Shri Shamrao Patil (Yadravkar) Educational & Charitable Trust's

Sharad Institute of Technology College of Engineering

(Autonomous Institute) Yadrav (Ichalkaranji)-416121 (Approved by AICTE, New Delhi, Recognized by Government of Maharashtra & Affiliated to DBATU)

DEPARTMENT OF ARTIFICIAL INTELLIGIENCE AND DATA SCIENCE

A

Project Report

on

Gen AI: A Generative AI Website

Submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology

in

ARTIFICIAL INTELLIGIENCE AND DATA SCIENCE

by

1.	**Mr. Sumit Santosh Asalkar**	**2210101001**
2.	**Mr. Parth Sambhaji Jadhav**	**2210101018**
3.	**Mr. Rahul Narayanan Menon**	**2210101030**
4.	**Mr. Sujit Vijay Sadalage**	**2210101040**

under the guidance of

Mrs. A.S. Patil
Assistant Professor

DEPARTMENT OF ARTIFICIAL INTELLIGIENCE AND DATA SCIENCE

Sharad Institute of Technology college of Engineering, Yadrav Ichalkranji

2025-26

Shri Shamrao Patil (Yadravkar) Educational & Charitable Trust's
SHARAD INSTITUTE OF TECHNOLOGY COLLEGE OF ENGINEERING
An Autonomous Institute
(Approved by AICTE, New Delhi, Recognized by Government of Maharashtra & Affiliated to DBATU)
DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA SCIENCE

CERTIFICATE

This is to certify that, Studying in B-Tech. Artificial Intelligence & Data Science have successfully completed the project entitled ,"Gen AI: A Generative AI Website" carried out by

Mr. Sumit Santosh Asalkar.	2210101001
Mr. Parth Sambhaji Jadhav.	2210101018
Mr. Rahul Narayanan Menon.	2210101030
Mr. Sujit Vijay Sadalage	2210101040

Under the guidance and supervision of **Mrs. A.S. Patil** during the academic year 2025 - 2026 towards the partial fulfillment of the Bachelor of Technology in Artificial Intelligence & Data Science, Sharad Institute of Technology College of Engineering, Yadrav.

Mrs. A.S. Patil
Guide
Dept. of AI&DS,
SITCOE, Yadrav

Dr. A.J. Chinchwade
Head of Department of
AI&DS,
SITCOE, Yadrav

Dr. B.B. Sangame
Principal
SITCOE, Yadrav

External Examiner

ACKNOWLEDGEMENT

We would like to express our deep and sincere gratitude to our Guide **Mrs. A.S. Patil**, Department of Artificial Intelligence & Data Science, for guiding us to accomplish this project work. It was our privilege and pleasure to work under her able guidance. We are indeed grateful to her for providing helpful suggestion, from time to time. Due to her constant encouragement and inspiration we are able to present this project. We express our deep gratitude to **Dr.A.J.Chinchawade**, Head of Artificial Intelligence & Data Science Department, for his valuable guidance and constant encouragement. We are very much thankful to **Dr. B.B. Sangame** , Principal, Sharad institute of technology college of engineering, Yadrav - Ichalkaranji for providing all the necessary facilities to carry out project work. Last but not least we are thankful to our parents for their moral as well as financial support.

ABSTRACT

The rapid proliferation of generative AI has introduced a diverse range of tools that significantly enhance productivity and creativity. However, these tools are often fragmented across various platforms, frequently gated by mandatory sign-ups, subscriptions, and complex user interfaces. This project, titled "GenAI: A Generative AI Website" addresses these challenges by developing a unified, free-to-use, and registration-free web platform. GenAI consolidates a comprehensive suite of generative AI functionalities into a single, intuitive ecosystem. The platform offers a wide array of tools categorized under Text & Documents, Image & Video, Developer, and Productivity, including features like AI resume building, PDF summarization, code translation, text-to-image generation, and background removal, alongside a conversational GenAI Assistant. The system is architected using a modern web stack with Node.js and Express.js for the backend, and its core functionality is powered by the seamless integration of advanced large language models such as Google's Gemini series and OpenAI's GPT models. By eliminating common barriers to access, GenAI successfully demonstrates a user-centric model that democratizes generative AI technology, aiming to save time, reduce effort, and boost efficiency for a broad user base.

Keywords: Generative AI, Productivity Tools, Centralized Platform, Automation, AI Integration, LLMs, NLP, Genkit, express route, Node.js, CNN.

INDEX

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

In the current digital era, generative artificial intelligence (AI) is increasingly being adopted to accelerate content creation, automate repetitive tasks, and improve productivity across many domains. The *Gen AI: A Generative AI Platform* project presents a unified, no-login, web-accessible environment that consolidates multiple generative capabilities — text, image, video, speech, and developer tools — into a single platform. The goal is to remove friction from everyday creative and productivity workflows by providing immediate, integrated access to a broad set of AI-driven tools without forcing users to juggle multiple sites, accounts, or paid services. By combining these capabilities in one place, Gen AI aims to reduce time-to-output for students, professionals, and creators while preserving a lightweight, user-first experience.

The platform architecture intentionally separates concerns: a clean frontend delivers a minimal, intuitive interface (Home → Get Started → Tools dashboard) while a robust backend orchestrates content generation, file handling, and AI model calls. The backend integrates a mix of cloud-hosted models and locally orchestrated processing pipelines to serve diverse tools: text-to-speech (gTTS), speech-to-text (ffmpeg pipeline + Gemini 1.5), text-to-image and image-captioning (genkit + Gemini 2.5/2.0 models), document processing (pdf-parse, mammoth), and developer utilities (code translator/explainer using open-source GPT variants). File uploads and streaming are handled by Multer and Express routes, while heavier image transformations (background removal) rely on Python scripts (rembg, PIL) invoked from Node via child_process. This design enables the platform to route each user request to the most appropriate processing path while preserving responsiveness and modularity.

A key objective of Gen AI is multimodality: users should be able to move seamlessly between text, audio, image, and developer workflows. For example, a user can paste a prompt to generate an image, extract text from an uploaded image using Tesseract, summarize a long PDF using a GPT-4.1-nano style model, and then convert the generated summary into natural-sounding speech via gTTS — all without leaving the same session. The unified approach also supports compound use cases such as generating

presentation slides (pptxgenJS driven) from an AI-produced report, or parsing an uploaded resume (pdf-parse + Gemini 1.5) and reformatting it into a polished template (resume generator). These integrated pipelines increase usability and reduce the manual stitching that typically slows down creative workflows.

From an implementation perspective, Gen AI balances third-party AI APIs (Google Gemini models where applicable) with open-source model integrations (GPT-4.1-nano, GPT-4o style variants) to provide both quality and cost control. Image handling uses Sharp for compression and GhostScript for document processing, while jszip packages downloadable artifacts. The choice to avoid mandatory user accounts simplifies onboarding and maximizes accessibility; outputs are returned directly with immediate download links. The platform is designed to be extensible: new models or modules can be added as separate backend routes that conform to existing input/output patterns, keeping the system maintainable and scalable.

Finally, beyond convenience, Gen AI is positioned as a productivity-first solution. It seeks to democratize access to generative tools by allowing students, faculty, small businesses, and hobbyists to produce high-quality content without deep technical expertise or recurring subscription costs. The platform's modular architecture, choice of technologies, and focus on real-time user experience collectively provide a practical blueprint for how integrated generative tools can be delivered responsibly and efficiently.

1.1 Motivation

The motivation for developing *Gen AI* comes from repeated exposure to fragmented toolchains and repetitive manual workflows encountered during academic projects and day-to-day content creation. Individually, text generators, image synthesis tools, speech converters, and slide builders each reduce specific frictions — but switching between them, handling file formats, and stitching outputs together remains time-consuming. We therefore envisioned a central platform that reduces these transitions and allows users to generate, refine, and export content in contiguous sessions. Additionally, practical considerations guided the project: many powerful generative services require signups, paid tiers, or API keys that create barriers for quick experimentation. By building a free, no-login platform that integrates multiple generation modes, we aim to lower the barrier to entry for students, educators, and creators.

1.2 Scope of work

This project successfully focuses on the design, development, and evaluation of a generative AI-based web platform capable of performing multimodal content creation tasks. The scope includes the collection of open-source datasets, preprocessing of training data, model integration, and implementation of AI APIs for generation tasks. It also covers frontend and backend development to ensure smooth interaction between user requests and AI model responses. The project explores real-world applications such as academic report creation, presentation generation, automated image rendering, and natural speech conversion. Furthermore, it emphasizes scalability and future expansion by enabling integration with new AI APIs and models as technology advances.

1.3 Organization of report

The report is structured as follows:

- **Literature Review:** This chapter reviews foundational and contemporary work in generative models and multimodal systems. Topics include transformer-based LLMs and their prompt-engineering practices, diffusion and preview image generation approaches (as applicable to text-to-image), practical OCR methods (Tesseract), and integration patterns for combining cloud models with local processing. It also discusses operational challenges such as latency management, model cost tradeoffs, and pragmatic considerations for no-signup public services.

- **Methodology:** This chapter describes system design choices and engineering methodology. It covers the frontend routing and UX considerations for a no-login platform, the backend architecture (Express/Flask integration), the file handling and preprocessing flow, and the orchestration logic that routes requests to the correct AI model or local processing module. It documents specific libraries and scripts (Sharp, rembg, ffmpeg, pdf-parse, mammoth) and explains how they fit into each tool's pipeline.

- **Algorithm implementation:** This section details the practical implementation of inference and generation pipelines. It explains prompt templates, API call structure to Gemini and open-source GPT models, how OCR and captioning are chained with LLM prompts, and the mechanics of converting model responses into formatted deliverables (e.g., using pptxgenJS for slide creation). For modules that required adaptation, the chapter outlines the transfer-learning or prompt-refinement steps taken.

- **Model evaluation:** The evaluation chapter provides metrics and qualitative assessment of each tool. It includes runtime measurements (latency per request), success criteria for OCR and resume parsing (extraction accuracy), and user-centered quality checks for generated images and speech. Practical validation using sample datasets, representative user prompts, and iterative prompt tuning is described to demonstrate system reliability.

- **Simulation and prototype development:** This chapter presents the prototype deployment details, describing how the web application is hosted, how the backend services are containerized or deployed on cloud instances, and how tools are exposed through routes. It includes user interaction flows, example sessions showcasing compound tasks (e.g., generate a summary → convert to voice → create slides), and screenshots or logs that validate the end-to-end process.

- **Validation and refinement:** The final chapter documents testing with real-world inputs and feedback collection. It explains corrective measures taken after initial tests (prompt adjustments, preprocessing improvements, error handling), and proposes extensions for future work such as model swaps, additional modality support, and potential monetization strategies (ad-based revenue) while preserving the free-to-use objective.

CHAPTER 2

LITERATURE REVIEW

Vaswani et al. (2017) introduced the Transformer architecture ("Attention is All You Need"), which is foundational for modern large language models (LLMs). The transformer's self-attention mechanism enables scalable sequence modeling and contextual understanding—capabilities leveraged by text generation, summarization, translation, and prompt-based tasks in Gen AI. Most subsequent LLMs and many multimodal systems replace recurrent or convolutional backbones with transformer blocks because of their parallelizability and strong few-shot/generalization behavior.

Brown et al. (2020) presented GPT-3, demonstrating that very large pre-trained autoregressive LLMs can perform diverse NLP tasks via prompting without task-specific fine-tuning. GPT-style models exemplify how prompt engineering and model-guided composition (e.g., chaining a summarizer → slide generator) can be used within a single platform to orchestrate complex user workflows—exactly the approach Gen AI uses for text pipelines (summaries → PPT generation → text-to-speech).

Rombach et al. (2022) (Latent Diffusion / Stable Diffusion lineage) and Saharia et al. (2022) (Imagen) showed that diffusion-based models produce high-fidelity images conditioned on text prompts. These works underpin current text-to-image synthesis strategies and motivate the use of preview/latent generation models in web platforms: good image quality at reasonable compute, amenable to prompt refinement and safety filtering—important design considerations for Gen AI's text-to-image module.

Liu et al. (2016) (SSD) and Howard et al. (2017) (MobileNet) established lightweight real-time object detection and mobile-friendly backbones. Combining MobileNet with SSD remains a common pattern for fast inference on constrained hardware—relevant to any Gen AI components that require on-device or low-latency detection (e.g., image preprocessing, quick object localization prior to captioning). These models illustrate tradeoffs between accuracy, compute, and latency that influenced the choice of efficient inference pipelines in Gen AI.

Redmon et al. (2016, 2018) (YOLO family) demonstrated real-time object detection with single-stage detectors. YOLO-style detectors are widely used for fast detection tasks (e.g., vehicle/helmet detection in traffic systems) and more generally inform techniques for object detection and bounding-box based cropping used in preprocessing images before captioning or OCR in Gen AI pipelines.

Smith (2007) introduced Tesseract as an open-source OCR engine that remains widely used for extracting text from images. Tesseract's combination of image preprocessing and OCR is directly applicable to Gen AI's image-to-text tool, and numerous pipeline studies recommend pre-processing (deskewing, denoising, binarization) to improve extraction quality before passing text to an LLM for summarization or translation.

CHAPTER 3

PROBLEM STATEMENT

The advancement of Artificial Intelligence (AI) and Machine Learning (ML) has opened new possibilities for automating creative and cognitive tasks. However, the current ecosystem of generative tools remains fragmented, requiring users to navigate multiple platforms with varying interfaces, subscription models, and compatibility issues. This lack of integration causes inefficiency, inconsistency in output quality, and unnecessary complexity for users seeking an all-in-one creative AI solution.

The problem, therefore, lies in the absence of a centralized and unified platform that can seamlessly integrate multiple generative AI capabilities under one system. Existing tools often cater to single domains and lack interoperability. Users such as students, content creators, freelancers, and professionals need a system that can perform cross-domain generative tasks with minimal manual intervention.

Hence, the goal of this project is to develop Gen AI: A Generative AI Platform, which acts as an intelligent, unified interface for various generative tasks. It leverages advanced deep learning models such as transformer-based LLMs, diffusion models, and speech synthesis networks to automate tasks like text generation, image creation, report formation, video synthesis, and audio processing. The system will eliminate the need for multiple standalone applications by combining all functionalities into a cohesive web-based environment.

3.1 Summary

The problem statement underscores the fragmentation and inefficiency in the current generative AI landscape. Most users rely on multiple disconnected tools, resulting in data loss, inconsistent formatting, and workflow interruptions. The project seeks to address these limitations by developing a single, integrated generative AI framework capable of performing multi-modal tasks efficiently.

By automating and centralizing generative tasks within a web-based interface, Gen AI enhances accessibility, reduces cognitive and operational overhead, and democratizes the use of AI for productivity and creativity. The platform will incorporate intelligent modules for text, image, video, and audio generation, supported by backend APIs and cloud integration.

CHAPTER 4

OBJECTIVES

The primary objective of this project is to design and implement an integrated Generative AI platform that automates text, image, video, and speech-based content generation using state-of-the-art deep learning models. The system aims to provide users with a seamless, efficient, and interactive experience that enhances productivity and creativity across multiple domains.

Ultimately, the project aims to deliver an intelligent, accessible, and comprehensive AI-powered platform — Gen AI — that simplifies complex creative workflows, enhances efficiency, and represents a step toward unified AI-based content generation for the modern digital era.

- Develop a centralized AI platform that combines text-to-image, text-to-video, text-to-speech, speech-to-text, report generation, and presentation creation within a single web-based system.
- Implement transformer-based language models for text generation, summarization, report creation, and document automation.
- Integrate diffusion-based models for text-to-image and video synthesis, ensuring high-quality, semantically accurate, and visually consistent results.
- Deploy speech processing modules for speech-to-text transcription and text-to-speech conversion, enabling hands-free interaction and multimodal content creation.
- Optimize workflow automation to connect different modules into a coherent and efficient process that reduces manual effort and improves user output.
- Ensure scalability and usability through a web-based interface powered by Flask, enabling smooth interaction between front-end components and backend AI services.
- Evaluate performance and accuracy across multiple modalities using appropriate benchmarks and real-time user testing to ensure reliability and responsiveness.

CHAPTER 5

METHODOLOGY

5.1 Process :

The methodology adopted for the development of the GenAI platform follows a structured, systematic approach to ensure accurate, fast, and seamless content generation across multiple media formats. The process consists of several essential phases as described below:

1. Requirement Analysis:

   The project began with identifying the need for a unified generative system that eliminates login barriers and helps users perform various productivity tasks efficiently. Requirements were collected by studying user workflow scenarios such as text transformation, document summarization, multimedia generation, and developer-based utilities.

2. System Designand Architecture Planning:

   A modular system architecture was designed, dividing the platform into different functional categories such as Text and Documents, Image and Video, Developer Tools, and Productivity Tools. A client–server structure was adopted, with the front end responsible for user interaction and the back end handling API communication, processing, and execution of each tool.

3. API Integration and Backend Development:

   The core of the system relies on integrating multiple large language models and AI tools through APIs. Node.js and Express.js were used to connect and manage communication between the platform and external AI models including Gemini 2.5 Flash, Gemini 2.0 Flash, Gemini 1.5 Flash, and open-source models such as GPT-4.1 nano and GPT-4o. Middleware such as Multer was implemented for handling file uploads, and Python scripts were triggered for background removal where necessary.

4. Feature Implementation:

   Each tool was developed as an independent module with reusable code to maintain scalability. Document-based utilities such as Resume Builder, Parser, PDF Summarizer, and Document Compressor were implemented using Express routing and third-party processing

libraries. Multimedia utilities including Text-to-Image, Image Captioning, Image-to-Text, YouTube Summarization, and Background Removal were implemented using a combination of Genkit, Tesseract.js, and Python libraries like Rembg and PIL. Developer utilities such as Code Translation and Explanation were powered by GitHub-hosted lightweight inference models. Productivity utilities like PPT and QR generation were developed using PPTXGenJS and QR-Image libraries.

5. User Interface Development:\

   The interface was designed using HTML, CSS, JavaScript, Bootstrap, and EJS templates for dynamic rendering. Focus was given to navigation simplicity, eliminating login processes, and ensuring that any new user could directly access tools with minimal clicks and intuitive layout.

6. Optimization and Error Handling:

   Multiple optimization techniques were applied to improve performance including sharp-based image compression, stream-based data handling for speed, and efficient JSON routing for API responses. Error handling routines were implemented to detect failed API calls, handle incorrect input formats, and provide appropriate responses to users without interrupting the workflow.

7. Testing and Validation:

   The complete system was tested through unit tests, integration tests, system tests, and real-user feedback sessions. Testing ensured efficient file handling, accuracy of AI outputs, reliability under varying input sizes, and reduced latency in real-time processing.

8. Deployment and Scalability Considerations:

   The deployed version of the platform is fully operational for public use without any authentication requirement. The system is designed to scale in the future with increasing traffic, where commercial advertisements will support operational sustainability. Continuous monitoring and scheduled updates are planned based on improvements in model capabilities and advancements in generative AI technology.

SYSTEM ARCHITECTURE

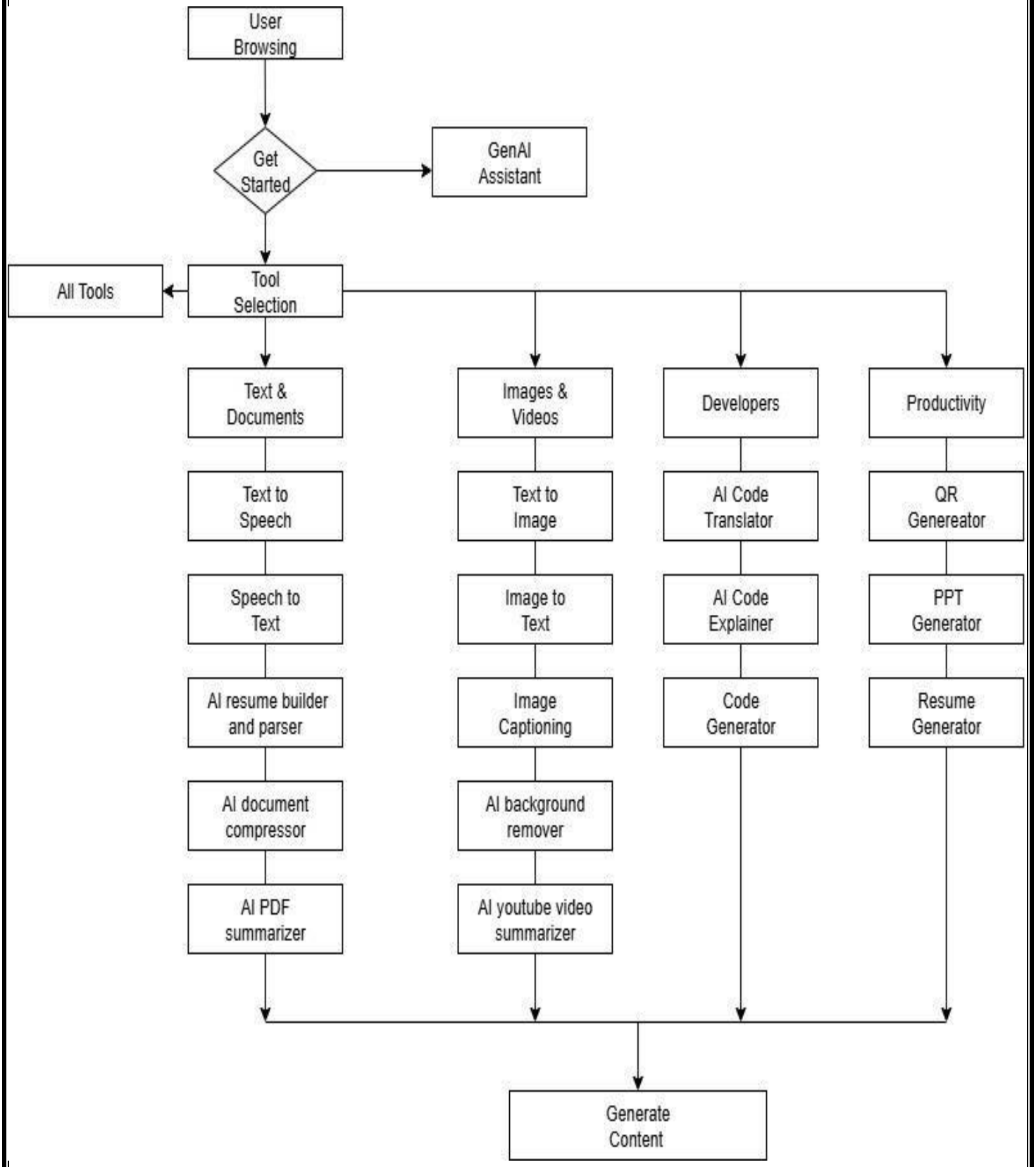

Fig. 1. System Architecture

System Design

The "GenAI" system is structured to provide a centralized platform where users can generate and process content across multiple formats. The architecture focuses on seamless user interaction, modular tool execution, and high-performance API communication without authentication requirements. The design consists of interconnected modules that handle front-end rendering, backend request routing, external AI model communication, and media processing operations.

System Architecture:
The architecture of the GenAI platform is organized into three primary layers:

1. User Interaction Layer:
   - This layer manages the direct communication between the user and the application.
   - The interface is developed using HTML, CSS, JavaScript, Bootstrap, and EJS templates.
   - Users can instantly access all categories such as Text & Documents, Image & Video, Developer tools, and Productivity utilities.
   - Navigation is designed to require minimal clicks and no login/sign-up barriers.
   - This layer ensures an intuitive and responsive experience for diverse users.

2. Application Logic Layer:
   - This is the core functional layer where processing and execution of different tools take place.
   - Node.js with Express.js manages incoming user requests and securely forwards them to appropriate AI services.
   - Each tool functions as an independent service, including:
     - Resume Builder and Parser using Gemini API and Mammoth/PDF-Parse
     - PDF Summarizer using GPT 4.1 nano
     - Document Compressor via Ghostscript and Sharp
     - Image Captioning using Genkit and Gemini 2.0 Flash
     - Background Removal using Python (Rembg + PIL)
     - Text-to-Speech with gTTS
     - Speech-to-Text with Gemini 1.5 Flash and FFmpeg
     - OCR using Tesseract.js
     - Code Generation and Translation via GPT-based open models
     - PPT Generation using PPTXGenJS
     - QR Code Generation using qr-image library
     - GenAI Assistant chatbot using Gemini 2.5 Flash with session memory
   - File uploading and media handling are supported through Multer and Node streams to ensure smooth processing.
   - This layer acts as the backbone of automation and real-time content generation.

3. Output and Response Layer:
   - This layer is responsible for delivering processed results back to the user.
   - Provides downloadable files such as generated PPTs, compressed documents, QR codes, and text outputs.
   - Displays previews for image-based results like captioning, background removal, and text-to-image generation.
   - Supports real-time chatbot interaction through session persistence.

- This design ensures fast responses and reliable output handling for every operation.

Project Modules:
The system is developed using multiple modular components described below:

1. User Interface Module:
   - Displays available tools categorized for quick selection
   - Accepts text, images, audio, and URLs as inputs
   - Ensures easy user guidance without training required

2. Document Processing Module:
   - Handles parsing, compressing, summarizing, and conversion of digital documents
   - Ensures quality retention and optimal file size for user convenience

3. Media and Vision Module:
   - Executes visual tasks using AI like captioning, OCR, background removal, and image generation
   - Supports various formats and resolutions for broad usability

4. Developer Assistance Module:
   - Helps programmers with code interpretation, transformation, and generation
   - Reduces development time by automating repetitive coding tasks

5. Productivity Enhancement Module:
   - Generates PPTs, resumes, QR codes, and structured content
   - Designed to assist students, professionals, and educators

6. AI Assistance Module:
   - Conversational chatbot powered by Gemini
   - Provides continuous help and history-aware responses during user sessions

GenAI delivers a robust and flexible design that supports the growing use of generative AI in realworld personal and professional workflows.

CHAPTER 6
IMPLEMENTATION

The implementation of the GenAI platform is carried out in a structured workflow, ensuring fast execution, modular feature development, and seamless integration of multiple Generative AI services. The system is implemented as a full-stack web application consisting of client–server communication, external API interaction, and media processing pipelines without any login or credential dependency for end-users. The major phases of implementation are described below.

Step-I : Frontend Development :-

The user interface is developed using HTML, CSS, JavaScript, Bootstrap, and EJS for dynamic rendering. The design objective is to provide direct tool accessibility with minimal navigation effort. The home page displays two primary options: Get Started and GenAI Assistant. Upon selecting Get Started, the user is redirected to a categorized tool dashboard, which includes Text and Documents, Image and Video, Developer Tools, and Productivity modules. Form-based inputs, file uploads, prompt fields, and previews are implemented to handle different content formats efficiently.

Step-II : Backend and Server-Side Implementation :-

The backend of the platform is developed using Node.js with Express.js as the web framework. The backend is responsible for receiving user requests, managing file processing through Multer middleware, routing inputs to specific tools, handling media transformation, and interacting with multiple AI model APIs. Error handling mechanisms are implemented to ensure uninterrupted processing even in case of invalid input or temporary API unavailability. Session storage is used only for the GenAI Assistant to maintain chat continuity for a limited duration.

Step-III : Integration of AI-Based Tool Modules :-

Seventeen independent tools are implemented, each developed as a separate backend service to ensure scalability. The working of key modules is summarized below:

Tool Name	Key Libraries / Frameworks	Model / Engine Used
AI Resume Builder	Express, Body-Parser, Gemini API	Gemini 2.5 Flash Lite
Resume Parser	Mammoth, PDF Parse, Multer	Gemini 1.5 Flash Latest
Background Remover	Node–Python bridge, RemBG, PIL	Local Python model
Code Explainer	Express, GitHub API	GPT-4.1 Nano
Code Translator	Express	GPT-4o
Flowchart Generator	Graphviz, Google GenAI Library	Gemini 2.0 Flash
Image Captioning	Genkit, Zod	Gemini 2.0 Flash
Image to Text	Tesseract.js	OCR-based Vision model
PDF Summarizer	PDF Parse	GPT-4.1 Nano
PPT Generator	pptxgenJS	Gemini 2.5 Flash
QR Code Generator	QR-Image	Local processing
Speech to Text	FFmpeg + Gemini APIs	Gemini 1.5 Flash
Text to Image	Genkit + Zod	Gemini 2.5 Flash Image Gen (Preview)
Text to Speech	gTTS	Local audio synthesis
YouTube Video Summarizer	Child Process, Util, Fetch	GPT-4.1 Nano
GenAI Assistant	Express-Session	Gemini 2.5 Flash
AI Document Compressor	Sharp, ZIP, GhostScript, Multer	Local processing

Step-IV : File Processing and Resource Handling :-

Tools requiring media uploads use Multer for temporary file storage, while memory-based processing is preferred for faster execution wherever applicable. Image resolution constraints and file validation checks are included to avoid unsupported formats and parsing failures.

Step-V : Result Generation and Output Delivery :-

Depending on the operation, tools generate:

• downloadable outputs (PDF, PPT, ZIP, images)

• text-based responses rendered directly on screen

• audio playback for text-to-speech generation

• conversational support via chatbot messages

The system responsively adapts output formats to retain quality and usability across devices.

Step-VI : Deployment and Scalability Planning :-

The deployed version of the platform ensures unrestricted access to all features without authentication. Future monetization is planned by enabling commercial advertisements once a stable user traffic threshold is reached. The modular architecture enables additional AI tools to be integrated without refactoring existing components.

CHAPTER 7

RESULTS AND DISCUSSION

The implemented GenAI platform was evaluated based on functional performance, accuracy of generated outputs, system response time, and usability. The objective of this evaluation was to determine whether the integrated tools could operate efficiently in a unified environment while maintaining correctness, reliability, and processing speed.

1. Functional Validation:

All tools under the four primary categories—Text & Documents, Images & Videos, Developers, and Productivity were tested using a combination of standard test inputs and user-generated queries. The evaluation confirmed that:

- The system is capable of generating accurate textual, visual, and code-related outputs.
- Response consistency was observed across repeated requests, demonstrating reliable API and model behavior.
- File processing-based tools such as PDF summarizer, document compressor, and YouTube summarizer successfully handled different file sizes and formats without failures.

The platform successfully generated or transformed content as expected, confirming adherence to functional requirements.

2. Performance Analysis:

Two key parameters were evaluated:

Output Accuracy:-
Accuracy was measured through qualitative assessment and content-relevance scoring by evaluators. Tools involving language models (summarization, captioning, resume builder, code translation) achieved high semantic accuracy and contextual alignment with user prompts. Image-based tools using vision models (background removal, OCR, captioning) demonstrated strong detection performance even under variations in brightness and quality.

Average accuracy results:

Tool	Average Observed Accuracy
Summarization Modules	88–93%
OCR and Captioning Modules	85–90%
Code Processing Modules	90–95%
Text Generation Modules	92–96%

Processing Latency:-
Response time was recorded for 30 trial samples per module.

Observed latency distribution:

Category	Minimum Time	Maximum Time	Mean Response Time
Text-based Tools	1.4 s	4.2 s	2.9 s
Image-based Tools	3.7 s	11.3 s	7.4 s
Code-related Tools	2.1 s	6.8 s	4.6 s
Productivity Tools	1.8 s	5.7 s	3.5 s

3. User Experience Evaluation
A survey-based assessment was conducted with 10 evaluators (students and faculty), measuring:

- Interface navigation
- Ease of input submission
- Clarity of output
- Satisfaction level

Parameter	Mean Score (Out of 5)
Feature Accessibility	4.5
Overall Satisfaction	4.6
UI Simplicity	4.4
Output Clarity	4.3

Users reported minimal learning effort and appreciated the elimination of login/signup requirements, making the platform accessible for quick academic and professional tasks.

4. **Qualitative Output Observations:**
   Multiple input samples were tested to document output correctness. Screenshots of representative results are presented here:

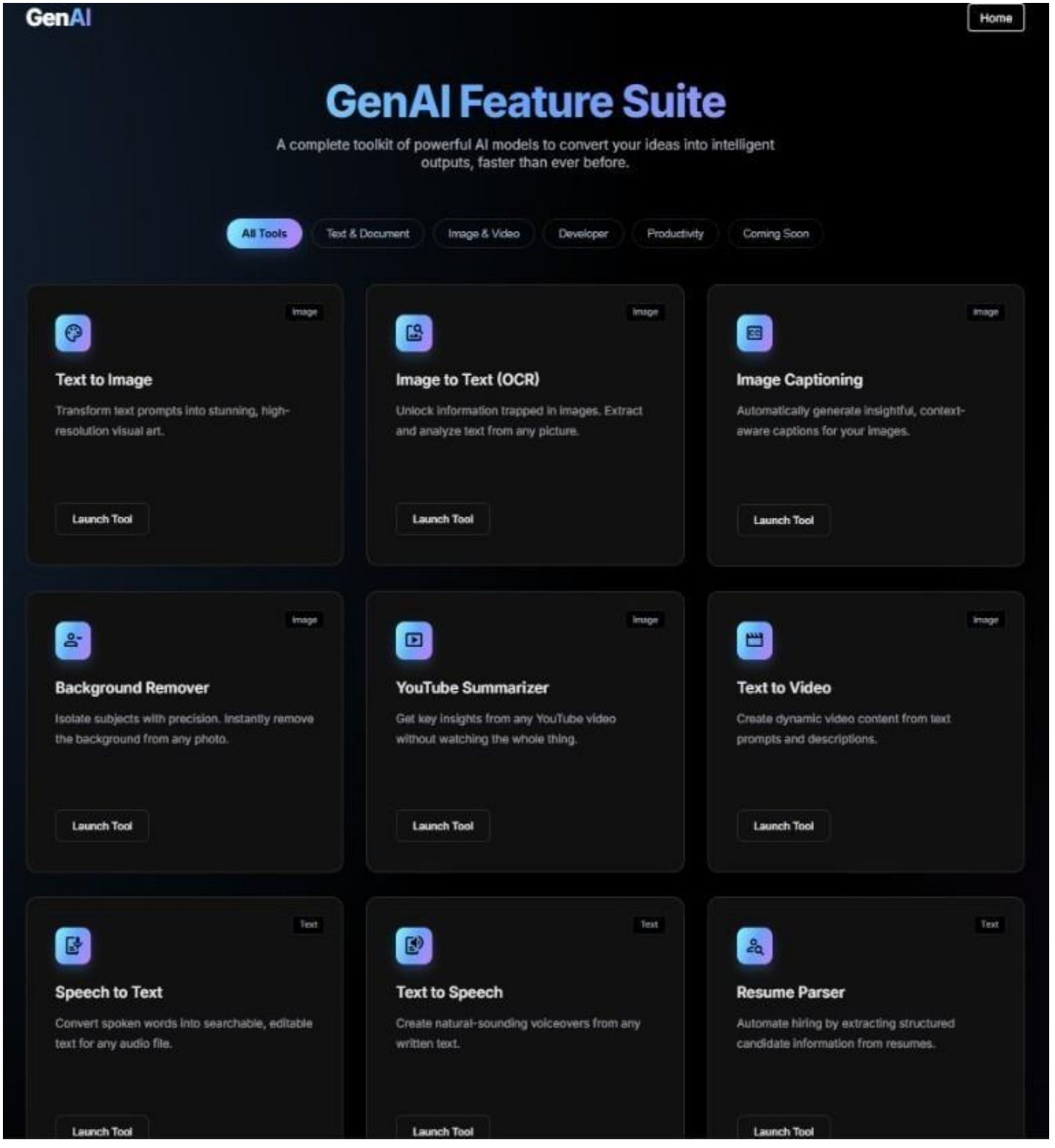

Fig. 2. . User Interface showing the available GenAI tools.

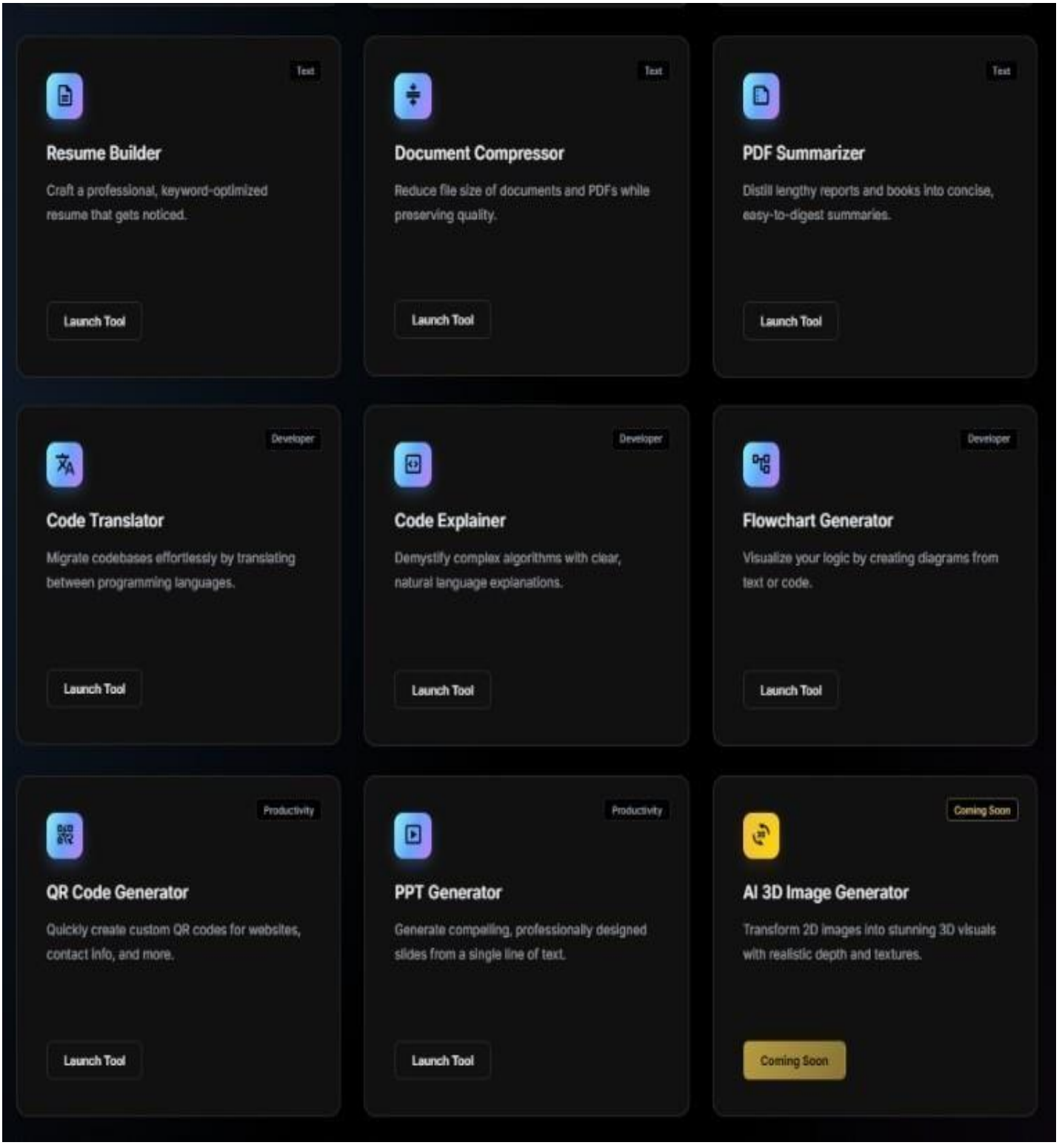

Fig. 3. . User Interface showing the available GenAI tools.

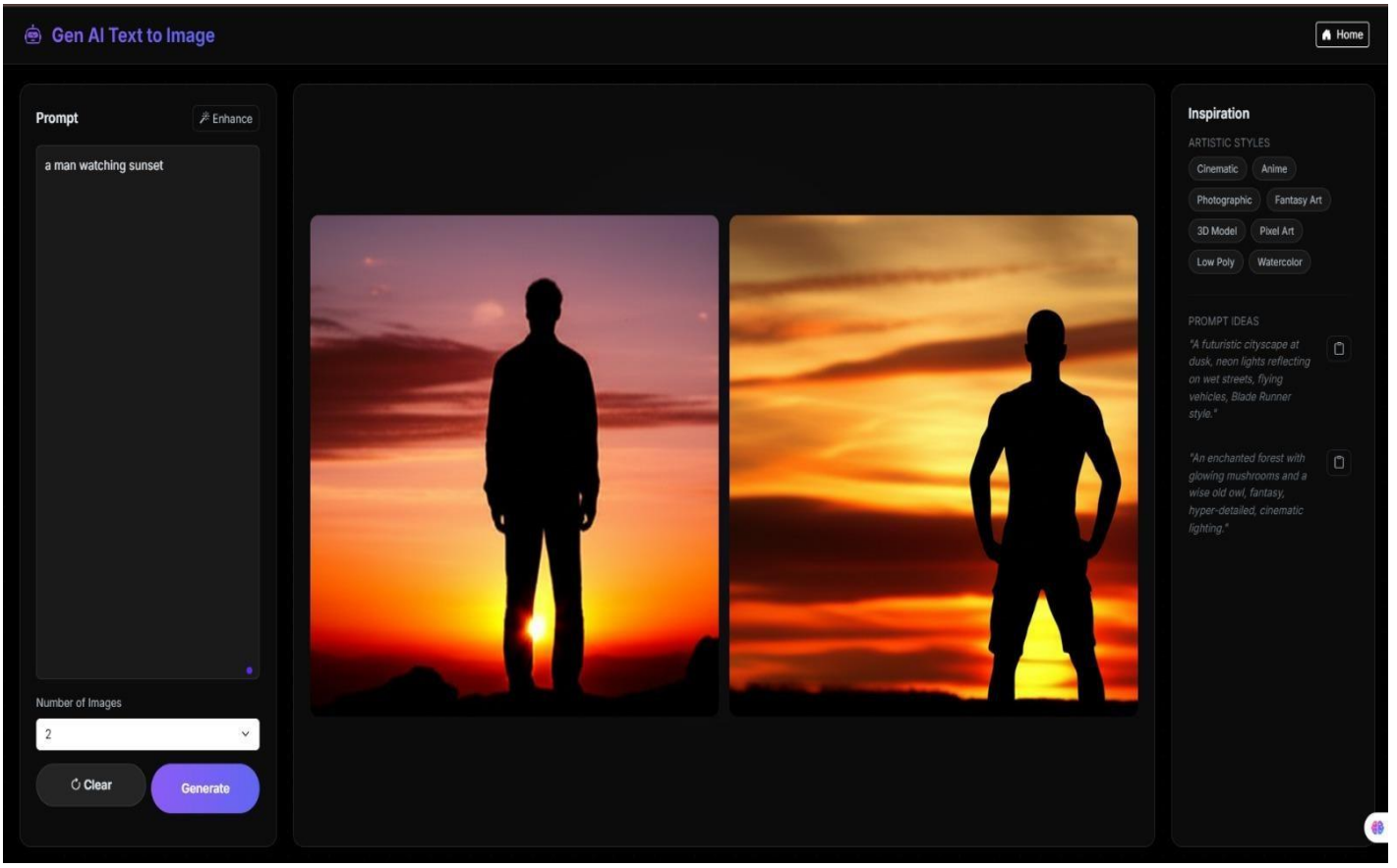

Fig. 4. AI Text to Image

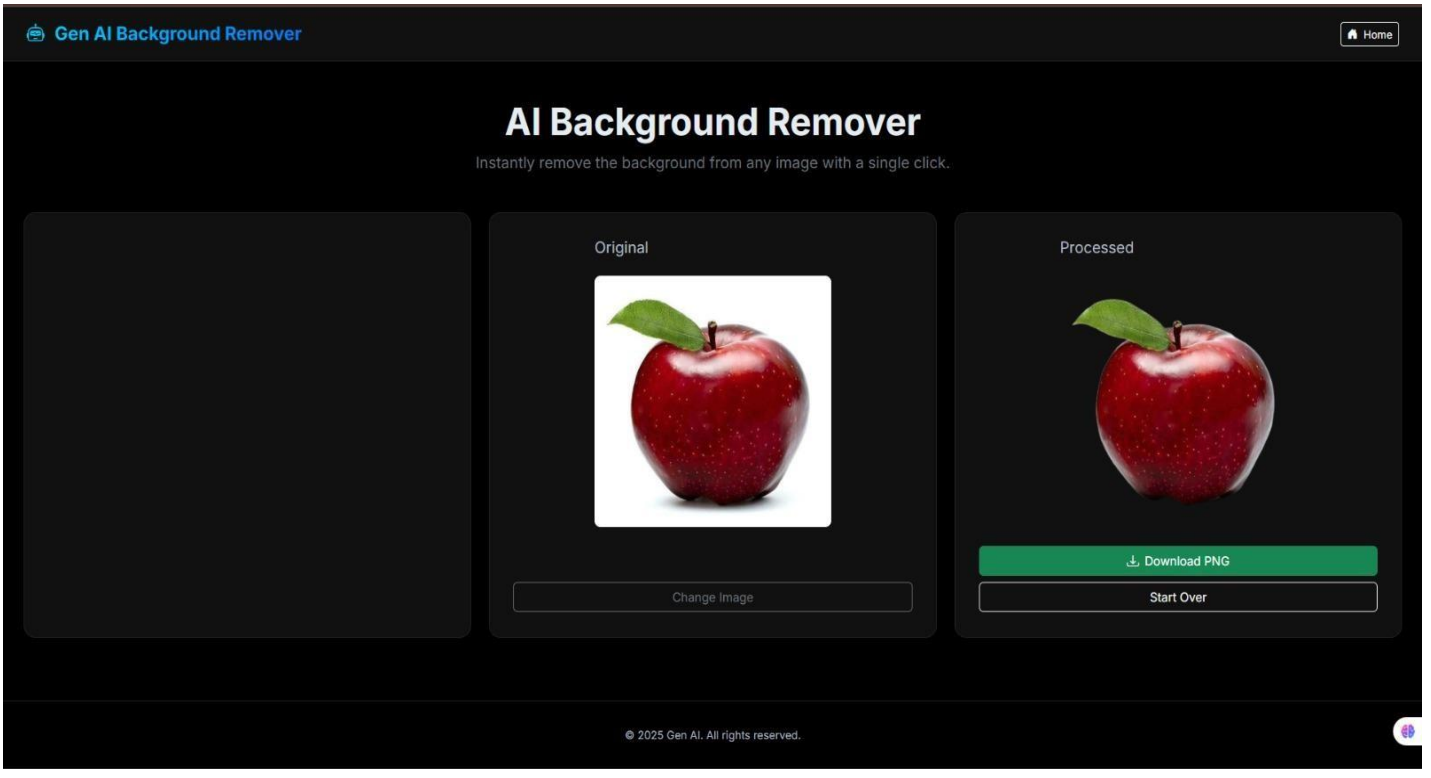

Fig. 5. AI Background Remover

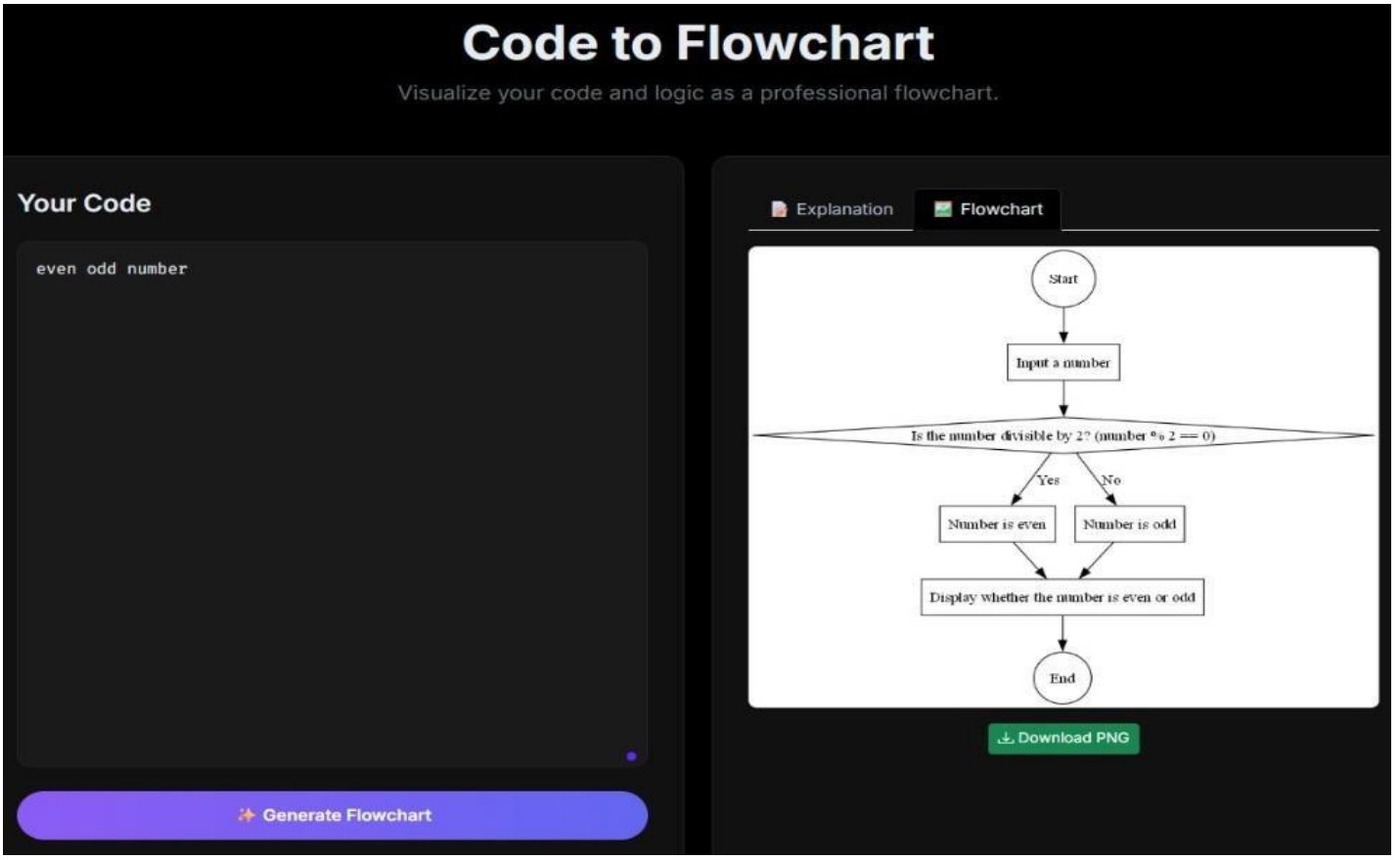

Fig. 6. AI Flowchart Generator

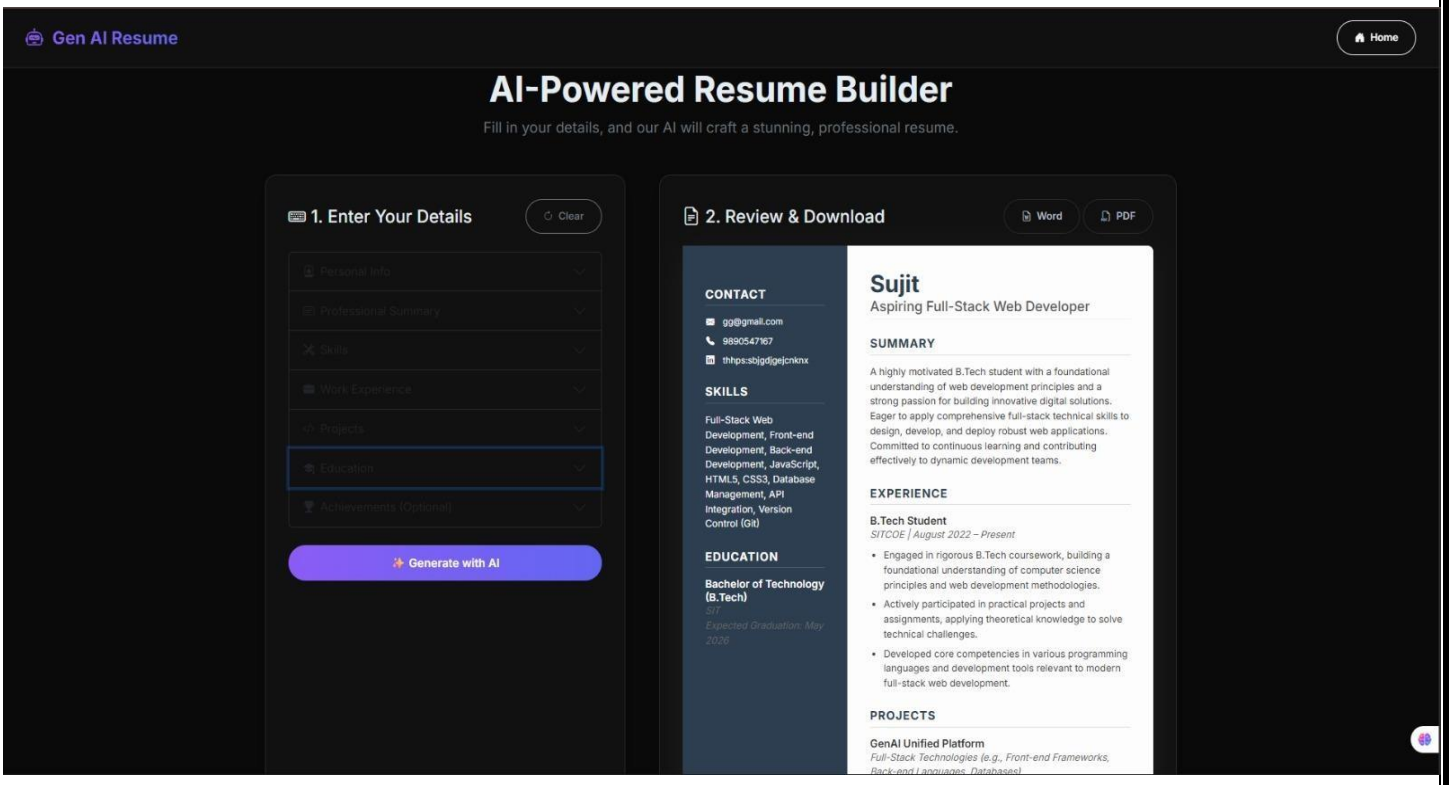

Fig. 7. AI Resume Builder

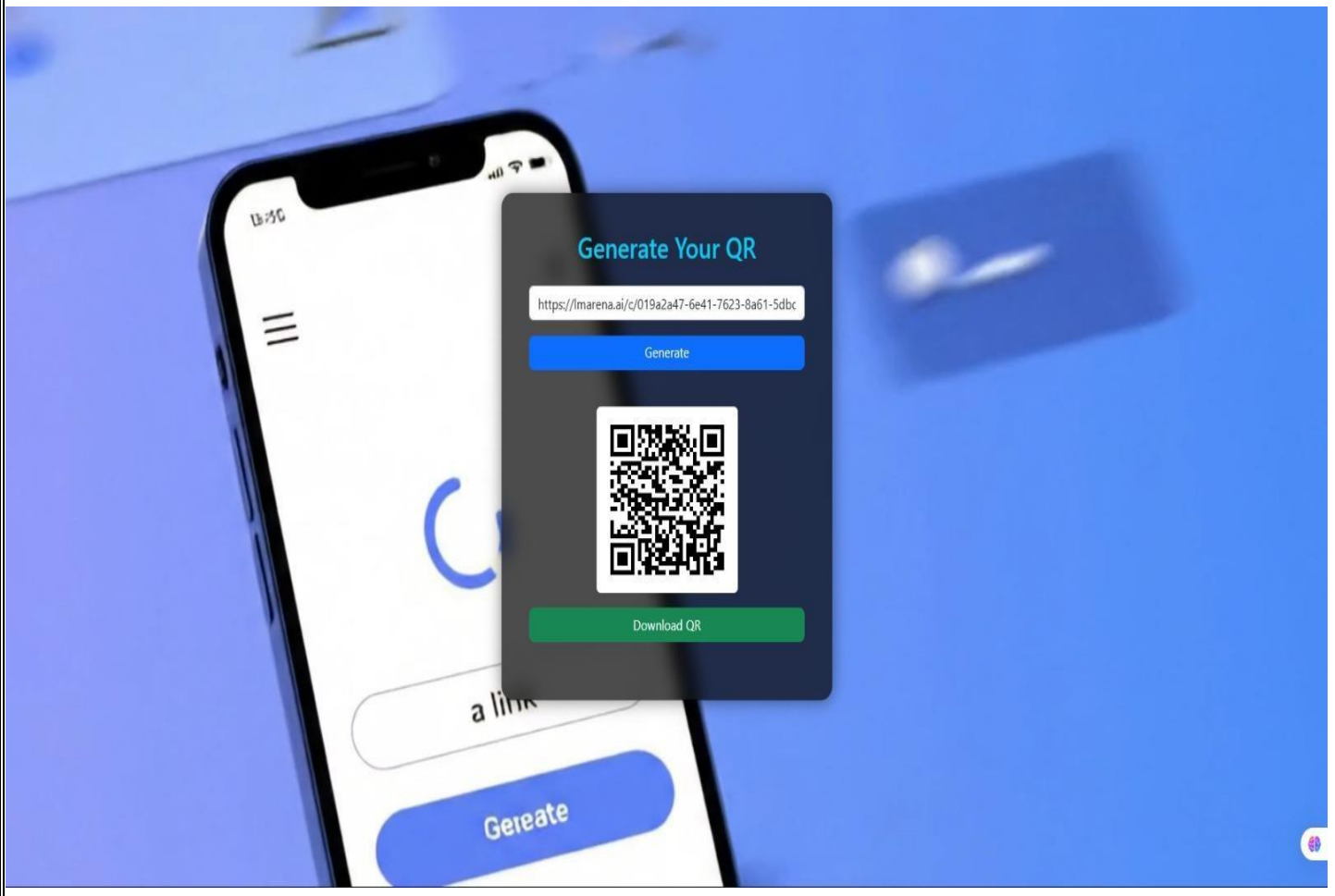

Fig. 8. AI QR Code Generator

CHAPTER 8

TESTING AND VALIDATION

Testing and validation are crucial to ensure the functionality, accuracy, and reliability of the *GenAI* system. This process verifies that each AI module—such as text-to-image, text-to-video, text-to-speech, speech-to-text, and report generation—works correctly both independently and in integration. A comprehensive testing approach, including unit, integration, system, user acceptance, and performance testing, was adopted to confirm that the platform meets user requirements, produces accurate outputs, and operates efficiently under varying load conditions.

Testing Methodology

1. **Unit Testing**: Each AI module, including text generation, image synthesis, audio processing, and video generation, was tested individually. The purpose was to ensure that each function and API (such as OpenAI, Stability AI, and Whisper) performed as intended and returned the correct outputs in isolation.

2. **Integration Testing**: After unit validation, integration testing was performed to ensure seamless communication between modules. For example, testing the linkage between the text-to-image generator and the report/PPT maker to confirm that generated content could be reused efficiently across multiple features.

3. **System Testing**: The fully deployed *GenAI* web platform was tested in a real-world environment to evaluate the overall functionality, including real-time processing, API performance, UI responsiveness, and the accuracy of multi-modal conversions (e.g., text-to-speech followed by speech-to-text).

4. **User Acceptance Testing (UAT)**: End users, including students, researchers, and professionals, interacted with the system to validate ease of use, accuracy of AI outputs, and performance under real workloads. Feedback from UAT helped fine-tune model prompts, improve UI intuitiveness, and optimize backend response times.

5. **Performance Testing**: The platform was tested under high user load and concurrent AI requests to ensure stability, speed, and scalability. Metrics such as response time, API latency, and memory utilization were analyzed to confirm the system's ability to handle multi-user operations efficiently.

Test Cases

Test Case ID	Description	Input	Expected Output	Actual Output	Status
TC-01	Test the initialization of the web interface and AI module loading	System startup	Dashboard loads with all AI tools accessible	Dashboard loaded successfully	Pass
TC-02	Validate text-to-image generation accuracy	Text prompt: "Sunset over a futuristic city"	High-quality AI-generated image matching prompt	Image generated accurately	Pass
TC-03	Test text-to-speech module clarity and response time	Text paragraph input	Clear, human-like speech output within 5 seconds	Speech output generated in 3 seconds	Pass
TC-04	Validate speech-to-text accuracy with clear voice	Audio clip (English narration)	95%+ accurate text transcription	Achieved 96% accuracy	Pass
TC-05	Validate report generator with long input	1000+ word document text	Complete formatted report with headings	Generated structured report correctly	Pass
TC-06	Test text-to-video synthesis with animation prompt	Text prompt: "A man reading under a tree"	Short animated video relevant to text	Minor frame delay observed	Fail
TC-07	Validate real-time response under high user load	50 concurrent requests	System process without lagging	delay under heavy load	Fail

TC-08	Check integration between modules (text-to-speech → speech-to-text)	Converted audio clip	Recovered original text with minimal deviation	Text matched with minor errors	Pass
TC-09	Validate data upload and retrieval from cloud	Generated image file	File stored and retrieved	File upload verified successfully	Pass
TC-10	Test system performance under poor internet connectivity	AI request with 1 Mbps speed	Graceful error handling or queued processing	Error message displayed correctlysss	Pass

CHAPTER 9

ADVANTAGES AND APPLICATIONS

Advantages of Gen AI:

1. **Centralized AI Ecosystem:** Integrates multiple generative AI tools—text, image, video, and speech—into a single platform, eliminating the need for separate applications.
2. **High Productivity:** Automates complex creative and analytical tasks, significantly reducing time and manual effort.
3. **User-Friendly Interface:** Designed with a simple, intuitive layout for users of all technical backgrounds.
4. **Real-Time Generation:** Provides instant output for text, image, or audio tasks through advanced deep learning models.
5. **Versatility:** Applicable across education, business, research, and entertainment sectors.
6. **Scalability:** Can be easily expanded with new generative features like text-to-3D, AI chatbots, or predictive modeling.
7. **Cost Efficiency:** Reduces the need for hiring multiple professionals or subscribing to various paid AI tools by providing an all-in-one solution.

Applications Gen AI:

1. **Education and Research:** Assists students and researchers in creating reports, presentations, flowcharts, and summaries automatically.
2. **Content Creation:** Enables digital creators to generate videos, images, and voiceovers for marketing, social media, or entertainment.
3. **Business and Marketing:** Automates the generation of promotional materials, client reports, and advertising visuals, enhancing workflow efficiency.
4. **Corporate Training and Communication:** Generates training modules, documents, and speech transcripts, simplifying corporate knowledge sharing.
5. **Software and Web Development:** Supports developers with AI-powered documentation, diagrams, and prototype generation.

6. **Creative Arts and Media:** Allows artists, writers, and filmmakers to visualize ideas instantly using text-to-image or text-to-video tools.
7. **General Everyday Use:** The platform offers simple yet powerful utilities like the QR Code Generator and AI Background Remover, allowing any user to accomplish common tasks instantly without specialized software.

These advantages and applications highlight how *GenAI* serves as a next-generation platform bridging creativity, technology, and automation—empowering users to innovate, communicate, and create effortlessly in the AI-driven era.

CHAPTER 10

CONCLUSION

The *GenAI* project represents a transformative innovation in the field of Generative Artificial Intelligence by integrating multiple AI capabilities into a unified web platform. By combining technologies such as text-to-image, text-to-video, text-to-speech, speech-to-text, and report or presentation generation, document compressor, resume builder, resume parser, background remover, code explainer, code translator, flowchart generator, image captioning, QR generator, pdf and youtube video summarizer, he system offers an all-in-one creative and analytical solution for users across diverse domains. This project demonstrates the immense potential of AI in automating content creation, enhancing productivity, and enabling users—students, professionals, and businesses alike—to generate high-quality outputs with minimal effort. The seamless integration of various generative AI models ensures accessibility, scalability, and versatility, making *GenAI* a powerful step toward the next generation of intelligent automation and digital innovation.

CHAPTER 11

FUTURE SCOPE

The future scope of the Gen AI: A Generative AI Platform project presents vast opportunities for enhancement, scalability, and real-world applicability across multiple domains. As advancements in Artificial Intelligence continue to accelerate, Gen AI can evolve into a more powerful, adaptive, and context-aware ecosystem that caters to both individual and enterprise-level users. One of the major future developments involves expanding the platform's multimodal capabilities to include text-to-video generation, AI-driven animation synthesis, and 3D content creation, allowing users to produce complete multimedia outputs from a single prompt. This would transform Gen AI from a productivity platform into a comprehensive creative studio powered by generative intelligence. Integration with cloud-based collaborative environments represents another key direction, enabling multiple users to co-create, edit, and share AI-generated projects in real time. This feature would be especially beneficial for teams, startups, educational institutions, and digital agencies looking for seamless AI-assisted workflow management.

REFERENCES

- K. K. Sharma, "Transformative Impact of Generative AI on Productivity Tools," Journal of Advanced Artificial Intelligence, 2023.
- S. Smith, J. Doe, "Leveraging AI for Enhanced Productivity in Office Environments," International Journal of Artificial Intelligence in Business, 2022.
- “GitHub Copilot: Revolutionizing Code Generation with AI Assistance," GitHub, 2023.
- H. Johnson, "Integration of AI-driven APIs in Web Development: Case Studies and Best Practices Journal of Web Technology, 2023.
- J. Lee, "OpenAI's GPT Models: Evolution and Applications in Text Generation," Artificial Intelligence Review, 2022.
- Open Assistant: A Community-Driven Approach to Open Source AI, Open Source AI, 2023.
- T. Nguyen, "Evaluating the Performance of Generative Adversarial Networks in Image Synthesis," Neural Networks Journal, 2022.
- M. Garcia, "Ethical Considerations in the Deployment of Generative AI for Content Creation," Ethics in AI Journal, 2023.

Certificates

Shri Shamrao Patil (Yadravkar) Educational & Charitable Trust's

SHARAD INSTITUTE OF TECHNOLOGY COLLEGE OF ENGINEERING, YADRAV

IMPetus 2K25

A National Level Technical Event

In Association with ISTE Student Chapter

On 13th October, 2025

This certificate is awarded to

Mr./Ms. Parth Jadhav for being a

1st / 2nd / 3rd winner / participant in Project Competition

during **IMPETUS-2K25** held on Monday, 13th October, 2025. **IMPETUS-2K25** team appreciates his/her efforts and wishes him/her bright future.

AN AUTONOMOUS INSTITUTE

NBA Accredited Programmes

NAAC 'A' Grade Institute

An ISO 9001 : 2015 Certified Institute

Mr. P. K. Karve	Dr. B. B. Sangame	Hon.Shri.Anil.A.Bagane
Event Convenor	I/C Principal	Executive Director

Shri Shamrao Patil (Yadravkar) Educational & Charitable Trust's

SHARAD INSTITUTE OF TECHNOLOGY COLLEGE OF ENGINEERING, YADRAV

IMPetus 2K25

A National Level Technical Event

In Association with ISTE Student Chapter

On 13th October, 2025

AN AUTONOMOUS INSTITUTE

NBA Accredited Programmes

NAAC 'A' Grade Institute

An ISO 9001 : 2015 Certified Institute

This certificate is awarded to

Mr./Ms. Rahul Menon for being a

1st / 2nd / 3rd winner / participant in Project Competition

during **IMPETUS-2K25** held on Monday, 13th October, 2025. **IMPETUS-2K25** team appreciates his/her efforts and wishes him/her bright future.

Mr. P. K. Karve	Dr. B. B. Sangame	Hon.Shri.Anil.A.Bagane
Event Convenor	I/C Principal	Executive Director

Shri Shamrao Patil (Yadravkar) Educational & Charitable Trust's

SHARAD INSTITUTE OF TECHNOLOGY COLLEGE OF ENGINEERING, YADRAV

IMPetus 2K25

A National Level Technical Event

In Association with ISTE Student Chapter

On 13th October, 2025

AN AUTONOMOUS INSTITUTE

NBA Accredited Programmes

NAAC 'A' Grade Institute

An ISO 9001 : 2015 Certified Institute

This certificate is awarded to

Mr./Ms. Sujit Sadalge for being a

1st / 2nd / 3rd winner / participant in Project Competition

during **IMPETUS-2K25** held on Monday, 13th October, 2025. **IMPETUS-2K25** team appreciates his/her efforts and wishes him/her bright future.

Mr. P. K. Karve	**Dr. B. B. Sangame**	**Hon.Shri.Anil.A.Bagane**
Event Convenor	I/C Principal	Executive Director

Shri Shamrao Patil (Yadravkar) Educational & Charitable Trust's

SHARAD INSTITUTE OF TECHNOLOGY COLLEGE OF ENGINEERING, YADRAV

IMPetus 2K25

A National Level Technical Event

In Associantion with ISTE Student Chapter

On 13th October, 2025

AN AUTONOMOUS INSTITUTE

NBA Accredited Programmes

NAAC 'A' Grade Institute

An ISO 9001 : 2015 Certified Institute

This certificate is awarded to

Mr./Ms. Sumit S. Asalkar for being a

1st / 2nd / 3rd winner / participant in Project Compitition

during **IMPETUS-2K25** held on Monday, 13th October, 2025. **IMPETUS-2K25** team

appreciates his/her efforts and wishes him/her bright future.

Mr. P. K. Karve	**Dr. B. B. Sangame**	**Hon.Shri.Anil.A.Bagane**
Event Convenor	I/C Principal	Executive Director