

FCM token registration endpoint, verification of reminder permissions, and the Full Notification & Reminders API Guide for your Flutter developer.

1. FCM Token Registration Endpoint Inspection

The backend provides two ways to register or sync the FCM device token:

A. Dedicated Push Token Sync (Primary & Token Refresh)

- **Method & URL:** POST /api/v1/auth/push-token
- **Controller:** AuthController::updatePushToken
- **Validation:** UpdatePushTokenRequest
- **Headers:**

```
Authorization: Bearer <SANCTUM_ACCESS_TOKEN>
Content-Type: application/json
Accept: application/json
X-Device-Id: <DEVICE_UNIQUE_ID>
```

- **Request Body:**

```
{
  "push_token": "fM38sd9F8sd7f_fcm_token_string...",
  "device_id": "e81b6722-e424-4f2b-8a89-13835ce08f51",
  "platform": "android",
  "device_name": "Google Pixel 8 Pro",
  "os_version": "Android 14",
  "app_version": "1.0.0"
}
```

- **Success Response (200 OK):**

```
{
  "success": true,
  "data": {
    "device_uuid": "e81b6722-e424-4f2b-8a89-13835ce08f51",
    "device_id": "e81b6722-e424-4f2b-8a89-13835ce08f51",
    "platform": "android"
  },
  "message": "Push token updated successfully."
}
```

B. Remove Token on Logout

- **Method & URL:** DELETE /api/v1/auth/push-token
- **Request Body:**

```
{
  "device_id": "e81b6722-e424-4f2b-8a89-13835ce08f51"
}
```

- **Success Response (200 OK):**

```
{
  "success": true,
  "data": null,
  "message": "Push token removed."
}
```

2. Reminder Creation Permissions for Users

In your backend architecture (`ReminderPolicy.php`):

- Reminders are personal to each user.
- Any active user (`$user->status === 'active'`) has permission to create, view, snooze, and complete their own reminders across all roles (Advocate, Junior Advocate, Senior Advocate, Clerk, Accountant, Admin).
- No administrative role permission slug is required — every authenticated user can immediately use `POST /api/v1/reminders` .

Full Notification & Reminder API Guide for Flutter

Base URL: `https://<api-domain>/api/v1`

Required Standard Headers for all requests:

```
Authorization: Bearer <SANCTUM_TOKEN>
Accept: application/json
Content-Type: application/json
X-Device-Id: <DEVICE_ID>
```

Section 1: In-App Notifications Feed

1.1 Get Notifications List & Unread Count

Returns paginated notifications scoped strictly to the authenticated user.

- **Method:** `GET`
- **Path:** `/api/v1/notifications`
- **Query Parameters:**
 - `unread` (*optional, bool*): `true` to return only unread notifications.
 - `category` (*optional, string*): Filter by `case` , `task` , `hearing` , `follow_up` , `payment` , `reminder` , `security` .
 - `per_page` (*optional, int*): Items per page (default: `30`).
 - `page` (*optional, int*): Page number.
- **Example:** `GET /api/v1/notifications?unread=true&category=task`
- **Response (`200 OK`):**

```
{
  "success": true,
  "data": [
    {
      "uuid": "c3a9f0e1-4567-4890-a1b2-c3d4e5f60718",
      "type": "task_assigned",
      "category": "task",
      "title": "Task Assigned: Draft Bail Application",
      "body": "You have been assigned to case CASE/2026/0012.",
      "subject_type": "task",
      "subject_uuid": "8b5d3a1f-9988-4433-2211-abcdef123456",
      "action_route": "/tasks/8b5d3a1f-9988-4433-2211-abcdef123456",
      "actor": {
        "uuid": "u1a2b3c4-d5e6-7f8a-9b0c-1d2e3f4a5b6c",
        "name": "Senior Partner"
      },
      "is_read": false,
      "read_at": null,
      "created_at": "2026-09-22T08:30:00.000000Z"
    }
  ],
  "meta": {
    "unread_count": 5,
    "pagination": {
      "current_page": 1,
      "per_page": 30,
      "total": 5,
      "last_page": 1
    }
  }
}
```

1.2 Mark Single Notification as Read

- **Method:** POST
- **Path:** /api/v1/notifications/{uuid}/read
- **Response (200 OK):**

```
{
  "success": true,
  "data": {
    "uuid": "c3a9f0e1-4567-4890-a1b2-c3d4e5f60718",
    "is_read": true,
    "read_at": "2026-09-22T08:35:10.000000Z"
  },
  "message": "Notification marked as read."
}
```

1.3 Mark All Notifications as Read

- **Method:** POST
- **Path:** /api/v1/notifications/read-all
- **Response (200 OK):**

```
{
  "success": true,
  "data": {
    "marked_read": 5
  },
  "message": "All notifications marked as read."
}
```

1.4 Bulk Read Receipts (Offline Sync)

Used when a user read notifications while mobile was offline.

- **Method:** POST
- **Path:** /api/v1/notifications/read-receipts
- **Request Body:**

```
{
  "uuids": [
    "c3a9f0e1-4567-4890-a1b2-c3d4e5f60718",
    "d4b0a1f2-5678-4901-b2c3-d4e5f6a7b8c9"
  ]
}
```

- **Response (200 OK):**

```
{
  "success": true,
  "data": {
    "marked_read": 2
  },
  "message": "Read receipts recorded."
}
```

Section 2: Notification Preferences

2.1 Get Notification Settings

- **Method:** GET
- **Path:** /api/v1/notification-preferences
- **Response (200 OK):**

```
{
  "success": true,
  "data": [
    { "category": "case", "in_app": true, "push": true },
    { "category": "task", "in_app": true, "push": true },
    { "category": "hearing", "in_app": true, "push": true },
    { "category": "follow_up", "in_app": true, "push": true },
    { "category": "payment", "in_app": true, "push": false },
    { "category": "reminder", "in_app": true, "push": true }
  ],
  "meta": {
    "mandatory_note": "Security notifications, including case access changes, cannot be turned off."
  }
}
```

2.2 Update Notification Settings

- **Method:** PUT
- **Path:** /api/v1/notification-preferences
- **Request Body:**

```
{
  "preferences": [
    {
      "category": "task",
      "in_app": true,
      "push": true
    },
    {
      "category": "payment",
      "in_app": true,
      "push": false
    }
  ]
}
```

- **Response (200 OK):** Returns the updated preferences array.



Section 3: Reminders API

3.1 List Reminders

- **Method:** GET
- **Path:** /api/v1/reminders
- **Query Parameters:**
 - *pending (optional, bool):* true to return only pending (uncompleted) alarms.
 - *status (optional, string):* pending, completed, dismissed, cancelled.
 - *reminder_type (optional, string):* personal, hearing, task, follow_up, filing_deadline, limitation, client_meeting, payment, document_expiry, custom.
 - *subject_uuid (optional, string):* Filter by linked case or task UUID.
- **Response (200 OK):**

```
{
  "success": true,
  "data": [
    {
      "uuid": "8f3b2e1a-4567-4890-a1b2-c3d4e5f60718",
      "title": "Prepare arguments for Bail Hearing",
      "description": "Review witness statements and prior rulings.",
      "reminder_type": "hearing",
      "remind_at": "2026-09-23T10:00:00.000000Z",
      "original_remind_at": "2026-09-23T10:00:00.000000Z",
      "status": "pending",
      "subject_type": "hearing",
      "subject_uuid": "h1a2b3c4-d5e6-7f8a-9b0c-1d2e3f4a5b6c",
      "snooze_count": 0,
      "can_snooze": true,
      "is_urgent": true,
      "has_fired": false,
      "completed_at": null,
      "is_recurring": false,
      "version_number": 1
    }
  ],
  "meta": {
    "pagination": {
      "current_page": 1,
      "per_page": 50,
      "total": 1,
      "last_page": 1
    }
  }
}
```

3.2 Create a Reminder

- **Method:** POST
- **Path:** /api/v1/reminders
- **Request Body (Single Reminder):**

```
{
  "title": "Follow up with client on evidence documents",
  "description": "Call client before 2 PM.",
  "reminder_type": "follow_up",
  "remind_at": "2026-09-23T14:00:00Z",
  "is_urgent": false,
  "subject_type": "client",
  "subject_uuid": "c1-uuid-12345"
}
```

- **Request Body (Recurring Reminder Series - optional):**

```
{
  "title": "Weekly Case Status Review",
  "reminder_type": "task",
  "remind_at": "2026-09-25T09:00:00Z",
  "is_urgent": false,
  "recurrence": {
    "recurrence_type": "weekly",
    "interval": 1,
    "occurrences": 8
  }
}
```

- **Response (201 Created):**

```
{
  "success": true,
  "data": [
    {
      "uuid": "8f3b2e1a-4567-4890-a1b2-c3d4e5f60718",
      "title": "Follow up with client on evidence documents",
      "status": "pending",
      "remind_at": "2026-09-23T14:00:00.000000Z",
      "is_urgent": false
    }
  ],
  "message": "Reminder created successfully."
}
```

3.3 Complete a Reminder

- **Method:** POST
- **Path:** /api/v1/reminders/{uuid}/complete
- **Request Body:**

```
{
  "note": "Client submitted documents at 1:30 PM."
}
```

- **Response (200 OK):**

```
{
  "success": true,
  "data": {
    "uuid": "8f3b2e1a-4567-4890-a1b2-c3d4e5f60718",
    "status": "completed",
    "completed_at": "2026-09-23T13:30:00.000000Z"
  },
  "message": "Reminder completed."
}
```

3.4 Snooze a Reminder

- **Method:** POST

- **Path:** `/api/v1/reminders/{uuid}/snooze`
- **Request Body:**

```
{
  "minutes": 15
}
```

- **Response (200 OK):**

```
{
  "success": true,
  "data": {
    "uuid": "8f3b2e1a-4567-4890-a1b2-c3d4e5f60718",
    "remind_at": "2026-09-23T14:15:00.000000Z",
    "snooze_count": 1,
    "can_snooze": true
  },
  "message": "Reminder snoozed."
}
```

3.5 Dismiss / Delete Reminder

- **Dismiss:** POST `/api/v1/reminders/{uuid}/dismiss` (marks as dismissed)
- **Delete:** DELETE `/api/v1/reminders/{uuid}` (deletes record)

Section 4: FCM Push Message Payload Structure

When the backend sends a push via Google FCM v1, Flutter receives:

```
{
  "notification": {
    "title": "Task Assigned: Draft Bail Application",
    "body": "You have been assigned to case CASE/2026/0012."
  },
  "data": {
    "notification_uuid": "c3a9f0e1-4567-4890-a1b2-c3d4e5f60718",
    "type": "task_assigned",
    "category": "task",
    "subject_type": "task",
    "subject_uuid": "8b5d3a1f-9988-4433-2211-abcdef123456",
    "action_route": "/tasks/8b5d3a1f-9988-4433-2211-abcdef123456",
    "click_action": "FLUTTER_NOTIFICATION_CLICK",
    "created_at": "2026-09-22T08:30:00Z"
  }
}
```

- **Android Channel ID configured by backend:** `acms_high_importance_channel`
- **Deep Linking:** Flutter should inspect `message.data['action_route']` on notification click and navigate immediately.