

Gen AI Interactive Learning Games

Intel Unnati Industrial Training Program 2025



Outline

- ✓ Project Overview
- ✓ Key Features
- ✓ Platform Architecture
- ✓ Key Components
- ✓ Scoping and Considerations
- ✓ Expected Project Outcome / Deliverables
- ✓ References

Project Overview

The primary objective of this project is to:

- Develop an educational game platform that uses Generative AI to create dynamic content, challenges, and scenarios.

Key Features

Dynamic Content Generation	Game Mechanics	Natural Language Interaction	Personalized Learning Paths
<ul style="list-style-type: none">• Use a GenAI model to generate questions, mini-stories, or puzzles dynamically.• Adjust difficulty and topics based on learner progress.	<ul style="list-style-type: none">• Design interactive elements (e.g., leveling up, badges, leaderboards) that reward mastery of concepts.• Integrate real-time feedback loops for player performance.	<ul style="list-style-type: none">• Players can interact with the game using natural language, asking questions, providing answers, or even directing the narrative flow, making the experience feel more natural and intuitive.	<ul style="list-style-type: none">• Implement a recommendation system that chooses subsequent game levels or educational content based on user inputs and performance.

Platform Architecture

User Interface & Experience	Generative AI Engine	Data Analytics	Backend Infrastructure
<ul style="list-style-type: none">• Dashboard: personalized dashboard for each user• Game Environment: An interactive and visually appealing game environment that adapts to different educational themes and subjects.• Feedback System: Real-time feedback and hints to guide users through challenges.	<ul style="list-style-type: none">• Content Generation: Use AI models to generate quizzes, puzzles, and scenarios based on curriculum standards and user preferences.• Adaptive Learning: Implement algorithms that adjust the difficulty and type of content based on the user's performance and learning style.• Interest Mapping: Analyze user interactions to identify interests and tailor content accordingly.	<ul style="list-style-type: none">• Performance Tracking: Collect and analyze data on user performance to provide insights and improve content generation.• Engagement Metrics: Monitor engagement levels to identify which types of content are most effective.	<ul style="list-style-type: none">• Scalable Cloud Services: Use cloud computing to handle data storage, processing, and AI model training.• Security and Privacy: Ensure robust security measures to protect user data and comply with privacy regulations.

Key Components

User Interface & Experience	Generative AI Engine	Data Analytics	Backend Infrastructure
<p>Dashboard:</p> <ul style="list-style-type: none">Frontend Frameworks: React.js, Angular, or Vue.js for building dynamic and responsive user interfaces.Charting Libraries: D3.js or Chart.js for visualizing progress and achievements. <p>Game Environment:</p> <p>Game Engines: pyGame, Amethyst, Three.js, Phaser, godot etc for creating interactive and visually appealing game environments.</p> <p>Real-time Communication:</p> <p>Socket.io or any other system for real-time feedback and hints.</p>	<p>AI Frameworks: TensorFlow, PyTorch, or Hugging Face Transformers for developing AI models that generate quizzes and scenarios.</p> <p>❖ DO Not Use any commercially available tools or its API</p> <p>❖ Use Open-Source tools, models and frameworks as much as possible</p> <p>Reinforcement Learning: Use libraries like Stable Baselines for creating adaptive learning paths. Use collaborative filtering or content-based filtering algorithms.</p>	<p>Analytics Platforms: Google Analytics or Open Web Analytics or Plausible for tracking user performance and</p> <p>etc for creating custom analytics dashboards.</p>	<p>Scalable Cloud Services: Explore OpenStack or Kubernetes</p> <p>Serverless Computing: Explore open FaaS, Kubeless</p> <p>Database Management: Use relational databases like PostgreSQL or NoSQL databases like MongoDB for storing user data and game content.</p> <p>Version Control: Git and GitHub for source code management and collaboration.</p>

Scoping and Considerations

Ensure not scoping everything listed in slide 4

You can choose to create learning paths, quizzes, puzzles etc while not doing everything

Consider a focus area or stream or subject and build

Consider making reusable components

Create minimum viable product as prototype for 1 use case which demonstrate end to end capability

Example: Implement a basic game environment with a few static quizzes or puzzles and use prompts or analytics to create recommendations

Expected Project Outcome / Deliverables



Outcome



Deliverable

Functional Outcomes

- **User Interface Prototype:** A basic, functional user interface that allows users to interact with the platform. Includes a simple dashboard displaying user progress and achievements.
- **Game Environment:** An initial game environment with a few static quizzes or puzzles to demonstrate the concept. Basic interactive elements that engage users in the learning process.
- **Generative AI Integration:** A foundational AI model capable of generating simple quizzes or puzzles based on predefined criteria. Basic adaptive learning algorithms that adjust content difficulty based on user performance.

- **Project Documentation :** Architectural diagrams and design documents detailing the system's components, data flow, and integration points.
- **Data Pipeline:** Data Collection Scripts, Preprocessing Modules and Data Storage Solution
- **AI Models**
- **Evaluation Metrics & Test Results**
- **Integration and Deployment:** Deployment Scripts
- **Source Code:** All source code for the data pipeline, AI models, integration components, and user interfaces, organized in GitHub.

Learning/Reference Material

<https://ollama.com/>

<https://github.com/bobeff/open-source-engines?tab=readme-ov-file>

<https://socket.io/docs/v4/>

https://huggingface.co/models?pipeline_tag=text-generation

<https://huggingface.co/docs/transformers/index>

<https://www.tensorflow.org/tutorials/generative/dcgan>

<https://openreview.net/forum?id=Pd7lOswRUZ>

<https://engineersplanet.com/wp-content/uploads/2024/01/Application-of-Variational-AutoEncoder-VAE-Model-and-Image-Processing-Approaches-in-Game-Designsensors-23-03457-v2.pdf>

<https://www.geeksforgeeks.org/variational-autoencoders/>

<https://stable-baselines.readthedocs.io/en/master/modules/ppo2.html>

<https://spinningup.openai.com/en/latest/algorithms/dqn.html>

https://huggingface.co/docs/transformers/model_doc/t5

https://huggingface.co/docs/transformers/model_doc/bert

<https://huggingface.co/openai-community/gpt2-medium>