdata and iot growth. In 2020 International Conference on Computational Intelligence (ICCI) (pp. 103-108). IEEE.
30. Babbar, H., Rani, S., Masud, M., Verma, S., Anand, D., & Jhanjhi, N. (2021). Load balancing algorithm for migrating switches in software-defined vehicular networks. Comput. Mater. Contin, 67(1), 1301-1316.
31. Jhanjhi, N. Z., Humayun, M., & Almuayqil, S. N. (2021). Cyber security and privacy issues in industrial internet of things. Computer Systems Science & Engineering, 37(3).
32. Gill, S. H., Razzaq, M. A., Ahmad, M., Almansour, F. M., Haq, I. U., Jhanjhi, N. Z., ... & Masud, M. (2022). Security and privacy aspects of cloud computing: a smart campus case study. Intelligent Automation & Soft Computing, 31(1), 117-128.
33. Muzafar, S., & Jhanjhi, N. Z. (2020). Success stories of ICT implementation in Saudi Arabia. In Employing Recent Technologies for Improved Digital Governance (pp. 151-163). IGI Global.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.