# Paperwise

Submitted in partial fulfillment of the requirements of the degree

**BACHELOR OF ENGINEERING** IN**COMPUTER ENGINEERING**

By

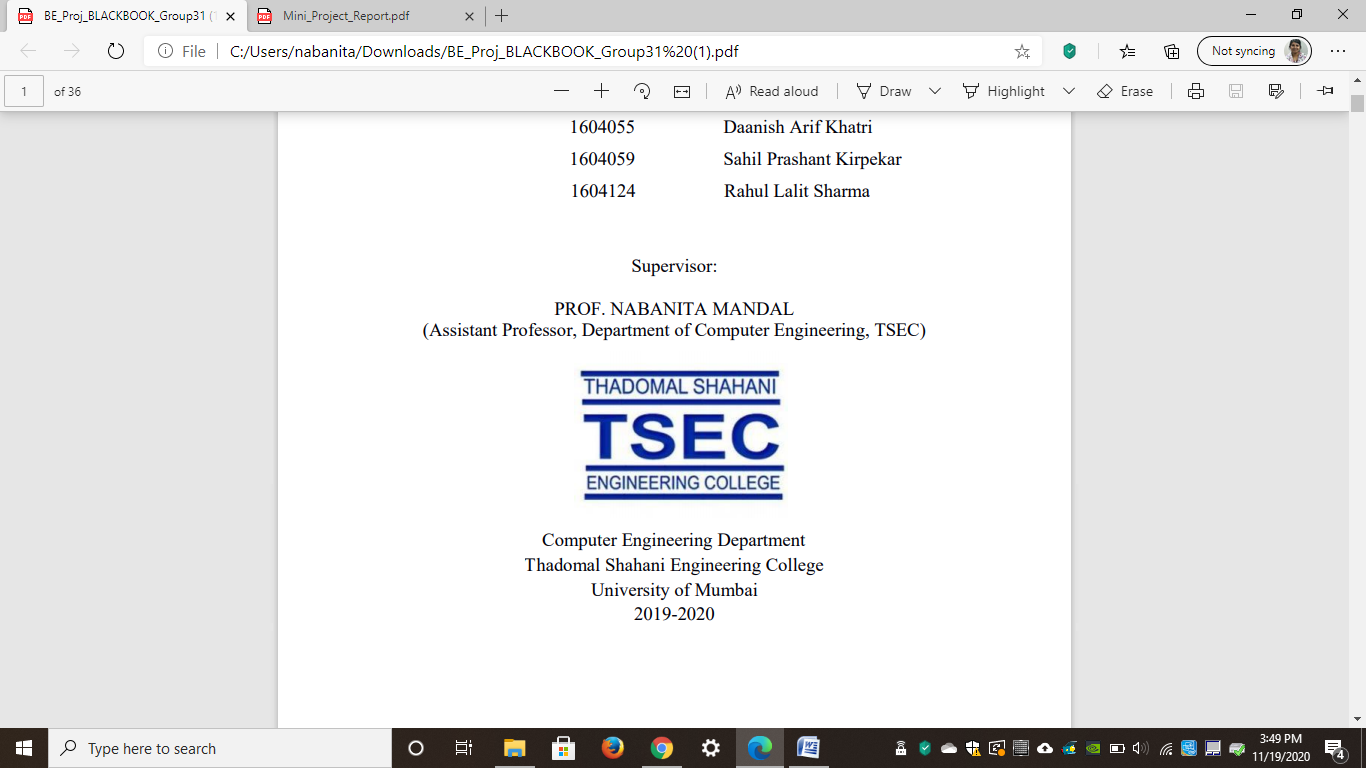
**Jatin Sachdev 2303145**

**Saina Sachdeva 2303146**

**Sagar Sawlani 2303157**

Supervisor

Prof. Jayant Gadge



**Computer Engineering Department**

**Thadomal Shahani Engineering College**

**Bandra(W), Mumbai - 400 050**

**University of Mumbai**

**(AY 2025-26)**

# CERTIFICATE

This is to certify that the Mini Project entitled “**Paperwise”** is a bonafide work of **Sagar Sawlani 2303157 , Saina Sachdeva 2303146 and Jatin Sachdev 2303145** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **“Bachelor of Engineering”** in **“Computer Engineering”.**

### Prof. Jayant Gadge

**Supervisor**

### Dr. Tanuja Sarode Dr. G. T. Thampi

**(Head of Department) (Principal)**

# Mini Project Approval

## This Mini Project entitled “Paperwise**”**

By **Sagar Sawlani 2303157 , Saina Sachdeva 2303146 and Jatin Sachdev 2303145** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

**Examiners**

**1………………………………………**

(Internal Examiner Name & Sign)

### 2…………………………………………

(External Examiner name & Sign)

Date: Place:

# Contents

### Abstract ii

### Acknowledgments iii

### List of Abbreviations iv

### List of Figures v

### List of Tables vi

### List of Symbols vii

### Introduction 1

* 1. Introduction
  2. Motivation
  3. Problem Statement & Objectives
  4. Organization of the Report

### Literature Survey 11

* 1. Survey of Existing System
  2. Limitation Existing system or research gap
  3. Mini Project Contribution

### Proposed System 18

* 1. Introduction
  2. Architecture/ Framework
  3. Algorithm and Process Design
  4. Details of Hardware & Software
  5. Experiment and Results
  6. Conclusion and Future work.

### References 32

# Acknowledgement

We would like to express our gratitude and thanks to **Prof. Jayant Gadge** for her valuable guidance and help. We are indebted for her guidance and constant supervision as well as for providing necessary information regarding the project. We would like to express our greatest appreciation to our principal **Dr. G.T. Thampi** and head of the department **Dr. Tanuja Sarode** for their encouragement and tremendous support. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of theproject.

Sagar Sawlani

Saina Sachdeva

Jatin Sachdev

# Chapter 1

**Introduction**

This chapter explains the aim, objectives and scope of the proposed system.

## **Introduction**

The digital age has led to an explosion in the volume of documents, articles, and transcripts that professionals must process, resulting in severe **information overload**. Professionals often spend significant time manually reading lengthy texts to extract crucial information, a process that is often time-consuming and inefficient. **PAPERWISE** is a web-based, AI-powered solution developed to streamline this process, enabling users to quickly interact with their documents and extract **semantically accurate information**.

## **Motivation**

The primary motivation behind this project is the lack of effective tools for deep contextual understanding of long documents. Existing summarization tools frequently fail to retain the necessary context or struggle with complex document formatting , which hinders the ability to answer follow-up queries accurately. There is a clear need for an intelligent system that not only extracts key points but also organizes content and provides **interactive, context-aware querying**. Our goal is to develop an advanced system that provides high-accuracy, relevant responses by relying *only* on the uploaded document’s content, thereby addressing the limitations of existing AI tools.

## **Problem Statement & Objectives**

**Problem Statement:** Professionals face information overload and inefficiencies when manually reading and extracting insights from lengthy documents, as traditional summarization tools lack context retention and interactive Q&A capabilities .

**Objectives:** The development of **PAPERWISE** is guided by the following objectives:

1. To develop automatic document summarization with **section-wise breakdowns**.
2. To enable **contextual Q&A** through a chat-based interface.
3. To implement **Retrieval-Augmented Generation (RAG)** for accurate and relevant responses.
4. To handle long documents effectively using **chunking and embeddings**.
5. To support **multi-document comparison** for deeper analysis.
   1. **Organization of the Report**

This report consists of three chapters. The first chapter deals with introduction of the topic, problem statement, motivation behind the topic and objectives. The second chapter is the Literature Survey. It includes all the research work done related to this topic. All information related to study of existing systems as well as learning of new tools is mentioned in this chapter. The third chapter is about the proposed system which is used in this project. The block diagram, techniques used, hardware and software used screenshots of the project are presented in this chapter. All the documents related to development of this project are mentioned in References

**Chapter 2**

**Literature Survey**

This chapter explains the concepts used in this project, study of existing system and contribution of this project

**2.1 Survey of Existing System**

The existing landscape of document analysis tools includes several systems with similar features:

* **NotebookLM:** This system breaks down text into tokens using a transformer architecture, which uses attention mechanisms to convert them into vector embeddings that help carry meaning.
* **Azure AI:** This tool uses natural language processing on extracted data to analyze text, with a main focus on summarization.

## **2.2 Limitation of existing system**

While studying the existing systems, several drawbacks were identified that represent a research gap:

* **Context Loss:** Traditional summarization tools often lose crucial context and cannot answer follow-up queries based on the original document.
* **Handling Long Documents:** Generic tools struggle with long documents due to input size limitations, making comprehensive analysis difficult.
* **Hallucination Risk:** Systems not grounded in the document’s content risk generating inaccurate or irrelevant information.
* **Lack of Interactive Context:** Existing systems often fail to provide a seamless, chat-based interface for interactive, contextual question-answering.

## **2.3 Mini Project Contribution**

**PAPERWISE** contributes to the field by addressing the limitations of existing systems:

* **Core Architecture:** The implementation of **Retrieval-Augmented Generation (RAG)** is the primary differentiator, ensuring that the final output is generated based *only* on the provided context.
* **Long Document Capability:** The system incorporates **chunking and embeddings** to effectively handle documents that exceed typical token limits.
* **Semantic Search:** By using vector storage and semantic search , the system retrieves the most semantically relevant passages, which is superior to simple keyword matching.
* **Analytical Features:** The project includes the functionality for **section-wise breakdown** and plans to support **multi-document comparison**.

# Chapter 3

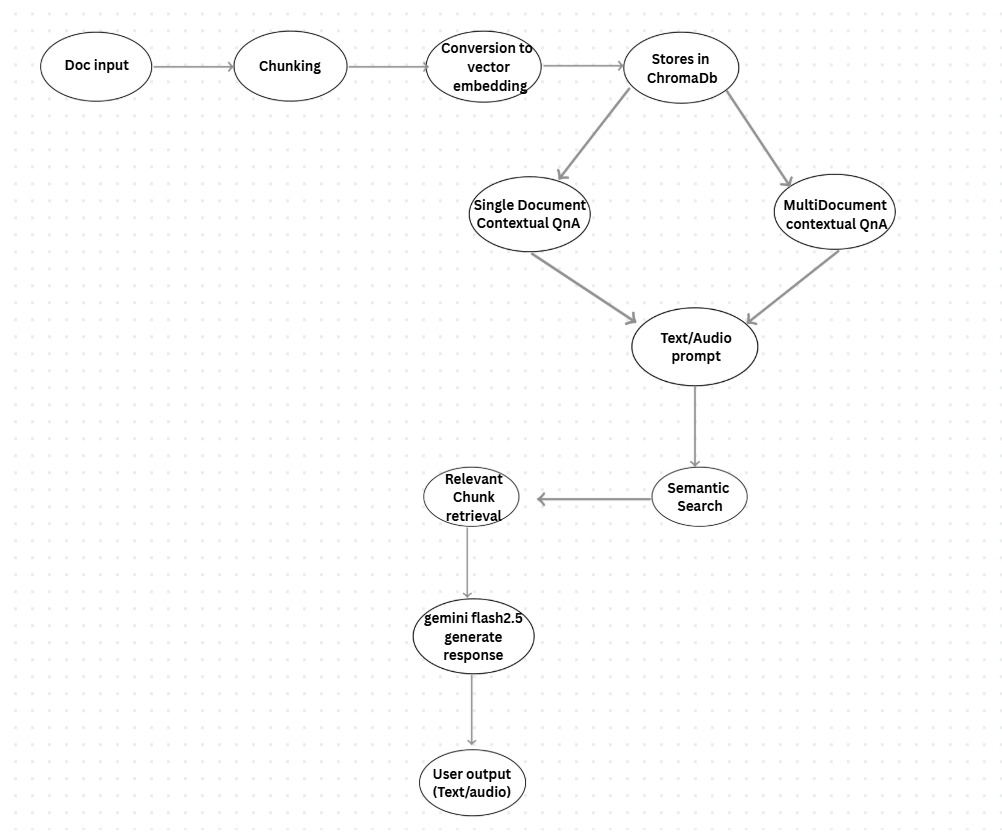
**Proposed System**

This chapter consists of detailed description about the methodology used, the hardware and software components, the tools used and also the screenshots of the project

**3.1 Introduction**

The **PAPERWISE** system is an AI-powered, web-based application built upon the **Retrieval-Augmented Generation (RAG)** framework. The primary technologies used include a **Large Language Model (LLM)**, an **Embedding Model**, and a **Vector Database**. The system is designed to parse and process various document formats (e.g., PDF, DOCX) using a selected programming language, tools, and technology.

## **3.2 Architecture**



The flow begins with the **User Uploads Document** component. The document undergoes **Preprocessing** , leading to **Chunking & Embedding**. These embeddings are instrumental in the RAG. The preprocessed data feeds into two main functions: **Summarization** (yielding **Automatic Summary** and **Section-wise breakdown** ) and **Contextual Q & A**. The Q&A component leverages **RAG For Context** and is presented through a **Chat Based Interface for Questions** , resulting in **Context Aware Answers** as part of the **User Output**.

## **3.3 Algorithm and Process Design**

The system development involved the following steps:

1. **Formulating the Problem statement:** We established the goal to solve the information overload problem by creating an AI tool for automatic summarization and contextual Q&A.
2. **Understanding the framework and requirements:** We finalized the **RAG** architecture as the core framework to ensure high accuracy and context-aware generation.
3. **Identifying tools/technology to be used:** We selected an appropriate **LLM**, an **Embedding Model**, and a **Vector Database** (e.g., ChromaDB or Pinecone) for handling high-dimensional vector storage.
4. **Finalizing the features to be included:** Key features were locked in: section-wise summarization, chat-based Q&A, chunking for long documents, and multi-document comparison .
5. **Development:** We implemented the RAG pipeline, focusing on document parsing, text chunking , vectorization , storage , semantic retrieval , and context-augmented generation.
6. **Testing:** The system was tested extensively with large documents to verify chunking efficiency and Q&A accuracy based *only* on the document context.
7. **Evaluation:** The performance was evaluated based on the speed of document ingestion and the semantic relevance of the generated summaries and query responses.

**3.4 Details of Hardware & Software**

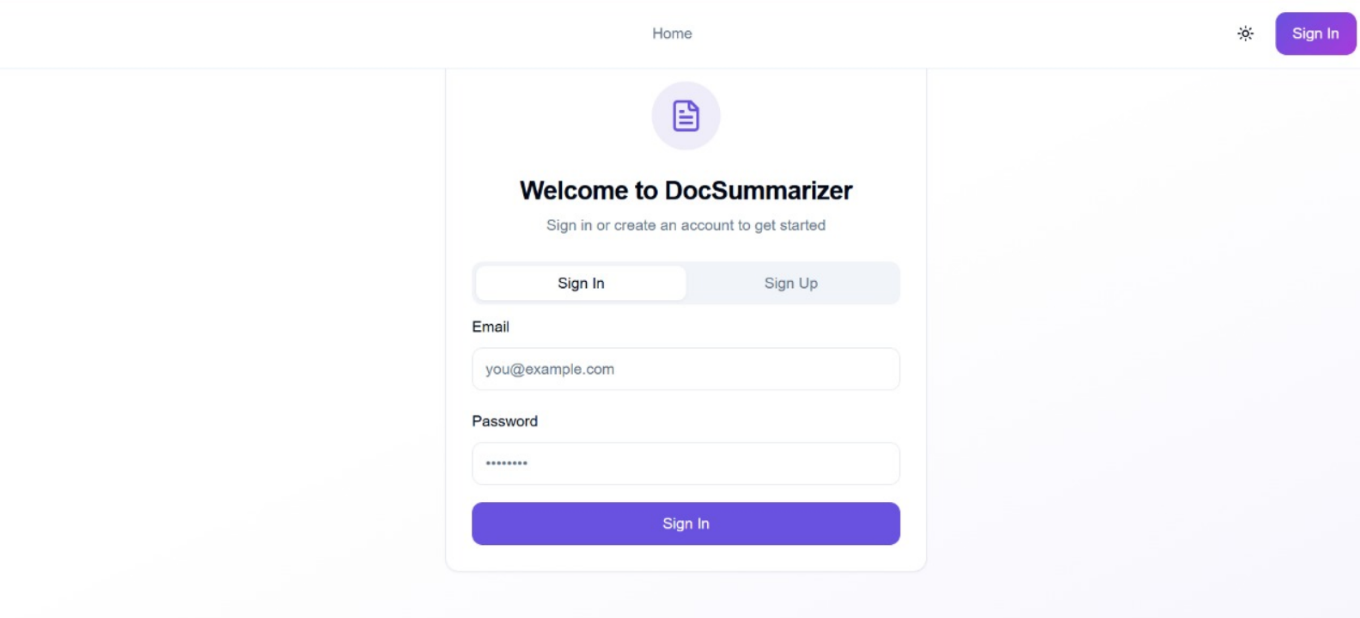
The **PAPERWISE** system requires a standard computing environment for its operation and development.

**Hardware Configuration:** The system can be run on a personal computer with at least an **Intel Core i5 (or equivalent)** Processor and **8 GB or higher** of RAM. The operating system used for development and testing is **Windows 10/11 or macOS/Linux**.

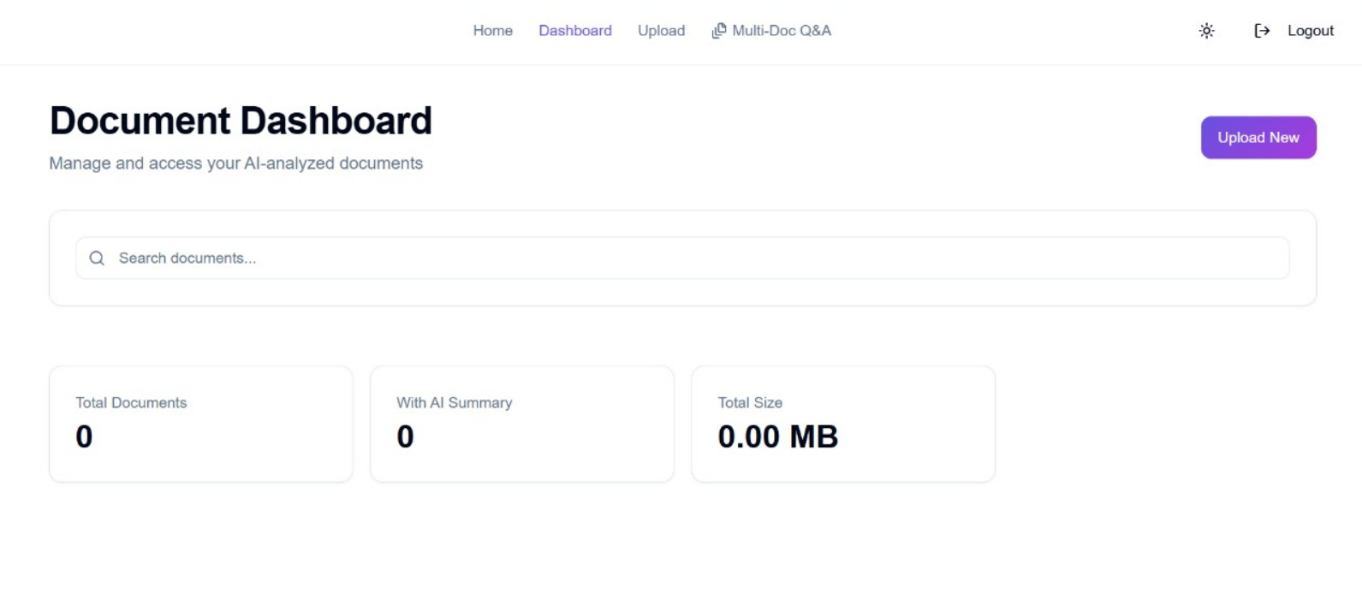
**Software Configuration:** The core architecture of the system is **Retrieval-Augmented Generation (RAG)**. The project utilizes a **Large Language Model (LLM)**, such as Gemini 2.5 Flash, for context-augmented generation. The vectorization of text chunks is performed by an **Embedding Model**, and the resulting high-dimensional numerical vectors are stored in a **Vector Database**, such as ChromaDB . The primary **Programming Language** used is Python, likely leveraging libraries like LangChain or FastAPI for the RAG pipeline. The user interface is developed using a **Web Framework** as ReactJS

**3.5 Results**

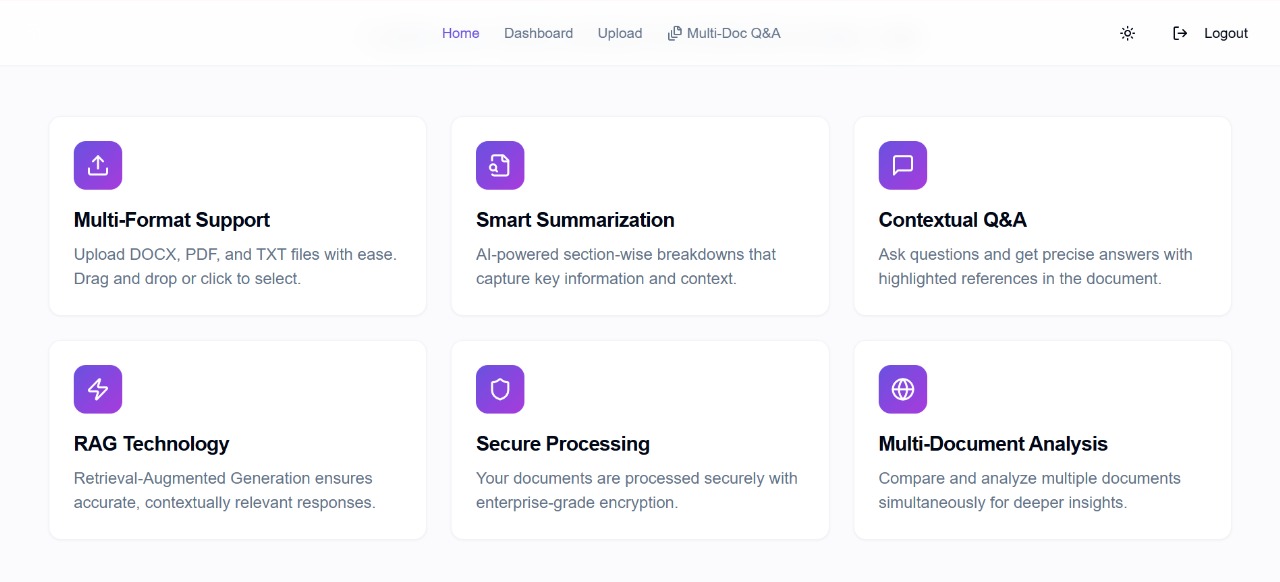
1. Authentication page:-



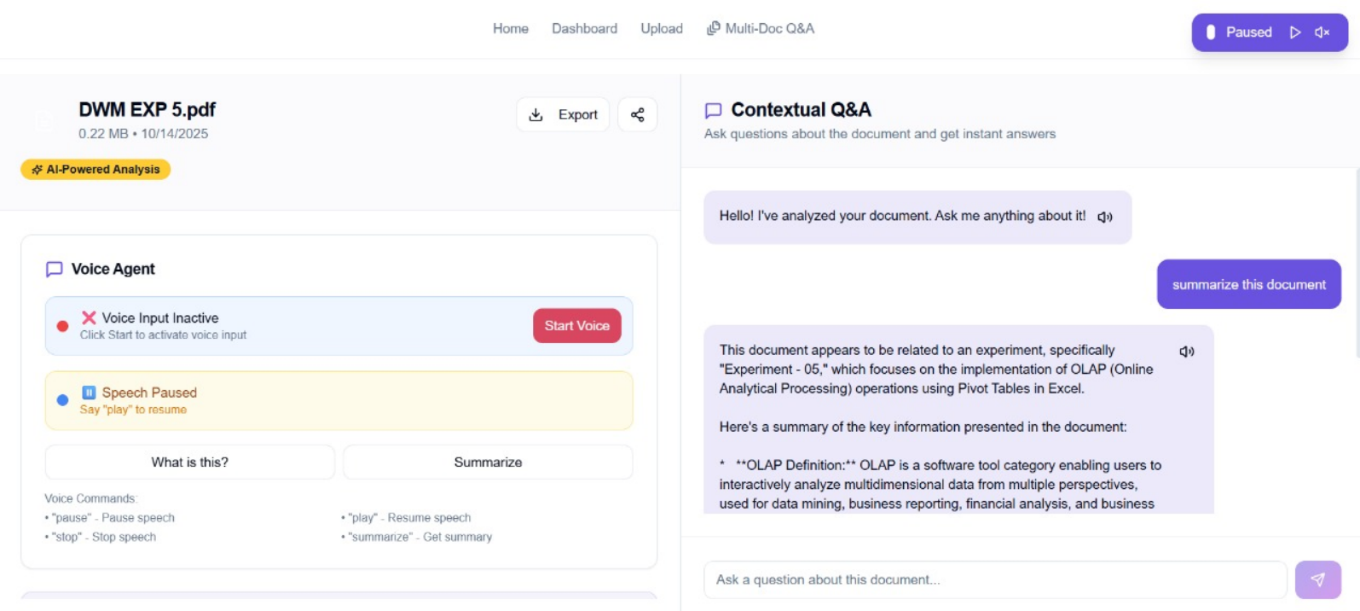
2. Dashboard:-



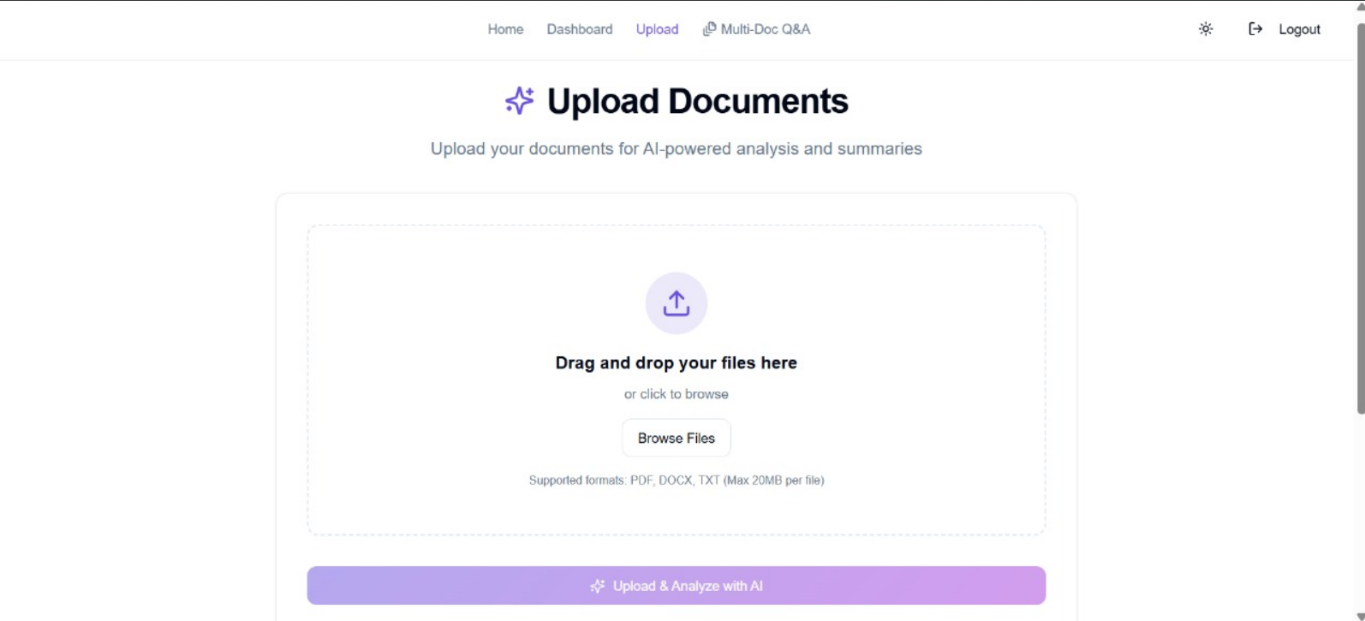
3. Home Page:-



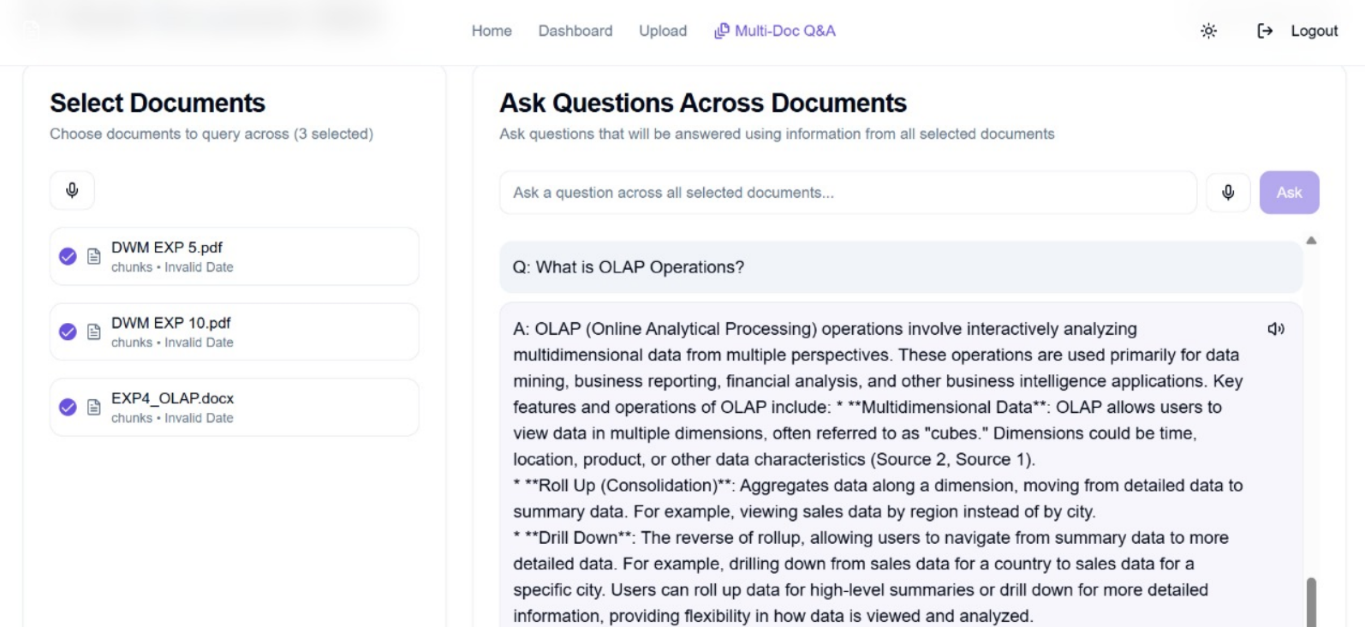
4. Contextual Q&A Page:-



5. Upload Page:-



1. Multi-Doc Q&A Page:-



**3.6 Conclusion and Future Work**

The **PAPERWISE** project successfully developed an AI-powered web-based tool that addresses the challenges of information overload and inefficient document analysis by leveraging the **Retrieval-Augmented Generation (RAG)** architecture. The system is capable of automatically generating summaries and section-wise breakdowns while enabling high-accuracy, contextual Q&A through its chat interface. By implementing chunking and vector embeddings, the tool efficiently handles long documents, making it a valuable asset for professionals across various sectors who need to extract relevant insights quickly and reliably.

**Future Work** Potential future enhancements for **PAPERWISE** include:

* Integrating a more sophisticated **Decision making Agent** powered by vector embeddings to provide deeper analytical recommendations or clustering of key topics.
* Expanding support for a wider variety of file types and complex document formats, such as structured data within documents.
* Implementing user authentication and role-based access for robust enterprise deployment.

**References**

### [1] M. B. C. M. A. A., "Vector Embeddings: A Comprehensive Guide for Developers," *Pinecone*, 2023. [Online]. Available: <https://www.pinecone.io/learn/vector-embeddings-for-developers/>

### [2] D. D. J. P. T., "What is a Vector Database?," *Pinecone*, 2023. [Online]. Available: <https://www.pinecone.io/learn/vector-database/>

### [3] J. Weng, "Generative AI Basics," *YouTube*, 14 Nov. 2023. [Video]. Available: [https://youtu.be/d4yCWBGFCEs?si=gvM1A0HsRKuoJLU](https://www.google.com/search?q=https://youtu.be/d4yCWBGFCEs%3Fsi%3DgvM1A0HsRKuoJLU)\_

### [4] S. K. A., "LLMs: Large Language Models," *YouTube*, 20 Sep. 2023. [Video]. Available: <https://youtu.be/7xTGNNLPyMI?si=EEqni_1VPP0W2qvF>