

# Skincare & Beauty Consultancy Platform

## API Definition Document

### 1 Introduction

This API Definition Document defines the communication contract between the backend and frontend of the Skincare & Beauty Consultancy Platform. It specifies the API's endpoints, methods, arguments, request and response formats, and error handling. This API will facilitate the following user actions:

1. User Login and Registration
2. Submit Skincare Concerns
3. Expert Matching
4. Consultation Scheduling
5. Product Recommendation
6. Follow-up Consultation
7. Rate and Review
8. Trending Makeup Products
9. Search & View Experts
10. Payment Processing

### 2 Base URL

The Base URL for all API requests is <https://api.glowssense.com/>. HTTPS is required for all API requests.

## 3 API Endpoints

### 3.1 Login

<b>Description</b>	Logs in the user to the platform using their credentials.
<b>Method</b>	POST
<b>Path</b>	/login
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "email": "user@example.com", "password": "userpassword" }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "userId": 1, "token": "abcdef12345" }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Invalid email or password" }

### 3.2 Registration

<b>Description</b>	Registers a new user to the platform.
<b>Method</b>	POST
<b>Path</b>	/register
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "email": "user@example.com", "password": "userpassword", "firstName": "John", "lastName": "Doe" }
<b>Successful Response Status</b>	201 Created
<b>Successful Response Body</b>	{ "message": "User registered successfully", "userId": 1 }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Email already in use" }

### 3.3 Submit Skincare Concerns

<b>Description</b>	Allows the user to submit their skincare concerns via a form.
<b>Method</b>	POST
<b>Path</b>	/submit-skincare-concerns
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "userId": 1, "concerns": "Dry skin, acne prone" }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "message": "Skincare concerns submitted successfully", "skincareId": 101 }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Invalid input data" }

### 3.4 Search & View Experts

<b>Description</b>	Allows users to search for beauty experts and view their details.
<b>Method</b>	GET
<b>Path</b>	/search-experts
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	N/A
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "experts": [{"id": 5, "name": "Dr. Emily", "specialization": "Acne Treatment"}] }
<b>Error Response Status</b>	500 Internal Server Error
<b>Error Response Body</b>	{ "error": "Unable to fetch experts" }

### 3.5 Expert Matching

<b>Description</b>	Matches the user with an expert based on their submitted skincare concerns.
<b>Method</b>	POST
<b>Path</b>	/match-expert
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "userId": 1, "skincareId": 101 }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "expertId": 5, "expertName": "Dr. Emily" }
<b>Error Response Status</b>	404 Not Found
<b>Error Response Body</b>	{ "error": "No expert found for your concerns" }

### 3.6 Consultation

<b>Description</b>	Allows users to schedule and attend consultations with matched experts.
<b>Method</b>	POST
<b>Path</b>	/schedule-consultation
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "userId": 1, "expertId": 5, "date": "2023-12-10", "time": "14:00" }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "message": "Consultation scheduled successfully", "consultationId": 123 }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Failed to schedule consultation" }

### 3.7 Payment Processing

<b>Description</b>	Processes payments for consultations and products.
<b>Method</b>	POST
<b>Path</b>	/process-payment
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "userId": 1, "amount": 50.00, "paymentMethod": "Credit Card" }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "message": "Payment successful", "transactionId": 9876 }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Payment failed" }

### 3.8 Product Recommendation

<b>Description</b>	Provides personalized product recommendations based on the user's skin concerns.
<b>Method</b>	POST
<b>Path</b>	/recommend-products
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "userId": 1, "skincareId": 101, "concerns": "acne" }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "products": [{"id": 1, "name": "Acne Treatment Cream", "price": "25.00"}] }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Unable to recommend products" }

### 3.9 Follow-up Consultation

<b>Description</b>	Allows users to schedule follow-up consultations after the initial session.
<b>Method</b>	POST
<b>Path</b>	/follow-up-consultation
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "userId": 1, "expertId": 5, "date": "2023-12-17", "time": "15:00" }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "message": "Follow-up consultation scheduled successfully", "followUpId": 456 }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Failed to schedule follow-up consultation" }

### 3.10 Rate/Review

<b>Description</b>	Allows users to rate and review their expert consultations.
<b>Method</b>	POST
<b>Path</b>	/rate-review
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	{ "userId": 1, "expertId": 5, "rating": 4, "review": "Great consultation!" }
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "message": "Review submitted successfully", "reviewId": 789 }
<b>Error Response Status</b>	400 Bad Request
<b>Error Response Body</b>	{ "error": "Unable to submit review" }

### 3.11 Trending Makeup Products

<b>Description</b>	Allows users to browse trending makeup products without personalized recommendations.
<b>Method</b>	GET
<b>Path</b>	/trending-makeup-products
<b>Query Params</b>	None
<b>Headers</b>	Content-Type: application/json
<b>Content Type</b>	application/json
<b>Body</b>	N/A
<b>Successful Response Status</b>	200 OK
<b>Successful Response Body</b>	{ "products": [{"id": 1, "name": "Lipstick", "price": "12.00"}] }
<b>Error Response Status</b>	500 Internal Server Error
<b>Error Response Body</b>	{ "error": "Unable to fetch trending products" }