CAPSTONE PROJECT

AI-POWERED LATEX DIAGRAM GENERATOR FOR ACADEMIC RESEARCH

Presented By:

- Student Name- Harshit Kumar
- College Name- Mahrishi Dayanand University
- Department- CSE & AI/ML

edunet foundation

OUTLINE

- **Problem Statement**
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Result (Output Images)**
- **Conclusion**
- **Future Scope**
- **References**

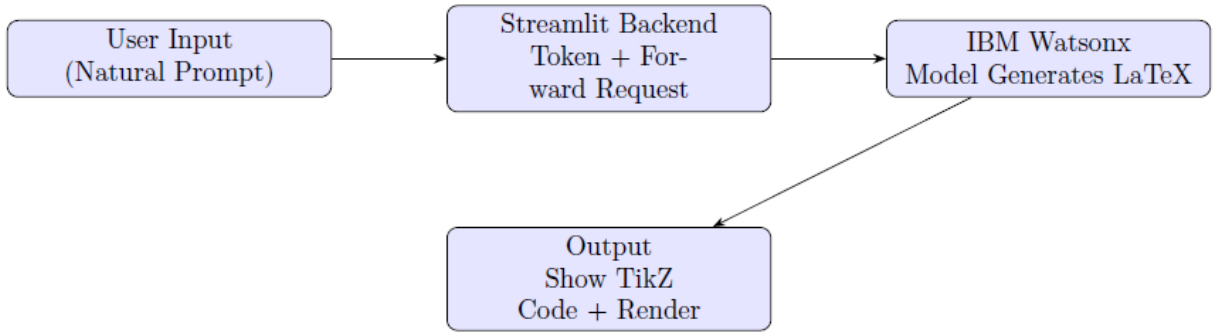
AI-POWERED LATEX DIAGRAM GENERATOR FOR ACADEMIC RESEARCH

An intelligent agent that automatically converts natural language descriptions and sketches into professional TikZ code for LaTeX documents. It provides real-time feedback, accepts plain-English refinement commands, and delivers publication-ready diagrams while eliminating manual coding complexity for researchers.

PROPOSED SOLUTION

- The proposed system aims to To build a fully online, AI-powered tool that generates LaTeX TikZ diagrams from natural language descriptions, using IBM Watsonx Foundation Models.The solution will consist of the following components:
- Data User Input (Frontend – Streamlit):
  - Token Generation: Streamlit securely fetches an IAM access token from IBM Cloud using the stored API_KEY.
  - Request Construction of prompt
- Processing (Streamlit Backend)
  - Authenticates using IBM API Key → fetches IAM token
  - Prepares a request with “prompt_variables”
  - Forwards it to IBM Watsonx Deployment
- Model Inference (Watsonx.ai)
  - Generates LaTeX TikZ code based on the input prompt.
- Output (Streamlit UI):
  - Displays TikZ code.

Example:



SYSTEM APPROACH

Frontend (Streamlit)
A lightweight UI for users to enter prompts and preview generated TikZ code & diagrams.(With Github Repo)

Backend (FastAPI or Direct Streamlit Integration) (With Github Repo)
Handles requests, fetches access tokens, and communicates with the deployed foundation model.

- **IBM Watsonx Deployment**
  - **IBM Prompt Lab: Granite model for** Code generation
  - Deployment ID & Space tied to secure API access
  - Generates LaTeX TikZ code from prompts

RESULT :PROMPT LAB

IBM watsonx | Upgrade | Harshit Kumar's Account | Dallas | HK

Projects / LaTeX TikZ Generator / LaTeX ai generator v3

AI guardrails on | Autosaved 6:34 AM | New prompt +

Chat | Structured | Freeform

Model: granite-3-8b-instruct

Input:

OSI model diagram

Output:

```
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{positioning}

\begin{document}

\begin{tikzpicture}[
   node distance = 2cm,
   every pin edge/.style = {thick, -latex},
   every node/.style = {align=center}
]

   % Layers
   \node (Physical) [rectangle, draw, fill=blue!20] {Physical};
   \node (Data Link) [rectangle, below=of Physical, draw, fill=green!20] {Data
Link};
   \node (Network) [rectangle, below=of Data Link, draw, fill=yellow!20]
{Network};
   \node (Transport) [rectangle, below=of Network, draw, fill=orange!20]
{Transport};
   \node (Session) [rectangle, below=of Transport, draw, fill=red!20] {Session};
   \node (Presentation) [rectangle, below=of Session, draw, fill=purple!20]
{Presentation};
   \node (Application) [rectangle, below=of Presentation, draw, fill=gray!20]
```

Stop reason: Stop sequence encountered
Tokens: 192 input + 355 generated = 547 out of 131072
Time: 4.4 seconds

Clear output | Generate

Model parameters

Decoding

Greedy | Sampling

Repetition penalty

1 — 2 — 1.1

Stopping criteria

Stop sequences

\end{document}

Min tokens: 0

Max tokens: 400

Enter up to 6 sequences to stop output after the minimum number of tokens is reached.

Close

edunet foundation

RESULT: RENDERING OUTPUT

Menu | Upgrade | ai generator | Review | Share | Submit | History | Layout | Chat

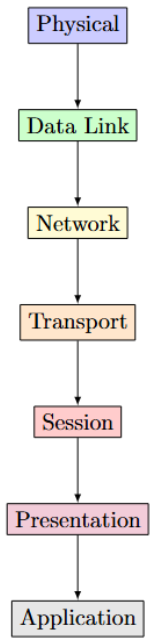
Code Editor | Visual Editor | Normal text | Recompile | 1 / 1 | 119%

main.tex

```latex
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{positioning}

\begin{document}

\begin{tikzpicture}[
    node distance = 1cm,
    every pin edge/.style = {thick, -latex},
    every node/.style = {align=center}
]

    % Layers
    \node (Physical) [rectangle, draw, fill=blue!20] {Physical};
    \node (Data Link) [rectangle, below=of Physical, draw, fill=green!20] {Data Link};
    \node (Network) [rectangle, below=of Data Link, draw, fill=yellow!20] {Network};
    \node (Transport) [rectangle, below=of Network, draw, fill=orange!20] {Transport};
    \node (Session) [rectangle, below=of Transport, draw, fill=red!20] {Session};
    \node (Presentation) [rectangle, below=of Session, draw, fill=purple!20] {Presentation};
    \node (Application) [rectangle, below=of Presentation, draw, fill=gray!20] {Application};

    % Arrows
    \draw[-latex] (Physical) -- (Data Link);
    \draw[-latex] (Data Link) -- (Network);
    \draw[-latex] (Network) -- (Transport);
    \draw[-latex] (Transport) -- (Session);
```

File outline

We can't find any sections or subsections in this file. Find out more about the file outline



RESULT : INPUT (STREAMLIT / FRONT-END)

kmbwnvunsjfzixbhjautzr.streamlit.app

Share

AI-Powered LaTeX TikZ Diagram Generator

Describe your diagram idea below:

create a butterfly cycle

Generate LaTeX Code

Contacting Watsonx.ai...

LaTeX code generated!

```
 \documentclass{article}
\usepackage{tikz}
\usetikzlibrary{positioning, decorations.markings}

\begin{document}

\begin{tikzpicture}[scale=2, every node/.style={circle, draw, minimum size=2cm}, font=\sffamily]
    % Egg
    \node (egg) [fill=green!20, label=above:{Egg}] { };
```

Manage app

RESULT :OUTPUT (STREAMLIT / FRONT-END)

kmbwnvunsjfzixbhjautzr.streamlit.app

Share

```latex
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{positioning, decorations.markings}

\begin{document}

\begin{tikzpicture}[scale=2, every node/.style={circle, draw, minimum size=2cm}, font=\sffamily]
    % Egg
    \node (egg) [fill=green!20, label=above:{Egg}] { };

    % Caterpillar
    \node (cat) [below left=of egg] [fill=yellow!20, label=above:{Caterpillar}] { };

    % Pupa
    \node (pup) [below right=of cat] [fill=brown!20, label=above:{Pupa}] { };

    % Butterfly
    \node (butterfly) [above right=of pup, fill=orange!20, label=above:{Butterfly}] { };

    % Arrows
    \draw[-latex, thick] (egg) -- (cat);
    \draw[-latex, thick] (cat) -- (pup);
    \draw[-latex, thick] (pup) -- (butterfly);
    \draw[-latex, thick, bend left=30, postaction={decorate, decoration={markings, mark=at position 0.95 with {\arrow{>}}}}] (butterfly) edge[out=90, in=-90] (eg
\end{tikzpicture}

\end{document}
```

Manage app

RESULT : RENDERING OUTPUT

Menu | Upgrade | ai generator | Review | Share | Submit | History | Layout | Chat

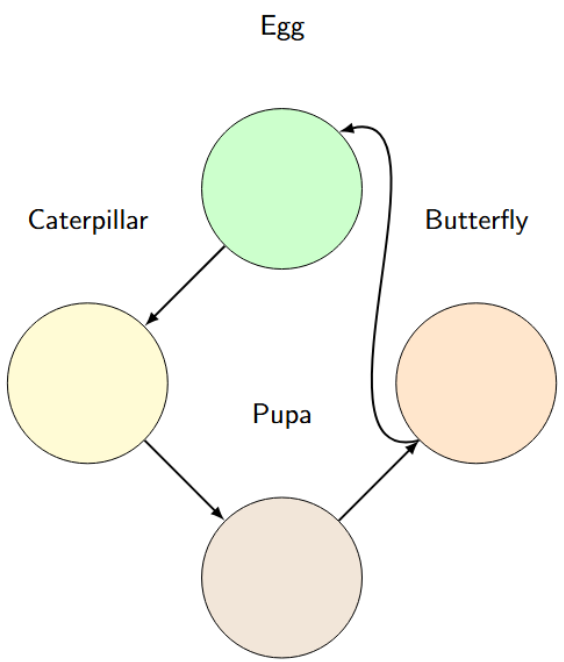
Code Editor | Visual Editor | Normal text | Recompile | 1 / 1 | 146%

main.tex

```latex
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{positioning, decorations.markings}

\begin{document}

\begin{tikzpicture}[scale=2, every node/.style={circle, draw, minimum size=2cm}, font=\sffamily]
    % Egg
    \node (egg) [fill=green!20, label=above:{Egg}] { };

    % Caterpillar
    \node (cat) [below left=of egg] [fill=yellow!20, label=above:{Caterpillar}] { };

    % Pupa
    \node (pup) [below right=of cat] [fill=brown!20, label=above:{Pupa}] { };

    % Butterfly
    \node (butterfly) [above right=of pup, fill=orange!20, label=above:{Butterfly}] { };

    % Arrows
    \draw[-latex, thick] (egg) -- (cat);
    \draw[-latex, thick] (cat) -- (pup);
    \draw[-latex, thick] (pup) -- (butterfly);
    \draw[-latex, thick, bend left=30, postaction={decorate, decoration={markings, mark=at position 0.95 with {\arrow{>}}}}] (butterfly) to[out=90, in=-90] (egg);
\end{tikzpicture}

\end{document}
```

File outline

We can't find any sections or subsections in this file. Find out more about the file outline



CONCLUSION

- The AI-Powered LaTeX TikZ Diagram Generator reliably transforms natural-language prompts into fully compile-ready TikZ code. Deployed on IBM Watsonx.ai with a simple Streamlit UI, it delivers an end-to-end, zero-install experience.

  Automating diagram creation bridges the gap between conceptual design and publication-ready graphics, saving hours of manual TikZ coding. For educators, researchers, and engineers, this tool democratizes access to high-quality technical visuals—streamlining documentation, accelerating iteration, and reducing barriers for those unfamiliar with LaTeX. In the broader AI landscape, it exemplifies how LLMs can be harnessed to automate niche, syntax-sensitive tasks with practical, tangible benefits.

  This solution streamlines the creation of technical diagrams—saving time, lowering the barrier to LaTeX, and empowering users across academia and engineering with on-demand, publication-quality visuals.

FUTURE SCOPE

- **Live Rendering & Export**: Integrate a serverless LaTeX compiler to provide in-app PDF or SVG previews, and enable one-click downloads of diagrams.

- **Preset Diagram Templates**: Offer a library of customizable, commonly used templates (flowcharts, UML, neural nets) to accelerate user workflows.

- **Multimodal Input**: Allow users to sketch rough diagrams or upload images, then have the AI refine or convert them into clean TikZ code.

- **Fine-Tuning & Domain Adaptation**: If supported, fine-tune smaller models on curated TikZ datasets to reduce syntax errors and increase styling consistency.

- **Collaborative Editing**: Add real-time sharing and annotation features so teams can co-edit diagram descriptions and code.

REFERENCES

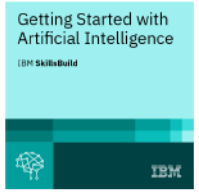
- Watsonx.ai Studio Docs:
  - https://dataplatform.cloud.ibm.com/docs/content/wsj/getting-started/welcome-main.html?context=cpdaas&audience=wdp
- Streamlit Frontend :
  - https://kmbwnvunsjfzixbhjautzr.streamlit.app/
- Overleaf for RENDERING OUTPUT :
  - https://www.overleaf.com/
- Scientific Paper Insight :
  - https://arxiv.org/pdf/2310.00367

Special thanks to Edunet Foundation

edunet foundation

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence



Harshit Kumar

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 20, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/c790b062-d9a8-40d1-aed6-0c5cc92041b8

IBM

edunet foundation

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Harshit Kumar

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 20, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/10339b0b-95a4-4332-9885-7f3809cfb177

IBM

edunet foundation

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Harshit Kumar

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 24 Jul 2025 (GMT) **Learning hours:** 20 mins

edunet foundation

THANK YOU

edunet foundation