Connect+: A customized tutoring and productivity web platform for philomaths

Reshu Agarwal, Yash Khandelwal, Vedika Mishra, Vinay Patidar
Indore Institute of Science and Technology

Abstract- **Connect+ is a unique platform designed to help students achieve their learning goals efficiently by providing specialized roadmaps. Inspired by the challenges students face in navigating academic and professional journeys, particularly in the tech domain, it adapts to individual time constraints and skill levels, creating personalized paths that balance reality and ambition. What sets Connect+ apart is its ability to simplify overwhelming information into structured, actionable steps. By analyzing user inputs such as available time, existing skills, and target outcomes, the platform delivers focused roadmaps aligned with the user's aspirations. This approach not only saves time but also boosts confidence by breaking down complex goals into manageable milestones. The development of Connect+ was very enlightening. It involved addressing real-world problems, understanding diverse user needs, and transforming these insights into a practical, user-friendly solution. The process deepened our understanding of problem-solving, design, and implementation.**

Keywords: **Connect+, time constraints, roadmap, technology, process model.**

I.INTRODUCTION

Adaptive learning systems represent a paradigm shift in education by employing machine learning algorithms to customize educational content to individual learners. These systems enhance engagement and learning efficiency by catering to a learner's specific skill level, knowledge gaps, and preferred pace. In the context of Computer Science (CS) education, such platforms are particularly impactful. Research by Gikandi and Morrow (2016) highlights how adaptive platforms provide personalized suggestions and immediate feedback, crucial for mastering foundational CS skills before advancing to complex topics. However, these systems face challenges, including the need for extensive datasets to train algorithms effectively. Additionally, concerns over data privacy are significant, as user data is essential for customization (Jiang et al. 2018). Studies like those by Wu et al. (2022) proposed differential privacy-focused adaptive models using federated learning, which protects user information while maintaining system effectiveness. Competency-based education (CBE) emphasizes the mastery of specific skills or competencies before progression. This model is particularly relevant to CS education, where foundational topics such as algorithms and data structures must be solidly understood. Self-paced learning models complement CBE by allowing learners to proceed at their own pace, accommodating individual schedules and commitments. Zilberman et al. (2021) demonstrate that self-paced learning enhances retention and satisfaction, especially when integrated with adaptive content. Together, these models form a robust foundation for personalized and effective CS education, aligning learning processes with individual needs and career goals.

The integration of recommendation systems in e-learning platforms has transformed how learners access educational content. These systems utilize algorithms like collaborative filtering and content-based filtering to suggest resources aligned with a learner's profile and progress. In CS education, recommendation systems guide students toward essential skills and resources, ensuring alignment with industry demands. Research by Yu et al. (2022) emphasizes the effectiveness of hybrid models that dynamically update learning roadmaps, enabling students to build comprehensive programming and software development skills.

Educational platforms are increasingly focusing on career-oriented learning paths. For CS students, roadmaps tailored to fields like data science or cybersecurity enhance satisfaction and job preparedness. Johnson et al. (2021) report that students following such roadmaps are better equipped to meet industry expectations. Skill gap analysis further bridges academic learning with industry needs, ensuring students acquire relevant competencies. LinkedIn's 2023 skills report highlights the growing demand for expertise in data science, machine learning, and cybersecurity, reinforcing the necessity of adaptive and career-aligned learning pathways.

User engagement is a critical factor in the success of educational platforms. Gamification, coupled with visual feedback mechanisms, has proven effective in motivating learners. Studies by Almutairi (2024) illustrate how features like badges and progress bars encourage consistent learning and trackable achievements. These elements are vital in maintaining user interest and enhancing overall learning outcomes.

Accessibility and customization are essential for creating inclusive learning platforms. Features such as adjustable timelines, difficulty level options, and screen reader compatibility ensure the platform caters to diverse learner needs. Existing literature also emphasizes that such functionalities enhance usability and effectiveness, making educational tools more inclusive.

Problem Identification

When it comes to students or learning enthusiasts, rapidly evolving technological landscape, many students feel overwhelmed by the abundance of learning resources, online courses, and advice available on the internet. While there is a wealth of content, structured guidance tailored to individual needs and timelines is notably absent. Learners often grapple with questions about where to begin, how much time to allocate to each subject, and how to ensure they are progressing toward their goals. There are uncountable sources of information, online courses, and variety of study platform available for a person who is keen to learn. But there is always a lack of systematic learning. Recently, a systematic literature review (Kusumastuti, et al. 2021) highlighted that existing MOOC system lacks adaptive personalized learning beyond content suggestions. The problem is well highlighted by Pressy (1927) as they emphasized the growing need of structured and intelligent learning path generation. Once this problem is resolved, another big constraint is timeline. A predefined timeline will make the process of learning a lot easier and more doable. Unfortunately, this problem is not new but yet is to be acknowledge while proposing an efficient learning online platform. In this regard, Chen et al. (2023) noted that while adaptive systems exist, very few incorporate time constraints and student specific goals in roadmap generation. Especially, computer science and technology stream is such a stream which evolves every minute time bifurcation. In that case, students from this branch continue to face challenges in navigating their educational journeys due to a lack of personalized and time-sensitive guidance. Hence, there is an undeniable need for a system that acknowledges individual constraints and provides tailored solutions in the form of customized roadmaps.

Present paper develops an innovative and novel learning platform named Connect + as it tackles the absence of guided, customizable learning paths—a challenge echoed in existing literature.

What is Connect+:

Connect+ is an innovative platform designed to help students achieve their academic and professional goals through customized learning roadmaps. Focused primarily on the technology sector, the platform considers each student’s available time, skill level, and career objectives to create a practical, time-bound, and goal-oriented step-by-step guide. By simplifying complex pathways into structured actions, Connect+ empowers learners to progress with confidence and clarity. Current platforms typically offer generic roadmaps or course recommendations that fail to consider the learner's existing skills, availability, and aspirations. As a result, students find themselves sifting through vast amounts of information, leading to confusion, burnout, or even the abandonment of their learning journey. Connect+ bridges this critical gap by providing customized and adaptive learning paths.

The primary objective of Connect+ is to support students by delivering personalized, time-sensitive roadmaps based on their current knowledge, available time, and targeted goals. The platform aims to minimize uncertainty, foster consistent learning, and enhance the likelihood of success in achieving both academic and career-oriented outcomes. It gathers essential user information such as current skill level, available time, and desired outcomes to create a phased roadmap comprising curated tasks and milestones. This adaptable, step-by-step plan strikes a balance between realism and ambition, ensuring learners are not overwhelmed by maintaining an equilibrium between learning intensity and achievable targets. Future enhancements may include progress tracking and dynamic resource recommendations. The essence of

Connect+ lies in its capacity to transform self-paced learning by bridging the gap between information overload and goal-oriented planning. While currently focusing on the technology sector, it has the potential to expand across various disciplines in the future.

Additionally, Connect+ integrates motivational features such as milestones and visual progress tracking to reduce burnout and encourage sustained engagement. By addressing the main limitations of existing systems, mainly their lack of contextual personalization, Connect+ empowers learners with clear, adaptable paths toward achieving their educational and professional goals.

II.MATERIALS AND METHODOLOGY

The development methodology for Connect+ carefully balances user-centric design with efficient, structured software engineering practices. To facilitate early user validation and continual improvement, a hybrid approach combining Prototype and Agile methodologies has been adopted (Singh et al. 2022). This strategy ensures the project remains flexible and responsive to user feedback while maintaining a systematic process for implementation and progress tracking. The process begins with the Prototype Model, creating an initial working version of the product featuring core functionalities such as time-based availability, skill-based matching, and interest-driven user suggestions. This prototype serves as a proof of concept, enabling stakeholders to visualize the application's flow, gather feedback, identify usability issues, and validate features before moving into full-scale development.

Following prototype validation, the project shifts to the Agile Model, which divides development into multiple sprints. Each sprint involves planning, design, development, testing, and review activities, promoting continuous improvement, rapid adaptation to changing requirements, and incremental delivery of functional features. This dual strategy is chosen to accommodate the dynamic nature of user needs while ensuring the timely delivery of a high-quality, functional product.

By integrating these two models, Connect+ benefits from Agile methodology's iterative development, adaptability, and user engagement, complemented by the visual clarity and functional emphasis of prototyping. This hybrid approach reduces project risks, enhances overall quality, and ensures the final system is relevant and user-friendly. Regular stakeholder communication, review cycles, and iterative releases form the foundation of this methodology, resulting in a stable, scalable, and user-focused application.

Process Model Adopted:

The process begins with Prototype Development, where the team identifies essential modules such as user registration, mentioning the domain, and time constraints. A functional prototype focusing on these core components is built to enable user interaction and facilitate meaningful feedback. Once validated, the process transitions into Agile development. This flexibility allows the team to incrementally build the application through sprint cycles, continuously incorporating user feedback and adjusting the development plan as needed.

Agile Sprints follow, with each sprint focusing on specific objectives: Sprint refines core functionalities, and adds advanced features like personalized suggestions, progress bars, and Google Calendar notifications, and Sprint is dedicated to final testing, performance tuning, and deployment preparation. The deployment phase involves rigorous functional and non-functional testing to ensure system stability and that all user requirements are met. After successful testing, the system is deployed and monitored for any issues or potential enhancements. Overall, this hybrid approach effectively manages risks, allows adaptability to change, and delivers a high-quality product that aligns with user expectations and project goals.

Planning And Scheduling

Effective planning and scheduling are essential for the successful completion of Connect+. The development process has been organized into clear, well-defined phases, each with specific goals, deliverables, and timelines. Visual tools like progress bars and milestone plans help illustrate the overall schedule, ensuring the project stays on track and within scope. Phase 1 consists of requirement analysis and prototype design, phase 2 involves feedback collection and prototype refinement, phase 3 was about agile sprint (core system implementation), phase 4 uses agile sprint (advanced features), phase 5 does testing and final integration under agile sprint and final phase 6 is about deployment and review. This structured schedule promotes timely progress, accountability, and flexibility to adapt based on feedback or unforeseen challenges.

Tools and Technologies

The development of Connect+ leverages a robust tech stack that combines modern web technologies with powerful backend support to deliver a smooth, interactive, and secure user experience (Oppenheim, 1975). These tools and technologies were selected based on their performance, scalability, community support, and alignment with the project's goals.

Frontend Technologies: HTML and CSS are used to structure and style the user interface, while JavaScript with React.js adds interactivity and dynamic behaviors. To enhance responsiveness and accelerate design, frameworks such as Bootstrap and Tailwind CSS are employed, utilizing utility classes for efficient UI development.

Backend Technologies: The backend is built with Python (Dean, 2020) using the Django (Paul, 2019) framework, offering a scalable, secure, and efficient foundation for user authentication, routing, and business logic handling. For database management, SQLite and PostgreSQL are utilized, providing lightweight yet powerful options suitable for both development and deployment.

Libraries & Apis: The Django REST Framework facilitates the creation of RESTful APIs, enabling seamless data communication between the frontend and backend. Automating periodic tasks like meeting reminders and availability checks is achieved through tools such as Cron Jobs and APScheduler.

Development & Collaboration Tools: Version control and team collaboration are managed via Git and GitHub. Development is primarily conducted using VS Code. Wireframing, UI design, and architectural diagrams are created with Figma, Draw.io and PlantUML.

Deployment Tools: For deployment, cloud platforms like Render and Heroku are used to host the Django backend and frontend interfaces. Optional containerization with Docker ensures consistent environments across development and deployment stages.

Overall, this carefully selected combination of tools streamlines development, fosters effective collaboration, and ensures that Connect+ remains scalable, responsive, and user-friendly.

Software Design:

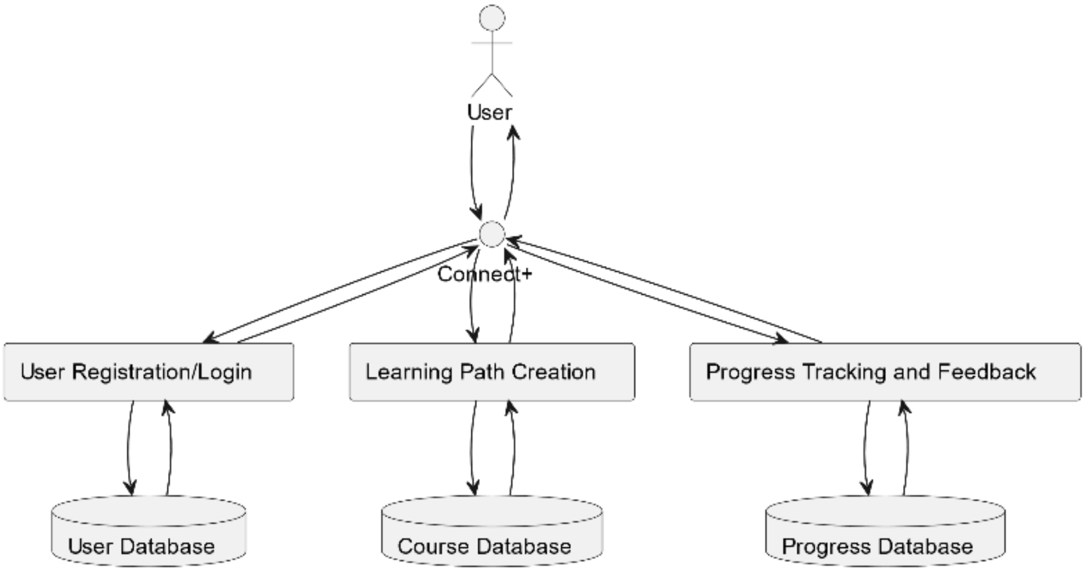

Figure 1: Data Flow Diagram

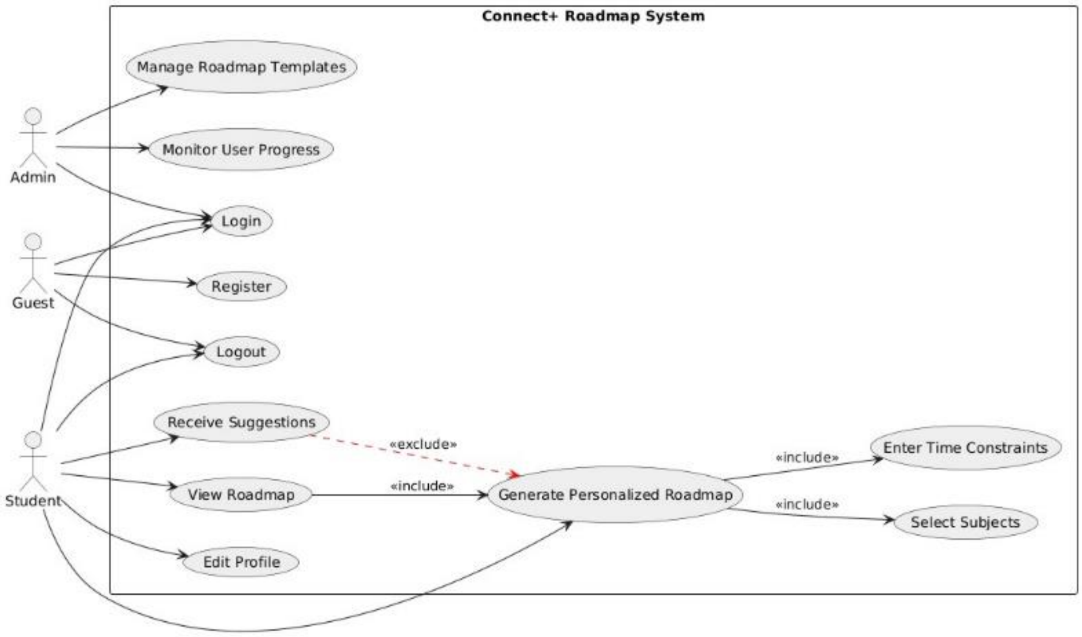

Figure 2: Use Case Diagram

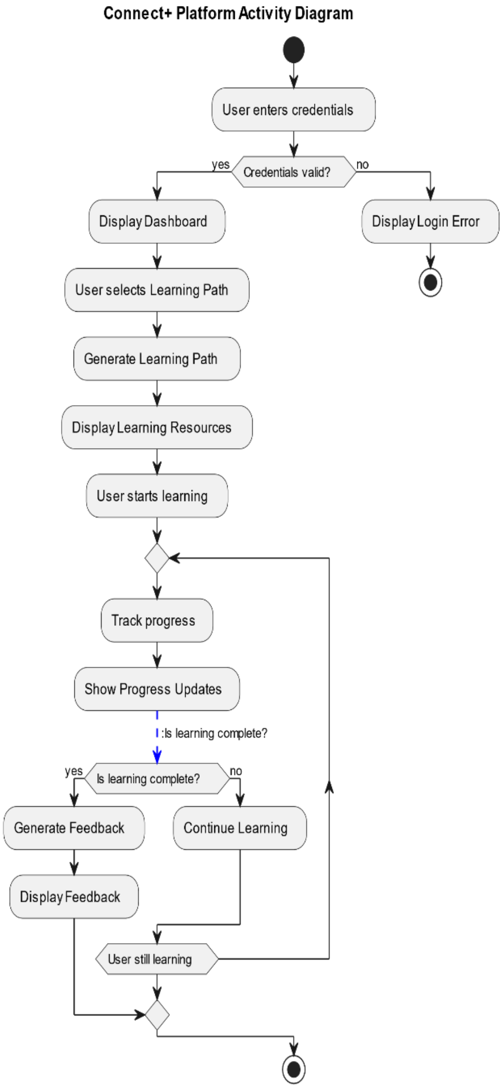

Figure 3: Activity Diagram

III.IMPLEMENTATION & TESTING

The Connect+ GUI design emphasizes user-friendliness, responsiveness, and accessibility. Its minimalistic, intuitive, and consistent layout across screens enhances the overall user experience. The interface features a clean layout with modern UI components, a calming color palette that reduces eye strain (mainly bluish shades), and effective iconography for quick visual cues. Consistent spacing, alignment, and typography improve readability and navigation throughout the application. The primary screens include: - Home screen, matchmaking dashboard, profile page and settings. Each component is designed to be responsive, ensuring smooth functionality across desktops, tablets, and mobile phones. Prototyping tools like Figma and Adobe XD were employed to develop the initial design mockups, which were then implemented using HTML, CSS, and JavaScript. This process ensured consistency between the visual mockups and the final product.

The Connect+ implementation plan follows a phased, modular deployment strategy aligned with the hybrid Agile + Prototype methodology. The process begins with deploying a prototype that showcases core functionalities such as login/registration, time-based matching, and interest tagging. Hosted on a local server for quick updates and testing, the prototype helps stakeholders visualize the application and provides valuable feedback.

After validating the prototype, the project advances to sprint-based deployment. Each sprint delivers specific enhancements or features, including real-time matching algorithms, user profile management, and Google Calendar notifications. At the end of each sprint, code is integrated, tested, and then deployed to a testing server.

Once all key features have been implemented and thoroughly tested, the final deployment occurs in the production environment. This phase involves setting up the environment, covering backend services, frontend deployment, and database configuration, along with performance testing and final quality checks. Throughout the process, regular progress tracking, feedback incorporation, and a rollback mechanism ensure a smooth and resilient implementation. This staged approach enables Connect+ to evolve with precision and stability.

User Navigation Flow

The user navigation flow in Connect+ has been strategically designed to ensure a seamless and intuitive journey from entry to engagement. The flow prioritizes ease of use, minimal user effort, and logical progression of actions.

User Journey Flow:

1. *Login/Registration → Dashboard*
2. *Dashboard → Create a Roadmap*
3. *Roadmap → User Profiles → Progress Bars*
4. *Progress Bars → Active Roadmaps*
5. *Profile Settings → Update Info → Save Changes → Return to Dashboard*

This logical flow ensures that users can perform actions efficiently without unnecessary steps, improving user satisfaction and engagement.

Testing Strategy

To ensure reliability and performance, a multi-level testing strategy was employed for Connect+. The testing process followed a bottom-up approach, starting from unit testing and progressing to user acceptance testing. Each level of testing is crucial for validating different aspects of the application, including functionality, integration, system coherence, and usability.

The key components of the testing strategy are unit testing, integration testing, system testing and user acceptance testing. The testing was conducted using both manual and automated tools such as PyTest (for backend logic) and Selenium (for frontend flow). Bug reports were tracked using a centralized issue tracker, and each sprint included a test-reporting phase to document outcomes and fixes.

IV.RESULTS AND EVALUATIONS

Functional Validation

The Connect+ platform was developed to provide personalized learning roadmaps and effective peer matching for students. The system's core modules—including login/registration, roadmap creation, user profile management, and matchmaking—were validated through comprehensive testing and real-world usage.

Login and Registration: Secure authentication and robust input validation were confirmed through both manual and automated tests. No critical vulnerabilities or usability issues were found.

Roadmap and Dashboard: The dynamic card layout and personalized dashboard were tested for responsiveness and correctness. All filters (skills, time availability) functioned as intended, and the roadmap generation logic produced accurate, user-specific results.

Profile Management: Users were able to update their information, skills, and availability seamlessly. Changes were reflected instantly in matchmaking and dashboard modules.

User Experience and Usability

A user-centric design approach was adopted, and the navigation flow was evaluated by a group of beta testers (students and faculty). Key findings include ease of Use as over 90% of users reported that the interface was intuitive and easy to navigate, onboarding because new users were able to register and set up their profiles without external assistance and finally satisfaction because majority of the users expressed satisfaction with the personalized recommendations and the clarity of progress indicators.

Testing Outcomes

A multi-level testing strategy ensured the reliability and robustness of Connect+. Unit testing achieved more than 95% pass rate across modules such as authentication, data validation, and scheduling logic using PyTest. Integrating testing (All major integrations e.g., matchmaking, Google Calendar, frontend-backend data flow) passed automated Selenium and Postman tests, confirming seamless module interaction. System testing showed that end-to-end scenarios, including registration, roadmap creation, and chat, were validated under simulated multi-user load. The system maintained stable performance and data integrity. Finally, user acceptance testing (UAT) revealed that Beta users confirmed the expectations of the platform with respect to usability, accuracy, and performance.

Performance Evaluation

As far as response time is concerned, average API response times were consistently below 250ms under moderate load (up to 50 concurrent users). Scalability is also consistent as the backend architecture (Django + PostgreSQL) handled concurrent operations without data loss or significant slowdowns. So far reliability is also under acceptability as no major crashes or data inconsistencies were observed during extended testing periods.

Conclusion:

Development of Connect+ has successfully created a platform that provides personalized learning roadmaps, helping users navigate their academic and career journeys more effectively. By offering tailored learning paths, particularly for Computer Science students, the platform allows individuals to chart a course based on their unique goals and preferences. It provides a structured, yet flexible way to develop skills, empowering users to take charge of their learning. While the platform has made significant strides in offering personalized recommendations, there's still much more to explore and improve in terms of content and user experience.

REFERENCES

[1] Almutairi, A. M. (2024). A systematic review of catching and comparing the most valuable gamification elements used in online learning. *Edelweiss Applied Science and Technology*, 8(4), 2181-2201.

[2] Chen, Z., Zhang, H., Liu, J., Wang, W., & Wang, J. (2023). Adaptive prescribed settling time periodic event-triggered control for uncertain robotic manipulators with state constraints. *Neural Networks*, 166, 1-10.

[3] Dean, J. W. (2020). Python for Data Science: Data Analysis and Machine Learning with Python, 1st Edition, New York, NY, Springer.

[4] Gikandi, J. W., & Morrow, D. (2016). Designing and implementing peer formative feedback within online learning environments. Technology, Pedagogy and Education, 25(2), 153-170.

[5] Jiang, Y., Pardo, A., & Siemens, G. (2018). Curriculum Graphs: Modeling Prerequisite Structures for Adaptive Learning Systems. *Proc. of the 8th Int'l Conf. on Learning Analytics and Knowledge (LAK)*, ACM.

[6] Johnson, O. W., Mete, G., Sanchez, F., Shawoo, Z., & Talebian, S. (2021). Toward Climate-Neutral Heavy Industry: An Analysis of Industry Transition Roadmaps. *Applied Sciences*, 11(5375), 16.

[7] Kusumastuti, D. L., Meyliana, Hidayanto, A. N., & Prabowo, H. (2021), Models of Adaptive Learning System in MOOC: A Systematic Literature Review. *9th International Conference on Information and Education Technology (ICIET).*

[8] Oppenheim, A.V., & Schafer, R.W. (1975). Digital Signal Processing, Englewood, N.J., Prentice Hall, 3rd Edition.

[9] Paul, D. J. (2019). Introduction to Web Development with Django, 2nd Edition, Boston, MA, Addison-Wesley.

[10] Pressey, S.L. (1927). A machine for automatic teaching of drill material. Sch. Soc., 25, 549–552.

[11] Singh, P., Gupta, S., & Sharma, P. (2022). Hybrid Recommender Systems for Personalized Learning. *International Journal of Computer Science & Engineering*, 15, 125-137.

[12] Wu, X., Zhang, Y., Shi, M., Li, P., Li, R., & Xiong, N. N. (2022). An adaptive federated learning scheme with differential privacy preserving. *Future Generation Computer Systems*, 127, 362-372.

[13] Yu, Z., Si, Z., Li, X., Wang, D., & Song, H. (2022). A novel hybrid particle swarm optimization algorithm for path planning of UAVs. *IEEE Internet of Things Journal*, 9(22), 22547-22558.

[14] Zilberman, N., Koedinger, K. S., & Mozer, R.J. (2021). Adaptive Learning Systems in Educational Technology. *Journal of Educational Technology & Society*, 24, 200-215.