EAP: Architecture Specification and Prototype

With the development of toEaseManage we intend to create a useful and acessible platform that helps users to organize projects as well as grow as web developers throughout the curricular unit. The website is for everyone, it can be used personally, for business, school, or even among friends.

A7: High-level architecture. Privileges. Web resources specification

This artefact describes the architecture of the web application to be developed, indicating the requests of data, the properties of each one, and the format of JSON responses. This specification uses the OpenAPI standard using YAML.

This artefact presents the documentation for toEaseManage, including the create, view, update and delete operations for each data resource.

1. Overview

Here we identify the modules that are part of our application. It's division was made taking into account the granularity and meaning of each one of the modules.

Module	Description
M01: Authentication and Individual Profile	Web features related to profile and managing a user's profile, such as login/logout and sign up actions, profile editing, password retrieval, and access to personal information.
M02: Project	Web resources associated with project interaction. Includes the following system features: create and delete, view, edit, and search projects.
M03: Tasks and Comments	Web features associated with project execution and details, as well as interaction with other project members. It includes the following system features: add/edit/comment tasks, plus you can edit or delete the comment made.
M04: Project Forum and Labels	Project Forum related web resources with the following system features: create, view, edit and delete forum messages. It is also possible to create new labels to use inside a project, to characterize tasks with one or more, and besides that being able to remove the label as well as delete it from the project.
M05: Invites and Notifications	Web resources associated to project invitations and notifications related to tasks, invitations and project modifications. It includes the following system features: create/list/accept/reject invitations and view/list notifications.
M06: User Administration	Web features associated with user management, specifically: viewing, searching and blocking users accounts. Plus, in a user's administration you can view system access details for each user.
M07: Static Pages	Web resources containning static content, such as information about the website and the developers of the website (about, contact, services).

There is an explaination about the division that was made mainly in modules M02, M03, M04. Since projects and tasks are the main big services of our project, and even if a task exists only inside projects, there are a lot of methods to handle with the requests proposed. Therefore, we come to the conclusion that they should be separated. Soon we understood that task comments should be agrouped within the tasks modules, since they exist due to the existence of tasks (MO3). However, we remained with a very large module if we included the project forum and project labels inside the project module. As they belong to a project we decided to create other module that included the two of them (M04). Concluding the modules division with three modules more similar in size and more in context with what they are working with.

2. Permissions

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

PUB	Public	Visitors of the page.
AUTH_USR	Authenticated User	User that has logged in.
OWN	Owner	User that owns the information (e.g. forum messages, comments, profile).
COOR	Coordinator	User that coordinates an instance (e.g. project coordinators).
MEMB	Member	Relation of membership with a project.
ADMIN	Administrator	Administrators of the system.

3. OpenAPI Specification

This section includes the complete API specification in OpenAPI (YAML).

Additionally here it is a link to the OpenAPI YAML file in the group's repository.

Link to the Swagger generated documentation: https://app.swaggerhub.com/apis/toEase/a7_openapi/1.0.0-oas3. We did not had an account in Swagger sowe started free trial of the account, for only 15 days. We don't know if the link is going to still available after this day, so please contact us if there is something wrong.

```yaml
openapi: 3.0.0
info:
  version: 1.0.0-oas3
  title: LBAW toEaseManage Web AP
  description: Web Resources Specification (A7) for toEaseManage
servers:
  - url: http://lbaw.fe.up.pt
    description: Production server
tags:
  - name: 'M01: Authentication and Individual Profile'
  - name: 'M02: Project'
  - name: 'M03: Tasks and Comments'
  - name: 'M04: Project Forum and Labels'
  - name: 'M05: Invites and Notifications'
```

```yaml
      - name: 'M06: Administration'
      - name: 'M07: Static Pages'
paths:
  /login:
    get:
      operationId: R101
      summary: 'R101: Login Form'
      description: 'Provide login form. Access: PUB'
      tags:
        - 'M01: Authentication and Individual Profile'
      responses:
        '200':
          description: OK. Show Log-in UI
    post:
      operationId: R102
      summary: 'R102: Login Action'
      description: 'Processes the login form submission. Access: PUB'
      tags:
        - 'M01: Authentication and Individual Profile'
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                email:
                  type: string
                password:
                  type: string
                required:
                  - email
                  - password
      responses:
        '302':
          description: Redirect after processing the login credentials.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull authentication. Redirect to user
profile
                  value: /users/{id}
                302Error:
                  description: Failed authentication. Redirect to login form.
                  value: /login
  /logout:
    get:
      operationId: R103
      summary: 'R103: Logout Action'
      description: 'Logout the current authenticated user. Access: AUTH_USR'
      tags:
```

```yaml
            - 'M01: Authentication and Individual Profile'
      responses:
        '302':
          description: Redirect after processing logout.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successful logout. Redirect to login form.
                  value: /login
  /register:
    get:
      operationId: R104
      summary: 'R104: Register Form'
      description: 'Provide new user registration form. Access: PUB'
      tags:
        - 'M01: Authentication and Individual Profile'
      responses:
        '200':
          description: OK. Show Sign-Up UI
    post:
      operationId: R105
      summary: 'R105: Register Action'
      description: 'Processes the register form submission. Access: PUB'
      tags:
        - 'M01: Authentication and Individual Profile'
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                name:
                  type: string
                email:
                  type: string
                password:
                  type: string
                image:
                  type: string
                  format: binary
              required:
                - email
                - password
                - name
      responses:
        '302':
          description: Redirect after processing the new user information.
          headers:
            Location:
              schema:
```

```
                       type: string
                    examples:
                      302Success:
                        description: Successfull registration. Redirect to login page
                        value: /login
                      302Error:
                        description: Failed registration. Redirect to register form.
                        value: /register
  /users:
    get:
      operationId: R106
      summary: 'R106: View user page'
      description: 'Show the authenticated user page. Access: AUTH_USR'
      tags:
        - 'M01: Authentication and Individual Profile'
      responses:
        '200':
          description: Ok. Show User Page UI
  /users/{id}:
    delete:
      operationId: R107
      summary: 'R107: Delete user'
      description: 'Delete user from service. Access: ADMIN, OWN'
      tags:
        - 'M01: Authentication and Individual Profile'
        - 'M06: Administration'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      responses:
        '200':
          description: Ok. User deleted
  /users/{id}/profile/:
    get:
      operationId: R108
      summary: 'R108: View user profile '
      description: 'Show the individual user profile page. Access: AUTH_USR'
      tags:
        - 'M01: Authentication and Individual Profile'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      responses:
        '200':
          description: OK. Show User Profile UI
  /users/{id}/update:
    get:
      operationId: R109
```

```yaml
        summary: 'R109: User edit profile Form '
        description: 'Show the individual user edit profile page. Access: OWN'
        tags:
          - 'M01: Authentication and Individual Profile'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        responses:
          '200':
            description: OK. Show User Profile UI
      post:
        operationId: R110
        summary: 'R110: Edit user profile'
        description: 'Edit individual user page. Access: OWN'
        tags:
          - 'M01: Authentication and Individual Profile'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        requestBody:
          required: true
          content:
            application/x-www-form-urlencoded:
              schema:
                type: object
                properties:
                  name:
                    type: string
                  image:
                    type: string
                    format: binary
        responses:
          '200':
            description: Ok. User updated.
  /recoverPassword:
    get:
      operationId: R111
      summary: 'R111: Recover password Form'
      description: 'Provide recover password form. Access: PUB'
      tags:
        - 'M01: Authentication and Individual Profile'
      responses:
        '200':
          description: OK. Show recover password UI
    post:
      operationId: R112
      summary: 'R112: Recover password Action'
      description: 'Processes the recover password form submission. Access: PUB'
```

```
        tags:
          - 'M01: Authentication and Individual Profile'
        requestBody:
          required: true
          content:
            application/x-www-form-urlencoded:
              schema:
                type: object
                properties:
                  email:
                    type: string
                  required:
                    - email
        responses:
          '302':
            description: Redirect after processing the recover password.
            headers:
              Location:
                schema:
                  type: string
                examples:
                  302Success:
                    description: Recover email sent. Redirect to login
                    value: /login
                  302Error:
                    description: Failed authentication. Redirect to login form.
                    value: /recoverPassword
  /resetPassword:
    get:
      operationId: R113
      summary: 'R113: Reset password Form'
      description: 'Provide reset password form. Access: AUTH_USR'
      tags:
        - 'M01: Authentication and Individual Profile'
      responses:
        '200':
          description: OK. Show reset password UI
    post:
      operationId: R114
      summary: 'R114: Reset password Action'
      description: 'Processes the reset password form submission. Access:
AUTH_USR'
      tags:
        - 'M01: Authentication and Individual Profile'
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                email:
                  type: string
                password:
```

```
                    type: string
                required:
                  - email
                  - password
          responses:
            '302':
              description: Redirect after processing the reset password.
              headers:
                Location:
                  schema:
                    type: string
                  examples:
                    302Success:
                      description: Password reseted. Redirect to login
                      value: /login
                    302Error:
                      description: Failed authentication. Redirect to home page.
                      value: /
  /api/users:
    get:
      operationId: R115
      summary: 'R115: Search Users API'
      description: 'Searches for users and returns the results as JSON. Access:
AUTH_USR.'
      tags:
        - 'M01: Authentication and Individual Profile'
      parameters:
        - in: query
          name: search_string
          description: String to use for full-text search
          schema:
            type: string
          required: false
        - in: query
          name: inProject
          description: Id of the project participating
          schema:
            type: integer
          required: false
        - in: query
          name: notInProject
          description: Id of the project users don't participate
          schema:
            type: integer
          required: false
        - in: query
          name: member
          description: Boolean with the membership flag value
          schema:
            type: boolean
          required: false
        - in: query
          name: coordinator
          description: Boolean with the coordinator flag value
```

```
              schema:
                type: boolean
              required: false
          - in: query
            name: blocked
            description: Boolean with the blocked flag value
            schema:
              type: boolean
            required: false
        responses:
          '200':
            description: Success
            content:
              application/json:
                schema:
                  type: array
                  items:
                    type: object
                    properties:
                      id:
                        type: integer
                      email:
                        type: string
                      password:
                        type: string
                      name:
                        type: string
                      image_path:
                        type: string
                      blocked:
                        type: string
                  example:
                    - id: 1
                      email: mferries0@yellowpages.com
                      password: t6AyMFhWp
                      name: Mahalia Ferries
                      image_path: ./img/default
                      blocked: false
  /api/projects:
    get:
      operationId: R201
      summary: 'R201: Search Projects API'
      description: 'Searches for projects and returns the results as JSON. Access:
AUTH_USR.'
      tags:
        - 'M02: Project'
      parameters:
        - in: query
          name: user_id
          description: Id of the user
          schema:
            type: integer
          required: false
        - in: query
```

```
        name: search_string
        description: String to use for full-text search
        schema:
          type: string
        required: false
    - in: query
      name: order
      description: String with the ordering atribute(alph/date)
      schema:
        type: string
      required: false
    - in: query
      name: favourite
      description: Boolean with the favourite flag value
      schema:
        type: boolean
      required: false
    - in: query
      name: coordinator
      description: Boolean with the coordinator flag value
      schema:
        type: boolean
      required: false
    - in: query
      name: member
      description: Boolean with the member flag value
      schema:
        type: boolean
      required: false
    - in: query
      name: archived
      description: Boolean with the archived flag value
      schema:
        type: boolean
      required: false
  responses:
    '200':
      description: Success
      content:
        application/json:
          schema:
            type: array
            items:
              type: object
              properties:
                id:
                  type: integer
                name:
                  type: string
                description:
                  type: string
                color:
                  type: string
                created_at:
```

```
                                  type: string
                          archived_at:
                              type: string
                  example:
                    - id: 5
                      name: e-enable scalable technologies
                      description: sagittis dui vel nisl duis ac nibh fusce lacus
purus
                      color: '#3fbcdd'
                      created_at: '2020-01-24'
                      archived_at: null
  /projects:
    get:
      operationId: R202
      summary: 'R202: Project creation Form'
      description: 'Provide project creation form. Access: AUTH_USR'
      tags:
        - 'M02: Project'
      responses:
        '200':
          description: OK. Show project creation UI
    post:
      operationId: R203
      summary: 'R203: Create Project'
      description: 'Processe the project creation form submission. Access:
AUTH_USR'
      tags:
        - 'M02: Project'
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                name:
                  type: string
                description:
                  type: string
                color:
                  type: string
              required:
                - name
      responses:
        '302':
          description: Redirect after processing the project creation form.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull creation. Redirect to project page
                  value: /projects/{id}
```

```
                  302Error:
                    description: Failed creation. Redirect to login form.
                    value: /users/{id}
    /projects/{id}:
      get:
        operationId: R204
        summary: 'R204: View project page'
        description: 'Show the individual project page. Access: MEMB, COOR'
        tags:
          - 'M02: Project'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        responses:
          '200':
            description: Ok. Show Project Page UI
      delete:
        operationId: R205
        summary: 'R205: Delete project'
        description: 'Delete project from service. Access: COOR, ADMIN'
        tags:
          - 'M02: Project'
          - 'M06: Administration'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        responses:
          '200':
            description: Ok. Project deleted
    /projects/{id}/details:
      get:
        operationId: R206
        summary: 'R206: View project details page'
        description: 'Show the individual project details page. Access: MEMB, COOR'
        tags:
          - 'M02: Project'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        responses:
          '200':
            description: Ok. Show Project details page UI
    /projects/{id}/edit/:
      get:
        operationId: R207
```

```
        summary: 'R207: Project edit Form '
        description: 'Show the individual project edit page. Access: COOR'
        tags:
          - 'M02: Project'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        responses:
          '200':
            description: OK. Show Project Update UI
    post:
      operationId: R208
      summary: 'R208: Edit project page'
      description: 'Edit individual project page. Access: COOR'
      tags:
        - 'M02: Project'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                name:
                  type: string
                description:
                  type: string
                color:
                  type: string
                archived_at:
                  type: string
      responses:
        '200':
          description: Ok. Project updated.
  /api/projects/addCoordinator:
    post:
      operationId: R209
      summary: 'R209: Add coordinator'
      description: 'Add new coordinator to project. Access: COOR'
      tags:
        - 'M02: Project'
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
```

```
              schema:
                type: object
                properties:
                  id_user:
                    type: integer
                  id_project:
                    type: integer
                required:
                  - id_user
                  - id_project
      responses:
        '302':
          description: Redirect after adding coordinator.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull added. Redirect to project page
                  value: /project/{id}/details
                302Error:
                  description: Failed add. Redirect to project page.
                  value: /project/{id}/details
  /api/projects/{id}/favourite:
    post:
      operationId: R210
      summary: 'R210: Modify favourite project'
      description: 'Modify favourite project. Access: MEMB, COOR'
      tags:
        - 'M02: Project'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      responses:
        '302':
          description: Redirect after favourite coordinator.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull favourite. Redirect to project page
                  value: /users
                302Error:
                  description: Failed favourite. Redirect to project page.
                  value: /users
  /api/projects/{id}/leave:
    post:
      operationId: R211
```

```yaml
        summary: 'R211: Leave project'
        description: 'Leave project. Access: MEMB, COOR'
        tags:
          - 'M02: Project'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        responses:
          '302':
            description: Redirect after leave project.
            headers:
              Location:
                schema:
                  type: string
                examples:
                  302Success:
                    description: Successfull leave. Redirect to user page
                    value: /users
                  302Error:
                    description: Failed favourite. Redirect to user page.
                    value: /users
  /api/tasks:
    get:
      operationId: R301
      summary: 'R301: Search Tasks API'
      description: 'Searches for tasks and returns the results as JSON. Access:
AUTH_USR.'
      tags:
        - 'M03: Tasks and Comments'
      parameters:
        - in: query
          name: project_id
          description: Id of the project
          schema:
            type: integer
          required: false
        - in: query
          name: user_id
          description: Id of the user
          schema:
            type: integer
          required: false
        - in: query
          name: search_string
          description: String to use for full-text search
          schema:
            type: string
          required: false
        - in: query
          name: finished
          description: Boolean with the finished flag value
```

```yaml
              schema:
                type: boolean
              required: true
            - in: query
              name: priority
              description: Integer with the priority value
              schema:
                type: integer
              required: false
            - in: query
              name: order
              description: String with the ordering atribute
              schema:
                type: string
              required: false
        responses:
          '200':
            description: Success
            content:
              application/json:
                schema:
                  type: array
                  items:
                    type: object
                    properties:
                      id:
                        type: integer
                      name:
                        type: string
                      description:
                        type: string
                      priority:
                        type: string
                      created_at:
                        type: string
                      finished_at:
                        type: string
                      task_number:
                        type: integer
                      due_date:
                        type: string
                      id_project:
                        type: integer
                      id_user:
                        type: integer
                  example:
                    - id: 2
                      name: Persevering radical time-frame
                      description: null
                      priority: 3
                      created_at: '2018-11-20'
                      finished_at: null
                      task_number: 0
                      due_date: null
```

```
                        id_project: 2
                        id_user: 2
    /projects/{project_id}/tasks:
      get:
        operationId: R302
        summary: 'R302: Task creation Form'
        description: 'Provide task creation form. Access: MEMB, COOR'
        tags:
          - 'M03: Tasks and Comments'
        parameters:
          - in: path
            name: project_id
            schema:
              type: integer
            required: true
        responses:
          '200':
            description: OK. Show task creation UI
    /tasks:
      post:
        operationId: R303
        summary: 'R303: Create Task'
        description: 'Processe the task creation form submission. Access: MEMB,
COOR'
        tags:
          - 'M03: Tasks and Comments'
        requestBody:
          required: true
          content:
            application/x-www-form-urlencoded:
              schema:
                type: object
                properties:
                  name:
                    type: string
                  description:
                    type: string
                  priority:
                    type: integer
                  due_date:
                    type: string
                  id_project:
                    type: integer
                  id_user:
                    type: integer
                required:
                  - name
                  - id_project
        responses:
          '302':
            description: Redirect after processing the task creation form.
            headers:
              Location:
                schema:
```

```
                      type: string
                  examples:
                    302Success:
                      description: Successfull creation. Redirect to task page
                      value: /tasks/{id}
                    302Error:
                      description: Failed creation. Redirect to project form.
                      value: /projects/{id}
  /tasks/{id}:
    get:
      operationId: R304
      summary: 'R304: View task page'
      description: 'Show the individual task page. Access: MEMB, COOR'
      tags:
        - 'M03: Tasks and Comments'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      responses:
        '200':
          description: Ok. Show Task Page UI
  /tasks/{id}/edit:
    get:
      operationId: R305
      summary: 'R305: Task edit Form '
      description: 'Show the individual task edit page. Access: MEMB,COOR'
      tags:
        - 'M03: Tasks and Comments'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      responses:
        '200':
          description: OK. Show Task Update UI
    post:
      operationId: R306
      summary: 'R306: Edit task page'
      description: 'Edit individual task page. Access: COOR, MEMB'
      tags:
        - 'M03: Tasks and Comments'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      requestBody:
        required: true
```

```
            content:
              application/x-www-form-urlencoded:
                schema:
                  type: object
                  properties:
                    name:
                      type: string
                    description:
                      type: string
                    priority:
                      type: integer
                    finished_at:
                      type: string
                    due_date:
                      type: string
                    id_user:
                      type: integer
        responses:
          '200':
            description: Ok. Task updated.
  /tasks/{id}/complete:
    post:
      operationId: R307
      summary: 'R307: Complete task'
      description: 'Complete task. Access: MEMB, COOR'
      tags:
        - 'M03: Project'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                today:
                  type: string
              required:
                - today
      responses:
        '302':
          description: Redirect after Complete task.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull complete. Redirect to task page
```

```
                              value: /tasks/{id}
                       302Error:
                         description: Failed complete. Redirect to task page.
                         value: /tasks/{id}
    /tasks/{id}/clone:
      post:
        operationId: R308
        summary: 'R308: Clone task'
        description: 'Clone task. Access: MEMB, COOR'
        tags:
          - 'M03: Project'
        parameters:
          - in: path
            name: id
            schema:
              type: integer
            required: true
        responses:
          '302':
            description: Redirect after Clone task.
            headers:
              Location:
                schema:
                  type: string
                examples:
                  302Success:
                    description: Successfull complete. Redirect to cloned task page
                    value: /tasks/{id}
                  302Error:
                    description: Failed complete. Redirect to cloned user page page.
                    value: /users
    /comments:
      post:
        operationId: R309
        summary: 'R309: Create comment'
        description: 'Processe the comment creation form submission. Access: MEMB,
COOR'
        tags:
          - 'M03: Tasks and Comments'
        requestBody:
          required: true
          content:
            application/x-www-form-urlencoded:
              schema:
                type: object
                properties:
                  content:
                    type: string
                  id_task:
                    type: integer
                  id_user:
                    type: integer
                required:
                  - content
```

```
                        - id_task
                        - id_user
            responses:
              '302':
                description: Redirect after processing the task creation form.
                headers:
                  Location:
                    schema:
                      type: string
                    examples:
                      302Success:
                        description: Successfull creation. Redirect to task page
                        value: /tasks/{id}
                      302Error:
                        description: Failed creation. Redirect to task form.
                        value: /tasks/{id}
  /comments/{id}:
    post:
      operationId: R310
      summary: 'R310: Edit task comment'
      description: 'Edit individual task comment. Access: OWN'
      tags:
        - 'M03: Tasks and Comments'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                content:
                  type: string
      responses:
        '200':
          description: Ok. Task comment updated.
    delete:
      operationId: R311
      summary: 'R311: Delete task comment'
      description: 'Delete task comment from service. Access: OWN, ADMIN'
      tags:
        - 'M03: Tasks and Comments'
        - 'M06: Administration'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
```

```yaml
      responses:
        '200':
          description: Ok. Comment deleted
  /tasks/labels:
    post:
      operationId: R312
      summary: 'R312: Assign label to task'
      description: 'Processe the label assignment to task submission. Access:
MEMB, COOR'
      tags:
        - 'M03: Tasks and Comments'
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                id_task:
                  type: integer
                id_label:
                  type: integer
                required:
                  - id_task
                  - id_label
      responses:
        '302':
          description: Redirect after processing the label assignment.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull assigned. Redirect to task page
                  value: /tasks/{id}
                302Error:
                  description: Failed assignment. Redirect to task form.
                  value: /tasks/{id}
    delete:
      operationId: R313
      summary: 'R313: Delete label assignment'
      description: 'Delete label assignment from service. Access: COOR, MEMB'
      tags:
        - 'M03: Tasks and Comments'
      parameters:
        - in: query
          name: id_task
          description: Id of the task
          schema:
            type: integer
          required: false
        - in: query
          name: id_label
```

```
          description: id of the label
          schema:
            type: string
          required: false
      - in: query
        name: id_task_label
        description: id of the task_label
        schema:
          type: string
        required: false
    responses:
      '200':
        description: Ok. Label assignment deleted.
/messages:
  post:
    operationId: R401
    summary: '401: Create message'
    description: 'Processe the message creation form submission. Access: MEMB,
COOR'
    tags:
      - 'M04: Project Forum and Labels'
    requestBody:
      required: true
      content:
        application/x-www-form-urlencoded:
          schema:
            type: object
            properties:
              content:
                type: string
              id_project:
                type: integer
              id_user:
                type: integer
            required:
              - content
              - id_project
              - id_user
    responses:
      '302':
        description: Redirect after processing the message creation form.
        headers:
          Location:
            schema:
              type: string
            examples:
              302Success:
                description: Successfull creation. Redirect to project page
                value: /project/{id}
              302Error:
                description: Failed creation. Redirect to project form.
                value: /project/{id}
/messages/{id}:
  post:
```

```yaml
      operationId: R402
      summary: 'R402: Edit forum message'
      description: 'Edit individual forum message. Access: OWN'
      tags:
        - 'M04: Project Forum and Labels'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                content:
                  type: string
      responses:
        '200':
          description: Ok. Task message updated.
    delete:
      operationId: R403
      summary: 'R403: Delete Forum message'
      description: 'Delete forum message from service. Access: OWN, ADMIN'
      tags:
        - 'M04: Project Forum and Labels'
        - 'M06: Administration'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      responses:
        '200':
          description: Ok. Message deleted
  /api/labels:
    get:
      operationId: R404
      summary: 'R404: Search Labels API'
      description: 'Searches for Labels and returns the results as JSON. Access:
AUTH_USR.'
      tags:
        - 'M04: Project Forum and Labels'
      parameters:
        - in: query
          name: project_id
          description: Id of the project
          schema:
            type: integer
          required: true
```

```
            - in: query
              name: name
              description: Name of the label
              schema:
                type: string
              required: false
        responses:
          '200':
            description: Success
            content:
              application/json:
                schema:
                  type: array
                  items:
                    type: object
                    properties:
                      id:
                        type: integer
                      name:
                        type: string
                      id_project:
                        type: integer
                  example:
                    - id: 4
                      content: Epoxy Flooring
                      id_project: 4
  /labels:
    get:
      operationId: R405
      summary: 'R405: Labels creation Form'
      description: 'Provide labels creation form. Access: MEMB, COOR'
      tags:
        - 'M04: Project Forum and Labels'
      responses:
        '200':
          description: OK. Show labels creation UI
    post:
      operationId: R406
      summary: 'R406: Create label'
      description: 'Processe the label creation form submission. Access: MEMB,
COOR'
      tags:
        - 'M04: Project Forum and Labels'
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                name:
                  type: string
                id_project:
                  type: integer
```

```yaml
                      required:
                        - name
                        - id_project
          responses:
            '302':
              description: Redirect after processing the label creation form.
              headers:
                Location:
                  schema:
                    type: string
                  examples:
                    302Success:
                      description: Successfull creation. Redirect to project page
                      value: /projects/{id}
                    302Error:
                      description: Failed creation. Redirect to project form.
                      value: /projects/{id}
  /labels/{id}:
    post:
      operationId: R407
      summary: 'R407: Edit label'
      description: 'Edit individual label. Access: COOR'
      tags:
        - 'M04: Project Forum and Labels'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
          required: true
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                name:
                  type: string
      responses:
        '200':
          description: Ok. Label updated.
    delete:
      operationId: R408
      summary: 'R408: Delete label'
      description: 'Delete label from service. Access: COOR, ADMIN'
      tags:
        - 'M04: Project Forum and Labels'
        - 'M06: Administration'
      parameters:
        - in: path
          name: id
          schema:
            type: integer
```

```
          required: true
       responses:
         '200':
           description: Ok. Label deleted
   /api/invites/:
     get:
       operationId: R501
       summary: 'R501: Search Invites API'
       description: 'Searches for Invites and returns the results as JSON. Access:
   AUTH_USR.'
       tags:
         - 'M05: Invites and Notifications'
       parameters:
         - in: query
           name: id_user
           description: Id of the user
           schema:
             type: integer
           required: true
         - in: query
           name: id_project
           description: Name of the project
           schema:
             type: string
           required: false
       responses:
         '200':
           description: Success
           content:
             application/json:
               schema:
                 type: array
                 items:
                   type: object
                   properties:
                     id:
                       type: integer
                     created_at:
                       type: string
                     id_user:
                       type: integer
                     id_project:
                       type: integer
                   example:
                     - id: 1
                       content: '2020-03-12'
                       id_user: 21
                       id_project: 1
     post:
       operationId: R502
       summary: 'R502: Create invite'
       description: 'Processe the invite creation submission. Access: COOR'
       tags:
         - 'M05: Invites and Notifications'
```

```
      requestBody:
        required: true
        content:
          application/x-www-form-urlencoded:
            schema:
              type: object
              properties:
                id_user:
                  type: integer
                id_project:
                  type: integer
                required:
                  - id_user
                  - id_project
      responses:
        '302':
          description: Redirect after processing the invite creation.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull creation. Redirect to project page
                  value: /projects/{id}
                302Error:
                  description: Failed creation. Redirect to project page.
                  value: /projects/{id}
/api/invites/{id}/accept:
  post:
    operationId: R503
    summary: 'R503: Accept invite'
    description: 'Accept invite of user. Access: OWN'
    tags:
      - 'M05: Invites and Notifications'
    parameters:
      - in: path
        name: id
        schema:
          type: integer
        required: true
    responses:
      '302':
        description: Redirect after accepting the invite.
        headers:
          Location:
            schema:
              type: string
            examples:
              302Success:
                description: Successfull accept. Redirect to project page
                value: /projects/{id}
              302Error:
                description: Failed accept. Redirect to user page.
```

```
                                        value: /users/{id}
      /api/invites/{id}/reject:
        post:
          operationId: R504
          summary: 'R504: Reject invite'
          description: 'Reject invite of user. Access: OWN'
          tags:
            - 'M05: Invites and Notifications'
          parameters:
            - in: path
              name: id
              schema:
                type: integer
              required: true
          responses:
            '302':
              description: Redirect after reject the invite.
              headers:
                Location:
                  schema:
                    type: string
                  examples:
                    302Success:
                      description: Successfull reject. Redirect to user page
                      value: /users/{id}
                    302Error:
                      description: Failed reject. Redirect to user page.
                      value: /users/{id}
      /api/notifications:
        get:
          operationId: R505
          summary: 'R505: Search Notifications API'
          description: >-
            Searches for Notifications and returns the results as JSON. Access:
            AUTH_USR.
          tags:
            - 'M05: Invites and Notifications'
          parameters:
            - in: query
              name: id_user
              description: Id of the user
              schema:
                type: integer
              required: true
          responses:
            '200':
              description: Success
              content:
                application/json:
                  schema:
                    type: array
                    items:
                      type: object
                      properties:
```

```yaml
                        id:
                          type: integer
                        content:
                          type: string
                        created_at:
                          type: string
                        id_project:
                          type: integer
                        seen:
                          type: boolean
                        id_user:
                          type: integer
                    example:
                      - id: 1
                        content: Invite to a new project!
                        id_project: 1
                        seen: false
                        id_user: 21
  /api/notifications/{user_id}/{notification_id}:
    post:
      operationId: R506
      summary: 'R506: See notification'
      description: 'See notification. Access: OWN'
      tags:
        - 'M05: Invites and Notifications'
      parameters:
        - in: path
          name: user_id
          schema:
            type: integer
          required: true
        - in: path
          name: notification_id
          schema:
            type: integer
          required: true
      responses:
        '302':
          description: Redirect after see the invite.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull seen. Redirect to project page
                  value: /projects/{id}
                302Error:
                  description: Failed seen. Redirect to user page.
                  value: /users/{id}
  /admin:
    get:
      operationId: R601
      summary: 'R601: Admin users page'
```

```yaml
            description: 'Provide admin users page. Access: ADMIN'
            tags:
              - 'M06: Administration'
            responses:
              '200':
                description: OK. Show admin page UI
      /admin/projects:
        get:
          operationId: R602
          summary: 'R602: Admin projects page'
          description: 'Provide admin projects page. Access: ADMIN'
          tags:
            - 'M06: Administration'
          responses:
            '200':
              description: OK. Show admin page UI
      /api/block/{user_id}:
        post:
          operationId: R603
          summary: 'R603: Block User'
          description: 'Block user of the webservice. Access: ADMIN'
          tags:
            - 'M06: Administration'
          parameters:
            - in: path
              name: user_id
              schema:
                type: integer
              required: true
          responses:
            '302':
              description: Redirect after blocking user.
              headers:
                Location:
                  schema:
                    type: string
                  examples:
                    302Success:
                      description: Successfull blocked. Redirect to admin page
                      value: /admin
                    302Error:
                      description: Failed block. Redirect to admin page.
                      value: /admin
      /api/unblock/{user_id}:
        post:
          operationId: R604
          summary: 'R604: Unblock User'
          description: 'Unblock user of the webservice. Access: ADMIN'
          tags:
            - 'M06: Administration'
          parameters:
            - in: path
              name: user_id
              schema:
```

```
                  type: integer
              required: true
        responses:
          '302':
            description: Redirect after unblocking user.
            headers:
              Location:
                schema:
                  type: string
                examples:
                  302Success:
                    description: Successfull unblocked. Redirect to admin page
                    value: /admin
                  302Error:
                    description: Failed unblock. Redirect to admin page.
                    value: /admin
  /:
    get:
      operationId: R701
      summary: 'R701: Home page'
      description: 'Provide home page. Access: PUB'
      tags:
        - 'M07: Static Pages'
      responses:
        '200':
          description: OK. Show home page UI
  /about:
    get:
      operationId: R702
      summary: 'R702: About page'
      description: 'Provide about page. Access: PUB'
      tags:
        - 'M07: Static Pages'
      responses:
        '200':
          description: OK. Show about page UI
  /contact:
    get:
      operationId: R703
      summary: 'R703: Contact page'
      description: 'Provide contact page. Access: PUB'
      tags:
        - 'M07: Static Pages'
      responses:
        '200':
          description: OK. Show contact page UI
    post:
      operationId: R704
      summary: 'R704: Submit contact form'
      description: 'Provide contact page. Access: PUB'
      tags:
        - 'M07: Static Pages'
      requestBody:
        required: true
```

```
            content:
              application/x-www-form-urlencoded:
                schema:
                  type: object
                  properties:
                    email:
                      type: string
                    name:
                      type: string
                    text:
                      type: string
                  required:
                    - email
                    - name
                    - text
      responses:
        '302':
          description: Redirect after submit contact.
          headers:
            Location:
              schema:
                type: string
              examples:
                302Success:
                  description: Successfull submit. Redirect to home page
                  value: /
                302Error:
                  description: Failed submit. Redirect to contact page.
                  value: /contact
  /service:
    get:
      operationId: R705
      summary: 'R705: Service page'
      description: 'Provide service page. Access: PUB'
      tags:
        - 'M07: Static Pages'
      responses:
        '200':
          description: OK. Show service page UI
  /404:
    get:
      operationId: R706
      summary: 'R706: Errro page'
      description: 'Provide erro page. Access: PUB'
      tags:
        - 'M07: Static Pages'
      responses:
        '200':
          description: OK. Show erro page UI
```

A8: Vertical prototype

This artefact has the necessary features implemented as well as other identified below. The vertical prototype abled us to get familiar with the tecnologies of the project and to build the main structure of our application.

The LBAW Framework is behind the architecture of our prototype and all layers were build above it. Until now, we already worked in: user interface, business logic and data access. We also implemented features of creation, addition, search, update, removal and deletion of information, the control of permissions and presentation of error messages.

1. Implemented Features

Inside this section we present the currently implemented web and system features.

1.1. Implemented User Stories

The user stories that are for now implemented in the prototype are described in the following table.

User Story reference	Name	Priority	Description	Related Functional Requirements
US0.1	See Home	high	As a User, I want to access the home page, so that I can see a brief presentation of the website	NA
US0.2	See About	high	As a User, I want to access the about page, so that I can see a complete description of the website as well as its creators	FR.061
US0.3	Consult Contacts	medium	As a User, I want to access contacts, so that I can come in touch with the platform creators	FR.063
US0.4	Consult Services	medium	As a User, I want to access the services information, so that I can see the website's services	FR.062
US1.1	Sign-in	high	As a Visitor, I want to authenticate into the system, so that I can access privileged information	FR.011
US1.2	Sign-up	high	As a Visitor, I want to register myself into the system, so that I can authenticate myself into the system	FR.012
US2.1	Create projects	high	As a AuthUser, I want to create my own projects, so that the users can work on the project	FR.201
US2.2	View projects	high	As a AuthUser, I want to view the projects that are allocated to me, so that I can work on them	FR.202
US2.3	Mark project as favorite	high	As a AuthUser, I want to mark the projects as favorite, so that I can easily find the projects that interests me the most	FR.204

User Story reference	Name	Priority	Description	Related Functional Requirements
US2.4	Logout	high	As a AuthUser, I want to log out of my account, so that I can leave the website safely	FR.011
US2.5	Delete account	high	As a AuthUser, I want to be able to delete my account, so that I can remove my information from the website	FR.014
US2.6	View profile	high	As a AuthUser, I want to view my profile information, so that I can check it	FR.021
US2.7	Edit profile	high	As a AuthUser, I want to edit my profile information, so that I can update it	FR.022
US2.14	Order Project	low	As a AuthUser, I want to be able to go through my project in an order of my choice (alphabetical, cronological), so that I can find it easily	FR.035
US3.1	Create Tasks	high	As a Member, I want to create tasks, so that I can add to dos to the project	FR.301
US3.2	Manage Tasks	high	As a Member, I want to manage tasks details, so that I can change task information such as due date and priority	FR.302
US3.3	Assign Tasks	high	As a Member, I want to assign tasks to members of the project, so that everyone knows what to do	FR.303
US3.4	View Tasks	high	As a Member, I want to view all the tasks, so that I can see what everyone needed to do	FR.304
US3.6	Complete Tasks	high	As a Member, I want to be able to check the tasks I have done, so that everyone knows it is completed	FR.306
US3.7	Search Tasks	high	As a Member, I want to be able to search tasks by keywords, so that I can easily find what I want	FR.312, FR.032
US3.8	Leave Project	high	As a Member, I want to be able to leave a project , so that I stop being part of the that project team	FR.309
US3.9	View Project Details	high	As a Member, I want to be able to check project information, so that I can know the members of the project along with other relevant information	FR.310

User Story reference	Name	Priority	Description	Related Functional Requirements
US3.10	View Team Profiles	high	As a Member, I want to be able to see other team members profiles , so that I can have more information about them	FR.311
US3.13	View Project Timeline	medium	As a Member, I want to be able to see project tasks done, so that I can have more understanding about the development of the project	FR.331
US4.1	Assign Coordinator	high	As a Coordinator, I want to be able to assign new coordinators to the project, so that I can have other Users helping me in the project management	FR.502
US4.2	Edit Project Details	high	As a Coordinator, I want to edit current project information, so that I can update information referent to the project	FR.503
US6.1	Login Admin Account	high	As an Admin, I want to be able to login in to my account, so that I can have an administrator profile	FR.041, FR.011
US6.2	Administer User	high	As an Admin, I want to search through the existing users, so that I can view, delete them	FR.042, FR.031
US6.5	Delete User	high	As a Admin, I want erase User accounts, so that they can not use their acount more.	FR.044
US6.6	Browse Projects	high	As an Admin, I want to be able to browse through projects, so that I can iterate through the existing projects	FR.701, FR.032
US6.7	View Project	high	As an Admin, I want view project details, so that I can validate the curret usage of the given features	FR.702

Some **features and requirements**, not previously noted as an user story we leave here in text:

1. (FR.501) Add user to project: for now this is possible, however not through the invite method, because the email feature is not yet implemented.
2. (FR.032) Full-text search: this is used in searching for projects or tasks, as well as for the Admin.
3. (FR.031) Exact match search: this is used in searching users to add them to projects, to assign them to tasks or as an Admin to simply search for them.

Some notes about the current **use of the application**:

1. To do some kind of search where there is a button search, it is all included in a form so it is necessary to click in the submit button. This happens in the user page to search for some specific filter or order

mode.

2. When you assign tasks when there is another user already assigned to it, another task is created assigned to the user you selected. It does not appear in the page of that task because it creates other that is only displayed in the project page. At the creation time, for now, you can only assign one project member, to the task and to assign new ones you need to add it afterawards.

3. For now, due to the fact that invites are not yet implemented, when an user is added to a project you need to refresh the page in order to see it appear in the members section.

4. Just to remember some rules talked in the previous artifacts, when an user wants to leave a project and it is the only coordinator it is not allowed to do it, having to leave the project with at least one coordinator besides him.

5. Note that to create new projects and tasks you need to give them a name, otherwise it does not submit the form.

6. For last, when authenticated the logo in the left top corner redirects every user to its user page (where projects are presented).

1.2. Implemented Web Resources

The web resources that were implemented in the present prototype are displayed in the next section.

Module M01: Authentication and Individual Profile

Web Resource Reference	Method	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
R106: View User Page	GET	/users
R107: Delete User	DELETE	/users/{id}
R108: View User Profile	GET	/users/{id}/profile
R109: User Edit Profile Form	GET	/users/{id}/update
R110: Edit User Profile	POST	/users/{id}/update
R111: Recover Password Form	GET	/recoverPassword
R114: Search Users API	GET	/api/users

Module M02: Project

Web Resource Reference	Method	URL
R202: Project creation Form	GET	/projects
R203: Create Project	POST	/projects

Web Resource Reference	Method	URL
R204: View project page	GET	/projects/{id}
R205: Delete project	POST	/projects/{id}
R206: View project details page	GET	/projects/{id}/details
R207: Project edit Form	GET	/projects/{id}/edit
R208: Edit project page	POST	/projects/{id}/edit
R209: Add coordinator	POST	/api/projects/addCoordinator
R210: Modify favourite project	POST	/api/projects/{id}/favourite
R211: Leave project	POST	/api/projects/{id}/leave

Module M03: Tasks and Comments

Web Resource Reference	Method	URL
R301: Search Tasks API	GET	/api/tasks
R302: Task creation Form	GET	/projects/{project_id}/tasks
R303: Create Task	POST	/tasks
R304: View task page	GET	/tasks/{id}
R305: Task edit Form	GET	/tasks/{id}/edit
R306: Edit task page	POST	/tasks/{id}/edit
R307: Complete task	POST	/tasks/{id}/complete
R308: Clone task	POST	/tasks/{id}/clone

Module M05: Invites and Notifications

Web Resource Reference	Method	URL
R501: Search Invites API	GET	/api/invites

Module M06: User Administration

Web Resource Reference	Method	URL
R601: Admin users page	GET	/admin
R602: Admin projects page	GET	/admin/projects

Module M07: User Administration

Web Resource Reference	Method	URL
R701: Home page	GET	/

Web Resource Reference	Method	URL
R702: About page	GET	/about
R703: Contact page	GET	/contact
R704: Submit contact form	POST	/contact
R705: Service page	GET	/service

2. Prototype

The prototype is available at: (link!!!!!!!!!!!!!)

Credentials:

- Admin User:
    - Email: admin@admin.com
    - Password: 123456
- Regular User:
    - Email: sbennallck2@is.gd
    - Password: 123456

The code is available at: (link!!!!!!!!!!!!!)

Revision history

Changes made to the first submission:

1.
    -

GROUP2102, 03/01/2021

- André Pereira, up201905650@up.pt
- Beatriz Lopes dos Santos, up201906888@up.pt
- Matilde Oliveira, up201906954@up.pt (editor)
- Ricardo Ferreira, up201907835@up.pt