**Attendance API Documentation**

Base path: /api/attendance/

✅ Endpoint 1: Create and List Attendance Sessions

URL: /api/attendance/sessions/

Methods: GET, POST

View Class: AttendanceSessionViewSet

Permissions:

• IsschoolAdmin (Full access)

• ISteacher (Full access)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

▶️ GET Request (List Sessions)

• Description: Lists attendance sessions. Each session represents one class on one date.

• Filters Available: class (UUID), date (YYYY-MM-DD) e.g., ?class=<uuid>&date=2025-08-21

• Pagination: Enabled (50 per page by default; page\_size query param, max 200)

Example Response (items):

[

  {

    "id": 123,

    "class\_obj": "uuid-of-class",

    "date": "2025-08-21",

    "taken\_by\_id": 42,

    "taken\_by\_role": "teacher"

  }

]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

▶️ POST Request (Create a Session and Records)

• Automatically sets: taken\_by (current user) and taken\_by\_role ("school\_admin" or "teacher").

• Business Rule: Frontend should send absentees only in records; all other enrolled students are marked present automatically.

• Validation:

  - Requires class and date.

  - Unique per class and date. Creating a duplicate returns 400.

  - Class enrollment is derived via StudentClass mapping.

Sample Payload:

{

  "class": "uuid-of-class",

  "date": "2025-08-21",

  "records": [

    { "student": "uuid-of-student-1", "status": "absent" },

    { "student": "uuid-of-student-2", "status": "absent" }

  ]

}

Successful Response (session with generated records):

{

  "id": 124,

  "class\_obj": "uuid-of-class",

  "date": "2025-08-21",

  "taken\_by\_id": 42,

  "taken\_by\_role": "teacher",

  "records": [

    {

      "id": 1,

      "student": "uuid-of-student-1",

      "student\_name": "Jane Doe",

      "admission\_number": "A-1001",

      "status": "absent",

      "marked\_at": "2025-08-21T09:30:00Z"

    },

    {

      "id": 2,

      "student": "uuid-of-student-3",

      "student\_name": "John Smith",

      "admission\_number": "A-1002",

      "status": "present",

      "marked\_at": "2025-08-21T09:30:00Z"

    }

  ]

}

Error Responses:

- 400 { "detail": "class and date are required." }

- 400 { "detail": "Attendance for this class and date already exists." }

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

✅ Endpoint 2: Retrieve Session Details

URL: /api/attendance/sessions/<pk>/

Methods: GET

View Class: AttendanceSessionViewSet (retrieve)

Permissions: IsschoolAdmin, ISteacher (Read)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

▶️ GET Request

• Description: Retrieves a single session with all records for that class/date.

Example Response:

{

  "id": 124,

  "class\_obj": "uuid-of-class",

  "date": "2025-08-21",

  "taken\_by\_id": 42,

  "taken\_by\_role": "teacher",

  "records": [

    {

      "id": 1,

      "student": "uuid-of-student-1",

      "student\_name": "Jane Doe",

      "admission\_number": "A-1001",

      "status": "absent",

      "marked\_at": "2025-08-21T09:30:00Z"

    }

  ]

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

✅ Endpoint 3: List Records in a Session (Nested)

URL: /api/attendance/sessions/<pk>/records/

Methods: GET

View Class: AttendanceRecordViewSet (list)

Permissions: IsschoolAdmin, ISteacher (Read)

• Pagination: Enabled (50 per page by default)

Example Response (items):

[

  {

    "id": 1,

    "student": "uuid-of-student-1",

    "student\_name": "Jane Doe",

    "admission\_number": "A-1001",

    "status": "absent",

    "marked\_at": "2025-08-21T09:30:00Z"

  }

]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

✅ Endpoint 4: Retrieve/Update/Delete a Record (Nested)

URL: /api/attendance/sessions/<session\_pk>/records/<pk>/

Methods: GET, PATCH, DELETE (PUT also supported)

View Class: AttendanceRecordViewSet

Permissions: IsschoolAdmin (Full), ISteacher (Full)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

▶️ GET Request

• Description: Retrieves a single attendance record within a session.

Example Response:

{

  "id": 1,

  "student": "uuid-of-student-1",

  "student\_name": "Jane Doe",

  "admission\_number": "A-1001",

  "status": "absent",

  "marked\_at": "2025-08-21T09:30:00Z"

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

✏️ PATCH Request (Update a Record)

• Allows updates to: status ("present" | "absent").

Sample Payload (PATCH):

{

  "status": "present"

}

Successful Response:

{

  "id": 1,

  "student": "uuid-of-student-1",

  "student\_name": "Jane Doe",

  "admission\_number": "A-1001",

  "status": "present",

  "marked\_at": "2025-08-21T09:30:00Z"

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

❌ DELETE Request

• Description: Deletes a single attendance record within the session.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Notes

• taken\_by\_role values: "school\_admin" or "teacher" based on the authenticated user’s role.

• Records are generated in bulk within a transaction when creating a session.

• Nested routing is used, powered by drf-nested-routers.

• Swagger is available at /swagger/ and documents the POST request body with records as an array of objects.