AI PPT GENERATOR WEBAPP

SYNOPSIS OF PROJECT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY

IN

Computer Science & Artificial Intelligence

SUBMITTED BY
Amritpal Singh
12201104
B49A

SUPERVISED BY

Dr. Ridhi Kapoor

DAV University, Village Sarmastpur
Jalandhar - Pathankot National Highway (NH-44)
Jalandhar - 144 001, Punjab

TABLE OF CONTENTS

Chapter 1 – Introduction 1
1.1 Purpose 1
1.2 Scope 1
1.3 Project Objective 2
1.4 Definitions, Acronyms, and Abbreviations 2
1.5 Purpose of the Document 3
1.6 Significance of the Project 3
Chapter 2 – Overall Description 4
2.1 Product Perspective 4
2.2 Product Features 4
2.3 User Classes and Characteristics 4
2.4 Operating Environment 5
2.5 Assumptions and Dependencies 5
Chapter 3 – Functional Requirements 6
3.1 Introduction 6
3.2 User Authentication Requirements 6
3.3 Topic Input & Outline Generation 6
3.4 Slide Content Generation 6
3.5 Image Generation Requirements 6
3.6 Slide Rendering & Editing 7
3.7 Project Management Requirements 7
3.8 System Integration Requirements 7
3.9 Summary 7
Chapter 4 – Non-Functional Requirements 8
4.1 Introduction 8
4.2 Performance Requirements 8
4.3 Security Requirements 8
4.4 Usability Requirements 8
4.5 Reliability Requirements 9
4.6 Availability Requirements 9
4.7 Scalability Requirements 9
4.8 Summary 9
5. Chapter 5 – System Models 10
5.1 Introduction 10
5.2 Context Level Diagram (Level 0 DFD) 10
5.3 DFD Level 1 Diagram 10
5.4 Use Case Diagram 11
5.5 Sequence Diagram & Activity Diagram 11
5.6 Summary 11

CHAPTER 1

INTRODUCTION

1.1 Purpose

The rapid growth of digital communication—especially in academic, corporate, and professional environments—has significantly increased the demand for high-quality presentations. Whether delivering a lecture, pitching a startup idea, presenting research findings, or explaining a technical workflow, presentations have become the primary medium for conveying structured information.

However, creating a visually appealing and well-organized presentation requires time, creativity, and design experience. Most users, particularly students and non-design professionals, often struggle to develop meaningful content, maintain consistent layouts, apply proper formatting, select relevant images, and ensure that the slides appear professional.

The aim of this project is to develop an AI-Based Automated Presentation Generator that allows users to produce complete presentations instantly by simply providing a topic. Using Gemini AI, the system automatically generates an outline, expands it into slide-level content, designs structured layouts using TailwindCSS templates, and integrates AI-generated images from ImageKit. The system eliminates repetitive manual work and reduces presentation creation time from hours to minutes.

Thus, the overall purpose of this project is to deliver a fully automated, user-friendly, scalable, and intelligent presentation creation platform accessible to both technical and non-technical users.

1.2 Scope

This project focuses on building a web-based automated PPT generation system powered by Gemini AI, Firebase, Clerk authentication, React.js, TailwindCSS, and ImageKit.io. The scope includes:

- AI-powered outline generation

- Slide-level text generation

- Automated layout selection & styling

- AI-based image creation and insertion

- Real-time editing interface for user modifications

- User authentication and premium features using Clerk

- Firebase Firestore for cloud-based project storage

- Workspace for managing multiple projects

- Scalable architecture capable of supporting large presentations

The system is designed to assist students, educators, professionals, content creators, and business presenters. The scope also includes subscription logic—free users have limited project access, while premium users enjoy full features and unlimited generation.

This project does not include exporting slides directly as PPT files, collaboration features, or voice-to-PPT input.

1.3 Project Objective

The core objective of the project is to automate the entire PPT creation process using AI.

Primary Objectives:

- Automatically generate a detailed outline from a user-given topic

- Convert the outline into complete slides with titles and bullet points

- Design slide layouts using pre-built templates

- Generate images automatically based on slide content

- Store and manage user projects in Firebase

- Allow real-time editing with an interactive UI

- Provide secure login, session management, and premium feature access

- Maintain high system scalability and fast performance

Secondary Objectives:

- Reduce presentation creation time by 90%

- Improve slide consistency and visual quality

- Ensure accessibility for non-designers

- Support cloud-based storage and accessibility

1.4 Definitions, Acronyms, and Abbreviations

AI – Artificial Intelligence

Gemini AI – Google's multimodal AI model used for text generation

Firestore – Cloud NoSQL database by Firebase

Clerk – Authentication and subscription management service

DFD – Data Flow Diagram

UML – Unified Modeling Language

React – JavaScript UI library

TailwindCSS – Utility-first CSS framework

API – Application Programming Interface

ImageKit – AI-powered image generation and processing service

1.5 Purpose of the Document

This synopsis provides a structured academic-level description of the AI-Based Automated Presentation Generator system. It outlines the project's goal, scope, modules involved, and the specific requirements for implementation.The document acts as a formal declaration of:

- The problem addressed
- The proposed methodology
- The system's architecture
- The functional and non-functional requirements
- The expected outcomes

This document is intended for faculty evaluators, project supervisors, and development teams.

1.6 Significance of the Project

The significance of this project lies in transforming how presentations are created. Traditional PPT tools require manual content writing, layout designing, and image searching. This system automated those processes, providing:

Educational Benefits:

- Faster lecture and study material creation

- Easy-to-use for students and teachers

Professional Benefits:

- Quick business pitch creation

- Time savings for corporate use

CHAPTER 2

OVERALL DESCRIPTION

2.1 Product Perspective

The AI-Based Automated Presentation Generator is a self-contained web application designed to automate the process of creating presentation slides. It sits between the end-user and several external services: Gemini AI for text generation, ImageKit for image generation, Clerk for user authentication, and Firebase Firestore for data storage. From the user's perspective, the product behaves like a modern web application: a responsive single-page interface built in React that communicates with cloud services to generate and persist content. Architecturally, the application follows a client–cloud model where the frontend handles user interactions and rendering, while AI calls and persistent storage are managed by cloud services.

2.2 Product Features

The primary features offered by the product include:

- Topic-to-outline conversion: Generate a complete, structured presentation outline rom as hort user topic.
- Slide content generation: Expand outline items into slide-level titles and bullet points using Gemini AI.
- Automated layout and styling: Produce HTML + TailwindCSS templates for each slide to ensure visual consistency.
- AI-based image generation: Create relevant visuals for slides via ImageKit and insert optimized images automatically.
- Real-time editing: Provide an interactive slide editor with instant preview and auto-save to Firestore.
- Project workspace: Manage multiple projects, open previous work, and continue editing.
- Authentication and subscription control: Secure login via Clerk with differentiation between free and premium users.

2.3 User Classes and Characteristics

The system targets the following user classes:

• Students: Need rapid generation of presentation material for assignments and seminars; prefer simple input and quick output.

• Educators: Require lecture slides and teaching material generation with editable templates.

• Professionals: Prepare pitch decks and business presentations that emphasize clarity and visual appeal.

• Content Creators and Researchers: Use the system to summarize documents and produce shareable slide decks.

All user classes are assumed to have basic internet literacy and access to a modern web browser.

2.4 Operating Environment

The application is designed to run in mainstream web environments. The frontend supports modern browsers (Google Chrome, Mozilla Firefox, Microsoft Edge) on desktop and mobile platforms. Development and deployment tools include Node.js and Vercel or Firebase Hosting. The system depends on external cloud services: Gemini API and ImageKit for AI tasks, Clerk for authentication, and Firebase Firestore for storage. A stable internet connection is required for API access and real-time synchronization.

2.5 Assumptions and Dependencies

Key assumptions and dependencies that affect the product design and implementation:

• Availability of external APIs: Gemini and ImageKit must be accessible and operational.

• Firebase availability: Firestore must be reachable for saving projects and slides.

•User accounts and subscription management rely on Clerk services.

• Users supply reasonably clear and concise topics; extremely vague inputs may require additional user prompts.

• The initial release will not include PPTX export or multi-user collaboration (these are planned enhancements).

CHAPTER 3

FUNCTIONAL REQUIREMENTS

3.1 Introduction

This chapter defines the functional requirements of the AI-Based Automated Presentation Generator. Functional requirements describe the core operations the system must perform, including user interactions, module behaviours, and expected outputs. These requirements ensure the system works reliably and provides the intended AI-driven automation for presentation creation.

3.2 User Authentication Requirements

• The system must allow users to sign up and log in securely using Clerk authentication.

• The system must validate subscription status and restrict features for free-tier users.

• The system must create and manage user sessions automatically.

• Only authenticated users may save, edit, or retrieve projects.

3.3 Topic Input & Outline Generation Requirements

• The system must allow the user to input a topic or theme for the presentation.

• The system must send the topic to Gemini AI for outline generation.

• The AI must return a structured outline with slide titles and bullet points.

• The generated outline must be visible and editable before proceeding to slide creation.

3.4 Slide Content Generation Requirements

• The system must convert the final outline into complete slide content.

• Slide titles, subtitles, and bullet points must be generated automatically.

• The AI must generate content in a clean, presentation-friendly structure.

• Users must be able to modify content in real time.

3.5 Image Generation Requirements

• The system must generate relevant images using ImageKit.io for each slide.

• The AI must interpret slide content to create appropriate visuals.

• Image URLs must be stored in Firestore for loading during slide rendering.

• Users may replace or regenerate images.

3.6 Slide Rendering & Editing Requirements

• The system must render slides dynamically using React and TailwindCSS.

• Users must be able to edit text, layout, and images directly.

• Changes must update instantly in preview mode.

• The editor must support adding, deleting, and reordering slides.

3.7 Project Management Requirements

• Users must be able to create new projects and save them to Firestore.

• The system must auto-save outlines, slides, and generated images.

• Users must be able to access previously created projects.

• Project metadata must include timestamps, user ID, and slide count.

3.8 System Integration Requirements

• The system must integrate Gemini AI for text generation.

• The system must integrate ImageKit for image creation.

• Firestore must store outlines, slide content, project info, and image URLs

• Clerk must manage authentication and subscription checks.

3.9 Summary

This chapter outlined all functional aspects of the system, including user authentication, AI outline generation, slide content creation, image generation, slide editing, and cloud-based project management. These requirements collectively ensure that the system delivers a seamless and automated presentation-building experience.

CHAPTER 4

NON-FUNCTIONAL REQUIREMENTS

4.1 Introduction

This chapter outlines the non-functional requirements that define how the system must behave in terms of performance, security, usability, reliability, and scalability. These requirements ensure that the AI-Based Automated Presentation Generator provides a smooth, predictable, and high-quality user experience.

4.2 Performance Requirements

• Outline generation must complete within 3–5 seconds.

• Slide content generation must remain responsive for long or complex topics.

• Real-time editor updates must apply instantly without lag.

• Image generation must complete within normal AI API response time.

• The system must support multiple users simultaneously without slowdown.

4.3 Security Requirements

• Clerk authentication must securely manage user login and sessions.

• Firestore rules must ensure only the owner can access their project data.

• API keys (Gemini, ImageKit) must be stored securely and never exposed to frontend.

• All communication between frontend and backend services must be encrypted.

• User data must be protected from unauthorized access or modification.

4.4 Usability Requirements

• The user interface must be simple, clean, and intuitive for beginners.

• The slide editor must allow easy editing, previewing, and navigation.

• System feedback must be clear, including error or loading messages.

• Users must be able to generate presentations with minimal actions.

• Interface should support all modern screen sizes (desktop-first).

4.5 Reliability Requirements

• Auto-save must ensure no loss of user data during editing.

• Firestore must consistently store and retrieve user projects.

• The system should remain stable even if AI requests temporarily fail.

• User sessions should persist reliably through Clerk authentication.

• The system must recover gracefully from network interruptions.

4.6 Availability Requirements

• The system should operate 24/7 except during maintenance.

• Firestore, Gemini AI, Clerk, and ImageKit must maintain high service uptime.

• Users must be able to access saved presentations at all times.

• System availability must not be impacted by the number of stored projects.

4.7 Scalability Requirements

• The system must scale with increasing user traffic and project volume.

• Firestore must store large numbers of outlines, slides, and images efficiently.

• AI workloads must scale automatically through cloud-based processing.

• Future enhancements (PPT export, collaboration, additional templates) must be integrable without major redesigns.

4.8 Summary

This chapter described all non-functional requirements necessary for ensuring a high-quality user experience. These requirements focus on system performance, security, usability, reliability, availability, and future growth capacity. They ensure that the AI-Based Automated Presentation Generator remains robust and scalable for long-term use.

CHAPTER 5

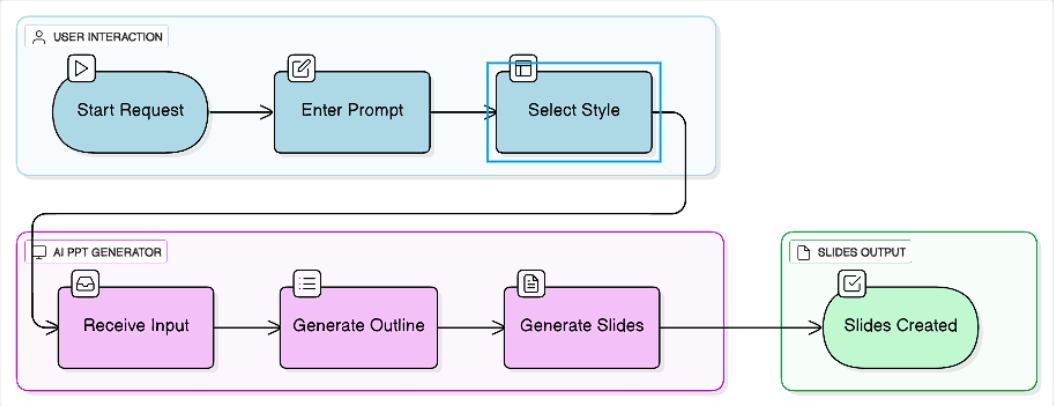
SYSTEM MODELS

5.1 Introduction

This chapter presents the system models used to represent the functional behavior, data flow, and structural architecture of the AI-Based Automated Presentation Generator. These diagrams help visualize how different modules interact, how data moves through the system, and how users interact with the platform. The models included in this chapter follow standard DFD and UML conventions and act as a blueprint for system implementation.

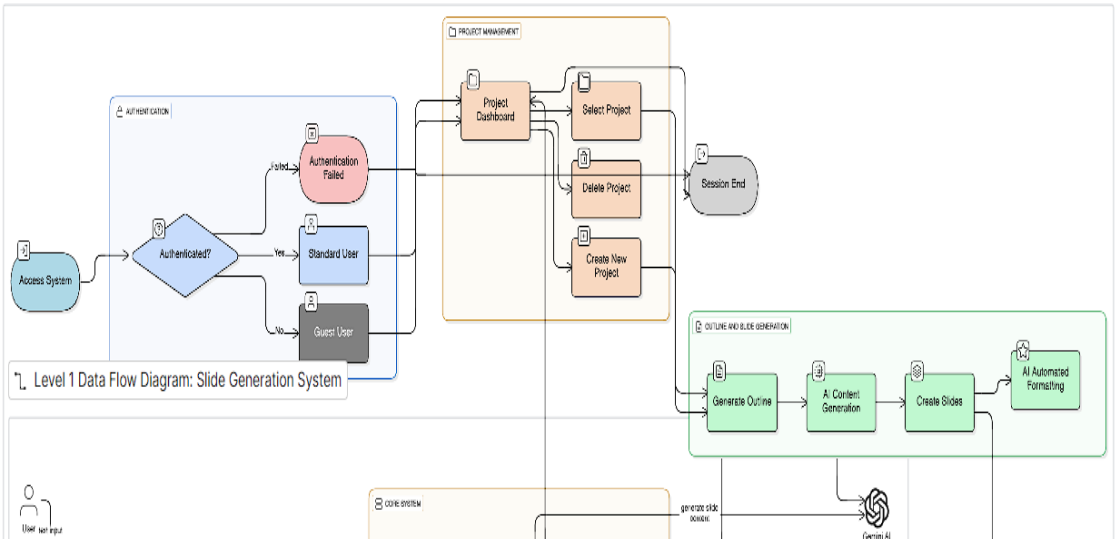
5.2 Context Level Diagram (Level 0 DFD)

The Level 0 Data Flow Diagram (Context Diagram) provides a high-level overview of the system. It shows the primary external entity (user) interacting with the system and highlights the major data exchanges. This diagram represents the overall boundary of the AI-Based Presentation Generator without internal processing details



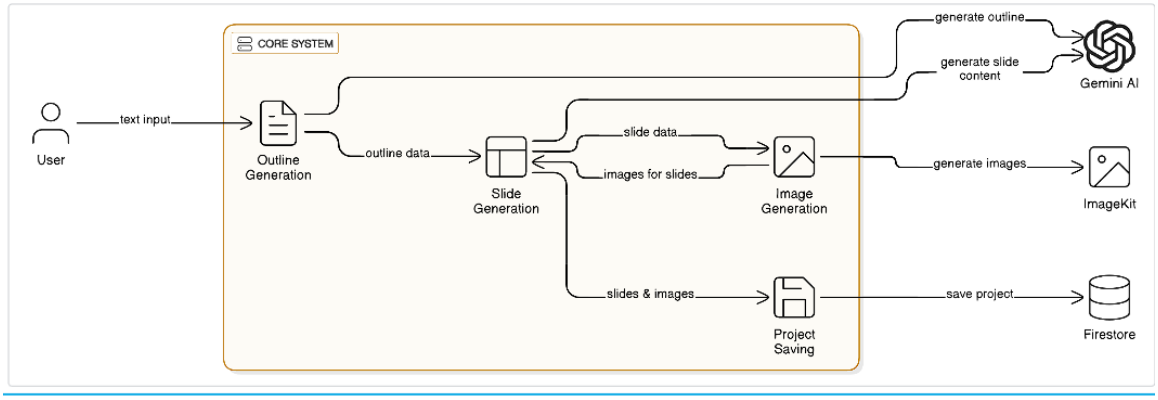
5.3 Use Case Diagram

The use case diagram identifies key system interactions from the user's perspective. It shows how the user authenticates, generates outlines, creates slides, edits slides, and manages multiple projects. It also highlights the AI interactions as system-level operations triggered by the user.



5.4 Sequence Diagram and Activity Diagram

The sequence diagram illustrates the step-by-step interaction between the user, frontend, Gemini AI, ImageKit, and Firebase Firestore. It highlights the order in which operations occur—from entering a topic to receiving AI generated content, previewing slides, and saving the final project. This model shows the dynamic flow of activities during presentation generation.



5.5 Summary

This chapter presented the essential system diagrams that visually explain how the AI-Based Automated Presentation Generator functions. The diagrams include the DFD Level 0, DFD Level 1, Use Case Diagram, Sequence Diagram, and Activity Diagram, each providing insight into different aspects of the system's structure and behavior. These placeholders can be replaced with the actual diagrams during the final submission.