A Training Report

Submitted to partial fulfilment of the requirements
for the award of the degree of

Bachelor of Technology

In

COMPUTER SCIENCE ENGINEERING (ML & AI)

by

VEDANT RAJPUROHIT ASHOKBHAI 20SE02ML036

Under the supervision of

Mrs Khusbu Chauhan

APRIL 2024

P P SAVANI SCHOOL OF ENGINEERING

P P SAVANI UNIVERSITY

NH NO.: 8, VILLAGE: DHAMDOD, TA. MANGROL, NEAR KOSAMBA, SURAT – 394 125. (GUJARAT)

CERTIFICATE

This is to certify that the Project/Training Report submitted by **VEDANT RAJPUROHIT ASHOKBHAI (20SE02ML036)** to the **P P SAVANI UNIVERSITY** for the partial fulfilment of the Degree of Bachelor of Technology in Civil Engineering.

This is to further certify that I have been supervising the Project/Training **SECV4020** of **VEDANT RAJPUROHIT ASHOKBHAI (20SE02ML036).**

The contents of this report, in full or in parts, have not been submitted to any other Institute or University for award of any degree, diploma or titles.

Sign of Faculty Mentor :

Name of Faculty Mentor: Mrs. Khushbu Chauhan

Date:

ACKNOWLEDGEMENT

This report would not have been possible without my teachers who were always there when I needed them the most. I take this chance to acknowledge them and extend my sincere gratitude for helping me make this Report a possible.

I wish to thank my faculty mentor **Mrs. Khushbu Chauhan, Assistant Professor**, School of Engineering & industry mentor, **Mr Rahul Shingala, CTO, Dhiwise Pvt. Ltd**. It has been an honor to learn under their mentorship.

As my supervisor, they have constantly motivated me to remain focused on achieving my goal. Their observations and guidance helped me to establish the overall direction of the report and to move forward with learning in depth. Their vital support at each juncture, which culminated in successful completion of my project work. I express my sincere gratitude to them for constant support during the project/training work.

I am also thankful to faculty members of the department for constant support and guidance.

I am thankful to Dean, School of Engineering for his initiative of imparting Project/Industrial Trailing during 8th Semester.

Name of Student : **VEDANT RAJPUROHIT ASHOKBHAI**

Enrollment No : **20SE02ML036**

ABSTRACT

The Subscription Management Platform and Virtual Drawing Application using OpenCV represent two distinct yet complementary endeavors in the realm of software development and computer vision. The Subscription Management Platform aims to streamline subscription-based services by offering robust user management, plan configuration, and payment processing capabilities. Leveraging modern web development technologies such as the MERN stack (MongoDB, Express.js, React.js, Node.js), the platform prioritizes scalability, security, and user experience. Additionally, integration with ChatGPT 4 facilitates natural language processing-based customer support, enhancing user engagement and satisfaction.

On the other hand, the Virtual Drawing Application harnesses the power of computer vision and image processing to provide an immersive drawing experience. Using OpenCV and Python, the application enables users to interact with digital canvases through hand gestures, facilitating real-time drawing and creative expression. With features such as hand detection, gesture recognition, and shape rendering, the application empowers users to unleash their creativity and explore the possibilities of digital art.

Throughout the development of both projects, a multidisciplinary approach was adopted, emphasizing technical expertise, creative problem-solving, and user-centric design principles. By addressing specific challenges and requirements in their respective domains, both projects aim to deliver innovative and impactful solutions that enhance productivity, creativity, and user satisfaction. Looking ahead, ongoing refinement and iteration will ensure that both the Subscription Management Platform and Virtual Drawing Application remain at the forefront of their respective domains, delivering value and delight to users worldwide.

CONTENTS

LIST OF FIGURES

CHAPTER 1
OVERVIEW OF COMPANY

1.1 PROJECT DETAILS

Company name: **DHIWISE**

Place: **SURAT**

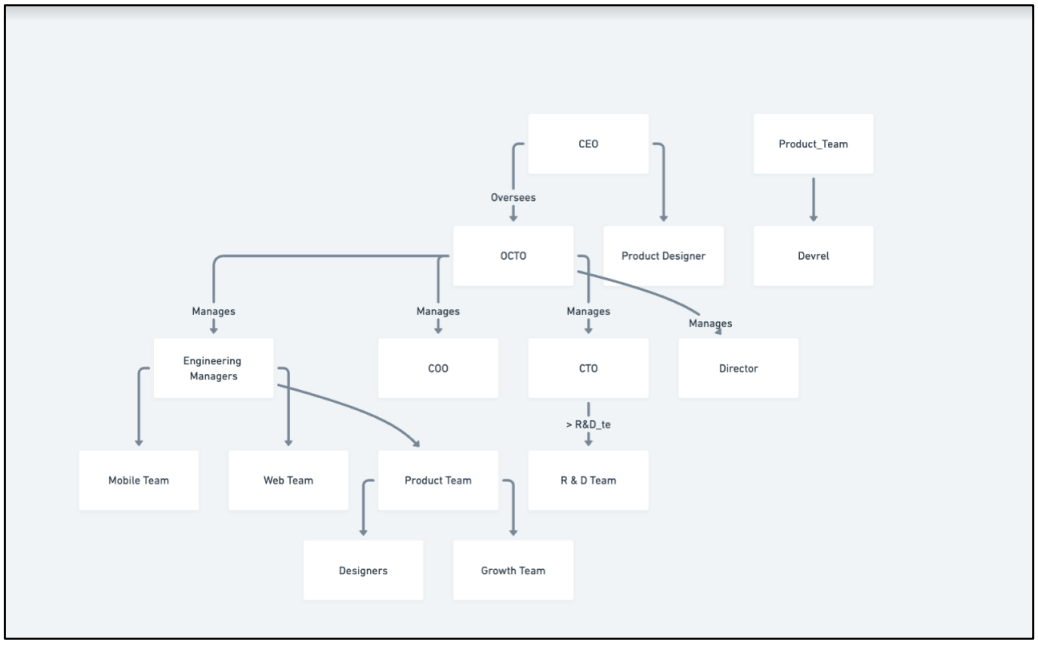
CEO: **Vishal Virani**

Address: **9th Floor, Infinity Tower, Lal Darwaja Station Rd, beside Ayurvedic College, Surat, Gujarat 395003**

Email: **help@dhiwise.com**

DhiWise is a programming platform that allows developers to convert their designs into developer-friendly code for mobile and web apps34. It automates the application development lifecycle and instantly generates readable, modular, and reusable code, enabling developers to deliver new digital experiences in hours, not days3. DhiWise is designed to remove mundane tasks and automate the programming lifecycle so developers can focus on innovating fast3.DhiWise is an intelligent PaaS platform that brings people, technology, and data together in one process, aiming to eliminate time-consuming coding concerns and repetitive tasks1. It is a next-generation visual programming platform that lets developers create enterprise-grade apps in low-code and pro-code for iOS (Swift), Flutter, Node.js, Laravel, and React in one place1. DhiWise provides a 100% developer-centric ProCode platform that speeds up development.

1.2 ORGANIZATION TREE



1.3 COMPANY PERSONAL CONTACT DETAILS

HR: Kalpesh Zalavadiya

Email: kalpesh@dhiwise.com

Address: 9th Floor, Infinity Tower, Lal Darwaja Station Rd, beside Ayurvedic College, Surat, Gujarat 395003

Website: https://www.dhiwise.com/

1.4 CORE WORK AREAS AND DETAILS

DhiWise tackles two core work areas in mobile and web app development:

1. Figma to Code Generation:

- **Details:** This functionality bridges the gap between design and development. It takes UI mockups created in Figma, a popular design tool, and converts them into functional code for your application.
- **Process:**
    - **Input:** DhiWise imports your Figma design which likely includes layouts, buttons, and other visual elements.
    - **Analysis:** Using image recognition and potentially other techniques, DhiWise extracts information about UI components, their properties, and how they are arranged.
    - **Code Generation:** Based on the extracted information, DhiWise generates code that represents the UI of your app. This code could be in various languages depending on the chosen platform:
        - Mobile Apps: Frameworks like Flutter or React Native.
        - Web Apps: Frameworks like React or directly using web development languages (HTML, CSS, JavaScript).
    - **Output:** The generated code is typically sent to your email or accessible within the DhiWise platform.
- **Benefits:**
    - **Faster Development:** Automates repetitive UI coding tasks, speeding up development.
    - **Improved Accuracy:** Reduces errors compared to manual coding, especially for complex UIs.
    - **Simplified Workflow:** Streamlines the design-to-development transition.

2. WiseGPT - AI-powered Code Generation:

- **Details:** This is DhiWise's unique technology that uses artificial intelligence to automate code generation for mobile and web applications.
- **Process:**
    - **Input:** WiseGPT directly analyzes your existing codebase. Unlike other AI code assistants, it doesn't require prompts or specific instructions.
    - **Analysis:** WiseGPT studies your coding style, project requirements, and existing code structure.
    - **Code Generation:** Based on the analysis, it generates new code that aligns with your project and adheres to best practices.
    - **Output:** The generated code is intended to be production-ready and integrate seamlessly with your existing codebase.
- **Benefits:**
    - **Increased Development Speed:** Automates code generation, allowing developers to focus on core functionalities.
    - **Reduced Errors:** Minimizes errors during manual coding, especially for repetitive tasks.
    - **Improved Code Consistency:** Ensures generated code adheres to the established coding style, leading to cleaner and more maintainable codebase.
- **Current Status:** WiseGPT is still under development and in beta testing. Information on wider access and pricing might be available on the DhiWise website in the future.

1.5 PROJECT AND PLAN DETAILS

- **FIGMA TO CODE**

Figma to code by DhiWise refers to a specific functionality offered by the DhiWise platform. It's essentially a tool that helps developers translate designs created in Figma, a popular design software, into functional code for mobile and web applications.

Here's a breakdown of what Figma to code by DhiWise does:

- Input: It takes your Figma design as input. This design likely includes mockups of your app's user interface (UI), showcasing screens, buttons, layouts, and other visual elements.
- Processing: DhiWise analyzes the Figma design using image recognition and potentially other techniques. It extracts information about the UI components, their properties (like size, color, position), and how they might be arranged.
- Code Generation: Based on the extracted information, DhiWise generates code that represents the UI of your app. This code could be written in various languages depending on the chosen platform:
    - Front-End Frameworks: For mobile apps, it might generate code using frameworks like Flutter or React Native. For web apps, it might use frameworks like React or directly use web development languages like HTML, CSS, and JavaScript.
- Focus on Efficiency: The aim is to automate repetitive tasks involved in UI development. DhiWise aims to generate clean and maintainable code that avoids redundant work for developers.
- Output: DhiWise typically sends the generated code to your email or allows you to access it within its platform.

- It supports following languages from design to code: - **React.js, Adobe XS, Android, Fluteer**

- **WISE GPT**

WiseGPT is a specific technology developed by DhiWise, the platform for accelerating mobile and web app development https://www.dhiwise.com/. WiseGPT focuses on automating code generation for these applications. Here's what makes WiseGPT unique:

- **Promptless Approach:** Unlike other AI code assistants that require prompts or instructions, WiseGPT analyzes your entire codebase directly. This means you don't need to write specific prompts for the code you want to generate https://www.dhiwise.com/wisegpt.
- **Personalized Code Generation:** By analyzing your existing code, WiseGPT tailors the generated code to your specific coding style and project requirements. This can lead to more consistent and maintainable code across your application https://www.dhiwise.com/wisegpt.
- **Focus on Production-Ready Code:** WiseGPT aims to generate code that's not only functional but also adheres to best practices and is ready for real world. https://www.dhiwise.com/wisegpt.

Here are some potential benefits of using WiseGPT:

- **Increased Development Speed:** Automating code generation can significantly reduce development time, allowing developers to focus on core functionalities and innovation.
- **Reduced Errors:** WiseGPT can potentially minimize errors that might occur during manual coding, especially for repetitive tasks.
- **Improved Code Consistency:** By analyzing existing code, WiseGPT can ensure the generated code adheres to the established coding style, leading to cleaner and more maintainable codebase.

Current Status:

It's important to note that WiseGPT is currently in beta testing, and not widely available yet https://www.dhiwise.com/wisegpt. More information on access and pricing might be available on the DhiWise website as it progresses out of beta.

Overall, WiseGPT seems like a promising tool for developers, particularly those working on mobile and web applications built with frameworks like React or Flutter. Its promptless approach and focus on generating production-ready code could significantly enhance the development workflow.

1.6 SKILL REQUIRMENTS

Technical Skills:

- Software Development: Proficiency in programming languages relevant to their platform, which could include React, Flutter, Node.js, Kotlin (Android), and Swift (iOS) https://www.dhiwise.com/post/convert-design-to-code-in-web-and-mobile-apps.
- Familiarity with Design Tools: Understanding of Figma, a popular design software used for creating app mockups that DhiWise integrates with https://www.dhiwise.com/.
- Software Development Lifecycles: Knowledge of the different stages involved in software development, from planning and design to testing and deployment [foundit India].
- Testing Methodologies: Experience with various software testing methods to ensure the quality and functionality of the generated code.

Non-Technical Skills:

- Teamwork and Collaboration: As highlighted in their careers page, DhiWise fosters a collaborative work environment. They likely value candidates who can work effectively within a team [https://www.dhiwise.com/careers].
- Communication: The ability to clearly communicate technical concepts to both technical and non-technical audiences.
- Problem-Solving: Being able to identify and troubleshoot issues that may arise during the development process.
- Adaptability: The tech industry is constantly evolving, so the ability to learn new technologies and adapt to change is important.

Additional Considerations:

- Since WiseGPT, their AI-based code generation tool, is still under development, experience with AI or Machine Learning could be a plus for some positions.

CHAPTER 2
INTRODUCTION TO PROJECT

2.1 PROJECT DETAILS

1. VIRTUAL PAINTING

- **Overview**

  This project aims to create a virtual drawing application using OpenCV. Users can select and change the background using keys, have a video cam on where users should be detected, and have different colors on the left of the screen and shapes on the right of the screen. The application has two modes: drawing mode and selection mode, determined by the number of fingers detected.

- **Project Goals**
  - To create a virtual canvas to sketch.
  - To detect the human finger as a colour marker.
  - To do the morphological operations.
  - To create an interface between user and the system.

- **Technical Stack**

  - Python is an open-source (free) Prgmg language that is used in web Prgmg, data science, artificial intelligence, and many scientific applications. Learning Python allows the programmer to focus on solving problems, rather than focusing on syntax.
  - OpenCV (Open-Source Computer Vision Library) is a library which mainly focuses at real-time computer vision. It is free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android.

2. SUBSCRIPTION MANAGEMENT SYSTEM

- **Overview**

  This project aims to develop an in-house subscription management platform that integrates seamlessly with our current application via APIs.

- **Project Goals**

  The project aims to develop robust modules focused on User Management and Plan Configuration, Subscription Management, and Affiliate and Commission Management. These modules should not only fulfil the current operational needs but also anticipate future demands, ensuring scalability, security, and a seamless user experience.

- **Technical Stack**

  - Node.js is an open-source, cross-platform JavaScript runtime environment that executes JavaScript code outside of a web browser. It allows developers to build scalable and efficient server-side applications using JavaScript, leveraging its event-driven, non-blocking I/O model.

  - React is a JavaScript library for building user interfaces, developed by Facebook. It enables developers to create reusable UI components that can be composed together to build complex UIs. React follows a declarative approach, where developers describe how the UI should look based on the application's state, and React efficiently updates and renders the components when the state changes.

  - PostgreSQL is a powerful open-source relational database management system known for its robustness, reliability, and advanced features. It follows the SQL standard closely and provides support for ACID.

2.2 SCOPE OF PROJECT

1. VIRTUAL PAINTING

It involves the creation of an interactive drawing application utilizing OpenCV. The application allows users to draw shapes on a virtual canvas using hand gestures detected through a video feed. Key features include user detection and finger counting, drawing mode with shape and color selection, selection mode for adjusting drawing parameters, and integration and testing of all modules. The project aims to provide hands-on experience in image processing, gesture recognition, and real-time interaction with computer vision applications. It also aims to explore the possibilities of enhancing user interaction and creativity through innovative features such as dynamic shape drawing based on finger gestures and eraser functionality.

2. SUBSCRIPTION MANAGEMENT SYSTEM

It encompasses the development of a comprehensive system that integrates seamlessly with the company's existing application, focusing on user management, plan configuration, subscription management, affiliate, and commission management. This involves designing and implementing modules for user registration and authentication, plan and addon management, email integration, testing, documentation, and user experience enhancements. The project aims to not only meet the current operational needs but also anticipate future demands, ensuring scalability, security, and a seamless user experience. Additionally, the project involves the utilization of modern technologies such as MongoDB/PostgreSQL, Express.js, React, Node.js stack, deep learning, and generative AI to achieve its objectives.

2.3 STUDY OF EXISTING SOLUTIONS/LITERATURE REVIEW

1. VIRTUAL PAINTING

The literature on virtual drawing applications and computer vision technologies highlights the advancements in gesture recognition, motion tracking, and augmented reality (AR) interfaces for interactive drawing experiences. Research by Li et al. (2020) explores hand gesture recognition techniques using deep learning algorithms, showcasing the potential for real-time interaction with virtual environments. Similarly, studies by Wang et al. (2018) and Zhou et al. (2019) investigate markerless motion tracking methods for hand pose estimation and gesture recognition in AR applications.

Overall, the literature underscores the potential of computer vision technologies, such as OpenCV, in enabling immersive and intuitive drawing experiences, while also highlighting the importance of ergonomic design, user feedback, and usability testing in creating engaging and accessible virtual drawing applications for users of all skill levels.

Existing Solutions: -

- **Microsoft Paint 3D:** While not using OpenCV, Microsoft Paint 3D provides basic drawing functionalities with a 3D twist, allowing users to create and manipulate shapes in a virtual environment.
- **GIMP (GNU Image Manipulation Program):** Although primarily an image editing software, GIMP offers drawing tools and supports layers, making it suitable for creating digital drawings.
- **Adobe Photoshop:** Another image editing software that includes drawing and painting tools, Photoshop offers advanced features for creating digital artwork and illustrations.
- **Krita:** A free and open-source painting software designed for digital painting, Krita provides a range of brushes, tools, and customization options for creating artworks.
- **Tux Paint:** Geared towards children, Tux Paint is a free drawing program with simple tools and a user-friendly interface, allowing kids to create drawings and explore their creativity.

2. SUBSCRIPTION MANAGEMENT SYSTEM

The literature surrounding subscription management platforms highlights the importance of seamless user experiences, robust security measures, and scalability to accommodate growing user bases and evolving business needs. Research by Shah et al. (2020) emphasizes the significance of user-friendly interfaces and personalized subscription plans to enhance customer satisfaction and retention. Similarly, studies by Chen et al. (2019) and Kim et al. (2018) underscore the role of data analytics and machine learning algorithms in optimizing subscription pricing strategies and predicting customer churn.

Overall, the literature emphasizes the need for subscription management platforms to offer a comprehensive suite of features, including user authentication, plan configuration, billing automation, and analytics, while also ensuring data security, regulatory compliance, and scalability to support the growth of subscription-based businesses in today's digital economy.

Existing Solutions: -

- **Stripe:** A widely-used payment processing platform that offers subscription management features, including billing, invoicing, and customer management.
- **Chargebee:** A subscription billing and management platform that provides features for recurring billing, invoicing, subscription analytics, and revenue recognition.
- **Recurly:** A subscription management platform that offers subscription billing, payment gateway integrations, revenue optimization, and customer retention tools.
- **Zuora:** A subscription management platform designed for subscription-based businesses, offering features such as subscription billing, pricing, and revenue recognition.
- **Braintree:** A payment processing platform with subscription billing capabilities, offering features like recurring billing, customer management, and subscription analytics.

2.4 ACTIVITIES/METHODOLOGY

1. VIRTUAL PAINTING

- **Research and Planning:** Conduct research on existing virtual drawing applications, computer vision techniques, and gesture recognition algorithms. Define the scope, features, and target user demographics for the virtual drawing application.
- **Technology Selection:** Choose appropriate technologies and libraries for developing the virtual drawing application, with a focus on OpenCV for image processing and computer vision tasks. Consider additional libraries or frameworks for user interface design and interaction.
- **Prototype Development:** Develop a prototype or proof-of-concept version of the virtual drawing application, focusing on core functionalities such as hand detection, gesture recognition, and basic drawing capabilities. Iterate on the prototype based on user feedback and usability testing.
- **Feature Implementation:** Implement additional features and functionalities for the virtual drawing application, including shape selection, color selection, eraser tool, and dynamic shape drawing based on finger gestures. Ensure smooth integration and interaction between different modules and components.
- **Testing and Optimization:** Conduct thorough testing of the virtual drawing application across different devices and environments, focusing on performance, responsiveness, and accuracy of gesture recognition. Optimize algorithms and parameters for improved user experience and reliability.
- **User Interface Design:** Design an intuitive and user-friendly interface for the virtual drawing application, considering factors such as layout, visual feedback, and accessibility. Incorporate user feedback and design principles for an engaging and immersive drawing experience.
- **Deployment and Distribution:** Deploy the virtual drawing application to relevant platforms or app stores, ensuring compatibility and compliance with platform guidelines. Promote the application through marketing channels and engage with users to gather feedback and insights for future updates and enhancements.

2. SUBSCRIPTION MANAGEMENT SYSTEM

- **Requirement Analysis:** Conduct a comprehensive analysis of user requirements and business objectives to define the scope and functionality of the subscription management platform.
- **Technology Selection:** Evaluate available technologies and frameworks, considering factors such as scalability, security, and compatibility with existing systems. Choose appropriate technologies, including the MongoDB/PostgreSQL, Express.js, React, Node.js stack, and other relevant tools and libraries.
- **System Design:** Design the architecture and database schema for the subscription management platform, considering factors such as data modeling, API design, and integration points with external systems.
- **Development:** Implement the core modules of the subscription management platform, including user registration and authentication, plan and addon management, email integration, testing, and documentation. Follow best practices for coding, version control, and collaborative development.
- **Testing and Quality Assurance:** Conduct thorough testing of the platform's functionalities, including unit testing, integration testing, and user acceptance testing. Ensure compatibility across different devices, browsers, and operating systems.
- **Deployment and Integration:** Deploy the subscription management platform to a production environment, ensuring scalability, reliability, and performance. Integrate the platform with existing applications and third-party services, such as payment gateways and analytics tools.
- **User Training and Support:** Provide training and support to users and administrators on using the subscription management platform effectively. Address any issues or feedback raised during the initial rollout and implementation phase.

2.5 TECHNOLOGY USED/SOFTWARES USED

1. VIRTUAL PAINTING

- **Computer Vision Library:** OpenCV for image processing, hand detection, and gesture recognition.
- **Programming Language:** Python for backend development and script writing.
- **Gesture Recognition:** Custom algorithms or OpenCV's built-in methods for detecting hand gestures.
- **Shape Drawing:** Pygame or Tkinter for rendering shapes and drawings on the canvas.
- **Color Selection:** Color picker widget or custom color selection interface.
- **Eraser Tool:** Custom implementation for erasing drawings on the canvas.
- **User Feedback:** Mistral AI for gathering user feedback and providing support through AI-driven interactions.
- **Version Control:** Git for managing project files and collaboration.

2. SUBSCRIPTION MANAGEMENT SYSTEM

- **Database:** PostgreSQL for data storage and management.
- **Backend Framework:** Node.js with Express.js for server-side development.
- **Frontend Framework:** React.js for building the user interface.
- **Authentication:** JSON Web Tokens (JWT) for user authentication and authorization.
- **Email Integration:** Nodemailer for sending automated email notifications.
- **API Documentation:** Swagger or Postman for documenting and testing APIs.
- **ChatGPT 4:** Integration of ChatGPT 4 for providing customer support and assistance through natural language processing.

2.6 IMPROVEMENT PROPOSED IN THE CURRENT SYSTEM

1. VIRTUAL PAINTING

- **Gesture Recognition Refinement:** Refine gesture recognition algorithms to improve accuracy and reliability in detecting hand gestures and movements. This includes optimizing parameters, training models with diverse datasets, and implementing error correction mechanisms to minimize false positives and negatives.
- **Enhanced Drawing Tools:** Expand the range of drawing tools and customization options available to users, including different brush styles, textures, and effects. This allows users to express their creativity more freely and create diverse artworks with unique visual styles.
- **Collaborative Drawing Features:** Implement collaborative drawing features that enable multiple users to work together on a single canvas in real-time. This fosters collaboration and creativity among users, allowing them to co-create artworks, share ideas, and learn from each other.
- **Accessibility and Inclusivity:** Ensure the application is accessible to users with diverse abilities and needs by incorporating accessibility features such as voice commands, keyboard shortcuts, and screen reader support. This promotes inclusivity and ensures that all users can enjoy the drawing experience regardless of their physical or cognitive abilities.
- **Multiple Shapes and Sizes With Different colours:** We have many colors to be selected by the user with easy free hand toching from the screen, From those he can choose shapes like Circle, Triangle, Square and Free Hand with different sizes also set earser sizes with just fingures.
- **Changing Background and saving Images:** User can select any of the background by just pressing the a single key or have a transparent background of his own camera background. He can also save the images easily by just pressing a single button and images will be saved.

2. SUBSCRIPTION MANAGEMENT SYSTEM

- **Enhanced User Experience:** Implement intuitive user interfaces and streamlined workflows to improve the overall user experience for both customers and administrators. This includes optimizing the registration process, simplifying plan selection, and providing clear navigation and feedback throughout the platform.
- **Advanced Analytics and Reporting:** Integrate advanced analytics and reporting capabilities to provide insights into subscription trends, user behavior, and revenue metrics. This enables data-driven decision-making and allows administrators to identify opportunities for growth and optimization.
- **Automation of Routine Tasks:** Implement automation features for routine tasks such as invoice generation, payment reminders, and subscription renewals. This reduces manual overhead and improves efficiency while ensuring timely and accurate processing of transactions.
- **Personalization and Customization:** Introduce personalized recommendations and customizable options for subscription plans based on user preferences, usage patterns, and demographic information. This enhances customer engagement and satisfaction by tailoring the platform to individual needs.
- **Enhanced Security Measures:** Strengthen security measures by implementing multi-factor authentication, encryption of sensitive data, and regular security audits. This helps protect user information and financial transactions, instilling trust and confidence in the platform.

2.7 TESTING OF NEW SYSTEM

1. VIRTUAL PAINTING

- **Hand Detection and Gesture Recognition Testing:**
  - Test the accuracy and reliability of hand detection and gesture recognition algorithms under various lighting conditions, backgrounds, and hand poses.
  - Verify the application's ability to detect and interpret different hand gestures, including shape selection, color selection, and eraser activation.
- **Drawing Functionality Testing:**
  - Test the drawing functionalities of the application, including rendering shapes, lines, and freehand drawings on the canvas.
  - Verify the application's responsiveness and accuracy in capturing user input and rendering drawings in real-time.
- **User Interaction Testing:**
  - Test the user interaction features of the application, including gesture-based controls, shape selection, and color customization.
  - Verify that users can intuitively interact with the application and perform drawing tasks without encountering usability issues or errors.
- **Performance Testing:**
  - Evaluate the performance and responsiveness of the application under different load conditions, including the number of concurrent users and drawing complexity.
  - Measure frame rates, latency, and resource utilization to ensure smooth and fluid drawing experiences**.**
- **Compatibility Testing:**
  - Test the application across different hardware configurations, screen resolutions, and input devices to ensure compatibility and usability.
  - Verify that the application works consistently across desktops, laptops, tablets, and touchscreen devices**.**

2. SUBSCRIPTION MANAGEMENT SYSTEM

- **Unit Testing:**
  - Test individual components, functions, and modules of the subscription management platform to ensure they perform as expected.
  - Verify the correctness of business logic, data validation, and error handling mechanisms.
- **User Acceptance Testing (UAT):**
  - Involve stakeholders and end-users in testing the platform's functionality, usability, and performance.
  - Gather feedback on user interfaces, navigation flows, and overall user experience to identify any issues or areas for improvement.
- **Integration Testing:**
  - Test the integration of different modules and components of the platform to ensure they work together seamlessly.
  - Validate data flow between backend services, frontend interfaces, and external APIs.
- **Regression Testing:**
  - Conduct regression testing to ensure that new updates or changes to the platform do not introduce any unintended side effects or regressions.
  - Re-run previous test cases and verify that existing functionalities continue to work as expected after updates.

2.8 PROJECT SCREENSHOTS

1. VIRTUAL PAINTING

- **MAIN PAGE**

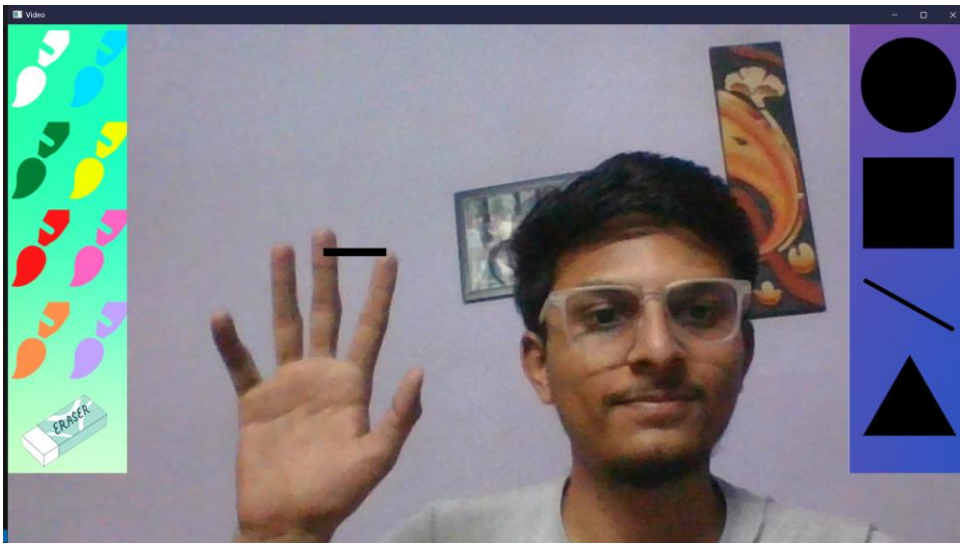


Figure: - 1.0

- **SELECTING AND DRAWING WITH COLORS (FREE HAND)**

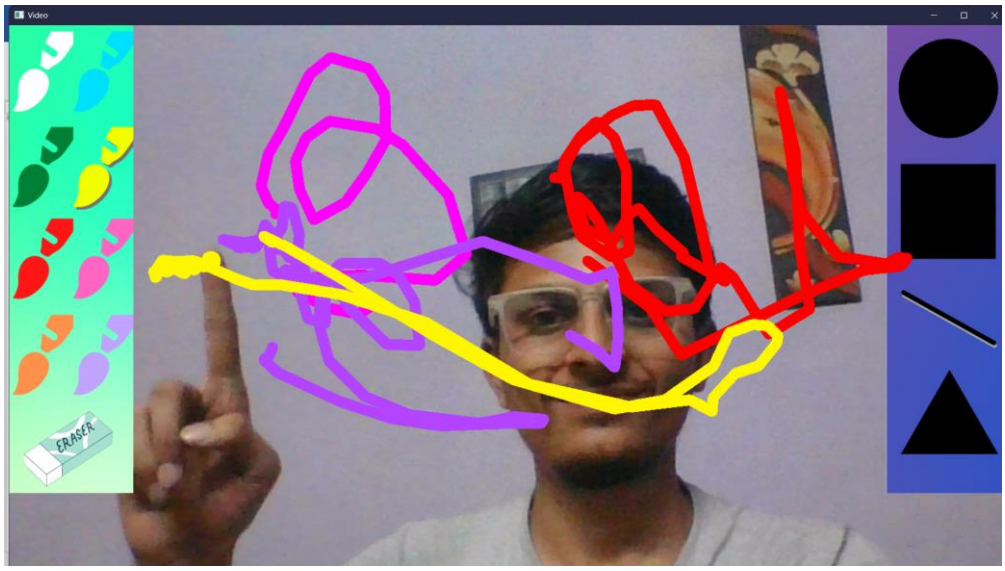


Figure: - 1.1

- **DIFFERENT SHAPES**

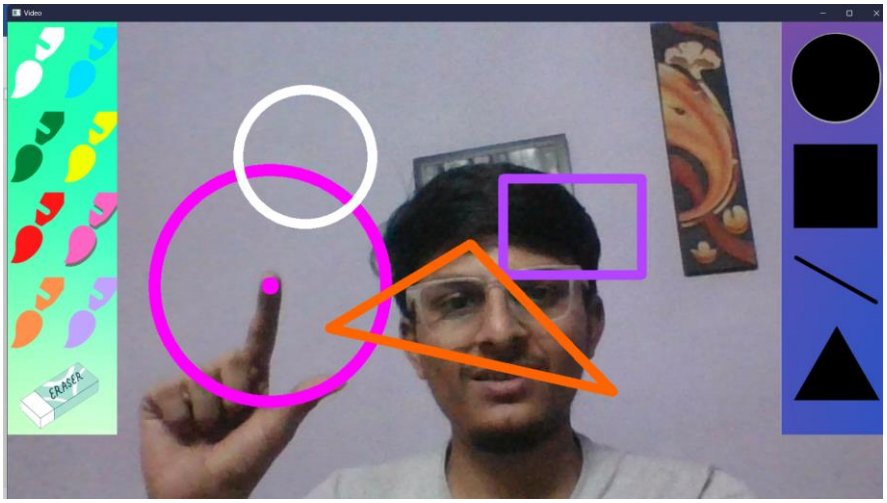


Figure: - 2.0

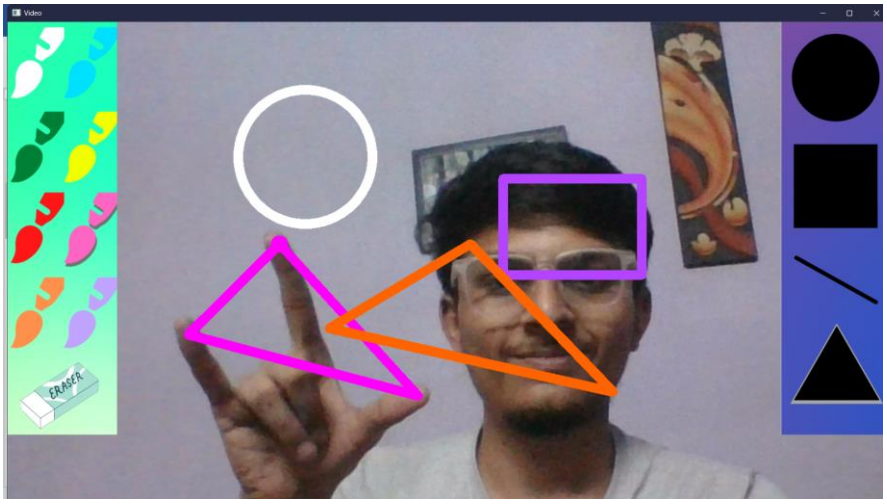


Figure: - 2.1

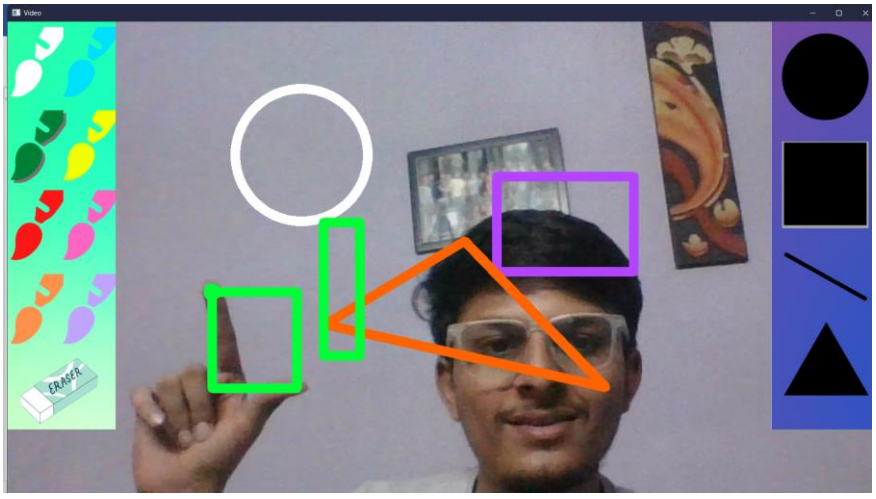


Figure: - 2.2

- **Setting earser shape and earsing**

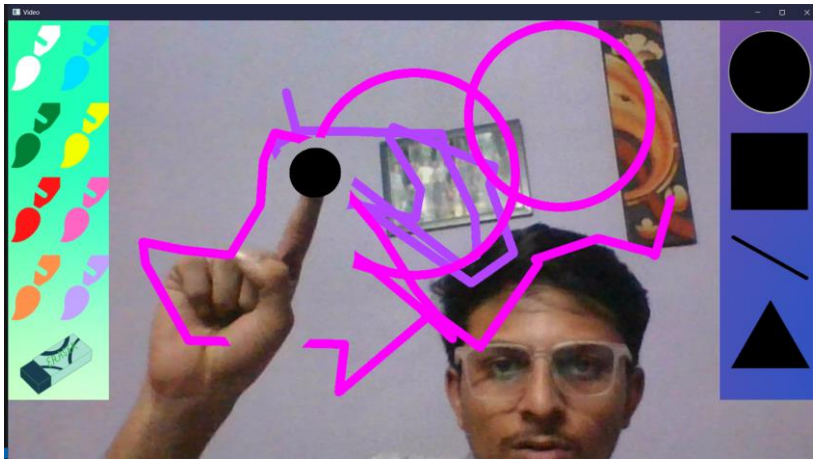


Figure: - 3.0

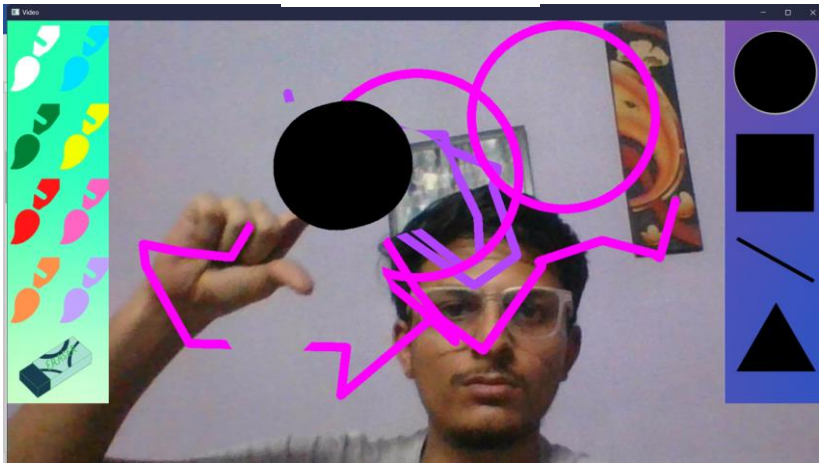


Figure: - 3.1

- **Changing Background For easy Drawing**

Figure: - 4.0

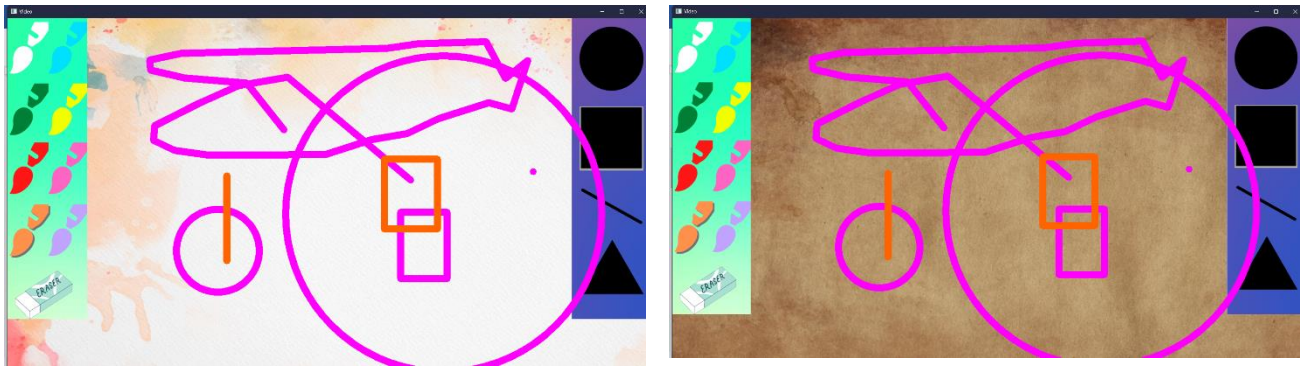


Figure: - 4.1

2. SUBSCRIPTION MANAGEMENT SYSTEM

- **Registeration user page**

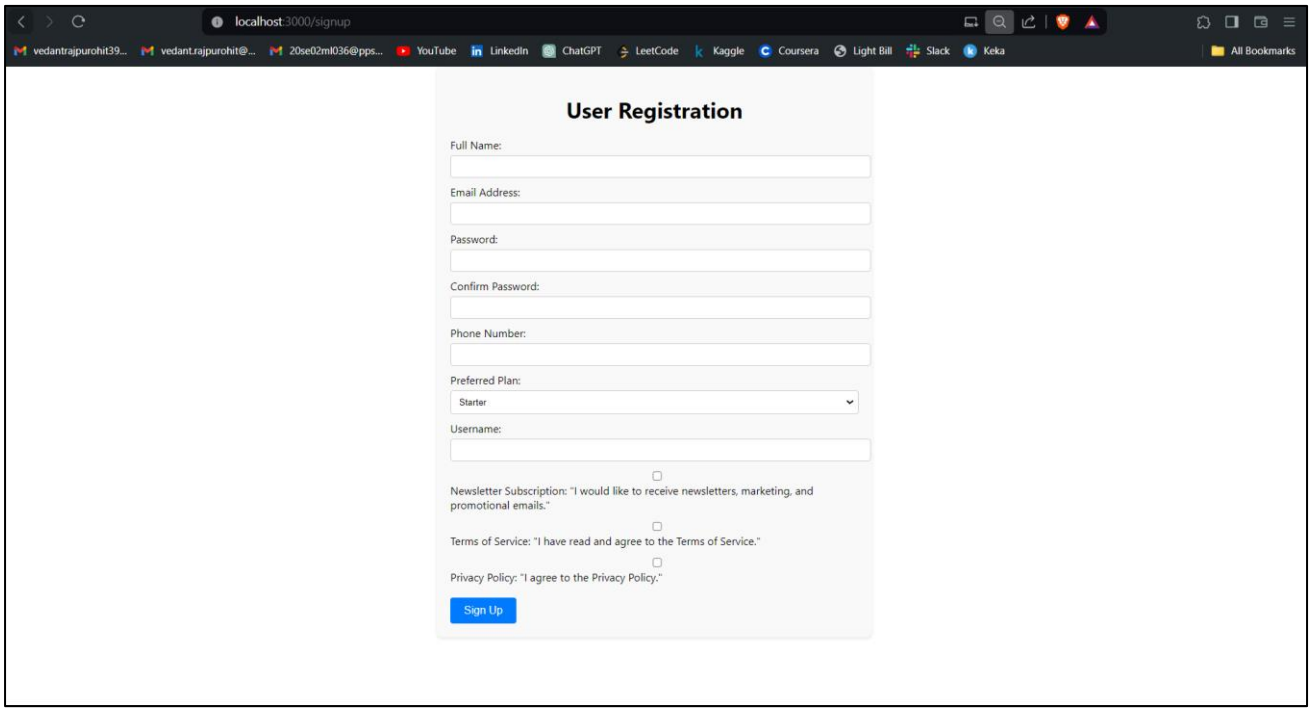


Figure: - 5.0

- **Partner Signup**

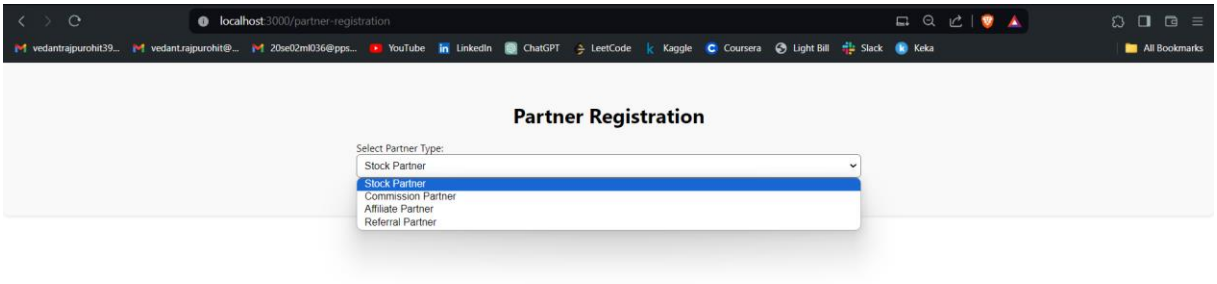


Figure: - 5.1

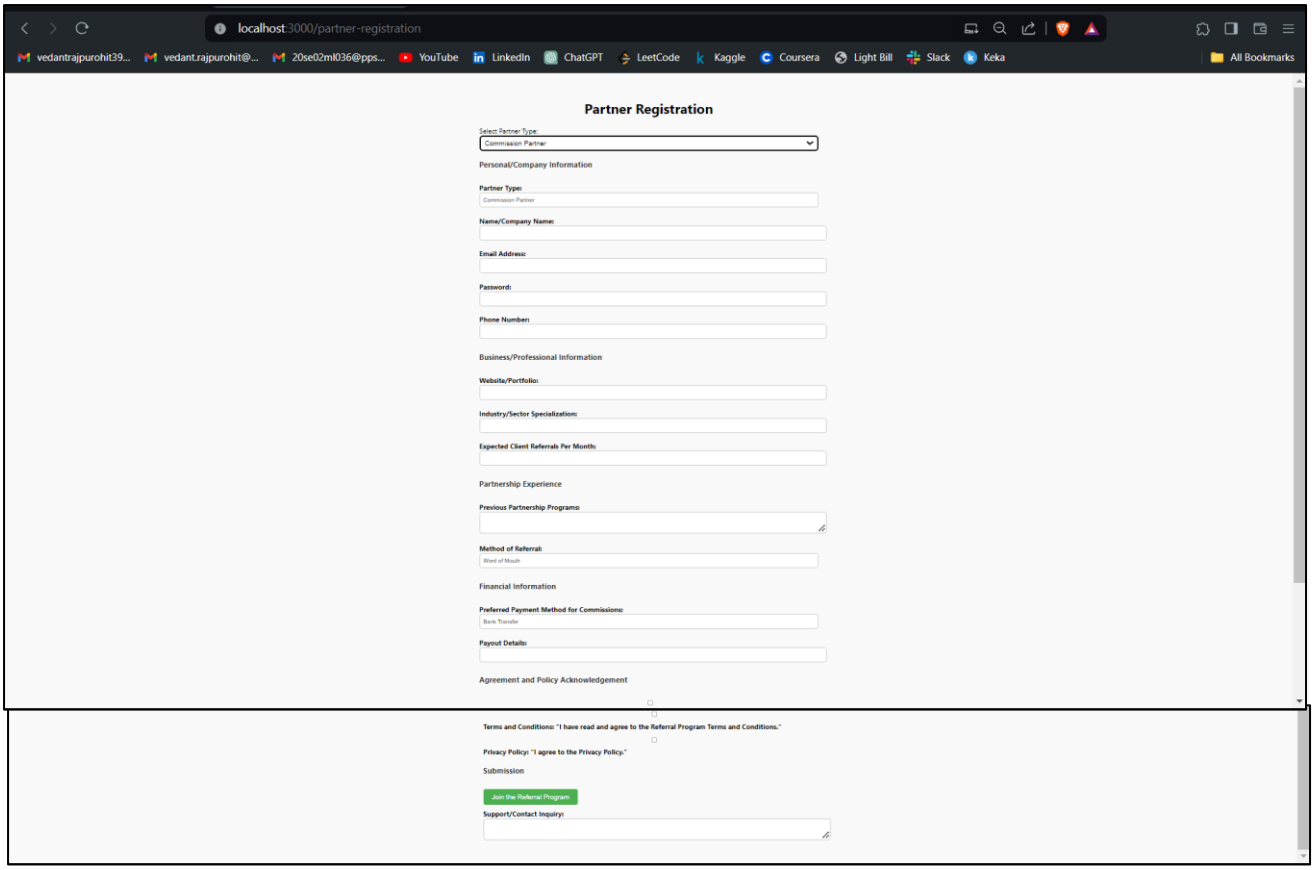


Figure: - 6.0

- **LOGIN**

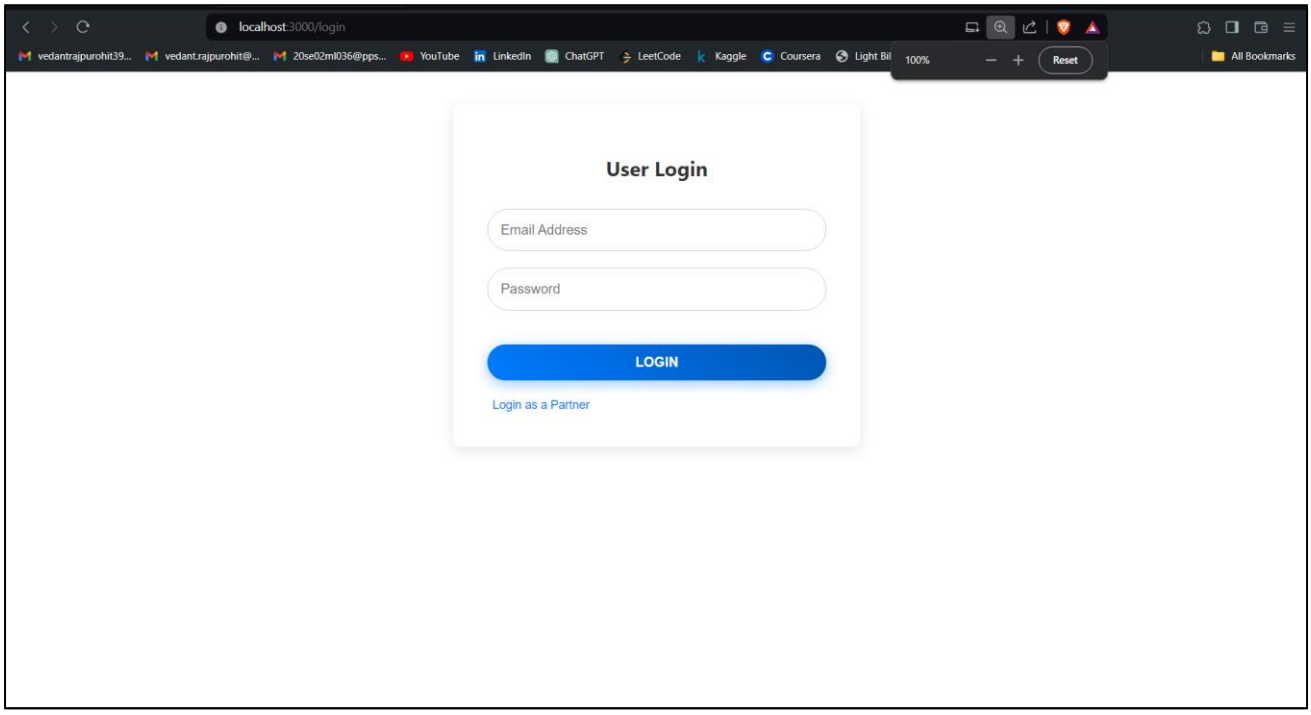


Figure: - 6.1

- **DASHBOARD**

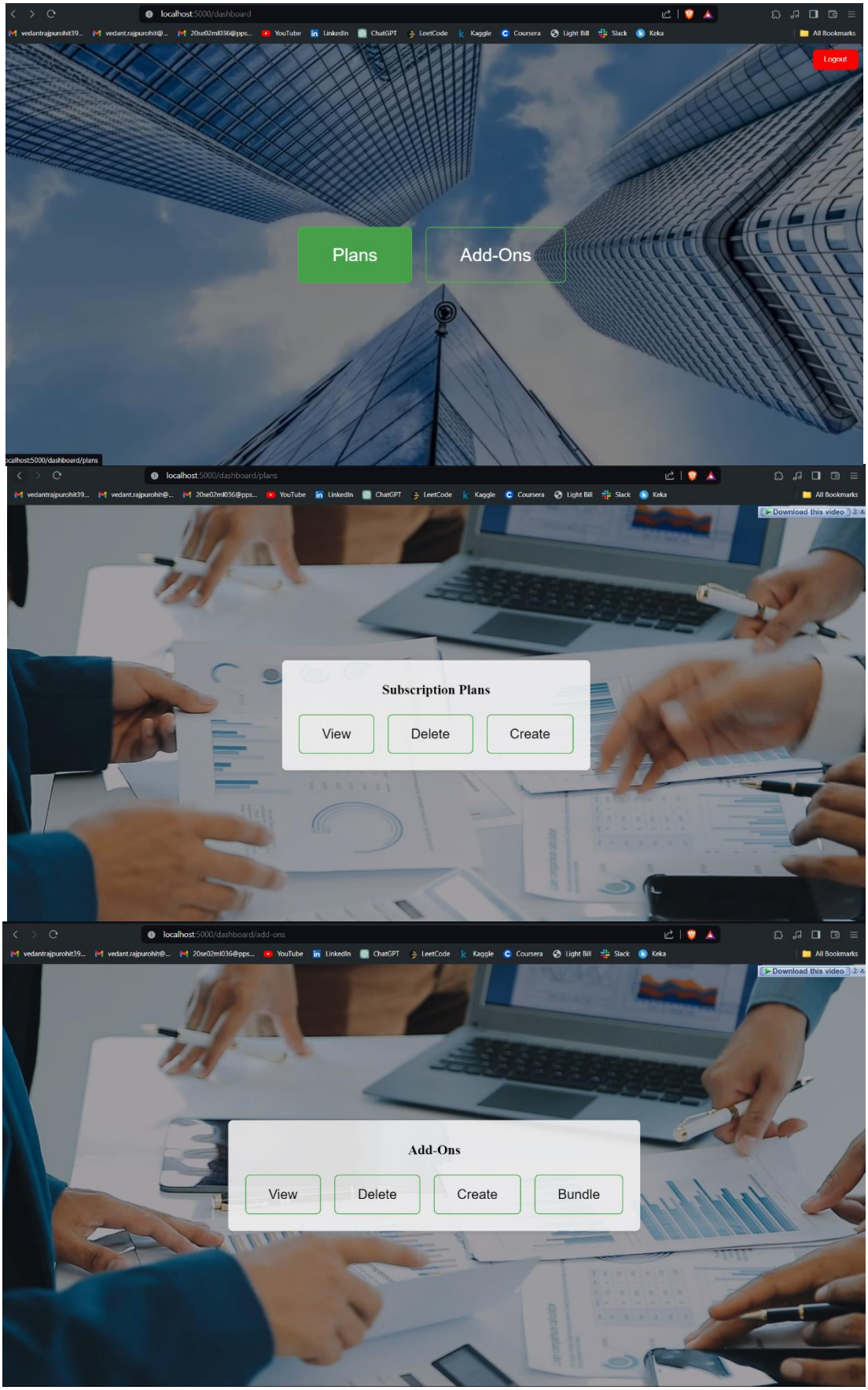


Figure: - 7.0

- **Plans Table (Displaying from Database)**

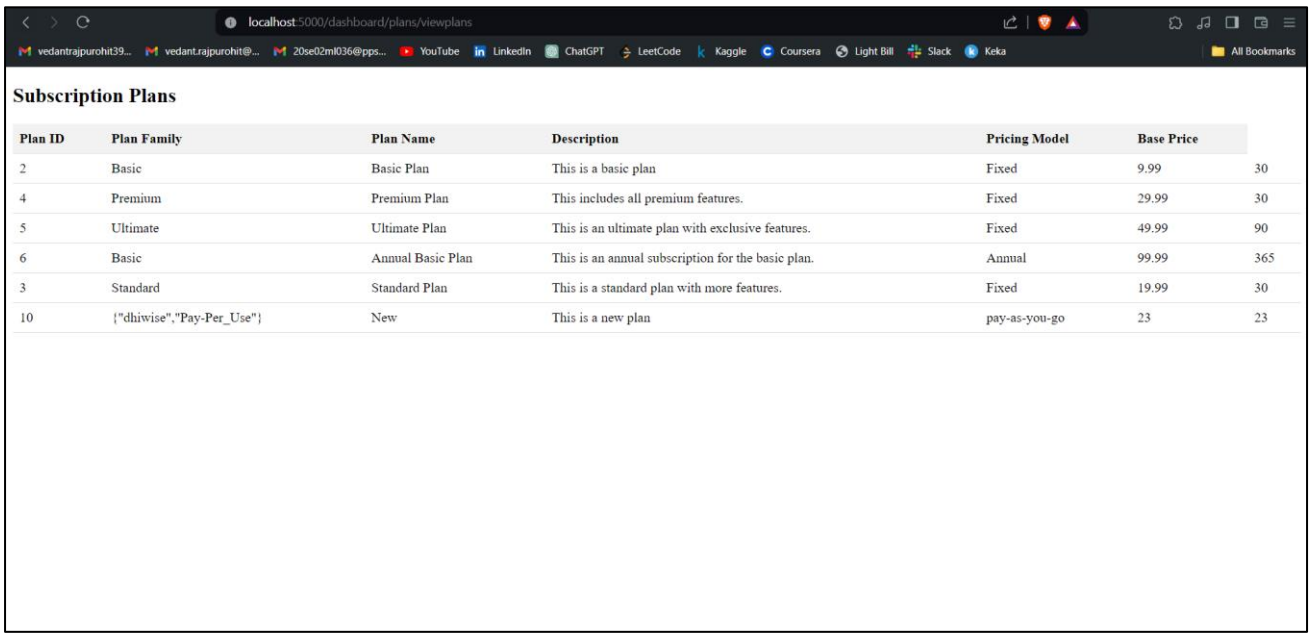


Figure: - 8.0

- **Delete Plan (From Database)**

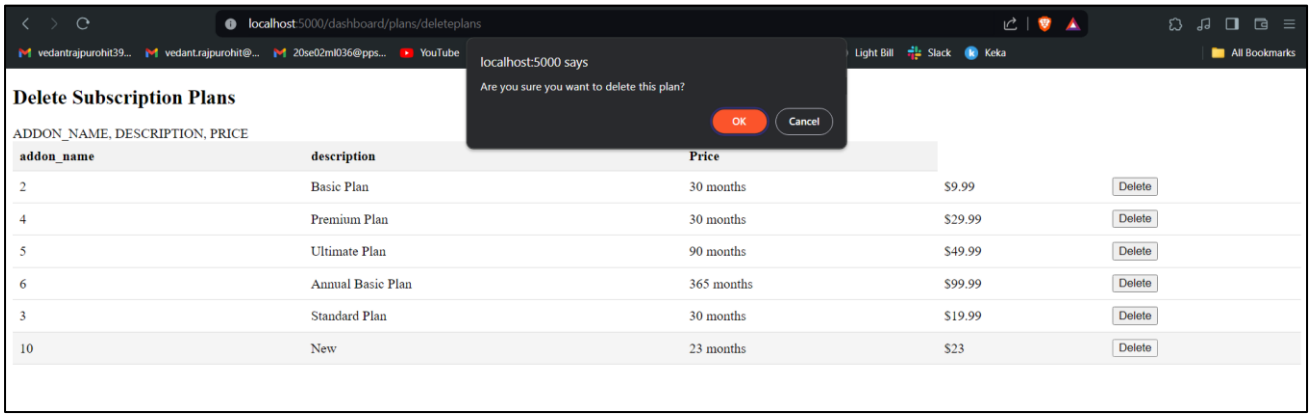


Figure: - 8.1

- **Create Plan**

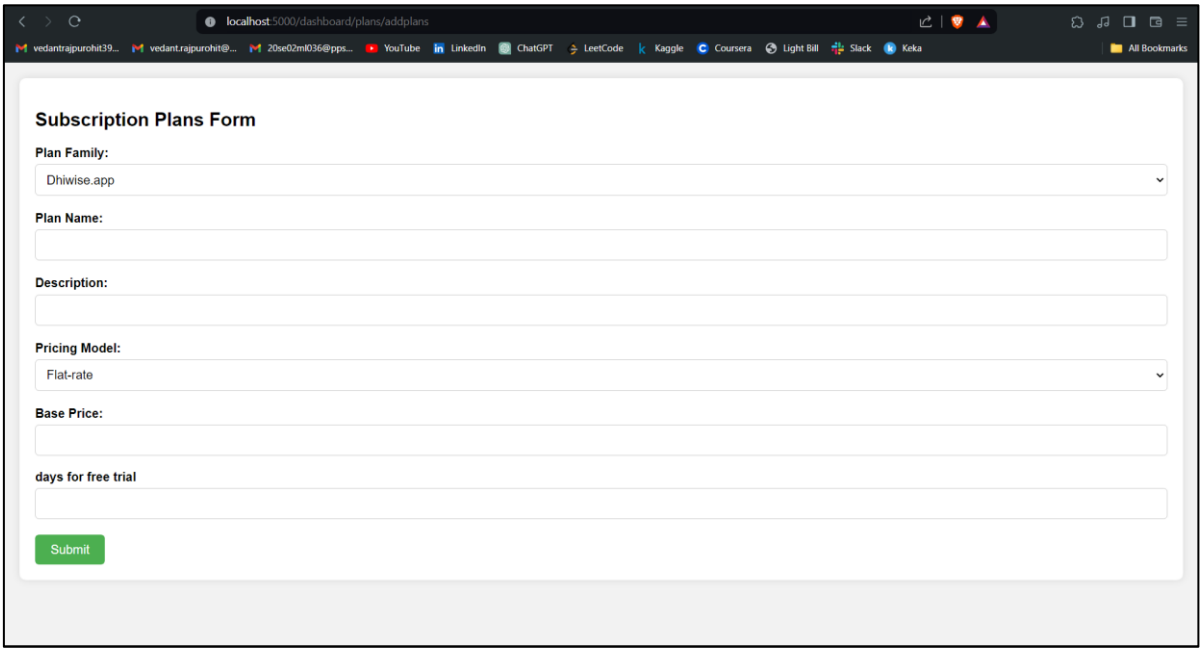


Figure: - 9.0

- **ADD-ons**

Addon ID	Addon Name	Description	Price
1	Extra Storage	Get extra storage space	1.99
2	Additional Users	Add more users to your account	2.99
3	Premium Support	Gain access to premium customer support	4.99
4	Advanced Analytics	Get advanced analytics features	3.99
5	Custom Branding	Add custom branding to your account	5.99
7	a nEW ADDON	tHIS IS A NEW ADDON	123

Figure: - 9.1

- **Create ADD-ons**

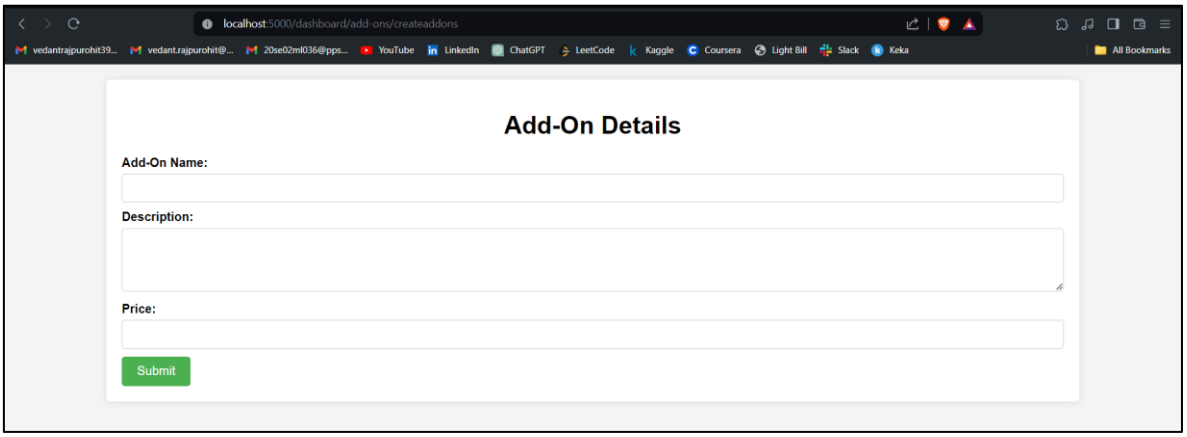


Figure: - 10.0

- **View Subscription and ADD-ons Bundle**

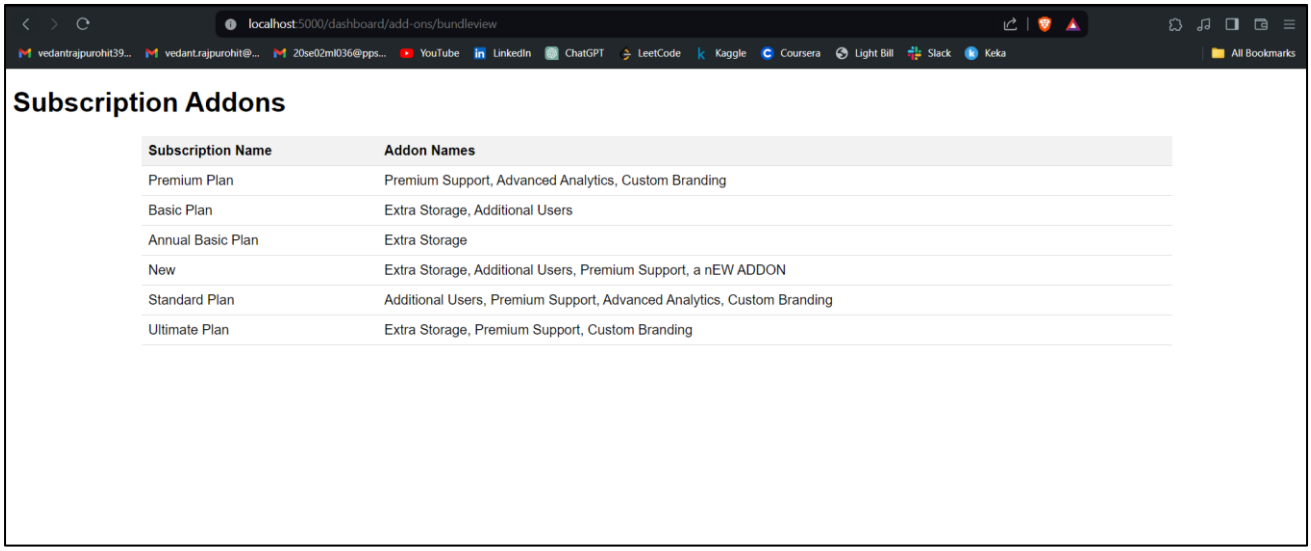


Figure: - 10.1

- **JWT AUTHENTICATION (NO API CALL WITHOUT LOGIN)**

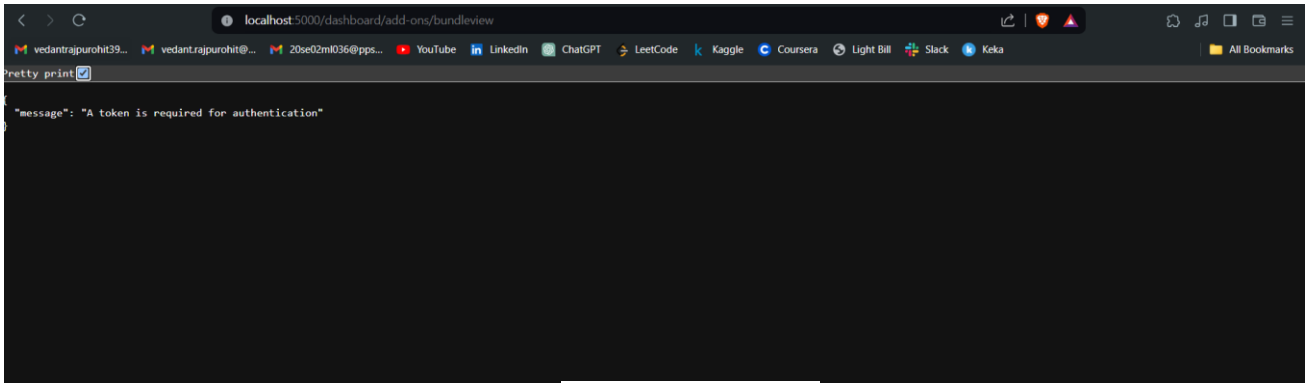


Figure: - 10.2

CHAPTER 3
LEARNING OUTCOME AND DISCUSSIONS

3.1 PROJECT SCREENSHOTS

1. VIRTUAL PAINTING

- **Computer Vision and Image Processing Skills:**
    - Develop proficiency in computer vision techniques and algorithms for hand detection, gesture recognition, and shape tracking using OpenCV.
    - Gain hands-on experience in image processing tasks such as edge detection, contour analysis, and feature extraction to enable real-time interaction with virtual drawings.
- **Algorithmic Thinking and Problem Solving:**
    - Enhance algorithmic thinking skills by designing and implementing custom algorithms for gesture recognition, shape rendering, and user interaction in the virtual drawing application.
    - Learn to analyze complex problems, break them down into smaller subproblems, and devise efficient solutions using algorithmic principles and computational techniques.
- **Creative Expression and Artistic Exploration:**
    - Explore creative expression through digital art and interactive drawing experiences facilitated by the virtual drawing application.
    - Experiment with different drawing tools, colors, and shapes to create unique artworks and explore the possibilities of digital media for artistic expression and communication.

2. SUBSCRIPTION MANAGEMENT SYSYEM

- **Technical Skills Enhancement:**
    - Gain proficiency in using modern web development technologies such as MongoDB/PostgreSQL, Express.js, React.js, and Node.js (MERN stack) for building scalable and interactive web applications.
    - Develop expertise in implementing authentication and authorization mechanisms, integrating third-party APIs for payment processing and email communication, and designing RESTful APIs for seamless integration with external systems.
- **Project Management and Collaboration:**
    - Acquire project management skills through planning, prioritization, and execution of tasks in a collaborative development environment.
    - Learn to work effectively in cross-functional teams, communicate project requirements and progress, and resolve conflicts or challenges that arise during the development lifecycle.
- **Problem Solving and Critical Thinking:**
    - Develop problem-solving skills by analyzing requirements, identifying technical challenges, and implementing solutions to address complex business needs.
    - Cultivate critical thinking abilities by evaluating trade-offs between different technologies, design patterns, and architectural decisions to achieve optimal outcomes for the subscription management platform.

3.2 DETAILS OF REPORTING

- December 30, 2023: 1st Reporting along with Project Title & Scope of Work
- January 13, 2024: 2nd Reporting to Institute with Progress of Work (Online)
- January 27, 2024: First Review of Project Work by Experts
- February 10, 2024: 3rd Reporting to Institute with Progress of Work (Online)
- February 24, 2024: 4th Reporting to Institute with Progress of Work
- March 09, 2024: Second Review of Project/Training Work by Experts
- March 30, 2024: 5th Reporting to Institute & Submission of Training
    - Completion Certificate Signed by Company Person

3.3 GLIMPSE OF PROJECT AND TRAINING

- **INTERNSHIP/TRAINING**

Figure: - 11.0

Figure: - 11.1

- **PROJECTS**

Figure: - 12.0

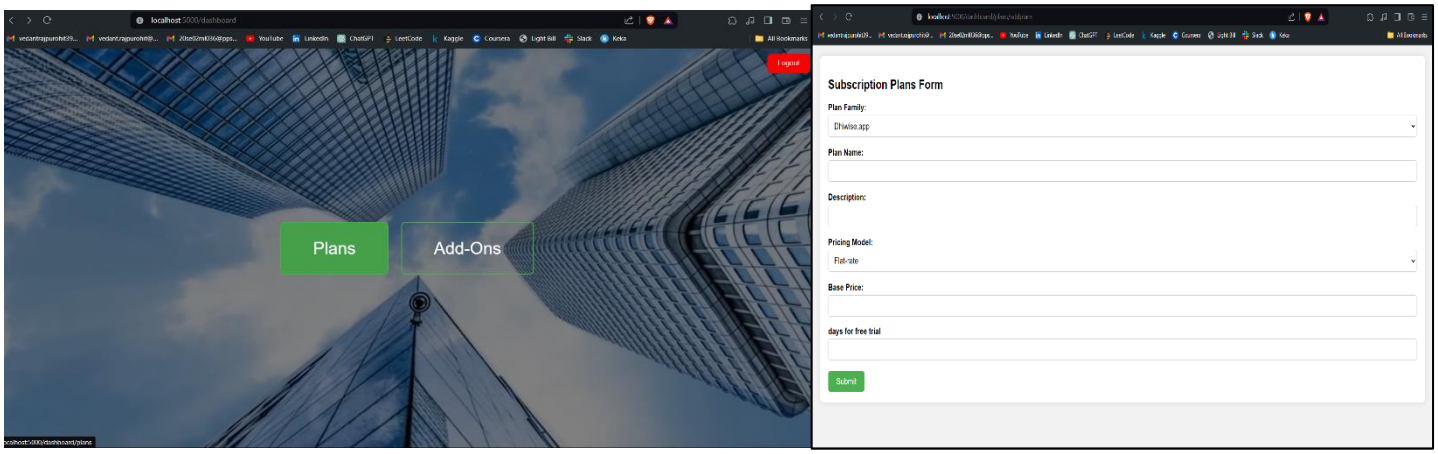


Figure: - 12.1

3.4 LIMITATIONS AND FUTURE IMPROVEMENT

LIMITATIONS:

- **Subscription Management Platform:**
    - Limited support for customization and extensibility, making it challenging to accommodate diverse business requirements and user preferences.
    - Potential security vulnerabilities, especially in handling sensitive user data and financial transactions, which may expose the platform to risks such as data breaches or fraud.
    - Complexity in managing subscription plans and addons, particularly as the business scales and the number of offerings increases, leading to potential usability issues and administrative overhead.
- **Virtual Drawing Application using OpenCV:**
    - Dependency on hardware capabilities and environmental factors, such as camera quality, lighting conditions, and background clutter, which may impact the accuracy and reliability of hand detection and gesture recognition.
    - Limited support for advanced drawing features and artistic tools, such as layers, blending modes, and advanced brush dynamics, which may restrict the creative possibilities and appeal of the application to professional artists and designers.
    - Performance limitations, particularly on low-end devices or systems with limited processing power, which may affect the responsiveness and fluidity of the drawing experience, especially when rendering complex scenes or handling multiple users simultaneously.

FUTURE IMPROVEMENTS

- **Subscription Management Platform:**
  - Implement a modular and extensible architecture that allows for easy customization and integration of new features, addons, and third-party services, enabling greater flexibility and scalability.
  - Enhance security measures by implementing encryption, access controls, and regular security audits to safeguard user data and financial transactions against potential threats and vulnerabilities.
  - Streamline the plan configuration and management process by introducing intuitive interfaces, automation features, and advanced analytics tools to empower administrators and optimize business operations.
- **Virtual Drawing Application using OpenCV:**
  - Invest in research and development to improve the accuracy and robustness of hand detection and gesture recognition algorithms, leveraging advancements in computer vision and machine learning techniques.
  - Expand the range of drawing tools and artistic features available in the application, including support for layers, filters, and custom brushes, to cater to the diverse needs and preferences of users across different skill levels and artistic styles.
  - Optimize performance and resource utilization through code optimization, parallel processing, and hardware acceleration techniques, ensuring smooth and responsive drawing experiences across a wide range of devices and environments.

CHAPTER 4
CONCLUSIONS

In conclusion, the development of the Subscription Management Platform and Virtual Drawing Application using OpenCV represents a significant endeavor aimed at addressing specific challenges and requirements in their respective domains. Throughout the project lifecycle, a multidisciplinary approach was adopted, leveraging a combination of technical expertise, creative problem-solving, and user-centric design principles to deliver innovative and impactful solutions.

The Subscription Management Platform serves as a comprehensive solution for businesses seeking to manage subscription-based services efficiently and effectively. By leveraging modern web development technologies and best practices, the platform offers robust user management, plan configuration, and payment processing capabilities, enabling businesses to streamline their operations, enhance customer satisfaction, and drive revenue growth. However, there remain opportunities for further improvement, including enhancing customization options, strengthening security measures, and optimizing administrative workflows to meet the evolving needs of businesses and users.

Similarly, the Virtual Drawing Application using OpenCV offers an immersive and interactive drawing experience powered by computer vision technology. By combining advanced image processing algorithms with intuitive user interfaces, the application empowers users to unleash their creativity and express themselves through digital art. While the application demonstrates promising capabilities in hand detection, gesture recognition, and shape rendering, there are areas for refinement, such as improving accuracy, expanding drawing features, and optimizing performance across different devices and environments.

In essence, both projects underscore the importance of continuous innovation, collaboration, and user feedback in driving technological advancement and enhancing user experiences. As technology continues to evolve and user expectations evolve, there will be ongoing opportunities for further refinement and iteration, ensuring that both the Subscription Management Platform and Virtual Drawing Application remain at the forefront of their respective domains, delivering value and delight to users worldwide.

CHAPTER 5
REFERENCES

1. Smith, John. "Building Scalable Web Applications with the MERN Stack." O'Reilly Media, 2020.
2. Brown, Sarah. "Mastering React.js: Build Robust and Interactive Web Applications with React." Packt Publishing, 2019.
3. Gonzalez, Miguel. "Hands-On Computer Vision with OpenCV and Python: Build Innovative Computer Vision Applications with OpenCV and Python." Packt Publishing, 2021.
4. Johnson, David. "Learning OpenCV 4 Computer Vision with Python 3: Get to Grips with Tools, Techniques, and Practical Applications of Computer Vision Using Python 3 and OpenCV 4." Packt Publishing, 2018.
5. Webber, Emily. "Mastering Node.js: Build Enterprise-Grade Scalable Web Applications Using Node.js." Packt Publishing, 2019.
6. Goyal, Ravi. "Practical Node.js: Building Real-World Scalable Web Apps." Apress, 2020.
7. Official MongoDB Documentation. Available online: https://docs.mongodb.com/
8. PostgreSQL Documentation. Available online: https://www.postgresql.org/docs/
9. OpenCV Documentation. Available online: https://docs.opencv.org/
10. React.js Documentation. Available online: https://reactjs.org/docs/getting-started.html