

SYDE-660 Project Final Report

**SkinSense: An AI-Assisted Skin Lesion Documentation
and Tracking System**

Zirui Chen (21215488), Kaiyuan Zhang (21180557)

Jiahui Yu (20832759), Christy Cui (21206718)

Team #2

SYDE 660 - Systems Design Graduate Workshop I

Prof. Cosmin Munteanu

August 15th, 2026

Executive Summary

This report presents a non-diagnostic, AI-assisted web application (SkinSense) designed to improve how general users document and track skin lesions over time. The project addresses three key gaps in current skin lesion self-documentation: inconsistent patient-captured images, fragmented lesion histories, and missing contextual information such as body location, symptoms, and dates. Rather than attempting to diagnose or screen for skin cancer, SkinSense is designed to support documentation and communication between users and healthcare professionals.

The proposed system combines guided image capture, automated image-quality checking, body-location recording, symptom documentation, lesion segmentation, quantitative morphology metrics, timeline comparison, and provider-ready report export. The system was developed as a functional capstone prototype using a modular architecture. The frontend is a React-based single-page application that supports the user workflow, lesion archive, timeline comparison, and PDF report export. The backend is a FastAPI service running in a GPU-supported environment, using computer vision and AI components including image preprocessing, SAM-based lesion contour extraction, OpenCV-based metric computation, and MedGemma-assisted objective description generation.

The prototype demonstrates a complete end-to-end workflow. A user can create a lesion record, upload or capture an image, mark the lesion location on a body diagram, complete a short symptom questionnaire, and submit the image for analysis. The user can then review the generated contour, description, and metrics, save the observation to an archive, compare records over time, and export a report for discussion with a healthcare provider. Technical verification confirmed that the frontend and backend can exchange data through the analysis endpoint and that the prototype can support image submission, result display, record saving, timeline review, and report generation.

Stakeholder validation supported the project's non-diagnostic positioning. Dermatologist feedback indicated that SkinSense would be most useful as a monitoring and communication tool, especially through its body-location selection, symptom questionnaire, timeline comparison, and provider-ready report. General user feedback from eight participants suggested that the workflow was understandable and that participants were generally comfortable using the app to organize lesion information before seeing a healthcare provider. Users particularly valued timeline comparison, report export, and guided image capture.

The project also identified important limitations and future improvement areas. Segmentation quality remains a key technical dependency because inaccurate contours can affect all downstream metrics. User feedback also showed that result metrics require clearer plain-language explanations, and dermatologist feedback emphasized the need for cautious wording, privacy transparency, and stronger guidance for seeking professional care when concerning symptoms are present. Overall, SkinSense demonstrates the feasibility of an AI-assisted documentation tool for longitudinal lesion tracking, while remaining clearly bounded as a functional prototype that requires further technical refinement and broader stakeholder validation before consideration for real-world use.

Executive Summary.....	1
1. Problem Context and Proposed Solution.....	4
1.1 Background.....	4
1.2 Current Gaps in Skin Lesion Self-Documentation.....	4
1.3 Proposed Solution.....	5
2. Stakeholder Needs and Design Objectives.....	5
2.1 Stakeholder Groups.....	5
2.2 Needs Assessment.....	6
2.3 Design Objectives.....	6
2.4 Design Boundaries.....	6
3. Project Approach and Management.....	7
3.1 Project Scope and Design Strategy.....	7
3.2 Work Breakdown Structure.....	7
3.3 Schedule Planning.....	9
3.4 Scope Control and Project Risks.....	10
4. System Requirements and Design Overview.....	10
4.1 System Scope.....	10
4.2 User and Functional Requirements.....	11
4.3 Technical Requirements.....	11
4.4 Safety and Ethical Requirements.....	12
5. System Design.....	15
5.1 Frontend System Design.....	15
5.2 Backend and AI Pipeline Design.....	15
5.3 Engineering Analysis and Metrics.....	16
5.4 Frontend-Backend Integration.....	16
6. Prototype Implementation.....	17
6.1 Frontend Prototype.....	17
6.2 Backend Prototype.....	17
6.3 Integration Status.....	18
6.4 Prototype Demonstration Scenario.....	18
7. Testing and Validation.....	24
7.1 Technical Verification.....	24
7.2 Usability Testing.....	25
7.3 Stakeholder Feedback.....	25
8. Recommendations and Future Work.....	28
8.1 Short-Term Improvements.....	28
8.2 Future Technical Development.....	29
8.3 Future Stakeholder Validation.....	29
9. Conclusion.....	30
References.....	30

Appendices.....	31
Appendix A: Detailed Stakeholder Analysis.....	31
Appendix B: Dermatologist Stakeholder Validation Interview Guide.....	33
Appendix C: User Validation Questionnaire.....	35
Appendix D: General User Validation Questionnaire Raw Results.....	37

1. Problem Context and Proposed Solution

1.1 Background

Melanoma remains an important public-health concern in Canada. In Ontario, 3,882 cases of cutaneous melanoma were recorded in 2022, accounting for 4.6% of all newly diagnosed cancers in the province [1]. Because changes in a skin lesion over time can provide useful context during clinical assessment, longitudinal visual monitoring is often used to support melanoma surveillance, particularly for high-risk patients. The 2024 European melanoma guideline recognizes sequential digital dermoscopy and whole-body photography as useful monitoring approaches for selected high-risk patients [2]. However, these methods are intended to support professional assessment, not replace clinical examination or histopathological confirmation when melanoma is suspected.

Outside the clinical setting, patients may notice changes in a lesion before receiving an assessment or between healthcare visits. In practice, however, they often rely on memory or occasional smartphone photographs to track these changes. These photos may be captured under inconsistent lighting, distances, angles, and framing, and may not be linked to dates, body location, symptom information, or objective measurements. As a result, healthcare professionals may receive limited longitudinal context about how a lesion has changed. This creates an opportunity to improve the consistency and continuity of skin lesion documentation between healthcare visits.

1.2 Current Gaps in Skin Lesion Self-Documentation

- **Inconsistent image quality**

The first gap is inconsistent image quality. Patient-captured photographs may be blurred, poorly illuminated, overexposed, incorrectly framed, or taken from different distances, making direct comparisons unreliable. In a 2024 pilot study, hospital staff produced significantly higher-quality smartphone dermoscopic images than patients, and 18.2% of patient-captured images were unsuitable for follow-up assessment [3]. Although the study involved dermoscopic attachments rather than standard smartphone photography, it demonstrates the importance of capture guidance and quality control.

- **Limited longitudinal organization**

The second gap is limited longitudinal organization. Many consumer tools emphasize one-time image assessment, while personal photographs are often stored without consistent body-location labels, symptom records, or comparable measurements. This makes it difficult to determine whether an apparent difference represents a physical change or simply a change in photography conditions. The project therefore prioritizes a chronological record for each lesion, including changes in border circularity, eccentricity, colour distribution, and axis ratio.

- **Missing clinical context and inappropriate diagnostic framing**

The third gap is missing clinical context and inappropriate diagnostic framing. An image alone cannot document lesion duration, growth, itching, bleeding, elevation, or other user-observed changes. In addition, unexplained diagnoses or risk predictions may create anxiety or false reassurance. For these reasons, the system must use conservative, descriptive language and direct users with persistent concerns to qualified healthcare professionals.

1.3 Proposed Solution

The proposed solution is **SkinSense, a web-based, AI-assisted skin lesion documentation and longitudinal tracking platform designed for general users**. It combines guided smartphone photography, automated image-quality validation, a structured questionnaire, morphological measurements, chronological visualization, and provider-ready reporting. These functions were selected because they directly address the identified problems of inconsistent images, fragmented histories, and missing contextual information.

During image capture, visual overlays guide the user regarding lighting, focus, framing, and camera distance. Computer-vision methods then evaluate skin visibility, blur, brightness, exposure, and positioning. Images that do not satisfy the established criteria are rejected, and the user is prompted to retake them. This quality gate improves consistency but does not certify that an image is clinically diagnostic.

For each accepted image, the platform collects user-reported information such as body location, duration, growth, itching, bleeding, and elevation. Image segmentation is then used to identify the lesion boundary and calculate descriptive features, including circularity, eccentricity, colour-histogram change, and axis ratio. Repeated entries are stored chronologically and displayed through neutral trend charts.

Finally, the system generates a provider-ready PDF containing the image timeline, questionnaire responses, morphological descriptions, and measured trends. The platform is not intended to diagnose, screen for, confirm, or exclude skin cancer. Its purpose is to improve documentation and communication so users can provide healthcare professionals with a more complete and consistent history. All clinical interpretation and medical decisions remain the responsibility of qualified healthcare providers.

2. Stakeholder Needs and Design Objectives

2.1 Stakeholder Groups

The primary stakeholders of SkinSense are general users who want to document and monitor skin lesions over time. This group includes individuals who have noticed a new or changing lesion, as well as higher-risk users with many moles, frequent sun exposure, a family history of skin cancer, or previous suspicious lesions.

Secondary stakeholders include dermatologists and family physicians. Dermatologists may use the exported lesion history as supporting information during assessment, while family physicians may use the record to better understand whether specialist referral should be considered. Caregivers and family members are also relevant stakeholders because they may help children, older adults, or users with limited mobility capture images and maintain records. A detailed stakeholder analysis, including stakeholder roles, needs, concerns, and design implications, is provided in [Appendix A](#).

2.2 Needs Assessment

The main need of users is not an automated diagnosis, but a reliable way to determine whether a lesion has visibly changed and to communicate that history to a healthcare provider. Users therefore need guidance for capturing consistent images, a simple method for recording lesion location and symptoms, and an organized timeline that allows observations to be compared over time.

Healthcare providers need clear and objective information rather than unexplained AI predictions. Useful information includes dated images, body location, reported symptoms, lesion contours, morphological measurements, and visible trends. Users also need understandable language, strong privacy protection, control over their records, and clear communication that the system does not replace professional medical assessment.

2.3 Design Objectives

The primary design objective of SkinSense is to improve the consistency and usefulness of skin-lesion documentation over time. The system should guide users in capturing acceptable images and use automated quality checks to identify problems such as blur, poor lighting, incorrect framing, or insufficient skin visibility. It should also allow users to record the lesion's body location and provide contextual information, including duration, itching, bleeding, growth, and elevation.

Another objective is to organize repeated observations of the same lesion into a clear chronological record. The system should generate an objective lesion contour, a non-diagnostic visual description, and quantitative shape and colour metrics that can be compared across different dates. These results should be displayed through timelines and trend visualizations so that users can recognize and document visible changes more easily.

The system should also support communication with healthcare professionals by allowing users to export a provider-ready PDF containing lesion images, dates, body location, symptom responses, descriptions, and measured trends. The interface should remain simple, browser-based, and accessible without requiring software installation. Finally, all results should use neutral language, clearly state that the system is not diagnostic, protect user information, and allow users to maintain control over and delete their records.

2.4 Design Boundaries

SkinSense is strictly a documentation and longitudinal tracking tool. It does not diagnose skin cancer, classify lesions, generate risk scores, recommend treatment, or confirm that a lesion is safe. All clinical interpretation and medical decisions remain the responsibility of qualified healthcare professionals.

The system is also limited by the quality of user-provided images and the accuracy of lesion segmentation. Poor lighting, blur, framing differences, or low contrast may reduce the reliability of the calculated metrics. The prototype does not use user accounts or a permanent server-side database; images and lesion records are stored within the client application, and users can delete their records. The current system should therefore be treated as a functional prototype requiring further clinical and stakeholder validation before real-world use.

3. Project Approach and Management

3.1 Project Scope and Design Strategy

This project was managed as a systems design capstone project with the goal of developing a functional, testable prototype. While the design objectives in Chapter 2 define what the system should achieve for users and healthcare providers, this section defines what the team committed to implementing within the project timeline and how the work was structured.

The project scope was limited to a non-diagnostic documentation and longitudinal tracking prototype. The team prioritized a complete end-to-end workflow in which a user can create a lesion record, submit an image for analysis, review objective outputs, save the observation to an archive, compare records over time, and export a provider-ready report. Diagnostic classification, cancer risk prediction, treatment recommendation, clinical decision-making, account management, and permanent server-side storage were excluded from the project scope.

The design strategy followed a modular minimum viable product approach. The system was divided into a browser-based frontend and a GPU-supported backend inference service. This allowed the user interface, archive workflow, questionnaire, timeline, and report export to be developed separately from the image-processing and AI pipeline. Early frontend work used mocked analysis outputs so that the team could validate the interaction flow before the backend was fully integrated.

This strategy reduced schedule risk and allowed multiple workstreams to progress in parallel. It also supported safer healthcare-related design by keeping the system focused on documentation and communication rather than diagnosis.

3.2 Work Breakdown Structure

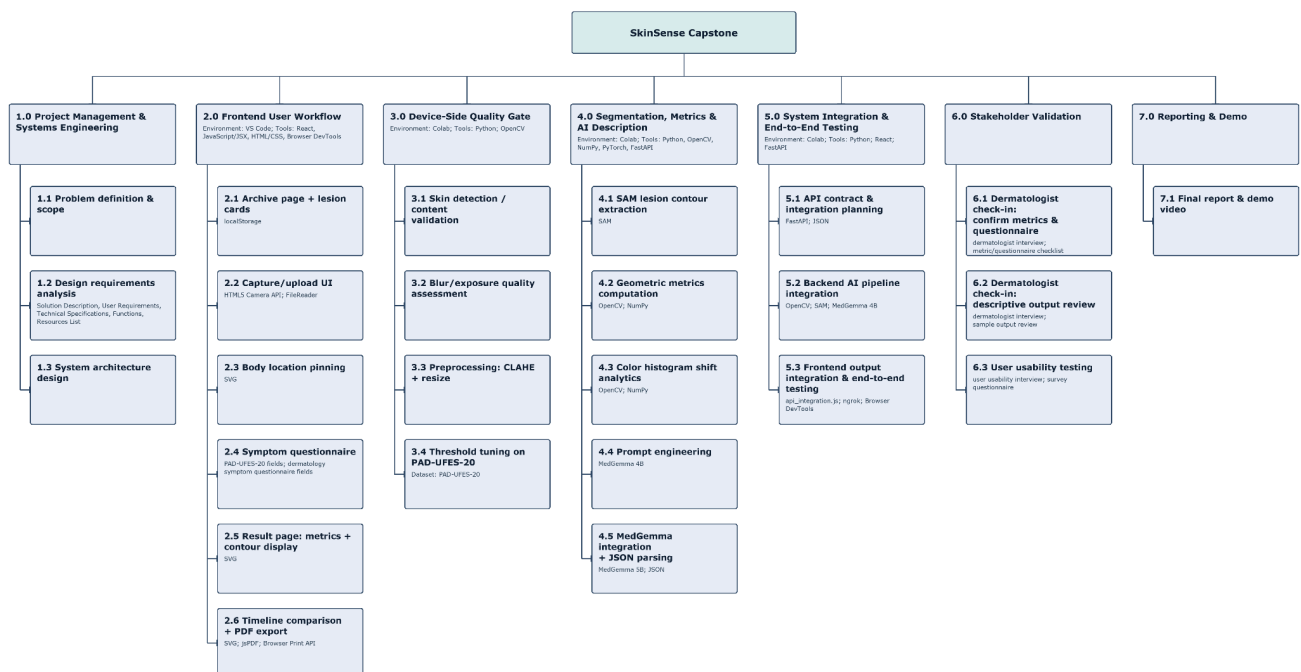


Figure 3.1 Work Breakdown Structure

The Work Breakdown Structure divided the project into major work packages that reflected both the design process and the technical implementation plan. The first work package focused on project management and systems engineering, including problem definition, design requirement analysis, and system architecture planning. These tasks established the project boundary and guided later design decisions.

The frontend user workflow was treated as a separate work package because it defined how users would interact with the system. This package included the archive page, lesion cards, capture/upload interface, body location pinning, symptom questionnaire, result page, timeline comparison, and PDF report export. Separating this work package allowed the team to build and test the user workflow even before the backend analysis pipeline was fully operational.

The backend-related work was divided into two main technical work packages. The first was the device-side quality gate, which included skin detection or content validation, blur and exposure assessment, preprocessing, and threshold tuning. The second was segmentation, metrics, and AI description, which included SAM contour extraction, geometric metric computation, colour variation analysis, prompt engineering, and MedGemma integration. This separation reflected the difference between image acceptance/preprocessing and the generation of final analysis outputs.

A separate system integration and end-to-end testing work package was included to connect the frontend, backend API, image-processing pipeline, AI outputs, archive logic, tracking features, and report export. This was necessary because individual components working separately did not guarantee that the full user workflow would function as a complete system.

The final work packages covered stakeholder validation and reporting. Stakeholder validation included dermatologist-oriented feedback and general user usability testing, while reporting and demo preparation focused on documenting the design process, prototype implementation, testing outcomes, and final system demonstration.

3.3 Schedule Planning

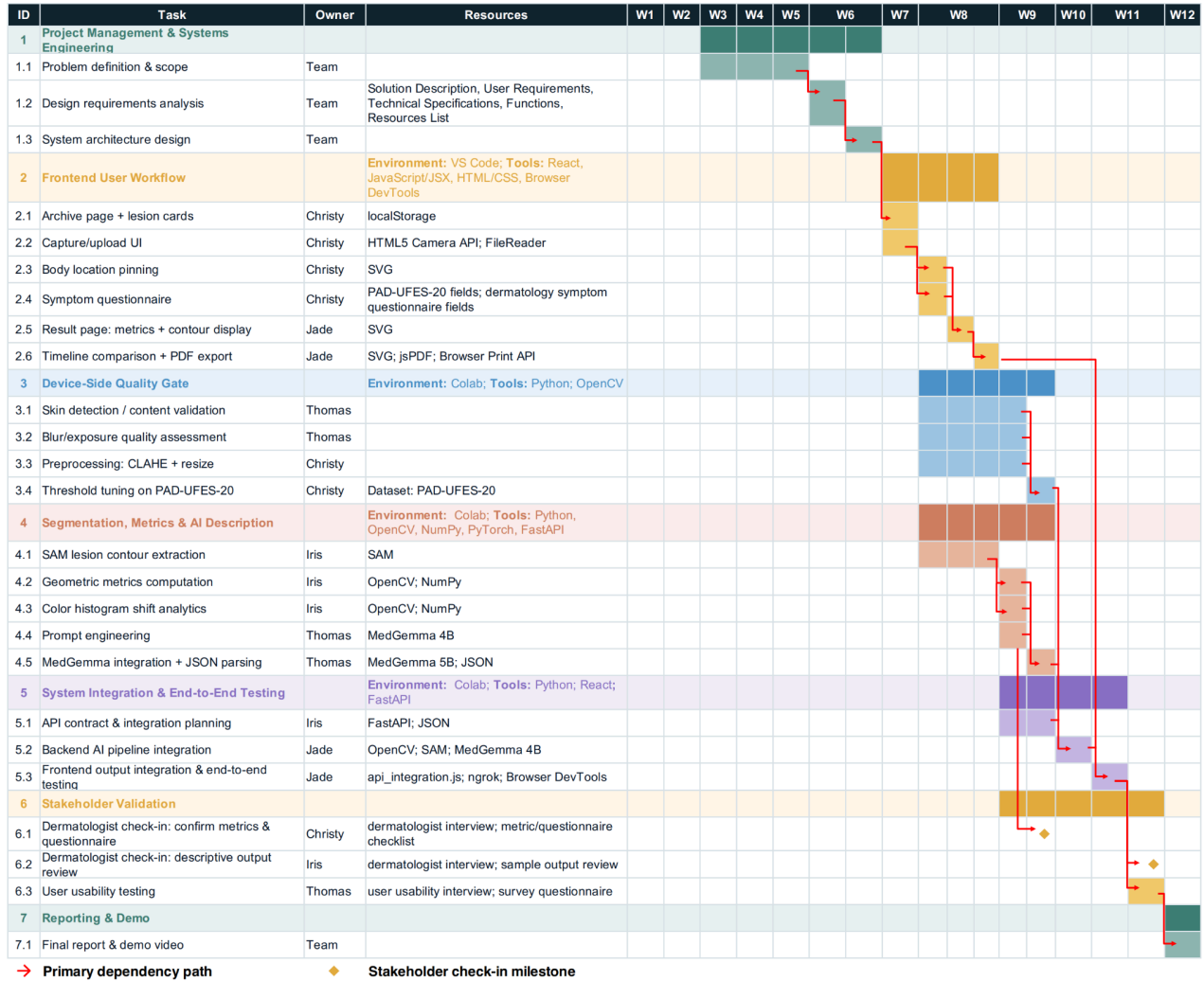


Figure 3.2 Gantt Chart

The project schedule was organized over a 12-week timeline. The early weeks focused on problem definition, stakeholder needs, design requirements, and architecture planning. These tasks were scheduled first because later implementation depended on a clear understanding of the problem boundary and non-diagnostic system scope.

Implementation work was then planned as parallel workstreams. The frontend user workflow was developed alongside the backend image-processing and AI pipeline. This allowed the team to make progress on the archive, upload flow, body map, questionnaire, result page, timeline, and report export while separate work continued on image quality checks, segmentation, metric computation, and AI-generated description.

The schedule intentionally included an integration phase. During this phase, the team defined the API contract, connected the React frontend to the FastAPI backend, integrated backend outputs into the result page, and tested the full workflow from image upload to report export. This integration phase was

important because the project depended not only on individual technical modules, but also on reliable data flow between them.

Stakeholder validation and usability testing were scheduled after the main prototype workflow became functional. This allowed participants to respond to a system that represented the intended user experience rather than isolated interface mockups. The final stage of the schedule was reserved for report writing, figure preparation, and demo video development.

3.4 Scope Control and Project Risks

Scope control was a major part of the project management strategy because SkinSense combines several complex areas: frontend interaction design, image quality assessment, lesion segmentation, quantitative metric computation, AI-generated description, backend integration, and report export. To keep the project feasible, the team maintained a clear MVP boundary and focused on a working non-diagnostic documentation workflow.

The first major risk was technical overextension. Each computational component, especially segmentation and AI description generation, could have expanded into a full independent research problem. The team controlled this risk by using existing tools such as SAM, OpenCV, FastAPI, and MedGemma rather than developing every algorithm from scratch.

The second risk was integration complexity. The frontend and backend were developed in different environments, with the frontend running as a React application and the backend running through a Colab-based FastAPI service exposed by ngrok. This created risks related to API compatibility, CORS, model loading time, response formatting, and backend availability. The team managed this risk by defining a narrow /analyze endpoint, using structured JSON responses, and adding frontend loading and error states.

The third risk was output reliability. Because all quantitative metrics depend on the segmentation contour, inaccurate masks can affect circularity, asymmetry, colour variation, and axis ratio at the same time. The team addressed this risk by presenting metrics as documentation aids rather than clinical conclusions and by avoiding diagnostic labels, risk scores, or treatment recommendations.

The fourth risk was stakeholder interpretation. In a healthcare-related system, users may over-trust AI-generated outputs if the interface appears too authoritative. To reduce this risk, the project consistently framed SkinSense as a documentation and communication tool. The system's language, report content, and design boundary all emphasize that clinical interpretation and medical decisions remain the responsibility of qualified healthcare professionals.

4. System Requirements and Design Overview

4.1 System Scope

SkinSense is a web-based application that documents and tracks skin lesions over time. The system allows users to take photographs of a lesion, mark its location on a body map, enter symptom-related

information, and receive a structured record of morphological description and four metrics measuring shape and color. Records are grouped by lesion, so repeated records of the same lesion are compared on a timeline, with changes visualized and can be exported to a report in table format, for further sharing with a dermatologist.

The scope of the system is strictly bounded. It is explicitly a documentation app without diagnosis. It does not provide classifications of lesions, risk levels, or medical advice. The app itself is not a replacement for a dermatologist's examination and diagnosis. The scope is a result of consideration for ethics and safety.

4.2 User and Functional Requirements

Since the primary users are individuals monitoring skin lesions over time, the user's need is not “is this lesion dangerous?” but “has this lesion changed, and if and how I can show that to my doctor”. For the secondary users, dermatologists, their need is objective records of lesions.

The functional requirements are thus summarized in **Table 4.1** below.

Table 4.1 Functional Requirements

ID	Functional Requirement	Explanation
1	Upload a lesion photograph, with an automated image quality check before acceptance.	Poor images generate unreliable measurements and reduce comparability over time.
2	Record the location of the lesion by placing a pin on a front/back body map.	Location is the key identification to match a new record to the same lesion already recorded.
3	Automatically detect whether the new pin is near an existing lesion and ask the user to confirm if they are the same one.	Users do not need to remember which lesion a photo is matched to.
4	Collect symptom-related information in a structured form (duration, itch, bleeding, size change, elevation).	Images alone cannot capture the context of a lesion.
5	Generate an objective morphological description and four quantitative metrics measuring shape and color for each lesion.	Descriptions and metrics help quantify each lesion.
6	Display metric trends of the same lesion over time and provide pairwise comparison.	Change over time is the primary clinical signal the system is designed to surface.
7	Export a lesion record to a PDF report containing images, metrics, and descriptions.	Record to be shown to a dermatologist outside the application.

4.3 Technical Requirements

The technical requirements consist of running two vision models (segmentation and description) with images uploaded by the users, which are computationally intensive and require GPU acceleration. So

the system was separated into a browser-based frontend and a GPU-backed inference service, communicating through a documented HTTP interface. This separation also allows either half to be replaced independently. The frontend does not depend on which models the backend runs, only on the response schema.

Table 4.2 Technical Requirements

ID	Technical Requirement	Implementation
1	Automatic segmentation of the lesion from a single photograph of skin with lesion.	Segment Anything Model (SAM, ViT-B checkpoint) with an automated central box prompt.
2	Generation of objective descriptive text based on the lesion image.	MedGemma 4B (google/medgemma-4b-it), used zero-shot with a constrained prompt.
3	Deterministic, reproducible quantitative metrics independent of the language model.	OpenCV contour analysis on the SAM mask (see Section 5.3).
4	GPU inference environment accessible from a browser client.	FastAPI service on a Colab T4 runtime, exposed through an ngrok HTTPS tunnel with CORS enabled.
5	Browser-based client requiring no end-user installation once deployed.	React single-page application built with Vite.
6	Explicit failure signalling when the inference service is unavailable; synthetic output must never be substituted for genuine analysis.	Connection failures surface an error state and return the user to the previous step; no record is created.

4.4 Safety and Ethical Requirements

The primary safety and ethical concern is that the SkinSense app must not act as a diagnostic tool that outputs risk levels or clinical recommendations because of two possible consequences. The first is false diagnosis of an actual dangerous lesion, which is more harmful, delaying a user from seeking care, and the second is false diagnosis of a non-dangerous lesion, which causes unnecessary anxiety for users. Other safety and ethical concerns also include data privacy, user control, and deletion of their lesion records.

Therefore, the resulting safety and ethical requirements are as follows:

Table 4.3 Safety and Ethical Requirements

ID	Safety and Ethical Requirements	Explanation
----	---------------------------------	-------------

1	No diagnostic or risk output.	The description of lesions are strictly structured in the format of what is visually observable and are prohibited from diagnosis or recommendations.
2	Explicit disclaimers.	The non-diagnostic nature of the system is stated in the application header, on the results screen, and in the exported PDF report
3	Neutral framing of change.	Metric differences between lesions are presented as objective measurements in numbers without implying risks or clinical interpretations.
4	Information privacy.	The system requires no account, no login, and no personally identifying information. Images and records are held in client-side applications only and are not uploaded to a server or database.
5	User control and deletion.	The user may delete any lesion record and all of its observations at any time, with confirmation.

4.5 High-Level System Architecture

SkinSense app follows a two-tier client-service architecture. The client is a self-contained React single-page application responsible for image capture, lesion localization, questionnaire input, record management, visualisation, and report export. All application state, including images and lesion history, is held in the client. The service tier is a stateless FastAPI application hosting the two vision models (segmentation and description) on a GPU runtime. It exposes a single analysis endpoint and holds no state between requests.

The two tiers are coupled only through a single HTTP interface and its response schema, detailed in **Section 5.2**. This narrow contract is the central architectural property of the system: the client is agnostic to which models run behind the interface, and the service has no knowledge of how results are stored, compared, or rendered. Either tier can therefore be replaced independently. The inference backend could be migrated from a notebook runtime to a hosted GPU service, or the models themselves substituted, without any change to the client.

Table 4.4 Architectural layers and responsibilities

Layer	Component	Responsibility
Client	React / Vite SPA	Capture, localisation, questionnaire, state management, trends, PDF export.
Transport	HTTPS via ngrok tunnel	Public, CORS-enabled access to the GPU runtime from the browser.

Service	FastAPI application	Request handling, image decoding, orchestration of the inference pipeline.
Inference	SAM (ViT-B)	Lesion boundary segmentation from an automated prompt.
Inference	OpenCV	Deterministic computation of the four morphological metrics from the mask.
Inference	MedGemma 4B	Constrained generation of one objective descriptive sentence.

4.6 End-to-End Workflow

Once a photo of lesion is uploaded, it goes through four user-facing stages. The fifth step is performed if required by the user. The workflow is linear but permits backward navigation: a step indicator allows the user to return directly to any completed stage, while forward navigation is blocked because later stages depend on data produced earlier.

Table 4.5 End-to-end workflow of each lesion photo uploaded

Stage	User Action	System Response
1. Capture	Uploads the lesion photo.	Runs an image quality check (skin presence, focus, lighting, framing) and blocks progression until it passes.
2. Locate	Places a pin on the front or back body diagram.	Compares the pin against existing lesion pins; if one lies within the proximity threshold, it asks the user to confirm whether it is the same lesion.
3. Context	Answers five structured questions.	Validates completeness and stores the responses with the pending observation.
4. Analyse	Confirms submission.	Posts the image to the backend; displays a loading state while SAM, OpenCV and MedGemma execute; renders the contour overlay, description and four metrics.
5. Record	Saves the observation.	Appends the observation to the matched lesion, or creates a new lesion record, and returns to the archive.

From the archive, the user may open any lesion to view its full timeline and metric trends across observations, compare any two observations by metric, or export the complete record as a PDF report. Trend visualisation requires at least two observations of the same lesion.

5. System Design

5.1 Frontend System Design

Client navigation uses a single view state variable rather than a routing library. Because the app has few screens and requires no deep linking, this setup corresponds to the requirement on information privacy that internal state remain externally unaddressable. We manage lesion state through a custom hook (`add`, `append`, `delete`) and keep draft observations separate so unsubmitted entries can be discarded without being appended to stored records.

The domain logic focuses primarily on two components. The body diagram uses an inline SVG with defined hit regions, so that a tap is accepted only if it falls within the body region; the pin is stored as normalised coordinates plus a front/back view flag, making it resolution-independent. Lesion identity matching is performed by Euclidean distance between the new pin and existing pins on the same view, with a fixed proximity threshold. If a match is found, the user is asked to confirm rather than matching silently, since an incorrect automatic match would damage the records and comparisons. The contour overlay component renders the returned polygon in a normalised coordinate space over the source image, allowing the same component to be reused at thumbnail, detail, and report scales without recomputation.

5.2 Backend and AI Pipeline Design

The service exposes one endpoint, `POST /analyze`, which accepts a multipart request carrying the lesion image and the serialised symptom responses. The image is decoded into an RGB array and passed through a three-stage pipeline, segmentation, metric computation, and description, after which the endpoint returns a JSON document containing an acceptance flag and, when accepted, the four computed metrics, a normalised contour polygon, and the descriptive sentence. If segmentation fails to isolate a plausible lesion, the request is rejected with an explanatory message rather than returning degraded results. No request state is retained after the response is issued.

Segmentation uses the Segment Anything Model with the ViT-B checkpoint. Because the system must operate without manual annotation, the prompt is generated automatically as a central bounding box covering the middle region of the frame, which encodes the assumption that the user has centred the lesion as instructed during capture. SAM returns three candidate masks; rather than accepting the highest-confidence mask unconditionally, the pipeline filters candidates by area fraction and selects the highest-scoring mask whose area falls within a plausible range for a lesion. This filter was added after observing that the unconstrained maximum-confidence mask frequently selected the entire skin field rather than the lesion, which produced metrics that were internally consistent but meaningless.

Description is generated by MedGemma 4B used zero-shot. The computed metrics are passed to the prompt, and the model output is constrained to a single sentence covering border, shape and colour, explicitly prohibits diagnosis, risk assessment and advice, and required to be returned as a JSON object. Output parsing is set defensive: the response is first parsed as JSON, and if that fails, leading labels and preamble are stripped and the first sentence retained, with a final generic fallback if no usable text is recovered. This layered parsing was necessary because an unconstrained generative model produces

variable formatting across runs, which is acceptable in a conversational setting but not in a record intended for longitudinal comparison.

5.3 Engineering Analysis and Metrics

The four quantitative metrics are computed from the SAM mask using OpenCV contour analysis, and are independent of the language model. This separation is implemented because the descriptive sentence is generative and can be variable, whereas the metrics are fixed for a given image and mask, and the metrics are the ones used for comparison over time, instead of the descriptive texts. The largest external contour of the selected mask is extracted, and the following quantities are derived from it.

Table 5.1 Quantitative morphological metrics and interpretation

Metric	Definition	Interpretation
Border circularity	$4\pi A / P^2$, where A is contour area and P is perimeter.	Approaches 1 for a smooth circular border; decreases with irregular or indented boundaries.
Asymmetry	Eccentricity of the fitted ellipse, $\sqrt{1 - (b/a)^2}$.	Zero for a circular lesion; approaches 1 as the lesion becomes increasingly elongated.
Colour variation	Standard deviation of hue within the mask, normalised to [0, 1].	Low for uniformly pigmented lesions; high where multiple tones are present.
Axis ratio	Ratio of major to minor axis of the fitted ellipse.	Equals 1 for a circular lesion and increases with elongation.

The four metrics were chosen to map onto the ABCD criteria introduced by Friedman, Rigel and Kopf [4], which identify asymmetry, border irregularity, colour variegation and diameter as the clinically salient morphological features of a pigmented lesion: eccentricity operationalises A, circularity operationalises B, and hue variance operationalises C, while D is deliberately omitted because there is no reference object with known size in the image uploaded thus no physical scale. Axis ratio is reported as a secondary shape descriptor, though it is derived from the same fitted ellipse as eccentricity and is therefore not statistically independent of it.

All four metrics are dimensionless shape and colour descriptors that remain valid if there are changes in camera distance, which makes them comparable across photos taken in different weeks or months. The principal limitation of the pipeline is segmentation quality: SAM performs well on lesions with high contrast against surrounding skin, but boundary accuracy decreases on lesions with gradual colour transitions, and because all four metrics are derived from the mask, segmentation error affects every metric at the same time.

5.4 Frontend-Backend Integration

Integration is implemented by a single asynchronous client function, which converts the captured image from data URL representation to a binary blob before upload. The service enables permissive CORS headers so that the browser may call the tunnelled endpoint directly. Because inference takes approximately ten to thirty seconds on the target GPU, the client sets the analysis state to pending

before issuing the request and renders an explicit loading screen. This both communicates progress to the user and prevents the results view from attempting to render a record that does not yet exist, which was the cause of an early rendering failure during integration.

For failure handling, if there is a disconnection between the frontend and backend, an explicit message of connection failure will be presented, and the user will fail to proceed to the next step of creating a record.

6. Prototype Implementation

6.1 Frontend Prototype

The implemented frontend prototype is a React-based single-page application that supports the complete user-facing workflow for documenting and tracking skin lesions over time. The prototype transforms the system requirements described in Chapter 4 and the frontend design described in Chapter 5 into a functional browser interface that users can interact with.

The first implemented module is the lesion archive. This screen allows users to view previously recorded lesions as individual lesion cards. Each card summarizes the lesion location, observation history, latest record, and available metric information. From the archive, users can either open an existing lesion record or start a new record. This module supports the main purpose of the system: organizing lesion observations longitudinally rather than treating each uploaded image as an isolated event.

The second implemented module is the new record workflow. Users can upload a lesion image, after which the app guides them through the remaining documentation steps. The workflow includes a body location screen, where the user places a pin on a front or back body diagram, and a symptom questionnaire screen, where the user records contextual information such as duration, itchiness, bleeding, growth, and elevation. These modules allow the prototype to combine image-based information with patient-reported context.

The result screen displays the backend-generated analysis output. It includes an objective lesion description, a contour overlay on the uploaded image, and four quantified morphology metrics: border circularity, asymmetry, color variation, and axis ratio. The prototype also includes a lesion detail and timeline view. After a result is saved, the observation is added to the relevant lesion record. Users can review past observations, compare metric changes over time, and export a provider-ready report. The report is designed to summarize the lesion images, location, symptom responses, objective descriptions, and quantitative metrics in a format that can be shared with a dermatologist.

6.2 Backend Prototype

The backend prototype provides the computational analysis service for the system. It is implemented as a FastAPI service running in a GPU-supported Google Colab environment and exposed to the frontend through an ngrok HTTPS tunnel. This setup allows the browser-based frontend to access GPU-based image analysis without requiring the user's device to run large models locally.

The backend has realized four main functions. First, it receives the uploaded lesion image and questionnaire responses through a single /analyze API endpoint. Second, it prepares the uploaded image for analysis by decoding it into a usable image array and applying the required image-processing steps. Third, it performs lesion segmentation and quantitative metric computation. Fourth, the backend generates a concise, non-diagnostic lesion description using MedGemma, constrained to describe visible morphology rather than provide risk assessment or medical advice.

The backend returns its output as a structured JSON response. When analysis is successful, the response includes an acceptance flag, the contour polygon, the quantitative metrics, and the generated description. If the image cannot be processed reliably, the backend returns a rejection message instead of producing a potentially misleading record.

6.3 Integration Status

The current prototype connects the React frontend with the FastAPI backend through a defined HTTP interface. When the user submits a new lesion record, the frontend converts the uploaded image into a binary file object, attaches the questionnaire responses as serialized JSON, and sends both to the backend /analyze endpoint using a multipart request.

During backend processing, the frontend displays a loading state to prevent the user from moving forward before the analysis is complete. Once the backend returns a response, the frontend parses the JSON output and renders the returned description, metrics, and contour overlay on the result screen. If the request fails because the backend is unavailable or the connection is interrupted, the interface shows an error state rather than creating a record with incomplete or synthetic analysis.

The integration demonstrates the separation between the frontend record-management layer and the backend inference layer. The frontend handles user interaction, lesion organization, timeline comparison, and report export, while the backend handles image analysis and AI-generated description. This confirms that the prototype can support the intended end-to-end workflow while keeping the AI pipeline separate from the client-side application logic.

6.4 Prototype Demonstration Scenario

The prototype can be demonstrated through a realistic lesion documentation scenario. A user begins on the archive screen, where existing lesion records are shown. The user selects the option to start a new record and uploads or captures an image of a skin lesion. The system then guides the user to mark the lesion location on the body diagram and complete the short symptom questionnaire.

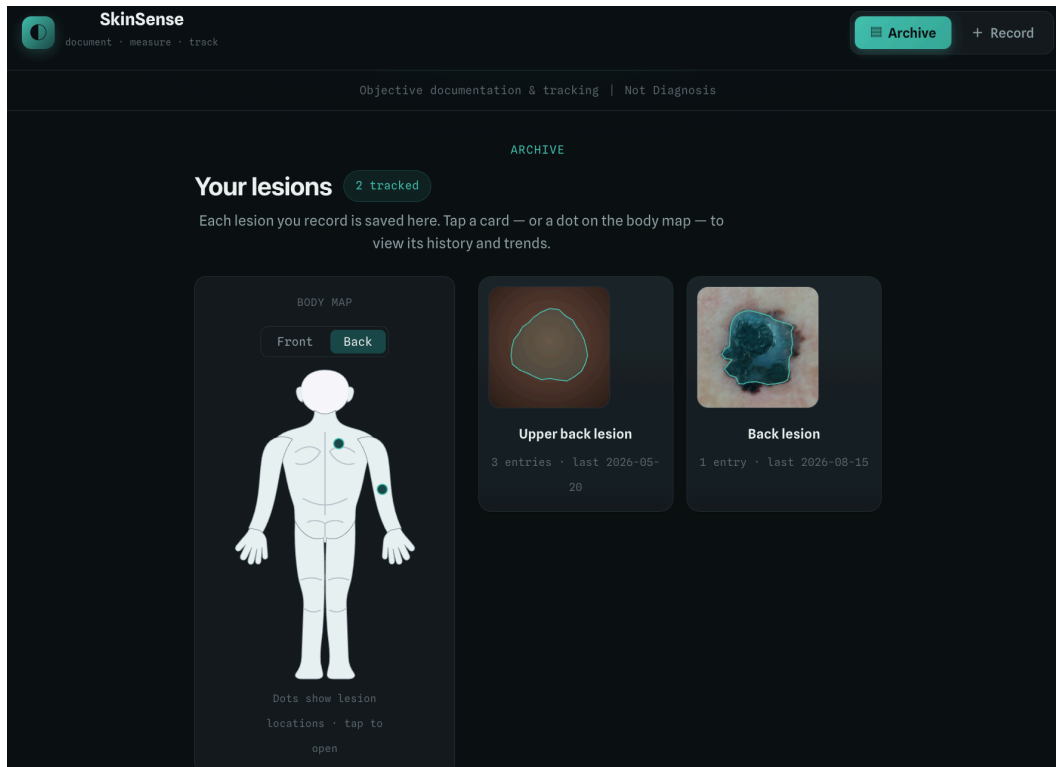


Figure 6.1 Archive Screen

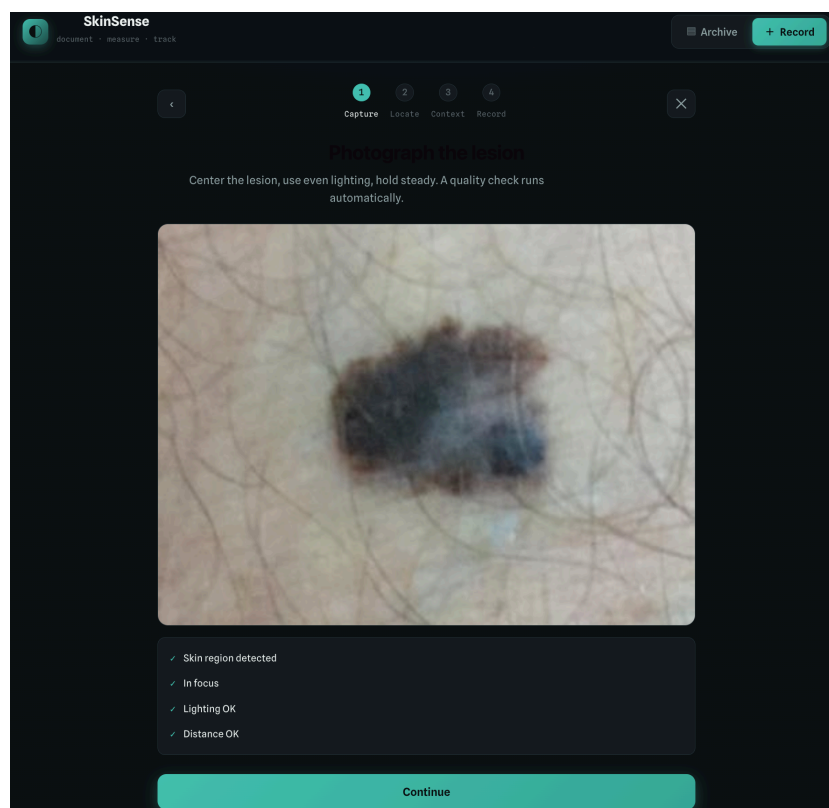


Figure 6.2 Image Upload Screen

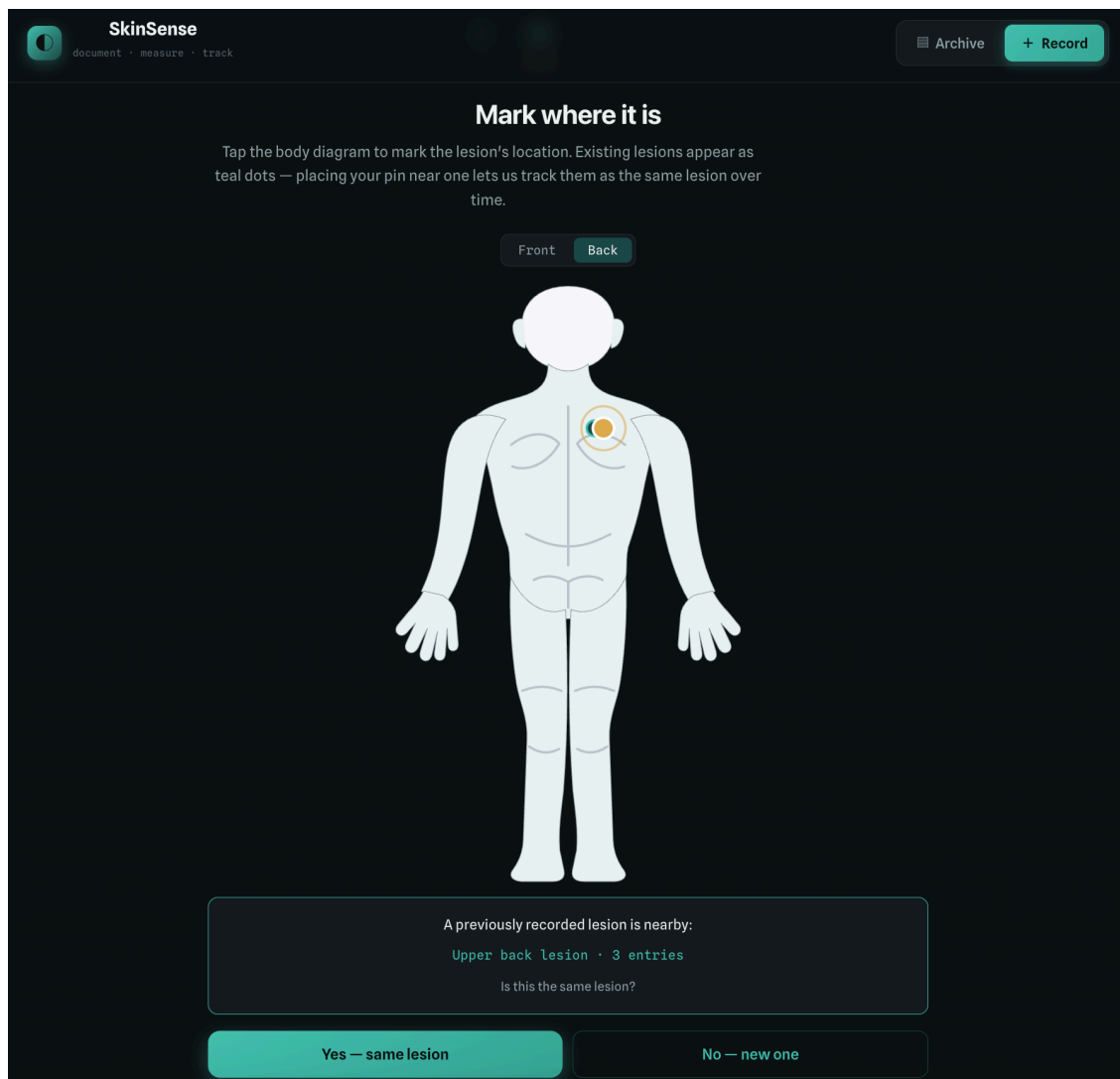


Figure 6.3 Body Location Mapping Screen

SkinSense
document · measure · track

Archive + Record

Objective documentation & tracking | Not Diagnosis

Capture Locate **Context** Record

A few questions

Recorded with the image as part of the lesion's history — not used to diagnose.

How long present? < 1 month

Itchy? Yes No

Bleeding? Yes No

Recently changed in size? Yes No

Raised above the skin? Yes No

Analyze

Figure 6.4 Symptom Questionnaire Screen

After the user submits the record for analysis, the frontend sends the image and questionnaire responses to the backend. The backend processes the image, segments the lesion, computes the morphology metrics, and generates an objective descriptive sentence. The frontend then displays the result page, including the lesion image with contour overlay, the four quantitative metrics, and the generated description.

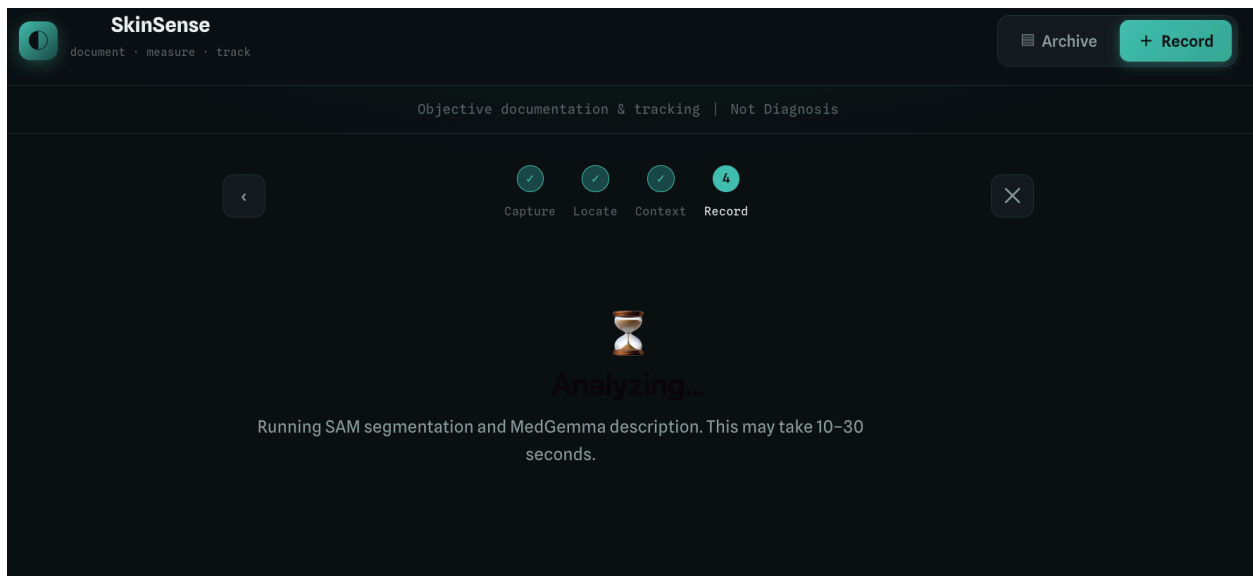


Figure 6.5 Analysis Loading State

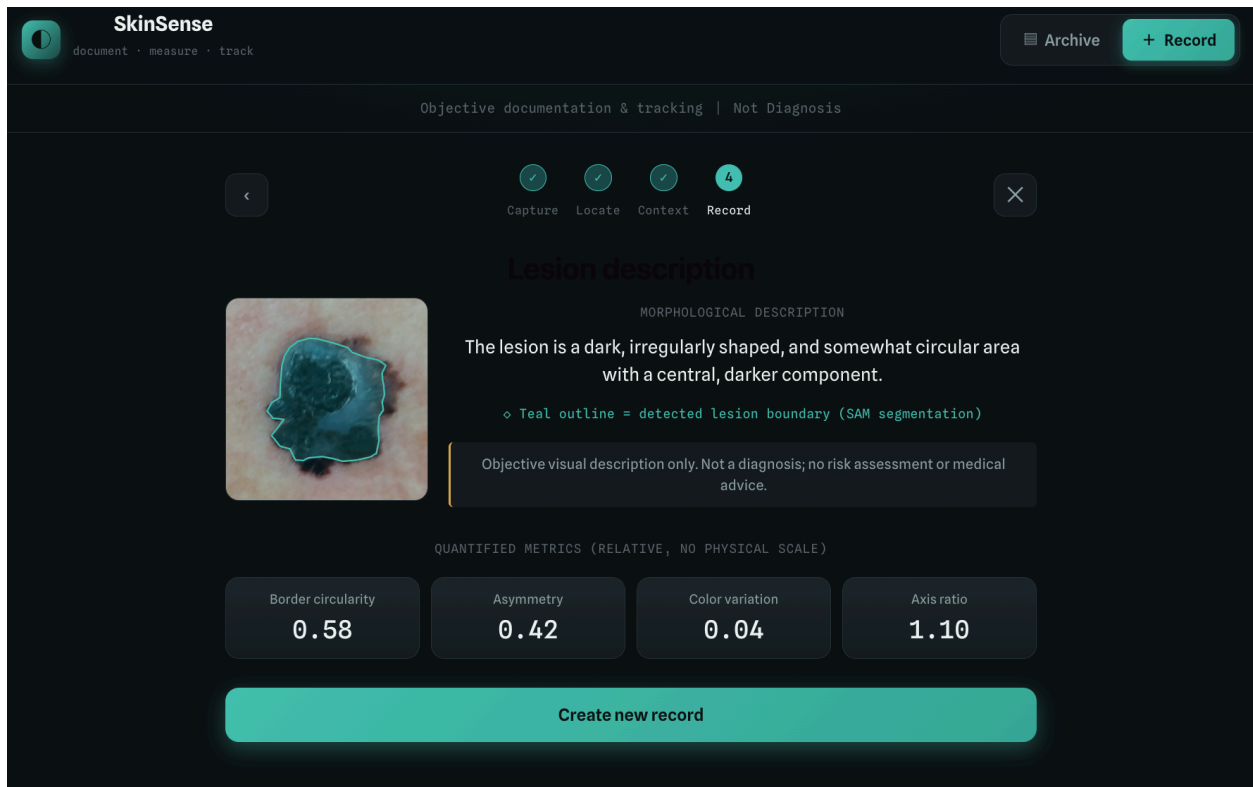


Figure 6.6 Result Screen

The user can save the result to the archive, where it becomes part of the lesion's observation history. If the same lesion has multiple observations, the user can open the lesion detail view to compare records over time and review metric changes. Finally, the user can export a provider-ready report containing the lesion images, body location, symptom responses, objective descriptions, and quantitative metrics for discussion with a dermatologist.

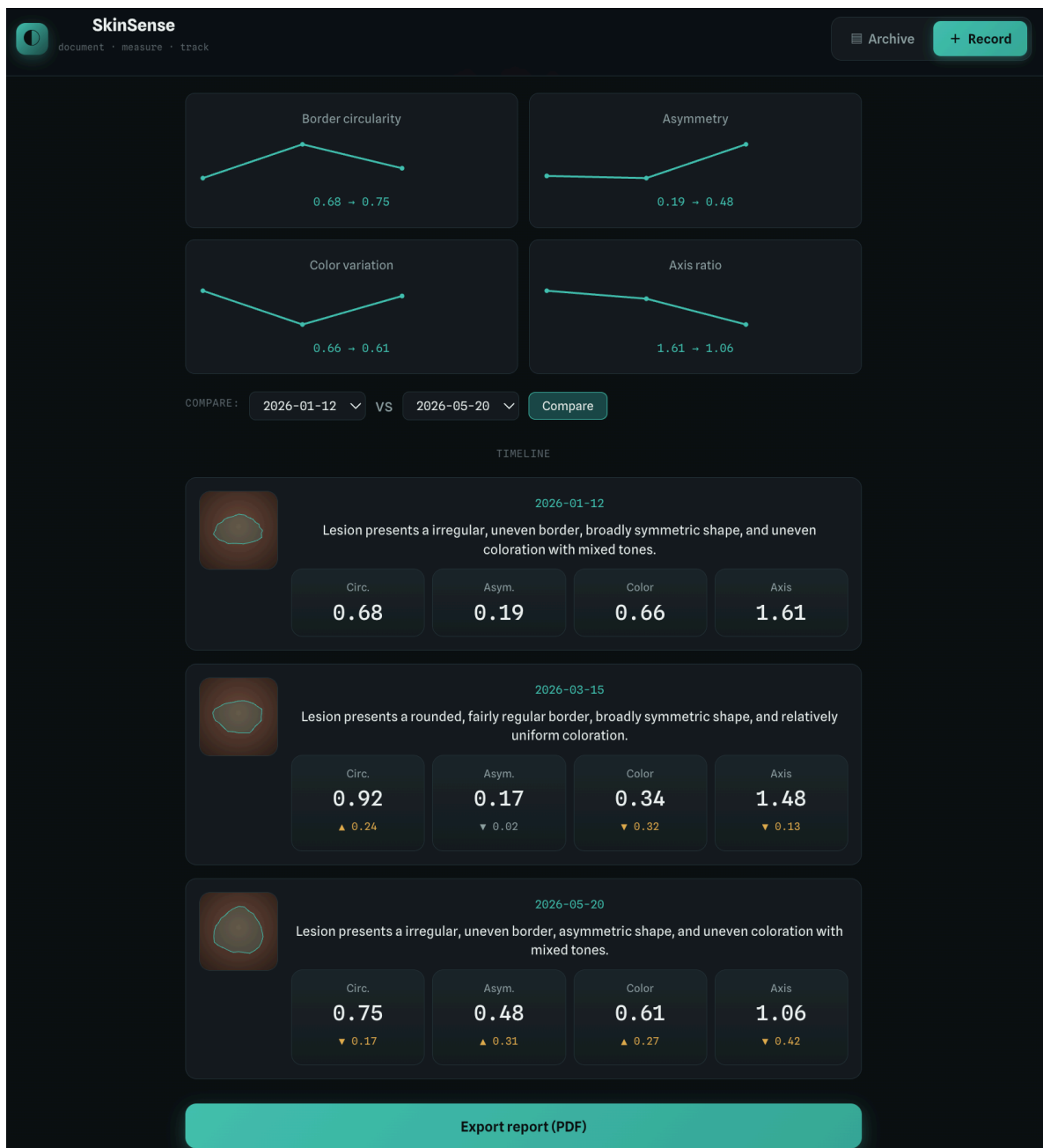


Figure 6.7 Lesion Timeline Comparison

SkinSense | Lesion Tracking Report

Upper back lesion · 3 entries · generated 2026-08-03

Teal outline = detected lesion boundary (SAM segmentation). Metrics are relative (no physical scale).

Image	Date	Circularity	Asymmetry	Color var.	Axis ratio	Description
	2026-01-12	0.68	0.19	0.66	1.61	Lesion presents a irregular, uneven border, broadly symmetric shape, and uneven coloration with mixed tones.
	2026-03-15	0.92	0.17	0.34	1.48	Lesion presents a rounded, fairly regular border, broadly symmetric shape, and relatively uniform coloration.
	2026-05-20	0.75	0.48	0.61	1.06	Lesion presents a irregular, uneven border, asymmetric shape, and uneven coloration with mixed tones.

This report contains objective morphological measurements and descriptions only. It is not a diagnosis and contains no risk assessment or medical advice. For clinical evaluation, consult a dermatologist.

Figure 6.8 Provider-Ready Report

7. Testing and Validation

7.1 Technical Verification

Technical verification was conducted to confirm the prototype could perform the intended end-to-end workflow described in Chapter 6. The purpose of this verification was to check whether each implemented module operated correctly as a functional prototype and whether data could be passed reliably between the frontend and backend.

The frontend was first verified by running through the full user workflow in the browser. This included opening the lesion archive, starting a new record, uploading a lesion image, selecting the lesion location on the body diagram, completing the symptom questionnaire, submitting the image for analysis, reviewing the result page, saving the record, reopening the lesion detail page, comparing observations over time, and exporting a provider-ready report. These checks confirmed that the main frontend screens were connected in the intended sequence.

The backend was verified by testing the /analyze endpoint with lesion images submitted from the frontend. The request was checked to ensure that the uploaded image and questionnaire responses were successfully received by the FastAPI backend. The backend pipeline was then observed to confirm that the image was decoded, passed through the segmentation and metric computation steps, and returned as a structured JSON response. The expected backend output included an acceptance flag, a contour polygon, quantitative morphology metrics, and an objective descriptive sentence.

Integration verification focused on whether the frontend could correctly use the backend response. After analysis was completed, the frontend was expected to display the returned contour over the original lesion image, show the four quantitative metrics, and present the generated description on the result

screen. The loading state was also checked to ensure that users received feedback while the backend was processing the image. In cases where the backend connection failed or the analysis response was unavailable, the frontend displayed an error state rather than creating an incomplete or synthetic record.

Overall, the technical verification confirmed that the prototype supports the primary documentation workflow: image submission, backend analysis, result display, record saving, timeline review, and report export. The verification also identified segmentation quality as a key dependency for the reliability of downstream metrics, since the contour produced by SAM directly affects the shape and color measurements shown to the user.

7.2 Usability Testing

Usability testing was designed to evaluate whether potential users could understand the system's workflow and whether the prototype supported their needs for documenting skin lesions over time. Since the system is intended for general users, the usability testing focused on clarity, ease of use, perceived usefulness, and user confidence in completing the record-creation process.

The testing procedure was structured around a realistic use scenario. Participants were asked to use the prototype as if they wanted to document a skin lesion for future comparison or for discussion with a dermatologist. The task sequence included viewing the archive, starting a new record, uploading or capturing an image, marking the lesion location on the body diagram, answering the symptom questionnaire, reviewing the generated result, saving the record, checking the timeline view, and exporting a report.

During the usability session, participants' interactions were observed to identify points of confusion or hesitation. Particular attention was paid to whether users understood the purpose of each screen, whether the step-by-step workflow felt logical, and whether the result page was easy to interpret.

The usability feedback was designed to be primarily open-ended. Rather than asking users to rate every interface element in detail, the testing emphasized questions such as what parts of the app they found most useful, how they would use the app in a real situation, what information they would expect to see before sharing a report with a healthcare provider, and what parts of the workflow felt unclear or unnecessary. This approach allowed participants to comment on the prototype from their own perspective rather than being restricted to predefined evaluation categories.

The expected outcome of usability testing is to determine whether the prototype communicates its non-diagnostic purpose clearly and whether users can complete the documentation workflow with minimal confusion. Since the system is designed to support long-term tracking, usability testing also helps evaluate whether users understand the value of saving observations and comparing lesion records over time. These findings will guide later revisions to interface wording, workflow order, result presentation, and report formatting.

7.3 Stakeholder Feedback

(1) Dermatologist

Dermatologist feedback generally supported the positioning of SkinSense as a monitoring and communication tool rather than a self-diagnosis tool (The interview guide used for dermatologist

stakeholder validation is provided in [Appendix B](#)). From the dermatologist's perspective, patients would likely use the system to select the affected body area, upload or capture a clear image, answer a short symptom questionnaire, and save repeated records over time. The main value of the system is therefore not to identify a disease, but to help patients organize visual and contextual information before speaking with a healthcare professional.

Several existing features were identified as useful for both patients and clinicians. For patients, body-location selection, guided image capture, the symptom questionnaire, and timeline comparison could help them document information that is often difficult to remember during an appointment. For healthcare professionals, the provider-ready PDF report was considered particularly helpful because it could combine lesion images, affected body location, symptom history, previous treatments, and changes over time in one document. The contour, size-related measurements, and colour comparison may also provide more objective documentation than subjective descriptions such as “better” or “worse.”

The feedback also suggested that the questionnaire could be expanded to capture more clinically relevant context. In addition to duration, itching, bleeding, growth, and elevation, future versions could ask about allergies, new medications, new skincare products, recent travel, exposure history, contact with others who have similar symptoms, and previous or current treatments. The dermatologist also noted that reminders for follow-up photos could help users capture repeat images under similar lighting, distance, and framing conditions, improving the reliability of timeline comparison. Clear privacy controls were also emphasized, including how photos and health information are stored, who can access them, and whether users can delete or export their records.

The main safety concern was that users might interpret AI-generated descriptions, contours, or numerical measurements as diagnostic conclusions. The dermatologist recommended that the system avoid definitive language and instead use cautious, descriptive wording. The feedback also emphasized that the numerical measurements should not be presented as highly precise clinical measurements, since their accuracy can be affected by lighting, camera angle, skin tone, image quality, and segmentation performance. Therefore, the outputs should be presented as supporting documentation rather than clinical evidence.

Overall, the dermatologist feedback reinforced the project's non-diagnostic scope. SkinSense should help patients prepare clearer information for healthcare providers, but it should not delay care or replace professional assessment. This feedback supports continued emphasis on neutral language, image-quality guidance, contextual symptom collection, privacy transparency, and clear instructions to seek medical attention when concerning or urgent signs are present.

(2) General user feedback

General user feedback was collected using a short validation questionnaire focused on usability, perceived usefulness, trust, and safety (The general user validation questionnaire is provided in [Appendix C](#)). The questionnaire asked participants about prior experience tracking skin lesions, perceived difficulty of the SkinSense workflow, the most useful and confusing features, comfort using the app before seeing a healthcare provider, and open-ended feedback on how they might use the system.

Based on responses from eight general-user participants, the workflow was generally perceived as understandable (The raw questionnaire responses are provided in [Appendix D](#)). The average workflow ease rating was 4.1 out of 5, suggesting that the step-by-step structure of uploading an image, marking body location, answering symptom questions, reviewing results, and saving records was perceived as manageable. Participants also reported moderate to high comfort with using the app to organize lesion information before seeing a healthcare provider, with an average comfort rating of 3.9 out of 5.

The most valued features were timeline comparison, provider-ready report export, and guided image capture. Participants indicated that these features could help them remember when changes occurred, organize lesion photos more consistently, and communicate more clearly with a healthcare professional. The report export feature was viewed as useful because it combines images, dates, body location, symptom responses, and measured changes into one document.

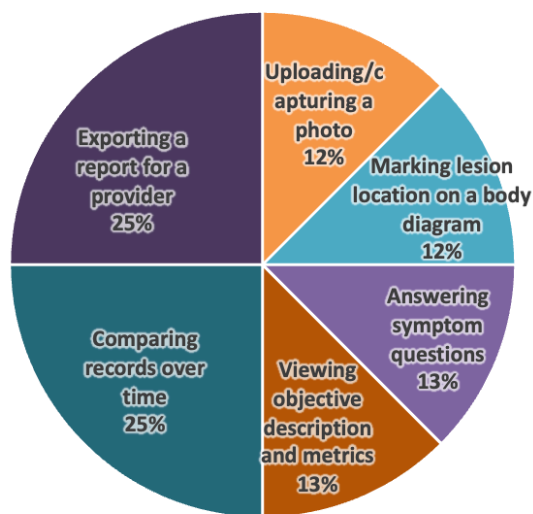


Figure 7.1 User Feedback on Most Useful Features

The main area of confusion was the result page, especially the meaning of quantitative metrics such as circularity, asymmetry, colour variation, and axis ratio. Several participants indicated that they would need clearer explanations to understand what the metrics represent and how they should be interpreted. This feedback suggests that the interface should present metrics as descriptive documentation aids rather than clinical judgments.

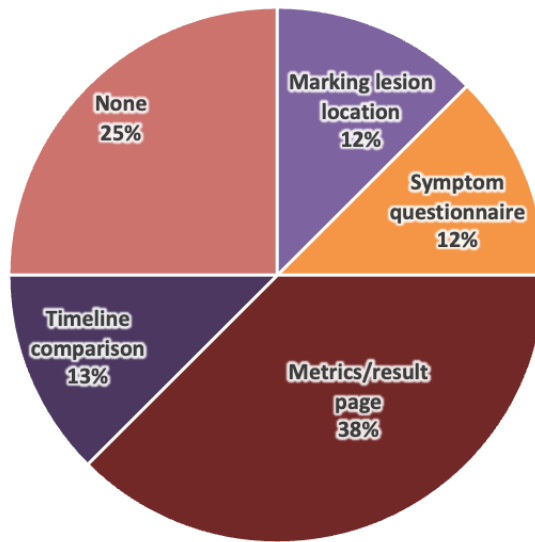


Figure 7.2 User Feedback on Most Confusing Features

Trust-related feedback focused on three areas: clearer metric explanations, better photo-taking guidance, and transparent privacy information. Participants wanted to know how their images and records would be stored, whether they could delete or edit records, and how to avoid poor-quality photos that might affect the results. These responses reinforce the importance of maintaining a non-diagnostic framing, providing plain-language explanations, and giving users control over their information.

Overall, general user feedback suggests that SkinSense is understandable and potentially useful as a documentation and communication tool. Users were most interested in features that help organize information over time and prepare for healthcare conversations. At the same time, the feedback shows that result interpretation, privacy communication, and photo guidance are critical design areas for improving user trust and reducing misunderstanding.

8. Recommendations and Future Work

8.1 Short-Term Improvements

The first short-term improvement should focus on making the quantitative metrics easier to understand. General-user feedback showed that the result page was the most confusing part of the system, especially the meaning of circularity, asymmetry, colour variation, and axis ratio. Future versions should therefore provide short plain-language explanations beside each metric. The values should continue to be presented as descriptive measurements rather than indicators of medical risk.

Image-capture guidance should also be improved. The reliability of longitudinal comparison depends on users taking photographs under reasonably consistent lighting, distance, angle, and framing. The interface could provide clearer instructions before capture and use previous lesion images as visual

references when users record the same lesion again. This would help reduce changes caused by photography conditions rather than actual changes in the lesion.

The questionnaire could also be expanded with optional information such as medications, skincare products, previous treatments, recent travel, and exposure history, as suggested by dermatologist feedback. In addition, privacy information should be more visible so users clearly understand how their images and records are stored, exported, and deleted.

8.2 Future Technical Development

The main technical priority is improving lesion segmentation. All four quantitative metrics are calculated from the contour generated by SAM, meaning that segmentation errors can directly affect the accuracy of all downstream measurements. Technical verification identified segmentation quality as one of the key dependencies of the system. Future work should therefore investigate more reliable lesion-specific segmentation methods and improved automatic prompting, particularly for lesions with low contrast or unclear boundaries.

The image-quality assessment could also be further developed to evaluate factors such as blur, exposure, framing, and lesion positioning more consistently. These checks should eventually be validated to determine whether they improve the repeatability of measurements across images taken at different times.

The current FastAPI backend running through Google Colab and ngrok is appropriate for a functional prototype, but a future deployed system would require a more stable and secure GPU-hosted environment. Future development could also explore standardized image scaling or reference markers to allow physical lesion dimensions to be measured more consistently.

8.3 Future Stakeholder Validation

The current stakeholder validation provides useful initial feedback, but a larger and more diverse sample would be required before real-world use. Future testing should involve more general users with different levels of technical experience, as well as additional dermatologists and family physicians.

Healthcare professionals could evaluate whether the provider-ready report contains useful and appropriately presented information for clinical conversations. Technical validation should also compare segmentation results against expert or manually defined lesion boundaries and evaluate how stable the quantitative metrics remain under changes in lighting, angle, and camera distance.

Future testing should also examine whether users correctly understand that SkinSense is a documentation tool rather than a diagnostic system. This is especially important because dermatologist feedback identified the possibility that users may interpret AI-generated descriptions, contours, or numerical values as medical conclusions.

9. Conclusion

SkinSense was developed as a web-based, AI-assisted system for improving the documentation and longitudinal tracking of skin lesions. The system addresses problems associated with inconsistent photographs, fragmented lesion histories, and missing contextual information by combining guided image capture, body-location recording, symptom documentation, lesion segmentation, quantitative morphology measurements, timeline comparison, and provider-ready report export.

The developed prototype demonstrates a complete end-to-end workflow from image submission to report generation. Technical verification confirmed that the frontend and backend can successfully support image analysis, result display, record saving, timeline review, and report export.

Initial stakeholder feedback also supports the usefulness of the proposed approach. General users rated the workflow ease at an average of 4.1 out of 5 and particularly valued timeline comparison, report export, and guided image capture. However, the feedback also identified important areas for improvement, including clearer explanations of quantitative metrics, stronger photo-taking guidance, privacy transparency, and improved segmentation reliability.

Overall, SkinSense demonstrates the feasibility of using AI-assisted image analysis to support more consistent lesion documentation and communication with healthcare professionals. However, the system remains a non-diagnostic functional prototype and requires further technical improvement, broader stakeholder evaluation, and clinical validation before it could be considered for real-world use.

References

- [1] Ontario Health (Cancer Care Ontario), Ontario Cancer Statistics 2022. Toronto, ON, Canada: Ontario Health (Cancer Care Ontario), 2022. [Online]. Available: <https://www.cancercareontario.ca/en/data-research/view-data/statistical-reports/ontario-cancer-statistics-2022/ch-4-cancers-incidence-2022>. Accessed: Aug. 10, 2026.
- [2] C. Garbe et al., “European consensus-based interdisciplinary guideline for melanoma. Part 1: Diagnostics - Update 2024,” *European Journal of Cancer*, vol. 215, p. 115152, Jan. 2025, doi: 10.1016/j.ejca.2024.115152.
- [3] S. Berglund, J. Paoli, P. Svensson, K. Terstappen, M. Gillstedt, and J. Dahlén Gyllencreutz, “Patients’ Ability to Take Dermoscopic Follow-Up Images of Atypical Melanocytic Lesions With Smartphones: A Pilot Study,” *Dermatology Practical & Conceptual*, vol. 14, no. 4, p. e2024268, Oct. 2024, doi: 10.5826/dpc.1404a268.
- [4] Abbasi NR, Shaw HM, Rigel DS, et al. Early Diagnosis of Cutaneous Melanoma: Revisiting the ABCD Criteria. *JAMA*. 2004;292(22):2771–2776. doi:10.1001/jama.292.22.2771

Appendices

Appendix A: Detailed Stakeholder Analysis

1. Stakeholder Groups

The stakeholders for SkinSense include direct users, healthcare professionals, and supporting users who may help with lesion documentation. General users and higher-risk users are the primary stakeholders because they directly interact with the application to document and monitor lesions over time. Dermatologists and family physicians are secondary stakeholders because they may review the exported records as supporting information during clinical assessment. Caregivers and family members are also relevant stakeholders because they may assist users who cannot easily monitor or photograph lesions independently.

2. Stakeholder Needs and Design Implications

Stakeholder Group	Role in the System	Key Needs	Concerns / Constraints	Design Implications
General users/patients	Main users who create lesion records and track changes over time.	Simple way to capture lesion photos, record symptoms, organize observations, and understand whether visible changes have occurred.	Limited medical knowledge; possible misunderstanding of AI outputs as diagnosis; inconsistent photo capture.	Use a guided step-by-step workflow, plain-language wording, image-quality guidance, and clear non-diagnostic disclaimers.
Higher-risk users	Users with many moles, frequent sun exposure, family history of skin cancer, or previous suspicious lesions.	Consistent long-term tracking and easier comparison of repeated lesion observations.	Need to monitor multiple lesions; potential anxiety if outputs appear too diagnostic.	Support lesion archive organization, body-location labeling, timeline comparison, and neutral trend visualization.
Dermatologists	Healthcare professionals who may review exported lesion histories during assessment.	Clear, objective, and organized supporting information, including dated images, body location, symptoms, prior treatments, and visible changes over time.	Outputs must not replace clinical examination, dermoscopy, or histopathological confirmation. Inaccurate contours or metrics may reduce trust.	Present images, symptoms, contours, metrics, and trends as supporting documentation only, using cautious non-diagnostic language.

Family physicians / primary care providers	May be the first healthcare professionals users consult before dermatologist referral.	Concise information that helps them understand symptom history and visible lesion changes.	Clinical judgment remains with the physician; app outputs cannot be used as a diagnosis.	Make the provider-ready report concise, structured, and easy to review during an appointment.
Caregivers/family members	May help children, older adults, or users with limited mobility capture images and maintain records.	Easy operation on behalf of another person and support for consistent photo capture.	Need privacy awareness and user consent when handling personal health images.	Keep the workflow simple, support clear record organization, and make privacy and deletion controls visible.

3. Key Stakeholder Priorities

The stakeholder analysis showed that the system should prioritize documentation, organization, and communication rather than diagnosis. General users need a guided and understandable workflow for recording lesions over time, while healthcare professionals need objective supporting information that is clearly separated from clinical interpretation. Higher-risk users and caregivers may benefit from structured archives, body-location labeling, and timeline comparison, but the system must avoid creating false reassurance or unnecessary anxiety.

Across stakeholder groups, the most important design priorities are consistent image capture, structured symptom collection, longitudinal record organization, provider-ready reporting, privacy transparency, and clear non-diagnostic framing. These priorities directly informed the system's functional requirements, technical requirements, safety requirements, and validation plan.

Appendix B: Dermatologist Stakeholder Validation Interview Guide

1. Purpose of Interview

The purpose of this interview is to collect dermatologist feedback on the SkinSense prototype, particularly whether the application's workflow, outputs, questionnaire, metrics, and report format are useful, understandable, and safely framed. The interview focuses on validating SkinSense as a non-diagnostic documentation and communication tool, not as a diagnostic or risk-prediction system.

2. Interview Format

This is a semi-structured interview. The interviewer will ask four open-ended questions and may ask follow-up questions for clarification. The participant may comment on any part of the prototype, including the user workflow, image capture process, symptom questionnaire, result page, timeline comparison, and provider-ready report.

3. Opening Script

SkinSense is a prototype web application designed to help users document and track skin lesions over time. It allows users to upload or capture lesion images, mark body location, answer symptom-related questions, view objective descriptions and visual metrics, compare records over time, and export a provider-ready report. The system is not intended to diagnose, screen for, confirm, or exclude skin cancer. We would like your feedback on whether the prototype outputs and workflow are useful, understandable, and appropriately bounded from a clinical and safety perspective.

4. Interview Questions

(1) How do you think patients would use SkinSense?

Possible follow-up questions:

- Would this be useful for tracking changes between visits?
- Are there specific patient groups or situations where this tool may be more useful?

(2) Which parts of the application would be most helpful to patients or healthcare professionals?

Possible follow-up questions:

- What information would be most helpful for a clinician reviewing the exported report?

(3) What important information or functionality, if any, is missing?

Possible follow-up questions:

- Are there additional symptom questions, risk factors, exposure history, treatment history, or privacy controls that should be included?
- Is there anything that would make the report more useful for clinical communication?

(4) What parts, if any, could be confusing, misleading, or create safety concerns?

Possible follow-up questions:

- Could users misunderstand the AI-generated description or numerical metrics?
- Is the non-diagnostic boundary clear enough?

5. Closing Question

Is there any additional feedback you would like to provide about how SkinSense could better support patient documentation and communication with healthcare professionals?

6. Use of Feedback

Feedback from this interview will be used to evaluate the prototype's usefulness, safety framing, and communication design. It will also inform future revisions to the questionnaire, result presentation, report format, and non-diagnostic wording.

Appendix C: User Validation Questionnaire

Intro Statement

SkinSense is a prototype app designed to help users document and track skin lesions over time. It provides objective descriptions, simple visual metrics, timeline comparison, and a report that users may share with a healthcare provider. It is not intended to diagnose skin conditions, predict cancer risk, or provide medical advice.

Section 1: Background

1. Have you ever taken photos of a skin spot, mole, or lesion to monitor it over time?
 - Yes / No / Not sure
2. If yes, what was difficult about tracking it?
 - Remembering dates
 - Taking consistent photos
 - Describing changes
 - Knowing what information to record
 - Organizing photos
 - Other: _____

Section 2: Usability and Workflow

3. How easy does the overall workflow seem?
 - 1 = Very difficult
 - 5 = Very easy
4. Which step seems most useful?
 - Uploading/capturing a photo
 - Marking lesion location on a body diagram
 - Answering symptom questions
 - Viewing objective description and metrics
 - Comparing records over time
 - Exporting a report for a provider
5. Which step seems most confusing or unnecessary?
 - Uploading/capturing a photo
 - Marking lesion location
 - Symptom questionnaire
 - Metrics/result page
 - Timeline comparison
 - Report export
 - None
 - Other: _____

Section 3: Trust and Safety

6. How comfortable would you feel using this app to organize lesion information before seeing a healthcare provider?
 - 1 = Not comfortable

- 5 = Very comfortable
- 7. What would make you trust the app more?
 - Clearer disclaimer
 - More explanation of metrics
 - Ability to edit or remove inaccurate outputs
 - Provider review
 - Better photo guidance
 - Data privacy explanation
 - Other: _____

Section 4: Overall Feedback

- 8. What is one thing you would improve, if at all?

Appendix D: General User Validation Questionnaire Raw Results

User Validation Results							
ID	Prior Photo Tracking	Main Difficulty	Workflow Ease /5	Most Useful Step	Most Confusing Step	Comfort /5	Improvement Suggestions
P1	Yes	Taking consistent photos	4	Comparing records over time	Metrics/result page	4	More explanation of metrics
P2	No	Knowing what information to record	5	Exporting a report for a provider	None	4	Data privacy explanation
P3	Yes	Remembering dates	4	Timeline comparison	Timeline comparison	3	Better photo guidance
P4	Not sure	Describing changes	3	Objective description and metrics	Metrics/result page	3	Provider review
P5	Yes	Organizing photos	4	Report export	None	5	Ability to edit/remove outputs
P6	No	Knowing what information to record	4	Symptom questions	Body location marking	4	Clearer disclaimer
P7	Yes	Taking consistent photos	5	Guided photo capture	Metrics/result page	4	Better photo guidance
P8	No	Describing changes	4	Body diagram location	Symptom questionnaire	4	Data privacy explanation