

EAP: Architecture Specification and Prototype

A7: Web Resources Specification

The present artifact acts as an outline of the web resources that are going to be needed to implement the vertical prototype. Alongside, it specifies the permissions used in the modules to manage the access of information and resources.

1. Overview

Web Resource Module	Description
M01 Authentication and Individual Profile	Web resources associated with user authentication and individual profile management. Includes the following system features: login/logout, registration, credential recovery, view and edit personal profile information.
M02 User Administration	Web resources associated with user management, specifically: view and search users, delete or block/unblock user accounts, verify user account, view user information, and view system access details for each user.
M03 Content	Web resources associated with social media content. Includes the following system features: posts search, get user feed and notifications, get community feed and post (or comment) creation, editing or deletion.
M04 Content Review	Web resources associated with interaction between user and social-media content. Includes the following system features: post/comment promotion or reaction, post reporting, account reporting.
M05 User Relationship	Web resources associated with relationships between users, either individually or in community. Includes the following system features: sending/accepting/refusing a friend request, creating and editing a community, joining/leaving a community, inviting a member to the community.

2. Permissions

This section defines the permissions used in the modules to establish the conditions of access to resources.

Acronym	Title	Description
PUB	Public	Users without privileges
USR	User	Authenticated users
OWN	Owner	User that are owners of the information (e.g. own profile, own items)
GOWN	Group Owner	User that are owners of a given group
ADM	Administrator	System administrators

3. OpenAPI Specification

In this section is listed the [OpenAPI specification](#) that describes the server's web resources:

```

openapi: 3.0.0

info:
  version: '1.0'
  title: 'UniLinks Web API'
  description: 'Web Resources Specification (A7) for UniLinks'

servers:
  - url: https://lbaw.fe.up.pt
    description: Production server

security: []

externalDocs:
  description: Find more info here.
  url: https://git.fe.up.pt/lbaw/lbaw2223/lbaw2222/-/wikis/Home

tags:
  - name: 'M01'
    description: 'Authentication and Individual Profile'
  - name: 'M02'
    description: 'User Administration'
  - name: 'M03'
    description: 'Content'
  - name: 'M04'
    description: 'Content Review'
  - name: 'M05'
    description: 'User relationship'
  - name: 'M06'
    description: 'Static content'

components:
  securitySchemes:
    cookieuser:
      type: apiKey
      in: cookie
      name: SESSIONID      #cookie used to save session id

paths:
  ##### M1 Authentication and Individual Profile #####
  /login:
    get:
      operationId: R101

```

```

summary: 'R101: Login Form'
description: 'Provide login form. Access: PUB'
tags:
  - 'M01'
responses:
  '200':
    description: 'Ok. Show log-in UI'
post:
  operationId: R102
  summary: 'R102: Login Action'
  description: 'Processes login credentials (form input) Access: PUB'
  tags:
    - "M01"
  requestBody:
    required: true
    content:
      application/x-www-form-urlencoded:
        schema:
          type: object
          properties:
            email:
              type: string
            password:
              type: string
            _token:
              type: string
          required:
            - email
            - password
            - _token
  responses:
    '302':
      description: 'Redirect after processing the login credentials.'
      headers:
        Location:
          schema:
            type: string
          examples:
            302Success:
              description: 'Successful authentication. Redirect to user profile.'
              value: '/feed'
            302Error:
              description: 'Failed authentication. Redirect to login form.'
              value: '/login'
      Set-cookie:
        schema:
          type: string
          description: sessionID
          example: SESSIONID=abcde12345; Secure; Path=/; HttpOnly; SameSite=Strict

```

```

/logout:
  get:
    security:
      - cookieuser: []
    operationId: R103
    summary: 'R103: Logout Action'
    description: 'Logout the current authenticated user. Access: Authenticated-USER, ADMIN'
    tags:
      - 'M01'
    responses:
      '302':
        description: 'Redirect after processing logout.'
        headers:
          Location:

```

```
    schema:
      type: string
    examples:
      302Success:
        description: 'Successful logout. Redirect to main page'
        value: '/'
  Set-cookie:
    schema:
      type: string
      description: Delete sessionID
      example: SESSIONID=""; Max-Age=0; Secure; Path=/; HttpOnly; SameSite=Strict
```

/register:

get:

```
  operationId: R104
  summary: 'R104: Register Form'
  description: 'Provide new user registration form. Access: PUB'
  tags:
    - 'M01'
  responses:
    '200':
      description: 'Ok. Show sign-up UI'
```

post:

```
  operationId: R105
  summary: 'R105: Register Action'
  description: 'Processes the new user registration form submission. Access: PUB'
  tags:
    - 'M01'
```

requestBody:

required: true

content:

application/x-www-form-urlencoded:

schema:

type: object

properties:

username:

type: string

name:

type: string

email:

type: string

birthday:

type: string

privacy:

type: string

university:

type: string

course:

type: string

password:

type: string

_token:

type: string

required:

- username
- name
- email
- password
- birthday
- university
- course
- privacy
- _token

responses:

'302':

description: 'Redirect after processing the new user information.'

headers:

Location:

schema:

type: string

examples:

302Success:

description: 'Successful authentication. Redirect to user profile.'

value: '/{account_tag}'

302Failure:

description: 'Failed authentication. Redirect to main page.'

value: '/'

Set-cookie:

schema:

type: string

description: sessionID

example: SESSIONID=abcde12345; Secure; Path=/; HttpOnly; SameSite=Strict

/endregistration:

get:

operationId: R114

summary: 'R114: Extra information register form'

description: 'Provide registration form with fields for extra information (name, privacy, location, description). Access: PUB'

tags:

- 'M01'

responses:

'200':

description: 'Ok. Show extra information form UI'

post:

operationId: R113

summary: 'R113: End Registration Action'

description: 'Processes the extra information to complete user registration (name, privacy, location, description). Access: PUB'

tags:

- 'M01'

requestBody:

required: true

content:

application/x-www-form-urlencoded:

schema:

type: object

properties:

name:

type: string

privacy:

type: string

location:

type: string

description:

type: string

_token:

type: string

required:

- _token

responses:

'302':

description: 'Redirect after processing the extra information register.'

headers:

Location:

schema:

```

        type: string
    examples:
        302Success:
            description: 'Successful authentication. Redirect to timeline.'
            value: '/timeline'
        302Failure:
            description: 'Failed authentication. Redirect to main page.'
            value: '/'

/recover_password:
  get:
    operationId: R106
    summary: 'R106: Recover Password Form'
    description: 'Provide un-authenticated user registration form. Access: PUB'
    tags:
      - 'M01'
    responses:
      '200':
        description: 'Ok. Show recover-password UI'
  post:
    operationId: R107
    summary: 'R107: Recover Password Action'
    description: 'Processes recover account credentials (email input) Access: PUB'
    tags:
      - "M01"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              email:
                type: string
              _token:
                type: string
            required:
              - email
              - _token
    responses:
      '302':
        description: 'Redirect after processing the recover account credentials.'
        headers:
          Location:
            schema:
              type: string
            examples:
              302Success:
                description: 'There is an account with that email. Redirect to security-email-code authentication and password change form.'
                value: '/recover_password/change_password'
              302Error:
                description: 'There is no account with that email. Redirect to main page.'
                value: '/'
        Set-cookie:
          schema:
            type: string
            description: Save Recover Email
            example: recoverEmail=up202005454@fe.up.pt; Secure; Path=/; HttpOnly; SameSite=Strict
/recover_password/change_password:

```

```

get:
  operationId: R108
  summary: 'R108: Recover Password Part 2 Action'
  description: 'Provide user form for account recovery token and new password. Access: PUB'
  tags:
    - 'M01'
  parameters:
    - in: cookie
      name: recoverEmail
      schema:
        type: string

  responses:
    '200':
      description: 'Ok. Show recovery email-token and new-password UI'
post:
  operationId: R109
  summary: 'R109: Recover Password Part 2 Action'
  description: 'Check email token and, if authentication is successful, change user password to new
password Access: PUB'
  tags:
    - "M01"
  requestBody:
    required: true
    content:
      application/x-www-form-urlencoded:
        schema:
          type: object
          properties:
            email:
              type: string
            _token:
              type: string
              #csrf token
            recovery_code:
              type: string
              #email token
            new_password:
              type: string
          required:
            - email
            - _token
            - recovery_code
            - new_password
  responses:
    '302':
      description: 'Redirect after processing the recovery credentials.'
      headers:
        Location:
          schema:
            type: string
      examples:
        302Success:
          description: 'Email recovery code validation and password change successful. Redirect
to main-page.'
          value: '/'
        302Error:
          description: 'Error changing passwords. Redirect to main-page.'
          value: '/'

/{account_tag}:
get:
  security:
    - cookieuser: [] #administrator can see any account
  operationId: R110
  summary: 'R110: View user profile'

```

```
description: 'Show the individual user profile. Access: Authenticated-USR, ADM'
tags:
  - 'M01'
parameters:
  - in: path
    name: account_tag
    schema:
      type: string
    required: true
responses:
  '200':
    description: 'Ok. Show view profile UI'
  '302':
    description: 'Non-existent/Private profile'
    headers:
      Location:
        schema:
          type: string
    examples:
      302Error:
        description: 'Forbidden profile. Redirect to homepage'
        value: '/'
```

```
/{account_tag}/edit:
  get:
    security:
      - cookieuser: [] #administrator can edit any profile
    operationId: R111
    summary: 'R111: Edit user profile Form'
    description: 'Get edit profile form. Access: Authenticated-USR, ADM'
    tags:
      - 'M01'
    parameters:
      - in: path
        name: account_tag
        schema:
          type: string
        required: true
    responses:
      '200':
        description: 'Ok. Show edit profile form'
  post:
    operationId: R112
    security:
      - cookieuser: []
    summary: 'R112: Edit profile'
    description: 'Change profile parts. Access: Authenticated-USR, ADM'
    tags:
      - "M01"
    requestBody:
      required: true
      content:
        multipart/form-data: # necessary for photos
        schema:
          properties:
            new_username:
              type: string
            new_name:
              type: string
            new_email:
              type: string
            new_birthday:
              type: string
            new_privacy:
```



```

        type: string
    new_university:
        type: string
    new_course:
        type: string
    new_password:
        type: string
    new_description:
        type: string
    new_location:
        type: string
    new_pronouns:
        type: string
    _token:          #csrf
        type: string
    image:           #profile_picture
        type: string
        format: binary

    required:
        - _token

```

responses:

```

'302':
    description: 'Redirect after processing the recovery credentials.'
    headers:
        Location:
            schema:
                type: string
    examples:
        302Success:
            description: 'Profile change successful. Redirect to profile.'
            value: '/{account_tag}'
        302Error:
            description: 'Error changing profile. Redirect to profile.'
            value: '/{account_tag}'

```

M2 User Administration

/user_search:

```

    get:
        security:
            - cookieuser: []
        operationId: R201
        summary: 'R201: Search for users tags'
        description: 'Search and list user tags. Access: Authenticated-USR, ADM'
        tags:
            - "M02"
        parameters:
            - in: query
              name: account_tag
              schema:
                  type: string
              required: true

```

responses:

```

'200':
    description: 'Ok. Show users with that user_tag (or part)' #admins can search all users

```

/account_tag/action:

```

    post:
        security:
            - cookieuser: [] #only admin can delete or unblock/block any account (user can only delete his own account)
        operationId: R202
        summary: 'R202: Block/Unblock/Delete Account'

```

```

description: 'Only admin can delete or unblock/block any account and user can only delete his own
account. Access: Authenticated-USR, ADM'
tags:
  - "M02"
requestBody:
  required: true
  content:
    application/x-www-form-urlencoded:
      schema:
        type: object
        properties:
          operation:
            type: string
            pattern: ^(block)|(unblock)|(delete)$
          _token:
            type: string
        required:
          - operation
          - _token

responses:
  '302':
    description: 'Redirect after processing the action.'
    headers:
      Location:
        schema:
          type: string
    examples:
      302UserDeleteSuccess:
        description: 'Your account was delleted successfully. Redirect to logout'
        value: '/logout'
      302AdminDeleteSuccess:
        description: 'User account was delleted successfully. Redirect to usersearch'
        value: '/usersearch?account_tag='
      302Admin(Un)BlockSuccess:
        description: 'User account block action successfull. Redirect to user profile'
        value: '/{account_tag}'
      302Error:
        description: 'Error. Redirect to main-page.'
        value: '/'

```

M3 Content

```

/post_content_search:
  get:
    security:
      - cookieuser: []
    operationId: R301
    summary: 'R301: Search for posts content'
    description: 'Search and list posts according to text search. Access: Authenticated-USR, ADM'
    tags:
      - "M03"
    parameters:
      - in: header
        name: text_search
        schema:
          type: string
        required: true
    responses:
      '200':
        description: 'Ok. Show posts with that content (or part)' #admins can search all posts

/feed:
  get:
    security:

```

```

    - cookieuser: []
  operationId: R302
  summary: 'R302: View Feed'
  description: 'Show the current logged user's feed. Access: Authenticated-USR, ADM'
  tags:
    - 'M03'
  responses:
    '200':
      description: 'Ok. Show feed'

/new_post:
  get:
    operationId: R303
    security:
      - cookieuser: []
    summary: 'R303: New Post Form'
    description: 'Provide new post form. Access: Authenticated-USR, ADM'
    tags:
      - 'M03'
    responses:
      '200':
        description: 'Ok. Show new post UI'
  post:
    operationId: R304
    security:
      - cookieuser: []
    summary: 'R304: Create Post'
    description: 'Create new post. Access: Authenticated-USR, ADM'
    tags:
      - "M03"
    requestBody:
      required: true
      content:
        multipart/form-data: # necessary for photos
          schema:
            properties:
              _token: #csrf
                type: string
              description:
                type: string
              parent_post:
                type: integer
              is_visible:
                type: boolean
              group_id:
                type: integer
              image1: #pic1
                type: string
                format: binary
              image2: #pic2
                type: string
                format: binary
              image3: #pic3
                type: string
                format: binary
              image4: #pic4
                type: string
                format: binary #max 4 pictures
            required:
              - _token
              - description
              - is_visible

  responses:

```

```

    '302':
      description: 'Redirect after creating post.'
      headers:
        Location:
          schema:
            type: string
      examples:
        302Success:
          description: 'Post creation successful. Redirect to post page.'
          value: '/{post_id}'
        302Error:
          description: 'Error creating post. Redirect to timeline.'
          value: '/feed'

/{post_id}:
  get:
    security:
      - cookieuser: []
    operationId: R305
    summary: 'R305: View Post'
    description: 'Show the post with path id. Access: Authenticated-USR, ADM'
    tags:
      - 'M03'
    parameters:
      - in: path
        name: post_id
        schema:
          type: integer
        required: true
    responses:
      '200':
        description: 'Ok. Show post'

/{post_id}/edit:
  post:
    operationId: R306
    security:
      - cookieuser: []
    summary: 'R306: Edit Post'
    description: 'Edit post. Access: Authenticated-USR, ADM'
    tags:
      - "M03"
    requestBody:
      required: true
      content:
        multipart/form-data: # necessary for photos
          schema:
            properties:
              _token: #csrf
                type: string
              description:
                type: string
              is_visible:
                type: boolean
              image1: #pic1
                type: string
                format: binary
              image2: #pic2
                type: string
                format: binary
              image3: #pic3
                type: string
                format: binary
              image4: #pic4

```

```

        type: string
        format: binary    #max 4 pictures
    required:
        - _token

responses:
    '302':
        description: 'Redirect after editing post.'
        headers:
            Location:
                schema:
                    type: string
        examples:
            302Success:
                description: 'Post edit successful. Redirect to post page.'
                value: '/{post_id}'
            302Error:
                description: 'Error editing post. Redirect to timeline.'
                value: '/feed'

```

/ {post_id} /delete:

```

post:
    operationId: R307
    security:
        - cookieuser: []
    summary: 'R307: Delete Post'
    description: 'Delete post. Access: Authenticated-USR, ADM'
    tags:
        - "M03"
    requestBody:
        required: true
        content:
            application/x-www-form-urlencoded:
                schema:
                    type: object
                    properties:
                        _token:    #csrf
                            type: string
                    required:
                        - _token

```

```

responses:
    '302':
        description: 'Redirect after editing post.'
        headers:
            Location:
                schema:
                    type: string
        examples:
            302Success:
                description: 'Post delete successful. Redirect to profile.'
                value: '/{account_tag}'
            302Error:
                description: 'Error deleting post. Redirect to post.'
                value: '/{post_id}'

```

M4 Content Review

/ {post_id} /react:

```

post:
    operationId: R401
    security:
        - cookieuser: []
    summary: 'R401: React Post'
    description: 'React Post Action. Access: Authenticated-USR, ADM'

```

```
tags:
  - "M04"
requestBody:
  required: true
  content:
    application/x-www-form-urlencoded:
      schema:
        type: object
        properties:
          _token: #csrf
            type: string
          up_vote:
            type: boolean
          remove:
            type: boolean #to remove reaction
        required:
          - _token
          - up_vote
```

```
responses:
  '302':
    description: 'Redirect after editing post.'
    headers:
      Location:
        schema:
          type: string
    examples:
      302Success:
        description: 'Post delete successful. Redirect to profile.'
        value: '/{account_tag}'
      302Error:
        description: 'Error deleting post. Redirect to post.'
        value: '/{post_id}'
```

```
/{post_id}/promotion:
  post:
    operationId: R402
    security:
      - cookieuser: []
    summary: 'R402: Promote Post'
    description: 'Promote Post Action. Access: Authenticated-USR, ADM'
    tags:
      - "M04"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              _token: #csrf
                type: string
              remove:
                type: boolean #to remove promotion
            required:
              - _token
```

```
responses:
  '302':
    description: 'Redirect after editing post.'
    headers:
      Location:
        schema:
          type: string
```

```
examples:
  302Success:
    description: 'Post delete successful. Redirect to profile.'
    value: '/{account_tag}'
  302Error:
    description: 'Error deleting post. Redirect to post.'
    value: '/{post_id}'
```

```
/{post_id}/report:
  post:
    operationId: R403
    security:
      - cookieuser: []
    summary: 'R403: Report Post'
    description: 'Report Post Action. Access: Authenticated-USR, ADM'
    tags:
      - "M04"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              _token: #csrf
                type: string
              description:
                type: string
              reason:
                type: string
            required:
              - _token
              - reason
    responses:
      '302':
        description: 'Redirect after reporting post.'
        headers:
          Location:
            schema:
              type: string
        examples:
          302Success:
            description: 'Post report successful. Redirect to post.'
            value: '/{post_id}'
          302Error:
            description: 'Post report unsuccessful. Redirect to post.'
            value: '/{post_id}'
```

```
/{account_tag}/report:
  post:
    operationId: R404
    security:
      - cookieuser: []
    summary: 'R404: Report Account'
    description: 'Report Account Action. Access: Authenticated-USR, ADM'
    tags:
      - "M04"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
```

```

    properties:
      _token:          #csrf
      type: string
    description:
      type: string
    reason:
      type: string
    required:
      - _token
      - reason

responses:
  '302':
    description: 'Redirect after reporting Account.'
    headers:
      Location:
        schema:
          type: string
    examples:
      302Success:
        description: 'Account report successful. Redirect to account.'
        value: '/{account_tag}'
      302Error:
        description: 'Account report unsuccessful. Redirect to account.'
        value: '/{account_tag}'

```

```

/notifications:
  get:
    security:
      - cookieuser: []
    operationId: R405
    summary: 'R405: View Notifications'
    description: 'Show user notifications. Access: Authenticated-USR, ADM'
    tags:
      - 'M04'
    responses:
      '200':
        description: 'Ok. Show notifications'

```

M5 User relationship

```

/{account_tag}/request:
  post:
    security:
      - cookieuser: []
    operationId: R501
    summary: 'R501: Send Friend Request'
    description: 'Send friend request action. Access: Authenticated-USR, ADM'
    tags:
      - "M05"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              _token:          #csrf
              type: string
            required:
              - _token
    responses:
      '302':
        description: 'Redirect after processing the friends request.'

```



```

    headers:
      Location:
        schema:
          type: string
        examples:
          302Success:
            description: 'The friend request was successfully done.'
            value: '/{account_tag}'
          302Error:
            description: 'The friend request went wrong.'
            value: '/{account_tag}'

/{account_tag}/remove_friend:
  post:
    security:
      - cookieuser: []
    operationId: R502
    summary: 'R502: Remove Friend '
    description: 'Remove friend action. Access: Authenticated-USR, ADM'
    tags:
      - "M05"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              _token:          #csrf
                type: string
            required:
              - _token
    responses:
      '302':
        description: 'Redirect after processing the friendship removal.'
        headers:
          Location:
            schema:
              type: string
            examples:
              302Success:
                description: 'The friendship removal was successfully done.'
                value: '/{account_tag}'
              302Error:
                description: 'The friendship removal went wrong.'
                value: '/{account_tag}'

/{account_tag}/accept_decline_request:
  post:
    security:
      - cookieuser: []
    operationId: R503
    summary: 'R503: Accept/Decline Friend Request'
    description: 'Accept/Decline Friend Request action. Access: Authenticated-USR, ADM'
    tags:
      - "M05"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              _token:          #csrf

```

```

        type: string
      operation:
        type: string
        pattern: ^(accept)|(decline)$
      required:
        - _token
        - operation
    responses:
      '302':
        description: 'Redirect after processing the acceptance/decline of a friend request.'
        headers:
          Location:
            schema:
              type: string
        examples:
          302Success:
            description: 'The friend request acceptance/decline was successfully done.'
            value: '/{account_tag}'
          302Error:
            description: 'The friend request acceptance/decline went wrong.'
            value: '/{account_tag}'

/create_group:
  get:
    security:
      - cookieuser: []
    operationId: R504
    summary: 'R504: Create group form'
    description: 'Provide login form. Access: Authenticated-USR, ADM'
    tags:
      - 'M05'
    responses:
      '200':
        description: 'Ok. Show create-group UI'
  post:
    security:
      - cookieuser: []
    operationId: R505
    summary: 'R505: Create group Action'
    description: 'Create a new group. Access: Authenticated-USR, ADM'
    tags:
      - "M05"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              name:
                type: string
              description:
                type: string
              is_public:
                type: boolean
              _token:
                type: string
            members:
              type: array
            items:
              type: object
              properties:
                id:
                  type: integer

```

```

        role:
          type: string
          pattern: ^(admin)|(pending)?
      required:
        - name
        - is_public
        - _token
        - members
  responses:
    '302':
      description: 'Redirect after processing the group creation.'
      headers:
        Location:
          schema:
            type: string
      examples:
        302Success:
          description: 'Successful group creation. Redirect to group_page.'
          value: '/group/{group_id}'
        302Error:
          description: 'Failed group creation. Redirect to user profile.'
          value: '/{account_tag}'
      Set-cookie:
        schema:
          type: string
          description: sessionID
          example: SESSIONID=abcde12345; Secure; Path=/; HttpOnly; SameSite=Strict

```

```

/group/{group_id}:
  get:
    security:
      - cookieuser: []
    operationId: R506
    summary: 'R506: View Group Feed (and members)'
    description: 'Show the group feed. Access: Authenticated-USR, ADM'
    tags:
      - 'M05'
    responses:
      '200':
        description: 'Ok. Show group feed'

```

```

/group/{group_id}/request:
  post:
    security:
      - cookieuser: []
    operationId: R507
    summary: 'R507: Send Group Member Request'
    description: 'Send group member request action. Access: Authenticated-USR, ADM'
    tags:
      - "M05"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              _token: #csrf
                type: string
              member_id:
                type: integer
            required:
              - _token
              - member_id

```

```
responses:
  '302':
    description: 'Redirect after processing the group member request.'
    headers:
      Location:
        schema:
          type: string
    examples:
      302Success:
        description: 'The group member request was successfully done.'
        value: '/group/{group_id}'
      302Error:
        description: 'The group member request went wrong.'
        value: '/group/{group_id}'
```

```
/group/{group_id}/remove_member:
post:
  security:
    - cookieuser: []
  operationId: R508
  summary: 'R508: Remove Member '
  description: 'Remove member action. Access: Authenticated-USR, ADM'
  tags:
    - "M05"
  requestBody:
    required: true
    content:
      application/x-www-form-urlencoded:
        schema:
          type: object
          properties:
            _token: #csrf
              type: string
            member_id:
              type: integer
          required:
            - _token
            - member_id
```

```
responses:
  '302':
    description: 'Redirect after processing the friendship removal.'
    headers:
      Location:
        schema:
          type: string
    examples:
      302Success:
        description: 'The member removal was successfully done.'
        value: '/group/{group_id}'
      302Error:
        description: 'The member removal went wrong.'
        value: '/group/{group_id}'
```

```
/group/{group_id}/edit:
post:
  security:
    - cookieuser: []
  operationId: R510
  summary: 'R510: Edit group '
  description: 'Edit group action. Access: Authenticated-USR, ADM'
  tags:
    - "M05"
  requestBody:
    required: true
```

```
content:
  application/x-www-form-urlencoded:
    schema:
      type: object
      properties:
        name:
          type: string
        description:
          type: string
        is_public:
          type: boolean
        _token:
          type: string
      required:
        - _token
responses:
  '302':
    description: 'Redirect after processing the group editing.'
    headers:
      Location:
        schema:
          type: string
    examples:
      302Success:
        description: 'The group editing was successfully done.'
        value: '/group/{group_id}'
      302Error:
        description: 'The group editing went wrong.'
        value: '/group/{group_id}'
```

```
/group/{group_id}/change_permissions:
  post:
    security:
      - cookieuser: []
    operationId: R511
    summary: 'R511: Change Member Permisison'
    description: 'Change Member Permisison action. Access: Group-Administrator, ADM'
    tags:
      - "M05"
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              _token: #csrf
              type: string
              member_id:
                type: integer
              admin:
                type: boolean
            required:
              - _token
              - member_id
              - admin
    responses:
      '302':
        description: 'Redirect after processing the member permissions changing.'
        headers:
          Location:
            schema:
              type: string
        examples:
```

```

302Success:
  description: 'The member permissions changing was successfully done.'
  value: '/group/{group_id}'
302Error:
  description: 'The member permissions changing went wrong.'
  value: '/group/{group_id}'

/{account_tag}/accept_decline_request_group:
post:
  security:
    - cookieuser: []
  operationId: R509
  summary: 'R509: Accept/Decline group member request'
  description: 'Accept/Decline group member request action. Access: Authenticated-USR, ADM'
  tags:
    - "M05"
  requestBody:
    required: true
    content:
      application/x-www-form-urlencoded:
        schema:
          type: object
          properties:
            _token:          #csrf
              type: string
            operation:
              type: string
              pattern: ^(accept)|(decline)$
          required:
            - _token
            - operation
  responses:
    '302':
      description: 'Redirect after processing the acceptance/decline of a group member request.'
      headers:
        Location:
          schema:
            type: string
      examples:
        302Success:
          description: 'The group member request acceptance/decline was successfully done.'
          value: '/group/{group_id}'
        302Error:
          description: 'The group member request acceptance/decline went wrong.'
          value: '/group/{group_id}'

##### M6 Content #####
/welcome:
get:
  operationId: R601
  summary: 'R601: Welcome page'
  description: 'Provide page with brief presentation of main features, About us section, Contacts s
ection, Ask for help section. Access: PUB'
  tags:
    - 'M06'
  responses:
    '200':
      description: 'Ok. Show new post UI'

/ads:
get:
  operationId: R602
  summary: 'R602: Ads'

```

```
description: 'Provide list of ads (related to specific university)'
tags:
  - 'M06'
parameters:
  - in: query
    name: university_name
    schema:
      type: string
    required: false
responses:
  '200':
    description: 'Ok. Show ads'
```

A8: Vertical prototype

In this artifact, the goal is to become familiarized with the technologies used to implement the project and to validate the previously presented architecture. The vertical prototype consists of an implementation of the project's features deemed necessary (labeled as high priority) and interfaces all the software layers, therefore standing as a barebone for latter stages in the development process.

1. Implemented Features

1.1. Implemented User Stories

User Story reference	Name	Priority	Description
US001	Log-In	High	As an <i>Unauthenticated User</i> I want to be able to authenticate into the social network so that I can have access to privileged information.
US002	Sign-Up	High	As an <i>Unauthenticated User</i> I want to be able to register myself into the social network so that I can authenticate and have access to privileged information.
US101	Search	High	As a <i>User</i> , I want to be able to search through the social network's users.
US102	View Profile	High	As a <i>User</i> , I want to be able to see a public profile.
US201	See Timeline	High	As an <i>Authenticated User</i> I want to be able to see my timeline, consisting of posts from my friends and the groups I belong to, as well as deciding its order.
US202	Create a Post	High	As an <i>Authenticated User</i> I want to be able to create a post.
US205	Log Out	High	As an <i>Authenticated User</i> I want to be able to log out.
US207	View Private Profile	High	As an <i>Authenticated User</i> , I want to be able to see a private profile if that user is my friend.
US402	View User Account	High	As a <i>Administrator</i> , I want to be able to view an user's account.
US401	Find User Account	High	As a <i>Administrator</i> , I want to be able to find an user's account.
US206	Edit Profile	High	As an <i>Authenticated User</i> I want to be able to edit my profile.
US405	Block and Unblock User Accounts	Medium	As a <i>Administrator</i> , I want to be able to block and unblock an user account.

1.2. Implemented Web Resources

Identify the web resources that were implemented in the prototype.

Module M01: Authentication and Individual Profile

Web Resource Reference	URL
R101: Login Form	GET /login
R102: Login Action	POST /login
R103: Logout Action	GET /logout
R104: Register Form	GET /register
R105: Register Action	POST /register
R114: Extra information register form	GET /endregistration
R113: End registration action	POST /endregistration
R110: View User Profile	GET /user/{account_tag}
R111: Edit User Profile Form	GET /user/{account_tag}/edit
R112: Edit Profile	POST /user/{account_tag}/edit

Module M02: User Administration

Web Resource Reference	URL
R201: Search user tags	GET /user_search
R202a: Block Account	POST /users/block
R202b: Unblock Account	POST /users/unblock

(R202a and R202b must be merged and url must be changed in the next phase)

(GET /users → lists all users (if you are admin) and enables blocking and unblocking through AJAX → must be merged into R201 in next phase)

Module M03: Content

Web Resource Reference	URL
R302: View Feed	GET /timeline
R303: Create Post	POST /new_post

2. Prototype

[Live Server Link](#)(TBD)

Credentials:\

admin\

user: conta@admin.pt\

password: 1234

regular user:\n
jorge@mail.pt\n
1234

[Link to the project's source code](#)

Revision history

GROUP2122, 21/11/2022

- Diogo Miguel Ferreira da Costa, [up202007770@edu.fe.up.pt](#) (Editor)
- André Ismael Ferraz Ávila, [up202006767@edu.fe.up.pt](#)
- Francisco Maria Lopes Pinto Pimentel Serra, [up202007723@edu.fe.up.pt](#)
- Manuel João Gomes Alves Amorim, [up202007485@edu.fe.up.pt](#)