create the openapi spec for this java controller:

```
  @Path("/{id}/appointments")
  @GET
  @Produces(MediaType.APPLICATION_JSON)
  public List<AppointmentUser> getAppointmentsForUser(@PathParam("id") Long userId) {
    return userService.getAppointmentsForUser(userId);
  }

  @Path("/{id}/appointments/{appid}")
  @DELETE
  @Produces(MediaType.APPLICATION_JSON)
  public Response getAppointmentsForUser(@PathParam("id") Long userId,
@PathParam("appid") Long appointmentId) {
    userService.removeAppointment(userId, appointmentId);
    return Response.noContent().build();
  }

  @Path("/{id}/appointments/{appid}")
  @POST
  @Produces(MediaType.APPLICATION_JSON)
  public Response inviteUserToAppointment(@PathParam("id") Long userId,
@PathParam("appid") Long appointmentId) {
    AppointmentUser appointmentUser = userService.inviteUserToAppointment(userId,
appointmentId);
    return Response.accepted(appointmentUser).build();
  }

  @Path("/{id}/appointments/{appid}/status/{status}")
  @PUT
  @Produces(MediaType.APPLICATION_JSON)
  public Response updateAppointmentStatus(@PathParam("id") Long userId,
@PathParam("appid") Long appointmentId, @PathParam("status")AppointmentStatus
status) {
    AppointmentUser appointmentUser = userService.updateAppointmentStatus(userId,
appointmentId, status);
    return Response.accepted(appointmentUser).build();
  }
```

ChatGPT

Certainly! Below is the OpenAPI specification for your Java controller:

yaml

Copy code

```
openapi: 3.0.0
info:
  title: User Appointments API
  version: 1.0.0
paths:
  /{id}/appointments:
    get:
      summary: Get Appointments for User
      parameters:
        - name: id
          in: path
          description: User ID
          required: true
          schema:
            type: integer
      responses:
        '200':
          description: Successful response
          content:
            application/json:
              example: |
                [
                  {
                    "appointmentId": 1,
                    "appointmentDetails": "Appointment details"
                  },
                  {
                    "appointmentId": 2,
                    "appointmentDetails": "Another appointment details"
                  }
                ]
  /{id}/appointments/{appid}:
    delete:
      summary: Remove Appointment
      parameters:
        - name: id
          in: path
          description: User ID
          required: true
          schema:
            type: integer
        - name: appid
          in: path
          description: Appointment ID
          required: true
          schema:
            type: integer
      responses:
        '204':
          description: No Content
    post:
      summary: Invite User to Appointment
      parameters:
        - name: id
```

```
          in: path
          description: User ID
          required: true
          schema:
            type: integer
        - name: appid
          in: path
          description: Appointment ID
          required: true
          schema:
            type: integer
      responses:
        '202':
          description: Accepted
          content:
            application/json:
              example: |
                {
                  "appointmentId": 3,
                  "appointmentDetails": "New appointment details"
                }
  /{id}/appointments/{appid}/status/{status}:
    put:
      summary: Update Appointment Status
      parameters:
        - name: id
          in: path
          description: User ID
          required: true
          schema:
            type: integer
        - name: appid
          in: path
          description: Appointment ID
          required: true
          schema:
            type: integer
        - name: status
          in: path
          description: Appointment Status
          required: true
          schema:
            type: string
            enum: ["SCHEDULED", "CANCELLED", "COMPLETED"]
      responses:
        '202':
          description: Accepted
          content:
            application/json:
              example: |
                {
                  "appointmentId": 3,
                  "appointmentDetails": "Updated appointment details",
                  "status": "SCHEDULED"
                }
```

This OpenAPI spec defines four endpoints based on your Java controller methods. It includes details about the path parameters, HTTP methods, request and response formats, and example responses. Make sure to adapt it according to your specific requirements and any additional details you may need in your API documentation.