

TELEAUTISM Application API Documentation

API Endpoints

1. User Management

Authentication

- **POST /api/auth/login**
 - Login with email/password, Google, or Facebook.
 - Request Body: { email, password } or { provider, token } (for Google/Facebook).
 - Response: { token, user_id, user_type } .
- **POST /api/auth/register**
 - Register a new user (patient, doctor, or admin).
 - Request Body: { first_name, last_name, email, password, phone, user_type, registration_method } .
 - Response: { user_id, token } .
- **POST /api/auth/logout**
 - Logout the user.
 - Request Body: { token } .
 - Response: { message: "Logged out successfully" } .
- **POST /api/auth/reset-password-otp**
 - Request a password reset with email or phone.
 - Request Body: { email || phone } .
 - Response: { otp } .
- **POST /api/auth/reset-password**
 - Reset a password.
 - Request Body: { password, new_password } .
 - Response: { message: "Reset password successfully" } .
- **POST /api/auth/verify-phone**
 - Verify the user's phone.
 - Request Body: { phone } .
 - Response: { otp } .

User Profile

- **GET /api/users/{user_id}**
 - Fetch user profile by ID.
 - Response: { user_data } .
- **PUT /api/users/{user_id}**
 - Update user profile.

- Request Body: { first_name, last_name, phone, profile_picture_url, etc. } .
- Response: { updated_user_details } .
- **DELETE /api/users/{user_id}**
 - Delete a user account.
 - Response: { message: "User deleted" } .

Role-Specific Data

- **GET /api/users/{user_id}/patient-data**
 - Fetch patient-specific data (if user_type = patient).
 - Response: { contact_info, address, current_medications, allergies, payment_methods, etc. } .
- **PUT /api/users/{user_id}/patient-data**
 - Update patient-specific data.
 - Request Body: { contact_info, address, current_medications, allergies, etc. } .
 - Response: { updated_patient_data } .
- **GET /api/users/{user_id}/doctor-data**
 - Fetch doctor-specific data (if user_type = doctor).
 - Response:
 - { specialties_id, hospital, years_of_experience, certifications, subscription_plan, etc. } .
- **PUT /api/users/{user_id}/doctor-data**
 - Update doctor-specific data.
 - Request Body:
 - { specialties_id, hospital, years_of_experience, certifications, consultation_fee, etc. } .
 - Response: { updated_doctor_data } .

2. Specialties

- **GET /api/specialties**
 - Fetch all specialties.
 - Response: [{ specialty_details }] .
- **GET /api/specialties/{specialty_id}**
 - Fetch details of a specific specialty.
 - Response: { specialty_details } .
- **POST /api/specialties**
 - Create a new specialty (admin only).
 - Request Body: { name, description } .
 - Response: { specialty_id } .
- **PUT /api/specialties/{specialty_id}**
 - Update a specialty (admin only).
 - Request Body: { name, description } .
 - Response: { updated_specialty_details } .
- **DELETE /api/specialties/{specialty_id}**
 - Delete a specialty (admin only).

- Response: { message: "Specialty deleted" } .
- **GET /api/specialties/{specialty_id}/doctors**
 - Fetch all doctors with a specific specialty.
 - Response: [{ doctor_details }] .

3. Appointments

- **GET /api/appointments**
 - Fetch all appointments for a user (patient or doctor).
 - Query Params:

user_id, role (patient/doctor), status (pending/confirmed/completed), payment_status (paid/unpaid)

.
 - Response: [{ appointment_details }] .
- **GET /api/appointments/{appointment_id}**
 - Fetch details of a specific appointment.
 - Response: { appointment_details, payment_details } .
- **POST /api/appointments**
 - Create a new appointment.
 - Request Body: { patient_id, doctor_id, date, type, notes, preferred_time, payment_method_id } .
 - Response: { appointment_id, payment_status, payment_url } .
- **PUT /api/appointments/{appointment_id}**
 - Update an appointment (e.g., change status or notes).
 - Request Body: { status, notes } .
 - Response: { updated_appointment_details } .
- **DELETE /api/appointments/{appointment_id}**
 - Cancel an appointment.
 - Response: { message: "Appointment canceled", refund_status } .
- **POST /api/appointments/{appointment_id}/payment**
 - Process payment for an appointment.
 - Request Body: { payment_method_id, amount, currency } .
 - Response: { payment_id, status, receipt_url } .

4. Prescriptions

- **GET /api/prescriptions**
 - Fetch all prescriptions for a user (patient or doctor).
 - Query Params: user_id, role (patient/doctor), status (active/completed/cancelled) .
 - Response: [{ prescription_details }] .
- **GET /api/prescriptions/{prescription_id}**
 - Fetch details of a specific prescription.
 - Response: { prescription_details } .
- **POST /api/prescriptions**

- Create a new prescription.
- Request Body: { patient_id, date, notes, medications } .
- Response: { prescription_id } .
- **PUT /api/prescriptions/{prescription_id}**
 - Update a prescription (e.g., change status or medications).
 - Request Body: { status, notes, medications } .
 - Response: { updated_prescription_details } .
- **DELETE /api/prescriptions/{prescription_id}**
 - Cancel a prescription.
 - Response: { message: "Prescription canceled" } .

5. Chats and Messages

- **GET /api/chats**
 - Fetch all chats for a user.
 - Query Params: user_id, type (direct/group) .
 - Response: [{ chat_details }] .
- **GET /api/chats/{chat_id}**
 - Fetch details of a specific chat.
 - Response: { chat_details, messages } .
- **POST /api/chats**
 - Create a new chat (direct or group).
 - Request Body: { title, participants, type } .
 - Response: { chat_id } .
- **POST /api/chats/{chat_id}/messages**
 - Send a message in a chat.
 - Request Body: { sender_id, content, message_type, referenced_message_id } .
 - Response: { message_id } .
- **PUT /api/chats/{chat_id}/messages/{message_id}**
 - Update a message (e.g., edit content).
 - Request Body: { content } .
 - Response: { updated_message_details } .
- **DELETE /api/chats/{chat_id}/messages/{message_id}**
 - Delete a message.
 - Response: { message: "Message deleted" } .

6. Groups

- **GET /api/groups**
 - Fetch all groups for a user.
 - Query Params: user_id, role (member/admin) .
 - Response: [{ group_details }] .

- **GET /api/groups/{group_id}**
 - Fetch details of a specific group.
 - Response: { group_details, members } .
- **POST /api/groups**
 - Create a new group.
 - Request Body: { name, description, created_by, members } .
 - Response: { group_id } .
- **PUT /api/groups/{group_id}**
 - Update group details (e.g., name, description).
 - Request Body: { name, description } .
 - Response: { updated_group_details } .
- **DELETE /api/groups/{group_id}**
 - Archive or delete a group.
 - Response: { message: "Group archived/deleted" } .

7. Patient Discussions

- **GET /api/discussions**
 - Fetch all discussions for a patient.
 - Query Params: patient_id, status (active/closed/archived) .
 - Response: [{ discussion_details }] .
- **GET /api/discussions/{discussion_id}**
 - Fetch details of a specific discussion.
 - Response: { discussion_details, messages } .
- **POST /api/discussions**
 - Create a new discussion.
 - Request Body: { group_id, patient_id, title } .
 - Response: { discussion_id } .
- **PUT /api/discussions/{discussion_id}**
 - Update discussion status (e.g., close or archive).
 - Request Body: { status } .
 - Response: { updated_discussion_details } .

8. History

- **GET /api/history/appointments**
 - Fetch appointment history for a user.
 - Query Params: user_id, role (patient/doctor) .
 - Response: [{ appointment_history }] .
- **GET /api/history/prescriptions**
 - Fetch prescription history for a user.
 - Query Params: user_id, role (patient/doctor) .

- Response: [{ prescription_history }] .
- **GET /api/history/groups**
 - Fetch group action history.
 - Query Params: group_id, action_type (created/member_added/etc.) .
 - Response: [{ group_history }] .
- **GET /api/history/payments**
 - Fetch payment history for a user.
 - Query Params: user_id, type (appointment/subscription), status (completed/pending/failed) .
 - Response: [{ payment_history }] .

9. Admin Endpoints

- **GET /api/admin/users**
 - Fetch all users (with filters for user_type, status, etc.).
 - Query Params: user_type, status .
 - Response: [{ user_details }] .
- **PUT /api/admin/users/{user_id}/status**
 - Update user status (e.g., suspend or activate).
 - Request Body: { status } .
 - Response: { updated_user_details } .
- **DELETE /api/admin/users/{user_id}**
 - Delete a user (admin-only).
 - Response: { message: "User deleted" } .
- **GET /api/admin/payments**
 - Fetch all payments (with filters).
 - Query Params: status, type, date_range .
 - Response: [{ payment_details }] .
- **PUT /api/admin/payments/{payment_id}**
 - Update payment status (e.g., mark as refunded).
 - Request Body: { status, notes } .
 - Response: { updated_payment_details } .

10. Payment Methods

- **GET /api/payment-methods**
 - Fetch all payment methods for a user.
 - Query Params: user_id .
 - Response: [{ payment_method_details }] .
- **GET /api/payment-methods/{payment_method_id}**
 - Fetch details of a specific payment method.
 - Response: { payment_method_details } .
- **POST /api/payment-methods**

- Add a new payment method.
- Request Body:


```
{ user_id, type, provider, card_number, expiry_date, billing_address, is_default } .
```
- Response: { payment_method_id } .
- **PUT /api/payment-methods/{payment_method_id}**
 - Update a payment method.
 - Request Body: { expiry_date, billing_address, is_default } .
 - Response: { updated_payment_method_details } .
- **DELETE /api/payment-methods/{payment_method_id}**
 - Remove a payment method.
 - Response: { message: "Payment method removed" } .

11. Payments

- **GET /api/payments**
 - Fetch all payments for a user.
 - Query Params: user_id, type (appointment/subscription), status (completed/pending/failed) .
 - Response: [{ payment_details }] .
- **GET /api/payments/{payment_id}**
 - Fetch details of a specific payment.
 - Response: { payment_details, receipt_url } .
- **POST /api/payments**
 - Process a new payment.
 - Request Body:


```
{ user_id, recipient_id, payment_method_id, type, related_entity_id, amount, currency } .
```
 - Response: { payment_id, status, transaction_id, receipt_url } .
- **POST /api/payments/{payment_id}/refund**
 - Process a refund for a payment.
 - Request Body: { reason, amount (optional for partial refund) } .
 - Response: { refund_id, status } .

12. Subscriptions

- **GET /api/subscription-plans**
 - Fetch all available subscription plans.
 - Response: [{ subscription_plan_details }] .
- **GET /api/subscriptions**
 - Fetch subscription details for a doctor.
 - Query Params: doctor_id .
 - Response: { subscription_details } .
- **POST /api/subscriptions**
 - Subscribe to a plan.

- Request Body: { doctor_id, plan_id, payment_method_id, auto_renew } .
- Response: { subscription_id, status, payment_id } .
- **PUT /api/subscriptions/{subscription_id}**
 - Update subscription (e.g., change plan or payment method).
 - Request Body: { plan_id, payment_method_id, auto_renew } .
 - Response: { updated_subscription_details } .
- **DELETE /api/subscriptions/{subscription_id}**
 - Cancel a subscription.
 - Request Body: { reason } .
 - Response: { message: "Subscription cancelled", effective_end_date } .