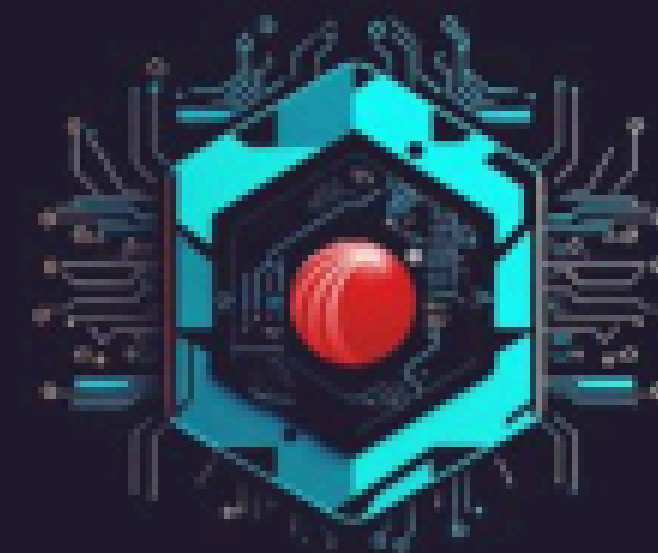


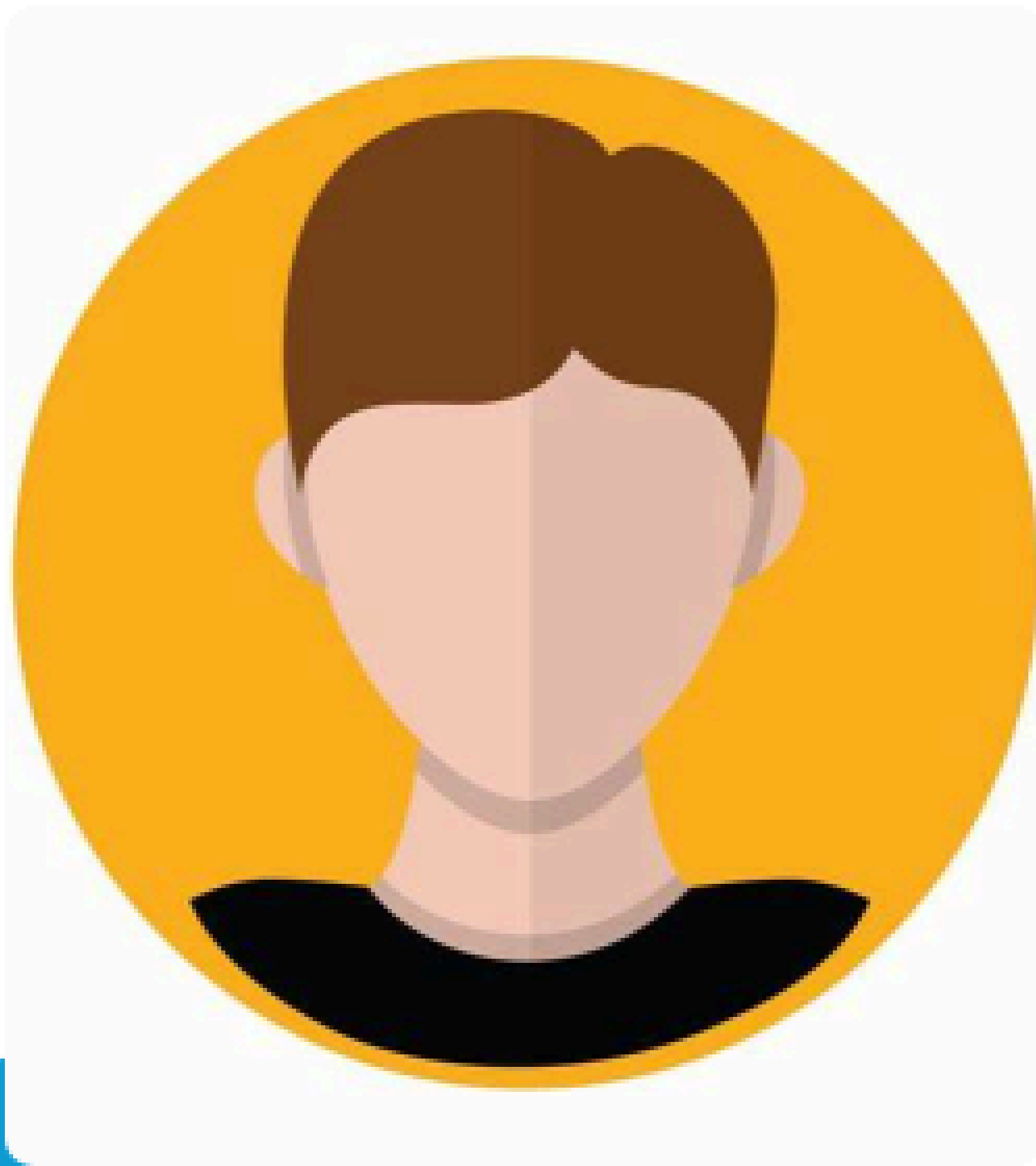
HACKATHON 2024



HACK-AI-THON
2024



RELX



ABOUT ME

Name: Ashmit Lal

Email Id: ashmitl1a110@gmail.com

Contact No: 9967443883

Github Account: Ashmit1004

Repo Link: LNRS_HACKATHON_ASHMIT

Topic: AI powered flowchart (text-to-flowchart)

College: MPSTME

Student Id: 70122100060

PROJECT OVERVIEW

- This project aims to create an automated system to generate various types of diagrams (flowcharts, sequence diagrams, mind maps, etc.) based on user input.
- It is built using Flask for the web interface, Graphviz for diagram generation, and spaCy for natural language processing (NLP).
- The system supports multiple diagram types, making it versatile for different use cases.



TECHNOLOGIES USED

FLASK

Python web framework used to create the front-end and handle user requests.

GRAPHVIZ

Used to generate diagrams like flowcharts, sequence diagrams, and mind maps.

SPACY

NLP library used for processing user input to extract conditions and actions for flowchart generation.

HTML & BOOTSTRAP

Used for designing the front-end user interface.

SYSTEM ARCHITECTURE

- User inputs diagram type and description through a web interface.
- Flask processes the request and passes the input to the diagram generation module.
- Graphviz creates a visual diagram based on the processed input.
- The generated diagram is rendered and displayed back to the user as a PNG image.

DIAGRAM TYPES SUPPORTED

Flowchart

Generated based on user conditions and actions.



UML Diagram

Represents object-oriented class structure.



Sequence Diagram

Represents the interaction between objects.



Mind Map

Visualizes a central concept with branching ideas.



Architecture Diagram

Shows components and their relationships.



Gantt Chart

Depicts tasks and project phases.



User Journey Map

Represents user actions throughout different stages of interaction.



Activity Diagram

Shows a workflow of stepwise activities and actions in a system.

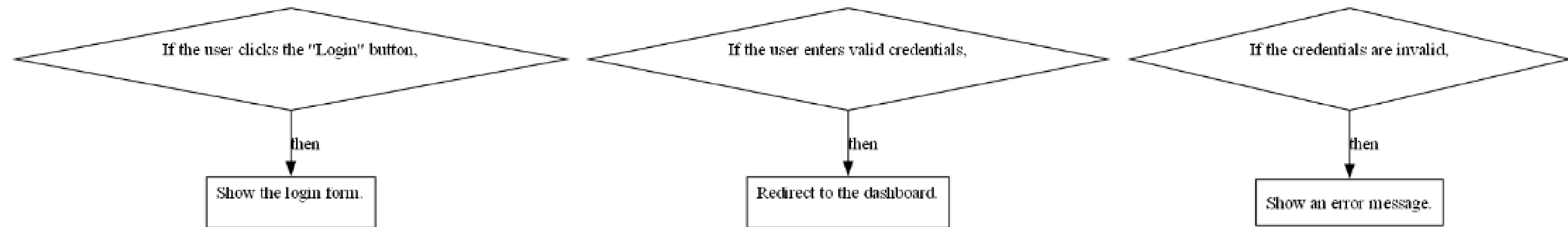


FLOWCHART EXAMPLE

Flowchart

If the user clicks the "Login" button, Show the login form. If the user enters valid credentials, Redirect to the dashboard. If the credentials are invalid, Show an error message.

Generated Diagram:



CHALLENGES AND SOLUTIONS



CHALLENGE

Parsing complex user input into diagram-friendly formats.



SOLUTION

Used spaCy to handle natural language processing for flowcharts.



CHALLENGE

Accurate diagram rendering for different types of diagrams.



SOLUTION

Graphviz provided flexible diagramming capabilities.

FUTURE IMPROVEMENTS

01

Integrate additional diagram types (e.g., ER diagrams, activity diagrams).



02

Add features like diagram customization (colors, shapes, labels).



03

Improve NLP capabilities for more complex input handling.



04

Allow users to choose specific colors for different elements in their diagrams (e.g., different colors for decision nodes, actions, participants).



Conclusion

- This project demonstrates an automated system for generating diagrams based on user input.
- It showcases how different technologies like Flask, Graphviz, and spaCy can work together to create flexible diagramming tools.
- The system can be expanded further for additional use cases in various fields such as software engineering, project management, and UX design.