LocallyYours

A PROJECT REPORT
For
Mini Project (KCA353)
Session (2024-25)
Submitted by

Dheeraj Jadaun (2300290140052)
Akshit Bansal (2300290140018)
Apoorva Chaudhary (2300290140033)
Archit Nirwal (2300290140037)

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of

MR. Shish Pal Sir

Assistant Professor

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206
(March-2025)

DECLARATION

I hereby declare that the work presented in this report entitled “LocallyYours", was carried out by me. I have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute. I have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs,experiments, results, that are not my original contribution. I have used quotation marks to identify verbatim sentences and given credit to the original authors/sources. I affirm that noportion of my work is plagiarized, and the experiments and results reported in the report arenot manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, I shall be fully responsible and answerable.

Name: Dheeraj Jadaun
Roll. No: 2300290140052
Branch: MCA

Name: Akshit Bansal
Roll. No: 2300290140018
Branch: MCA

Name: Apoorva Chaudhary
Roll. No: 2300290140033
Branch: MCA

Name: Archit Nirwal
Roll. No: 2300290140037
Branch: MCA

CERTIFICATE

Certified that Dheeraj Jadaun (2300290140052), Akshit Bansal (2300290140018), Apoorva Chaudhary (2300290140033), Archit Nirwal (2300290140037) have carried out the project work having "**LocallyYours**" for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU),Lucknow under my supervision. The project report embodies original work, and studies arecarried out by the student herself and the contents of the project report do not form the basisfor the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Dheeraj Jadaun (2300290140052)
Akshit Bansal (2300290140018)
Apoorva Chaudhary (2300290140033)
Archit Nirwal (2300290140037)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Mr. Shish Pal
Assistant Professor
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Dr. Arun Tripathi
Head
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Abstract

The AI Content Generator is a full-stack web application designed to help users generate high-quality content using artificial intelligence. Built with React, Next.js, and Tailwind CSS, the application offers a user-friendly interface with secure authentication via Clerk. The backend is powered by PostgreSQL with Drizzle ORM, while AI-driven content generation is facilitated through Google's API. Users can select from various content templates, input relevant details, and receive generated content, all while managing usage through a credit-based system. The project demonstrates the integration of advanced technologies to provide a seamless, interactive content creation experience.

ACKNOWLEDGEMENTS

Success in life is never attained single handedly. My deepest gratitude goes to my thesis supervisor, Mr. Shish Pal (Assistant Professor) for her guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Arun Kumar Tripathi, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Dheeraj Jadaun (2300290140052)
Akshit Bansal (2300290140018)
Apoorva Chaudhary (2300290140033)
Archit Nirwal (2300290140037)

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1: Introduction

1.1 Project Description

The **Locally Yours** project is a full-stack AI-powered content generation platform that leverages cutting-edge technologies to create personalized, locally relevant content. The platform offers a user-friendly interface built using React, Next.js, and Tailwind CSS, while its backend is powered by PostgreSQL and Drizzle ORM. The project integrates Google's AI API to generate high-quality content tailored to the specific needs of users, with a particular focus on local audiences.

Locally Yours allows users to log in securely, select content templates, and input relevant details to generate content quickly and easily. With a credit-based free usage model and subscription plans for premium features, the platform is both accessible and scalable, ensuring sustainable growth.

1.2 Project Scope

The scope of the **Locally Yours** project includes:

- **Frontend**: A responsive and interactive dashboard that allows users to choose from various templates, input necessary details, and generate content.
- **Backend**: Integration with Google’s AI API for content generation and Razorpay for subscription management.
- **User Management**: User authentication via Clerk, allowing users to manage their account securely.
- **Database**: A PostgreSQL database to store user data, content templates, and generated content.

The project is designed to offer an easy-to-use platform that simplifies content generation for local businesses, bloggers, and content creators.

1.3 Future Scope

The future scope of **Locally Yours** includes:

- Expanding the range of content templates to cater to more specific local requirements.

- Integrating more advanced AI models for content refinement.
- Introducing features like collaboration tools and multi-language support.
- Scaling the platform to handle a larger user base and more advanced AI capabilities.

1.4 Identification of Need

There is a growing demand for personalized content creation, especially for local businesses and content creators who need relevant and timely content to engage their audience. Traditional content creation is often time-consuming and expensive. **Locally Yours** addresses this need by automating the content generation process using AI, providing users with an easy, cost-effective solution to generate high-quality, locally relevant content.

1.5 Problem Statement

Many businesses and individuals struggle to create high-quality content that resonates with their target audience. The lack of easy-to-use tools that can generate local, relevant content efficiently presents a significant challenge. **Locally Yours** aims to solve this problem by offering a web platform that automates content creation, making it more accessible to users of all technical backgrounds.

1.6 Software/Technology Used in Project

The following technologies and tools were used in the development of **Locally Yours**:

- **Frontend**: React, Next.js, Tailwind CSS
- **Backend**: Node.js, Express.js
- **Database**: PostgreSQL with Drizzle ORM
- **AI Integration**: Google AI API for content generation
- **User Authentication**: Clerk for secure user login
- **Payment Integration**: Razorpay for subscription and credit management

1.6.1 Non-Functional Requirement

- **Performance**: The platform must handle high volumes of requests, ensuring fast content generation.
- **Scalability**: The system must be able to scale efficiently to accommodate growing user demand.

- **Security**: The platform must securely store user data and ensure safe transactions.
- **Usability**: The user interface must be intuitive and easy to navigate, even for non-technical users.

1.6.2 Functional Requirement

- **User Authentication**: Secure login and registration system.
- **Template Selection**: Users can choose from multiple content templates.
- **Content Generation**: The system generates content based on user input via the Google AI API.
- **Subscription Management**: Razorpay integration for managing credits and premium subscriptions.

1.7 Project Schedule

The project was developed from **August 5, 2024**, to **November 10, 2024**, with clear milestones and timelines for each phase, including requirements gathering, system design, development, testing, and deployment.

1.7.1 PERT Chart

A PERT (Program Evaluation and Review Technique) chart is used to visually represent the project schedule, identifying the tasks that must be completed and their dependencies.

1.7.2 Gantt Chart

A Gantt chart is used to illustrate the timeline of the project, breaking down the development stages into manageable tasks with start and end dates.

Chapter 2: Feasibility Study

2.1 Introduction

The feasibility study evaluates whether the **Locally Yours** project can be successfully implemented from a technical, economic, and operational perspective. It assesses the project's viability by considering various factors like available resources, potential revenue, and operational efficiency.

2.2 Main Aspects

- **Technical Feasibility**: The project uses a modern technology stack, ensuring that all components (frontend, backend, AI integration, and database) are feasible within the scope of the project.
- **Economic Feasibility**: The platform is designed with a cost-effective approach, utilizing free and open-source technologies where possible. The revenue model based on subscription plans ensures economic sustainability.
- **Operational Feasibility**: The system is easy to use, scalable, and maintainable, ensuring that the platform can grow with increasing demand.

2.2.1 Technical Feasibility

The technology stack for **Locally Yours** (React, Next.js, Tailwind CSS, PostgreSQL, Google AI API) is widely adopted and has been proven to work well in similar projects. The backend infrastructure supports scalability and ensures smooth interaction with the database and AI API.

2.2.2 Economic Feasibility

The project has a low initial cost due to the use of open-source technologies. Revenue will be generated through a subscription-based model, with free usage options that limit access to premium features. The operational costs are manageable, mainly involving cloud hosting services and API usage.

2.2.3 Operational Feasibility

The platform is designed for easy user interaction, with an intuitive UI/UX and a simple content generation process. The system can be easily maintained and updated, ensuring operational efficiency.

2.3 Benefits

The **Locally Yours** project provides the following benefits:

- **Time-saving**: AI-powered content generation saves user time.
- **Customization**: The platform offers customizable templates for localized content.
- **Cost-effective**: A free usage model and affordable premium subscriptions make it accessible.
- **Scalable**: The platform can handle increased user traffic as it grows.

2.4 Software Requirements Specification (SRS)

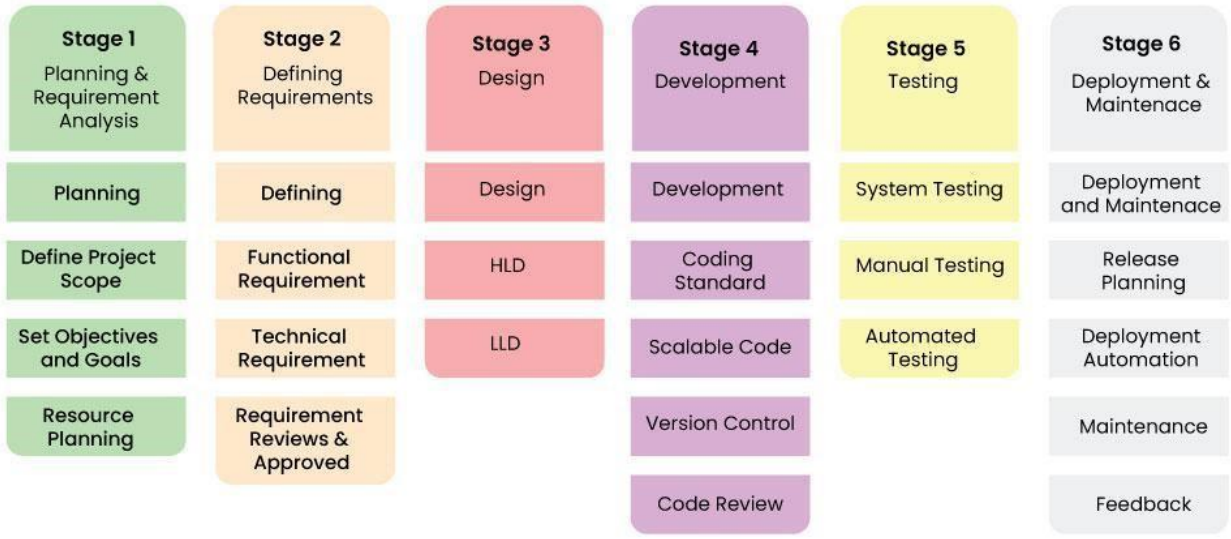
The Software Requirements Specification (SRS) outlines both functional and non-functional requirements for the project:

- **Functional Requirements**: User authentication, template selection, content generation, and subscription management.
- **Non-Functional Requirements**: Performance, security, scalability, and usability.

Chapter 3: Design

3.1 Introduction

The design phase focuses on structuring the **Locally Yours** system to meet the project's requirements. This chapter covers the software development lifecycle (SDLC), the software engineering paradigm, the system architecture, and the control flow graph.

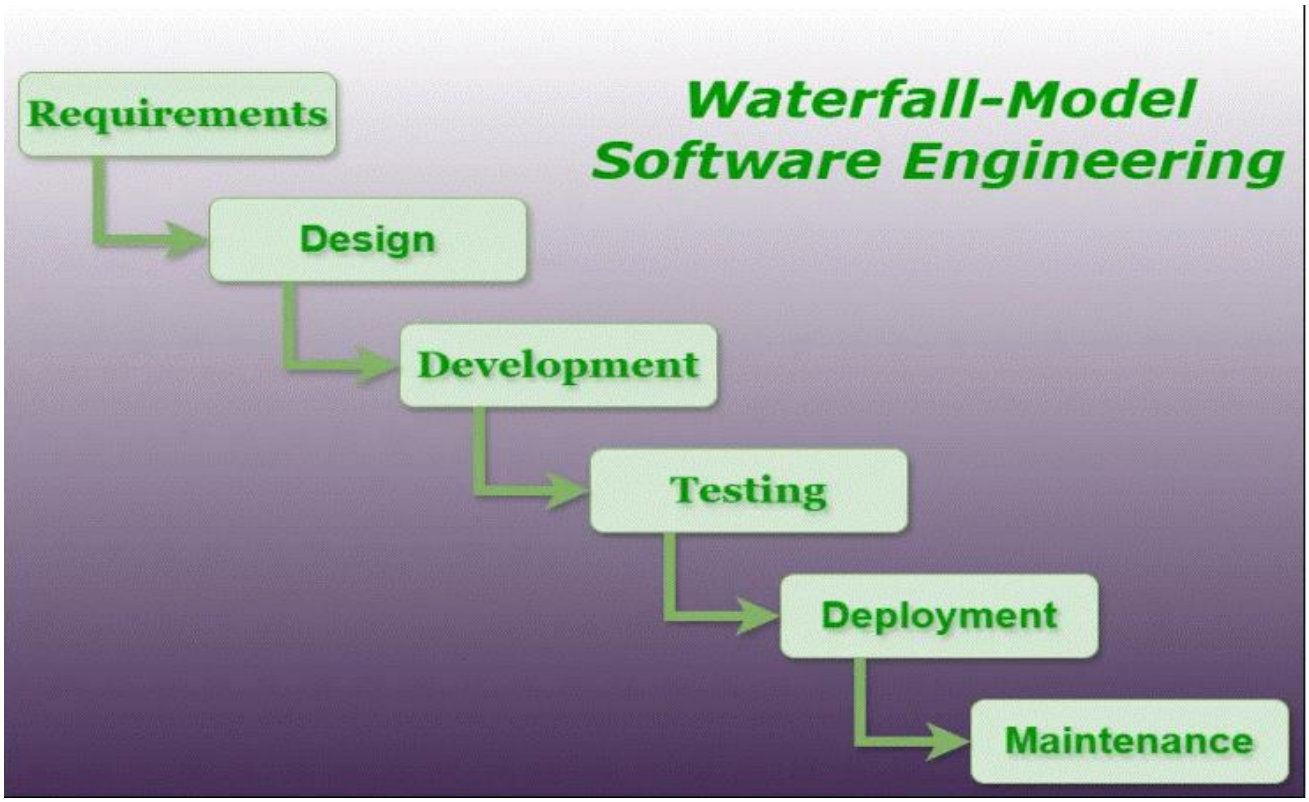


3.2 Analysis

The analysis phase identified key requirements, such as secure user authentication, content template selection, and AI integration. These requirements guide the design of the system.

3.3 Software Development Life Cycle (SDLC)

Locally Yours follows the **Waterfall Model**, which ensures a structured approach to development. The phases include requirement gathering, system design, implementation, testing, deployment, and maintenance.



3.4 Software Engineering Paradigm

The **Agile Methodology** is integrated within the Waterfall model, allowing for flexibility and iterative improvements during the development process.

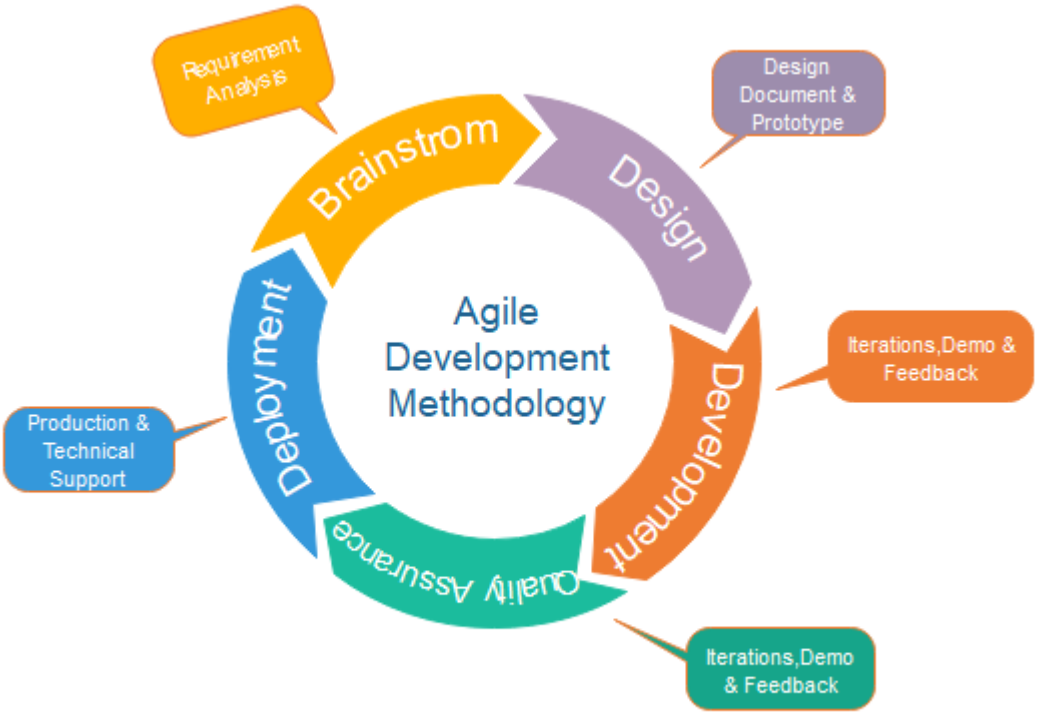


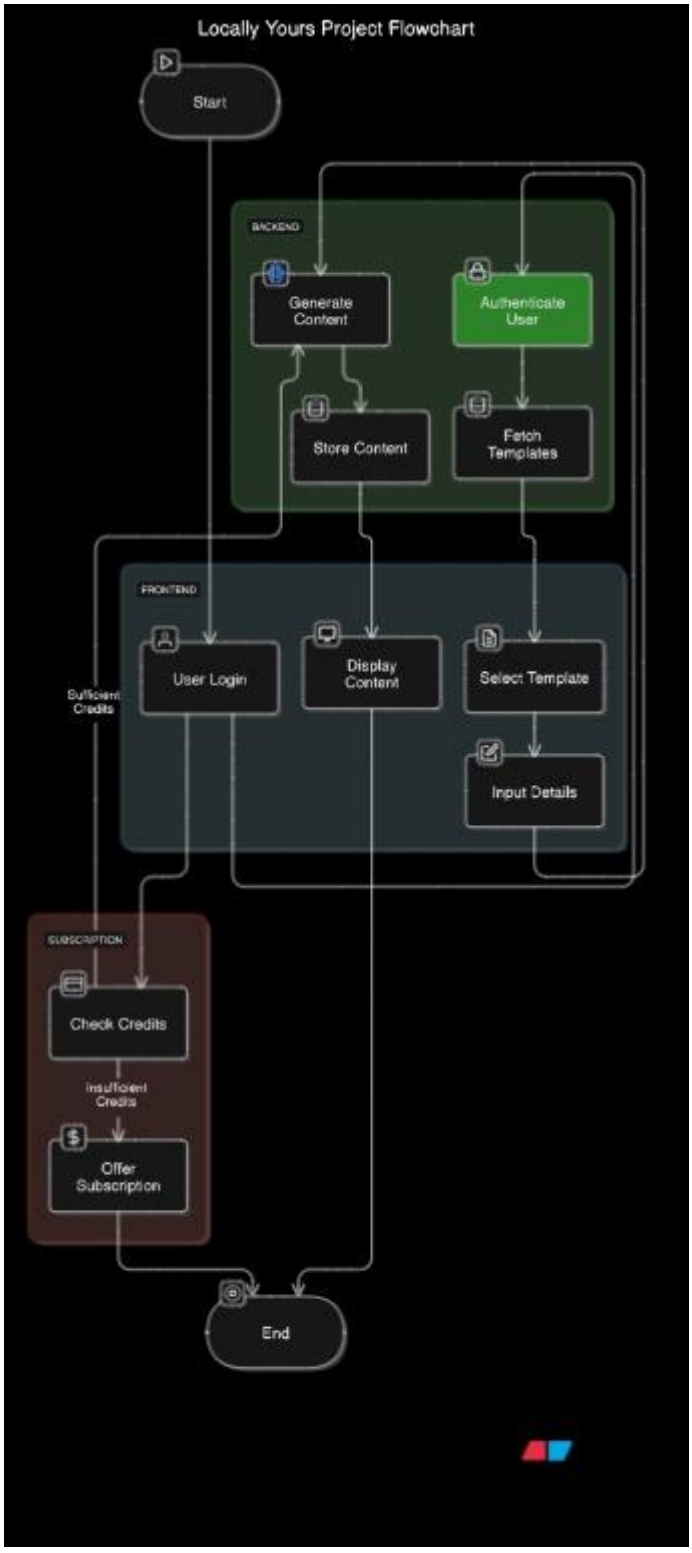
Fig. Agile Model

3.5 Architecture of the System

The architecture is a **Client-Server** model, where the frontend is built with React and Next.js, the backend uses Node.js and PostgreSQL, and AI content generation is handled via Google's API.

3.6 Control Flow Graph

The control flow graph defines the sequence of operations for processes like user login, template selection, content generation, and subscription management. It ensures clear and structured operations for both the frontend and backend components


Locally Yours Project Flowchart
Start
BACKEND
Generate Content
Authenticate User
Store Content
Fetch Templates
FRONTEND
User Login
Display Content
Select Template
Input Details
Sufficient Credits
SUBSCRIPTION
Check Credits
Insufficient Credits
Offer Subscription
End

CHAPTER 4

PROJECT SCREENSHOTS

4.1 Home Page

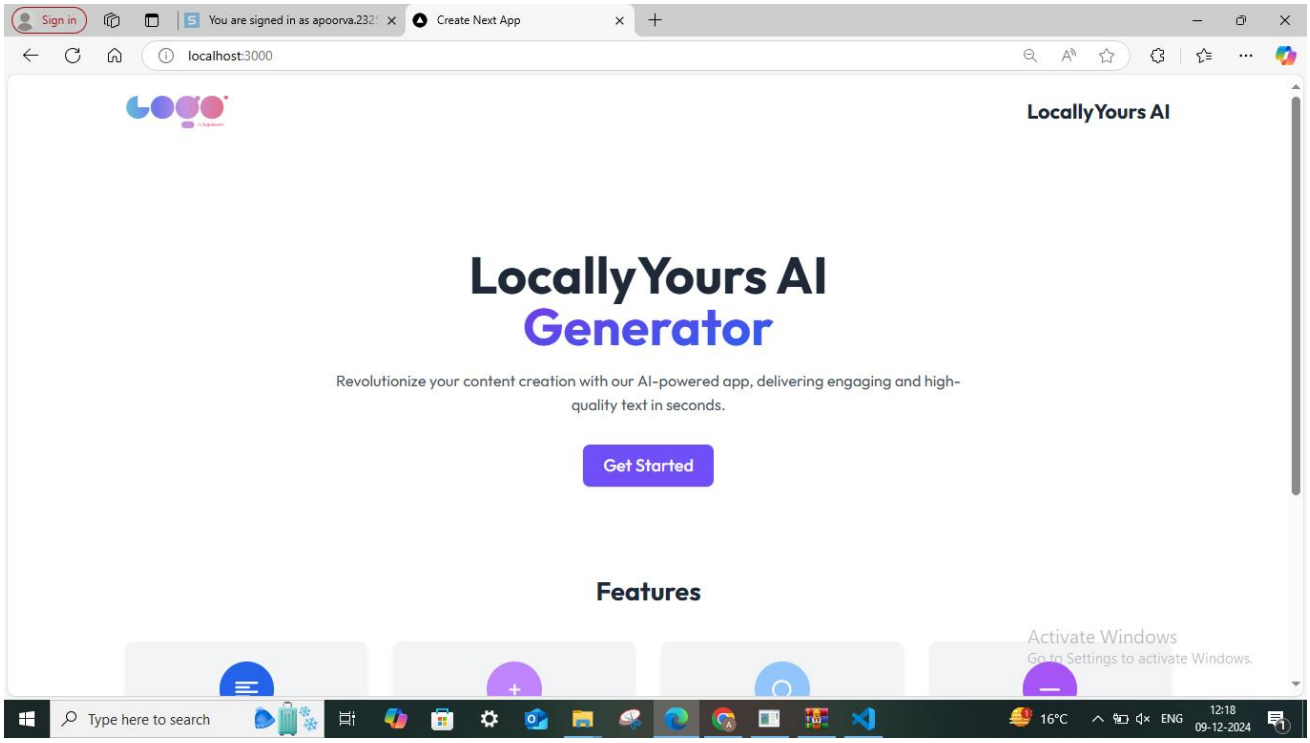


Fig.4.1 Home Page

4.2 Login in

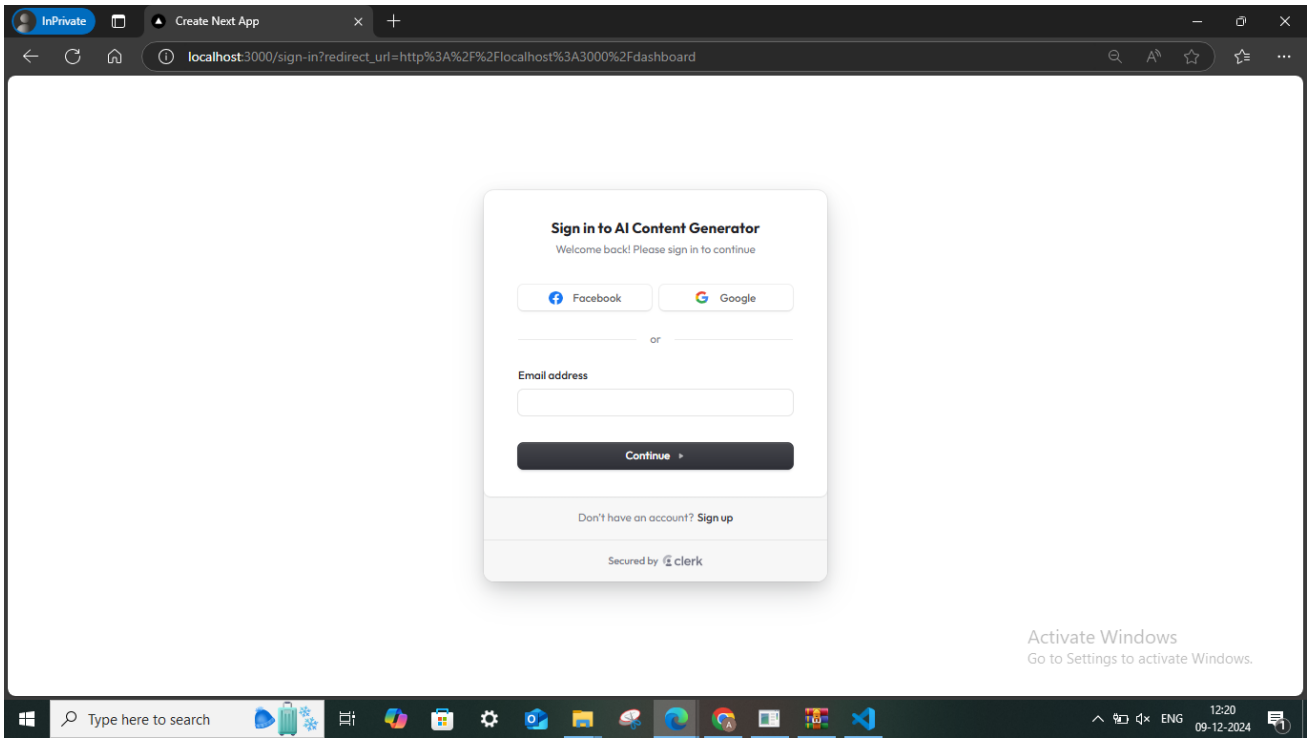


Fig.4.2 Login in

4.3 Dashboard

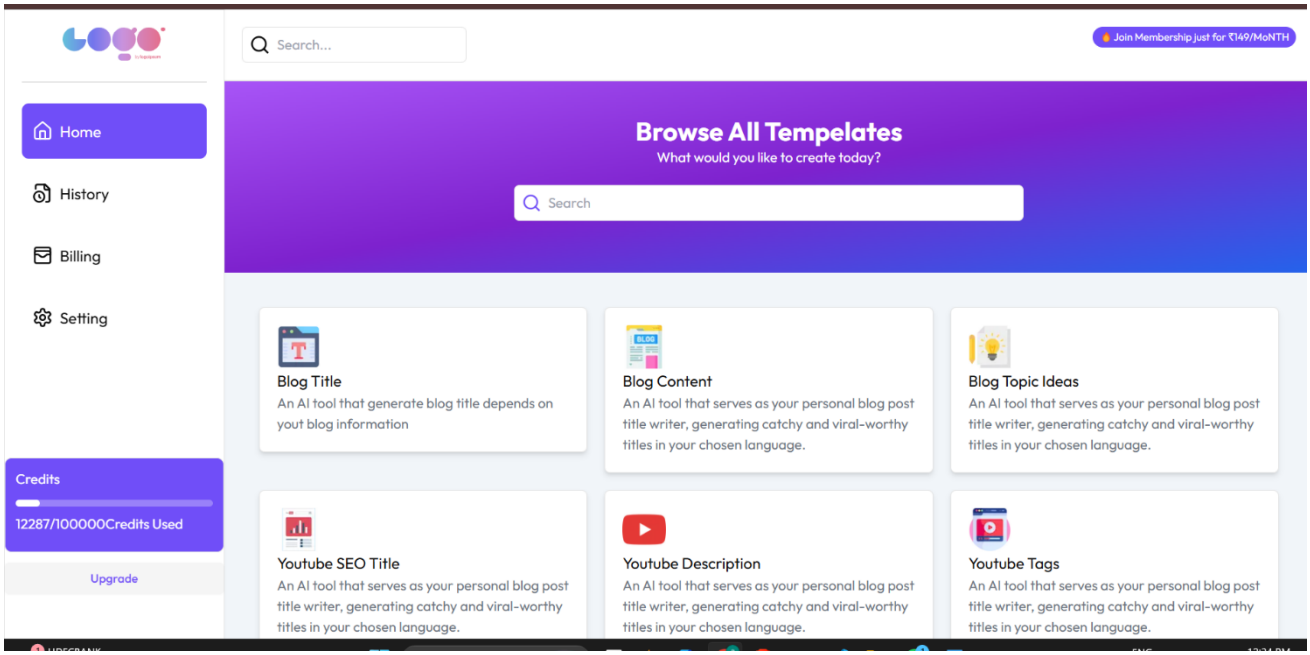


Fig.4.3 Dashboard

4.4 Content Page

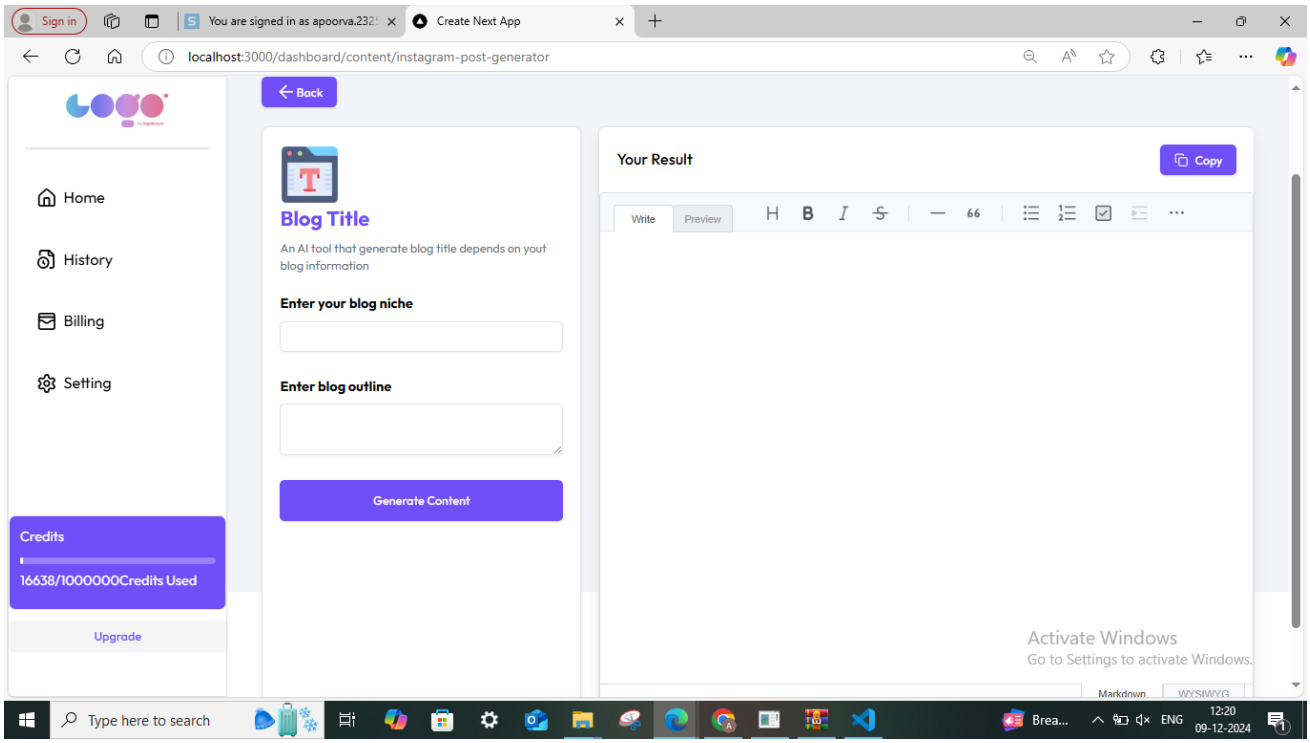


Fig.4.3 Content Page

4.5 History Page

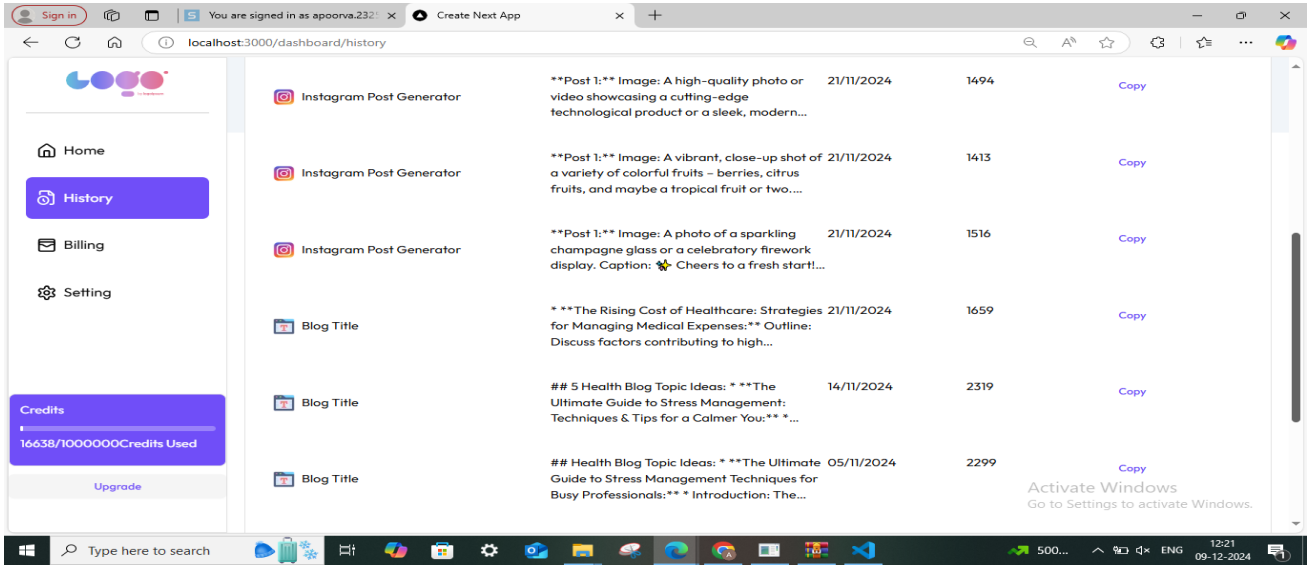


Fig.4.5 History Page

4.6 Setting Page

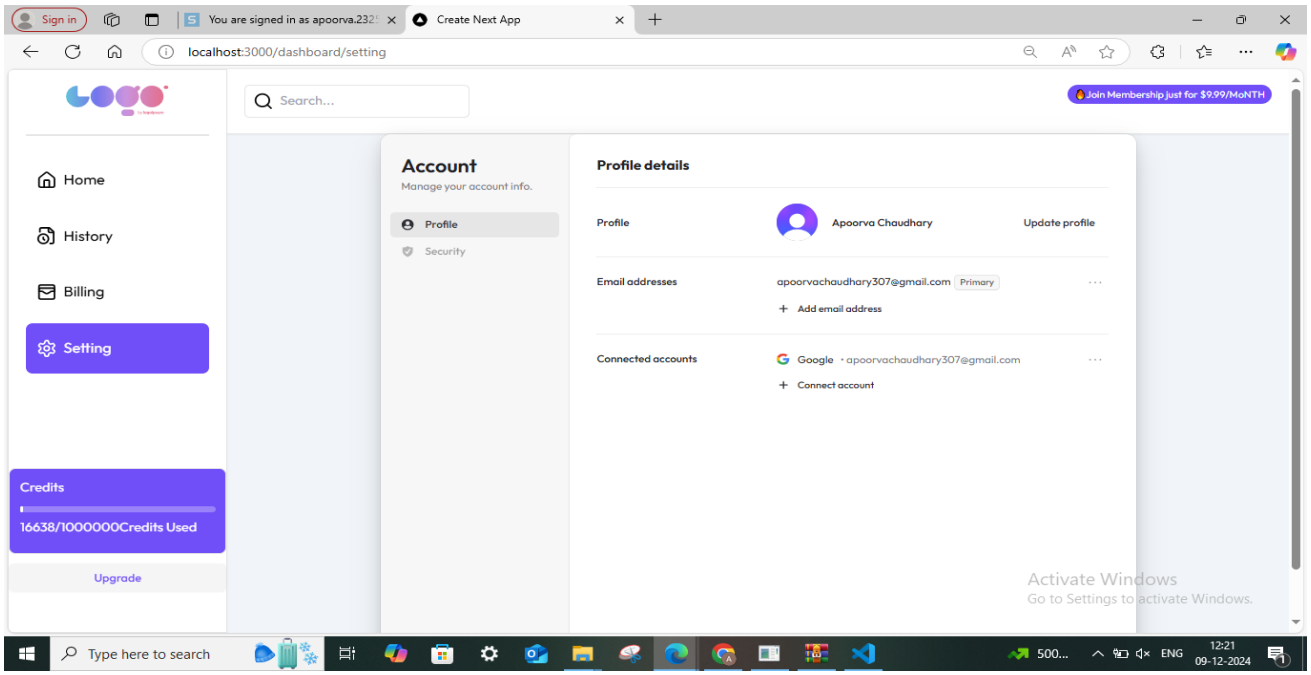


Fig.4.6 Setting Page

4.7 Billing Page

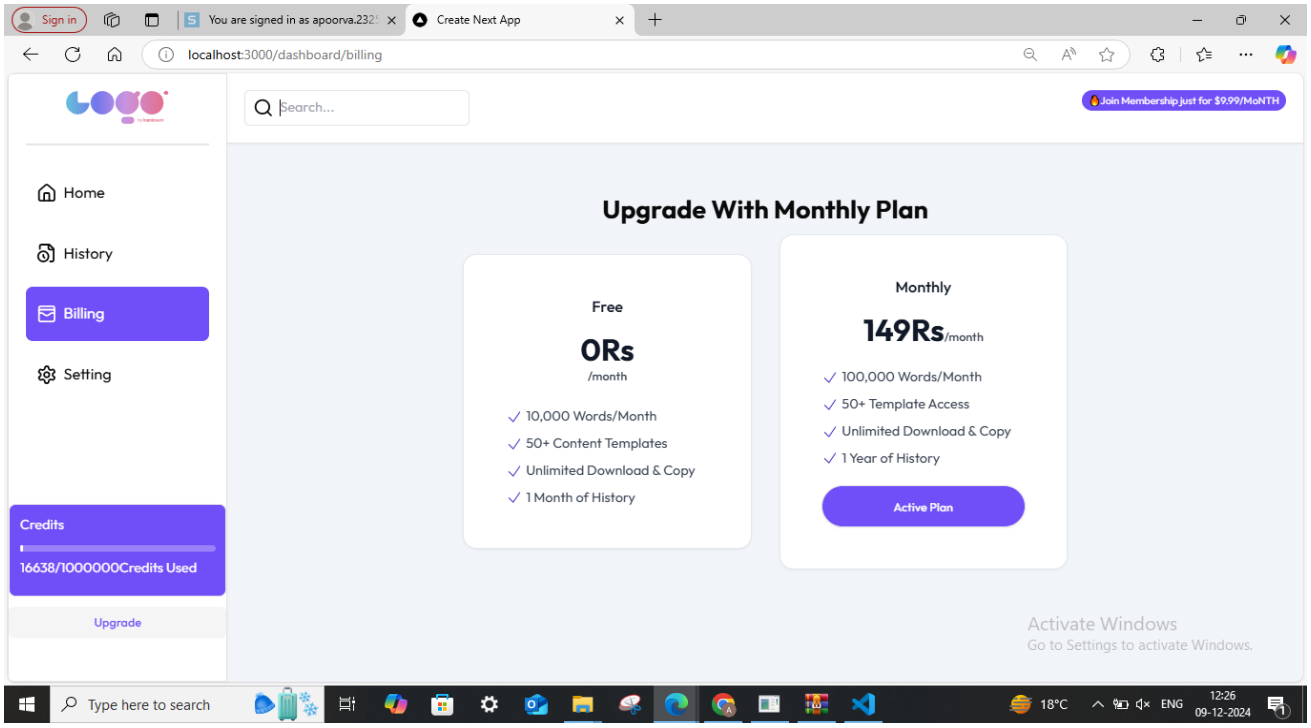


Fig.4.7 Billing page

Chapter 5: Testing

5.1 Introduction

Testing is an essential phase in the software development lifecycle that ensures the quality, functionality, and reliability of a system. For the **Locally Yours** project, a comprehensive testing strategy was adopted to ensure that the platform operates as expected and meets the requirements specified in the Software Requirements Specification (SRS).

The objective of testing is to identify and rectify any defects, validate the system's functionality, and ensure that it performs efficiently and reliably. Different types of testing were carried out at various levels to ensure that the **Locally Yours** platform is robust and user-friendly.

5.1.1 Testing Objectives

The key objectives of testing in the **Locally Yours** project were:

- **Verify Functionality**: Ensure that the system behaves as expected and meets functional requirements, such as content generation, user authentication, and subscription management.
- **Ensure Usability**: Test the user interface to ensure that it is intuitive and easy to navigate.
- **Identify Defects**: Find and resolve any bugs or issues that may cause the system to malfunction or behave unexpectedly.
- **Test Security**: Ensure that user data is protected, and the authentication system is secure.
- **Ensure Performance**: Validate that the system performs well under different conditions, such as varying traffic or content generation loads.
- **Validate Integration**: Ensure that all components of the system, such as the frontend, backend, database, and AI API, work together seamlessly.

5.1.2 Testing Principles

The following principles guided the testing process for **Locally Yours**:

- **Early Testing**: Testing was initiated early in the development process to catch defects at an early stage, reducing the cost of fixing them later.
- **Comprehensive Testing**: A range of testing types, including functional and non-functional testing, was carried out to ensure complete system validation.

- **Regression Testing**: After each change or addition to the system, regression testing was performed to ensure that the new code did not break existing functionality.
- **Automation**: Where applicable, automated testing tools were used to speed up the testing process and improve efficiency, especially for repetitive tasks such as unit testing and performance testing.

5.2 Level of Testing

Testing was conducted at different levels, ensuring that all aspects of the **Locally Yours** platform were thoroughly tested. The following sections describe each level of testing in detail.

5.2.1 Unit Testing

Unit testing focuses on individual components or modules of the application, ensuring that each one functions correctly in isolation. The main objectives of unit testing were:

- To test the logic behind individual functions, such as content generation, template selection, and user authentication.
- To test individual API calls, database queries, and AI integration.
- To verify that small units of code produce the expected results.

Tools used for unit testing:

- **Jest**: A JavaScript testing framework was used to test the individual components, ensuring that functions and modules worked as expected.
- **React Testing Library**: Used to test React components, ensuring that they render correctly and interact with the user as intended.

5.2.2 Integration Testing

Integration testing checks whether different modules or components of the system interact as expected. The objective is to test the integration points between various subsystems such as the frontend, backend, database, and third-party services like Google AI API and Razorpay. Key integration tests performed include:

- Testing the interaction between the frontend and backend, such as data flow from the UI to the server.
- Ensuring that the AI content generation API correctly integrates with the system and returns the expected results.

- Verifying that payment processing via Razorpay works seamlessly and updates user credits correctly.

Tools used for integration testing:

- **Postman**: Used to test API endpoints and simulate interactions with external services (e.g., Razorpay, Google AI API).
- **Supertest**: A tool for testing HTTP requests, ensuring that the API returns the expected responses.

5.2.3 System Testing

System testing is the final level of testing, where the entire application is tested as a whole. It involves testing the complete functionality of **Locally Yours** to ensure that all components work together as expected. The objectives of system testing were:

- To test the platform as a whole, from the user login process to content generation and subscription management.
- To simulate real-world usage scenarios, ensuring that the system performs efficiently under varying loads.
- To identify any issues that may arise when all modules and services interact with one another.

System testing included the following:

- **End-to-End Testing**: Simulating a user's journey through the platform, including logging in, selecting a template, generating content, and making a payment.
- **Usability Testing**: Assessing the ease of use of the platform, ensuring that users can navigate the dashboard, select templates, and generate content without difficulty.
- **Security Testing**: Ensuring that sensitive user information is handled securely, including checking the authentication flow and ensuring data privacy.
- **Load Testing**: Simulating high traffic and load conditions to verify that the system can handle multiple concurrent users without performance degradation.

Tools used for system testing:

- **Selenium**: Automated testing tool for performing end-to-end tests in a browser environment.
- **Apache JMeter**: Used for performance and load testing, ensuring that the platform can handle high traffic and stress.

Conclusion of Testing

The testing phase of the **Locally Yours** project was crucial in ensuring that the platform functions as expected and provides a seamless experience to users. Through a combination of unit, integration, and system testing, the project achieved high reliability, security, and performance. Any bugs or issues identified during testing were addressed promptly, and the platform was optimized for user satisfaction and scalability.

Testing will continue as part of the maintenance phase to ensure that future updates and features are delivered without introducing new issues. Overall, the testing phase contributed significantly to the success of the **Locally Yours** project, ensuring a robust and dependable content generation platform.

Chapter 6: Conclusion & Future Scope

6.1 Conclusion

The **Locally Yours** project has successfully achieved its goal of creating a full-stack AI-driven content generation platform that enables users to generate personalized, locally relevant content efficiently. By integrating advanced technologies like React, Next.js, Tailwind CSS, and PostgreSQL, the project ensures a seamless, responsive user experience while maintaining high performance on the backend.

The platform's main functionality, powered by Google's AI API, allows users to select from a variety of templates, input relevant data, and generate high-quality content quickly and accurately. Additionally, the secure user authentication system, credit management, and subscription features powered by Razorpay make the platform easy to use and scalable.

The testing phase of the project confirmed that **Locally Yours** performs well under different conditions, with all components working together as expected. The system is secure, reliable, and user-friendly, meeting the functional and non-functional requirements outlined in the Software Requirements Specification (SRS).

In conclusion, **Locally Yours** has effectively addressed the need for easy-to-use, AI-powered content generation tools, offering businesses and content creators an affordable, scalable solution for creating locally relevant content. The project demonstrates how combining modern web technologies and AI can create innovative, user-friendly solutions to real-world challenges.

6.2 Future Scope

Although **Locally Yours** is functional and meets the current requirements, there is significant potential for further development and enhancement. The following are some possible future enhancements and directions for the project:

- **Expansion of Templates**: Currently, the platform offers a limited set of content templates. In the future, additional templates can be added to cater to different industries, languages, and niches, providing a more comprehensive tool for users.
- **Multi-Language Support**: Implementing multi-language support would allow **Locally Yours** to cater to a global audience, enabling users from different regions to generate content in their preferred language.

- **Enhanced AI Capabilities**: As AI technology evolves, more advanced content generation models could be integrated, allowing for more sophisticated content creation, such as more personalized or context-aware content.
- **Collaboration Features**: Allowing multiple users to collaborate on content generation within the platform could enhance the value of **Locally Yours** for businesses, agencies, or content teams working together.
- **Advanced Analytics and Reporting**: Providing users with detailed analytics and reports on their generated content (e.g., performance metrics, user engagement, etc.) could help them refine their strategies and improve their content quality.
- **Mobile Application Development**: While the current platform is web-based, developing a mobile application for iOS and Android could expand the accessibility of **Locally Yours**, making it even easier for users to generate content on the go.
- **Integration with Other AI Platforms**: Integrating additional AI tools, such as AI-driven image or video generation, could further enhance the platform's capabilities, making it a more versatile tool for content creators.

In summary, **Locally Yours** has a strong foundation and significant room for growth. The platform could expand its offerings, improve its AI capabilities, and scale its services to meet the needs of an even wider user base.

6.3 Bibliography

The bibliography section lists the resources referenced and consulted during the development of the **Locally Yours** project. Below are the key references:

1. **React Documentation**. React. https://reactjs.org/docs/getting-started.html
2. **Next.js Documentation**. Next.js. https://nextjs.org/docs
3. **Tailwind CSS Documentation**. Tailwind CSS. https://tailwindcss.com/docs
4. **PostgreSQL Documentation**. PostgreSQL. https://www.postgresql.org/docs/
5. **Drizzle ORM Documentation**. Drizzle ORM. https://www.npmjs.com/package/drizzle-orm
6. **Google AI API Documentation**. Google Cloud. https://cloud.google.com/ai
7. **Razorpay API Documentation**. Razorpay. https://razorpay.com/docs/
8. **Clerk Authentication Documentation**. Clerk. https://clerk.dev/docs
9. **Selenium Documentation**. Selenium. https://www.selenium.dev/documentation/en/
10. **Jest Documentation**. Jest. https://jestjs.io/docs/en/getting-started
11. **Apache JMeter Documentation**. Apache JMeter. https://jmeter.apache.org/
12. **Postman Documentation**. Postman. https://www.postman.com/docs

CHAPTER 7 REFRENCES

1. https://www.kaggle.com/datasets/saurabhbagchi/books-dataset
2. https://community.hubspot.com/t5/CMS-for-Marketers/Learn-HTML-amp-CSS-and-JavaScript-amp-jQuery-book-recommendation/td-p/651970
3. https://www.youtube.com/watch?v=1YoD0fg3_EM
4. https://www.codewithfaraz.com/content/259/creating-a-book-store-website-using-html-css-and-javascript
5. https://thectoclub.com/news/web-development-books/