

FINAL PROJECT REPORT

Project Aim & Description

A mobile skincare app that analyzes skin through a quiz, delivers personalized product recommendations, builds routines, and supports consistency with reminders and local weather insights. It includes authentication, filtering, data persistence, and Firebase integration.

Functional Requirements	Non Functional Requirements
<ul style="list-style-type: none"> Account creation and email/password login using Firebase Auth for sign-up and sign-in flows. Social sign-in support with Google and Apple, then navigation based on quiz completion status. Password reset flow via email for account recovery. Skincare quiz with multi-step questions, progress tracking, and answer collection leading to results screen. Save and load quiz answers to/from Firestore for the authenticated user. Generate personalized recommendations based on quiz results and display them in the product view. Filter recommended products by brand, label/category, and price range. Build and manage skincare routines (morning/night), including reorder and selecting products per step. Persist routine selections locally via shared preferences for continuity across sessions. Weather and air-quality display based on current location + WeatherAPI integration. Reminder notifications (daily/weekly/twice daily) with user-configurable schedules and persistence of settings. Sign-out and account deletion with proper reauthentication for sensitive actions. 	<ul style="list-style-type: none"> Security & identity management: Use Firebase Authentication for sign-in, and require reauthentication for sensitive actions like account deletion; password reset is also supported for recovery scenarios. Data persistence: Quiz answers are stored in Firestore, and routine/reminder preferences are persisted locally via SharedPreferences for quick restore on app start. Performance: Product lists are cached to reduce repeated network fetches and improve UI responsiveness in product/routine flows. Usability: The quiz provides progress indicators and step-by-step navigation to guide users through the flow clearly. Privacy & permissions: Location access is requested only as needed to fetch local weather data. Reliability: Notifications use timezone initialization and scheduled delivery logic to align reminders with user locale and timing constraints

User Stories

- As a new user, I want to create an account so I can access personalized skincare features.
- As a returning user, I want to log in with email or Google/Apple so I can quickly access my profile and data.
- As a user, I want to reset my password so I can recover access if I forget it.
- As a user, I want to take a guided skincare quiz so the app can understand my skin and goals.
- As a user, I want to see my skin profile results and recommendations so I can understand what products fit my needs.
- As a user, I want to filter recommended products by brand/label/price so I can pick options within my budget and preferences.
- As a user, I want to build and reorder my skincare routine so it matches my habits and schedule.
- As a user, I want to see weather and air-quality info so I can adjust my skincare for local conditions.
- As a user, I want reminders so I stay consistent with my skincare routine.

Test Cases

Test Case	Steps & Expected Result
Account Creation	Steps: Open create account screen → enter email/password/confirm password → submit. Expected: Firebase account is created and the user is redirected to the quiz flow.
Email/Password Login	Steps: Open login screen → enter valid credentials → sign in. Expected: Firebase sign-in succeeds and user transitions to quiz or weather based on stored quiz data.
Password Reset	Steps: Open reset dialog → enter email → submit. Expected: Firebase sends password reset email and confirmation Snackbar appears.
Quiz Completion	Steps: Start quiz → answer all questions → submit. Expected: Results screen appears showing answers and the next-step prompt.
Recommended Generation	Steps: Finish quiz and continue → recommendations are generated. Expected: Recommended products list is prepared and stored, and user can proceed to routine selection flow.
Product Filtering	Steps: Open product page → set brand/label/price filters. Expected: Product list updates to only items matching selected filters.
Routine Building	Steps: Open routine page → select products for steps → reorder steps. Expected: Selected products appear per routine step and order updates are persisted.
Weather & Air Quality	Steps: Open weather page with location permission granted. Expected: Current city, temperature, humidity, and air quality display correctly.
Reminder Scheduling	Steps: In profile page, set daily/weekly/twice reminders → save. Expected: Notification schedule is applied and stored in preferences.
Account Deletion	Steps: In profile page, trigger delete with reauthentication. Expected: User is reauthenticated and account data is deleted in Firebase.

Team Roles & Responsibilities

Joseph	Alexis
Backend & Integration : Implemented the core backend integrations and data services (Firebase authentication, Firestore quiz storage, and local persistence for routines/reminders), and produced lightweight UI elements for validation/testing of service flows during development.	Application Logic & UX/Quiz : Owned the primary application logic, including the full multi-step quiz flow and results pipeline, and crafted the user interface to deliver a polished, professional visual experience across key screens and pages.
Joint Responsibilities	
Implemented CRUD flows and ListView-based data displays, verified the app runs on an emulator, and produced the final report deliverables (aim/description, requirements, user stories, test cases, and role breakdown).	