NOT PI VISION

NOT π

NOT SO IMPORTANT, BUT
JUST HERE TO BREAK THE CIRCLE.

Plan By: **Sabir Ali Mondal**

Problem Statement

- **Traditional study materials are static and text-heavy.**
- Students struggle with complex/abstract concepts without visuals.

- Teachers/Students lack quick, affordable tools for dynamic study.
- **No alternative to PPT study or Video editing for visualize .**

- Low-resource regions lack access to powerful devices or EdTech.

Solution Overview

- Use full potential of **Frontend** powered by **AI** .
- **Code Based Visual .**

- Syllabus/topics into visual course.
- Presentation:

    Visuals with voice narrations like Teacher .

    Book like Pages with highlighting Reader .

- Runs on any devices via browser.

GitHub : https://github.com/Sabir-Ali-Mondal/Not-Pi-Vision

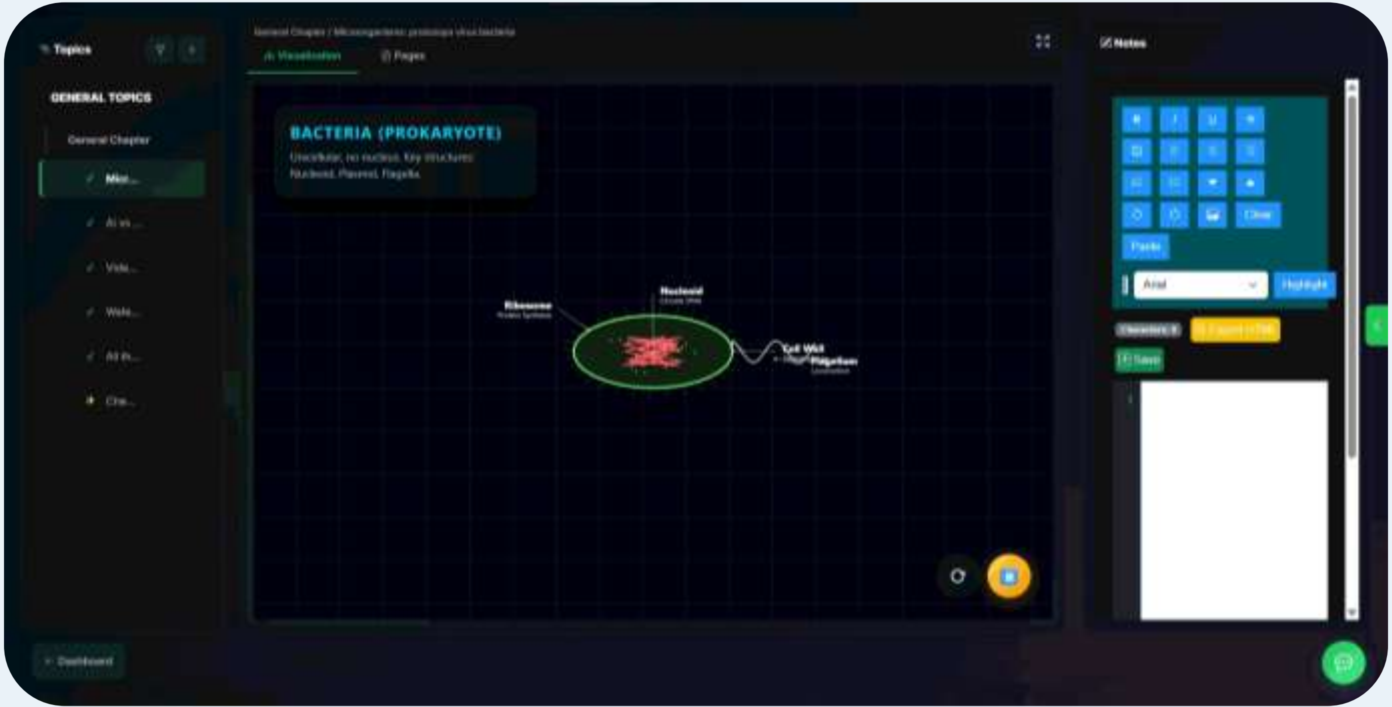
AI : Code/Text Generating AI vs Video Generation AI

Aspect	Not Pi Vision (Code)	Video Generation AI
Time to Generate	Instant (1–2 minutes)	High (30 minutes+)
Energy/Compute Use	Very Low (text output only)	Very High (GPU rendering)
Precision for Learning	**Math/Science/Graph/Interactive/Book like Diagrammable/Precised**	**Artistic, often imprecise**
Style Control	Full control (colours, lines, shapes)	Limited control (depends on model style)
Interactivity	Possible	None (static playback)
File Size	Small (few KBs of code)	Large (MBs of video files)
Post-Editing	Easy (edit code, instant change)	Hard (requires re-rendering)
Best for Academic Diagrams	Yes	No (blurry, not structured)

For study: Not Pi Vision vs Traditional AI Tools

Feature	Not Pi Vision (Code-Based Visuals)	Traditional AI Tools (Static Media)
Visual Generation	Live visuals via code. HTML/JS (fast, flexible)	Generates videos/images (slow, static)
Experience	**Dynamic 2D/3D, Interactive , Book like Diagrammable**	**Often cartoonish or ultra realistic**
Ease of Use	One-click, no heavy setup	Requires editing and re-rendering
Performance	Runs on low-end devices, browser-only	Needs high compute and strong internet
Technology	Pure frontend (HTML/JS/CSS, p5.js) Optional Backend	GPU-heavy

Visuals and Pages:



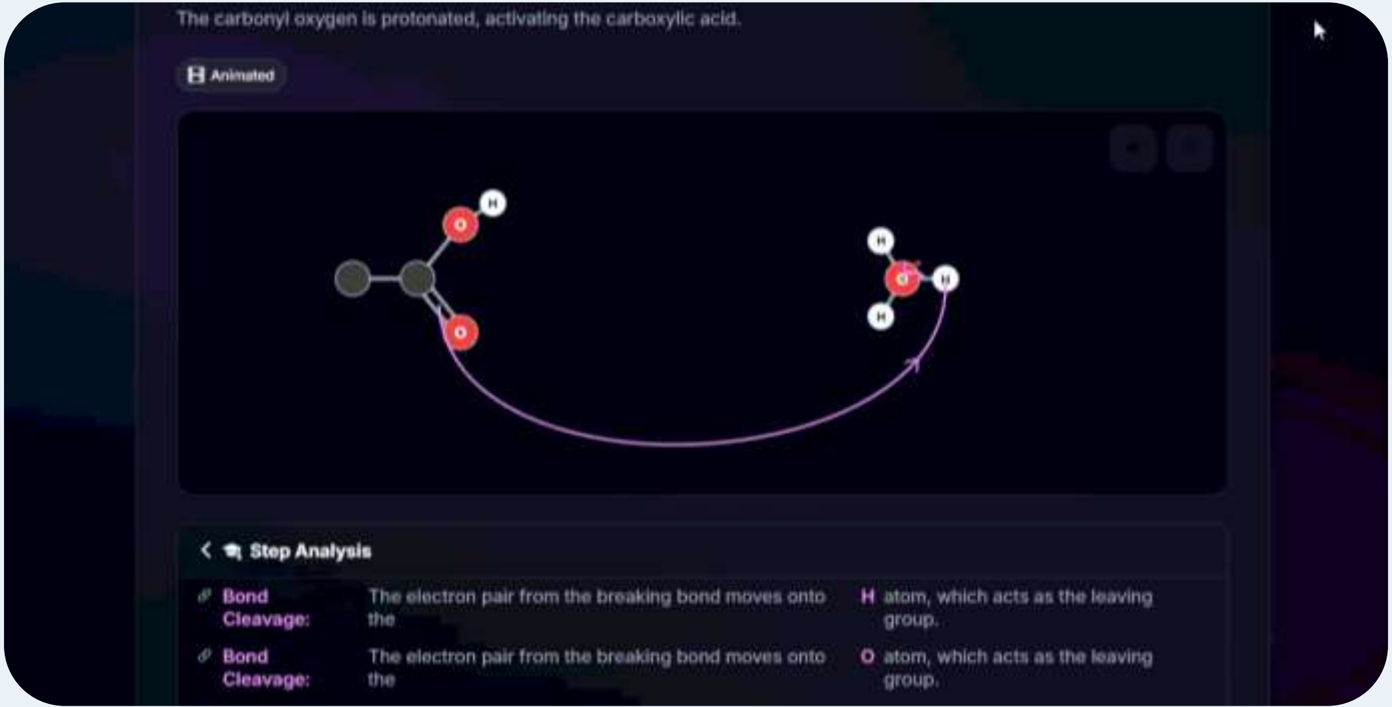
General Chapter / Microorganisms: protozoa virus bacteria

BACTERIA (PROKARYOTE)

Unicellular, no nucleus. Key structures: Nucleoid, Plasmid, Flagella.

Special Templates: (JSON powered)

Chemistry Mechanism | Graph | Mindvoice | Presentation



The carbonyl oxygen is protonated, activating the carboxylic acid.

Step Analysis

Bond Cleavage:	The electron pair from the breaking bond moves onto the	H atom, which acts as the leaving group.
Bond Cleavage:	The electron pair from the breaking bond moves onto the	O atom, which acts as the leaving group.

Features

- Standard Features (Ordinary but Essential):
  - Workspace management
  - Responsive UI
  - Offline storage (Indexed DB)
  - Reading with highlighting
- **Unique Features (Stand-out Innovations):**
  - AI syllabus deconstruction (raw text → structured Units/Chapters/Topics)
  - **Output (Visualization with podcast + Pages with reader)**
  - **Smart image linking (auto Google search suggestions)**

AI/API Integration

Prompt Generation – App creates master for AI model .

[Researched 1100 lines of prompt is ready]

- Frontend Method – Opens ChatGPT/Perplexity in a new tab with the prompt auto filled:

  Example: window.open(`https://chatgpt.com/?q=${encodeURIComponent(prompt)}`, '_blank’);

- Backend Option – Higher models can be integrated via API using a Node.js backend.

Scaling Plan

- Less failure ratio.
- **Personalized Q&A tutoring with live streaming AI.**

Already so many features are implemented and developing .

Future Vision

- Global adoption starting with India-like low-resource regions.
- Continuous evolution .
- ***ChatGPT-like chatbots can provide visualizations through frontend-powered code, making interactive visual responses accessible to non-developers directly within the chat interface.***