**Project Report:**

**SparkGen - AI-Powered Interactive Learning Games**

**1. Introduction**

In today’s digital age, traditional learning methods often fail to engage students effectively. SparkGen is an AI-powered interactive learning platform featuring AI-generated quizzes, achievement tracking, and an AI chatbot assistant for enhanced learning. Built with React, Node.js (Express), and MongoDB, it uses the DeepSeek R1 API via OpenRouter to generate dynamic quizzes and provide chatbot support.

**2. Problem Statement**

**Gen AI Interactive Learning Games**

Traditional learning environments lack interactivity, adaptability, and engagement, leading to lower retention rates among students. Current educational platforms either provide static content or require extensive manual effort to create personalized learning experiences. There is a need for a solution that integrates AI to dynamically generate interactive quizzes and learning activities tailored to individual learners.

**3. Objectives**

* Develop an AI-driven platform for interactive learning.
* Implement Generative AI to create personalized quizzes and challenges.
* Enhance user engagement through gamification elements such as XP, levels, and achievements.
* Provide real-time feedback and adaptive difficulty for improved learning outcomes.

**4. Technology Stack**

* **Frontend:** React.js (HTML + React for UI), served using Vite
* **Backend:** Node.js + Express.js
* **Database:** MongoDB Atlas, with Mongoose for schema management
* **AI Integration:** DeepSeek R1 API (via OpenRouter**)**
* **Deployment:** Vercel (frontend) - [sparkgen-chi.vercel.app](https://sparkgen-chi.vercel.app), Render (backend) , MongoDB Atlas (database)
* **UI Framework:** Shadcn-UI with Tailwind CSS

**5. Features**

**5.1 Gamified Learning**

* AI-powered quizzes: Math Quiz, Science Quiz, General Knowledge Quiz (Escape Room style), Riddles, Gen AI Image Finding Game, Grammar Quiz, and Word Scramble.
* Points, levels, and achievement badges to incentivize learning.
* Leaderboards for competitive engagement.

**5.2 AI-Powered Personalization**

* AI chatbot for interactive guidance and hints, available on all pages.
* AI-generated quizzes with adaptive difficulty.
* Real-time feedback and recommendations based on user performance.

**5.3 User Engagement & Progress Tracking**

* Profile page displaying XP, levels, and achievements.
* Leaderboards for competitive learning.
* AI-powered performance analysis and study recommendations.

**6. Implementation Details**

* **Frontend:** React.js-based UI with client-side routing via react-router-dom.
* **Backend:** Express-based REST API with JWT authentication (bearer access and refresh tokens).
* **Database:** MongoDB Atlas for cloud-based user data storage and quiz tracking.
* **AI API Integration:** DeepSeek R1 API for dynamic quiz generation.
* **Hosting & Deployment:**
  + Frontend on Vercel - [sparkgen-chi.vercel.app](https://sparkgen-chi.vercel.app)
  + Backend on Render
  + Database on MongoDB Atlas

**7. Deployment**

* GitHub Repository: [charankarthic/sparkgen](https://github.com/charankarthic/sparkgen)
* Demo Video: [Watch Here](https://drive.google.com/file/d/1xBifSdpTJnbtQwSsuKARz_7fXxjyykQR/view?usp=drivesdk)
* Setup Steps:
  1. Clone the repository: git clone https://github.com/charankarthic/sparkgen.git
  2. Install dependencies (npm install) for both frontend (client/) and backend (server/)
  3. Configure .env files for API keys and database connection.
  4. Start local development: npm run start (frontend at http://localhost:5173, backend at http://localhost:3000)
  5. Deploy on Render (backend) and Vercel (frontend).

**8.Design and Working**

**8.1 Design:**

* Amonochromatic look (black & white) for a professional and clean UI. A left-side menu bar is included, consisting of the Home, Games, Achievements, and Profile pages. The interactive scoreboard updates based on user performance stored in MongoDB. Users can track achievements, leaderboard scores, delete accounts, or log out as needed. A floating chatbot button serves as an AI assistant, explaining answers with steps. The website also supports a night/day mode toggle.

**8.2 Working**

1. **User Authentication:** Once logged in, user data is stored in the MongoDB Atlas database.
2. **Starting a Quiz:** When a user clicks “Start Game,” a user prompt is generated, including difficulty level, quiz type, and user level.
3. **AI Quiz Generation:** The DeepSeek R1 API (via OpenRouter) is used to generate five multiple-choice questions (MCQs) based on the user prompt.
4. **Processing API Request:** The request is sent to OpenRouter API, and the generated quiz questions are displayed on the website.
5. **Retry Mechanism for API Calls:** The system will attempt three retries to fetch quiz questions from the OpenRouter API. If all three attempts fail, preset mock data will be displayed to ensure seamless user experience and avoid errors.
6. **Submitting Answers**: Once all questions are answered, the Submit button is enabled, allowing users to submit their quiz.
7. **Score Calculation & Review:**
   * The system calculates the score based on correct answers.
   * Users can review their answers in the Review section, where mistakes are highlighted.
8. **Achievements & Leaderboard:**
   * The user’s progress, scores, and leaderboard rankings are updated.
   * Correct answers reward users with higher scores and achievements.

**9. Future Scope and Development**

* We aim to expand Sparkgen by developing **more games and interactive quizzes** that are specifically tailored to different topics and content areas. This will enhance user engagement and **improve the learning experience** by providing **topic-specific, AI-driven interactive learning modules**. Future updates will focus on **personalized content adaptation, immersive learning techniques, and AI-based educational enhancements**.
* Transition from a monochromatic look to a more engaging and visually appealing UI with animations, transitions, and colour enhancements.
* Integration of AI-based interactive learning modules for more immersive experiences.
* Personalized content adaptation using AI to tailor quizzes and learning paths.
* Advanced gamification elements like badges, streaks, and additional reward systems.
* Improved AI chatbot capabilities with deeper explanations and contextual learning.
* Multimodal AI with speech recognition, AR/VR integration, and AI emotion recognition.

**10. Conclusion**

SparkGen revolutionizes learning by combining AI, gamification, and interactive quizzes. By adapting to individual learning styles, it ensures an engaging and effective educational experience, making learning more enjoyable and rewarding for students worldwide.

**Resources and Links:**

* **GitHub Repository:** [**charankarthic/sparkgen**](https://github.com/charankarthic/sparkgen)
* **Demo Video:** [**Watch Here**](https://drive.google.com/file/d/1xBifSdpTJnbtQwSsuKARz_7fXxjyykQR/view?usp=drivesdk)
* **Frontend (Vercel Live working website link):** [**sparkgen-chi.vercel.app**](https://sparkgen-chi.vercel.app)

**Loading Time Disclaimer:**

**Our backend is hosted on Render's free tier, which puts the server to sleep after 15 minutes of inactivity. As a result, the first request after inactivity may take 20-30 seconds to load. Additionally, we use the DeepSeek R1 API via OpenRouter.com, which takes around 20 seconds to generate quiz questions and chatbot responses. We appreciate your patience while the AI processes your request.**