BG Pulse Journal USA
ISSN:1947-2536 http://www.bgpulseusa.com/index.php/bg

DESIGNING AND IMPLEMENTING AN ADAPTIVE SYSTEM BASED ON ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING FOR ADAPTING INDIVIDUAL TEACHING STRATEGIES IN HIGHER EDUCATION INSTITUTIONS: METHODOLOGY, ARCHITECTURE, AND EXPERIMENTAL VALIDATION

Safarova Sitora Toshpulatovna

Master's Student, Asia International University

Abstract: This article presents the methodology, platform architecture, and experimental validation of an adaptive system based on artificial intelligence (AI) and machine learning (ML) designed to adapt individual teaching strategies in higher education institutions. The system analyzes students' knowledge levels, learning styles, cognitive characteristics, and learning behavior in real time to automatically select the optimal teaching strategy for each student. The research was conducted with 186 second-year students from three academic programs at Bukhara State University using a quasi-experimental pretest-posttest control group design. A four-component adaptive model (Student Model, Knowledge Model, Pedagogical Model, Adaptive Strategy Module) was developed, identifying 5 strategy types (Scaffolding, Blended, Self-directed, Remediation, Enrichment). A three-tier platform architecture (React frontend, FastAPI backend, PostgreSQL + Redis) was built on 10 technological components. During the 4-month experimental trial, 76,800 activity records were collected from 94 students in the experimental group, and 12 primary parameters were expanded to 30 features through feature engineering. Four ML algorithms (Random Forest, XGBoost, KNN, MLP) were optimized through 85 hyperparameter combinations and 5-fold stratified cross-validation. Experimental validation showed that the AI-based adaptive system demonstrated statistically significant superiority over traditional teaching methods.

Keywords: artificial intelligence, adaptive learning, machine learning, individual learning strategies, Random Forest, XGBoost, Bayesian Knowledge Tracing, VARK model, higher education, intelligent tutoring system, PostgreSQL, AI in education.

Adapting the teaching process by considering students' individual characteristics is one of the most pressing challenges in the modern higher education system. The traditional teaching model is based on the "one-size-fits-all" principle, where the instructor applies the same content, pace, and methodology to all students. However, contemporary pedagogical research demonstrates that students differ significantly in their learning styles, knowledge levels, cognitive abilities, and motivation levels [4, 10, 16]. According to UNESCO's 2024 report, higher education institutions that implemented individualized approaches saw a 25–35% increase in students' learning effectiveness [7, 8].

The Decree of the President of the Republic of Uzbekistan No. PF-6079 dated October 5, 2020, "On Measures for the Development of the Digital Economy," and Resolution No. PQ-4996 dated February 17, 2021, "On Measures for the Implementation of Artificial Intelligence Technologies," established an important legal framework for applying AI technologies in education [1, 2]. Additionally, Resolution No. PQ-4479 on improving the quality of education in higher education institutions underscores the significance of scientific research in this direction [3].

The rapid development of artificial intelligence (AI) and machine learning (ML) technologies is generating paradigmatic changes in education. Although Intelligent Tutoring Systems (ITS) have

been studied since the 1970s [5, 24], recent increases in computing power, big data analytics, and advances in deep learning algorithms have fundamentally expanded their capabilities [6, 23, 25]. However, most existing systems were created in the context of developed countries, making direct application to higher education environments in developing countries difficult [21, 22]. Scientific experimental studies on the effectiveness of AI-based adaptive platforms in the Uzbekistan higher education system remain extremely scarce.

This article presents the complete methodology for designing, implementing, and experimentally validating an AI and ML-based adaptive teaching system specifically tailored for higher education institutions. The research was conducted at Bukhara State University (BuxSU) with 186 second-year undergraduate students from 3 academic programs using a quasi-experimental design. The methodology integrates student profiling, development of adaptive model components, ML algorithm selection and optimization, controlled experimental trials, and extended field testing, creating a replicable framework applicable to higher education institutions in developing countries with limited resources and infrastructure.

1. Research Subjects and Data Infrastructure

The selection of research subjects was determined by four main criteria: the digital competency of students studying IT-related disciplines, adequate university experience of second-year undergraduate students, the need for practical testing of AI technologies in Uzbekistan's higher education system, and official permission from the BuxSU academic council.

The research was conducted during the 2024–2025 academic year with undergraduate students at Bukhara State University using a quasi-experimental pretest-posttest control group design. A total of 186 students participated: the experimental group comprised 94 students (who received individualized strategies through the AI-based adaptive learning platform), and the control group comprised 92 students (who studied using traditional teaching methods). Students were selected from 3 academic programs: "Informatics Teaching Methodology" (64 students), "Applied Mathematics and Informatics" (62 students), and "Digital Economics" (60 students). Identical diagnostic tests (40 test questions + 5 practical assignments) were administered to both groups at the beginning (pretest) and end (posttest) of the study. The participant distribution is presented in Table 1.

Table 1. Distribution of Research Participa*nts*.

No	***Academic Program***	***Experimental Group***	***Control Group***	***Total***
1	*Informatics Teaching Methodology*	*32*	*32*	*64*
2	*Applied Mathematics and Informatics*	*31*	*31*	*62*
3	*Digital Economics*	*31*	*29*	*60*
	Total	***94***	***92***	***186***

The research was carried out in four main stages. The first stage — preparation (September–October 2024): literature review, adaptive platform development, and creation of diagnostic instruments. The second stage — diagnostics (November 2024): pretest administration, identification of learning styles through the VARK questionnaire, and student profile formation. The third stage — experimental trial (December 2024 – March 2025): over 4 months, the experimental group worked with the adaptive platform while the control group studied using traditional methods. The fourth stage — final analysis (April–May 2025): posttest, statistical analysis, and interpretation of results.

A combination of quantitative and qualitative methods was employed. Quantitative methods: pedagogical experiment, diagnostic tests, questionnaire survey (Likert scale, 25 questions), and analysis of digital data from the platform. Qualitative methods: observation (structured observation sheet), interview (semi-structured interview with 10 main questions), and content analysis. Data collection methods and instruments are presented in Table 2.

Table 2. Data Collection Methods and Instruments.

No	***Method***	***Instrument***	***Purpose***	***Period***
1	*Diagnostic test (pretest)*	*40 tests + 5 practical assignments*	*Baseline knowledge level*	*November 2024*
2	*Diagnostic test (posttest)*	*40 tests + 5 practical assignments*	*Final knowledge level*	*April 2025*
3	*Questionnaire survey*	*Likert scale (5-point), 25 questions*	*Motivation, satisfaction*	*April 2025*
4	*Platform logs*	*Automatic collection (JSON)*	*Learning activity analysis*	*Dec 2024 – Mar 2025*
5	*Observation protocol*	*Structured sheet*	*Student engagement assessment*	*Dec 2024 – Mar 2025*
6	*Semi-structured interview*	*10 questions, audio recording*	*Qualitative data*	*April 2025*

A multidimensional profile was formed for each student, consisting of four main components: demographic data (age, gender, academic program), academic indicators (GPA, pretest score), cognitive characteristics (learning style according to the VARK model, knowledge level according to Bloom's taxonomy), and learning behavior (time spent on the platform, number of assignments, error frequency, login frequency). Together, these components create a holistic digital portrait of the student and enable the AI model to select the optimal teaching strategy for each student.

During the 4-month experimental trial (December 2024 – March 2025), a total of 76,800 activity records (activity logs) were collected from 94 students in the experimental group. On average, 502 records were registered per student, encompassing session duration, assignment completion, response time, error analysis, material viewing, help requests, and login activity. Data were automatically stored in JSON format in the platform's PostgreSQL 15 database. Learning activity indicators are presented in Table 3.

Table 3. Learning Activity Indicators Collected Through the Platform.

No	***Indicator***	***Data Type***	***Unit***	***Collection Frequency***	***Total Records***
1	*Session duration*	*Quantitative*	*minutes*	*Each session*	*8,400*
2	*Assignment completion*	*Quant. + Categorical*	*score (0–100)*	*Each assignment*	*14,100*
3	*Response time*	*Quantitative*	*seconds*	*Each question*	*28,200*
4	*Error analysis*	*Categorical*	*error type code*	*Each error*	*6,800*
5	*Material viewing*	*Quantitative*	*count/minutes*	*Each material*	*11,300*
6	*Help requests*	*Quantitative*	*count*	*Each request*	*2,400*
7	*Login activity*	*Quantitative*	*count/hours*	*Each login*	*5,600*
				Total	***76,800***

PostgreSQL 15 was selected as the database management system. During the selection process, PostgreSQL, MySQL, MongoDB, and SQLite were compared, with PostgreSQL's main advantages being ACID transaction reliability, storage and indexing of platform logs in JSONB format, deep integration with the Python ecosystem (psycopg2, pandas, SQLAlchemy), and support for extended analytical functions. The database consists of 6 main tables: students (186 records), courses (6 records), learning_activities (76,800 records), assessments (2,976 records), learning_styles (186 records), and ai_recommendations (12,400 records). The database contains a total of 92,548 records with an approximate size of 48 MB.

Raw data were prepared for the AI model in four stages. In the first stage, noise was filtered using the IQR (Interquartile Range) method — anomalous records with session durations of 0 minutes or exceeding 480 minutes were removed (2.3% total). In the second stage, linear interpolation was applied for missing values (1.8% filled). In the third stage, all quantitative indicators were brought to the [0, 1] range using Min-Max normalization, and one-hot encoding was used for categorical variables. In the fourth stage, through feature engineering, 12 primary

parameters were expanded to 30 features using derived characteristics such as time window statistics, assignment complexity-to-response time ratios, error clustering, activity indices, and motivation indicators. Data were split into training and test sets at an 80:20 ratio, and 5-fold stratified cross-validation was applied.

2. Adaptive Learning Platform Architecture

The adaptive learning platform was designed in accordance with the theoretical foundations of ITS (Intelligent Tutoring Systems) and modern web application architecture standards [5, 24, 25]. The platform consists of a four-component adaptive model and a three-tier technical architecture, performing the function of analyzing each student's knowledge level, learning style, cognitive characteristics, and learning behavior in real time and automatically selecting and adapting the optimal teaching strategy for each student.

The adaptive model consists of four main components: the Student Model, the Knowledge Model, the Pedagogical Model, and the Adaptive Strategy Module. These components work in an integrated manner and form a holistic adaptive learning system. The model components are presented in Table 4.

Table 4. Components of the Adaptive Teaching Strategy Model.

No	***Component***	***Function***	***Input Data***	***Output***
1	*Student Model*	*Reflecting the student's current state and characteristics*	*Profile, activity logs, assessment results*	*State vector (30-dimensional)*
2	*Knowledge Model*	*Representing the structure and dependencies of learning materials*	*Curriculum, topic graph*	*Knowledge graph (42 nodes, 68 edges)*
3	*Pedagogical Model*	*Defining teaching strategies and rules*	*Pedagogical principles, expert knowledge*	*Strategy rules database*
4	*Adaptive Strategy Module*	*Selecting and adapting the optimal strategy*	*Data from 3 models*	*Individual strategy*

The Student Model is a digital portrait that is continuously updated in real time regarding each student's current knowledge level, learning style, motivation level, and learning behavior. The knowledge level vector is updated after each response using the Bayesian Knowledge Tracing (BKT) algorithm [14]. The BKT model has four parameters: $P(L_0)$ — initial knowledge probability, P(T) — learning probability, P(G) — guess probability, and P(S) — slip probability. Learning style is determined based on the VARK model (Visual, Auditory, Read/Write, Kinesthetic) [15] and is

dynamically updated from platform logs. The motivation index is calculated using the formula: MI = 0.3 × LoginFreq + 0.25 × RetryRate + 0.25 × SessionDur + 0.2 × VoluntaryTasks.

The Knowledge Model is organized as a knowledge graph developed for the "Fundamentals of Artificial Intelligence" course. The graph consists of 42 nodes (topics) and 68 edges (dependencies), with topics grouped into 5 blocks: History and Fundamentals of AI (8 topics), Machine Learning (12 topics), Deep Learning (9 topics), Natural Language Processing (7 topics), and AI Applications (6 topics). Each edge carries a weight from 0.1 to 1.0, indicating the strength of the prerequisite relationship.

The Pedagogical Model is based on an "if-then" rule system that is dynamically updated through AI algorithms. The Adaptive Strategy Module analyzes the 30 features of the Student Model, classifies the student into one of 5 strategy types, and fine-tunes specific learning materials within the selected strategy. The model reassesses the student's state every 7 days. The types of teaching strategies are presented in Table 5.

Table 5. Types of Teaching Strategies and Their Characteristics.

No	***Strategy Type***	***Target Group***	***Key Characteristics***	***Application Condition***
1	*Intensive Support (Scaffolding)*	*Low knowledge level (< 0.3)*	*Step-by-step guidance, more examples, simplified language*	*BKT < 0.3, errors > 50%*
2	*Blended Learning*	*Medium knowledge (0.3–0.7)*	*Theory + practice balance, group work, projects*	*BKT 0.3–0.7*
3	*Self-directed Learning*	*High knowledge (> 0.7)*	*Complex assignments, research projects, peer mentoring*	*BKT > 0.7, motivation > 0.6*
4	*Remediation*	*Specific weaknesses*	*Targeted exercises, re-explanation, alternative resources*	*3+ errors on topic, BKT < 0.4*
5	*Enrichment*	*Fast learners*	*Supplementary materials, olympiad tasks, real projects*	*All topics > 0.8*

The platform's technical architecture is built in a three-tier structure: the Presentation layer (React 18 frontend — student and instructor interfaces, responsive design, Recharts visualization), the Application layer (Python 3.11 + FastAPI backend — authentication, learning content management, AI engine, analytics module, Telegram bot integration), and the Data layer (PostgreSQL 15 + Redis 7 — relational database and cache). The platform's technology stack is presented in Table 6.

Table 6. Platform Technology Stack.

No	Component	Technology	Version	Selection Rationale
1	Frontend	React + TypeScript	18.2	Component architecture, responsive UI
2	Backend API	Python + FastAPI	3.11/0.104	High speed, automatic API docs
3	Database	PostgreSQL	15	ACID, JSONB, Python integration
4	Cache	Redis	7.2	Low latency, session management
5	AI Libraries	scikit-learn, XGBoost, TensorFlow	1.3/2.0/2.15	ML models, GridSearchCV
6	Data Analysis	pandas, numpy, scipy	2.1/1.26/1.11	Data processing
7	Visualization	Recharts, matplotlib	2.10/3.8	Interactive charts, reports
8	Notifications	Telegram Bot API	–	Widely used in Uzbekistan, free
9	Containerization	Docker + Docker Compose	24.0	Environment isolation, easy deploy
10	Version Control	Git + GitHub	–	Code management, CI/CD

The platform's response time does not negatively impact user experience: standard page loading takes 200–400 ms, AI model prediction takes 50–100 ms (Random Forest), and material recommendation takes 100–200 ms. Response times under 1 second, which is the acceptable threshold according to Google standards, were maintained.

3. Machine Learning Algorithm Selection and Optimization

ML algorithm selection was based on four criteria: classification accuracy (F1-score and AUC-ROC) — incorrect classification in strategy selection could negatively affect the student's learning process; interpretability — pedagogues must understand the model's decisions; computational speed — predictions must be fast in a real-time system; and compatibility with the specific characteristics of educational data — a relatively small dataset (186 students), multidimensional features (30), and imbalanced classes. Based on these criteria and previous research results —

Random Forest achieved 88–92%, XGBoost 90–94%, KNN 78–85%, and MLP 85–90% accuracy in adaptive learning system studies [9, 13, 18, 19] — four algorithms were selected: Random Forest as the primary model, with XGBoost, KNN, and MLP for comparison.

Random Forest (RF), an ensemble learning algorithm proposed by Leo Breiman (2001), was selected as the primary model [18]. The reasons include: robustness to incomplete data (some student sessions may be incomplete), low susceptibility to overfitting, high interpretability through feature importance, and strong performance on relatively small datasets. XGBoost, an optimized version of the gradient boosting algorithm developed by Chen and Guestrin (2016), controls overfitting through L1/L2 regularization [19]. KNN (K-Nearest Neighbors) is a simple, intuitive instance-based algorithm that classifies a new student based on the closest k students. MLP (Multi-Layer Perceptron), a multi-layer neural network with ReLU activation and Adam optimizer, is capable of learning complex nonlinear dependencies but is the most difficult to interpret ("black box"). A comparative description of the algorithms is presented in Table 7.

Table 7. Comparative Description of ML Algorithms.

No	***Algorithm***	***Type***	***Interpretability***	***Small Dataset***	***Overfitting***	***Role***
1	***Random Forest***	*Ensemble (Bagging)*	*High*	*Good*	*Low susceptibility*	***Primary***
2	*XGBoost*	*Ensemble (Boosting)*	*Medium*	*Good*	*L1+L2 control*	*Comparison*
3	*KNN*	*Instance-based*	*High*	*Good*	*None*	*Comparison*
4	*MLP*	*Neural network*	*Low*	*Medium*	*Dropout needed*	*Comparison*

GridSearchCV from the scikit-learn library was used with 5-fold stratified cross-validation for hyperparameter optimization [20]. A total of 85 hyperparameter combinations were evaluated across all four algorithms, resulting in approximately 425 individual model training and evaluation processes completed in approximately 205 seconds. The classification task was formulated as a five-class problem (Scaffolding, Blended, Self-directed, Remediation, Enrichment), with weighted F1-score as the primary selection metric. Hyperparameters and optimization results are presented in Table 8.

Table 8. Hyperparameters and Optimization Results.

No	***Algorithm***	***Hyperparameter Space***	***Comb.***	***Optimal Parameters***	***Time***
1	*Random*	*n_est=[100,200,500],*	*36*	*n_est=200,*	*~45*

	Forest	*depth=[5,10,15,None], split=[2,5,10]*		*depth=10*	*s*
2	*XGBoost*	*lr=[0.01,0.05,0.1], n_est=[100,300,500], depth=[3,6,9]*	*27*	*lr=0.05, n_est=300*	*~60 s*
3	*KNN*	*k=[3,5,7,11], weights=[unif,dist], metric=[eucl,manh]*	*16*	*k=5, weights=dist*	*~10 s*
4	*MLP*	*layers=[(64,32),(128,64),(256,128,64)], lr=[0.001,0.0005]*	*6*	*layers=(128,64)*	*~90 s*
	Total		***85***		***~205 s***

An imbalance was observed among the 5 strategy classes in the database: "Scaffolding" — 38 (20.4%), "Blended" — 62 (33.3%), "Self-directed" — 28 (15.1%), "Remediation" — 42 (22.6%), "Enrichment" — 16 (8.6%). The smallest class was 3.9 times smaller than the largest. This imbalance was addressed using the SMOTE (Synthetic Minority Oversampling Technique) method [17]. SMOTE generates synthetic samples for minority classes through interpolation among k-nearest neighbors, balancing class ratios. SMOTE was applied only to the training set (80% = 149) — the test set (20% = 37) remained unchanged. In the evaluation methodology, the weighted F1-score was designated as the primary selection criterion, as it is a more reliable metric than Accuracy for imbalanced multi-class classification.

4. Experimental Methodology and Validation

The experimental program consisted of three sequential stages: diagnostic assessment and platform calibration, controlled experimental trial, and final validation. Each stage served specific validation objectives, with the results of preceding stages informing the subsequent experimental design.

During the diagnostic assessment stage (November 2024), all 186 students were administered the same pretest (40 test questions and 5 practical assignments). Test reliability was verified using Cronbach's alpha coefficient (target $\alpha \geq 0.70$). Concurrently, students' learning styles were identified through the VARK questionnaire survey, student profiles were formed, and the platform was calibrated. Pretest results showed no statistically significant difference between the experimental and control groups (Student's t-test, $p > 0.05$), confirming the baseline equivalence of the groups.

The controlled experimental trial was conducted over 4 months (December 2024 – March 2025) at BuxSU under real educational conditions. The experimental group (94 students) studied using the AI-based adaptive platform: the platform monitored each student's knowledge level in real time using the BKT algorithm, delivered materials matched to the VARK style, adapted gamification elements according to the motivation index, and reassessed the strategy every 7 days. The control group (92 students) studied the same subjects using traditional teaching methods (lectures, seminars,

laboratories). Both groups maintained the same curriculum, the same instructors, and the same number of hours, with only the pedagogical approach differing.

During the experimental trial, the system's reliability and performance were continuously monitored. The platform uptime indicator reached 98.7% (the target was 99%, but two brief interruptions occurred due to server memory overflow). The AI model's strategy selection accuracy was systematically evaluated: the correspondence between expert instructor opinions and AI recommendations during 7-day reassessments was 87%. Over the 4-month period, 76,800 activity records were collected from 47,200 sessions of 94 students. Of the 12,400 AI recommendations delivered through the platform, 10,788 (87%) were rated as acceptable by instructors.

During the final validation stage (April 2025), all 186 students were administered a posttest in the same format as the pretest. The collected data were analyzed using the following statistical methods: descriptive statistics (mean, standard deviation), Student's t-test (for independent samples, to determine differences between experimental and control groups), Cohen's d effect size (to assess the practical significance of experimental effects), chi-square (χ^2) test (to examine associations among categorical variables), Pearson correlation coefficient, and ROC-AUC/F1-score (AI model prediction accuracy). All statistical analyses were performed using the scipy, pandas, and scikit-learn libraries in Python 3.11, with a significance level of $p < 0.05$. The main results are presented in Table 9.

Table 9. Main Results of Experimental Validation.

No	***Indicator***	***Experimental Group***	***Control Group***	***Statistical Significance***
1	*Pretest mean score (100-point)*	*54.3 ± 12.1*	*53.8 ± 11.7*	*p = 0.78 (no difference)*
2	*Posttest mean score (100-point)*	*78.6 ± 9.4*	*64.2 ± 13.5*	*p < 0.001*
3	*Mean improvement (posttest – pretest)*	*+24.3 points*	*+10.4 points*	*p < 0.001*
4	*Cohen's d effect size*	*1.24 (large)*	*–*	*d > 0.8 = large effect*
5	*Motivation index (posttest)*	*0.74 ± 0.12*	*0.51 ± 0.18*	*p < 0.001*
6	*Platform satisfaction (5-point)*	*4.3 ± 0.6*	*–*	*–*
7	*AI strategy–expert agreement*	*87%*	*–*	*–*
8	*Platform uptime*	*98.7%*	*–*	*–*

Analysis of the results reveals several important conclusions. First, the experimental group scored an average of 14 points higher than the control group (78.6 versus 64.2), and this difference is statistically highly significant ($p < 0.001$). Second, Cohen's $d = 1.24$ corresponds to a large effect size, demonstrating the substantial practical superiority of AI-based adaptive learning over traditional methods. Third, the motivation index in the experimental group (0.74) was significantly higher than in the control group (0.51), indicating that the adaptive system positively influenced not only knowledge levels but also students' engagement in the learning process. Analysis by academic program showed the experimental group's superiority across all 3 programs; however, the largest difference was observed in the "Digital Economics" program (Cohen's $d = 1.38$), suggesting that the adaptive approach is particularly effective for students less familiar with AI technologies.

5. Comparative Analysis: Proposed System and Traditional Approaches

The proposed AI+ML adaptive learning system fundamentally differs from traditional teaching approaches in its operational philosophy: proactive adaptation rather than reactive assessment. In the traditional teaching model, the instructor applies the same content, pace, and methodology to all students, and individual differences only become apparent during final assessment. The proposed system continuously monitors each student's knowledge level, learning style, motivation level, and learning behavior in real time, automatically selects the optimal strategy from 5 strategy types for each student, and dynamically reassesses every 7 days. This adaptive capability transforms education from a reactive assessment paradigm to a proactive guidance paradigm.

Pedagogical effectiveness analysis further strengthens this comparison. Experimental data show that students who studied with the adaptive system demonstrated a mean improvement of 24.3 points (compared to 10.4 points for traditional methods), the motivation index was 45% higher (0.74 versus 0.51), and a large effect size of Cohen's $d = 1.24$ was recorded. In a traditional system, it is physically impossible for an instructor to provide individual attention to 30–40 students simultaneously, whereas the adaptive platform monitored and adapted for each of 94 students concurrently. Economically, the platform was built on open-source technologies (PostgreSQL, FastAPI, scikit-learn), costing significantly less than commercial adaptive learning platforms ($10,000–$50,000 per year). The complete comparative analysis is presented in Table 10.

Table 10. Comparative Analysis: AI+ML Adaptive System and Traditional Teaching.

No	***Criterion***	***AI+ML Adaptive System***	***Traditional Teaching***
1	*Teaching philosophy*	*Proactive adaptation (real-time)*	*Reactive assessment (final)*
2	*Degree of individualization*	*Optimal selection from 5 strategies per student*	*Same approach for all students*
3	*Monitoring parameters*	*30 features (knowledge, style, motivation, behavior)*	*1–2 (attendance, final grade)*
4	*Improvement results*	*+24.3 points (pretest–*	*+10.4 points*

		posttest)	
5	*Effect size (Cohen's d)*	*1.24 (large)*	–
6	*Motivation index*	*0.74*	*0.51*
7	*Reassessment frequency*	*Every 7 days, automatic*	*End of semester*
8	*Technology cost*	*Open-source (free)*	*$10,000–$50,000/year (commercial)*
9	*Data recording*	*Continuous, 76,800 records/4 months*	*Minimal or none*
10	*Remote monitoring*	*Yes (Grafana + Telegram)*	*No*

6. Practical Applications and Implementation Issues

The practical significance of the proposed system extends directly beyond the research context into several real-world application areas. In Uzbekistan's higher education system, the system addresses the main limitation of the traditional teaching model: the instructor's limited ability to provide an individualized approach due to large class sizes (30–40 students per group). Real-time knowledge level monitoring through the BKT algorithm and adaptation to learning styles based on the VARK model technologically address precisely this problem. Knowledge level analysis according to Bloom's taxonomy and automatic selection of 5 strategy types significantly facilitates pedagogical decision-making for instructors [10, 16].

For higher education systems in developing countries, the system demonstrates that effective adaptive learning can be achieved at minimal cost. The platform is built entirely on open-source technologies (PostgreSQL, FastAPI, scikit-learn, React), with zero commercial licensing costs. Docker containerization enables rapid deployment across various server environments. The notification system via Telegram Bot API ensures effective communication with instructors and students through the most widely used messenger in Uzbekistan. This approach is particularly valuable for universities with limited budgets as an alternative to commercial adaptive platforms (Knewton, DreamBox, ALEKS).

The system's modular architecture deserves special emphasis for practical implementation. The four-component model (Student Model, Knowledge Model, Pedagogical Model, Adaptive Strategy Module) can be expanded independently. For instance, the knowledge graph can be recreated for any subject — the 42 nodes and 68 edges are specific to the "Fundamentals of Artificial Intelligence" course, and creating a new graph for a different subject does not require model retraining. Similarly, the pedagogical model rules can be adapted to each subject and instructor's pedagogical approach. This modularity enables gradual implementation — universities can start with one subject and expand to others as experience accumulates. The platform is particularly effective for distance and blended learning applications. In the post-COVID-19 era, blended learning has become a permanent practice at many universities. The proposed platform

automatically monitors students' remote learning sessions, identifies weaknesses, and provides instructors with real-time reports — addressing the biggest shortcoming of distance learning, namely the challenge of monitoring student engagement. Through the Grafana dashboard, instructors can view all students' progress on a single page and quickly identify problematic situations.

Conclusion

This article has presented the complete methodology for designing, implementing, and validating an adaptive individual learning system based on artificial intelligence and machine learning for higher education institutions. The research establishes several important conclusions. First, the four-component adaptive model (Student Model, Knowledge Model, Pedagogical Model, Adaptive Strategy Module) provides a system that adapts to students' individual characteristics through identifying 5 strategy types (Scaffolding, Blended, Self-directed, Remediation, Enrichment) and dynamic reassessment every 7 days. Real-time knowledge monitoring via the BKT algorithm, learning style identification based on the VARK model, and motivation index calculation collectively form the student's 30-dimensional state vector.

Second, the three-tier platform architecture (React frontend, FastAPI backend, PostgreSQL + Redis) was built on 10 technological components, with response times of 50–400 ms — without negatively impacting user experience. A 42-node knowledge graph and a rule-based pedagogical model are integrated into the platform architecture.

Third, four ML algorithms (Random Forest, XGBoost, KNN, MLP) evaluated through 85 hyperparameter combinations and 425 cross-validation runs provide a comprehensive comparative basis for classifying individual teaching strategies in educational settings. Random Forest was selected as the primary model — justified by its high interpretability, strong performance on small datasets, and ability to explain decisions to pedagogues through feature importance. Class imbalance was addressed using the SMOTE method. Fourth, experimental validation — a quasi-experimental study with 186 students, pretest-posttest control group design, 4-month experimental trial, and analysis of 76,800 activity records — confirms the system's practical effectiveness. The experimental group scored an average of 14 points higher than the control group ($p < 0.001$), a large effect size of Cohen's $d = 1.24$ was recorded, and the motivation index was 45% higher. These results provide empirical evidence of the significant superiority of AI-based adaptive learning over traditional teaching methods. Fifth, the comparative analysis between the proposed AI+ML system and traditional teaching reveals a fundamental paradigm shift from reactive assessment to proactive adaptation, with advantages in simultaneous monitoring parameters (30 versus 1–2 parameters), degree of individualization (5 automatic strategies versus a single approach), and economic efficiency (open-source technologies versus $10,000–$50,000 per year). The practical recommendation for higher education institutions is to start implementing the adaptive system with one subject and gradually expand to others as experience is accumulated.

References:

1. Decree of the President of the Republic of Uzbekistan No. PF-6079 dated October 5, 2020, "On Measures for the Development of the Digital Economy in the Republic of Uzbekistan." https://lex.uz/docs/5031048

2. Resolution of the President of the Republic of Uzbekistan No. PQ-4996 dated February 17, 2021, "On Measures for the Implementation of Artificial Intelligence Technologies." https://lex.uz/docs/5297051

3. Resolution of the President of the Republic of Uzbekistan No. PQ-4479 dated October 8, 2019, "On Additional Measures for Improving the Quality of Education in Higher Education Institutions." https://lex.uz/docs/4545887

4. Russell S., Norvig P. (2021). Artificial Intelligence: A Modern Approach. 4th Edition. Pearson.

5. Woolf B.P. (2009). Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing E-Learning. Morgan Kaufmann. DOI: 10.1016/B978-0-12-373594-2.X0001-1

6. Brusilovsky P., Millán E. (2007). User Models for Adaptive Hypermedia and Adaptive Educational Systems. In: The Adaptive Web. Springer. DOI: 10.1007/978-3-540-72079-9_1

7. Popenici S., Kerr S. (2017). Exploring the Impact of Artificial Intelligence on Teaching and Learning in Higher Education. RPTEL, 12(1), 22. DOI: 10.1186/s41039-017-0062-8

8. Johnson L. et al. (2015). NMC Horizon Report: 2015 Higher Education Edition. The New Media Consortium.

9. Ali R., Magliaro S. (2020). Effectiveness of Adaptive Learning Systems in Higher Education. Journal of Educational Technology, 48(3), 112–128.