```yaml
openapi: 3.0.0
servers:
  - url: http://localhost:3000
info:
  version: 1.0.0
  title: APIs
  description: ''
  contact: {}
paths:
  /api/v1/auth/signup/:
    post:
      summary: Sign up for new user
      operationId: signup
      tags:
        - Auth
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/signupSchema'
      responses:
        '200':
          description: <h2>important</h2> JWT must contains userId and email in payload
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/AuthSuccessfullResponse'
        '400':
          description: Password is not valid, Email is already exist
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'

  /api/v1/auth/login/:
    post:
      summary: Login
      operationId: login
      tags:
        - Auth
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/loginSchema'
      responses:
        '200':
          description: <h2>important</h2> JWT must contains userId and email in payload
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/AuthSuccessfullResponse'
        '400':
          description: Authentication failed!
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
  /api/v1/groups/:
    get:
      summary: Get groups
      operationId: getGroups
      tags:
        - Group
      security:
        - ApiKeyAuth: []
      responses:
        '200':
          description: Groups list
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/GroupListResponse'
    post:
      summary: Create a group
      operationId: createGroup
      tags:
        - Group
      security:
        - ApiKeyAuth: []
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/CreateGroupSchema'
      responses:
        '200':
          description: Group successfully created!
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/GroupCreationSuccessfullResponse'
        '400':
          description: creation failed! (User is already member of a group)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
  /api/v1/groups/my/:
    get:
      summary: Get detailed information of group of user
      operationId: myGroup
      tags:
        - Group
      security:
        - ApiKeyAuth: []
      responses:
        '200':
          description: Group
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/GroupResponse'
        '400':
          description: Fetch failed (User is not joined to any group yet)!
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
  /api/v1/join_requests/:
    get:
      summary: Get join requests of user
      operationId: getJoinRequestsOfUser
      tags:
        - Join Request
      security:
        - ApiKeyAuth: []
      responses:
        '200':
          description: >-
            List of user's join requests. (Newest first)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/JoinRequestListResponse'
    post:
      summary: Send join request to a group
      operationId: createJoinRequest
      tags:
        - Join Request
      security:
        - ApiKeyAuth: []
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/SendJoinRequestSchema'
      responses:
        '200':
          description: Request has been sent!
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/OperationSuccessfullResponse'
        '400':
          description: Request failed! (User is already member of a group)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
  /api/v1/join_requests/group/:
    get:
      summary: Get join requests of user's group
      operationId: getJoinRequestsOfGroup
      tags:
        - Join Request
      security:
        - ApiKeyAuth: []
      responses:
        '200':
          description: >-
            List of requests to the group (Newest first)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/JoinRequestListResponse'
        '400':
          description: Fetch failed (User doesn't have essential permissions)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
  /api/v1/join_requests/accept/:
    post:
      summary: Accept a join request to user's group
      operationId: accpetJoinRequest
      tags:
        - Join Request
      security:
        - ApiKeyAuth: []
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/AcceptJoinRequestSchema'
      responses:
        '200':
          description: User joined the group
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/OperationSuccessfullResponse'
        '400':
          description: >-
            Acceptance failed! (User is already member of a group or acceptor
            doesn't have right permissions to accept request)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
  /api/v1/connection_requests/:
    get:
      summary: Get connection request of group
      operationId: getLinkRequestsOfGroup
      tags:
        - Connection Request
      security:
        - ApiKeyAuth: []
      responses:
        '200':
          description: List of connection requests to the group (Newest first)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/LinkRequestListResponse'
        '400':
          description: Fetch failed (User doesn't have essential permissions)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
    post:
      summary: Send a connection request
      operationId: sendLinkRequest
      tags:
        - Connection Request
      security:
        - ApiKeyAuth: []
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/ConnectionRequestSchema'
      responses:
        '200':
          description: Connection
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/OperationSuccessfullResponse'
        '400':
          description: Request doesn't delivered! (User doesn't have right permissions)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'
  /api/v1/connection_requests/accept:
    post:
      summary: Accept a connection request
      operationId: accpetLinkRequest
      tags:
        - Connection Request
      security:
        - ApiKeyAuth: []
      requestBody:
        required: true
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/ConnectionRequestSchema'
      responses:
        '200':
          description: Connection has been accepted
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/OperationSuccessfullResponse'
        '400':
          description: Request doesn't delivered! (User doesn't have right permissions)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'

  /api/v1/chats/:
    get:
      summary: get chats
      operationId: getChats
      tags:
        - Chat
      security:
        - ApiKeyAuth: []
      responses:
        '200':
          description: Chats list (newest first)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/ChatListResponse'
  /api/v1/chats/{user_id}/:
    get:
```

```
  summary: get messages in a chat
  operationId: getMessages
  tags:
    - Chat
  security:
    - ApiKeyAuth: []
  parameters:
    - name: user_id
      in: path
      description: user id
      required: true
      schema:
        type: integer
  responses:
    '200':
      description: Chat messages (last messages first)
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/MessagesListResponse'
    '400':
      description: bad request
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/BadRequestError'
post:
  summary: send a message to the user
  operationId: sendMessage
  tags:
    - Chat
  security:
    - ApiKeyAuth: []
  parameters:
    - name: user_id
      in: path
      description: user id
      required: true
      schema:
        type: integer
  requestBody:
    required: true
    content:
      application/json:
            schema:
              $ref: '#/components/schemas/SendMessageSchema'
      responses:
        '200':
          description: Message sent!
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/OperationSuccessfullResponse'
        '400':
          description: >-
            Message isn't delivered! (User is currently not allowed to send
            message to this user)
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/BadRequestError'

components:
  parameters: {}
  schemas:
    signupSchema:
      type: object
      properties:
        name:
          type: string
          example: Mohsen
        email:
          type: string
          example: test@gmail.com
        password:
          type: string
          example: 12345678@
    loginSchema:
      type: object
      properties:
        email:
          type: string
          example: test@gmail.com
        password:
          type: string
          example: 12345678@
    CreateGroupSchema:
      type: object
  properties:
    name:
      type: string
      example: Gank of four
    description:
      type: string
      example: We are here for design ...
SendJoinRequestSchema:
  type: object
  properties:
    groupId:
      example: 12
AcceptJoinRequestSchema:
  type: object
  properties:
    joinRequestId:
      example: 12
SendMessageSchema:
  type: object
  properties:
    message:
      type: string
ConnectionRequestSchema:
  type: object
  properties:
    groupId:
      type: integer
      example: 20
GroupResponse:
  type: object
  properties:
    group:
      type: object
      properties:
        name:
          type: string
          example: test
        description:
          type: string
          example: This is a test group
        members:
          type: array
          items:
            type: object
            example:
              id: 15
              name: Richard
              email: test@gmail.com
              rule: Owner
AuthSuccessfullResponse:
  type: object
  properties:
    token:
      type: string
      example: hello
    message:
      type: string
      example: successful
GroupCreationSuccessfullResponse:
  type: object
  properties:
    group:
      type: object
      properties:
        id:
          type: string
          example: 1
    message:
      type: string
      example: successful
GroupListResponse:
  type: object
  properties:
    groups:
      type: array
      items:
        type: object
        example:
          id: 1
          name: gank of four
          description: We are here for design ...
JoinRequestListResponse:
  type: object
  properties:
    joinRequests:
      type: array
      items:
        type: object
```

```yaml
        example:
          id: 14
          groupId: 10
          userId: 1
          date: 1646739337
ChatListResponse:
  type: object
  properties:
    chats:
      type: array
      items:
        type: object
        example:
          userId: 15
          name: test user
MessageResponse:
  type: object
  properties:
    message:
      type: string
      example: Hi
    date:
      type: integer
      example: 1646739337
    sentby:
      type: integer
      example: 10
MessagesListResponse:
  type: object
  properties:
    messages:
      type: array
      items:
        $ref: '#/components/schemas/MessageResponse'
LinkRequestListResponse:
  type: object
  properties:
    requests:
      type: array
      items:
        example:
          connectionRequestId: 12
          groupId: 14
          sent: 1646739337
    OperationSuccessfullResponse:
      type: object
      properties:
        message:
          type: string
          example: successful
    AuthDuplicateEmailError:
      type: object
      properties:
        error:
          type: object
          properties:
            enMessage:
              type: string
              example: email is already registered!
    AuthFailError:
      type: object
      properties:
        error:
          type: object
          properties:
            enMessage:
              type: string
              example: Email or password is not correct!
    BadRequestError:
      type: object
      properties:
        error:
          type: object
          properties:
            enMessage:
              type: string
              example: Bad request!
  securitySchemes:
    ApiKeyAuth:
      type: apiKey
      in: header
      name: Authorization
tags: []
```