International Journal of Scientific Research and Engineering Development-– Volume X Issue X, Year

Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Asthetix: An AI-Assisted Clinic Management and Beauty Simulation Platform

Lokanindi Akshit*

*(Department of Computer Science and Engineering, Hyderabad Institute of Technology and Management, Hyderabad, India
Email: akshitlokanindi@gmail.com)

----------**********----------

Abstract:

Dermatology clinics, medical spas and aesthetic treatment centers handle appointments, patient information, treatment notes and consultations every day. When these activities are spread across separate tools, staff lose time and records become inconsistent, and patients find it hard to picture what a procedure might change. This paper presents Asthetix, a web-based prototype that places clinic management and an image-based beauty simulation in one platform. The prototype provides role-based sign-in for patients, staff and administrators, patient registration and records, treatment history, appointment booking with conflict prevention, service and user administration, and a simulation page that validates an uploaded photograph and shows a clearly labeled illustrative result. It is implemented as a React single-page application and stores demonstration data in the browser. Six test cases were run against the prototype and all gave the expected result. The evidence supplied is a software workflow; it does not establish clinical validity, a trained computer-vision model, facial-landmark detection in the running application, or measured user outcomes. This paper explains the motivation, design, interface modules, data flow, security considerations, limitations and a plan for future evaluation.

Keywords — **Clinic Management System, Beauty Simulation, Image Processing, Computer Vision, Role-Based Access Control, Appointment Scheduling, Patient Records, Healthcare Software**.

----------**********----------

I. INTRODUCTION

Healthcare and beauty services are adopting digital tools to improve patient experience and clinic efficiency. A dermatology clinic or medical spa schedules visits, keeps medical history and allergies, documents each treatment, and talks with patients about what a treatment can and cannot achieve. When appointment software, record keeping and visual consultation are handled in disconnected tools, staff may re-enter the same details, miss updates and find it difficult to follow one patient through the clinic.

The central idea of Asthetix is to treat these activities as one connected workflow. A patient creates an account and books a slot. Staff register the patient, keep history and allergies, and add treatment notes after each visit. During a consultation, a photograph can be previewed with a simple illustrative change, always marked as a simulation. Administrators maintain the list of services and decide which users hold which role. The platform supports clinic operations and consultation; it does not diagnose, treat, or replace the judgment of a clinician.

A. Motivation

The project is motivated by the need for one place where administrative work and visual consultation meet. A single record of a patient, with appointments and treatment notes beside it, can reduce duplicated entry and make follow-up easier. A clearly labeled preview can also give a patient and a clinician something concrete to discuss, provided that it is never presented as a promise of the result.

B. Problem Statement

Many clinics use separate systems for scheduling, patient records, treatment documentation and aesthetic visualization. This separation creates administrative overhead and makes it hard to keep

 ©IJSRED: All Rights are Reserved

one consistent workflow. Patients also struggle to understand possible changes before a procedure. The problem addressed in this work is the absence of a unified prototype that supports clinic administration and image-based simulation together, with organized records, controlled access and honest communication about the limits of simulated images.

An effective system should keep the number of decisions a user must make small. Forms should say which fields are required, validation messages should explain how to fix a problem, and every simulated image should say what it is. Public trust also matters: patients are more willing to use a booking channel when they can see that their request was received and what its status is.

II. RELATED WORK

Clinic management systems commonly provide patient registration, appointment scheduling, billing and record maintenance, although the features vary by product and many do not include visual simulation. Image-editing and virtual try-on applications provide visual effects but are usually not designed around clinical records or appointment workflows.

On the technical side, computer-vision libraries such as OpenCV support image resizing, filtering and transformation [1], and MediaPipe provides facial-landmark detection that can locate regions of the face for controlled manipulation [2]. Deep-learning frameworks such as PyTorch and TensorFlow allow trained models to generate or transform images [7][8]. Their quality depends on training data, implementation and validation, so in a healthcare setting their outputs must be treated as illustrations. Web frameworks and services such as React [4], FastAPI [3] and PostgreSQL [5] are widely used to build the interface, services and storage of such platforms, and the OWASP Top 10 [6] lists the main security risks that a platform handling personal data should address.

A. Research Gap

The reviewed technologies show that records, scheduling and image processing are each well supported. A practical gap remains in joining them in one understandable workflow with role-based access, so that staff do not switch between systems and patients have one interface for booking and visual consultation. Table I summarizes the comparison used in this work. The entries for existing approaches are general characterizations rather than measurements of particular products.

TABLE I
COMPARISON OF EXISTING APPROACHES AND ASTHETIX

Functionality	Clinic systems	Simulation tools	Asthetix
Patient records	Usually supported	Usually limited	In prototype
Appointments	Usually supported	Usually limited	In prototype
Image visualization	Varies	Core capability	Integrated page
Treatment notes	Usually supported	Usually limited	In prototype
Role-based access	Product dependent	Product dependent	Required design item

B. Contribution of the Present Work

The contribution is an integrated prototype supported by eleven interface screenshots and six executed test cases. The paper explains how the citizen-facing and staff-facing pages relate, identifies the information that each record needs, and states the boundaries of the implementation. In particular, landmark detection and AI-based transformation are described as planned extensions, not completed functions.

III. OBJECTIVES AND SCOPE

The main objective is to design and demonstrate a web-based platform that combines clinic management and image-based simulation. The specific objectives are to provide account registration and sign-in; support role-based access for patients, staff and administrators; keep patient profiles, medical history and allergies; document treatments; let users book, view and cancel appointments while preventing clashes; allow administrators to manage services and user roles; and offer a simulation page that validates images and labels every result.

 ©IJSRED: All Rights are Reserved

A. Scope of the Prototype

The prototype covers three roles. A patient can view a dashboard, book an appointment, cancel own appointments and use the simulation. Staff can in addition register and edit patients, record treatments and change the status of any appointment. An administrator can also add or remove services and change the role of other users. The interface includes summary cards for patients, today's appointments, pending requests and services, and a searchable patient list.

B. Out of Scope

The prototype does not connect to a server or database, integrate with electronic health record systems, send reminders, handle billing, or run a trained computer-vision model. The simulation applies whole-image filters in the browser and does not detect facial landmarks. These functions are not claimed as completed capabilities. They are considered future work once requirements, data availability, privacy controls and validation procedures are established.

The scope is also limited in what it asserts. The screenshots document the interface and the executed tests document the behavior of the described rules. Neither shows that the platform improves clinic efficiency or patient outcomes, which would require a pilot with real users.

IV. PROPOSED SYSTEM

Asthetix is organized around a public area and a signed-in area. The public area contains the home page, services and clinicians, the simulation page and project information. After sign-in, each user sees navigation that matches the role. The interface uses a consistent visual language, with a teal accent for the clinic and a rose accent reserved for simulation notices so that illustrative content is always distinguishable.

A. Patient Management Module

The module stores a name, age, gender, phone number, email, medical history, allergies and consultation notes for each patient, together with a dated list of treatments. Registration validates that the name is present, the age lies between 1 and 120 and the phone number has ten digits. Only staff and administrators may create, edit or delete patient records. Deleting a patient also removes that patient's appointments, and the interface asks for confirmation first.

B. Appointment Module

An appointment stores the patient, service, doctor, date, time, reason and status, which is Pending, Confirmed, Completed or Cancelled. A new request starts as Pending. The booking form lists the available time slots, disables a slot already taken by the same doctor or the same patient on that date, and rejects dates in the past. Patients can cancel their own appointments, and staff can set any status.

C. Beauty Simulation Module

The simulation page lets a user choose a photograph and one of three illustrative changes: skin smoothing, brightening or even tone, with an adjustable strength. The page validates the file type and size, resizes the image, applies the chosen change on an HTML canvas and shows the result beside the original with a comparison slider. Every result carries a visible "simulation, illustrative" banner, and the downloadable image has the same label written into it. The image is processed in the browser and is not uploaded.

D. Administration and Security Module

Administrators add and remove services and change user roles, except their own. Authentication verifies the account, and authorization is checked for each protected page and again inside each data operation, so hiding a menu item is never the only protection.

V. SYSTEM ARCHITECTURE AND METHODOLOGY

The system is a layered web application. The presentation layer contains the pages and forms. A data layer holds the services, doctors, users, patients, appointments and treatment records, and applies validation and permission checks before any change is saved. In the prototype this layer keeps its data in the browser's local storage. Its functions correspond one to one with the endpoints that a FastAPI and PostgreSQL service would provide [3][5], so replacing it with a server later does not change the pages.

 ©IJSRED: All Rights are Reserved

A. Workflow Overview

A user opens the application and signs in. Based on the role, the platform shows the matching dashboard. Staff register patients and manage appointments and treatment records. When requested, a user opens the simulation page, supplies an image, and receives a labeled illustrative result. The general flow is: sign-in, role-based dashboard, patient and appointment services, image processing if requested, and results and record management.

B. Technologies

Table II lists the technologies used in the prototype and those planned for a full deployment.

TABLE II
TECHNOLOGIES

Technology	Purpose
React and Vite [4]	Components and build tooling
React Router	Pages and role-protected routes
Browser local storage	Stand-in database for the demo
HTML canvas	Image resize and filters
FastAPI, PostgreSQL [3][5]	Planned backend and database
OpenCV, MediaPipe [1][2]	Planned landmark detection
PyTorch, TensorFlow [7][8]	Optional trained models

C. Data Model

A patient record has a unique identifier, profile fields, history, allergies, notes and a list of treatments. An appointment refers to a patient, a service and a doctor by identifier and carries a date, a time and a status. A user account has an email, a role and, for patients, a link to a patient record. Keeping appointments separate from patient details makes access control and later retention rules easier to apply.

A complete implementation should also record the previous value, new value, time and acting user whenever a status changes, so that disputes can be resolved. The prototype stores only the current status.

VI. PROTOTYPE IMPLEMENTATION

The evidence for the prototype consists of eleven interface screenshots taken from the running application in a desktop browser. They document the interface and visible workflow and are not treated as proof of performance under real clinic traffic.

A. Home Page

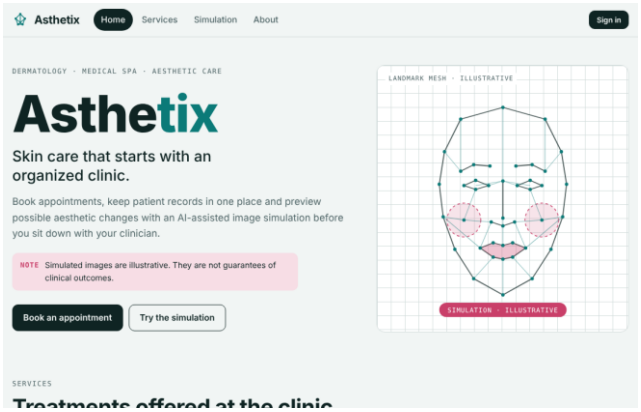


Fig. 1 Asthetix home page with the illustrative landmark mesh.

The home page introduces the platform, offers booking and simulation, lists services and explains the three user roles. It states beside the main action that simulated images are illustrative. The facial-landmark drawing is a static illustration of the planned detection step and is labeled as such.

B. Account Sign-in and Validation

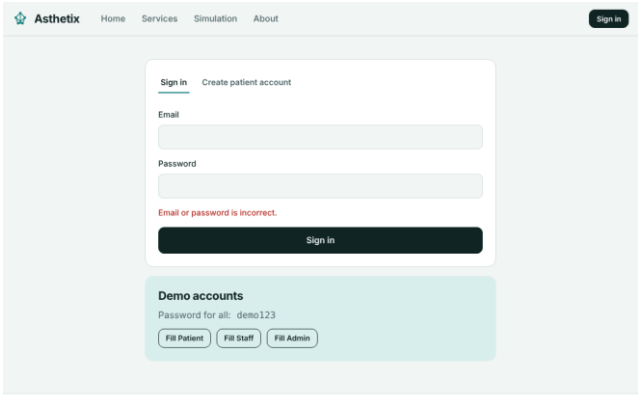


Fig. 2 Sign-in page showing a validation message. Demo accounts can be filled with one click.

The sign-in page accepts an email and password and also lets a new patient create an account. Validation checks that required fields are present, the email has a valid format, the password has at least six characters and the email is not already

 ©IJSRED: All Rights are Reserved

registered. Error messages state what to correct. Demo accounts are provided for the three roles, and the page warns that only demonstration data should be used. Passwords are stored in plain text in the prototype, which is acceptable only for the demonstration and not for any real use.

C. Administrator Dashboard

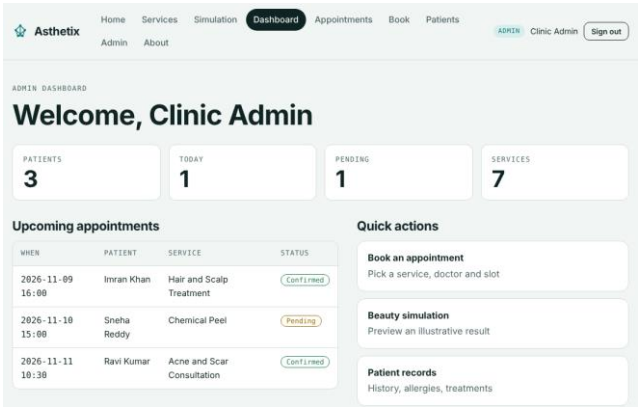


Fig. 3 Administrator dashboard with summary cards and upcoming appointments.

The dashboard is the starting point after sign-in. For staff and administrators it shows the number of patients, appointments today, pending requests and services, followed by upcoming appointments and quick actions. A patient sees only upcoming and completed counts for the patient's own appointments and a short view of the patient's own record. The numbers are calculated from the stored records and change when the data changes.

D. Patient Records

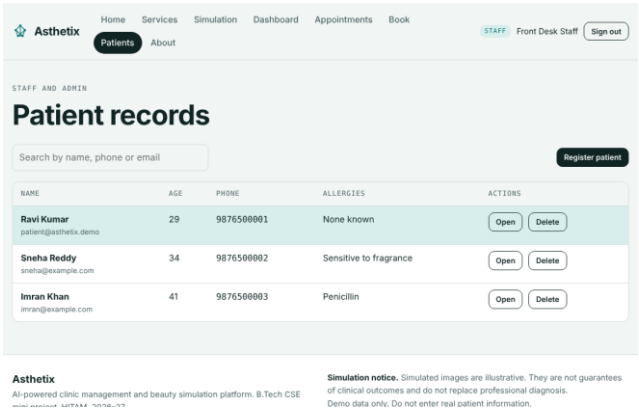


Fig. 4 Staff view of the searchable patient list.

Staff can search patients by name, phone or email, register a new patient, open a record, or delete it after confirmation. The list shows age, phone and allergies so that safety information is visible at a glance.

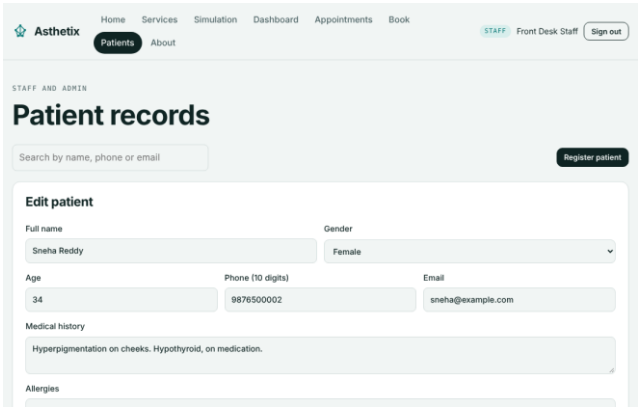


Fig. 5 Patient record with history, allergies, notes and treatment entries.

Opening a record shows the editable profile and a treatment history. A staff member adds a dated entry with a service and a short note, and entries appear newest first. Sensitive fields are not available to the patient role through the staff pages.

E. Appointment Booking

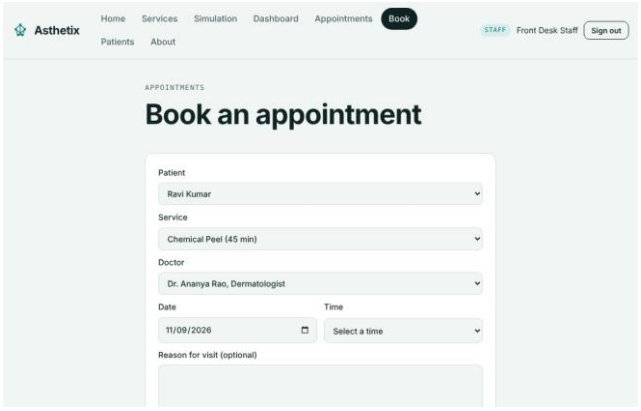


Fig. 6 Booking form with service, doctor, date and time slot selection.

The form asks for a patient (for staff), a service, a doctor, a date and a time. Slots that are already taken are disabled and marked, and a request that still clashes is rejected with a message. A confirmation screen shows the date, time and status after a successful request.

 ©IJSRED: All Rights are Reserved

F. Appointment List

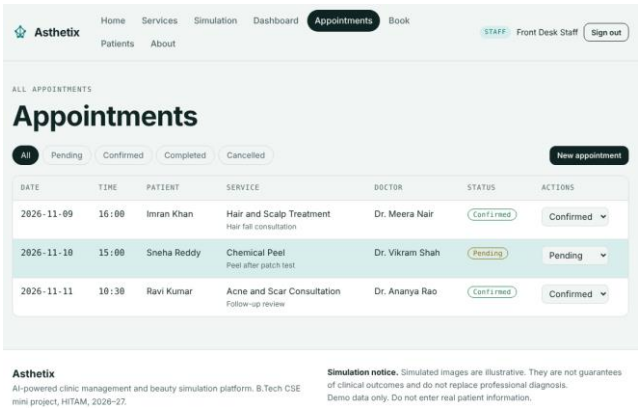


Fig. 7 Appointment list with status filters and staff status control.

The list shows date, time, patient, service, doctor and status and can be filtered by status. Staff change a status with a selector. A patient sees only the patient's own appointments and a Cancel button for those that are still open.

G. Administration

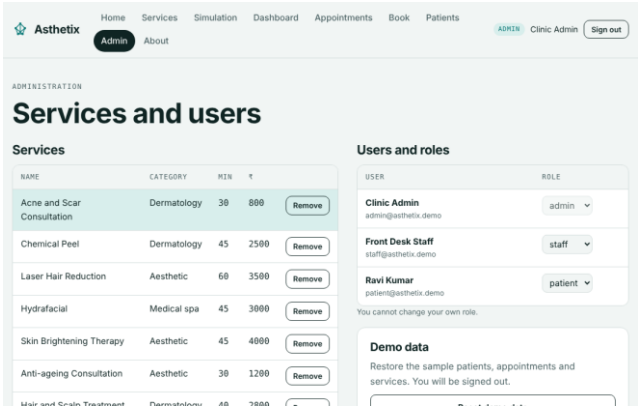


Fig. 8 Administration page for services and user roles.

Administrators add a service with a category, duration and price, remove services, and change a user's role. The role selector for the signed-in administrator is disabled so that an administrator cannot lock themselves out. A reset action restores the demonstration data after a confirmation step.

H. Simulation Input Validation

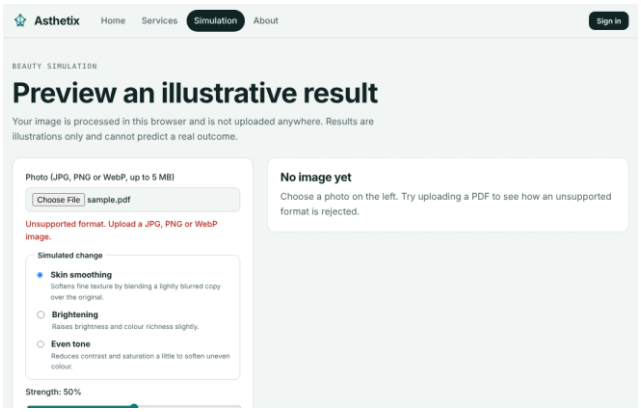


Fig. 9 Rejection of an unsupported file type on the simulation page.

The simulation page accepts JPEG, PNG and WebP files up to 5 MB. A file of another type, or a larger one, is rejected with a message that names the problem and how to fix it. A file that cannot be read as an image is also rejected.

I. Simulation Result

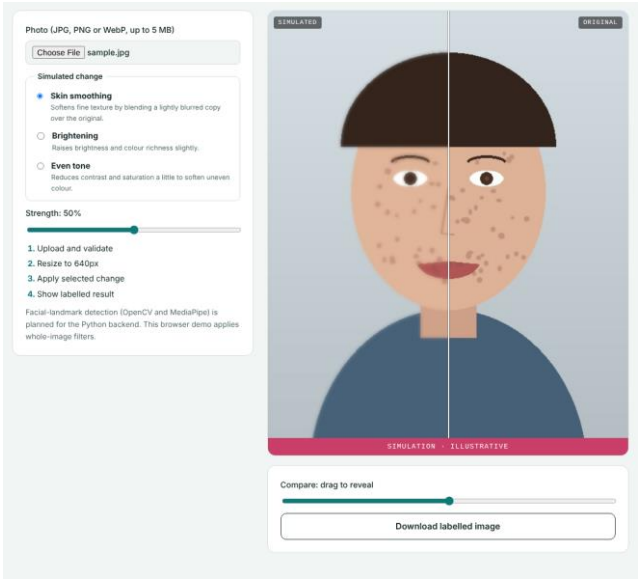


Fig. 10 Illustrative result beside the original with a comparison slider.

For a valid image, the page resizes the picture to a working width, applies the selected change at the chosen strength and displays the result with a comparison slider and a persistent banner. The photograph used here is a synthetic drawing created for testing, not a real person. The change is a whole-image filter. It is not guided by facial landmarks and does not model any treatment, so the

 ©IJSRED: All Rights are Reserved

result must not be read as a prediction of any clinical outcome.

J. Access Control

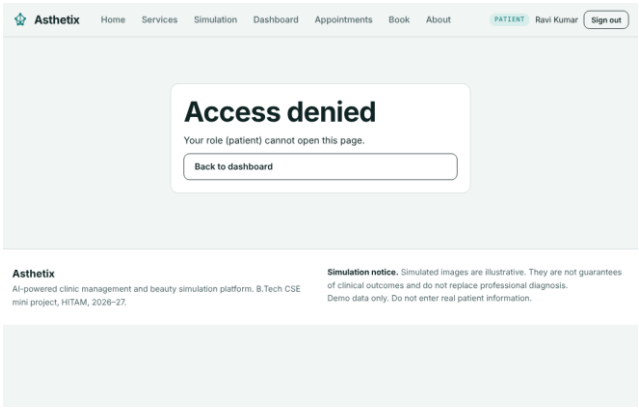


Fig. 11 A patient attempting to open a staff-only page.

When a signed-in patient opens a page reserved for staff or administrators, the application shows an access-denied message, and a visitor who is not signed in is redirected to the sign-in page. The same permission rules are applied again in the data layer for each write operation.

VII. SECURITY, PRIVACY AND RELIABILITY

A clinic platform collects personal details, medical history and photographs, so it should collect only what it needs and limit access by role. In a production service, passwords must be stored with a modern salted hashing method, data must travel over encrypted connections, and uploads must be validated on the server by type and size. Administrative routes require authorization on the server, not just hidden menu items. Logs should avoid storing sensitive medical details, and patient images and records should never be exposed through public paths [6].

A. Privacy Considerations

Retention periods, consent and access rules must be defined before any real clinical use. Users should be told why images and health information are collected, who can see them and how a correction or deletion can be requested. The demonstration prototype does not meet these requirements and must be used with sample data only.

B. Reliability

The prototype keeps booking and patient forms intact when a validation error occurs, so the user can correct the entry without retyping. A full service should also handle network failures and repeated submissions without creating duplicate records, return a unique identifier after a request is accepted, and allow a safe retry.

VIII. TESTING AND EVALUATION

Six test cases were executed against the running prototype through the browser interface. Table III lists each scenario with the observed result. Each was run once on one desktop browser, so they verify the described rules and are not a statistical evaluation.

TABLE III
TEST CASES AND OBSERVED RESULTS

ID	Scenario	Observed result	Status
TC01	Register a patient with valid fields	Record created and listed	Pass
TC02	Submit with missing or invalid fields	Messages: name required; phone must have 10 digits	Pass
TC03	Book an available slot; check the same slot	Appointment requested; slot then disabled for that doctor	Pass
TC04	Open staff and admin pages as a patient or visitor	Access denied; visitor sent to sign-in	Pass
TC05	Upload a PDF; upload a file over 5 MB	Both rejected with clear messages	Pass
TC06	Run simulation on a valid image	Result shown with illustrative banner	Pass

A. Further Evaluation Plan

A fuller evaluation should include usability tests with representative patients and staff who complete tasks such as registering, booking, finding a record and interpreting a status, with completion time, errors and requests for help recorded. Functional tests should cover boundary values, conflicting appointments and failed requests through the service interface using a tool such as Postman. Security tests should attempt to read another user's record, change a status as a patient and reuse an expired session. If a trained model is added, its output should be assessed against labeled examples

 ©IJSRED: All Rights are Reserved

by qualified reviewers, and its limits reported openly.

B. Discussion

The prototype brings together patient identity, records, appointments, services and a visual preview in one workflow, which may be easier to follow than separate tools. The interface alone cannot guarantee that a clinic runs better. Staffing, clear procedures, reliable updates and clinical governance are required. The system should be seen as an information and coordination layer that supports the clinic's existing processes.

IX. ADVANTAGES AND APPLICATIONS

- Centralized approach to clinic administration.
- Organized patient profiles, history and appointment information.
- An interactive way to discuss possible aesthetic changes, always labeled as illustrative.
- Less repeated administrative entry.
- Modular design that can add services later.
- Explicit role-based access for sensitive information.

Asthetix may suit dermatology clinics, medical spas, aesthetic treatment centers and other organizations that need patient workflow management and image-based consultation support. Its suitability for real use depends on security, privacy, usability and clinical validation.

X. LIMITATIONS AND FUTURE ENHANCEMENTS

The present work is a prototype-level demonstration. It stores data in the browser and has no server, so it offers no shared records, secure password storage or audit history. The simulation applies whole-image filters; it does not detect landmarks, use a trained model or predict any real outcome. Quality depends on image quality and the chosen change. The tests were run once on one browser, and no study with real clinic users was carried out.

Future work should proceed in stages. First, a backend with hashed passwords, HTTPS, server-side validation and a relational database should replace the browser storage. Second, a Python service using OpenCV and MediaPipe [1][2] should add landmark-guided changes, and a validated model could be considered later [7][8]. Third, appointment reminders, a mobile application, analytics dashboards, integration with health record systems where permitted, and stronger consent and audit controls can be added. A small pilot in one clinic should then assess usability, time saved and administrative fit before any broader claim is made.

XI. CONCLUSION

Asthetix presents a web-based approach that combines clinic management with an image-based simulation. Its interface supports role-based sign-in, patient records and treatment history, appointment booking with conflict prevention, service and user administration, and a validated, clearly labeled simulation page. Eleven screenshots and six executed test cases illustrate these functions and the intended workflow. The prototype provides a foundation for further development, but its usefulness must be established through a secure backend, privacy and consent controls, and evaluation with real clinicians and patients before any claim of clinical or operational impact is made.

The principal value of the proposed system is the structured connection between a patient, the clinic's records and a visual consultation aid. By keeping these together and being honest about what a simulation is, Asthetix can help make aesthetic care workflows more organized and transparent.

ACKNOWLEDGMENT

The author thanks the project guide, Mr. T. Raghavendra Gupta, the Head of the Department, Dr. Dharmendra K Roy, and the Department of Computer Science and Engineering, Hyderabad Institute of Technology and Management, for their guidance and support.

REFERENCES

[1] OpenCV, "OpenCV documentation." [Online]. Available: https://docs.opencv.org/
[2] Google, "MediaPipe documentation." [Online]. Available: https://ai.google.dev/edge/mediapipe/solutions/guide
[3] FastAPI, "FastAPI documentation." [Online]. Available: https://fastapi.tiangolo.com/
[4] React, "React documentation." [Online]. Available: https://react.dev/
[5] PostgreSQL Global Development Group, "PostgreSQL documentation." [Online]. Available: https://www.postgresql.org/docs/

 ©IJSRED: All Rights are Reserved

[6] OWASP Foundation, "OWASP Top 10." [Online]. Available: https://owasp.org/www-project-top-ten/
[7] PyTorch, "PyTorch documentation." [Online]. Available: https://pytorch.org/docs/
[8] TensorFlow, "TensorFlow guide." [Online]. Available: https://www.tensorflow.org/guide

 ©IJSRED: All Rights are Reserved