AIR SCRIBE DECIPHERER SYSTEM

PROJECT THESIS

SUBMITTED

TO

AWH ENGINEERING COLLEGE

KUTTIKKATTOOR, KOZHIKODE - 8

IN PARTIAL FULFILMENT
OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

OF

Master Of Computer Applications

BY

ARCHANA. T

.2001.

AWH ENGINEERING COLLEGE
CALICUT.KERALA.INDIA

DEPARTMENT OF COMPUTER APPLICATIONS
AWH ENGINEERING COLLEGE KUTTIKKATTOOR,
KOZHIKODE
APRIL 2024

DEPARTMENT OF COMPUTER APPLICATIONS

AWH ENGINEERING COLLEGE

KOZHIKODE

CERTIFICATE

This is to certify that this thesis entitled ***"AIR SCRIBE DECIPHERER SYSTEM"*** *submitted herewith is an authentic record of the thesis work done by* ***ARCHANA. T (AWH22MCA-2011)*** *under our guidance in partial fulfillment of the requirements for the award of* ***Master of Computer Applications*** *from APJ Abdul Kalam Technological University during the academic year 2024.*

Mrs. Sruti Sudevan
Assistant Professor
Dept. of Computer Applications
Head of the Department

Mrs. Sruti Sudevan
Assistant Professor
Dept. of Computer Applications
Project Guide

External Examiner

Internal Examiner

ACKNOWLEDGEMENT

I express my sincere gratitude to our beloved principal **Dr. Sabeena M V** for providing me an opportunity with the required facilities for doing this project. I express my hearty thanks to **Mrs. Sruti Sudevan**, Head of the department of Computer Applications and Assistant professor**,** for her guidance. I am thankful to all other staff of the MCA department for their encouragement, timely guidance, valuable suggestions and inspiring ideas given throughout this project. I am grateful to my friends for the way they have cooperated, expected me to achieve success and have always stirred my ambition to do the best. Above all, I am grateful to the almighty, who has showered His blessings on me throughout my life and throughout the project.

ARCHANA. T

ABSTRACT

Air scribe Decipherer System is a type of hand gesture recognition system, that used to recognize finger movements to create linguistic characters and numbers in free space. Writing in the air with fingers is practically a metaphor of pen-based writing. It is used for detecting intended writing among extraneous finger movements. This system that writes the exact mean of the motion that is drawn in front of the webcam with the help of OpenCV. The output is shown in text format corresponding to the finger movements.

The Air scribe Decipherer System represents advancement in human-computer interaction, enabling text input through mid-air hand gestures. Integrating cutting-edge technologies like computer vision and machine learning, this system interprets these gestures into recognizable characters, and numbers presenting users with a touchless and intuitive means of generating text. Additionally, the system provides features like air canvas for drawing and enable audio for recognized characters, it offers a seamless and engaging user experience while opening up innovative possibilities in education, art, and accessibility.

CONTENTS

INTRODUCTION

1. INTRODUCTION

Air scribe decipherer system is a type of gesture recognition. It provides the feature of air writing, that is the system allows to write in air by the use of finger movements to create linguistic characters or words in free space. Writing in the air with is practically a metaphor of pen-based writing. It is used for detecting intended writing among extraneous finger movements. This system that writes the exact mean of the motion that is drawn in front of the webcam with the help of OpenCV. The output is shown in text format corresponding to the finger movements. Air-writing can be used as an alternative input method for interacting with computers or devices.

Air-writing systems have the potential to significantly improve communication and learning experience. By leveraging gesture recognition technology, these systems offer alternative ways to express oneself, participate in educational activities, and engage with technology, promoting inclusivity and accessibility. In the case individuals, who may have difficulty or be unable to use vocal communication, can use air writing as a gesture-based communication method. They can convey messages, ask questions, or participate in conversations through finger movements.

Also, this system can be used to help in educational tools. The combination of visual and learning through air writing can enhance understanding and retention of information. Air scribe decipherer system can be applied in educational settings to facilitate interactive learning. Teachers or students can use gestures to annotate virtual presentations, illustrate concepts, or interact with educational content in a more engaging manner. Moreover, the system provides a feature that enables audio for recognized characters, enhancing the user experience by providing auditory confirmation of interpreted gestures.

Along with character recognition it also provides air canvas feature, which expands its functionality beyond just text input. Air Canvas allows users to draw in free space using finger movements, essentially creating a virtual canvas in the air. By recognizing and interpreting these gestures, the system accurately translates them into digital drawings, opening up possibilities for artistic expression and creative interaction without the need for physical tools or surfaces.

SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

System analysis is the process of gathering and interpreting facts, problems and using the information to recommend improvements of the system.

2.1 Existing system

Existing air writing systems utilize various technologies such as computer vision, motion sensors, and machine learning algorithms to interpret hand movements in free space as text or commands. These systems typically involve capturing hand gestures through devices like webcams or motion-tracking sensors, analysing the movements, and converting them into readable text or actionable commands.

Disadvantages:

- Real-Time Processing: Achieving real-time processing of gestures can be computationally intensive. Delayed or laggy responses can impact the user experience, especially in applications that require quick and precise interactions.
- Privacy Concerns: Gesture recognition systems may raise privacy concerns, especially if they involve capturing and storing user data. Ensuring the security and privacy of user information is crucial.
- Adaptability to Individual Preferences: Systems might not easily adapt to individual users' preferred writing or gesturing styles. Customization options may be limited, impacting the overall user experience.
- Integration Challenges: Integrating gesture recognition into existing applications or systems may pose challenges. Compatibility issues and the need for additional software or hardware components can be obstacles.
- Cost: Some advanced gesture recognition systems, especially those used in specialized industries or applications, can be expensive. This cost factor may limit widespread adoption.

2.2 Proposed system

The proposed Air scribe Decipherer system could enhance the existing air-writing and character recognition setup by incorporating advanced machine learning algorithms for more accurate and context-aware recognition.

This system aims to translate the exact motion drawn in free space into linguistic characters or words. OpenCV would play a crucial role in processing the video feed from the webcam, extracting relevant information, and recognizing the intended writing gestures. The output is then presented in text format, corresponding to the finger movements. Deep learning models could be trained to understand a vocabulary of character and improve the system's adaptability to individual users' writing styles.

Such advancements aim to refine the precision, expand the functionality, and offer a more interaction with the digital environment, making air-writing a versatile and efficient means of input. This system is cost-effective because it does not require any additional hardware. It allows real time processing and can be easily integrated with existing systems.

The proposed system provides several advantages:

- User friendliness
- Flexibility
- Cost effective
- The system should provide better security and control over data
- The new system combines the services in a single website.
- It is trouble free to use.
- Best user interface.

2.3 Module Description

This project has two modules:

ADMIN:

- Login
- View and verify registered users
- View approved users
- View reviews & ratings from users
- Change password

USER:

- Register
- Login
- View & edit profile
- Enable video capture
- Perform air canvas painting
- Perform air scribe & alphabet recognition
- Perform air scribe & number recognition
- Enable audio for recognized characters
- View, add reviews & ratings
- Change password

2.4 Sprint

Sprint 1

MODULE	TASK	PENDING TASK ANY	HOURS FOR COMPLETION	EXPECTED DATE OF COMPLETION	ACTUAL DATE OF COMPLETION	REASON FOR DEVIATION
Requirement analysis	Observation and brainstorming	-		29/01/2024		-
	Mvc architecture	-				-
	Use case	-				-
	Sequence diagram	-				-
	User stories	-				-
Admin, User	Table design					
	Installing packages					
	Login					
	Testing					

Sprint 2

MODULE	TASK	PENDING TASK ANY	HOURS FOR COMPLETION	EXPECTED DATE OF COMPLETION	ACTUAL DATE OF COMPLETION	REASONFORDEVIATION
Admin	Form design	-				-
	Approve/ Reject Users	-				-
	View approved users	-				-
	View reviews & ratings	-				-
	Change password	-				-
	Template	-				-
	Validation	-				-

Sprint 3

MODULE	TASK	PENDING TASK ANY	HOURS FOR COMPLETION	EXPECTED DATE OF COMPLETION	ACTUAL DATE OF COMPLETION	REASONFORDEVIATION
Customer	Form design					
	Registration	-				-
	View profile	-				-
	Edit profile					
	Enable video capture	-				-
	Air canvas painting	-				-
	Air writing & alphabet recognition	-				-
	Air writing & number recognition	-				-
	Enable audio	-				-
	View reviews & rating	-				-
	Add reviews & rating	-				-
	Change password	-				-
	Template	-				-
	Validation	-				-

2.5 User Stories

- As an admin, I should able to login into the air scribe decipherer system.
- As an admin, I should able to approve or reject the registered users.
- As an admin, I should able to view the approved users.
- As an admin, I should able to view reviews and ratings from users
- As an admin, I should able to change my password.

- As a user, they should able to register to the platform by providing personal information and credentials.
- As a user, they should able to log in to account using username and password.
- As a user, they should able to view and edit their profile.
- As a user, they should able to perform air scribe by activating video capturing.
- As a user, they should able to draw on air canvas.
- As a user, they should able to air scribe and preform alphabet recognition.
- As a user, they should able to air scribe and preform number recognition.
- As a user, they should able to enable audio for recognized characters.
- As a user, they should able to add and view reviews and ratings.
- As a user, they should able to change their password.

FEASIBILITY STUDY

3. FEASIBILITY STUDY

An analysis of the ability to complete a project successfully, taking into account legal, economic, technological, scheduling, and other factors is considered a feasibility study. Rather than just diving into a project and hoping for the best, feasibility study allows project managers to investigate the possible negative and positive outcomes of a project before investing too much money and time.

3.1 Economic Feasibility

The economic analysis is done to determine the benefits and savings that are expected from the candidate system and compare them with costs. Thus, coming to a conclusion on whether the system is economically feasible or not. This study presents tangible and intangible benefits from the project by comparing the developments and operational costs. The technique of cost benefit analysis is often used as a basis for assessing economic feasibility.

Air scribe decipherer system including air writing for character recognition, air canvas for virtual drawing, and enabling audio for enhanced user experience, the system demonstrates remarkable cost-effectiveness and time efficiency. Consequently, it emerges as economically feasible, offering substantial benefits in communication, education, and creative expression while mitigating operational costs and resource utilization.

3.2 Technical Feasibility

The technical requirements for the Air scribe decipherer system prioritize economic efficiency and do not necessitate the use of additional software. This means that the system can be implemented utilizing existing technologies. This application is developed using Python, leveraging its readily available and cost-free development kits. Other libraries used, such as OpenCV, MediaPipe, etc. are free and open-source. This aspect ensures the technical feasibility of the system, making it accessible and straightforward to implement without incurring significant additional expenses.

3.3 Operational feasibility

The Air scribe decipherer system working is quite easy to use and learn due to its simple but attractive interface. Users require no special training for operating the system. Technical performance encompasses evaluating the system's capability to accurately interpret air-written characters. Acceptance revolves around the current system and its personnel

3.4 Behavioral feasibility

This analysis entails examining the system's functionality upon installation and assessing the political and managerial landscape in which it will be deployed. People typically exhibit resistance to change, although computers have often facilitated such transitions. Given the practical utility of the proposed air writing system, it is anticipated to garner wide acceptance among users, thus appealing to a broad audience.

3.5 Software feasibility

This system, developed in a high-end software environment, seamlessly adapts to various environments with minimal adjustments. It is fully compatible with different operating systems and browsers, ensuring versatile execution.

3.6 Hardware feasibility

This system can be developed with the existing resources. The air scribe decipherer system is hardware-wise feasible because it system that relies solely on a webcam and does not needs absolutely no new hardware.

SOFTWARE ENGINEERING PARADIGM

4. SOFTWARE ENGINEERING PARADIGM

The software engineering paradigm which is also referred to as a software process model or Software Development Life Cycle (SDLC) model is the development strategy that encompasses the process, methods and tools. SDLC describes the period of time that starts with the software system being conceptualized.

4.1 Agile model

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. At the end of the iteration, a working product is displayed to the customer and important stakeholders. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks.

At the end of the iteration, a working product is displayed to the customer and important stakeholders. Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Agile software development is an umbrella term for a set of frameworks and practices based on the values and principles expressed in the Manifesto for Agile Software Development and the 12 Principles behind it. When user approach software development in a particular manner, it’s generally good to live by these values and principles and use them to help figure out the right things to do given users particular context. One thing that separates Agile from other approaches to software development is the focus on the people doing the work and how they work together. Solutions evolve through collaboration between self-organizing cross-functional teams utilizing the appropriate practices for their context.

Scrum

Scrum is an agile framework for managing knowledge work, with an emphasis on software development. It is designed for teams of three to nine members, who break their work into actions that can be completed within time boxed iterations, called "sprints", no longer than one month and most commonly two weeks, then track progress and re-plan in 15-minute stand-up meetings, called daily scrums.

Scrum is an iterative and incremental framework for managing product development. It defines "a flexible, holistic product development strategy where a development team works as a unit to reach a common goal", challenges assumptions of the "traditional, sequential approach to product development, and enables teams to self-organize by encouraging physical co-location or close online collaboration of all team members, as well as daily face-to-face communication among all team members and disciplines involved.

In the project management, scrum, sometimes written Scrum or SCRUM, is a framework for developing, delivering, and sustaining products in a complex environment, with an initial emphasis on software development, although it has been used in other fields including research, sales, marketing and advanced technologies. It is designed for teams of ten or fewer members, who break their work into goals that can be completed within time-boxed iterations, called sprints, no longer than one month and most commonly two weeks. The scrum team assess progress in time-boxed daily meetings of 15 minutes or less, called daily scrums (a form of stand-up meeting). At the end of the sprint, the team holds two further meetings: the sprint review which demonstrates the work done to stakeholders to elicit feedback, and sprint retrospective which enables the team to reflect and improve.

A key principle of Scrum is the dual recognition that customers will change their minds about what they want or need and that there will be unpredictable challenges-for which a predictive or planned approach is not suited. As such, Scrum adopts an evidence based empirical approach accepting that the problem cannot be fully understood or defined up front, and instead focusing on how to maximize the team's ability to deliver quickly, to respond to emerging requirements, and to adapt to evolving technologies and changes in market conditions. Many of the terms used in Scrum (e.g., scrum master) are typically written with leading capitals (e.g., Scrum Master) or as conjoint words written in camel case (e.g., Scrum Master). To maintain an encyclopedic tone, however, this article uses normal sentence case for these terms-unless they are recognized marks. This is occasionally seen written in all -capitals, as SCRUM. The word is not an acronym, so this is not correct; however, it likely arose due to an early paper by Ken Schwaber which capitalized SCRUM in its title. While the trademark on the term Scrum itself has been allowed to lapse, so that it is deemed as owned by the wider community rather than an individual, the leading capital is retained-except when used with other words.

SYSTEM REQUIREMENT SPECIFICATION

5. SYSTEM REQUIREMENT SPECIFICATION

5.1 Software Requirements

- Operating system: windows 7 or Above / Linux new versions
- Front end: HTML, CSS
- Backend: MYSQL
- Language used: python
- IDE: JETBRAINS PYCHARM
- Frame work: Python Django

5.2 Hardware Requirements

• Processor	: intel core i3 or Above
• RAM	: 8 GB
• Storage	: 256 GB SSD

5.3 ML Libraries

- OpenCV
- MediaPipe
- Pygame
- TensorFlow

SYSTEM DESIGN

6. SYSTEM DESIGN

System design is the first in the development phase for many engineered products or systems. It may define the process of applying various techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realisation.

6.1 Database design

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data.

In the relational model these are the tables and views. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system. The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Usually, the designer must: Determine the relationships between the different data elements and superimpose a logical structure upon the data on the basis of these relationships.

Normalisation

Normalization is the procedure of converting a relation into a standard form to address data redundancy and maintain data integrity by eliminating anomalies through the decomposition of relations. To do this normal forms or rules for structuring relations are used.

Insertion anomaly: Inability to add data to the database due to absence of other data.

Deletion anomaly: Unintended loss of data due to deletion of other data. **Update anomaly:** Data inconsistency resulting from data redundancy and partial update. **Normal forms:** These are the rules for structuring relations that eliminate anomalies.

1. **First normal form (1NF)**

   A relation is in first normal form when all attribute values are atomic, ensuring that no attribute contains a set of values or a repeating group.

2. **Second normal form (2NF)**

   A relation is in the second normal form if, in addition to being in the first normal form, it meets one of the following conditions:

   - Primary key is a not a composite primary key
   - No non key attributes are present
   - Every non key attribute is fully functionally dependent on full set of primary keys

3. **Third normal form (3NF)**

   A relation is said to be in third normal form if there exist no transitive dependencies.

Transitive Dependency: Transitive dependence occurs when two non-key attributes depend on each other and the primary key. Applying this normalization principle involves decomposing data into multiple tables to maintain consistent data.

6.2 TABLES

Login

Field Name	**Datatype**	**Description**
LOGIN_id	int(11)	Primary Key
Username	varchar(50)	Not Null
Password	varchar(50)	Not Null
Type	varchar(50)	Not Null

User

Field Name	**Datatype**	**Description**
User_id	int	Primary key
Name	varchar(50)	Not null
Email	varchar(50)	Not null
Phone	varchar(50)	Not null
DOB	varchar(50)	Not null
Gender	varchar(50)	Not null
Status	varchar(50)	Not null
Login_id	int	Foreign key

Review

Field Name	**Datatype**	**Description**
Review_id	int	Primary key
Rating	int	Not null
Review	varchar(200)	Not null
User_id	int	Foreign key

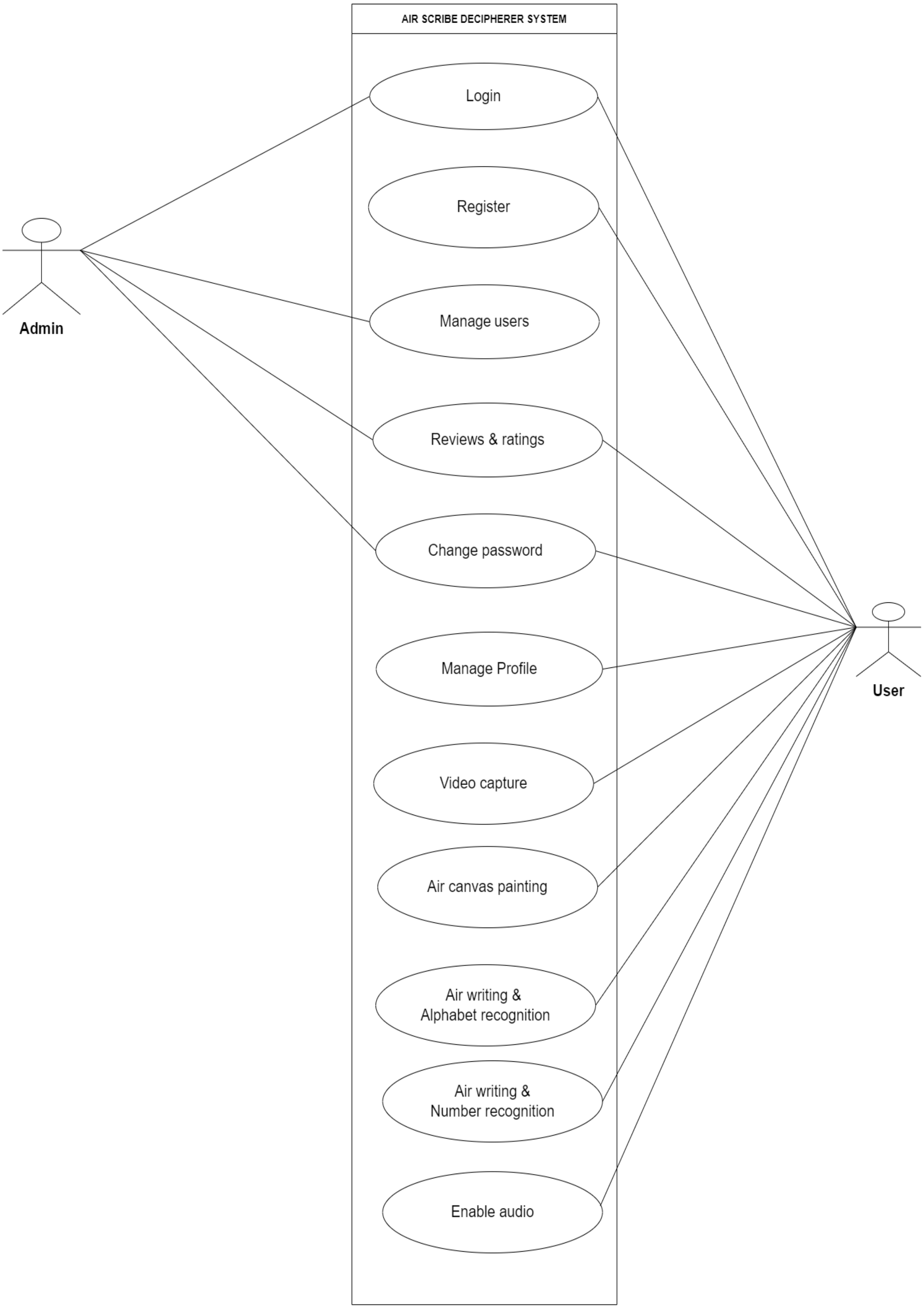
6.3 UML Designs

The Unified Modelling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artefacts of the software systems, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing object-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

A sequence diagram is a type of UML diagram that visualizes the interactions and message exchanges between different objects or components in a system over a specific period of time. It shows the flow of control and the order of message invocations, allowing to understand the dynamic behavior of the system. Sequence diagrams are commonly used to model the behavior of a single use case or a specific scenario.

A use case diagram is a type of UML diagram that represents the functionality of a system from the user's perspective. It provides a high-level view of the system's behavior and shows how users or actors interact with the system to accomplish specific goals or tasks. Use case diagrams are useful for capturing and visualizing the requirements of a system and identifying the actors involved and the use cases they participate in.

6.4 Use Case Diagram



6.5 Scenario

Admin:

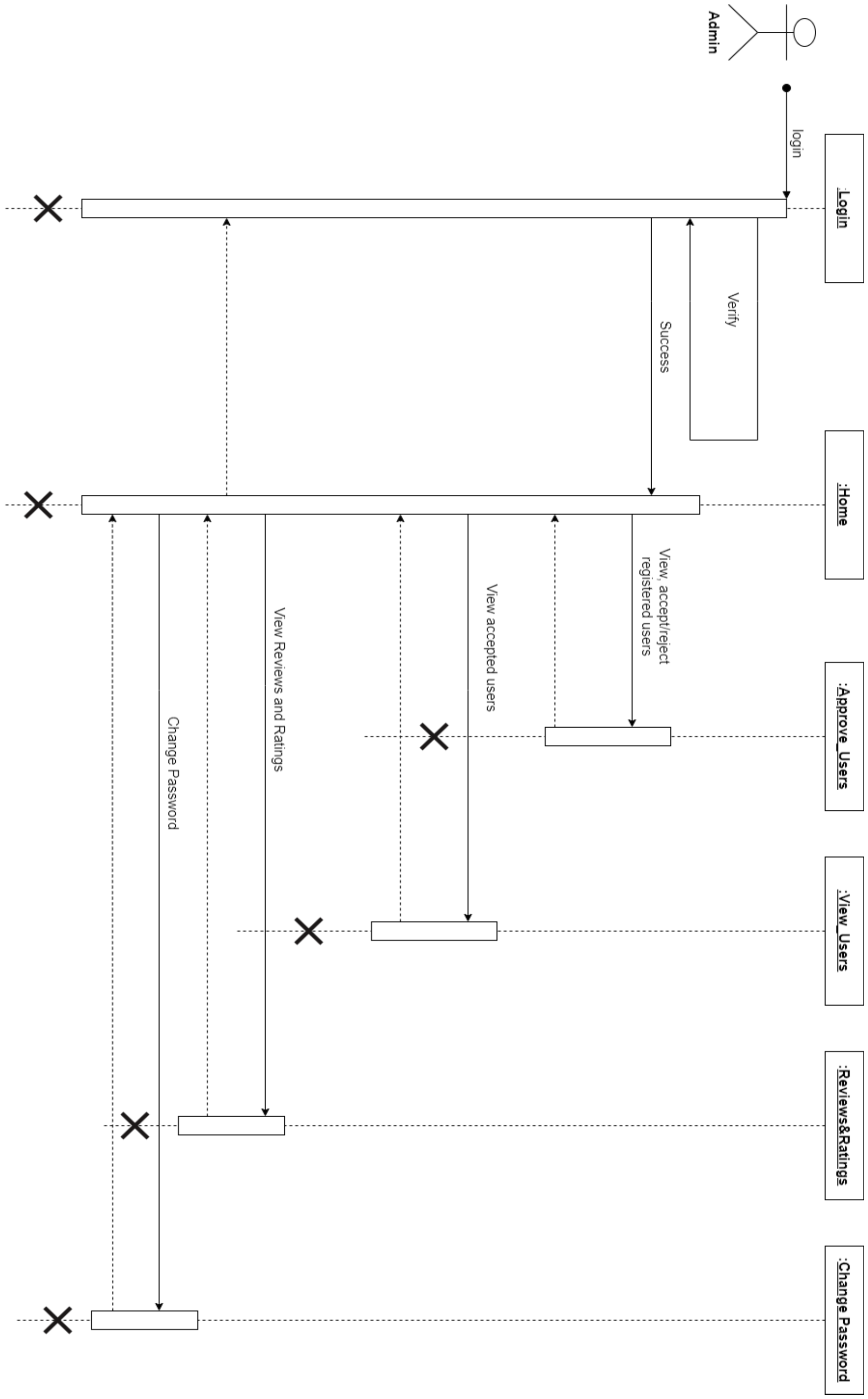
- Can login
- Can approve or reject registered users
- Can view approved users
- View reviews and ratings from users
- Can change password

Customer:

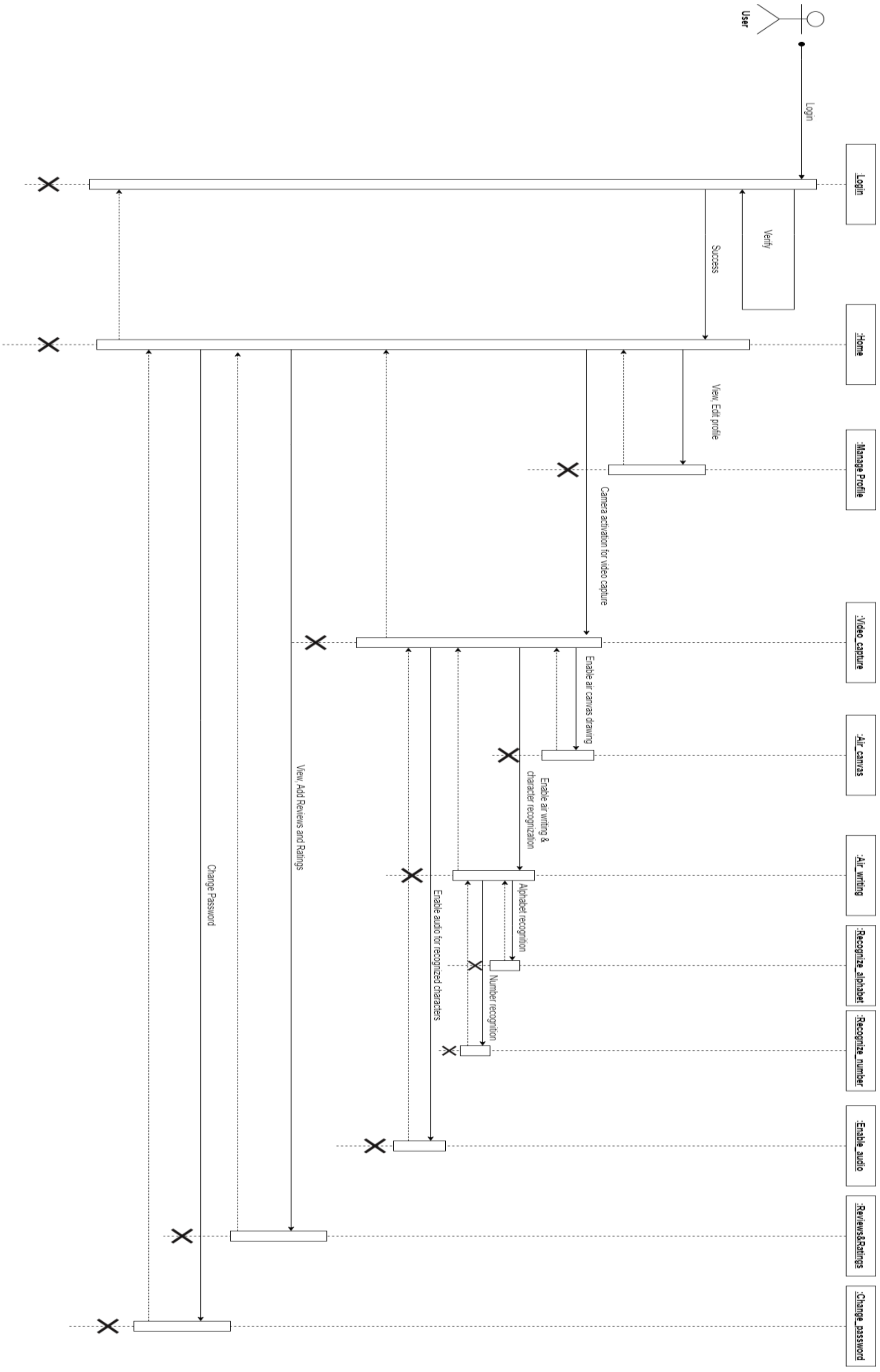
- Can register
- Can login
- Can view and edit profile
- Can activate video capture
- Can draw on air canvas
- Can perform air writing with alphabet recognition
- Can perform air writing with number recognition
- Can enable audio for recognized characters
- Add and view reviews and ratings
- Can change password

6.6 Sequence Diagram

Admin



User


User
Login
:Login
Verify
Success
:Home
View, Edit profile
:Manage Profile
Camera activation for video capture
:Video_capture
Enable air canvas drawing
:Air_canvas
Enable air writing & character recognization
:Air_writing
Alphabet recognition
:Recognize_alphabet
Number recognition
:Recognize_number
Enable audio for recognized characters
:Enable_audio
View, Add Reviews and Ratings
:Reviews&Ratings
Change Password
:Change_password

SYSTEM DEVELOPMENT

7. SYSTEM DEVELOPMENT

System development is a series of operations aimed at manipulating data to generate computer system output, with the primary activities during the development phase categorized into two major sequences:

- External system development
- Internal system development.

The key external system activities are:

- Implementation
- Planning,
- Equipment acquisition
- Installation.

7.1 Coding

The purpose of code is to facilitate the identification and retrieval of items of information. A code is an ordered collection of symbols designed to provide unique identification of entity or an attribute. Code also show interrelationship among different items. Codes are used to identify, access, sort, matching records. The code ensures that only one value of code with a single meaning is applied to give entity or attribute as described in various ways.

Python

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It was created by Guido van Rossum and first released in 1991, with subsequent versions continuously improving the language's features and capabilities. Python's design philosophy emphasizes code readability and simplicity, making it an ideal language for both beginners and experienced developers.

One of Python's defining features is its clean and expressive syntax, which prioritizes human readability. This means that Python code is typically easier to understand and write, reducing the time and effort required for development and maintenance. Python's syntax uses indentation to delimit blocks of code, rather than relying on braces or keywords, further enhancing readability.

Python is an interpreted language, meaning that code is executed line by line by an interpreter, rather than compiled into machine code before execution. This enables rapid development and testing, as changes to the code can be immediately run and evaluated. Python also supports interactive mode, allowing developers to enter commands and see the results in real-time, which is particularly useful for experimenting and prototyping.

Another key feature of Python is its support for multiple programming paradigms, including procedural, object-oriented, and functional programming. This versatility allows developers to choose the most appropriate paradigm for solving different types of problems, making Python suitable for a wide range of applications. Python's standard library is extensive, providing a rich set of modules and packages for various tasks, such as file I/O, networking, database access, web development, and more. This comprehensive standard library reduces the need for external dependencies and accelerates development by providing pre-built solutions to common programming challenges. Python's dynamic typing system allows variables to be assigned without explicitly specifying their data type, making the language more flexible and adaptable. Automatic memory management via garbage collection simplifies memory allocation and deallocation, reducing the risk of memory leaks and other memory-related errors.

Python has a large and active community of developers who contribute to its development, share knowledge, and create third-party libraries and frameworks. This vibrant ecosystem further extends Python's capabilities and facilitates collaboration and learning. As a result, Python is widely used across various domains, including web development, data science, machine learning, artificial intelligence, automation, scientific computing, and more. Its versatility, ease of learning, and robustness make it a preferred choice for a wide range of projects and applications.

HTML

HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It defines the content and structure of web content. It is often assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes, and other items.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. The inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997.

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of content and presentation, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

Bootstrap

Bootstrap is a widely used front-end framework that facilitates the development of responsive and mobile-first websites and web applications. Originally developed by Twitter, Bootstrap offers a comprehensive set of pre-designed HTML, CSS, and JavaScript components aimed at simplifying the process of building attractive and consistent user interfaces.

With an extensive collection of CSS components, including buttons, forms, navigation bars, and alerts, Bootstrap streamlines UI development by furnishing pre-styled elements that are readily deployable. Complemented by JavaScript plugins like modal dialogs, carousels, and tooltips, Bootstrap enriches web experiences with interactive functionality while maintaining simplicity in implementation. Embracing a mobile-first approach, Bootstrap prioritizes the design and functionality for smaller screens, progressively enhancing user experience for larger devices.

Moreover, Bootstrap's customizable nature empowers developers to tailor the framework to specific project requirements, leveraging its extensive documentation and vibrant community support for guidance and collaboration. In essence, Bootstrap stands as a versatile solution, empowering developers to craft modern and responsive web applications efficiently, thereby contributing significantly to the evolution of web development practices.

Django

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, this allows developers to concentrate on application development without the necessity of reinventing fundamental components. OpenCV, being both free and open-source, benefits from a vibrant and engaged community, extensive documentation, and various options for support services, both free and paid.

Django includes a built-in administration interface that automatically generates CRUD (Create, Read, Update, Delete) interfaces for managing data models. This feature, known as the Django Admin, simplifies the process of creating administrative panels for managing site content, making it easy to perform common administrative tasks without writing additional code. Its powerful URL routing system, which allows developers to map URLs to views using regular expressions or simple path patterns. This enables clean and intuitive URL structures for web applications and facilitates the creation of RESTful APIs (Application Programming Interfaces) for serving data to client applications.

Django comes with a secure authentication system, user management, and authorization features out of the box, providing robust tools for managing user accounts, permissions, and access control. Additionally, Django's built-in security features include protection against common web vulnerabilities such as CSRF (Cross-Site Request Forgery), XSS (Cross-Site Scripting), and SQL injection attacks. Django's extensive ecosystem of reusable components, known as "Django apps," provides a wide range of functionality for common web development tasks. These apps cover areas such as user authentication, session management, form handling, file uploading, caching, internationalization, and more. Developers can leverage these pre-built components to accelerate development and focus on building the unique features of their applications.

Libraries

Python's extensive standard library, a key strength, offers a diverse set of tools for various tasks. It supports common formats and protocols like MIME and HTTP for Internet-facing applications. The library includes modules for tasks such as creating graphical user interfaces, connecting to databases, generating random numbers, working with precision decimals, handling regular expressions, and conducting unit testing.

SQLyog

SQLyog is a powerful GUI tool to manage MySQL and MariaDB servers and databases in physical, virtual, and cloud environments. SQLyog is developed by Webyog Inc. SQLyog was available free of charge, but with closed source code, until v3.0 when it was made a fully commercial software. SQLyog's intuitive graphical user interface makes managing all aspects of MySQL databases easy. Simple operations can be accomplished using the many pre-defined tools and functions while more complicated tasks can be built using the graphical editor which helpfully generates queries in correct SQL syntax to perform and learn from. SQLyog is able to handle databases of all sizes and can use SSH an HTTP tunnelling to make remote access simple and secure. Data transfer from external ODBC-compliant databases is also easy using the in-built import tool.

Prominent features of SQLyog are:

- 64-bit binaries are available from version 11.0.
- Editor with syntax highlighting and various automatic formatting options
- Intelligent Code Completion
- Data manipulations (INSERT, UPDATE, DELETE) may be done from a spreadsheet-like interface. Both raw table data and a result set from a query can be manipulated.
- Visual Schema Designer
- Visual Query Builder
- Query Formatter
- Connectivity options: Direct client/server using MySQL API (SSL supported), HTTP/HTTPS tunnelling, SSH tunnelling
- Wizard-driven tool for import of data from ODBC-databases
- Backup Tool for performing unattended backups. Backups may be compressed and optionally stored as a file-per-table as well as identified with a timestamp.
- "SQL Scheduler and Reporting Tool" - a tool for scheduling and automating execution of any sequence of SQL statements. Result of queries may be sent as HTML-formatted reports.
- Schema/Structure Synchronization and Data Synchronization
- Query Profiler and Redundant Index Finder
- All automated jobs have mail alerting and reporting options.
- Full character set/Unicode support
- A "Data Search" feature using a Google-type search syntax translated transparently for user to SQL.
- Form view to display one row at a time
- Foreign key lookup
- Visual Data Compare

OpenCV

OpenCV (Open Source Computer Vision) is a free cross-platform computer vision library for real-time image processing. OpenCV was Initially developed by Intel. The OpenCV software has become a de-facto standard tool for all things related to Computer Vision. OpenCV is written in C and C++. It runs under the most popular operating systems, such as GNU/Linux, OS X, Windows, Android, iOS, etc. It is available for free under the Apache 2 license.

There is active development on interfaces for Python, Ruby, Matlab, and other languages making it easily accessible through commands like “pip install opencv” for Python users and “git opencv” for version control. The OpenCV library contains over 2500 algorithms, extensive documentation, source code, and sample code for real-time computer vision. Developers using the Python package, along with other Python libraries, can seamlessly integrate OpenCV into their projects using commands like “python opencv.” This integration is made even more convenient through package managers, which provide a simple process for installation and version control. Since its first release in 2000 under the BSD license and later under the Apache 2 license, the open-source library has been used in numerous cutting-edge applications, products, and research projects. These applications include stitching camera images together in satellite or web maps, image scan alignment, medical image noise reduction, object analysis, security, surveillance and intrusion detection systems, automatic monitoring, and safety systems, manufacturing AI inspection, camera calibration, defense and military applications, and unmanned aerial, ground, and underwater vehicles. OpenCV has even been used in sound and music recognition, where vision recognition techniques have been applied to sound spectrogram images.

OpenCV was built for maximum efficiency and performance of computing-intensive vision tasks. Therefore, it has a strong focus on real-time applications of AI vision. The open-source software is written in optimized C and can take advantage of multicore processors (multi-threading). The goal of OpenCV is to provide an easy-to-use computer vision infrastructure that helps people build sophisticated vision applications quickly by providing over 500 functions that span many areas in vision. OpenCV is often used in factory product inspection, medical imaging, security analysis, human-machine interface, camera calibration, stereo vision (3D vision), and robotic vision. The comprehensive image processing capabilities support video stream processing, image stitching (combining multiple cameras), camera calibration, and diverse image pre-processing tasks. Because machine learning is essential in computer vision, OpenCV contains a complete, general-purpose ML Library focused on statistical pattern recognition and clustering. To enhance performance, OpenCV has supported NVIDIA CUDA and GPU acceleration since 2011, providing functionality for handling installed packages, site packages, and OpenCV binaries, along with explicit control over data movement between CPU and GPU memory through the OpenCV GPU module.

OpenCV is used by big enterprises and government institutions, for example, Google, Toyota, IBM, Microsoft, SONY, Siemens, and Facebook. Also, well-known computer vision startups use OpenCV to build powerful computer vision products and AI solutions – including viso.ai. Many research centers use OpenCV, such as Stanford, MIT, INRIA, Cambridge, and CMU. The use cases of computer vision are vast. While most are aware of rather popular use cases in security and video surveillance or self-driving cars, fewer people get to see the use cases in specific industries such as industrial manufacturing, restaurants, or retail analytics. The rapid advances in computer vision over the past years have made it possible for companies across industries to develop purpose-built computer vision applications that solve highly specific problems (detect product issues, count objects, analyze behavior, etc.).

OpenCV was initially created as part of an Intel Research initiative to advance CPU-intensive applications. One of the authors working for Intel noticed that some top university groups, such as the MIT Media Lab, used well-developed and internally open computer vision infrastructures to accelerate the development work. The code was shared between students and provided a significant head start in developing custom vision applications without reinventing the basic functions from scratch. Accelerating commercial computer vision development by providing a common infrastructure to build on has always been the primary goal of the OpenCV developer teams at Intel. Advancing vision-based commercial applications by making portable, performance-optimized code available for free would increase the need for fast processors – the core business of Intel. Driving upgrades to faster processors would generate more income for Intel than selling some extra software. This is probably why the free and open code originated from a hardware vendor rather than a software company. The open-source library receives many user contributions, and central development has largely moved outside Intel. It was later supported by Willow Garage and the computer vision startup Itseez which Intel acquired in 2016. Downloading and Installing OpenCV Since 2012, support for OpenCV was taken over by a non-profit foundation named OpenCV.org, which maintains a developer site and a user website. Official releases, including the Python version, contrib modules, and built binaries, can be obtained from SourceForge. Alternatively, allows to access the latest sources from the OpenCV GitHub repository.

Recently, no-code and low-code development has become a new way for enterprises and organizations to deliver and maintain solutions much faster and more efficiently. Computer vision development is usually very complex and requires numerous iteration cycles. Hence, the delivery of computer vision greatly benefits from visual development and an automated deployment infrastructure of no-code technology. The computer vision platform Viso Suite provides the capabilities of OpenCV as modular building blocks that can be used to rapidly build computer vision applications without writing code from scratch. This allows teams to use OpenCV faster and facilitates integration with different hardware such as cameras, edge computers, and machine learning models. No-code helps to bridge the gap between seasoned computer vision engineers and business teams and makes it possible to adjust solutions to changing business requirements and advancing technology. The ability to use OpenCV without coding leverages the full economic potential of computer vision and lowers the risk and costs of computer vision. Hence, we have built Viso Suite to provide an end-to-end solution for teams to build, deploy, and operate all their computer vision applications in one place – using the best software for Computer Vision, including OpenCV.

Computer vision: Computer vision is a primary field of Artificial Intelligence technology that allows computers to extract information from digital images and videos to take specific actions. While there are different definitions of Artificial Intelligence (AI), all concepts generally focus on replicating human intelligence with machines.

The technologies of machine vision aim to imitate human vision; it involves the interpretation of digital images or videos for understanding and recognizing objects and scenes in them. This is achieved through a combination of software and hardware that mimics the workings of the human visual system. Some common methods used in machine vision are pattern recognition, feature extraction, and image processing.

AI vision, or Computer Vision, involves a set of image-processing techniques to make computers see and understand visual information. Computers act as “the brains” that perform advanced computing tasks and apply sophisticated algorithms to analyze images or videos provided by optical sensors or cameras (the “eyes”). Virtually any image of any camera can be used to apply AI vision algorithms.

The ability to make computers see with AI and perceive the physical world using visual sensors is becoming an integral technology to digitize and automate operations effectively. In recent years, machine learning technologies – especially deep learning, have shown great success in computer vision applications across industries. Most applications leverage Artificial Intelligence with IoT (AIoT), Cloud Computing, and Edge AI to deliver and deploy computer vision anywhere and at scale. Leveraging AI vision tech, computers can perform face recognition, read handwriting, recognize objects, classify human movement, perform automated inspection, or detect critical situations automatically with image recognition.

Develop Computer Vision with OpenCV Computer Vision is non-trivial, and achieving good performance at a reasonable cost is the basis of scalable computer vision. When working with cameras, the image data is frequently corrupted by noise and distortions stemming from variations in the physical world (lighting, reflections, movement, imperfections of the lens (distortion, field of view), sensor, or mechanical setup (angle, position, height). To overcome these daunting challenges, developers need to build sophisticated computer vision pipelines that model the path of the data flows. This application logic combines different tasks, from acquiring the frames to preprocessing them (denoising, filtering, dewarping, etc.) and fleeting them into one or multiple vision algorithms. OpenCV provides a standard toolset for developers to solve computer vision problems. In some cases, high-level functionalities in the library will be sufficient to solve the more complex problems in AI vision. However, writing conventional code can quickly become complex, and hard to understand and maintain or update as business requirements or regulations change. To accelerate the development, the Viso Suite platform uses no-code technology to leverage the capabilities of OpenCV. This allows developers to build their computer vision pipelines visually with modular building blocks. The no-code editor and automated AI model management facilitate collaboration and make building and maintaining computer vision pipelines much faster. Using OpenCV with no code benefits both vision experts and newly trained developers with basic knowledge. Vision Capabilities of OpenCV OpenCV is probably the most versatile computer vision tool used in a broad field of computer vision tasks ranging from image recognition and 2D or 3D analysis to motion tracking, facial recognition, and more.

The following are the most prominent capabilities of OpenCV:

- Real-time object detection: Object detection technology is used to apply image recognition and locate specific objects in video data or images, such as cars, humans, animals, and specific parts or equipment in industrial manufacturing.

- Image segmentation: Image segmentation applies image processing algorithms to divide an image into different segments. Segmentation is usually applied to simplify, change, or enhance the image, often combined with subsequent computer vision tasks. An example is autonomous driving, where image segmentation is used to determine the road.

- Movement and gesture recognition: Human pose and gesture recognition are used to interpret and understand the gestures of human beings through video analysis. Body, hand, or facial movements can be recognized and categorized to assign a pre-defined category. Movement analysis is often part of pose estimation to analyze the body movements with reference key points (joints, limbs). Calculating the object pose provides a method to understand how the object is situated in a 3D space, for example, how it is rotated.

- Face Recognition: Automatic face recognition is used to identify humans by detecting a human face and matching it with a database based on detected facial features. The FaceRecognizer of OpenCV provides a set of popular face recognition algorithms to use in real applications.

- Augmented Reality: Augmented reality (AR) allows real-time interaction between the real world and the virtual world. Therefore, augmented reality aims to augment the physical world around us with computer-generated perceptual information. Advanced Computer Vision OpenCV Techniques OpenCV also offers more sophisticated techniques extending beyond the basic functionalities.

- Key Feature Detection and Description SIFT (Scale-Invariant Feature Transform): SIFT is an OpenCV algorithm for detecting and describing key features in images. Known for being effective in object recognition and image stitching, SIFT’s scale-invariant nature makes it resilient to variations in object size and orientation.

- SURF (Speeded-Up Robust Features): SURF is another algorithm for keypoint detection and image feature description. Building on SIFT, SURF offers increased computational speed, useful for real-time applications. Object Detection Techniques OpenCV provides pre-trained models for object detection. These pre-built models simplify model implementation, which allows users to get started without the need for extensive manual configuration.

- Haar Cascades: Haar Cascades, implemented through cv2.CascadeClassifier() , are useful in detecting specific patterns like faces and eyes.

- Deep Learning-Based Methods: OpenCV provides integrations with popular deep learning models, such as Single Shot Multibox Detector (SSD) and You Only Look Once (YOLO).

- Image Segmentation and Motion Analysis Segmentation Techniques: Dive deeper into image segmentation using techniques like GrabCut and MeanShift. These split images based on color, texture, and intensity.

- Optical Flow: Lucas-Kanade and Farneback algorithms can track motion between consecutive frames in a video by calculating the displacement of pixels over time.

- Image Morphology and Enhancement Morphological Operations: Morphological operations like erosion, dilation, opening, and closing can be used for shape analysis and noise removal. Erosion shrinks objects and removes small details, while dilation expands objects and fills in gaps. Opening, a combination of erosion followed by dilation, helps in noise reduction and breaking narrow connections. Closing, done through dilation followed by erosion, helps in closing small holes and gaps.

- Image Filtering: Gaussian, Sobel, Laplacian, and others can be applied for image enhancement and feature extraction. Gaussian filters smooth images by reducing noise and blurring. Sobel filters highlight edges by computing the gradient. Laplacian filters detect regions of rapid intensity change, emphasizing fine details.

- Camera Calibration and Stereo Vision Camera Calibration: Camera calibration estimates intrinsic and extrinsic camera parameters to fix distortions and improve the accuracy of image measurements. The cv2.calibrateCamera() function enables the calibration by capturing images of a calibration pattern from different viewpoints. The parameters, such as focal length and distortion coefficients, allow for precise 3D reconstruction and accurate perspective correction. This is useful in applications like augmented reality.

- Stereo Vision Techniques: Stereo vision techniques, such as StereoBM and StereoSGBM, focus on depth estimation from images captured by a stereo camera setup. StereoBM, created with cv2.StereoBM_create() , uses block matching to determine disparities and then calculate depth information. StereoSGBM, implemented with cv2.StereoSGBM_create() , uses semi-global matching for more robust results. Background Subtraction and Image Stitching Background Subtraction: Background subtraction algorithms, like MOG2 and KNN, isolate moving objects in video streams. These methods use a contrast-based approach, differentiating foreground and background elements. Evaluation metrics often include precision, recall, and F1 score to assess the accuracy of object detection and delineation.

- Image Stitching: Image stitching involves combining multiple images to create a panoramic view. OpenCV provides functions for feature matching, homography estimation, and perspective transformation to align and blend images. Evaluation can include visual inspection for seam quality and geometric accuracy assessment, ensuring that the results are cohesive and distortion-free.

Applications built with OpenCV Most people are aware of computer vision being widely popular in security and surveillance. Some people are aware of niche applications in safety monitoring, unmanned flying vehicles (drones), or biomedical analysis. But few are aware of how pervasive machine vision has become in industrial manufacturing virtually everything that is mass-produced has been automatically inspected at some point using computer vision.

Use cases built with OpenCV Since OpenCV is a development kit, there are countless use cases can be build with OpenCV, including: Recognition of objects for counting and object tracking Analyzing medical images to support human diagnosis (Medical Imaging) Recognition of advertisements in TV footage or logo recognition with AI vision Player tracking in sports and fitness, scene recognition, and execution quality analysis Counting the number of people in public places such as airports (Crowd analysis) Robotic automation for intelligent, vision-based interfaces (Intelligent Screens) Automatic inspection and video analysis with always-on computer vision (for example, at airports in aviation). Image search on digital platforms, in web-based applications Detecting defects or faults during manufacturing processes (Smart Factory) Counting the number of vehicles on a highway (Traffic Analysis) CCTV camera applications to detect physical violence, attacks, and traffic violations (Smart City) Explore more in our complete list of computer vision use cases. Computer Vision by Industry In many industries, computer vision provides a competitive advantage, allowing the transformation of laborious and expensive processes with highly tailored, integrated, and high-performing computer vision applications.

MediaPipe

MediaPipe is a comprehensive framework designed to simplify the development of complex computer vision applications. It offers a wide range of pre-trained models and customizable components, allowing developers to integrate sophisticated perception capabilities into their projects with relative ease. Developed by the Google Research Brain Team, MediaPipe is built on TensorFlow, making it accessible to a broad audience of researchers, engineers, and developers.

MediaPipe stands out as a powerful and versatile framework that has gained significant attention and adoption. This framework provides a robust set of tools and solutions for building applications that involve perceptual computing tasks, such as object detection, facial recognition, hand tracking, and pose estimation.

MediaPipe is an open-source framework for building pipelines to perform computer vision inference over arbitrary sensory data such as video or audio. In computer vision pipelines, those components include model inference, media processing algorithms, data transformations, etc. Sensory data such as video streams enter the graph, and perceived descriptions such as object-localization or face-keypoint streams exit the graph. Using MediaPipe, such a perception pipeline can be built as a graph of modular components. MediaPipe is currently in alpha at v0.7, and there may still be breaking API changes. Stable APIs are expected by v1.0.

The MediaPipe framework is mainly used for rapid prototyping of perception pipelines with AI models for inferencing and other reusable components. It also facilitates the deployment of computer vision applications into demos and applications on different hardware platforms. The configuration language and evaluation tools enable teams to incrementally improve computer vision pipelines.

Most MediaPipe solutions use different TFLite models (TensorFlow Lite) for hand, iris, or face landmarks, palm detection, pose detection, Template-based Feature Matching (KNIFT), or 3D Object Detection (Objectron) tasks.

The advantages of MediaPipe are:

- End-to-end acceleration: Use common hardware to build-in fast ML inference and video processing, including GPU, CPU, or TPU.
- Build once, deploy anywhere: The unified framework is suitable for Android, iOS, desktop, edge, cloud, web, and IoT platforms.
- Ready-to-use solutions: Prebuilt ML solutions demonstrate the full power of the MediaPipe framework.
- Open source and free: The framework is licensed under Apache 2.0, fully extensible, and customizable.

Currently, MediaPipe supports Ubuntu Linux, Debian Linux, macOS, iOS, and Android. The MediaPipe framework is based on a C++ library (C++ 11), making it relatively easy to port to additional platforms.

Typically, image or video input data is fetched as separate streams and analyzed using neural networks such as TensorFlow, PyTorch, CNTK, or MXNet. Such models process data in a simple and deterministic method: one input generates one output, which allows very efficient processing execution. MediaPipe, on the other hand, operates at much higher-level semantics and allows more complex and dynamic behavior. For example, one input can generate zero, one, or multiple outputs, which cannot be modeled with neural networks. Video processing and AI perception require streaming processing compared to batching methods. OpenCV 4.0 introduced the Graph API to build sequences of OpenCV image processing operations in the form of a graph. In contrast, MediaPipe allows operations on arbitrary data types and has native support for streaming time-series data, making it much more suitable for analyzing audio and sensor data.

The MeidaPipe Framework consists of three main elements: A framework for inference from sensory data (audio or video) A set of tools for performance evaluation, and Re-usable components for inference and processing (calculators).

The main components of MediaPipe:

- Packet: The basic data flow unit is called a “packet.” It consists of a numeric timestamp and a shared pointer to an immutable payload.
- Graph: Processing takes place inside a graph which defines the flow paths of packets between nodes. A graph can have any number of input and outputs, and branch or merge data.
- Nodes: Nodes are where the bulk of the graph’s work takes place. They are also called “calculators” (for historical reasons) and produce or consume packets. Each node’s interface defines a number of in- and output ports.
- Streams: A stream is a connection between two nodes that carries a sequence of packets with increasing timestamps.

MediaPipe allows a developer to prototype a pipeline incrementally. A vision pipeline is defined as a directed graph of components, where each component is a node (“Calculator”). In the graph, the calculators are connected by data “Streams.” Each stream represents a time-series of data “Packets.” Together, the calculators and streams define a data-flow graph. The packets which flow across the graph are collated by their timestamps within the time-series. Each input stream maintains its own queue to allow the receiving node to consume the packets at its own pace. The pipeline can be refined incrementally by inserting or replacing calculators anywhere in the graph. Developers can also define custom calculators. While the graph executes calculators in parallel, each calculator executes on at most one thread at a time. This constraint ensures that the custom calculators can be defined without specialized expertise in multithreaded programming.

MediaPipe supports GPU computing and rendering nodes and allows to combine multiple GPU nodes and mix them with CPU-based nodes. There are several GPU APIs on mobile platforms (OpenGL ES, Metal, Vulkan). There is no single cross-API GPU abstraction. Individual nodes can be written using different APIs, allowing them to take advantage of platform-specific features when needed. This enables GPU and CPU nodes to provide advantages of encapsulation and composability while maintaining efficiency.

he MediaPipe tracer module records timing events across the graph to record events with several data fields (time, packet timestamp, data Id, node Id, stream ID). The tracer also reports histograms of various resources, such as the elapsed CPU time across each calculator and stream. The tracer module records timing information on demand and can be enabled using a configuration setting (in GraphConfig). The user can also completely omit the tracer module code using a compiler flag. The recorded timing data enables reporting and visualization of individual packet flows and individual calculator executions. The recorded timing data helps diagnose several problems, such as unexpected real-time delays, memory accumulation due to packet buffering, and collating packets at different frame rates. The aggregated timing data can be used to report average and extreme latencies for performance tuning. Also, the timing data can be explored to identify the nodes along the critical path, whose performance determines end-to-end latency.

The MediaPipe visualizer is a tool for understanding the topology and overall behavior of the pipelines. It provides a timeline view and a graph view. In the timeline view, the user can load a pre-recorded trace file and see the precise timings of data as it moves through threads and calculators (nodes). In the graph view, the user can visualize the topology of a graph at any point in time, including the state of each calculator and the packets being processed or being held in its input queues.

Google MediaPipe provides a framework for cross-platform, customizable machine learning solutions for live and streaming media to deliver live ML anywhere. The powerful tool is suitable for creating computer vision pipelines and complex applications. MediaPipe is fully integrated into the Viso Suite platform that provides an end-to-end solution for AI vision applications. With Viso Suite, businesses can take full advantage of MediaPipe while covering the entire lifecycle of computer vision in one unified solution, with no-code and automated infrastructure.

Handtracking Using Mediapipe: MediaPipe’s Hand Tracking model employs complex algorithms and deep learning techniques to analyse video frames and identify the position and movement of hands within those frames. It can detect multiple hands simultaneously, providing highly accurate coordiMediaPipe is an open-source framework for building pipelines to perform computer vision inference over arbitrary sensory data such as video or audio.nates for each detected hand.

Pygame

Pygame is a free and open-source cross-platform library for the development of multimedia applications like video games using Python. It uses the Simple DirectMedia Layer library and several other popular libraries to abstract the most common functions, making writing these programs a more intuitive task. This library includes several modules for playing sound, drawing graphics, handling mouse inputs, and sound libraries etc., designed to be used with the Python programming language. It is also used to create client-side applications that can be wrapped in standalone executables.

Pygame, developed by Pete Shinners, has been a staple for game development since its release on October 28, 2000. Known for its versatility, it supports programming languages such as Python, C, Cython, and Assembly language, enabling developers to create a wide range of games and multimedia applications. Under the GNU Lesser General Public License, Pygame has seen consistent updates, with its stable release 2.5.0 launched on June 24, 2023, further enhancing its capabilities and ensuring its relevance in the ever-evolving landscape of game development.

Pygame uses the Simple DirectMedia Layer (SDL) library, with the intention of allowing real-time computer game development without the low-level mechanics of the C programming language and its derivatives. This is based on the assumption that the most expensive functions inside games can be abstracted from the game logic, making it possible to use a high-level programming language, such as Python, to structure the game. Other features that SDL does have include vector math, collision detection, 2D sprite scene graph management, MIDI support, camera, pixel-array manipulation, transformations, filtering, advanced freetype font support, and drawing. Applications using Pygame can run on Android phones and tablets with the use of Pygame Subset for Android (pgs4a). Sound, vibration, keyboard, and accelerometer are supported on Android.

Pygame is a popular Python module designed for game application development, offering extensive libraries and features for creating games with unique functionalities. It is portable and can run on almost all operating systems, making it accessible for developers. It also has a large user base and community support, with millions of downloads and visits to the official website. Additionally, Pygame is released under the GNU General Public License, allowing developers to create games that are free, open source, shareware, freeware, and commercial. This tool handles many low-level details, which really helpful for developers.

Pygame has several advantages:

- Intuitive Syntax and Structure
- Abundance of Helpful Resources and Documentation
- Beginner-Friendly Community Support
- Works Seamlessly on Multiple Operating Systems
- Ideal for Reaching a Wider Audience
- Reduced Development Time and Effort

PyGame is built upon the Simple DirectMedia Layer (SDL) library, which facilitates the interface and management of input devices like a keyboard, mouse, or joystick, as well as output devices like a screen or speaker. Furthermore, SDL also offers features like vector mathematics, collision detection, 2D sprite management, pixel matrix manipulation, transformations, filtering, and drawing. It also supports MIDI and freetype fonts.Applications based on PyGame can be run on Android smartphones and tablets using a Pygame Subset for Android. This includes support for sound, vibrations, keyboard input, and accelerometer functionality.

TensorFlow

TensorFlow is a very popular end-to-end open-source platform for machine learning. It was originally developed by researchers and engineers working on the Google Brain team before it was open-sourced. The TensorFlow software library replaced Google's DistBelief framework and runs on almost all available execution platforms (CPU, GPU, TPU, Mobile, etc.). The framework provides a math library that includes basic arithmetic operators and trigonometric functions. TensorFlow is currently used by various international companies, such as Google, Uber, Microsoft, and a wide range of universities. Keras is the high-level API of the TensorFlow platform. It provides an approachable, efficient interface for solving machine learning (ML) problems, with a focus on modern deep learning models. The TensorFlow Lite implementation is specially designed for edge-based machine learning. TF Lite is optimized to run various lightweight algorithms on various resource-constrained edge devices, such as smartphones, microcontrollers, and other chips. TensorFlow Serving offers a high-performance and flexible system for deploying machine learning models in production settings. One of the easiest ways to get started with TensorFlow Serving is with Docker. For enterprise applications using TensorFlow, check out the computer vision platform Viso Suite which automates the end-to-end infrastructure around serving a TensorFlow model at scale.

Advantages of TensorFlow are:

- Support and library management: TensorFlow is backed by Google and has frequent releases with new features. It is popularly used in production environments.

- Open-sourced: TensorFlow is an open-source platform that is very popular and available to a broad range of users.
- Data visualization: TensorFlow provides a tool called TensorBoard to visualize data graphically. It also allows easy debugging of nodes, reduces the effort of looking at the whole code, and effectively resolves the neural network.
- Keras compatibility: TensorFlow is compatible with Keras, which allows its users to code some high-level functionality sections and provides system-specific functionality to TensorFlow (pipelining, estimators, etc.).
- Very scalable: TensorFlow's characteristic of being deployed on every machine allows its users to develop any kind of system.
- Compatibility: TensorFlow is compatible with many languages, such as C++, JavaScript, Python, C#, Ruby, and Swift. This allows a user to work in an environment they are comfortable in.
- Architectural support: TensorFlow finds its use as a hardware acceleration library due to the parallelism of work models. It uses different distribution strategies in GPU and CPU systems. TensorFlow also has its architecture TPU, which performs computations faster than GPU and CPU. Therefore, models built using TPU can be easily deployed on a cloud at a cheaper rate and executed at a faster rate. However, TensorFlow's architecture TPU only allows the execution of a model, not training it.

TensorFlow is among the most popular end-to-end open-source machine learning platforms with a comprehensive set of tools, resources, and libraries. TensorFlow is especially useful for building and deploying applications related to computer vision that are powered by machine learning. TensorFlow is one of the easiest computer vision tools and allows users to develop computer vision-related machine learning models for tasks like facial recognition, image classification, object detection, and more. Tensorflow, like OpenCV, also supports various languages like Python, C, C++, Java, or JavaScript. For real-world computer vision projects, the TensorFlow Lite is a lightweight implementation for on-device machine learning with edge devices. As part of TensorFlow, TF Lite greatly accelerates edge ML implementations with reduced model size and high accuracy at much higher efficiency, making it possible to run ML everywhere.

Different features of TensorFlow are:

- Easy model building: TensorFlow offers multiple levels of abstraction so it allows to choose the right one for the needs. Build and train models by using the high-level Keras API, which makes getting started with TensorFlow and machine learning easy. If more flexibility is desired, eager execution facilitates immediate iteration and intuitive debugging. For large ML training tasks, use the Distribution Strategy API for distributed training on different hardware configurations without changing the model definition.
- Robust ML production anywhere: TensorFlow has always provided a direct path to production. Whether it's on servers, edge devices, or the web TensorFlow facilitates easy training and deployment of models across various languages and platforms. TFX is recommended for comprehensive production ML pipelines. For running inference on mobile and edge devices, use TensorFlow Lite. Train and deploy models in JavaScript environments using TensorFlow.js.
- Powerful experimentation for research: Build and train state-of-the-art models without sacrificing speed or performance.
- TensorFlow gives the flexibility and control with features like the Keras Functional API and Model Subclassing API for creation of complex topologies. For easy prototyping and fast debugging, use eager execution.
- TensorFlow also supports an ecosystem of powerful add-on libraries and models to experiment with, including Ragged Tensors, TensorFlow Probability, Tensor2Tensor and BERT.

TensorFlow is heavily used by data scientists, software developers, and educators, as an open-source platform for machine learning using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) that flow between them. This flexible architecture allows machine learning algorithms to be described as a graph of connected operations. They can be trained and executed on GPUs, CPUs, and TPUs across various platforms without rewriting code, ranging from portable devices to desktops to high-end servers. This means programmers of all backgrounds can use the same toolsets to collaborate, significantly boosting their efficiency. Developed initially by the Google Brain Team for the purposes of conducting machine learning and deep neural networks (DNNs) research, the system is general enough to be applicable in a wide variety of other domains as well.

There are three distinct parts that define the TensorFlow workflow, namely preprocessing of data, building the model, and training the model to make predictions. The framework inputs data as a multidimensional array called tensors and executes in two different fashions. The primary method is by building a computational graph that defines a dataflow for training the model. The second, and often more intuitive method, is using eager execution, which follows imperative programming principles and evaluates operations immediately.

Using the TensorFlow architecture, training is generally done on a desktop or in a data center. In both cases, the process is sped up by placing tensors on the GPU. Trained models can then run on a range of platforms, from desktop to mobile and all the way to cloud.

TensorFlow also contains many supporting features. For example, TensorBoard, which allows users to visually monitor the training process, underlying computational graphs, and metrics for purposes of debugging runs and evaluating model performance. Tensor board is the unified visualization framework for Tensorflow and Keras.

Keras is a high-level API that runs on top of TensorFlow. Keras furthers the abstractions of TensorFlow by providing a simplified API intended for building models for common use cases. The driving idea behind the API is being able to translate from idea to a result in as little time as possible.

TensorFlow can be used to develop models for various tasks, including natural language processing, image recognition, handwriting recognition, and different computational-based simulations such as partial differential equations.The key benefits of TensorFlow are in its ability to execute low-level operations across many acceleration platforms, automatic computation of gradients, production-level scalability, and interoperable graph exportation. By providing Keras as a high-level API and eager execution as an alternative to the dataflow paradigm on TensorFlow, it’s always easy to write code comfortably.

TensorFlow has demonstrated its versatility and effectiveness across various industries and applications, showcasing its potential for solving complex business challenges. For instance, Airbus leverages TensorFlow for image processing and video detection, allowing them to extract and analyze information from satellite images, providing valuable real-time insights to clients. Additionally, Kakao utilizes TensorFlow's time series algorithms to predict the completion rate of ride-hailing requests, enhancing operational efficiency. NERSC, a scientific research center, achieved remarkable scale capabilities by leveraging TensorFlow, scaling a deep learning application to over 27,000 NVIDIA V100 Tensor Core GPUs, pushing the boundaries of computational power. PayPal effectively combats fraud by employing TensorFlow for deep transfer learning and generative modeling, enabling the recognition of intricate fraud patterns while enhancing the experience of legitimate customers through expedited identification processes. Moreover, SwissCom leverages a custom-built TensorFlow model for text recognition, enabling efficient classification and intent determination of customer inquiries, enhancing overall business operations. Finally, Twitter utilizes TensorFlow to develop its Ranked Timeline feature, ensuring users stay updated on their most important tweets amidst a vast network of users they follow, showcasing TensorFlow's ability to optimize user experience and engagement. These diverse applications highlight TensorFlow's adaptability and effectiveness in addressing critical business needs across various domains.

Keyboard

The keyboard module is a Python package or module that comes with many built-in functions that are helpful for us to get full control over the keyboard. We can use the functions of the keyboard module in Python programs and get full control over the keyboard of the device on which we are working. Unlike many other big libraries, the keyboard module is a small library of Python which is designed to perform only a specific set of actions. The keyboard module of Python can be used to simulate key presses, register hotkeys, hook global events, and perform many other actions. We can perform all these actions through the functions of the keyboard module by importing the keyboard module in the example Python programs.

The Python library keyboard offers comprehensive keyboard control functionalities, enabling users to manage keyboard inputs effectively. With this library, users can access a wide range of features including capturing keystrokes, recording keyboard activities, and blocking keys until a specific input is received. Additionally, it facilitates simulating key presses and supports complex hotkeys, enhancing its versatility. Notably, the library captures all keys, including events from on-screen keyboards, ensuring thorough monitoring. Compatible with both Windows and Linux operating systems, the keyboard module enables users to listen to and send keyboard events seamlessly, making it a valuable tool for various applications.

Keyboard has different features such as,

- Global event hook on all keyboards (captures keys regardless of focus).
- Listen and send keyboard events.
- Works with Windows and Linux (requires sudo), with experimental OS X support.
- Pure Python, no C modules to be compiled.
- Zero dependencies. Trivial to install and deploy, just copy the files.
- Complex hotkey support with controllable timeout.
- Includes high level API (e.g. record and play, add_abbreviation).
- Maps keys as they actually are in layout, with full internationalization support.
- Events automatically captured in separate thread, doesn't block main program.
- Tested and documented.
- Doesn't break accented dead keys (I'm looking at you, pyHook).
- Mouse support available via project mouse.

The keyboard module offers a suite of functionalities aimed at simplifying keyboard manipulation and automation. Among its features are the ability to simulate key presses, facilitating automated input in scripts and applications. Developers can utilize the module to record keyboard activities, capturing and logging user input or monitoring keyboard usage. Furthermore, users can block specific keys until a designated key is entered, granting control over keyboard input in diverse scenarios. Additionally, the module supports the creation of complex hotkeys, empowering developers to define custom key combinations for executing specific actions or commands. One of the keyboard module’s key advantages is its cross-platform compatibility. It works seamlessly on Windows and Linux operating systems, ensuring developers can leverage its capabilities across different environments without any compatibility issues.

Keyboard has several advantages:

- keyboard module is small and easy to use.
- It can simulate key presses, register hotkeys, and hook global events.
- You can use it to enter keys and record keyboard activities.
- You can even block any key of the keyboard.

- It captures all keys, including onscreen keyboard events.
- Operating systems: Linux and Windows are supported.

In Python, the keyboard module interacts with keyboard shortcuts and keys. It is easy to use. It is used in game development, automation tasks, and other applications. It can be easily simulates key presses, register hotkeys, hook global events, and perform various actions. So that it allows to make games, automate tedious chores, and design unique keyboard shortcuts.

SYSTEM TESTING AND IMPLEMENTATION

8. SYSTEM TESTING AND IMPLEMENTATION

Testing is the vital to the success of the system. It makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved in this project. It is the stage of implementation, which ensures that system works accurately and effectively before the live operation commences. It is a confirmation that all are correct and opportunity to show users that the system must be tested and show that the system will operate successfully and produce expected results under expected conditions. Software testing is a crucial element of software quality assurance and represents the unlimited review of specification, design and coding. Testing represents an interesting anomaly for the software. During the earlier definition and development phase, it was attempted to build the software from an abstract concept to implement. Testing is a set of activity that can be planned in advance and conducted. Systematically, this is aimed at ensuring that the system works accurately and efficiently before live operations commences.

8.1 Types of Testing

Different types of testing are:

- Unit testing
- Integration testing
- Validation testing
- System testing
- User acceptance testing

Unit Testing

Unit testing is a software testing technique that focuses on testing individual units or components of a software system in isolation. The purpose of unit testing is to ensure that each unit functions correctly and produces the expected outputs when provided with specific inputs. A unit can be a function, method, class, module, or any other discrete part of the software. Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases. All modules were tested individually as soon as they were completed and were checked for their correct functionality. Unit testing deals with testing a unit as a whole. This would test the interaction of many functions but confine the test within one unit. This testing is carried out during the programming stage itself. In this testing step each Module is found to be working satisfactorily as regard to the expected output from the module.

The benefits of unit testing include:

- Early detection of defects: Unit testing helps catch bugs and issues early in the development cycle, reducing the cost and effort required to fix them.
- Improved code quality: Unit tests serve as documentation and ensure that units are designed with testability and modularity in mind, resulting in cleaner and more maintainable code.
- Regression prevention: By having a comprehensive suite of unit tests, developers can easily rerun the tests to ensure that any changes or modifications to the codebase do not introduce new bugs or regressions.

However, unit testing also has some limitations. It does not guarantee the absence of defects in the integrated system, as issues can arise when units interact with each other. Integration testing is necessary to validate the proper integration and interaction between units. Additionally, achieving complete code coverage can be challenging, especially in complex systems with conditional logic and exceptional cases.

Integration Testing

Integration testing is a systematic technique for testing to overcome the errors associated within the interface. In this System all the modules such as login, registration, admin in web, user in web, and the final hardware and software are combined and then the entire program is tested as a whole. Thus, in the integration testing step all the errors in the implementation of the system are corrected. Data can be lost across an interface; one module can have an adverse effect on others; sub-functions when combined may not produce the desired major functions; integration testing is a systematic testing for constructing the program structure. The objective is to take unit tested modules and to combine them and test it as a whole.

- The system contains various components and they have to be combined and tested.
- The software is embedded into hardware and tested.
- The modules are combined and tested.
- The hardware part tested separately.

Validation Testing

Validation testing is the process of assessing a new software product to ensure that its performance matches consumer needs. Product development teams might perform validation testing to learn about the integrity of the product itself and its performance in different environments.

Developers can perform validation testing themselves, or collaborate with quality assurance professionals, external validation testing professionals or clients to identify elements of the code to improve. Developers can also combine this type of testing with other useful techniques like product verification, debugging and certification to help ensure the product is ready for the market.

Validation differs from verification testing, another important phase of the product development process. Verification testing is the process of confirming that the way a product performs meets the predetermined product specifications. Developers can perform this kind of testing throughout the development process. After verifying that the final product meets the design specifications, the team can move on to the validation process to ensure those specifications can meet user needs.

System Testing

After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it doesn't produce the required data in the specific format. The output displayed or generated by the system under consideration is tested by, asking the user about the format displayed. The output format on the screen is found to be correct as the format was designed in the system phase according to the user needs. Hence the output testing doesn't result in any correction in the system.

User acceptance testing

User acceptance testing (UAT), also called application testing or end-user testing, is a phase of software development in which the software is tested in the real world by its intended audience. User acceptance testing is often the last phase of the software testing process and is performed before the tested software is released to its intended market. The goal of user acceptance testing is to ensure software can handle real-world tasks and perform up to development specifications.

In user acceptance testing, users are given the opportunity to interact with the software before its official release to see if any features have been overlooked or if it contains any bugs. User acceptance testing can be done in-house with volunteers, by paid test subjects using the software or by making the test version available for download as a free trial. The results from the early testers are forwarded to the developers, who make final changes before releasing the software commercially.

8.2 Implementation

Implementation is the stage of project, when theoretical design is turned in to a working system. The most crucial stage is achieving a successful system and confidence that the new system will be work effectively. It involves careful planning, investigation of the manual system and to new system. Implementation means converting a new or revised system design into an operational one. The implementation includes all those activities that take place to convert from the old system to the new one.

There are several activities involved while implementing a project:

- Careful planning.
- Investigating the current system and its constraints on implementation.
- Design of methods to achieve the changeover.

- Training of the staff in the changeover procedure and evaluation of change over method.

Implementation is the final stage and it is an important phase. The first task in implementation was the implementation planning, that is deciding on methods to be adopted. After the system was implemented successfully, training of the user was one of the most important sub tasks of the developer. For this purpose, the user or system manual were prepared and handled over to the user to operate the developed system. So, change over plays a vital role, which checks the developed tool for the following requirements, and then only the user accepted the developed tool. The changeover took place only when the system had been proved to the satisfaction of the system analysis and other implementation activities have been completed.

SYSTEM MAINTENANCE

9. SYSTEM MAINTENANCE

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited.

There are 3 types of maintenance:

- Corrective (Fixing bugs/errors)
- Adaptive (Updates due to environment changes)
- Perfective (Enhancements, requirements changes)

Maintenance is enigma of the system development. The definition of the software maintenance can be given describing four activities that are undertaken after the program is released for use. The maintenance activity occurs since it is unreasonable to assume that software testing will uncover all in a large system. The second activity that contributes the definition of maintenance occurs since rapid changes are encountered in every aspects of computing. The third activity involves recommendation for new capabilities, modification to the existing functions and general enhancements when the software is used. The fourth maintenance activity occurs when software is changed to improve future maintainability or reliability.

FUTURE ENHANCEMENT

10. FUTURE ENHANCEMENT

To enhance the accessibility and versatility of the Air Scribe Decipherer system, future improvements include developing a mobile application version for smartphones and tablets, ensuring portability and on-the-go accessibility. Additionally, expanding gesture recognition capabilities to make it easier for people with disabilities to use by improving how it understands their movements. Furthermore, by providing the ability for it to understand different languages so more people can use it around the world. Moreover, advancing gesture recognition to recognize complete words or sentences would streamline communication, increasing efficiency in conveying complex messages. These enhancements collectively aim to make the system more user-friendly, versatile, and inclusive across various settings.

CONCLUSION

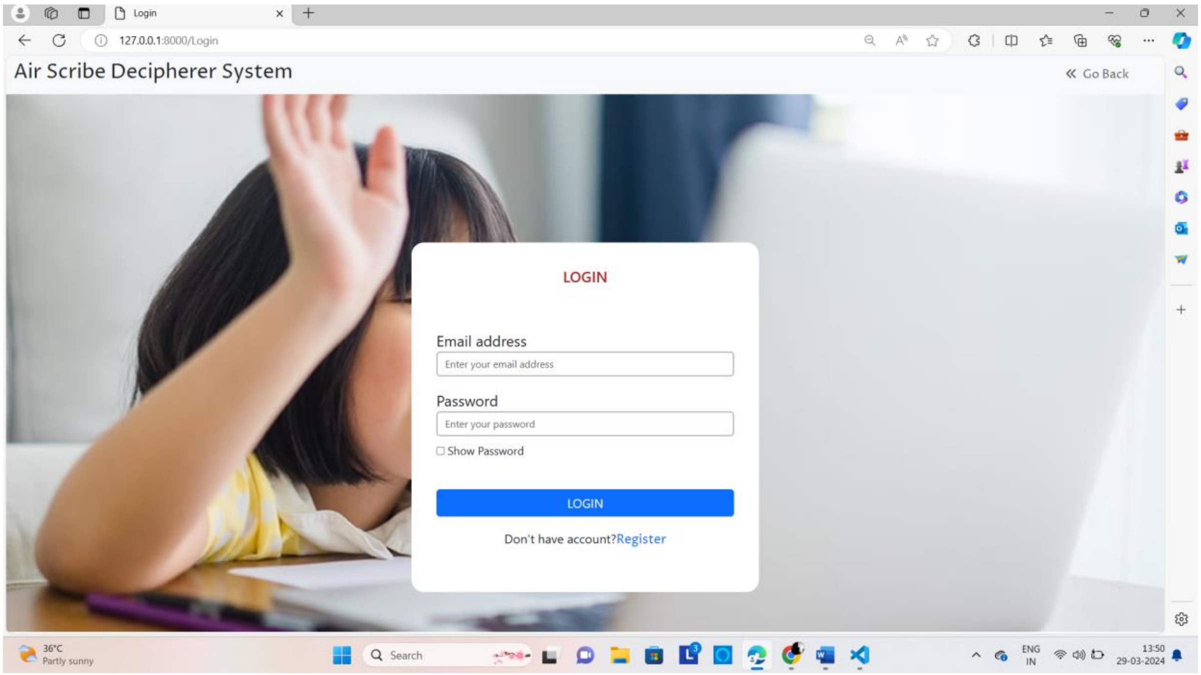
11. CONCLUSION

The Air Scribe Decipherer System utilizes gesture recognition to enable air writing through finger movements, a metaphorical emulation of traditional pen-based writing. Using OpenCV, it accurately translates these movements into text format, offering an innovative input method for computers and devices. This technology holds promise for enhancing inclusivity and accessibility, particularly for individuals with communication challenges, as it provides an alternative means of expression through gesture-based communication. Moreover, in educational contexts, it facilitates interactive learning experiences by allowing users to annotate presentations, illustrate concepts, and interact with content dynamically, supplemented by audio for recognized characters.

Additionally, the Air Scribe Decipherer System goes beyond text input by adding an air canvas feature. This allows users to draw freely in space using finger movements, turning the area around them into a digital canvas. By understanding these gestures accurately, the system lets users make digital drawings without needing physical tools. This not just helps people be creative but also allows them to interact in creative ways, making the system useful for many things other than just typing or writing.

APPENDIX

LOGIN



Email address

Enter your email address

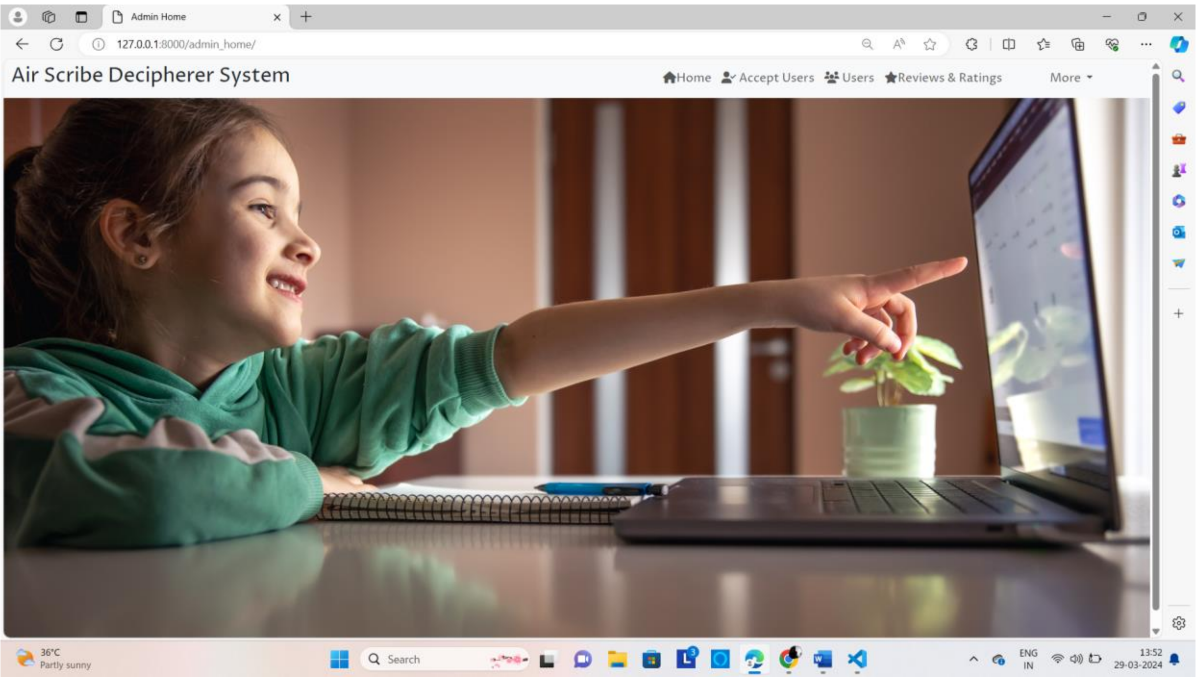
Password

Enter your password

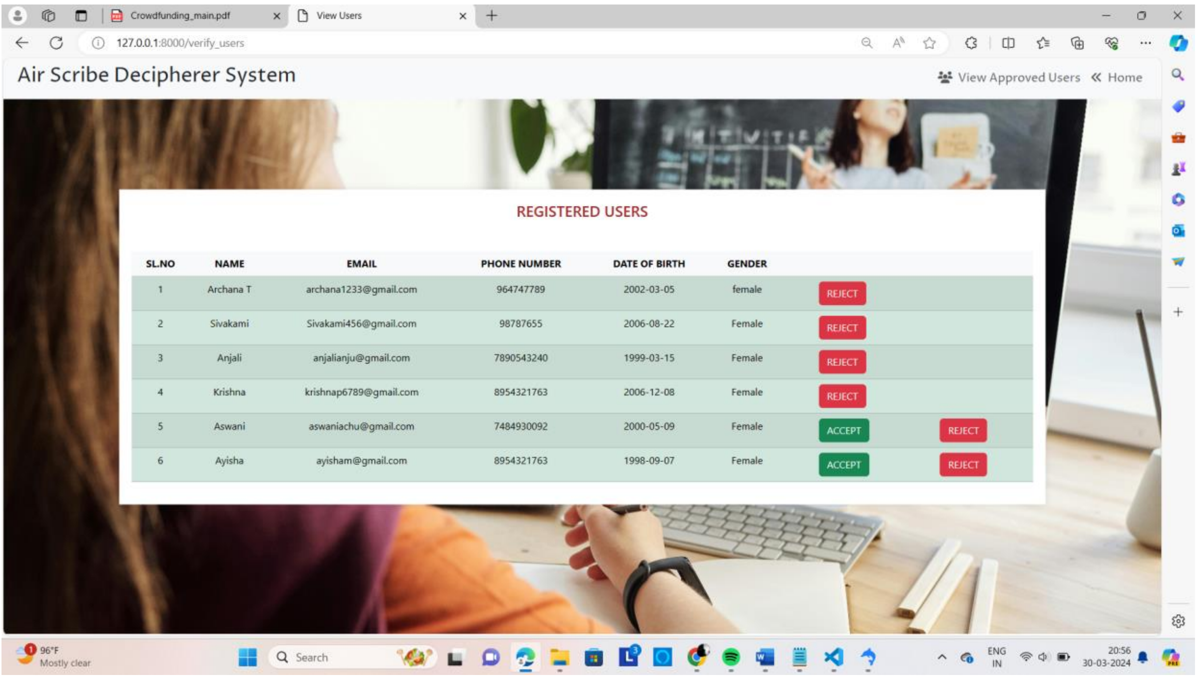
□ Show Password

Don't have account?Register

ADMIN HOME



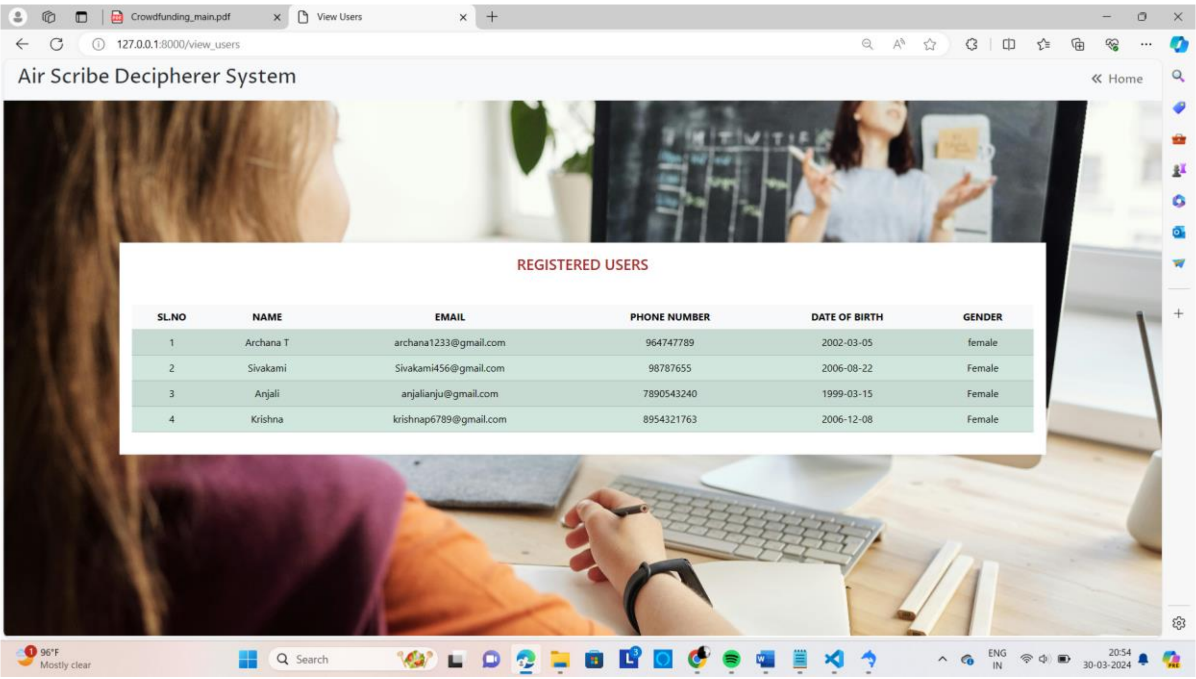
ACCEPT USERS



REGISTERED USERS

SL.NO	NAME	EMAIL	PHONE NUMBER	DATE OF BIRTH	GENDER		
1	Archana T	archana1233@gmail.com	964747789	2002-03-05	female	REJECT	
2	Sivakami	Sivakami456@gmail.com	98787655	2006-08-22	Female	REJECT	
3	Anjali	anjalianju@gmail.com	7890543240	1999-03-15	Female	REJECT	
4	Krishna	krishnap6789@gmail.com	8954321763	2006-12-08	Female	REJECT	
5	Aswani	aswaniachu@gmail.com	7484930092	2000-05-09	Female	ACCEPT	REJECT
6	Ayisha	ayisham@gmail.com	8954321763	1998-09-07	Female	ACCEPT	REJECT

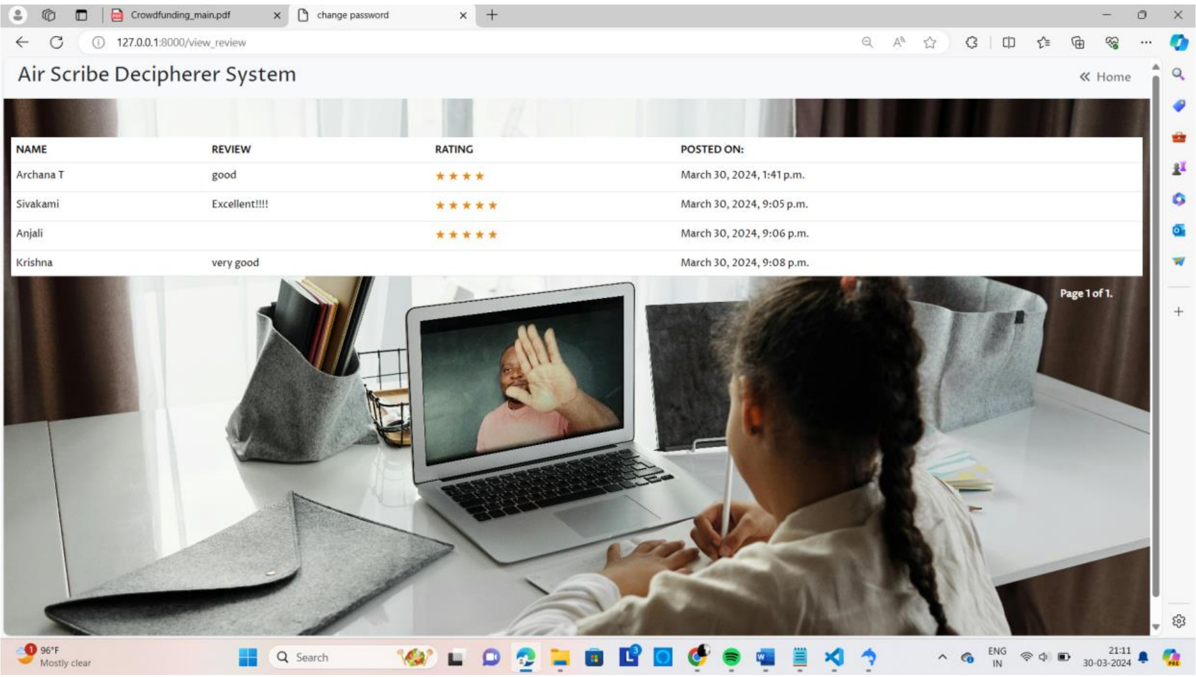
VIEW USERS



REGISTERED USERS

SL.NO	NAME	EMAIL	PHONE NUMBER	DATE OF BIRTH	GENDER
1	Archana T	archana1233@gmail.com	964747789	2002-03-05	female
2	Sivakami	Sivakami456@gmail.com	98787655	2006-08-22	Female
3	Anjali	anjalianju@gmail.com	7890543240	1999-03-15	Female
4	Krishna	krishnap6789@gmail.com	8954321763	2006-12-08	Female

VIEW REVIEWS & RATINGS

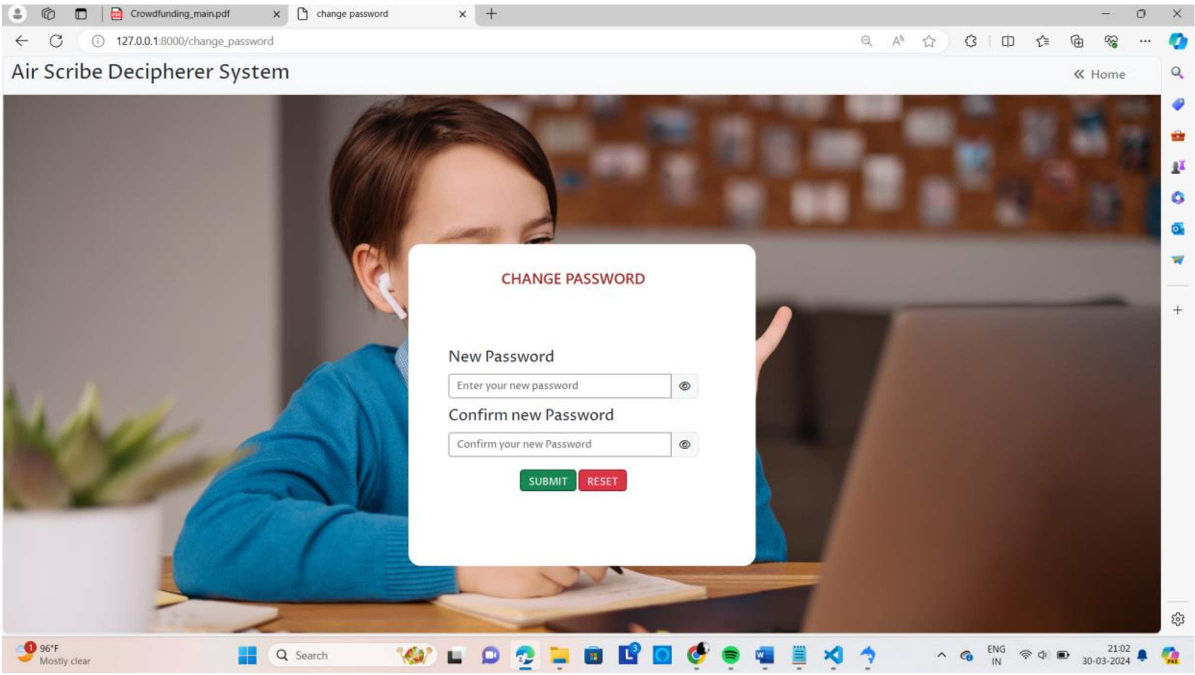


Air Scribe Decipherer System

NAME	REVIEW	RATING	POSTED ON:
Archana T	good	★★★★	March 30, 2024, 1:41 p.m.
Sivakami	Excellent!!!!	★★★★★	March 30, 2024, 9:05 p.m.
Anjali		★★★★★	March 30, 2024, 9:06 p.m.
Krishna	very good		March 30, 2024, 9:08 p.m.

Page 1 of 1.

CHANGE PASSWORD



Air Scribe Decipherer System

CHANGE PASSWORD

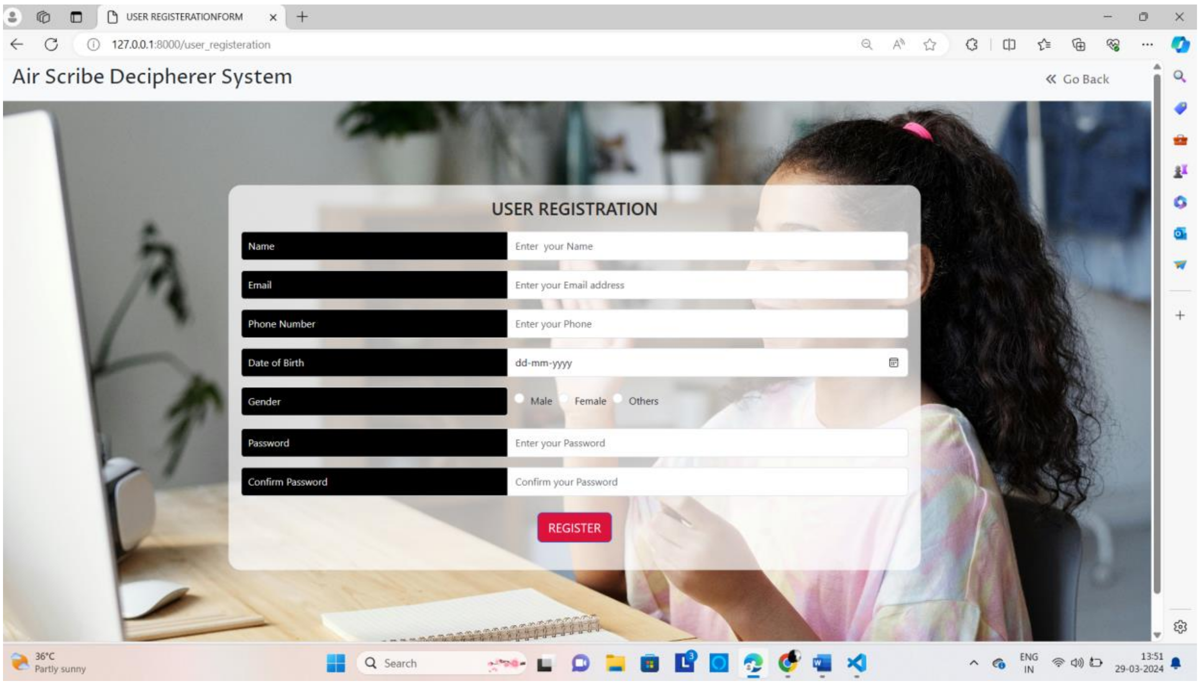
New Password

Enter your new password

Confirm new Password

Confirm your new Password

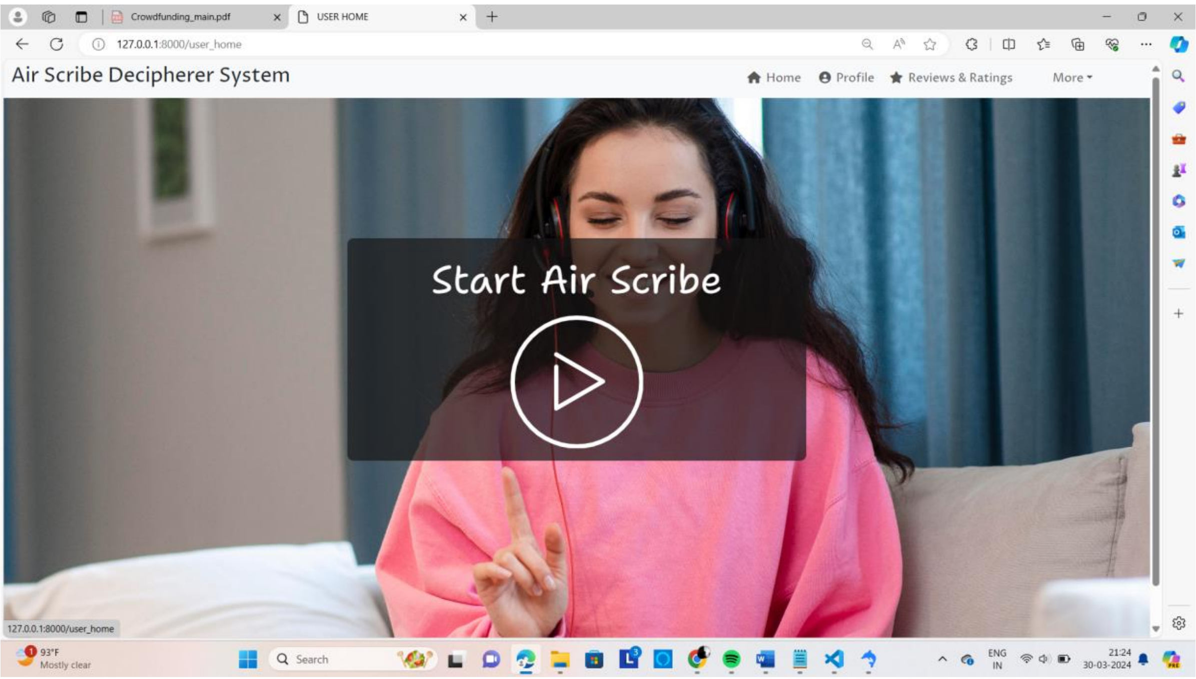
USER REGISTERATION



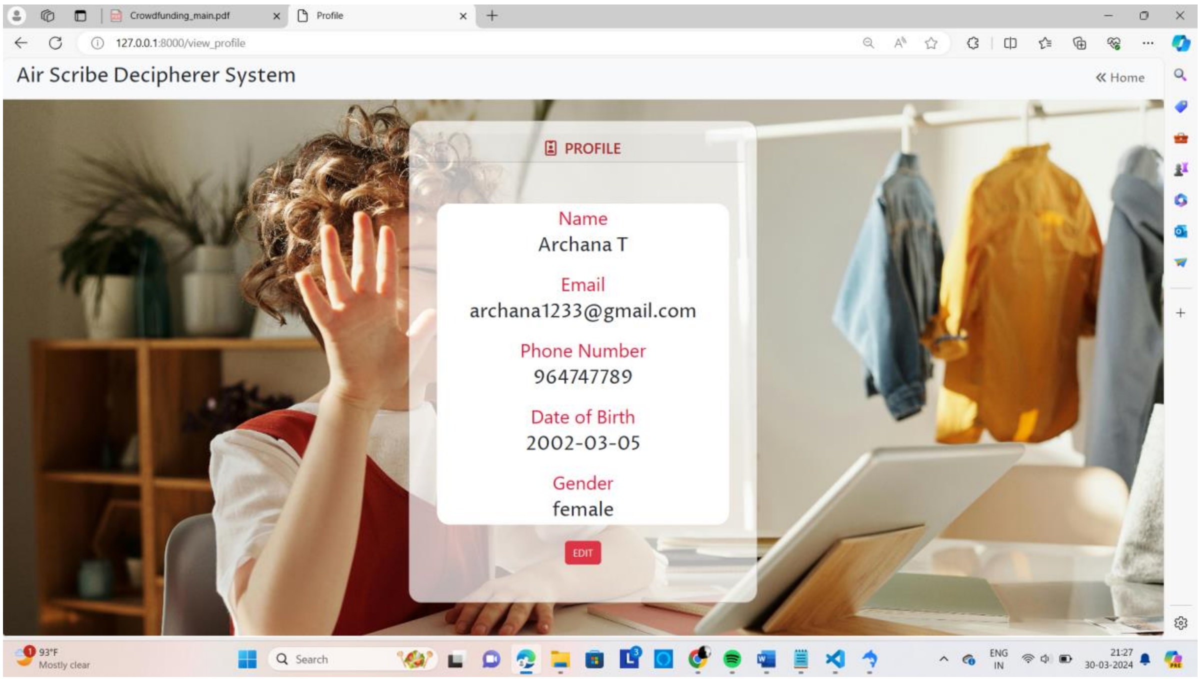
USER REGISTRATION

Name	Enter your Name
Email	Enter your Email address
Phone Number	Enter your Phone
Date of Birth	dd-mm-yyyy
Gender	○ Male ○ Female ○ Others
Password	Enter your Password
Confirm Password	Confirm your Password

USER HOME



VIEW PROFILE



Name
Archana T

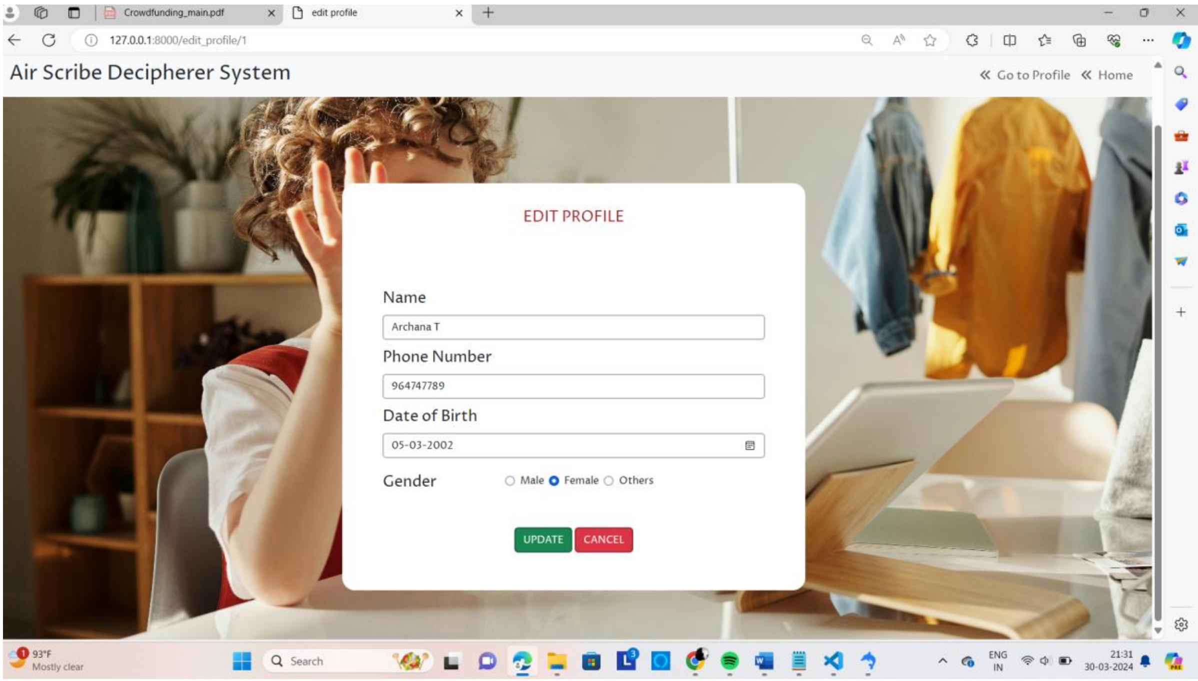
Email
archana1233@gmail.com

Phone Number
964747789

Date of Birth
2002-03-05

Gender
female

EDIT PROFILE



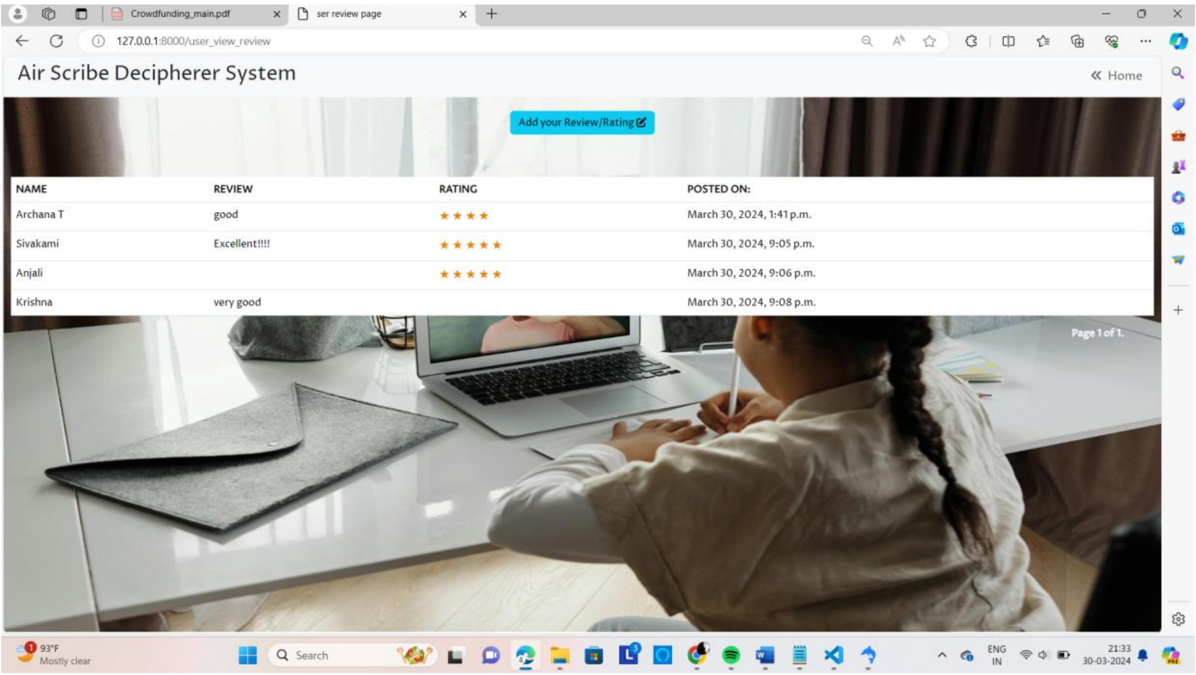
Name: Archana T

Phone Number: 964747789

Date of Birth: 05-03-2002

Gender ○ Male ● Female ○ Others

REVIEWS & RATINGS

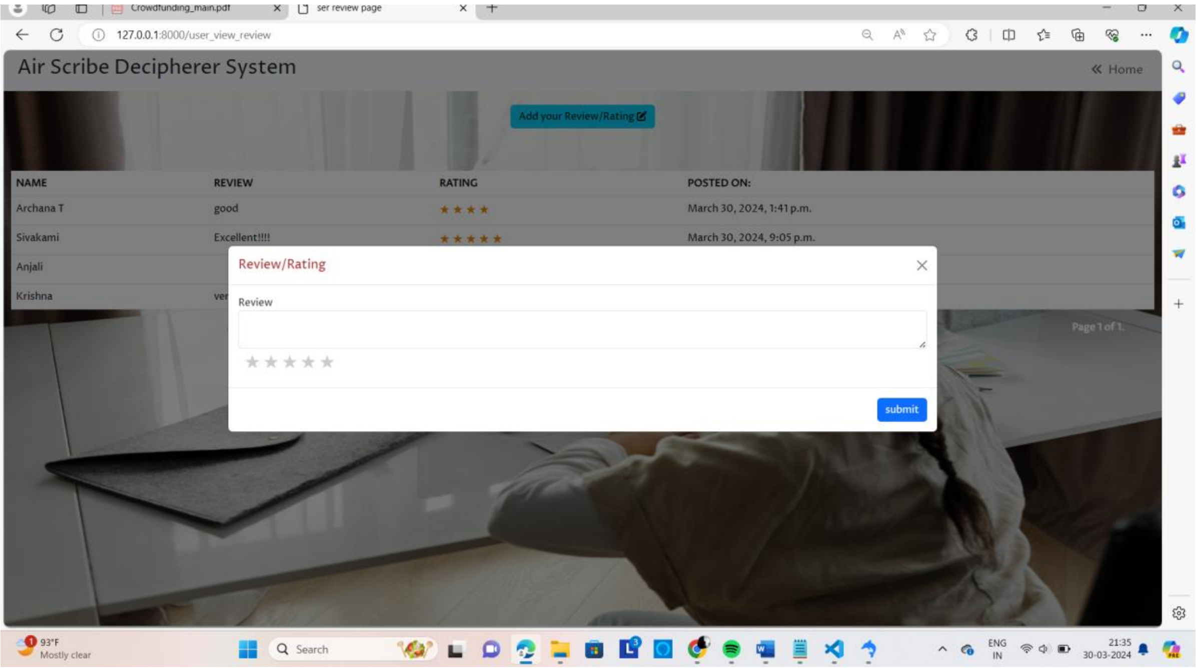


Air Scribe Decipherer System

NAME	REVIEW	RATING	POSTED ON:
Archana T	good	★★★★	March 30, 2024, 1:41 p.m.
Sivakami	Excellent!!!!	★★★★★	March 30, 2024, 9:05 p.m.
Anjali		★★★★★	March 30, 2024, 9:06 p.m.
Krishna	very good		March 30, 2024, 9:08 p.m.

Page 1 of 1.

ADD REVIEW& RATING



Air Scribe Decipherer System

NAME	REVIEW	RATING	POSTED ON:
Archana T	good	★★★★	March 30, 2024, 1:41 p.m.
Sivakami	Excellent!!!!	★★★★★	March 30, 2024, 9:05 p.m.
Anjali			
Krishna	ver		

Review/Rating ×

Review

☆☆☆☆☆

submit

BIBLIOGRAPHY

BIBLIOGRAPHY

Websites

[1] https://www.w3schools.com/python/python_intro.asp

[2] https://www.w3schools.com/bootstrap/bootstrap_get_started.asp

[3] https://viso.ai/computer-vision/opencv/

[4] https://developers.google.com/mediapipe/solutions/vision/gesture_recognizer

[5] https://www.tensorflow.org/about

Books

[1] *Think Python: An Introduction to Software Design, Allen B. Downey,2nd Edition, O'Reilly Media, Inc., 2015*

[2] *Learning OpenCV: Computer Vision with the OpenCV, Gary Bradski, 2nd Edition, O'Reilly Media, Inc., 2008*

[3] *Hands-on ML Projects with OpenCV: Master computer vision and Machine Learning using OpenCV and Python, Mugesh S, 1st Edition., AVA Publishing, 2023*

[4] *Deep Learning with TensorFlow and Keras, Amita Kapoor, 3rd Edition, Packt Publishers, 2022*

[5] *Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications (Grayscale Indian Edition), Chip Huyen, 1st Edition, O'Reilly Media, 2002*