EAP: Architecture Specification and Prototype

A7: Web Resources Specification

This artifact documents the architecture of the web application to be developed, indicating the catalog of resources, the properties of each resource, and the format of JSON responses. This specification adheres to the OpenAPI standard using YAML.

This artifact presents the documentation for UPortoEvent, including the CRUD (create, read, update, delete) operations.

1. Overview

The modules of the application are identified and briefly described with the web resources associated in the individual documentation. More information about web resources associated with each module can be found in each module's separate documentation in the OpenAPI specification.

Module	Description
M01 - Authentication and Individual Profile	Web component that concerns itself with the user's authentication and each user's profile. It covers, inter alia, login/logout, registration and view and edit profile data.
M02 - Events	Web component that deals with event creation, management and promotion. Among others, contains Polls, Tags and Comments APIs.
M03 - Notifications	Web component that handles notifications.
M04 - User Administration and Static pages	Web resources associated with user management such as view, search and delete users, view and change user information, manage reports and user support tickets. Web resources with static content are associated with this module: dashboard, search, contact, about.

2. Permissions

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

Abbreviation	Permission type	Description
PUB	Public	Unauthenticated users
USR	User	Authenticated users, have priviliges over their profile pages
OWN	Owner	Users who created events, have priviliges on their own events
COM	Comment Author	User who authored an comment can delete it
PAR	Event participant	User who take part in an event
ADM	Administrator	Users with extended permissions, system managers, super users

3. OpenAPI Specification

OpenAPI specification in YAML format to describe the web application's web resources.

OpenAPI specification file

Swagger documentation

A8: Vertical Prototype

The Vertical Prototype includes the implementation of two or more user stories (the simplest) and aims to validate the architecture presented, also serving to gain familiarity with the technologies used in the project.

The implementation is based on the LBAW Framework. The prototype implements pages for visualizing, inserting, editing and removing information, as well as functionality for access management.

1. Implemented Features

1.1. Implemented User Stories

ID	Actor	Name	Priority	Description
us02	User	View Event	High	As a User, I want to navigate through a specific public event so I can see more detailed information

ID	Actor	Name	Priority	Description
vi01	Visitor	Log In	High	As a visitor, I want to be able to log-in, to get a status of authenticated user.
vi02	Visitor	Sign Up	High	As a visitor, I want to be able to create my profile on the site to become an authenticated user.
au01	Auth. user	Home	High	As an authenticated user, I want to access my home page, where I can see all information about the website.
au03	Auth. user	Create Event	High	As an authenticated user, I want to create new events by myself to become an event organizer.
au04	Auth. user	Log Out	High	As an authenticated user, I want to be able to log out from service for privacy purposes.
au06	Auth. user	Edit Profile	High	As an authenticated user, I want to edit my profile whenever I please, to keep it up to date.
au07	Auth. user	Delete Profile	High	As an authenticated user, I want to delete my profile if I feel like it.
au08	Auth. user	View Profile	High	As an authenticated user, I want to see my profile and others to check information I want.
au12	Auth. user	Comment	Medium	As an authenticated user, I want to ask any questions I'd like about an event, even if I'm not a member of this event yet.

ID	Actor	Name	Priority	Description
ad01	Admin	Delete Event	High	As an administrator, I want to be able to delete events on the site, if they violate the rules of the service.

1.2. Implemented Web Resources   M01 - Authentication and Individual Profile

Web Resource Reference	URL
R101: Sign In Form	/login
R102: Sign In Action	POST /login
R103: Logout Action	POST /logout
R104: Sign Up Form	/register
R105: Sign Up Action	POST /register
R106: View User Profile	GET /profile/{id}
R107: Edit Profile	PATCH /profile/{id}
R108: Delete Account	POST /profile/{id}

M02 - Events

Web Resource Reference	URL
R202: Create Event	POST /create_event
R203: Get Event Information	GET /event/{id}
R207: Edit Event	PATCH /event/{id}
R212: Get Comments	GET /event/{id}
R213: Add a comment	POST /event/{id}
R214: Delete Comment	DELETE event/{id}/comment
R219: Like a comment	POST /event/{id}
R220: Unlike a comment	POST /event/{id}

M04 - User Administration and Static pages

Web Resource Reference	URL
R401: View Homepage	GET /home

2. Prototype

The Prototype is available at https://lbaw22122.lbaw.fe.up.pt/

Credentials:

for admin:
email: admin@example.com
password: 1234

for casual user:
email: user@example.com
password: 123456 ## Revision history

1. 26/11/2022 - Late delivery

GROUP22122, 26/11/2022

- João Sousa, up201904739@up.pt

- Mikhail Ermolaev, up202203498@up.pt
- David Burchakov, up202203777@up.pt (editor)
- Válter Castro, up201706546@up.pt