

SMART TOURISM GUIDE

API Reference Document

All Endpoints — Request & Response Schemas — v1.0

Base URL: <https://api.tourguide.com/api/v1>

All requests must include the header: Authorization: Bearer <JWT> except public endpoints.

All timestamps are ISO 8601 format: 2025-01-15T10:30:00Z

All IDs are UUID strings unless otherwise noted.

Method colour legend: **[GET]** **[POST]** **[PUT]** **[PATCH]** **[DELETE]**

1. Auth Endpoints

[POST] /auth/register — Register a new tourist account

Auth: PUBLIC

Request Body:

```
{
  "email":      "ali@example.com",      // required, unique
  "password":   "SecurePass123!",       // required, min 8 chars
  "firstName":  "Ali",                  // required
  "lastName":   "Yilmaz",               // required
  "phoneNumber": "05301234567",         // optional
  "preferredLanguage": "tr"             // optional: tr | en | de
}
```

Response 201:

```
{
  "userId":      "uuid",
  "email":       "ali@example.com",
  "firstName":   "Ali",
  "lastName":    "Yilmaz",
  "role":        "TOURIST",
  "token":       "eyJhbGciOiJIUzI1NiIs..."
}
```

[POST] /auth/login — Authenticate and receive JWT token

Auth: PUBLIC

Request Body:

```
{
  "email":      "ali@example.com",
  "password":   "SecurePass123!"
}
```

Response 200:

```
{
  "token":       "eyJhbGciOiJIUzI1NiIs...",
  "userId":      "uuid",
  "firstName":   "Ali",
  "role":        "TOURIST",
  "expiresAt":   "2025-02-15T10:30:00Z"
}
```

[POST] /auth/refresh — Refresh an expiring JWT token

Auth: TOURIST, MODERATOR, CONTENT_EDITOR, SUPERADMIN

Request Body:

```
{
```

```
  "token": "eyJhbGciOiJIUzI1NiIs..."
}
```

Response 200:

```
{
  "token": "eyJhbGciOiJIUzI1NiIs...",
  "expiresAt": "2025-02-16T10:30:00Z"
}
```

[POST] `/auth/logout` — *Invalidate current token*

Auth: TOURIST, MODERATOR, CONTENT_EDITOR, SUPERADMIN

Request Body: (empty)

Response 204: No Content

2. User Endpoints

[GET] `/users/me` — *Get current user profile*

Auth: TOURIST

Response 200:

```
{
  "userId":      "uuid",
  "email":       "ali@example.com",
  "firstName":   "Ali",
  "lastName":    "Yilmaz",
  "phoneNumber": "05301234567",
  "profilePhotoUrl": "https://minio.../photo.jpg",
  "preferredLanguage": "tr",
  "ageGroup":    "ADULT",
  "expPoints":   1250,
  "level":       5,
  "createdAt":   "2025-01-01T10:00:00Z"
}
```

[PATCH] `/users/me` — *Update current user profile*

Auth: TOURIST

Request Body (all fields optional):

```
{
  "firstName":   "Ali",
  "lastName":    "Yilmaz",
  "phoneNumber": "05301234567",
  "preferredLanguage": "en",
  "ageGroup":    "ADULT"
}
```

Response 200: Updated user object (same schema as GET `/users/me`)

[POST] `/users/me/photo` — *Upload profile photo*

Auth: TOURIST

Request: multipart/form-data

`file: <binary image> // JPEG or PNG, max 5MB`

Response 200:

```
{
  "profilePhotoUrl": "https://minio.../photo.jpg"
}
```

2.1 Favorites

[GET] `/users/me/favorites` — *List all favorite places*

Auth: TOURIST

Response 200:

```
[
  {
    "favoriteId": "uuid",
    "place": {
      "placeId": "uuid",
      "name": "Saat Kulesi",
      "category": "LANDMARK",
      "latitude": 38.4192,
      "longitude": 27.1287,
      "avgRating": 4.7
    },
    "savedAt": "2025-01-10T09:00:00Z"
  }
]
```

[POST] `/users/me/favorites` — *Add a place to favorites*

Auth: TOURIST

Request Body:

```
{
  "placeId": "uuid"
}
```

Response 201:

```
{
  "favoriteId": "uuid",
  "placeId": "uuid",
  "savedAt": "2025-01-10T09:00:00Z"
}
```

[DELETE] `/users/me/favorites/{favoriteId}` — *Remove a place from favorites*

Auth: TOURIST

Response 204: No Content

3. Place Endpoints

[GET] `/places/nearby` — *Get nearby places based on user location*

Auth: TOURIST

Query Parameters:

Field	Description
lat	double — User latitude (required)
lng	double — User longitude (required)
radius	int — Search radius in metres, default 1000, max 5000
category	string — Filter: RESTAURANT LANDMARK MUSEUM CAFE HOTEL PARK
page	int — Page number, default 0
size	int — Page size, default 20, max 50

Response 200:

```
{
  "content": [
    {
      "placeId": "uuid",
      "externalId": "ChIJ...",
      "name": "Saat Kulesi",
      "nameTr": "Saat Kulesi",
      "nameEn": "Clock Tower",
      "category": "LANDMARK",
      "latitude": 38.4192,
      "longitude": 27.1287,
      "distanceMeters": 320,
      "avgRating": 4.7,
      "reviewCount": 1243,
      "openNow": true,
      "photoUrl": "https://minio.../place.jpg"
    }
  ],
  "totalElements": 47,
  "totalPages": 3,
  "currentPage": 0
}
```

[GET] `/places/{placeId}` — *Get full details of a single place*

Auth: TOURIST

Response 200:

```
{
  "placeId": "uuid",
  "name": "Saat Kulesi",
  "nameTr": "Saat Kulesi",
  "nameEn": "Clock Tower",
  "category": "LANDMARK",
  "latitude": 38.4192,
  "longitude": 27.1287,
  "distanceMeters": 320,
  "avgRating": 4.7,
  "reviewCount": 1243,
  "openNow": true,
  "photoUrl": "https://minio.../place.jpg"
}
```

```
{
  "nameEn": "Clock Tower",
  "category": "LANDMARK",
  "latitude": 38.4192,
  "longitude": 27.1287,
  "avgRating": 4.7,
  "reviewCount": 1243,
  "details": {
    "description": "İzmir'in simgesi...",
    "address": "Konak Meydanı, İzmir",
    "phone": "+90 232 000 0000",
    "website": "https://..."
  },
  "openingHours": {
    "monday": "09:00-18:00",
    "tuesday": "09:00-18:00",
    "wednesday": "09:00-18:00",
    "thursday": "09:00-18:00",
    "friday": "09:00-18:00",
    "saturday": "10:00-16:00",
    "sunday": "closed"
  },
  "photos": ["https://minio.../1.jpg", "https://minio.../2.jpg"],
  "isFavorited": false
}
```

[GET] `/places/search` — Search places by keyword

Auth: TOURIST

Query Parameters:

Field	Description
q	string — Search keyword (required, min 2 chars)
lat	double — Optional location bias latitude
lng	double — Optional location bias longitude
page	int — Page number, default 0
size	int — Page size, default 20

Response 200: Same paginated structure as GET `/places/nearby`

4. Quest Endpoints

[GET] `/quests` — *List all active quests*

Auth: TOURIST

Query Parameters:

Field	Description
lat	double — Optional: filter quests near this location
lng	double — Optional: filter quests near this location
status	string — Filter: ALL IN_PROGRESS COMPLETED NOT_STARTED (default: ALL)

Response 200:

```
[
  {
    "questId": "uuid",
    "title": "İzmir Keşif Questi",
    "description": "İzmir'in ikonik noktalarını keşfet",
    "expReward": 500,
    "totalSteps": 5,
    "userProgress": {
      "completedSteps": 2,
      "status": "IN_PROGRESS"
    },
    "region": "Konak, İzmir",
    "thumbnailUrl": "https://minio.../quest.jpg"
  }
]
```

[GET] `/quests/{questId}` — *Get quest details with all steps*

Auth: TOURIST

Response 200:

```
{
  "questId": "uuid",
  "title": "İzmir Keşif Questi",
  "description": "İzmir'in ikonik noktalarını keşfet",
  "expReward": 500,
  "steps": [
    {
      "stepId": "uuid",
      "order": 1,
      "place": {
        "placeId": "uuid",
        "name": "Saat Kulesi",
        "latitude": 38.4192,
        "longitude": 27.1287
      }
    }
  ]
}
```



```
    },
    "hint": "Konak Meydanı'nda seni bekliyor",
    "requiresPhoto": true,
    "userStatus": "COMPLETED"
  }
],
"userProgress": {
  "completedSteps": 2,
  "status": "IN_PROGRESS",
  "startedAt": "2025-01-10T09:00:00Z"
}
}
```

[POST] /quests/{questId}/start — *Start a quest*

Auth: TOURIST

Request Body: (empty)

Response 201:

```
{
  "userQuestId": "uuid",
  "questId": "uuid",
  "status": "IN_PROGRESS",
  "startedAt": "2025-01-10T09:00:00Z"
}
```

[POST] /quests/{questId}/steps/{stepId}/verify — *Submit step verification (GPS + photo)*

Auth: TOURIST

Request: multipart/form-data

```
photo: <binary image> // JPEG or PNG, max 5MB
latitude: 38.4192 // User current latitude
longitude: 27.1287 // User current longitude
```

Response 200:

```
{
  "verificationId": "uuid",
  "status": "APPROVED" | "PENDING_REVIEW" | "REJECTED",
  "message": "Step completed!" | "Under review" | "GPS check failed",
  "expEarned": 100,
  "confidenceScore": 0.87, // Present when status is APPROVED or PENDING_REVIEW
  "confidenceThreshold": 0.85, // The threshold configured for this step by admin
  "badgeAwarded": { // Present only if this was the final step and quest has a badge
    "badgeId": "uuid",
    "name": "Explorer",
    "iconUrl": "https://minio.../explorer.png"
  }
}
```


5. ToDo Endpoints

[GET] `/todos` — *Get current user's ToDo list*

Auth: TOURIST

Response 200:

```
[
  {
    "todoId": "uuid",
    "note": "Mutlaka git!",
    "isCompleted": false,
    "createdAt": "2025-01-10T09:00:00Z",
    "place": {
      "placeId": "uuid",
      "name": "Efes Antik Kenti",
      "latitude": 37.9395,
      "longitude": 27.3417
    }
  }
]
```

[POST] `/todos` — *Create a new ToDo item*

Auth: TOURIST

Request Body:

```
{
  "placeId": "uuid", // optional - link to an existing place
  "note": "Mutlaka git!" // optional - free-text note
}
```

Response 201:

```
{
  "todoId": "uuid",
  "placeId": "uuid",
  "note": "Mutlaka git!",
  "isCompleted": false,
  "createdAt": "2025-01-10T09:00:00Z"
}
```

[PATCH] `/todos/{todoId}` — *Update a ToDo item*

Auth: TOURIST

Request Body (all fields optional):

```
{
  "note": "Güncellendi",
  "isCompleted": true
}
```

Response 200: Updated ToDo object

[DELETE] `/todos/{todoId}` — *Delete a ToDo item*

Auth: TOURIST

Response 204: No Content

6. Route Endpoints

[GET] `/routes` — *List all active routes*

Auth: TOURIST

Query Parameters:

Field	Description
lat	double — Optional: return only routes whose radius contains this point
lng	double — Optional: paired with lat

Response 200:

```
[
  {
    "routeId": "uuid",
    "name": "Kemeraltı Çarşısı Rotası",
    "description": "Tarihi çarşığı keşfet",
    "centerLat": 38.4126,
    "centerLng": 27.1384,
    "radiusMeters": 800,
    "totalPlaces": 7,
    "estimatedMinutes": 120,
    "expReward": 300,
    "thumbnailUrl": "https://minio.../route.jpg",
    "isNearby": true
  }
]
```

[GET] `/routes/{routeId}` — *Get full route with ordered stops*

Auth: TOURIST

Response 200:

```
{
  "routeId": "uuid",
  "name": "Kemeraltı Çarşısı Rotası",
  "description": "Tarihi çarşığı keşfet",
  "centerLat": 38.4126,
  "centerLng": 27.1384,
  "radiusMeters": 800,
  "expReward": 300,
  "stops": [
    {
      "order": 1,
      "estimatedMinutes": 15,
      "notes": "Dıştan mutlaka bir fotoğraf çek",
      "place": {
        "placeId": "uuid",
        "name": "Kızlarağası Hanı",
        "latitude": 38.4128,

```

```
        "longitude": 27.1382,  
        "category": "LANDMARK"  
    }  
}  
]  
}
```

7. Chat Endpoints

[POST] `/chat/sessions` — *Start a new chat session*

Auth: TOURIST

Request Body:

```
{
  "language": "tr"    // optional: tr | en | de (default: user preference)
}
```

Response 201:

```
{
  "sessionId": "uuid",
  "startedAt": "2025-01-10T09:00:00Z"
}
```

[POST] `/chat/sessions/{sessionId}/messages` — *Send a message and receive AI response*

Auth: TOURIST

Request Body:

```
{
  "content": "Kemeraltı'nda kahvaltı için ne önerirsin?",
  "latitude": 38.4126,    // optional: current location for context
  "longitude": 27.1384    // optional
}
```

Response 200:

```
{
  "messageId": "uuid",
  "role": "ASSISTANT",
  "content": "Kemeraltı'nda sabah kahvaltısı için...",
  "sentAt": "2025-01-10T09:00:05Z",
  "functionCalls": [
    {
      "type": "ADD_TODO",
      "payload": { "placeId": "uuid", "note": "Chatbot önerdi" }
    }
  ]
}
```

Note: functionCalls array is present only when the chatbot performs an action (e.g., adding to ToDo list). The client should execute these actions automatically.

[GET] `/chat/sessions/{sessionId}/messages` — *Retrieve message history for a session*

Auth: TOURIST

Query Parameters:

Field	Description
page	int — Page number, default 0
size	int — Page size, default 20

Response 200:

```
{
  "content": [
    {
      "messageId": "uuid",
      "role": "USER" | "ASSISTANT",
      "content": "...",
      "sentAt": "2025-01-10T09:00:00Z"
    }
  ],
  "totalElements": 24,
  "totalPages": 2
}
```

[DELETE] `/chat/sessions/{sessionId}` — End and archive a chat session

Auth: TOURIST

Response 204: No Content. Session is persisted to PostgreSQL before deletion from Redis.

8. Image Recognition Endpoints

[POST] /images/identify — *Identify a landmark from a photo*

Auth: TOURIST

Request: multipart/form-data

```
photo:      <binary image>    // JPEG or PNG, max 5MB
latitude:   38.4192             // User current latitude (required for pilot
zone check)
longitude:  27.1287             // User current longitude
```

Response 200:

```
{
  "identified": true,
  "label":      "İzmir Saat Kulesi",
  "labelEn":    "İzmir Clock Tower",
  "confidence": 0.91,
  "source":     "CUSTOM_MODEL" | "GOOGLE_VISION",
  "place": {
    "placeId":  "uuid",
    "name":      "Saat Kulesi",
    "latitude":  38.4192,
    "longitude": 27.1287,
    "category":  "LANDMARK"
  },
  "description": "İzmir Saat Kulesi, 1901 yılında inşa edilmiş..."
}
```

Response 200 (not identified):

```
{
  "identified": false,
  "message":    "Landmark could not be identified. Try a clearer photo."
}
```

Response 403 (outside pilot zone):

```
{
  "error":      "OUTSIDE_PILOT_ZONE",
  "message":    "This feature is not yet available in your region."
}
```

9. Review Endpoints

[GET] `/places/{placeId}/reviews` — *Get approved reviews for a place*

Auth: TOURIST

Query Parameters:

Field	Description
page	int — Page number, default 0
size	int — Page size, default 10
sort	string — NEWEST HIGHEST_RATING LOWEST_RATING (default: NEWEST)

Response 200:

```
{
  "content": [
    {
      "reviewId": "uuid",
      "rating": 4,
      "comment": "Harika bir yer!",
      "user": {
        "firstName": "Ali",
        "profilePhotoUrl": "https://minio.../photo.jpg"
      },
      "createdAt": "2025-01-10T09:00:00Z"
    }
  ],
  "totalElements": 124,
  "totalPages": 13,
  "avgRating": 4.7
}
```

[POST] `/places/{placeId}/reviews` — *Submit a review for a place*

Auth: TOURIST

Request Body:

```
{
  "rating": 4, // required, integer 1-5
  "comment": "Harika bir yer!" // optional
}
```

Response 201:

```
{
  "reviewId": "uuid",
  "rating": 4,
  "comment": "Harika bir yer!",
  "status": "PENDING",
  "message": "Your review has been submitted and is pending moderation."
}
```

[DELETE] `/places/{placeId}/reviews/{reviewId}` — *Delete own review*

Auth: TOURIST

Response 204: No Content

10. Badge Endpoints

[GET] `/users/me/badges` — *Get current user's earned badges*

Auth: TOURIST

Response 200:

```
[
  {
    "badgeId": "uuid",
    "name": "Explorer",
    "description": "Keşfedenlerin rozeti",
    "iconUrl": "https://minio.../explorer.png",
    "earnedAt": "2025-01-10T09:00:00Z"
  }
]
```

[GET] `/badges` — *List all available badges (catalog)*

Auth: TOURIST

Response 200:

```
[
  {
    "badgeId": "uuid",
    "name": "Explorer",
    "description": "Keşfedenlerin rozeti",
    "iconUrl": "https://minio.../explorer.png",
    "earnedByUser": true // whether the current user has earned this badge
  }
]
```

10.1 Badge Management (Content Editor)

[GET] `/admin/editor/badges` — *List all badges including inactive*

Auth: SUPERADMIN, CONTENT_EDITOR

Response 200:

```
[
  {
    "badgeId": "uuid",
    "name": "Explorer",
    "description": "Keşfedenlerin rozeti",
    "iconUrl": "https://minio.../explorer.png",
    "isActive": true,
    "createdAt": "2025-01-01T10:00:00Z"
  }
]
```

[POST] `/admin/editor/badges` — *Create a new badge*

Auth: SUPERADMIN, CONTENT_EDITOR

Request: multipart/form-data

```
name:      Explorer
description: Keşfedenlerin rozeti
icon:      <binary image>    // PNG, max 2MB, square recommended
```

Response 201:

```
{
  "badgeId": "uuid",
  "name": "Explorer",
  "description": "Keşfedenlerin rozeti",
  "iconUrl": "https://minio.../explorer.png",
  "isActive": true
}
```

[PUT] `/admin/editor/badges/{badgeId}` — *Update a badge*

Auth: SUPERADMIN, CONTENT_EDITOR

Request: multipart/form-data (all fields optional):

```
name:      Explorer Updated
description: Güncellenmiş açıklama
icon:      <binary image>    // optional, replaces existing icon
```

Response 200: Updated badge object

[DELETE] `/admin/editor/badges/{badgeId}` — *Soft-delete a badge*

Auth: SUPERADMIN, CONTENT_EDITOR

Note: Soft delete only. Users who already earned this badge retain it. It will no longer appear in the catalog for new quests.

Response 200:

```
{
  "badgeId": "uuid",
  "isActive": false
}
```

11. Notification Endpoints

[GET] `/notifications` — *Get user notification inbox*

Auth: TOURIST

Query Parameters:

Field	Description
<code>unreadOnly</code>	boolean — Return only unread notifications (default: false)
<code>page</code>	int — Page number, default 0
<code>size</code>	int — Page size, default 20

Response 200:

```
{
  "content": [
    {
      "notificationId": "uuid",
      "type": "QUEST_APPROVED" | "QUEST_REJECTED" | "ROUTE_NEARBY" |
      "LEVEL_UP" | "BADGE_EARNED",
      "title": "Quest adımı onaylandı!",
      "body": "Saat Kulesi adımı tamamladın. +100 EXP kazandın.",
      "isRead": false,
      "createdAt": "2025-01-10T09:00:00Z"
    }
  ],
  "totalElements": 8,
  "unreadCount": 3
}
```

[PATCH] `/notifications/{notificationId}/read` — *Mark a notification as read*

Auth: TOURIST

Request Body: (empty)

Response 200:

```
{
  "notificationId": "uuid",
  "isRead": true
}
```

[PATCH] `/notifications/read-all` — *Mark all notifications as read*

Auth: TOURIST

Request Body: (empty)

Response 200:

```
{
  "updatedCount": 3
}
```

[POST] `/notifications/device-token` — *Register FCM device token for push notifications*

Auth: TOURIST

Request Body:

```
{
  "deviceToken": "fcm-token-string",
  "platform":    "ANDROID" | "IOS"
}
```

Response 200:

```
{
  "registered": true
}
```

12. Admin — Superadmin Endpoints

[GET] `/admin/superadmin/users` — *List all users with roles and status*

Auth: SUPERADMIN

Query Parameters:

Field	Description
role	string — Filter: TOURIST MODERATOR CONTENT_EDITOR SUPERADMIN
isActive	boolean — Filter by active/suspended status
page	int — Page number, default 0
size	int — Page size, default 20

Response 200:

```
{
  "content": [
    {
      "userId": "uuid",
      "email": "ali@example.com",
      "firstName": "Ali",
      "lastName": "Yilmaz",
      "role": "MODERATOR",
      "isActive": true,
      "createdAt": "2025-01-01T10:00:00Z"
    }
  ],
  "totalElements": 150
}
```

[PATCH] `/admin/superadmin/users/{userId}/role` — *Assign a role to a user*

Auth: SUPERADMIN

Request Body:

```
{
  "role": "MODERATOR" | "CONTENT_EDITOR" | "TOURIST"
  // SUPERADMIN role cannot be assigned via this endpoint
}
```

Response 200:

```
{
  "userId": "uuid",
  "role": "MODERATOR"
}
```

[PATCH] `/admin/superadmin/users/{userId}/status` — *Suspend or reactivate a user*

Auth: SUPERADMIN

Request Body:

```
{
  "isActive": false    // false = suspend, true = reactivate
}
```

Response 200:

```
{
  "userId": "uuid",
  "isActive": false
}
```

13. Admin — Content Editor Endpoints

12.1 Place Management

[POST] `/admin/editor/places` — *Create a new place*

Auth: SUPERADMIN, CONTENT_EDITOR

Request Body:

```
{
  "name": "Kadifekale",
  "nameTr": "Kadifekale",
  "nameEn": "Velvet Castle",
  "category": "LANDMARK",
  "latitude": 38.4057,
  "longitude": 27.1445,
  "details": {
    "description": "M.Ö. 4. yüzyılda inşa edilmiş...",
    "address": "Kadifekale, İzmir"
  },
  "openingHours": {
    "monday": "08:00-20:00"
  }
}
```

Response 201: Full place object

[PUT] `/admin/editor/places/{placeId}` — *Update a place*

Auth: SUPERADMIN, CONTENT_EDITOR

Request Body: Same schema as POST `/admin/editor/places` (all fields optional)

Response 200: Updated place object

[DELETE] `/admin/editor/places/{placeId}` — *Soft-delete a place*

Auth: SUPERADMIN, CONTENT_EDITOR

Response 200:

```
{
  "placeId": "uuid",
  "isActive": false
}
```

12.2 Quest Management

[POST] `/admin/editor/quests` — *Create a new quest*

Auth: SUPERADMIN, CONTENT_EDITOR

Request Body:

```

{
  "title":      "İzmir Keşif Questi",
  "description": "İzmir'in ikonik noktalarını keşfet",
  "expReward":  500,
  "region":     "Konak, İzmir",
  "badgeId":    "uuid",    // optional – badge awarded on quest completion
  "steps": [
    {
      "order":      1,
      "placeId":    "uuid",
      "hint":       "Konak Meydanı'nda seni bekliyor",
      "requiresPhoto": true,
      "confidenceThreshold": 0.85 // optional, default 0.80, range 0.0-
1.0
    }
  ]
}

```

Response 201: Full quest object

[PUT] `/admin/editor/quests/{questId}` — *Update a quest*

Auth: SUPERADMIN, CONTENT_EDITOR

Request Body: Same schema as POST (all fields optional)

Response 200: Updated quest object

[DELETE] `/admin/editor/quests/{questId}` — *Soft-delete a quest*

Auth: SUPERADMIN, CONTENT_EDITOR

Response 200:

```

{
  "questId": "uuid",
  "isActive": false
}

```

12.3 Route Management

[POST] `/admin/editor/routes` — *Create a new route*

Auth: SUPERADMIN, CONTENT_EDITOR

Request Body:

```

{
  "name":      "Kemeraltı Çarşısı Rotası",
  "description": "Tarihi çarşayı keşfet",
  "centerLat": 38.4126,
  "centerLng": 27.1384,
  "radiusMeters": 800,
  "expReward": 300,
  "stops": [
    {

```

```
    "order":          1,
    "placeId":        "uuid",
    "estimatedMinutes": 15,
    "notes":          "Dıştan mutlaka bir fotoğraf çek"
  }
]
}
```

Response 201: Full route object

[PUT] `/admin/editor/routes/{routeId}` — *Update a route*

Auth: SUPERADMIN, CONTENT_EDITOR

Request Body: Same schema as POST (all fields optional)

Response 200: Updated route object

[DELETE] `/admin/editor/routes/{routeId}` — *Soft-delete a route*

Auth: SUPERADMIN, CONTENT_EDITOR

Response 200:

```
{
  "routeId": "uuid",
  "isActive": false
}
```

14. Admin — Moderator Endpoints

13.1 Review Moderation

[GET] `/admin/moderator/reviews` — *List reviews pending moderation*

Auth: SUPERADMIN, MODERATOR

Query Parameters:

Field	Description
status	string — PENDING APPROVED REJECTED (default: PENDING)
page	int — Page number, default 0
size	int — Page size, default 20

Response 200:

```
{
  "content": [
    {
      "reviewId": "uuid",
      "rating": 2,
      "comment": "Kötü bir deneyim...",
      "status": "PENDING",
      "user": {
        "userId": "uuid",
        "firstName": "Ali"
      },
      "place": {
        "placeId": "uuid",
        "name": "Saat Kulesi"
      },
      "createdAt": "2025-01-10T09:00:00Z"
    }
  ],
  "totalElements": 14
}
```

[PATCH] `/admin/moderator/reviews/{reviewId}` — *Approve or reject a review*

Auth: SUPERADMIN, MODERATOR

Request Body:

```
{
  "status": "APPROVED" | "REJECTED",
  "moderatorNote": "Contains inappropriate content" // optional
}
```

Response 200:

```
{
  "reviewId": "uuid",
}
```

```
"status":    "REJECTED"
}
```

13.2 Quest Verification Queue

[GET] `/admin/moderator/verifications` — *List pending quest step verifications*

Auth: SUPERADMIN, MODERATOR

Query Parameters:

Field	Description
status	string — PENDING APPROVED REJECTED (default: PENDING)
page	int — Page number, default 0
size	int — Page size, default 20

Response 200:

```
{
  "content": [
    {
      "verificationId": "uuid",
      "status":        "PENDING",
      "confidenceScore": 0.63,
      "photoUrl":      "https://minio.../verify.jpg",
      "submittedAt":   "2025-01-10T09:00:00Z",
      "user": {
        "userId":      "uuid",
        "firstName":   "Ali"
      },
      "quest": {
        "questId": "uuid",
        "title":   "İzmir Keşif Questi"
      },
      "step": {
        "stepId": "uuid",
        "place": {
          "name":      "Saat Kulesi",
          "latitude":  38.4192,
          "longitude": 27.1287
        }
      }
    }
  ],
  "totalElements": 6
}
```

[PATCH] `/admin/moderator/verifications/{verificationId}` — *Approve or reject a quest verification*

Auth: SUPERADMIN, MODERATOR

Request Body:

```
{
  "status": "APPROVED" | "REJECTED",
  "moderatorNote": "Photo does not match the location" // optional
}
```

Response 200:

```
{
  "verificationId": "uuid",
  "status":          "REJECTED",
  "expAwarded":      0
}
```

15. Standard Error Response Format

All error responses follow a consistent structure handled by `GlobalExceptionHandler`:

```
{
  "timestamp": "2025-01-10T09:00:00Z",
  "status":    400,
  "error":     "BAD_REQUEST",
  "message":   "Validation failed",
  "path":      "/api/v1/places/nearby",
  "details": [
    "lat: must not be null",
    "lng: must not be null"
  ]
}
```

HTTP Status	Error Code	When
400	BAD_REQUEST	Invalid request body or missing required fields
401	UNAUTHORIZED	Missing or expired JWT token
403	FORBIDDEN	Valid token but insufficient role for this endpoint
403	OUTSIDE_PILOT_ZONE	Image recognition requested outside active pilot regions
404	NOT_FOUND	Requested resource does not exist or is soft-deleted
409	CONFLICT	Duplicate resource (e.g., email already registered)
413	FILE_TOO_LARGE	Uploaded file exceeds 5MB limit
422	UNPROCESSABLE	GPS check failed for quest verification
500	INTERNAL_ERROR	Unexpected server error
503	AI_UNAVAILABLE	Python microservice unreachable and all fallbacks exhausted