A Major Proposal Project Report on

AI NEXUS: A Unified Multi-Modal Artificial Intelligence

Submitted in Partial Fulfillment of the Requirements for
the Degree of **Bachelors in Software Engineering**
Under Pokhara University

Submitted by :-
Mr. Mehfuz Alam, 211724
Mr. Subash Magar, 211735
Mr. Ashok Bhujel, 211705

Submitted to :-
NCIT

Date: 11 December, 2025

Department of Software Engineering

NEPAL COLLEGE OF INFORMATION TECHNOLOGY

Balkumari,Lalitpur, Nepal.

ABSTRACT

The rapid evolution of Artificial Intelligence (AI) has resulted in the development of multiple specialized models capable of processing text, images, audio, and performing autonomous reasoning through agentic systems. However, these AI capabilities are often fragmented across separate tools, making them difficult for students, developers, and organizations to integrate into practical applications. This project, **AI Nexus**, proposes a unified multi-modal AI platform combining text-to-image generation, image captioning, text-to-speech, speech-to-text transcription, agentic AI reasoning, tool-integrated intelligent search, and Retrieval-Augmented Generation (RAG) based question-answering into a single accessible web system.

The system uses FastAPI as the backend service layer and Streamlit + HTML/CSS/JS for a rapid, interactive, Python-native frontend interface. The platform integrates four custom modules developed in separate modules:

(1) Text-to-Image Generation using Stable Diffusion,

(2) Whisper Speech-to-Text for transcription,

(3) Image Captioning using Vision-Language models, and

(4) Agentic AI using Agno + Phidata for web search and RAG tasks.

The platform aims to demonstrate practical engineering use cases of LLMs, diffusion models, and neural speech systems, while providing a centralized, easy-to-use interface for multi-modal AI capabilities. The proposed system enables educational institutions, developers, and enterprises to experiment, deploy, and test AI workflows efficiently.

Keywords: *Artificial Intelligence, FastAPI, Streamlit, Whisper, Stable Diffusion, Agentic AI, RAG, Multi-Modal AI, Speech Processing, Image Captioning.*

Contents

ABSTRACT i

2. PROBLEM STATEMENT 1

3. PROJECT OBJECTIVES 2

4. SIGNIFICANCE OF THE STUDY 3

5. SCOPES AND LIMITATIONS 4

SCOPES 4

LIMITATIONS 4

6. LITERATURE REVIEW 5

7. PROPOSED METHODOLOGY / TECHNICAL DESCRIPTION 7

OVERALL ARCHITECTURE 7

PIPELINE METHODOLOGY 7

DEVELOPMENT METHODOLOGY 8

AGILE METHODOLOGY 8

FIGURES 9

USECASE DIAGRAM 9

8. PERFORMANCE ANALYSIS METHODOLOGY & VALIDATION SCHEME 10

PERFORMANCE EVALUATION CRITERIA 10

VALIDATION SCHEME 10

9. PROPOSED DELIVERABLES / OUTPUTS 11

10. PROJECT TASKS AND TIME SCHEDULE 12

10.1 LIST OF TABLES 13

TIME SCHEDULE (Gantt Chart) 13

WORKLOAD DISTRIBUTION 14

11. BIBLIOGRAPHY/REFERENCES 15

2. PROBLEM STATEMENT

AI services today are scattered across multiple platforms text-to-image tools, speech recognition systems, image captioning models, and agent-based AI systems all exist separately. Non-technical users struggle to integrate these tools, while developers face challenges in combining them into unified applications. There is a need for a single, accessible, multi-modal AI platform that integrates all major AI capabilities under one system.

3. PROJECT OBJECTIVES

1. Develop a unified platform integrating multiple AI services:

    1.1 Text-to-Image

    1.2 Image Captioning

    1.3 Speech-to-Text

    1.4 Text-to-Speech

    1.5 RAG-based Question Answering

    1.6 Agentic AI Workflows with tools (AI News Anchoring System)

2. Provide a fast, user-friendly interface using Streamlit + HTML/CSS/JS.

3. Build scalable backend services using FastAPI.

4. Allow real-time multi-modal data processing (image, text, and audio).

5. Demonstrate practical engineering applications of AI models.

6. Enable future extensibility for additional AI modules.

4. SIGNIFICANCE OF THE STUDY

1. Centralizes multiple AI technologies into one deployable system.

2. Useful for students, researchers, and developers experimenting with AI.

3. Reduces the complexity of handling separate AI APIs or dashboards.

4. Demonstrates integration of LLM agents with web tools (DuckDuckGo, RAG pipelines).

5. Serves as a starter platform for real-world AI products.

6. Helps institutions modernize their curriculum with practical AI engineering applications.

5. SCOPES AND LIMITATIONS

SCOPES

This project integrates multi-modal AI workflows, including text, image, and audio processing, into a unified system. It brings together four independent modules—Text-to-Image generation, Whisper Speech-to-Text, Image Captioning, and Agentic AI with tool support—into a single end-to-end pipeline. The system supports realtime processing through a web-based UI, powered by a FastAPI backend for efficient API management. Additionally, it implements RAG-based question answering using embeddings, and extends capabilities through Agentic AI connected to external search engines and tool calling, enabling intelligent, automated decision-making across modalities.

LIMITATIONS

1. Dependent on model weights and GPU availability.

2. Streamlit not suited for highly complex enterprise UI.

3. Some components may experience latency on CPU-only systems.

4. Internet dependency for web-search tool usage.

5. Not yet optimized for high concurrent user load.

6. LITERATURE REVIEW

The rapid advancement of multimodal artificial intelligence has led to the development of frameworks and models capable of integrating language, vision, and tool-based reasoning into unified workflows. LangChain provides a robust platform for building **agentic AI pipelines**, enabling large language models (LLMs) to interact with external tools and APIs programmatically, thus supporting multi-step reasoning and decision-making [1].

Vision-language models, such as CLIP, have demonstrated the ability to **learn transferable visual representations from natural language supervision**, allowing AI systems to generalize effectively across image understanding and captioning tasks [2].

Search APIs, such as the DuckDuckGo API, enable agentic systems to access external knowledge in real-time, enhancing retrieval and reasoning capabilities [3].

The Hugging Face Transformers library offers comprehensive pre-trained models for natural language processing, providing the foundation for tasks such as text classification, question answering, and embedding generation [4].

Additionally, Retrieval-Augmented Generation (RAG) techniques combine embeddings and vector search with LLMs to improve accuracy on knowledge-intensive tasks, outperforming purely generative approaches [5].

OpenAI's Whisper model has been widely adopted for **speech-to-text transcription**, demonstrating robustness in multilingual and noisy environments [6].

For web application deployment and API management, frameworks such as FastAPI provide **scalable backend services**, enabling real-time interaction between front-end interfaces and AI modules [7].

Latent Diffusion Models, as implemented in Stable Diffusion, allow **high-resolution image synthesis** by projecting complex images into a compressed latent space for efficient generation [8].

Documentation and tooling provided by Agno AI further enable developers to orchestrate agentic AI workflows and integrate external APIs seamlessly [9].

Finally, front-end frameworks like Streamlit allow rapid development of interactive Python-based applications, facilitating real-time visualization and multi-modal data processing [10].

Collectively, these works provide the foundational components for developing integrated multi-modal AI platforms that combine image, text, audio, and tool-based reasoning into unified, practical systems.

These studies collectively form the theoretical foundation for the unified multi-modal architecture implemented in AI Nexus.

7. PROPOSED METHODOLOGY / TECHNICAL DESCRIPTION

OVERALL ARCHITECTURE

1. **Frontend:** Streamlit + HTML/CSS/JS

2. **Backend:** FastAPI

3. **AI Modules:** (Overall Python)

    3.1 Stable Diffusion (Text-to-Image)

    3.2 Whisper (Speech-to-Text)

    3.3 Image Captioning Model

    3.4 Agno + Phidata (Agentic AI)

4. **Database:** Vector Store for RAG (Pinecone)

5. **External Tools:** DuckDuckGo Search, File handling

PIPELINE METHODOLOGY

1. User uploads text/image/audio.

2. FastAPI receives request and dispatches to the respective AI module.

3. Model processes the request (generate image, caption, transcript, etc.).

4. Output returned to frontend in real-time.

5. For agentic tasks:

    5.1 LLM decides if a tool is needed

    5.2 Executes tool (web search, file retrieval)

    5.3 Returns action result

6. Streamlit displays output interactively.

DEVELOPMENT METHODOLOGY

AGILE METHODOLOGY

Agile is an iterative and flexible development approach where the project is divided into small, deliverable units called sprints. Instead of completing everything at once, Agile focuses on incremental development, where each sprint delivers one fully functional AI module. Work is planned, developed, tested, and reviewed within short sprint cycles, enabling frequent testing, quick feedback, and continuous improvement. This ensures faster delivery, high adaptability, and better alignment with changing project requirements.

Why Agile?

1.Multiple independent modules

2.Continuous integration

3.Rapid prototyping

4.Ability to test each AI service as soon as it is completed

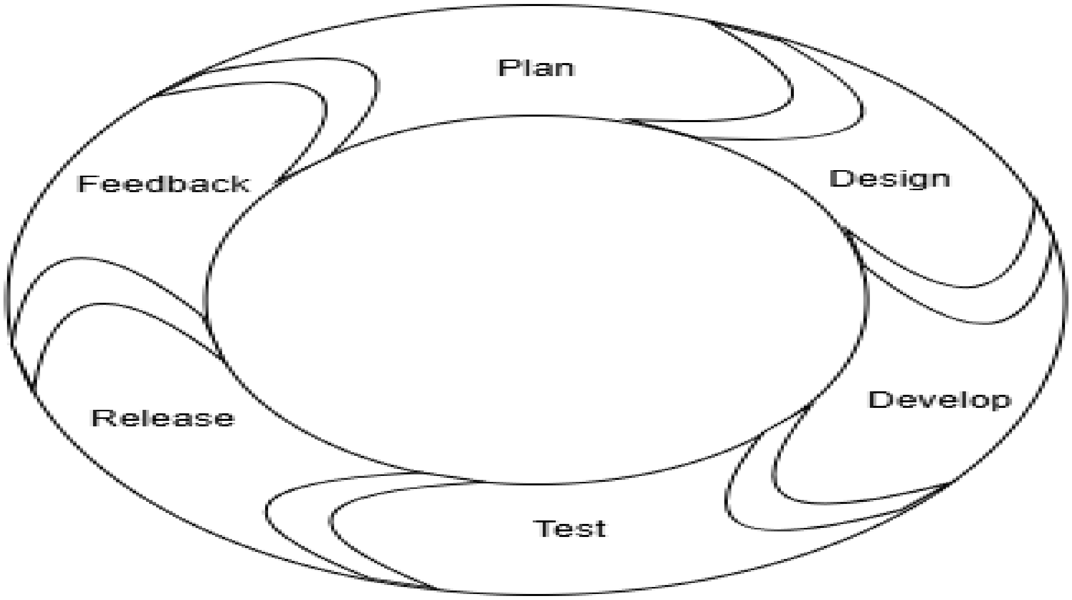

Fig: Agile Methodology

FIGURES

USECASE DIAGRAM

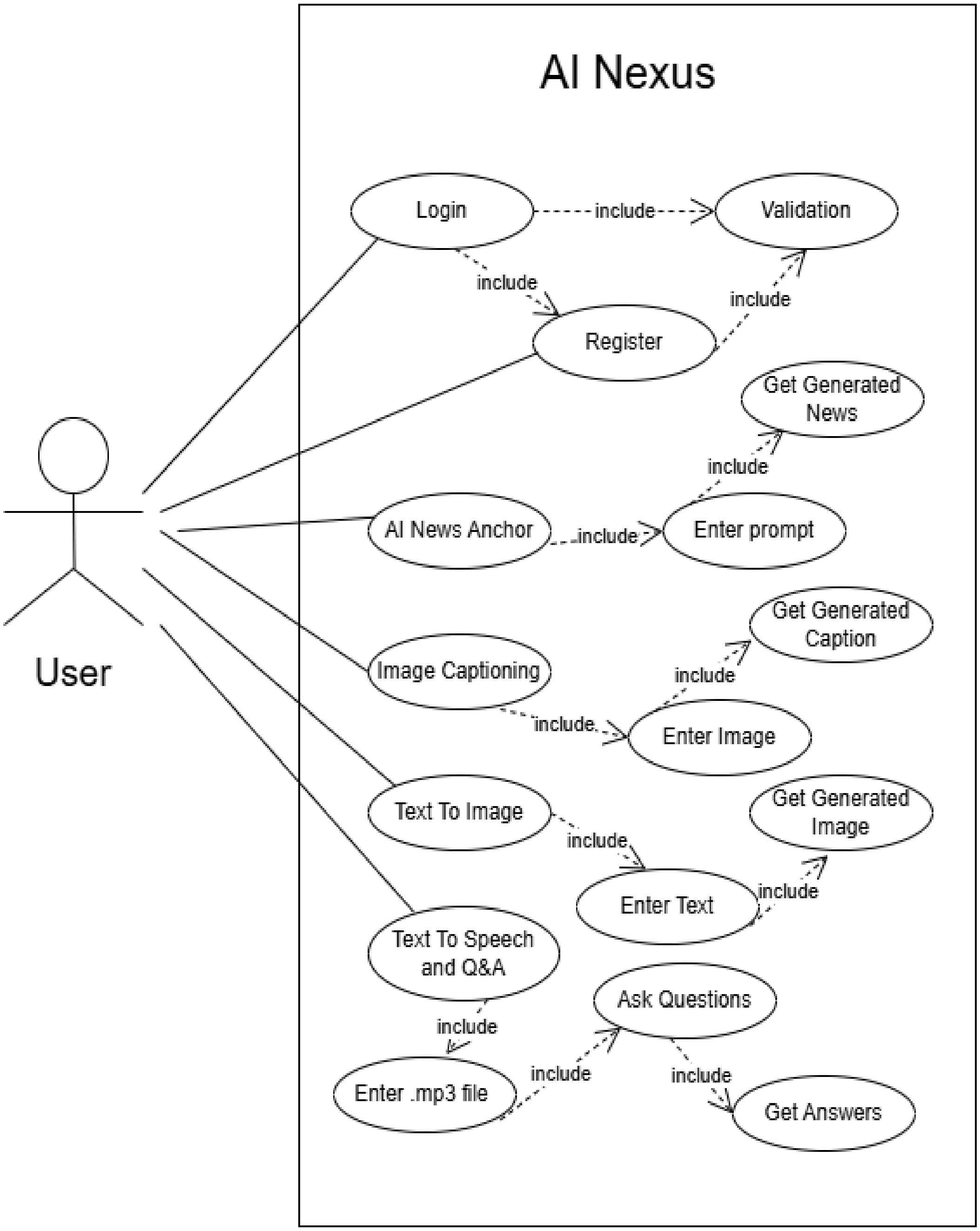

Fig : Usecase Diagram of AI Nexus

8. PERFORMANCE ANALYSIS METHODOLOGY & VALIDATION SCHEME

PERFORMANCE EVALUATION CRITERIA

1. Response Time:

   (Image generation time)

   (Speech transcription time)

   (Agent tool execution time)

2. Quality Evaluation:

   Image quality (subjective + sampling)

   Accuracy of STT transcription

   Relevance of RAG answers

VALIDATION SCHEME

1.Use test dataset for audio (Whisper benchmarking).

2.Use standard image caption validation sets.

3.Manual evaluation for agentic tasks.

4.Black-box testing for all endpoints.

5.User acceptance testing (UAT).

9. PROPOSED DELIVERABLES / OUTPUTS

1. AI Nexus Web Application

2. FastAPI Backend with all routes

3. Streamlit Frontend UI

4. **Working AI Modules:**

    4.1 Text-to-Image Generator

    4.2 Whisper STT transcriber

    4.3 Image Caption Generator

    4.4 Agentic AI with Tools

    4.5 RAG-based Q&A system

5. Complete Source Code (GitHub)

6. Technical Documentation

7. UML Diagrams

8. ER Diagram

9. Final Report + PPT

10. PROJECT TASKS AND TIME SCHEDULE

We had created a workflow based on the amount of time we had until the sixth semester is over. Thus, that gave us less than three months to work on. Our project progress is divided into designing, coding and testing phases. A final release has done after packing and finalization to the application. The project schedules have been followed as per requirements and time constraints involved in chart below. The chart does not include numerous informal conversations with the users of the system which had further aided in the development of the system.

10.1 LIST OF TABLES

TIME SCHEDULE (Gantt Chart)

S NO.	OBJECTIVES	MANGSIR				POUSH				MAGH				FALGUN			
1	Feasibility Study	▨	▨	▨													
2	Requirements			🟥	🟥	🟥	🟥	🟥	🟥								
3	Design				🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨		
4	Review And Update							🟩	🟩	🟩	🟩	🟩	🟩				
5	Programming							🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦		
6	Testing														🟪	🟪	
7	Implementation															🟧	
8	Documentation	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	

INDEX	
▨	Feasibility Study
🟥	Requirements
🟨	Design
🟩	Review And Update
🟦	Programming
🟪	Testing
🟧	Implementation

WORKLOAD DISTRIBUTION

MEMBER	**MAIN ROLE**
Mehfuz Alam	AI, Frontend, Backend, Documentation
Subash Magar	Frontend, UI/UX, Documentation
Ashok Bhujel	Backend, Documentation, Analysis

11. BIBLIOGRAPHY/REFERENCES

[1] H. C. e. al., "LangChain Documentation," LangChain Community, 10 October 2025. [Online]. Available: https://docs.langchain.com/. [Accessed 10 December 2025].

[2] A. Radford, "Whisper: Robust Speech Recognition via Large-Scale Weak Supervision," OpenAI, San Francisco, USA, 2022.

[3] I. DuckDuckGo, "DuckDuckGo Search API – Technical Documentation," DuckDuckGo, Inc., 9 March 2023. [Online]. Available: https://duckduckgo.com/api. [Accessed 12 January 2025].

[4] I. Hugging Face, "Transformers Documentation," Hugging Face, Inc., 20 June 2024. [Online]. Available: https://huggingface.co/docs/transformers. [Accessed 12 January 2025].

[5] E. P. e. a. Patrick Lewis, "Retrieval-Augmented Generation for KnowledgeIntensive NLP," NeurIPS / Meta AI, Vancouver, Canada, 2020.

[6] R. Rombach, "High-Resolution Image Synthesis with Latent Diffusion Models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, USA, 2022.

[7] S. Ramírez, "FastAPI Documentation," FastAPI Project / Sebastián Ramírez, 1 September 2024. [Online]. Available: https://fastapi.tiangolo.com/. [Accessed 12 January 2025].

[8] J. W. K. e. a. Alec Radford, "Learning Transferable Visual Models From Natural Language Supervision," PMLR, Virtual Conference, 2021.

[9] P. Team, "Agno AI Documentation," Agno AI, 15 August 2024. [Online]. Available: https://docs.agno.ai/. [Accessed 12 January 2025].

[10] S. I. (. team), "Streamlit Documentation," Snowflake Inc. (Streamlit), 5 October 2024. [Online]. Available: https://docs.streamlit.io/. [Accessed 12 January 2025].