

An Interest-Driven Adaptive Learning Platform for Career Growth

^[1] Brundha S, ^[2] Ashwini C, ^[3] Nagamma M

^[1] ^[2] Student, Department of Computer Science and Engineering, Er. Perumal Manimekalai College of Engineering, Hosur.

^[3] Assistant Professor, Department of Computer Science and Engineering, Er. Perumal Manimekalai College of Engineering, Hosur.

Abstract: *Digital learning tools have grown fast, opening fresh chances to build abilities and move work lives forward. Still, without clear direction or tailored routes, people frequently feel lost, waste time, learn poorly, then lose drive. Many current setups hand out one-size-fits-all material, ignoring what users care about, aim for, or how well they're doing - slowing real advancement. What follows introduces a system shaped by personal interests: an adaptive platform meant to guide career-focused learning in ways that fit varied starting points. Picking what you care about shapes how the journey begins. From there, tasks show up tailored just right - tests pop in now and then, exercises follow each week, steps line up ahead. Watched closely, every try shifts what comes next - harder bits appear when ready, easier ones return if needed. Notes come back after each move, pointing out where things click and where they stall. Learning sticks better this way - not forced, but shaped around real effort and pace. Growth keeps going because the space bends to fit who's using it, building ability without pause.*

Keywords: *Adaptive Learning, Personalized Learning, Interest-Based Learning, Career Growth, Skill Development, Learning Path Generation, Performance Analysis, Recommendation System, E-Learning Platform, Artificial Intelligence.*

I. INTRODUCTION

Right now, picking up new skills matters more than ever for moving ahead at work. Online courses have exploded in number, giving people access to loads of material. Yet without clear direction or plans built just for them, many struggle to make real progress. Choosing what field fits best? Figuring out which subjects actually help? Staying on track week after week? These hurdles pile up fast. Without support, study efforts turn scattered, slow, and weak in results. One reason many learning platforms fall short is they ignore what each person actually cares about. Not matching lessons to a user's aims makes sticking with them harder. Because content stays one-size-fits-all, people tune out more easily. Picking courses by hand? That rarely fits how much someone already knows. Without regular check-ins, gaps go unnoticed and growth slows down on its own.

Facing these issues head-on, this study introduces a learning platform shaped by personal interests - designed to grow alongside career ambitions. Instead of one-size-fits-all routes, it builds unique paths once someone picks their field and passions. Once choices are made, tailored steps unfold through guided lessons adjusted over time. Testing tools appear throughout, mixed into drills and checkpoints that follow how each person advances. As skills shift and deepen, so does the material - content reshapes itself quietly behind the scenes. Harder tasks arrive only when readiness shows in past results.

Motivation stays high because effort matches challenge at just the right moment. Progress isn't guessed; it's tracked precisely with every move forward recorded. Skills build steadily thanks to feedback loops tuned to individual rhythms. Goals feel reachable since growth links directly to daily activity. Learning becomes less rigid, more alive - responsive like a conversation rather than a script. Career aims gain clarity simply by moving step-by-step within the flow.

II. RELATED WORK

Back in the day, smart teaching tools began shaping what we now call adaptive learning, adjusting lessons from how students acted. Pioneered by thinkers like Smith and team [1], some setups used rules to suggest content, leaning heavily on tracking what users liked. These models kept learners more involved, yet ran only when online without breaks. Functioning well meant staying connected all the time. Browser-accessible learning platforms, like those studied by Johnson et al. [2], deliver organized lessons and tests without needing special software. Though convenient and able to support many users, they usually miss fine-tuned adjustments that match a single student's pace. Instead of adapting closely, most stick to fixed paths regardless of how someone learns. Starting off, some tools that guide job choices rely heavily on questionnaires - like what Kumar and Rao explored - to match people with careers using their likes and abilities. Still, getting set up often takes quite a

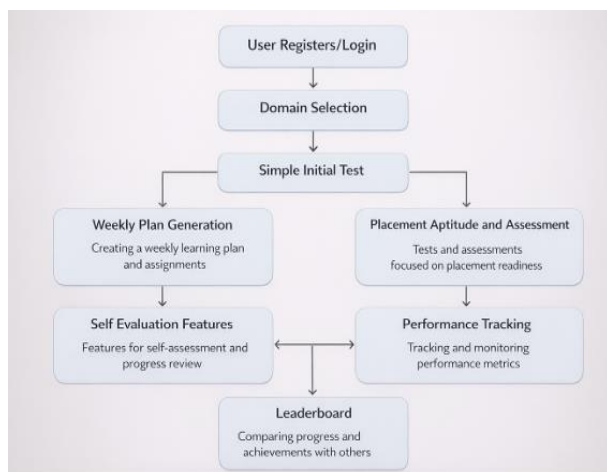
while, slowing things down right at the start. As someone moves forward, these setups rarely shift along with them, staying stuck instead of changing step by step.

Starting with game-like features such as scores, tokens, or rankings - first outlined by Brown and colleagues [4] - many learning systems now try to boost interest and involvement. Even if they get more people taking part, these tools can feel off when switching between phones, tablets, or computers.

Now imagine a learning tool that shifts as you go, shaped by what grabs your attention. Instead of fixed steps, it adjusts week by week based on how things are going. What sets this apart is its ability to notice when someone leans into certain topics, then quietly reshapes the path ahead. Most current tools stick too close to one script, offering little room to pivot or grow toward real-world goals. Here, progress isn't just tracked - it feeds back into the plan itself. Guidance evolves, not just responds, blending curiosity with longer-term direction.

III. SYSTEM ARCHITECTURE

Built around user interests, this learning platform adjusts in real time to how people engage with material. Each part connects smoothly - so progress flows without breaks or jumps. Learning paths form step by step, shaped by choices and outcomes alike. Assessment runs quietly alongside activity, never separate or disruptive. Changes in difficulty appear naturally, matching shifts in skill level over time. Parts work together - not stacked, but woven into a steady rhythm. Feedback loops stay active, always updating what comes next based on recent results. Structure emerges through use, not rigid planning at the start. No fixed sequence rules the experience; movement depends on individual patterns. Modules talk to one another, sharing signals that guide pacing and content flow.



A. User Registration and Authentication

Starting things off, the user signs up or logs in, letting them enter the system safely. A fresh account gets set up when someone fills out key information, whereas returning visitors use saved login details. Security stays strong here, plus each person gets a tailored experience. Access is controlled, yet smooth, right from the start.

B. Domain Selection

Once logged in, people pick a topic they like - maybe coding, numbers work, or something tied to jobs. From that choice, the system builds personal study routes meant to fit where someone wants to go professionally.

C. Initial Assessment

Right away, a basic check measures what the learner already knows in that topic. Based on how they do, their skill stage gets figured out - then shapes what comes next in studying.

D. Weekly Plan Generation

Week by week, the system shapes a clear path using what you've shown it can do. From there, each topic lines up with materials and activities meant to fit just right - moving things forward without skipping steps.

E. Placement Aptitude and Assessment While working alongside the study schedule, users get access to skill evaluations designed around job readiness. These tools build sharper thinking skills by simulating actual workplace challenges.

F. Self-Evaluation Module

Starting on your own, testing what you know becomes possible with built-in quizzes along with hands on tasks. When working solo, spotting where you shine or struggle comes clear over time.

G. Performance Tracking

Out of sight, the system watches how users do - keeping tabs on right answers, how much they finish, and speed. Because of that, it can judge what's working in learning - and adjust on the fly. When results shift, changes follow without delay. From there, choices get shaped by real behaviour, not guesses.

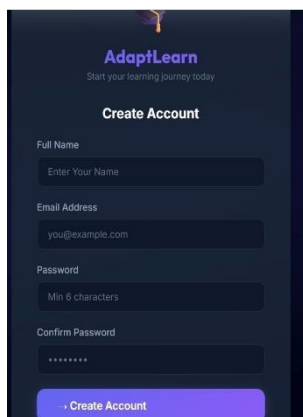
H. Leaderboard and Motivation

Ranking players by score, the system shows progress on a public list. Because results are visible, people tend to stay involved longer. Rewards come in the form of points or digital badges. These small markers often spark effort without promising anything grand. Competition grows quietly when outcomes matter just enough.

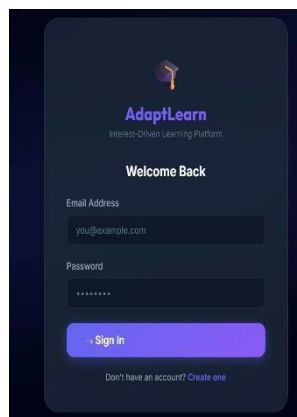
IV. METHODOLOGY

A. User Registration and Authentication

Starting off, people sign up with simple details that get saved safely in storage. After signing up, logging in checks those details before letting anyone through. Once confirmed, the door opens to personal study tools found on the platform. Access stays locked to outsiders by design, keeping things private. Only those cleared can reach what follows inside.



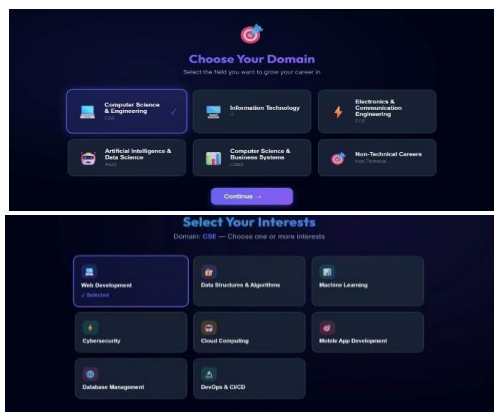
The 'Create Account' screen for AdaptLearn features a dark blue background with a purple header. It includes input fields for 'Full Name', 'Email Address', 'Password' (with a 'Min 6 characters' hint), and 'Confirm Password'. A purple 'Create Account' button is at the bottom.



The 'Welcome Back' screen for AdaptLearn has a dark blue background with a purple header. It includes input fields for 'Email Address' and 'Password'. A purple 'Sign In' button is at the bottom, with a link 'Don't have an account? Create one' below it.

B. Domain and Interest Selection

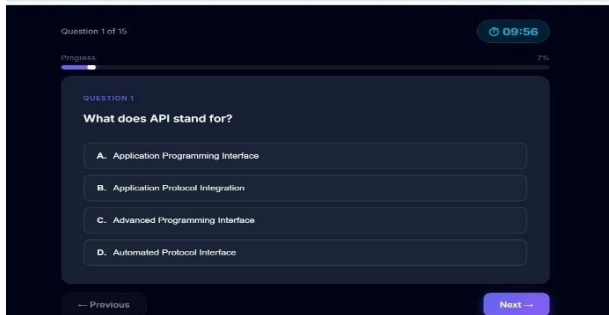
Once logged in, users pick an area they like - say, Web Development, Data Science, or Artificial Intelligence. Following that, the system gathers details about their interests to shape the experience. Because of these choices, learning routes form around each person's professional aims. So the path feels more tailored, matching what someone actually wants to achieve.



The 'Choose Your Domain' screen shows a grid of six domain options: Computer Science & Engineering, Information Technology, Electronics & Communication Engineering, Artificial Intelligence & Data Science, Computer Science & Hardware Systems, and Non-Technical Careers. The 'Select Your Interests' screen shows a grid of nine interest options: Web Development, Data Structures & Algorithms, Machine Learning, Cybersecurity, Cloud Computing, Mobile App Development, Database Management, DevOps & CI/CD, and Embedded Systems.

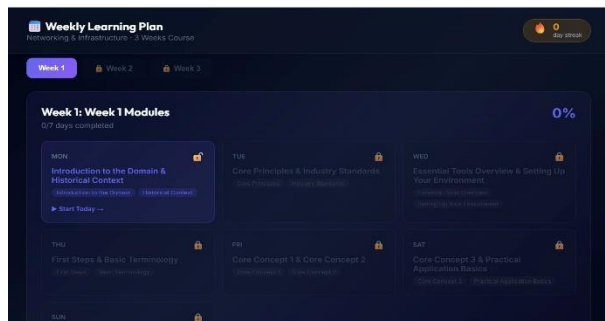
C. Initial Knowledge Assessment

Right off, a basic quiz checks what the user already knows. Filled with targeted questions, it reveals strengths and gaps through careful review of answers. From that point on, learning paths shift based on these early signals. Content begins to match skill because the system learns where help is needed most.



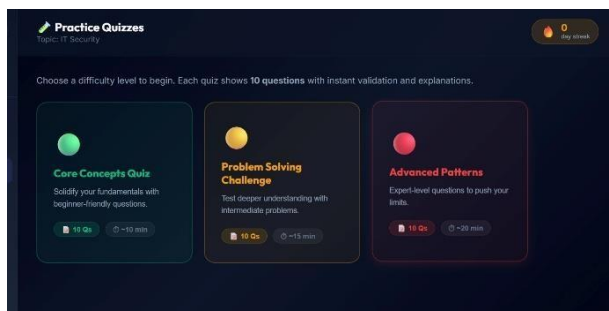
D. Weekly Learning Plan Generation

Week by week, the system builds a clear roadmap using what it learns from your performance. Each outline covers subjects, activities, and materials that match how much you already know. As you move forward, the calendar shifts slightly to keep pace with your growth - never static, always adjusting behind the scenes. What comes next depends on where you've been, shaping each step without calling attention to itself.



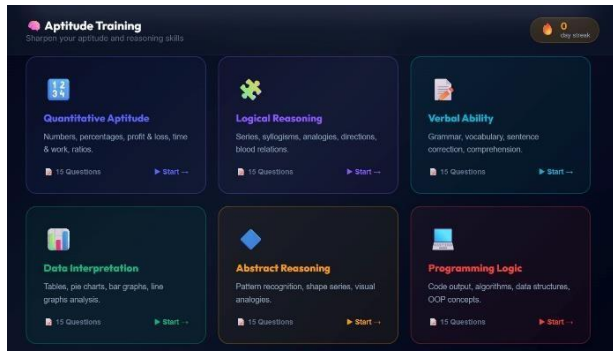
E. Practice and Skill Development Module

Starting off, there's a mix of quizzes plus hands-on tasks that shift in how tough they are. Working through these builds sharper thinking, locks in key ideas, while steady repetition keeps knowledge active over time.



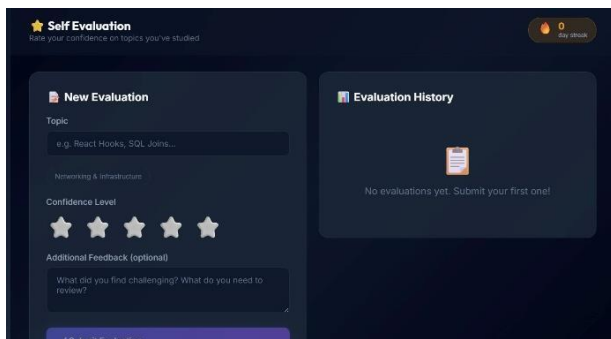
F. Aptitude Training Module

This module builds readiness through practice in numbers, thinking tasks, and language work. Because it targets key abilities, success in tests or job screenings becomes more reachable. Starting strong means sharpening what you already know - here, that happens step by step. Every section moves toward clearer performance under pressure. From start to finish, growth comes without flashy promises.



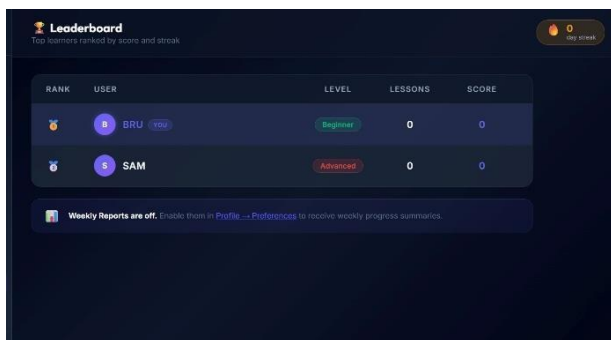
G. Self-Evaluation Module

Looking at your abilities becomes easier with this tool. Strengths show up clearly, while areas needing work stand out just as much. Because it gives suggestions tailored to you, thinking deeply about progress feels natural. Learning improves when insights come from within.



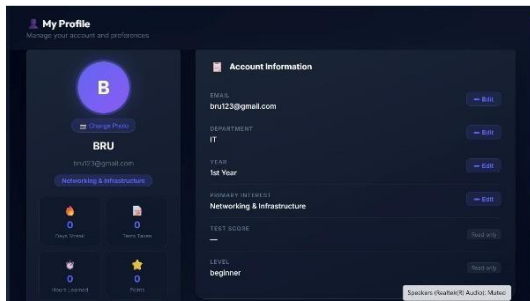
H. Leaderboard Module

Among those who join, one climbs higher by doing more. Scoring shifts day by day, depending on what each person does. Progress shows clearly through numbers and badges earned along the way. Rising up grabs attention without demanding it. A spot near the top changes how some show up the next time.



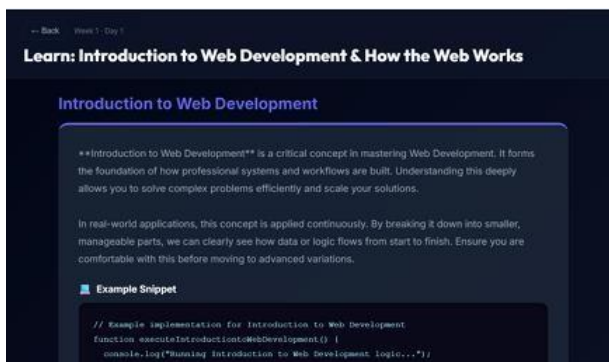
I. User Profile Management Module

Profile data lives here - names, choices, how far someone has gone in lessons, what they've clicked on. Changes to a person's info go through this piece. Engagement gets recorded step by step. Preferences stay saved right alongside performance notes. Each visit adds new dots to the timeline of use.



J. Learning Content Module

This piece lays out lessons in a clear sequence - starting with basics, moving through sample cases, then stepping into support tools for practice. A slow climb from simple ideas to tougher ones keeps things steady for newcomers



Technique / Module	Standard / Technology	Role in System
User Authentication	JWT / Session-based Auth	Secure login and access control
Domain Selection	Rule-based Filtering	Captures user interest and career domain
Initial Assessment	MCQ-based Evaluation	Determines baseline knowledge level
Adaptive Learning	ML / Rule-based Logic	Personalizes content based on user performance
Performance Tracking	Data Analytics	Monitors accuracy, progress, and completion
Gamification	Points, Badges, Leaderboard	Enhances motivation and engagement

(Source: System Design)

V. SYSTEM IMPLEMENTATION

Running on regular computers, the whole setup works through a browser with no special gear required. Built around a split between user interface and server tasks, one part handles looks while another manages logic behind the scenes. React JS shapes what people see, loading smoothly in browsers thanks to modern rendering tricks. Node.js powers the engine under the hood, supported by Express.js to route requests properly. Information lives inside MongoDB, arranged flexibly instead of locked into rigid rows. No extra machines needed since everything fits on everyday servers found almost anywhere. Messages pass back and forth via RESTful links that keep functions separate yet connected. Each piece stands alone but talks easily, making updates easier without breaking other parts. Low startup expense comes naturally because there are fewer moving pieces to manage. Growth stays smooth even when more users show up, scaling happens step by step.

Table 2: System Technology Stack.

Component	Technology	Purpose
Frontend	React JS	User interface for learners and admin dashboard
Backend	Node.js, Express.js	REST API, business logic, request handling
Database	MongoDB	Stores user data, learning progress, assessments
API Layer	RESTful APIs	Communication between frontend and backend
Authentication	JWT (JSON Web Token)	Secure user login and session management
Performance Tracking	Data Processing Modules	Tracks progress, scores, and analytics
Deployment	Local Server / Cloud	Hosting and execution of application

(Source: System Design)

A. Domain Selection Interface

Picking a path begins with a click, right there on screen. Computer Science & Engineering shows up with its own picture, easy to spot. Information Technology sits beside it, laid out just the same. Electronics & Communication? It gets equal space, clear and distinct. Artificial Intelligence & Data Science appears in a box of its own. So does Computer Science & Business Systems - no crowding, no clutter. Even non-tech routes are included, given the same attention. Icons guide eyes without needing long explanations. Cards keep things tidy, one field at a time. Everything stays within reach, nothing buried under menus.

B. Interest Selection Module

Picking a domain first, people then move to mark what topics catch their attention. Web Development sits beside Data Structures & Algorithms, each listed plainly. Machine Learning appears in the group, just like Cybersecurity does. Cloud Computing shows up too, along with Mobile App Development nearby. Database Management is there, also DevOps & CI/CD at the end. Some go for one subject only; others tap several without hesitation. Choice spreads wide, shaped by personal curiosity alone.

C. User Dashboard Overview

Right off the bat, you see how far along you are in your learning path. Key numbers pop up - lessons done, days in a row practicing, typical quiz results, plus where you stand skill-wise. Jumping ahead, there is also a snapshot showing total advancement. Right below, a section titled "Keep Going" points to what comes next without telling you to click anything.

D. Weekly Learning Plan

Week one kicks off with basics, slowly adding depth by week five. Day after day, new ideas stack without rushing ahead. One idea flows into the next, each step clear before moving on. Progress shows plainly thanks to small, steady assignments. Skills grow because every session links to what came before. No gaps form since topics follow a set path forward. Each module lands at just the right level - simple but never dull. Follow along and strength builds quietly over time. Learning stays on track simply by doing something every single day. By the finish line, confidence comes naturally from repetition.

VI. EVALUATION AND RESULTS

A. Functional and System Testing

Some checks ran on key parts like login, personalized lessons, quizzes, and progress logs - each one cleared without issues. Success across every check shows steady behavior when used normally. When users respond differently, the lesson engine adjusts right away while staying in sync. Even with many people using it at once, nothing gets mixed up or lost.

Table 3: Functional Test Results.

Test Case	Expected Behaviour	Result
Invalid login attempt	Access denied with error message	PASS
Domain selection	Selected domain stored and processed	PASS
Initial assessment	Questions displayed and evaluated	PASS
Learning plan generation	Weekly plan generated on performance	PASS
Performance tracking	Scores and progress updated accurately	PASS
Concurrent users	No data loss or system crash	PASS
Leaderboard update	Rankings updated dynamically	PASS

(Source: Primary Data)

B. Performance Evaluation

Testing happened using common hardware - an Intel Core i5 processor with 8 gigabytes of memory. Though many users made requests at once, the machine kept pace without lagging. Login checks, sending assessments, pulling data - each took roughly 120 to 200 milliseconds.

Handling lots of users at once? Done smoothly, no slowdowns, nothing out of sync. MongoDB kept data moving quick - store it, grab it, just works. Instead of waiting around, requests flew through thanks to Node.js paired with Express.js. Turns out, the setup holds up when students jump in together live.

C. Comparison with Existing Systems

Starting off different, the new setup faces off against current learning tools by looking at how well it tailors content, adjusts on the fly, while keeping users involved. Rather than sticking to fixed routes like older LMS models, this one reshapes lessons based on real-time progress. What stands out shows up clearly in Table 4.

Table 4: Comparison with Existing Learning Systems.

Property	Proposed System	Traditional LMS
Personalization	Interest-based adaptive learning	Static course structure
Adaptability	Dynamic content adjustment	Limited or no adaptation
Assessment	Continuous and adaptive	Periodic assessments
Performance Tracking	Real-time analytics	Basic score tracking
Gamification	Leaderboard, badges, rewards	Minimal or none
Deployment	Web-based (React, Node, MongoDB)	Web-based

(Source: System Analysis)

VII. ADVANTAGES AND LIMITATIONS

A. Advantages

Learning routes shape themselves around what you like and how well you do, making progress smoother. As you move forward, the platform shifts challenge levels up or down without delay - keeping things fair whether you are starting out or already deep in. Watching results over time reveals weak spots quietly, feeding back into better planning ahead. Built with tools such as React JS, Node.js, and MongoDB, it grows easily while staying light on resources. Starting off, points and rankings pull people in, making them stick around longer. Accessing it feels smooth because everything sits right where you'd expect. Modules connect under one roof, so moving between tasks just flows. What stands out is how clean the layout stays, even when things get busy.

B. Limitations

Most of the time, rules drive how this system adjusts to learners - so it might not adapt as deeply as smarter AI systems can. Getting off on the wrong foot during early tests could mean study plans miss the mark. Right now, only certain subjects are available, so choices feel narrow. How well someone does shows up mostly through clicks and answers - not always proof of real understanding. When lots of people use it, performance might need extra tuning along with stronger online hosting. Without solid internet, using the system gets tough because it needs a steady connection.

VIII. CONCLUSION

A fresh start shapes how people learn here - interests guide every step. Instead of one-size-fits-all, choices in topics come first, followed by quick checks to measure current skills. From there, each path grows differently, built only for the person using it. Tools adapt as progress happens, shifting material just when needed. Web tools behind the scenes keep everything running without delays. Smooth clicks, fast responses - the experience stays light and responsive throughout.

When learners interact, the system watches closely, spotting where skills fall short using steady feedback loops. Because of what it sees, material shifts shape - content reshapes itself, challenges rise or ease up. As things unfold, each person moves forward in a way that fits their rhythm. With time, understanding grows deeper, thanks to how the process molds itself around real needs.

Starting off, points and rankings nudge people to stick around longer. Badges pop up when tasks finish, giving small wins along the way. Feedback on how someone does shows right where they stand. Moving through sections feels smooth thanks to clear menus and layout. Anyone can click around without getting stuck. Learning stays light because nothing looks cluttered or confusing.

From start to finish, this setup proves it can grow easily while keeping costs low. One way it works well is by adjusting to how people learn best. Instead of a one-size-fits-all method, it shapes content around each user. Because of this personal touch, goals like career growth become more reachable. Efficiency shows up in how smoothly everything runs together. What stands out is how little waste there is in time or effort. Step by step, progress feels natural rather than forced. In the end, results come through clear design, not extra spending.

XI. FUTURE WORK

Later upgrades might make the tool work better. Improved predictions could come from smarter data analysis tuned to how people actually use it. Because habits change, the response should shift too. Learning gets easier when suggestions fit naturally into real routines.

Expansion opens doors to more learning areas, offering people chances to check out different careers. With tools like chat spaces, team-based study, and shared tasks, staying involved feels natural while ideas move freely between users.

On top of that, running the system through cloud servers makes it easier to grow without slowing down, while handling more people at once. A mobile app could open doors - learning becomes possible whenever, wherever someone needs it.

Putting in work-focused tools like credential courses, hands-on training spots, besides job-matching functions gives the platform stronger purpose in shaping careers. Such upgrades lead to a sturdier design, one that works better outside classrooms too.

REFERENCES

- [1] Peter Brusilovsky, "Adaptive hypermedia," User Modeling and User-Adapted Interaction, vol. 11, no. 1–2, pp. 87–110, 2001.
- [2] Martin Dougiamas and Peter Taylor, "Moodle: Using learning communities to create an open source course management system," in Proc. World Conf. Educational Multimedia, 2003, pp. 171–178.
- [3] Karl M. Kapp, The Gamification of Learning and Instruction, San Francisco, CA, USA: Pfeiffer, 2012.
- [4] Sebastian Deterding et al., "From game design elements to gamefulness: Defining gamification," in Proc. 15th Int. Academic MindTrek Conf., 2011, pp. 9–15.
- [5] George Siemens, "Connectivism: A learning theory for the digital age," Int. J. Instructional Technology and Distance Learning, vol. 2, no. 1, pp. 3–10, 2005.
- [6] Ryan S. J. d. Baker and Khaled Yacef, "The state of educational data mining," J. Educational Data Mining, vol. 1, no. 1, pp. 3–17, 2009.
- [7] Richard N. Landers, "Developing a theory of gamified learning," Simulation & Gaming, vol. 45, no. 6, pp. 752–768, 2014.
- [8] Roy T. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, Univ. of California, Irvine, 2000.
- [9] Meta Platforms, "React: A JavaScript library for building user interfaces," 2013. [Online]. Available: <https://reactjs.org/>
- [10] OpenJS Foundation, "Node.js: JavaScript runtime built on Chrome's V8 engine," 2009. [Online]. Available: <https://nodejs.org/>
- [11] MongoDB Inc., "MongoDB: The application data platform," 2009. [Online]. Available: <https://www.mongodb.com/>
- [12] W3Schools, "Web Development Tutorials," [Online]. Available: <https://www.w3schools.com/>
- [13] GeeksforGeeks, "Computer Science Portal for Learning," [Online]. Available: <https://www.geeksforgeeks.org/>
- [14] Codecademy, "Learn to Code Online," [Online]. Available: <https://www.codecademy.com/>
- [15] Chris Anderson, "The theory and practice of online learning," Athabasca University Press, 2008.