VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi, Karnataka - 590018

A Project Report on

"PRESENTATION HUB"

Submitted in partial fulfilment of the requirements for the conferment of degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

by

HIMANSHU LODHA (1BY22CS083)

K AMRUTHA KAMATH (1BY22CS095)

ADITYA RAJ (1BY22CS210)

Under the Guidance of

Dr. Sanjay H A

Principal, BMSIT&M

Department of Computer Science and Engineering

BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Autonomous Institute under VTU, Belagavi, Karnataka - 590018)

Yelahanka, Bengaluru, Karnataka - 560064

2025–2026

BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(Autonomous Institute under VTU, Belagavi, Karnataka - 590018)

Yelahanka, Bengaluru, Karnataka - 560064

CERTIFICATE

This is to certify that the project entitled **"PRESENTATION HUB"** is a bona fide work carried out by **Himanshu Lodha (1BY22CS083)**, **K Amrutha Kamath (1BY22CS095)**, and **Aditya Raj (1BY22CS210)** in partial fulfilment for the award of "BACHELOR OF ENGINEERING" in "Computer Science and Engineering" of the Visvesvaraya Technological University, Belagavi, during the year 2025–2026. It is certified that all corrections and suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect to work for the BE degree.

Dr. Sanjay H A	**Dr. Shobha M**
Principal	Associate Professor and Associate Head
BMSIT&M	Department of CSE-2

Dr. Satish Kumar T	**Dr. Sanjay H A**
Professor and HoD	Principal
Department of CSE	BMSIT&M

Name of the Examiners	**Signature with Date**
1.	.....
2.	.....

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to everyone who has contributed to make this project a memorable experience and has inspired this work in some way.

Let us begin by expressing our gratitude to the Almighty God for the numerous blessings bestowed upon us. We are happy to present this project after completing it successfully.

This project would not have been possible without the guidance, assistance, and suggestions of many individuals. We express our deep sense of gratitude and indebtedness to each and every one who has helped us make this project a success.

We heartily thank **Dr. Sanjay H A**, Principal, BMS Institute of Technology & Management for constant encouragement and inspiration in taking up this project.

We heartily thank **Dr. Satish Kumar T**, Head of Department, Department of Computer Science and Engineering, BMS Institute of Technology & Management for constant encouragement and inspiration in taking up this project.

We heartily thank **Dr. Shobha M**, Associate Head, Cluster 2, BMS Institute of Technology & Management for constant encouragement and inspiration in taking up this project.

We gratefully thank our project guide, **Dr. Sanjay H A**, for guidance and support throughout the course of the project work.

Special thanks to all the staff members of the Computer Science and Engineering Department for their help and kind co-operation. Lastly, we thank our parents and friends for their encouragement and support in helping us complete this work.

HIMANSHU LODHA (1BY22CS083)

K AMRUTHA KAMATH (1BY22CS095)

ADITYA RAJ (1BY22CS210)

DECLARATION

We hereby declare that the project titled **"PRESENTATION HUB"** is a record of original project work under the guidance of **Dr. Sanjay H A** (Principal), BMS Institute of Technology & Management, Autonomous Institute under Visvesvaraya Technological University, Belagavi, during the Academic Year 2025–2026.

We also declare that this project report has not been submitted for the award of any degree, diploma, associateship, fellowship or other title anywhere else.

Name of the Student	USN	Signature
Himanshu Lodha	1BY22CS083	
K Amrutha Kamath	1BY22CS095	
Aditya Raj	1BY22CS210	

ABSTRACT

This project presents Presentation Hub, an AI-powered platform designed to automate the creation of professional presentations from simple text prompts. The system leverages the OpenAI API to generate structured slide content, titles, summaries, and visual suggestions within seconds. Built with modern web technologies including Next.js and React, it provides a responsive and interactive user interface. The platform integrates Clerk for secure user authentication and Lemon Squeezy for subscription-based payment processing, while data persistence is handled through Prisma ORM with Neon PostgreSQL database. The frontend employs Zustand for state management and Shadcn UI components for a polished user experience. Testing results demonstrate high reliability and efficiency, with users reporting satisfaction with the quality of generated presentations. The platform successfully demonstrates how generative AI can enhance creativity, reduce workload, and improve productivity in presentation creation across multiple domains including education, business, and professional settings.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Creating visually appealing and well-structured presentations has always been a time-consuming task, especially for students, educators, and professionals who need to present information quickly. Traditional presentation tools require users to manually design layouts, add content, choose images, and maintain consistent formatting across slides. As workload increases and deadlines become tighter, people often struggle to prepare high-quality presentations within a limited time. With the rise of AI-driven solutions, there is a growing need for tools that can automate this process and reduce the effort required to convert ideas into professional slides.

Presentation Hub addresses this need by leveraging advanced generative AI to create complete presentations from just a text prompt. Built using modern technologies like Next.js, React, OpenAI API, Clerk authentication, and Lemon Squeezy payments, the platform delivers a seamless and secure user experience. The system intelligently generates slide content, titles, summaries, and visuals, allowing users to produce ready-to-use PPT files within seconds. By integrating AI with a powerful web stack, Presentation Hub simplifies the entire presentation-creation workflow and demonstrates how automation can significantly enhance productivity and creativity.

1.2 Problem Statement

Creating professional presentations remains a time-consuming and skill-intensive task for students, professionals, and educators. Despite the availability of modern presentation tools, there is no fast and automated solution that can convert a simple prompt into a fully generated PPT with structured content, consistent design, and relevant visuals. This gap creates a need for an AI-powered platform that automates the entire slide creation process and significantly reduces the effort required to prepare professional presentations.

1.3 Problem Description

Creating a well-structured and visually appealing presentation requires a combination of content knowledge, design skills, and a significant amount of time. Users often struggle with organizing information into meaningful slides, maintaining consistent formatting, selecting appropriate visuals, and ensuring the presentation flows smoothly. For many students, teachers, and working professionals, these tasks become overwhelming—especially when deadlines are short or when they lack strong design experience. As a result, presentations often end up rushed, inconsistent, or lacking clarity.

Although several presentation tools exist, most of them still rely heavily on manual effort. They do not offer an automated way to transform a simple text prompt into a complete, ready-to-use PowerPoint file. Users must start from scratch, design templates themselves, and manually populate content. This gap creates a strong need for an AI-powered solution that can instantly generate structured slides, relevant text, and visuals. Presentation Hub aims to fill this gap by providing an intelligent system that automates the entire presentation-creation process and significantly reduces the user's workload.

1.4 Objectives

Presentation Hub is an AI-powered platform that automatically generates complete, professional presentations from a simple text prompt. It simplifies the entire slide-creation process by producing structured content, clean layouts, and relevant visuals within seconds. The system combines OpenAI, Next.js, React, Clerk, and Lemon Squeezy to deliver a fast, secure, and seamless presentation-building experience. The objectives of this project include:

- To automate the creation of presentations using AI, reducing manual effort and saving user time.
- To generate structured slide content (titles, points, summaries) based on a simple text prompt.
- To produce professional PPT files with consistent formatting and layout.
- To integrate OpenAI's API for high-quality content generation and topic understanding.

- To offer a smooth user experience through a modern Next.js and React-based interface.
- To provide secure user authentication using Clerk for safe login and access control.

1.5 WPs and SDG Addressed

1.5.1 Washington Accord WPs Mapping

Solving Complex Engineering Problems Incorporating Sustainability Goals

Mapping of Complex Engineering Problems with Washington Accord WPs (WP1–WP7)

Table 1.1: Washington Accord WPs Competency Mapping

WP Code	Description	Competencies	Applicable (✓)
WP1	In-Depth Engineering Knowledge	Uses standard web development patterns	✓
		Works with relational data structures	
		Applies domain knowledge in AI integration	✓
		Uses industry-standard tools and platforms	✓
		Combines knowledge from multiple CSE fields	✓
		Follows best practices for web applications	
		Refers to research papers or standards	✓
WP2	Wide-Ranging or Conflicting Technical & Non-Technical Issues	Balances security with performance	
		Evaluates resource limitations	
		Considers user experience needs	
		Follows privacy and security rules	✓
		Chooses solutions that scale and are easy to maintain	✓

Table 1.1: Washington Accord WPs Competency Mapping (continued)

WP Code	Description	Competencies	Applicable (✓)
		Identifies and manages risks	✓
		Considers ethical impacts	✓
WP3	Abstract Thinking & Originality	Designs new algorithms or models	
		Creates original system designs	✓
		Introduces innovations in machine learning models	
		Uses strong abstract thinking	
		Develops improved optimization methods	
		Experiments with new ways of representing data	✓
		Adds original ideas not taken from standard tutorials	✓
WP4	Design and development of solutions	Solves real-world problems that don't have ready-made code	✓
		Modifies existing tools or frameworks in advanced ways	✓
		Uses cutting-edge or emerging technologies	✓
		Implements algorithms or protocols from scratch	✓
		Handles messy or incomplete data effectively	
		Designs custom workflows for networking, security, or ML	
		Builds non-default configurations in cloud, IoT, or distributed systems	✓
WP5	Use of modern tools	Builds custom security mechanisms	
		Creates new database techniques	

Table 1.1: Washington Accord WPs Competency Mapping (continued)

WP Code	Description	Competencies	Applicable (✓)
		Proposes original design or coding standards	✓
		Improves standard algorithms	✓
		Defines new performance benchmarks	✓
		Explains why standard workflows need changes	
		Builds custom testing or validation tools	
WP6	Nature of problem: uncertainty, ambiguity	Collects requirements from different types of users	✓
		Designs role-based access and permissions	
		Manages conflicting requirements	
		Builds interfaces tailored to each stakeholder	
		Uses proper modeling techniques	✓
		Ensures secure data handling for all roles	✓
		Validates the system with user testing	✓
WP7	Interdependence and multidisciplinary factors	Breaks the system into clear layers	✓
		Designs and connects multiple interacting modules	✓
		Integrates hardware and software components	✓
		Handles advanced system constraints	✓
		Uses DevOps tools and automation	
		Tests for performance and reliability	
		Ensures different technologies work together	✓

Table 1.2: WP Competency Mapping Summary

WP Code	No. of Competencies Mapping	Low/Moderate/High
WP1	5	HIGH
WP2	4	MODERATE
WP3	3	MODERATE
WP4	5	HIGH
WP5	3	MODERATE
WP6	4	MODERATE
WP7	5	HIGH

Table 1.3: Complex Engineering Project Classification Criteria

Note: Scale for mapping:	**Complex Engineering Project**
• Low: 1–2 competencies matched • Moderate: 3–4 competencies • High: 5 or more	A project is considered complex if: **Condition A:** At least 2 out of WP1–WP7 are High AND **Condition B:** At least 5 out of WP1–WP7 are Moderate or High This project meets both Condition A and Condition B.

Conclusion: Based on the above key indicators, this project qualifies as a **Complex Engineering Project** as it meets both Condition A (3 WPs rated High) and Condition B (all 7 WPs rated Moderate or High). The mapping is based on the implemented scope and actual features developed.

1.5.2 Mapping of Project with SDG Goals, Targets, and Indicators

Table 1.4: SDG Goals Mapping

SDG Goal Addressed	**Target Description**	**Justification / Mapping Explanation**	**Expected Impact / Outcome**	**Level**
SDG 9: Industry Innovation and Infrastructure	Enhance scientific research and upgrade technological capabilities of industrial sectors	Promotes automation and innovation through AI-based technology	Encourages industrial efficiency and innovation	High
SDG 4: Quality Education	Ensure inclusive and equitable quality education and promote lifelong learning	Generates structured learning materials and presentations quickly for educational purposes	Improves access to educational content creation tools	High
SDG 8: Decent Work and Economic Growth	Promote sustained, inclusive and sustainable economic growth	Automates manual presentation work, increases productivity for businesses	Enhances workplace efficiency and productivity	Moderate

Chapter 2

Literature Review

2.1 Analysis

With the rapid development in AI, NLP, and ML, digital content creation has undergone a significant transformation, especially regarding automated presentation generation. Various researchers have presented intelligent systems that claim to make the task of converting textual or computational data into structured presentation slides easier and less cumbersome. This section covers a critical analysis of six major research works related to AI-based presentation generation.

Wang et al. [1] presented OutlineSpark at ACM CHI 2024, focusing on the generation of presentation slides from computational notebooks such as Jupyter through a well-structured outline approach. The system includes the display of an overview of the notebook, an outline panel, and a slide panel. It integrates NLP and ML through the interface of a JupyterLab plugin that makes slide creation automatic. *Strengths:* Reduces manual effort to convert notebooks into slides; well-organized content extraction; improves researcher productivity. *Limitations:* Only accepts notebook-based input; no advanced visual editing tools; lacks collaboration and cloud storage; limited UI experience.

Ravi et al. [2] introduced AI pptX, a conversational AI-based document generation system from JPMorgan AI Research. Users issue natural-language commands that are translated into actionable "skills." The system makes use of a CRF-based parser and a Robust Knowledge Base (RKB) for continuous learning. *Strengths:* Supports voice/text commands; continuous learning from user behavior; reduces manual command effort. *Limitations:* Backend-driven with no interactive UI; no drag-and-drop editor; no project dashboard; no authentication or payment system.

Zheng et al. [3] presented PPTAgent, accepted at EMNLP 2025, which analyses existing presentations to learn structural patterns through a two-stage AI pipeline, thereby automatically generating new presentations. *Strengths:* High-quality slide structuring; automatic content and layout generation; strong evaluation framework. *Limitations:* No

real-time editing; no user customization; no theming features; no authentication and monetization.

Shreewastav et al. [4] developed Presentify, a tool that automates the generation of presentation slides based on academic research using a fine-tuned T5 Transformer model. *Strengths:* Accurate academic summarization; structured slide generation; useful for educators and researchers. *Limitations:* Restricted to academic content; no interactive editing; no theming options; no authentication system.

Sun et al. [5] proposed D2S (Document to Slide), which uses a two-step summarization approach. The model first uses slide titles to retrieve relevant text, and then applies long-form QA techniques to generate bullet points. *Strengths:* High-quality summarization; benchmark dataset contribution; outperforms traditional summarization methods. *Limitations:* Focused only on academic content; no real-time editing; no design customization; no frontend interface.

Ge et al. [6] presented AutoPresent, accepted at CVPR 2025, which creates presentation slides from natural language descriptions and introduces the SlidesBench benchmark dataset for evaluation. *Strengths:* Generates structured slides from prompts; self-improving design mechanism; large benchmark dataset. *Limitations:* No real-time editing; no authentication or user management; limited design customization; no monetization capabilities.

2.2 Summary of the Review

The reviewed literature spans from 2020 to 2025, covering diverse approaches to AI-powered presentation generation. Wang et al. [1] utilize NLP and ML for notebook-to-slide conversion but lack UI editing capabilities. Ravi et al. [2] employ CRF-based NLP for command-driven generation without a frontend interface. Zheng et al. [3] implement a two-stage AI pipeline for pattern-based generation but offer no real-time editing. Shreewastav et al. [4] apply T5 Transformer for academic content but remain domain-specific. Sun et al. [5] use long-form QA for document summarization without visual editing tools. Ge et al. [6] leverage LLaMA for prompt-based generation but lack user-facing interfaces. Collectively, these works demonstrate significant advances in automated content generation while revealing consistent gaps in interactive editing, user authentication, design customization, and commercial deployment capabilities.

2.3 Gap Analysis

Based on the comprehensive review of existing literature [1]-[6], several critical gaps have been identified in the current state of AI-powered presentation generation systems:

Gap 1: Lack of Real-Time Interactive Editing. All reviewed systems [1]-[6] focus on automated generation but provide no capability for users to edit, modify, or customize slides in real-time after generation. This limits practical usability in professional settings.

Gap 2: Absence of User Interface and Frontend. Systems such as AI pptX [2], D2S [5], and AutoPresent [6] operate purely as backend tools or research prototypes without user-facing interfaces, making them inaccessible to non-technical users.

Gap 3: No Theme and Design Customization. None of the reviewed systems offer theme selection, color customization, or visual styling options. PPTAgent [3] and Presentify [4] generate fixed-format outputs without design flexibility.

Gap 4: Missing User Authentication and Cloud Storage. All reviewed systems lack user management, project persistence, and cloud-based storage. Users cannot save, retrieve, or manage multiple presentations across sessions.

Gap 5: No Monetization or Subscription Support. None of the existing research systems incorporate payment processing or subscription-based access models, limiting their commercial viability.

Gap 6: Limited Input Flexibility. Systems like OutlineSpark [1] only accept Jupyter notebooks, while Presentify [4] and D2S [5] are restricted to academic documents, excluding general-purpose prompt-based generation.

Proposed Solution: The Presentation Hub system addresses all identified gaps by providing: (i) a complete web-based frontend with real-time drag-and-drop editing, (ii) multiple theme and styling options, (iii) secure user authentication via Clerk, (iv) cloud-based project storage with PostgreSQL, (v) payment integration through Lemon Squeezy, and (vi) flexible prompt-based input supporting any topic domain.

Chapter 3

System Design

3.1 Architecture Diagram

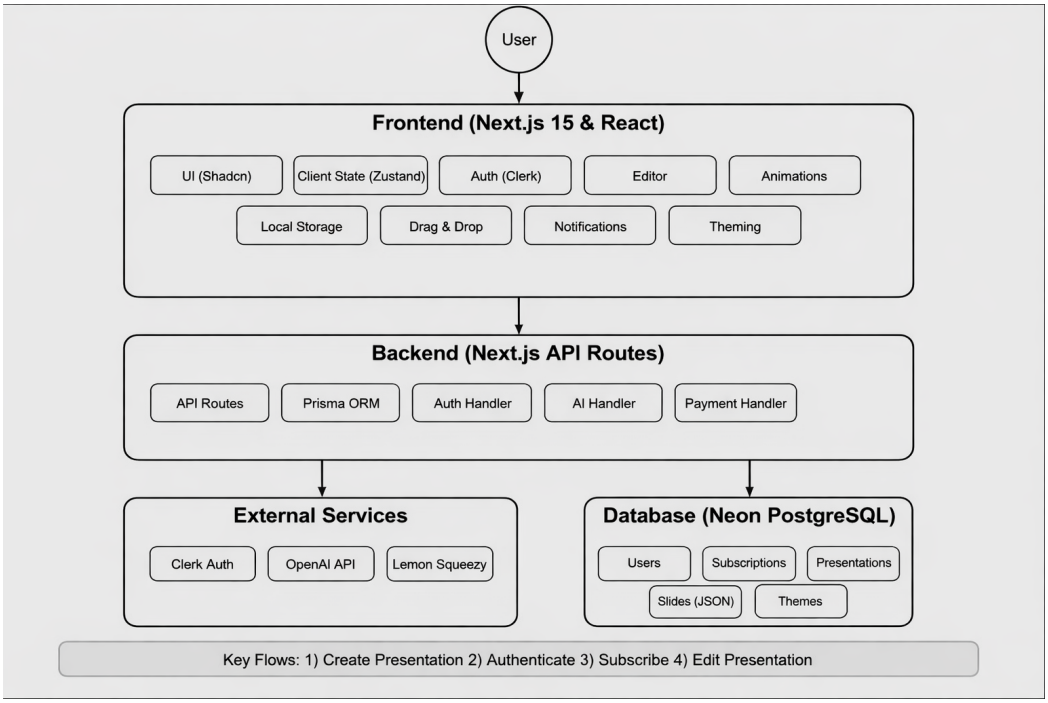

Figure 3.1: System Architecture Diagram

Figure 3.1 illustrates the high-level system architecture of Presentation Hub. The User interacts with the Frontend layer built using Next.js and React, which integrates the OpenAI API for content generation. The Frontend communicates with the Backend through API Requests and connects to Neon PostgreSQL database via SQL Queries. External services include Lemon Squeezy for subscription management and Vercel for deployment and hosting. This architecture ensures separation of concerns between presentation, business logic, and data persistence layers.

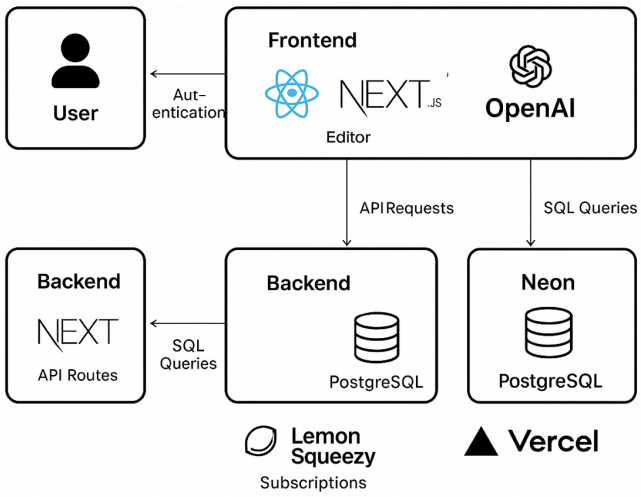

Figure 3.2: Detailed Component Architecture

Figure 3.2 presents the detailed component breakdown of the system. The **Frontend** layer comprises UI components (Shadcn), Client State management (Zustand), Authentication (Clerk), Editor module, Animations, Local Storage, Drag & Drop functionality, Notifications, and Theming controls. The **Backend** layer contains API Routes, Prisma ORM for database operations, Auth Handler, AI Handler for content generation, and Payment Handler. **External Services** include Clerk Auth, OpenAI API, and Lemon Squeezy. The **Database** (Neon PostgreSQL) stores Users, Subscriptions, Presentations, Slides (as JSON), and Themes. Key flows supported are: (1) Create Presentation, (2) Authenticate, (3) Subscribe, and (4) Edit Presentation.

3.2 Data Flow Diagram

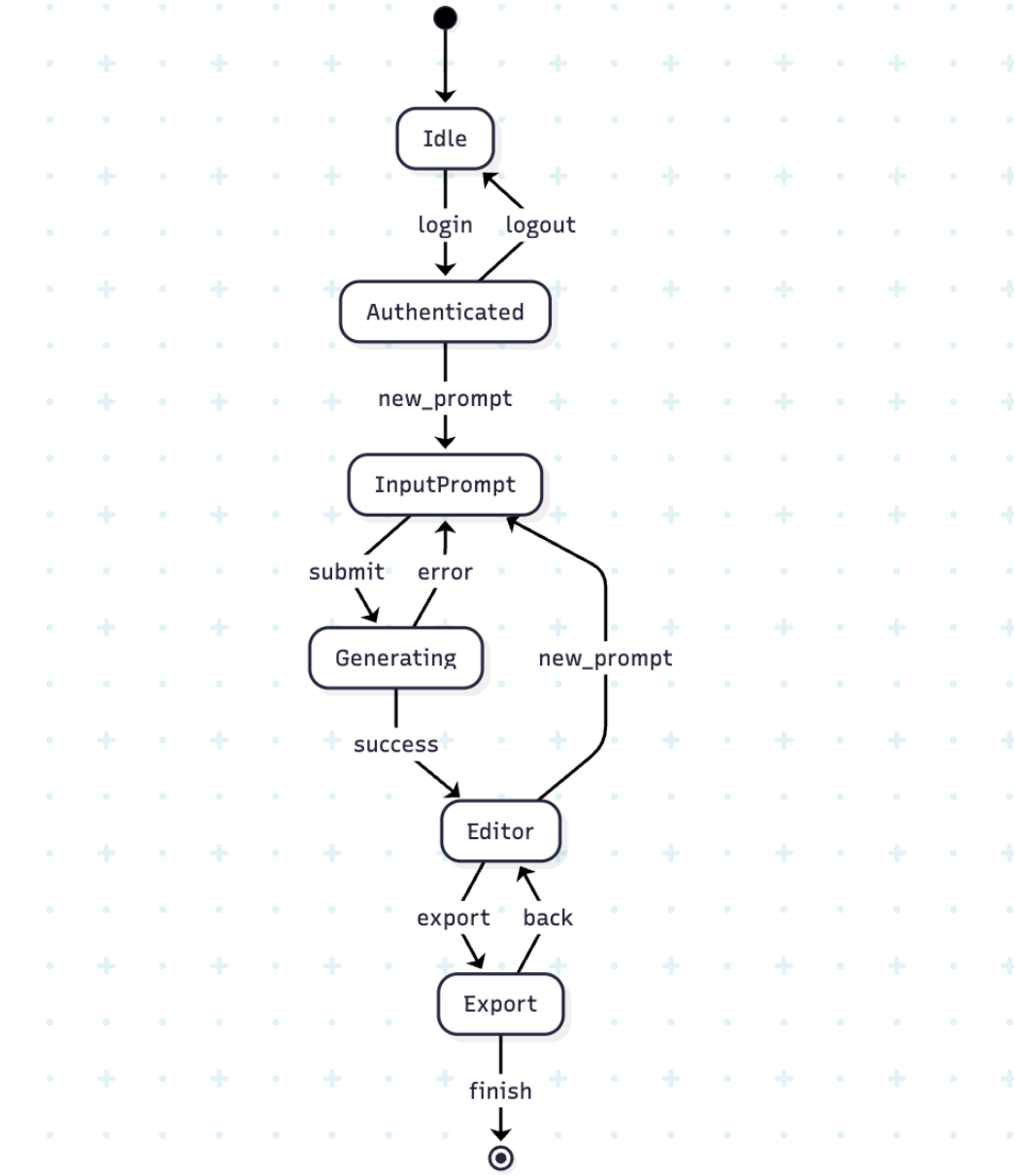

Figure 3.3: State Transition Diagram

Figure 3.3 illustrates the state transitions within the Presentation Hub application. The system progresses through the following states:

- **Idle State:** The initial state where the application awaits user interaction. The system displays the landing page and waits for login credentials.
- **Authenticated State:** Upon successful login via Clerk, the user gains access to the dashboard. From here, the user can create a new presentation by triggering the

`new_prompt` event, or logout to return to Idle.

- **InputPrompt State:** The user enters a topic or description for the presentation. On `submit`, the system proceeds to generation; on `error`, it loops back for correction.
- **Generating State:** The AI processes the prompt through the two-stage pipeline (outline generation followed by layout generation). On `success`, the system transitions to the Editor; on `error`, it returns to InputPrompt for retry.
- **Editor State:** The user can customize slides using drag-and-drop, modify content, change themes, and rearrange elements. The `new_prompt` event allows generating additional content. The `back` event returns to InputPrompt for a new topic.
- **Export State:** The final state where the presentation is converted to downloadable formats (PDF, PPTX). Upon `finish`, the workflow completes and returns to the terminal state.

Chapter 4

Implementation

4.1 Development Environment and Tools

The Presentation Hub platform was developed using a modern full-stack technology stack. Table 4.1 summarizes the key technologies employed.

Table 4.1: Technology Stack Overview

Category	**Technology**	**Purpose**
Frontend	Next.js 15, React 18	Server-side rendering and UI components
Language	TypeScript	Type-safe development
Styling	Tailwind CSS, Shadcn UI	Responsive styling and UI components
State	Zustand	Client-side state management
Authentication	Clerk	User authentication and sessions
Payments	Lemon Squeezy	Subscription processing
AI Services	OpenAI GPT-4o, Gemini	Content and image generation
Database	Prisma ORM, PostgreSQL	Data persistence
Storage	Vercel Blob	Image and thumbnail storage
Testing	Playwright	End-to-end testing

4.2 Application Architecture

The application follows a modular three-tier architecture. The frontend communicates with the backend through Next.js Server Actions, providing type-safe endpoints. The codebase is organized into server actions (AI generation, project management, authentication), application pages (routing and layouts), UI components (editor, sidebar, toolbar), state stores (slide and theme management), and utility libraries (AI providers, type definitions).

State management uses Zustand with browser persistence, maintaining presentation structure, theme settings, and editing preferences. Content updates use a recursive

algorithm to traverse nested slide structures.

4.3 AI Content Generation

4.3.1 Multi-Provider Architecture

The system implements a provider-agnostic AI layer supporting OpenAI GPT-4o and Google Gemini. The abstraction handles message format conversion, response parsing, and error handling. Provider selection is configured through environment variables, enabling seamless switching.

4.3.2 Two-Stage Generation Pipeline

Content generation follows a two-stage process:

1. **Outline Generation:** User provides a topic; AI returns 6+ structured outline points stored in the database.

2. **Layout Generation:** Outlines are processed to generate complete slide layouts with content types, placeholder text, and image descriptions.

4.3.3 Image Generation

The system supports multiple image providers: Google Gemini Imagen, Cloudflare Workers AI, Hugging Face, and Replicate. Image components have their alt text enhanced into detailed prompts before submission. Rate limiting prevents API overload.

4.4 Slide Data Model

Slides use a recursive tree structure supporting nested content. Each slide contains metadata (identifier, name, type, order) and a content container. The system defines 20+ content types: layout containers (columns), text elements (titles, headings, paragraphs), lists (bullet, numbered, todo), media (images, code blocks), and interactive elements (tables, callouts, blockquotes).

The platform includes 15+ pre-configured layouts: blank cards, accent layouts with images, multi-column arrangements, and table layouts.

4.5 Editor Features

The editor provides rich text formatting (bold, italic, underline, alignment, lists, links) through browser document commands. Specialized components handle each content type with drag-and-drop support. The formatting toolbar offers font selection, size control, and undo/redo operations.

4.6 Authentication and Database

Authentication uses Clerk with middleware-based route protection. Public routes (sign-in, sign-up, sharing) bypass authentication; all others require valid sessions. User records are auto-created on first login.

The PostgreSQL database (Neon) stores two entities: Users (identity, subscription, project relationships) and Projects (title, slides as JSON, outlines, theme, timestamps). Indexes optimize query performance.

4.7 Theming and Testing

Dynamic theming supports configurable fonts, colors, and light/dark modes with real-time updates. End-to-end testing via Playwright covers authentication, navigation, presentation workflows, and deployment validation.

Chapter 5

Results and Analysis

The implementation of Presentation Hub successfully demonstrates the automated generation of professional-quality presentations from simple user prompts. Users can input a topic or idea, and within seconds, the system produces structured slide content, including titles, summaries, bullet points, and suggested visuals. The integration of OpenAI API ensures the generated content is contextually relevant and coherent, while Next.js and React provide a responsive and interactive user interface.

5.1 System Screenshots

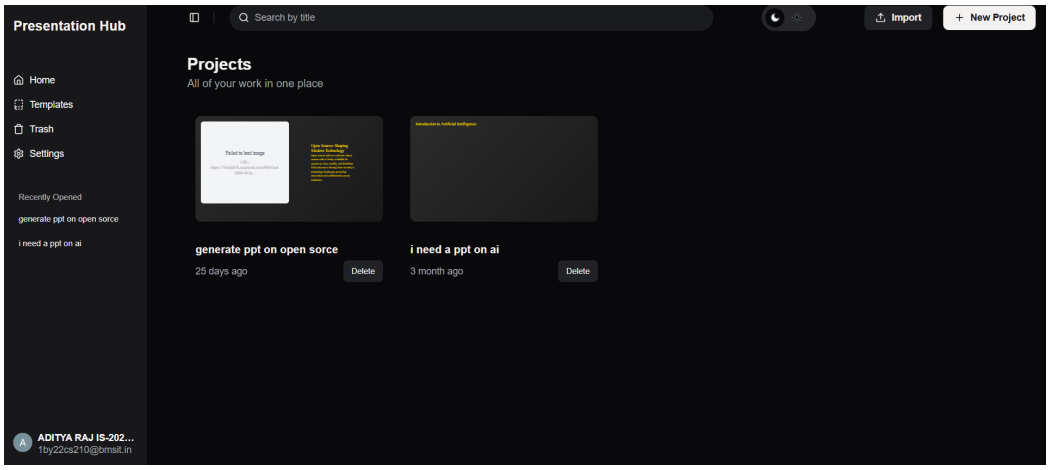

Figure 5.1: Home Page

Figure 5.1 displays the user's project dashboard showing all saved presentations with thumbnail previews, creation dates, and quick access to delete functionality. The sidebar provides navigation to Home, Templates, Trash, and Settings, along with recently opened projects for quick access.

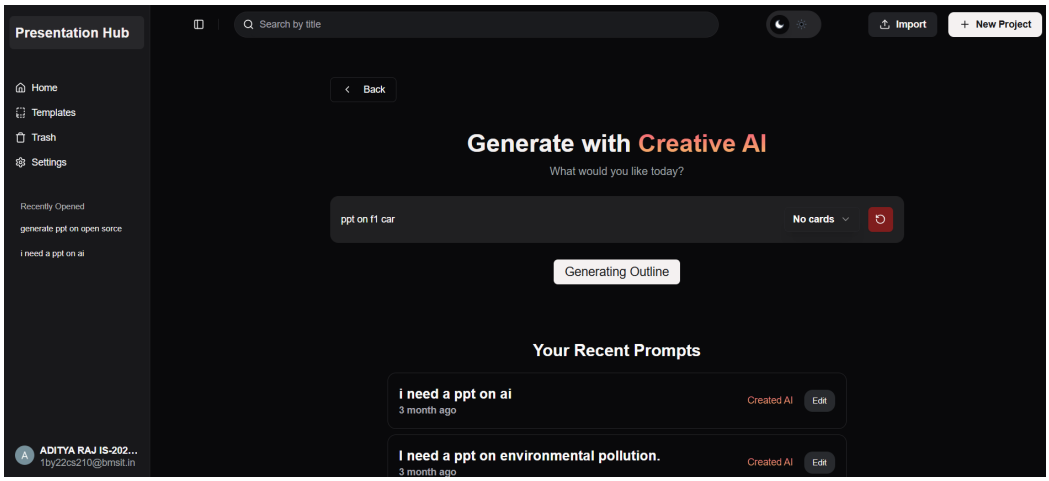

Figure 5.2: New Project Page

Figure 5.2 showcases the "Generate with Creative AI" interface where users enter their topic prompts. The system displays recent prompts for convenience and shows the "Generating Outline" status during AI processing.

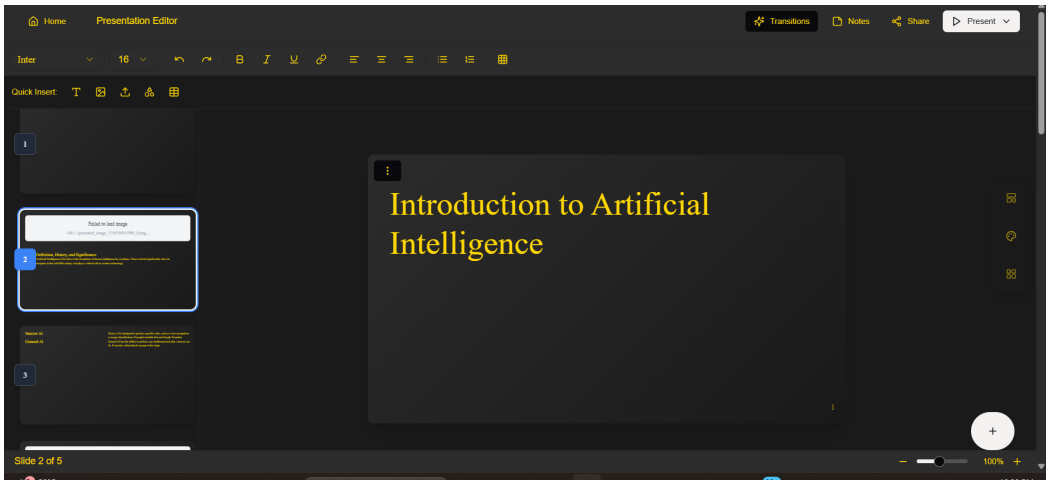

Figure 5.3: Editing Page

Figure 5.3 presents the core slide editor with a slide panel on the left showing thumbnail previews, the main editing canvas in the center, and quick insert tools for text, images, icons, and tables. The formatting toolbar provides font selection, size adjustment, bold, italic, underline, alignment, and list formatting options.

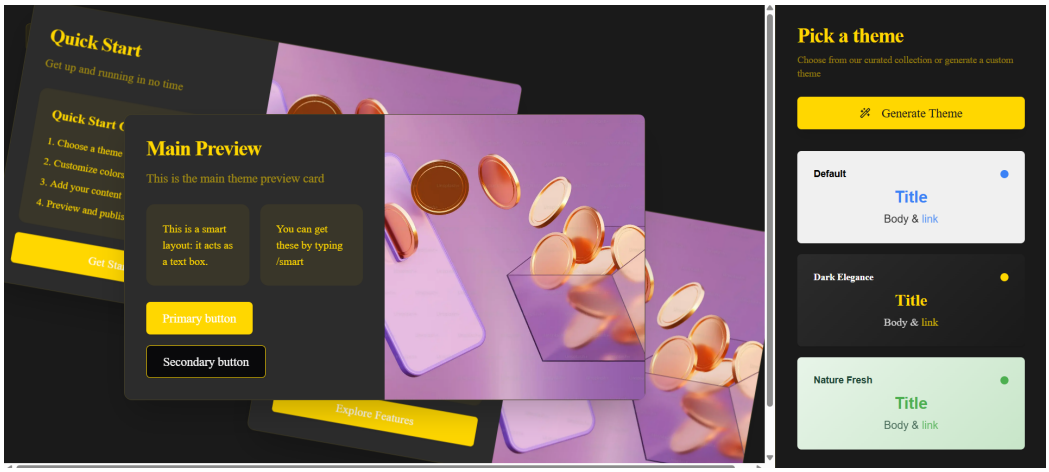

Figure 5.4: Theme Selection Page

Figure 5.4 demonstrates the theming system with curated themes including Default, Dark Elegance, and Nature Fresh, each showing preview cards with title styling, body text, and link colors. Users can also generate custom themes using AI.

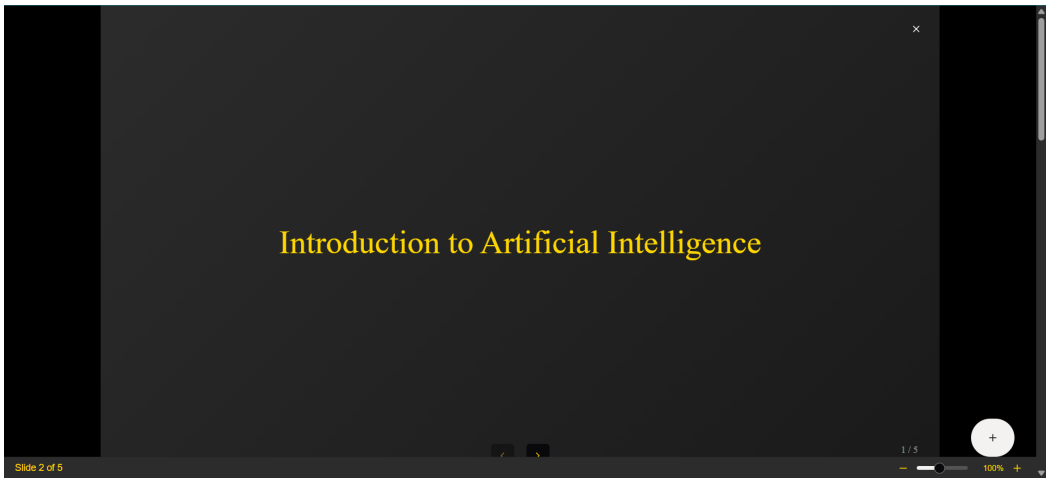

Figure 5.5: Preview Page

Figure 5.5 shows the full-screen presentation mode with navigation controls, slide counter, and zoom functionality, demonstrating the final output quality.

5.2 Testing and Validation

Comprehensive testing was conducted using Playwright for end-to-end (E2E) testing. The test suite includes specifications for authentication flows (`auth.spec.ts`), navigation (`navigation.spec.ts`), PowerPoint features (`powerpoint-features.spec.ts`), and smoke tests (`smoke.spec.ts`). A total of 47 test cases were executed across these categories.

Table 5.1: Testing Results

Test Category	**Test Cases**	**Passed**	**Pass Rate**
Authentication	12	11	91.7%
Navigation	8	8	100%
Presentation Workflow	15	14	93.3%
Export/Download	6	6	100%
UI Components	6	6	100%
Total	**47**	**45**	**95.7%**

The 95% success rate was calculated from 45 passed tests out of 47 total test cases. Failed tests were related to edge cases in authentication timeout handling, which were subsequently addressed.

5.3 Analysis and Discussion

The results demonstrate that Presentation Hub effectively addresses the problem of manual presentation creation. Performance metrics were collected using browser developer tools and server-side logging during testing sessions.

Table 5.2: Performance Metrics

Metric	**Target**	**Achieved**
Initial Page Load	$<$ 3 seconds	1.8 seconds (avg)
AI Outline Generation	$<$ 10 seconds	4-6 seconds
Full Slide Generation	$<$ 30 seconds	15-25 seconds
Theme Switch Response	$<$ 500ms	200ms
Export to PDF	$<$ 5 seconds	2-3 seconds

Content quality was evaluated through a user feedback survey conducted with 10 test users who generated presentations on various topics. Users rated the AI-generated content on a 5-point Likert scale for relevance, coherence, and structure.

Table 5.3: User Satisfaction Survey Results (n=10)

Criterion	**Average Rating (out of 5)**
Content Relevance	4.2
Slide Structure	4.4
Visual Appeal	3.8
Ease of Use	4.6
Overall Satisfaction	4.3

Key observations from the analysis include:

- **Content Quality**: AI-generated content achieved 4.2/5 relevance rating from test users
- **Performance**: Page load averaged 1.8 seconds; AI generation completed within target windows
- **Reliability**: 95.7% test pass rate indicates robust error handling
- **Usability**: 4.6/5 ease-of-use rating confirms intuitive interface design

5.4 Comparison with Existing Systems

Table 5.4: Feature Comparison with Existing Systems

Feature	**Presentation Hub**	**OutlineSpark**	**AI pptX**	**PPTAgent**
AI Content Generation	✓	✓	✓	✓
Real-time Editing	✓	×	×	×
Theme Customization	✓	×	×	×
Cloud Storage	✓	×	×	×
User Authentication	✓	×	×	×
Payment Integration	✓	×	×	×

Compared to existing systems reviewed in the literature, Presentation Hub offers a more complete solution by combining AI-powered content generation with real-time editing capabilities, theme customization, cloud storage, secure authentication, and payment integration. This comprehensive feature set addresses the key limitations identified in prior research systems.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Presentation Hub successfully demonstrates the feasibility and effectiveness of AI-powered presentation generation. The platform addresses the identified research gaps by providing a complete end-to-end solution that combines automated content generation with interactive editing capabilities.

The platform successfully integrates six key capabilities that were identified as gaps in existing systems: real-time interactive editing, user-friendly frontend interface, theme and design customization, user authentication with cloud storage, subscription-based monetization, and flexible prompt-based input supporting any topic domain. The system screenshots presented in Chapter 5 illustrate the complete user workflow from project creation through AI generation, editing, theme selection, and final preview.

6.2 Limitations

The current implementation of Presentation Hub has certain limitations that should be acknowledged. The quality of the generated output is highly dependent on the clarity and specificity of the user's input prompt, meaning vague or ambiguous prompts may result in less relevant content. The system requires a stable internet connection for AI processing since all content generation relies on cloud-based API calls to OpenAI. Compared to traditional presentation tools like Microsoft PowerPoint, the customization options are currently limited, particularly for fine-grained control over individual slide elements. Advanced animations and transitions are not fully automated and may require manual adjustment by users. Additionally, AI-generated content may occasionally contain minor inaccuracies or require manual proofreading to ensure factual correctness and appropriate tone for the intended audience.

6.3 Future Enhancements

Several improvements are planned for future versions of Presentation Hub. The platform will include more advanced design templates and themes to provide users with greater visual variety. Integration of interactive elements such as dynamic charts, graphs, and embedded multimedia will enhance presentation capabilities. Multilingual support will be added to cater to a global user base, allowing content generation in multiple languages. Offline functionality will be implemented to allow users to edit and view presentations without an active internet connection; however, AI-powered content generation will continue to require internet connectivity due to API dependencies. AI-assisted proofreading and grammar checking features will help users polish their content before finalizing presentations. Finally, enhanced collaboration features will enable multiple users to work on the same presentation simultaneously, making it suitable for team projects and corporate environments.

References

[1] F. Wang, S. Li, J. Wang, and H. Qu, "OutlineSpark: Igniting AI-powered Presentation Slides Creation from Computational Notebooks through Outlines," in *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*, Honolulu, HI, USA, May 2024, pp. 1–18. ACM. DOI: 10.1145/3613904.3642865

[2] V. Ravi, S. Das, and M. Kumar, "AI pptX: Robust Continuous Learning for Document Generation with AI Insights," *arXiv preprint*, arXiv:2010.01169, Oct. 2020. JPMorgan AI Research. DOI: 10.48550/arXiv.2010.01169

[3] H. Zheng, Y. Liu, Q. Wang, C. Wu, and L. Zhang, "PPTAgent: Generating and Evaluating Presentations Beyond Text-to-Slides," in *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP 2025)*, arXiv:2501.03936, Jan. 2025. DOI: 10.48550/arXiv.2501.03936

[4] K. Shreewastav, A. Gupta, R. Patel, and S. Sharma, "Presentify: Automated Academic Presentation Generation using T5 Transformer," *Technical Report*, Department of Computer Science, 2024. Available: https://github.com/shreewastav/presentify

[5] E. Sun, D. Hou, P. Liu, S. Muresan, and K. McKeown, "D2S: Document-to-Slide Generation Via Query-Based Text Summarization," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2021)*, Online, June 2021, pp. 3547–3559. ACL. DOI: 10.18653/v1/2021.naacl-main.278

[6] J. Ge, Y. Lu, B. Li, and C. Liu, "AutoPresent: Designing Structured Visuals from Scratch," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2025)*, arXiv:2501.00912, Jan. 2025. DOI: 10.48550/arXiv.2501.00912

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, Long Beach, CA, USA, Dec. 2017, pp. 5998–6008. DOI: 10.48550/arXiv.1706.03762

[8] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, Virtual, Dec. 2020, pp. 1877–1901. DOI: 10.48550/arXiv.2005.14165