Learning Path Dashboard For Enhancing Skills: A Personalized Adaptive E-Learning System

Bhuvaneshvari[1], Gowthami T[2], Roopitha S[3], Aniket G Desai[4]

Abstract- *Personalized learning is increasingly important in modern education, addressing diverse learning styles and indi-vidual skill gaps. This project presents a frontend prototype of a Learning Path Dashboard that simulates AI-inspired adaptive learning using mock data and rule-based logic. The system demonstrates visualization of learning trajectories, skill progres-sion tracking, interactive learning path generation, and mentor guidance without requiring backend infrastructure or AI imple-mentation. The prototype validates UI flow, feature integration, and user experience design, providing a foundation for future AI-driven recommendation systems and backend services.*

Keywords: Personalized Learning, Adaptive E-Learning, Learning Path Visualization, Frontend Prototype, Mock Data Simulation, Skill Gap Analysis.

I. INTRODUCTION

Continuous skill development is essential in rapidly evolv-ing fields such as data science, artificial intelligence, cloud computing, and software engineering. According to the World Economic Forum's Future of Jobs Report, over 50% of em-ployees will require significant reskilling by 2025 due to automation and emerging technologies.

Traditional e-learning platforms often provide static content that does not adapt to individual learner needs, resulting in low engagement and difficulty in identifying skill gaps. The challenge of personalized education has been recognized since Benjamin Bloom's work on the "2 Sigma Problem," which showed that one-on-one tutoring significantly outperforms traditional classroom instruction.

This project introduces a frontend prototype of a Learning Path Dashboard designed to simulate personalized learning experiences. Using structured mock data and rule-based logic, the system visualizes skill progression, generates sequential learning paths, and demonstrates adaptive learning mecha-nisms. The prototype objectives include: (1) creating an intu-itive interface for tracking skill development, (2) implementing visual representations of learning paths with prerequisites, (3) simulating adaptive recommendations through rules, and (4) providing a scalable foundation for future AI and backend integration.

II. RELATED WORK

a. Adaptive Learning Systems

Adaptive e-learning and intelligent tutoring systems employ AI techniques such as collaborative filtering, content-based

All authors are with the Department of Computer Science and Systems Engineering, REVA University, Bengaluru, India.

recommendations, and knowledge graphs to personalize learn-ing. Systems like Carnegie Learning's Cognitive Tutor and ALEKS use cognitive models to adjust instruction based on learner performance, but require substantial development effort and infrastructure.

b. Learning Path Optimization

Learning path optimization leverages graph-based models, where skills and learning resources are nodes with prerequisite edges. Algorithms such as topological sorting and shortest path methods generate ordered learning sequences. Recent research incorporates learner preferences, difficulty levels, and time constraints into path generation.

C. Differentiation of Current Work

Unlike full adaptive learning systems requiring complete backend stacks and AI models, this work focuses on frontend-only simulation. It demonstrates user interaction patterns, visualization techniques, and rule-based adaptation without machine learning infrastructure. This enables rapid prototyping and user experience validation before investing in complex AI and backend services.

III. COMPARATIVE ANALYSIS OF EXISTING SYSTEMS

Table I compares major e-learning approaches with the proposed prototype.

TABLE I
Comparison of E-Learning Systems

System	Personalized	Adaptive	Analytics	Path Viz
Generic MOOCs	No	No	Limited	No
Rule-Based LMS	Partial	No	Limited	No
AI Tutoring	Yes	Partial	Moderate	Partial
Prototype	**Yes (Mock)**	**Simulated**	**UI Only**	**Yes**

Generic MOOCs offer fixed sequences with minimal adap-tation. Rule-based LMSs add conditional logic but lack so-phisticated adaptation. AI tutoring systems provide genuine personalization but are costly. The prototype demonstrates adaptive concepts at the UI level with minimal infrastructure.

IV. PROBLEM STATEMENT

Modern learners face several interconnected challenges:

- **Lack of visual guidance:** Traditional platforms provide text-based progress indicators without intuitive visualiza-tions showing skill progression and prerequisites.
- **Unclear sequencing:** Learning resources are often pre-sented in arbitrary order rather than sequences optimized for knowledge building, leading to confusion and frustra-tion.
- **Limited engagement:** Static content delivery without personalization results in decreased motivation and higher dropout rates.
- **Difficulty identifying skill gaps:** Without comprehensive assessment tools, learners cannot efficiently prioritize learning efforts.
- **Minimal guidance:** Many platforms lack mentorship features, leaving learners to navigate complex skill land-scapes independently.

The prototype addresses these challenges using mock data and interactive UI elements that simulate adaptive learning paths, skill gap highlighting, and mentor-style guidance.

V. LEARNING PATH DASHBOARD PROTOTYPE

The prototype consists of four interconnected modules.

A. User Profiling Module

Learner information is collected via multi-step forms cap-turing:

- Basic details and educational background
- Current skill levels across domains
- Learning goals and target roles
- Time availability and learning preferences

Predefined mock profiles (e.g., ”Aspiring Data Analyst,” ”Career-Switching Developer”) demonstrate different scenar-ios and validate system handling of various starting points.

B. Skill Progress Panel

Skills appear as interactive cards with progress bars and color-coded proficiency levels (Beginner, Intermediate, Ad-vanced, Expert). Each card displays:

- Current proficiency percentage
- Visual progress indicator with color gradients
- Proficiency level badge
- Recommended next steps
- Time estimate to next level

Progress updates are simulated using predefined activity data and rule-based state changes to reflect learning ad-vancement. The panel responds visually with animations and updated statistics based on these simulated state transitions.

C. Learning Path Visualization

Learning paths are displayed using interactive graph visu-alizations showing:

- Nodes representing skills or modules
- Directed edges indicating prerequisites
- Color coding: green (completed), yellow (in-progress), gray (future/locked)
- Alternative paths for different backgrounds
- Estimated completion time per segment

Rule-based logic implements path generation through topo-logical sorting. The algorithm identifies target role require-ments, compares with current skills, finds gaps, and orders missing skills respecting prerequisites. Users can interact with the visualization to explore alternatives and understand recommendation rationale.

D. Mentor Panel

The mentor interface provides contextual guidance through:

- Personalized welcome messages based on profiles
- Progress-based encouragement
- Tips for overcoming common challenges
- Resource recommendations (articles, videos, platforms)
- Answers to frequently asked questions

The current mentor module is a UI-level prototype using predefined templates and does not perform natural language understanding. Messages are template-based with variable insertion based on learner state. Future versions will incorpo-rate natural language processing for dynamic, conversational interactions.

VI. UI WALKTHROUGH AND MOCK DATA

A. Sample Learner Profile

TABLE II
Sample Learner Profile (Mock Data)

Field	Sample Data
Name	Alice Smith
Current Skills	Python (Intermediate), SQL (Beginner)
Goal	Data Analyst
Weekly Hours	10–15
Learning Style	Visual + Hands-on

B. *Skill Mapping (Mock)*

TABLE III
Sample Skill Mapping (Mock)

Skill	Prerequisite	Est. Hours
Data Visualization	Python Basics	20
Statistics	Math Fundamentals	30
SQL Queries	None	15
ML Basics	Python + Statistics	40
Pandas	Python Basics	30

C. *User Journey Example*

For Alice targeting a Data Analyst role, the system:

1. Maps required skills (Python, SQL, Statistics, Data Visualization, Excel)
2. Compares with current skills (Python: Intermediate, SQL: Beginner)
3. Identifies gaps (Statistics, Data Visualization, Advanced SQL)
4. Generates personalized path:
    - Week 1-4: Statistics Fundamentals (30 hours)
    - Week 5-7: Advanced SQL (25 hours)
    - Week 8-10: Data Visualization (20 hours)
    - Week 11-16: ML Basics (40 hours)
5. Displays visual path with mentor guidance

This workflow demonstrates how mock data enables realis-tic personalized trajectory simulation.

VII. SYSTEM ARCHITECTURE (FRONTEND)

The system follows a modular, component-based design using **React.js with Vite** build tooling.

A. Component Structure

- **App (Root):** Top-level component managing routing and global state
- **Dashboard Container:** Layout with header, navigation, and content area
- **Profile Manager:** Components for profile creation and editing
- **Skill Tracker:** Skill visualization components (cards, progress bars, badges)
- **Path Visualizer:** Graph visualization components for learning paths
- **Mentor Interface:** Guidance display components

B. State Management

React Context API manages application state:

- **UserContext:** Learner profile, skills inventory, prefer-ences
- **PathContext:** Selected path, completed nodes, progress state
- **MentorContext:** Message history and guidance state

C. Data Layer

Mock data is organized as JSON files:

- **learners.json:** Learner personas with skills and goals
- **skills.json:** Skill definitions with prerequisites and re-sources
- **roles.json:** Career roles mapped to required skills
- **mentorMessages.json:** Template messages keyed by state

A DataService module abstracts data access, making future API integration straightforward by replacing JSON reads with HTTP requests without changing UI logic. All analytics are simulated at the frontend level using computed state derived from mock datasets.

VIII. DEVELOPMENT TOOLS AND APPROACH

A. Development Methodology

Agile, sprint-based development with two-week cycles in-cluded:

- Sprint planning with feature prioritization
- Daily progress tracking
- Component development with testing
- UI/UX review sessions
- Sprint retrospectives

Design-first approach using Figma ensured visual consis-tency before implementation.

B. Tools and Platforms

- Git/GitHub for version control
- VS Code for development
- React.js 18+ with Vite
- Tailwind CSS for styling
- D3.js and Recharts for visualizations
- React Testing Library for component tests
- Mock JSON data files

C. Proposed Future Enhancements

- Backend REST APIs (Node.js/Express or Python/FastAPI)
- Database integration (MongoDB or PostgreSQL)
- JWT-based authentication
- AI-based recommendation engine
- NLP-powered mentor chatbot
- Real-time progress tracking
- Mobile app using React Native
- Integration with learning platforms

IX. PROTOTYPE EVALUATION

Evaluation uses mock data scenarios to validate:

- Correct profile-to-path mapping
- Proper prerequisite enforcement
- Interactive skill progression visualization
- Context-appropriate mentor messages
- Multiple learner personas (beginner, intermediate, ad-vanced)

A. Test Scenarios

Scenario 1 - Complete Beginner: Profile with no technical skills targeting web developer. System generates comprehen-sive path from HTML/CSS through JavaScript and React.

Scenario 2 - Career Switcher: Marketing background targeting data science. System identifies programming and math gaps, creating longer foundational path.

Scenario 3 - Advanced Learner: Strong Python/ML back-ground targeting AI researcher. System focuses on advanced topics like deep learning and research methods.

Mock validation confirms correct prerequisite handling, responsive UI across devices, and proper state propagation across components.

X. DISCUSSION

A. Key Achievements

The prototype successfully demonstrates:

- Frontend simulation of adaptive learning without back-end/AI
- Visual skill gap analysis helping learners understand next steps
- Rule-based path generation creating logical learning se-quences
- Scalable architecture facilitating future enhancements
- Effective stakeholder communication through interactive prototype

B. Limitations

- Reliance on predefined mock data without real learner interactions
- No backend infrastructure for persistence or authentica-tion
- No genuine AI-driven personalization
- Limited to manually created skill mappings
- Template-based mentor guidance rather than conversa-tional AI
- User behavior and engagement patterns are simulated and do not reflect real-world learner variability

XI. FUTURE WORK

Future development will transform the prototype into a production system:

A. Backend and AI Integration

- RESTful APIs with database persistence
- JWT authentication and authorization
- Collaborative filtering recommendation engine
- NLP chatbot for conversational mentoring
- Reinforcement learning for path optimization

B. Advanced Features

- Real-time skill assessment through integrated challenges
- Peer collaboration and study groups
- Gamification with badges and achievements
- Calendar integration and study scheduling
- Mobile applications for iOS/Android
- External platform integration (Coursera, Udemy)

XII. SECURITY, PRIVACY, AND ETHICS

The current prototype uses mock data only. Future imple-mentation will include:

A. Security Measures

- HTTPS encryption
- Secure authentication protocols
- Password hashing with bcrypt
- Input validation and sanitization
- Rate limiting and abuse prevention

B. Privacy Protection

- GDPR compliance with clear consent
- Transparent data collection policies
- User rights for data access, export, deletion
- Data anonymization for analytics
- Encryption at rest and in transit

C. Ethical AI Considerations

When implementing AI features:

- Fairness and bias mitigation in recommendations
- Transparency in algorithmic decisions
- Human oversight of automated recommendations
- Protection against discrimination
- Regular auditing for unintended biases

XIII. CONCLUSION

This paper presents a frontend prototype of a Learning Path Dashboard that demonstrates adaptive learning concepts using mock data and rule-based logic. The system successfully showcases personalized learning path visualization, skill pro-gression tracking, prerequisite-aware sequencing, and mentor guidance through a well-designed interface.

The prototype validates core design concepts without re-quiring complex backend infrastructure or AI implementation, enabling rapid prototyping and early feedback. The modular architecture with clear component boundaries facilitates future integration of backend services, databases, and AI recommen-dation engines.

While operating with mock data and frontend logic only, the system demonstrates the feasibility and value of adaptive learning platforms. It establishes a foundation for evolving into a production-ready system that can scale personalized education to large populations while maintaining the intuitive user experience established in this prototype.

REFERENCES

[1] K. Schwab, *The Fourth Industrial Revolution*. Crown Business, 2017.
[2] World Economic Forum, “The Future of Jobs Report 2023,” 2023.
[3] B. Bloom, “The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring,” *Educational Researcher*, vol. 13, no. 6, pp. 4–16, 1984.
[4] K. VanLehn, “The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems,” *Educational Psycholo-gist*, vol. 46, no. 4, pp. 197–221, 2011.
[5] P. Brusilovsky and E. Milla´n, “User Models for Adaptive Hypermedia and Adaptive Educational Systems,” in *The Adaptive Web*, Springer, 2007, pp. 3–53.
[6] C. M. Chen, “Intelligent Web-based Learning System with Personalized Learning Path Guidance,” *Computers & Education*, vol. 51, no. 2, pp. 787–814, 2008.