Design Review

Team A1: **Isaiah Weekes**, Siena Lee, Corin Cho

Use Case + Use Case Requirements

Problem

Managing changing skin conditions can be challenging, and existing solutions are often costly, difficult to access, or lack personalization.

Application

A smart mirror that provides personalized skin analysis for 4 conditions: acne, oiliness, sunburns, and wrinkles
Provides skin care recommendations based on the analysis

Use-Case Requirements

Accurate Skin Detection - Want a minimum of 85% accuracy per class

Image Capture + CV Inference Latency - from the moment of button pressed for analysis, LCD display should display information within 7 sec

Consistent feedback + Easy User Experience - each user session must provide accurate analysis with appropriate product recommendations.
** User study will be conducted

Quantitative Design Requirements

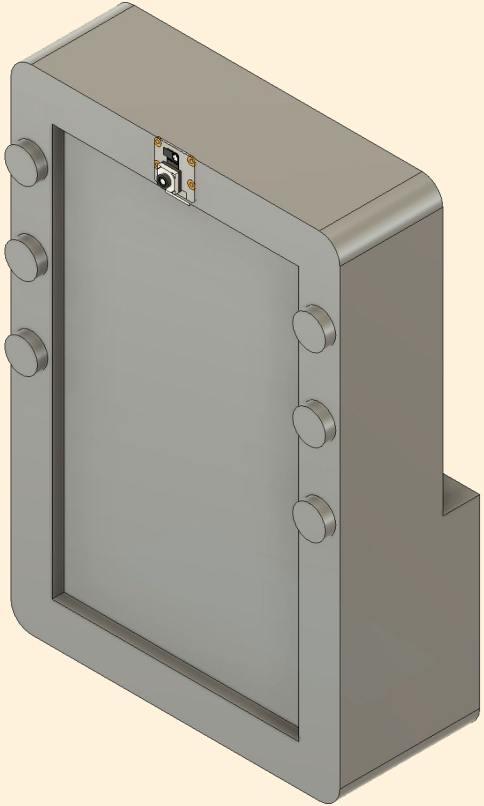
Use Case	Design Requirements
Need accurate detection of the skin for the classifications of 4 conditions	Minimum resolution of camera: 12MP → ensure detailed skin imaging Classification accuracy: >=85% on the 4 skin conditions → Model capacity up to 60 million parameters to achieve target accuracy.
Low Latency User Experience	Inference time less than 4 sec per image on RPi Communication from camera to RPI and RPI to screen display → less than 1 sec End-To-End Latency Goal (capture, analyze, display): ≤ 5sec
Consistent Feedback + Easy User Experience	On-system app (w/ tools like SQLite and Tkinter) → display current session results + past user session results Add LED strips around camera for consistent lighting Make local database of skincare products mapped to 4 classes + confidence levels

Solution Approach

Health & Wellness
Deliver accessible skin condition analysis with personalized skincare recommendations, allowing users to discover best suited products

Safety
Image data processing is done locally on RPi, minimizing data transmission and focusing on protecting user privacy

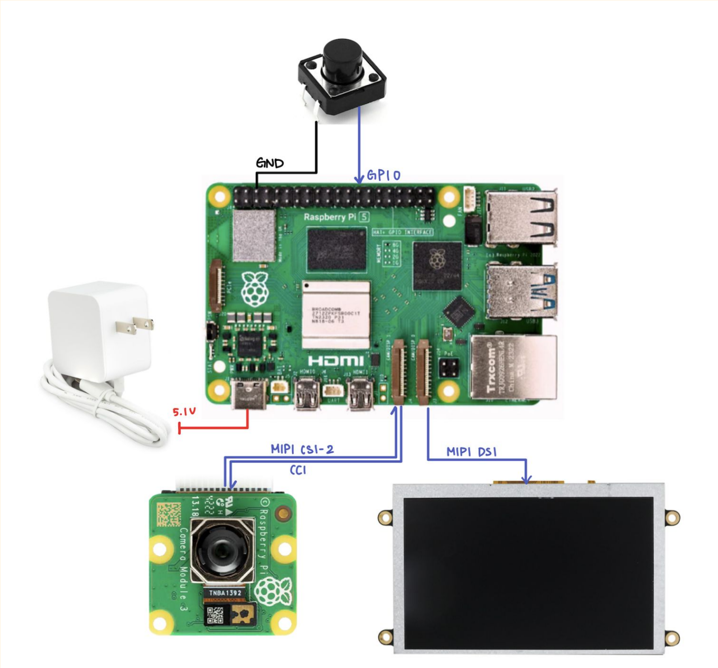
Economic
Using affordable components (RPI and open source ML models) helps keep our product cost effective

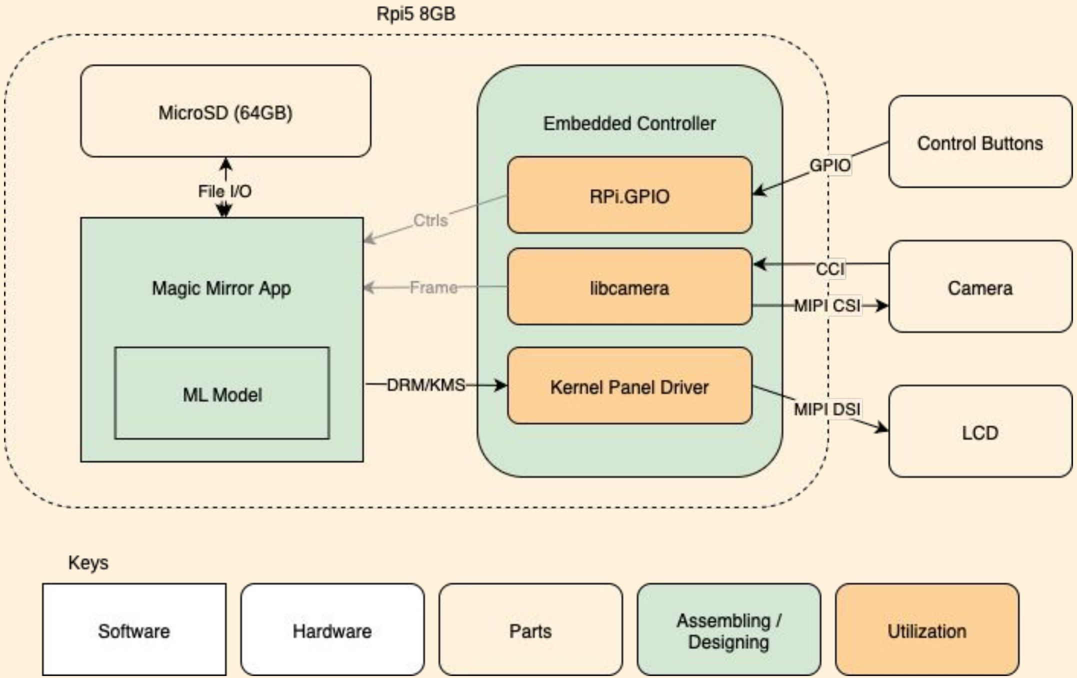


Our Approach

- Build a standing smart mirror (12 inch * 8 inch) for daily skin analysis
- Lower budget solution with only one RPI board and a cost efficient camera and screen
- Keep all of our user data locally on the RPi board so that user privacy is protected
- Have a local database on skincare (w extensive research and expert feedback) for accurate recommendations based on skin condition classification + confidence level

System Specification: Hardware





System Specification: Software

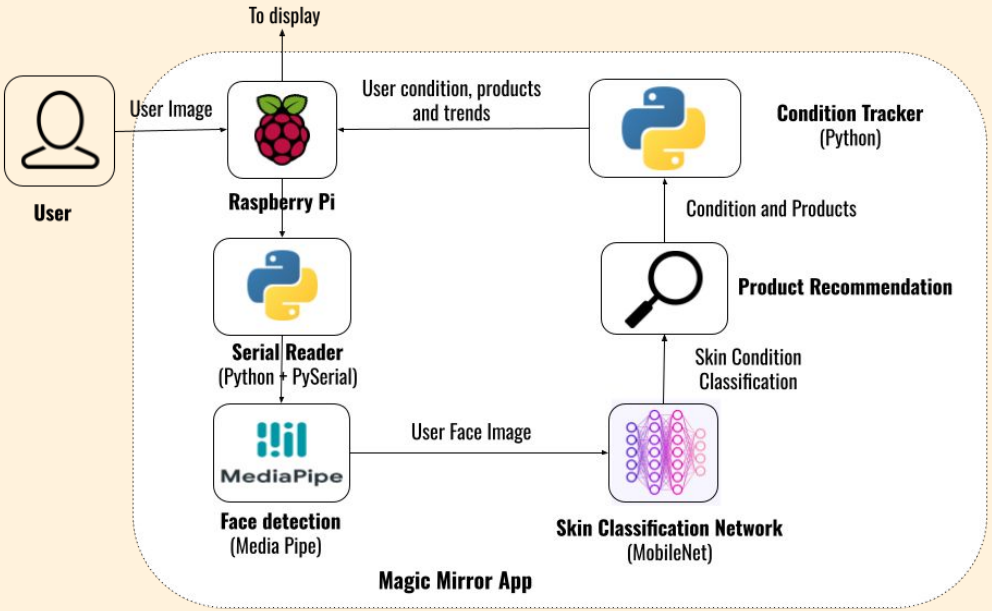
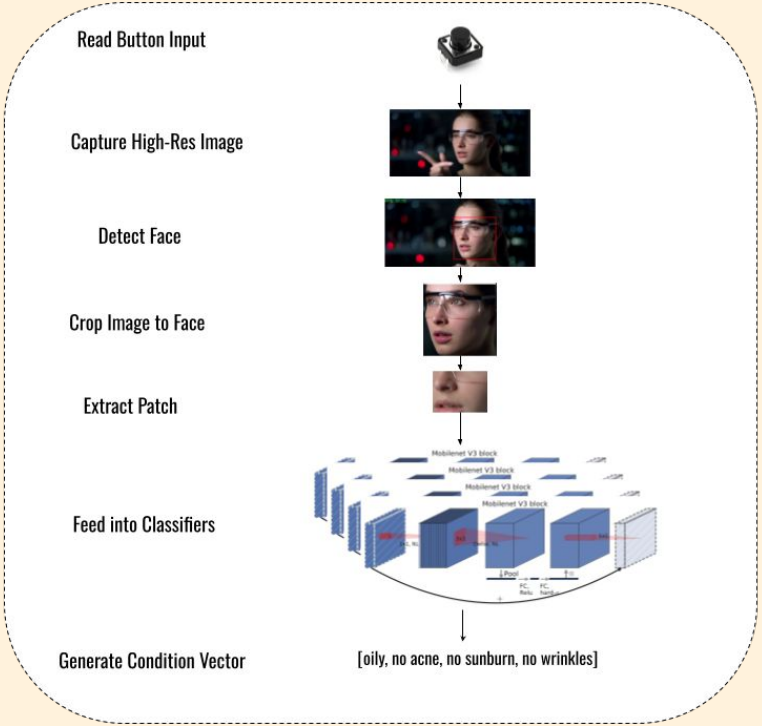


Image Processing Pipeline

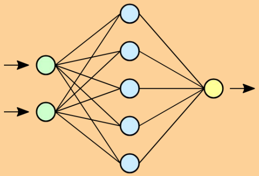


Implementation Plan - Software

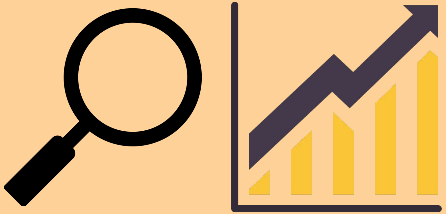
Downloaded

Downloaded + Data Augmentation

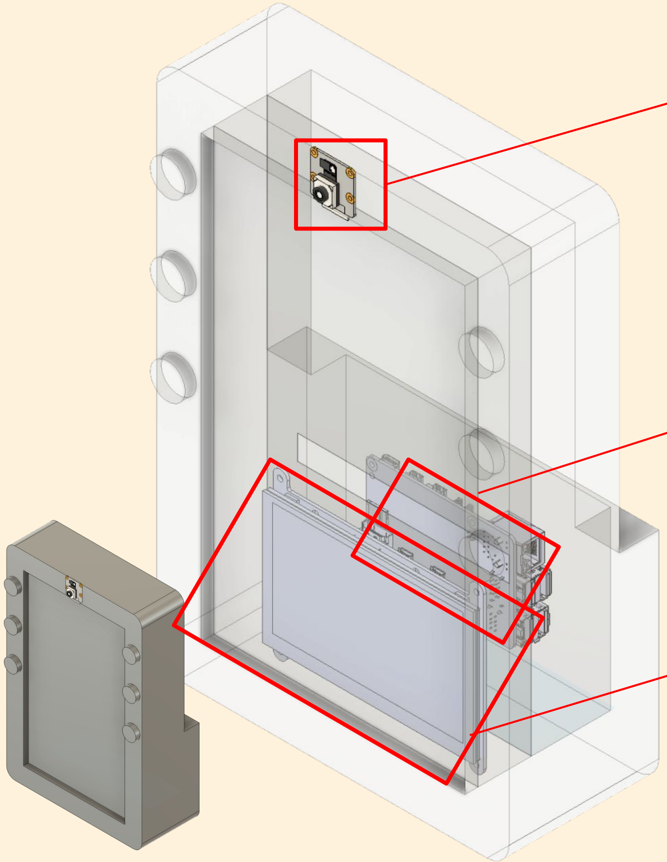
Pre-training + Fine Tuning



Custom-Built App



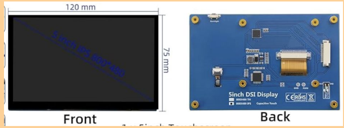
Implementation Plan - Hardware



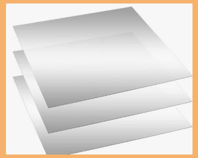
Camera
Arducam IMX519
(16MP)
Connector type:
MIPI CSI camera connector via FPC cable

SBC Board: RPi 5 8GB
Attached right behind the screen for secure connection.
Located inside the stand that supports the mirror.

Screen Display
5 Inch Raspberry Pi Touchscreen
Connect via DSI port using FFC



Additional Components
Two Way Mirror Glass:
Attach the mirror film to the frame and use a cardboard cutout with a display window so the film's back is fully black except where the display shows
LED Light Strips:
Attach to the wooden frame along the top two-thirds of the mirror to provide lighting, controlled by an on/off switch.
Wooden Frame:
Laser-cut plywood



Test, Verification, and Validation

Test	Criteria
Input (Camera and Control Buttons) Accuracy	Of all commands inputted by the user, >= 95% of generated responses (image captured and button triggered events) shall be relevant
Face Detection and Feedback Accuracy	System can detect users and give correct positioning feedback >= 95% of usage
ML Model Classification Accuracy	85% accuracy for each condition in test datasets

Test	Criteria
System App	Responds successfully to inputs tested >=95% of usage Utilize memory and create trend graph after 3 days of usage, reflecting >=95% accuracy
Image Capture to LCD Latency	No input/output will take longer than 5 seconds (Failure control: ML accelerator)
LCD Accuracy	>=95% of the display shown shall reflect what is driven by the systems app (current analysis result or trend)

Test, Verification, and Validation

User Study

- Run 5 user tests for at least a week, more preferably with people with different skin conditions and have had previous consultation with a professional

- Gather the following data and feedback:
  - Camera to display latency
  - Analysis accuracy - compare with professional consultation
  - Rated UI/UX on a scale of 1 to 5
  - Trend mechanism reflects past data

Project Management

Team Member	Task
Siena	Device Integration ● Sub-device Communication (Camera and buttons)
Corin	Device Integration ● Primarily will be focusing on screen display / mirror film ● Will also be working with CAD model to plan out precise placements
Isaiah	Systems App ● Face detection ● ML model training
All	Mirror building + System App Integration

Schedule Moving Forward

		Task	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13
		Research + Design										
Finished	All	Research board + microcontrollers for our product (RPI or Jetson)										
Finished	All	Parts list (camera, microcontrollers, processor, etc)										
Finished	Isaiah	Find datasets to train model										
In Progre...	Corin	Research CV libraries / software integration										
In Progre...	Siena	Research skin products for solutions										
In Progre...	Siena	Research libraries for communication protocols										
		Prototyping										
Finished	All	Finalize parts list										
In Progre...	All	Order Parts List 1 - board and sensors needed for input detection										
In Progre...	Siena	UART protocol design										
In Progre...	Siena	CSI protocol design										
Not start...	All	Order Parts List 2 - parts for display and physical product										
In Progre...	Corin	CAD Design for physical mirror										
		Implementation										
Finished	Isaiah	Create face detection script										
Not start...	Isaiah	Pretrain image classification model on datsets										
Finished	Isaiah	Train classifiers on datasets										
In Progre...	Isaiah	Test classifiers on live data										
In Progre...	Isaiah	Create full CV pipeline										
	Isaiah	Implement CV pipeline on edge processor										
	Isaiah	Evaluate pipeline on edge processor										
	Isaiah	Integrate CV system with skincare product recommender system										
	Isaiah	Implement skincare product recommender system										
Not start...	Corin	Assist Isaiah on a local database + skincare recommender script based on confidence										
Not start...	All	Design Magic Mirror App										
	Siena	Bring up camera										
Not start...	Corin	Bring up screen display + make necessary connections to board and power										
Not start...	Corin	Connect LCD display with board to see on-system app UI on screen										
	Corin	Place screen display behind the mirror film to test reflectiveness										
	Siena	Test the LED lighting for optimal image quality										
	Siena	Connect board and camera for input testing										
	All	Build Physical Mirror										
		Testing										
Not start...	Corin	Verify all device connections with SBC										
	Siena	Verify camera functions										
Not start...	Corin	Test screen display updates w/low latency + is visible through mirror										
	All	Test timely updates on on-system app with each user session										
	All	Improvements upon user study										
	All	User study										