

# Event Manager Documentation

## User documentation

This project is a web application built for enhancing (not only) self-held events advertisement. With its simple and user-friendly interface, features such as viewing, filtering, creating and managing events of different categories are available.

There are three access levels a user can possess:

- Unregistered users;
- Registered users;
- Administrators.

An unregistered user is able to view the list of all events, which are currently active and are present on the main page. An event is represented with an "event card", which contains all the necessary information about the oncoming activity, such as:

Title, Description, Location, Starting Date and Time, Location and Entry price (or free entry note).

The event's card is accompanied by a small image to provide a better vision on the future event.


An event must be of one of the following 5 categories:

- Movie;
- Party;
- Performance;
- Rave;
- Sport.

Filtering option is also presented based on the categories above.

Events


Any



**kill bill 2**

open air movie show


Location: Azgura museum  
Start at: 2025-01-11 17:05:53  
Entry: free entry



**kick the fn base**

dnb rave techno kill me night


Location: Module club Octoberska  
Start at: 2025-01-11 17:05:53  
Entry: 10€



**house house house**

sunday house party at axidental point everyone should come ! ! !

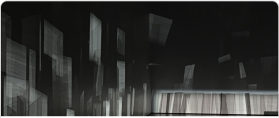
Location: Accidental point  
Start at: 2025-01-11 17:05:53  
Entry: 5€



**Sunday city run**

lets run together


Location: Park Gorkey  
Start at: 2025-01-19 07:00:00  
Entry: free entry



**the dark room**

modern art exhibition


Location: Vershny art space  
Start at: 2025-01-16 21:28:00  
Entry: free entry



**new era**

disco party

Location: Accidental point  
Start at: 2025-01-17 18:50:00  
Entry: 5€



**QWER IS HERE**

performance of anonymous artist QWER

Location: unknown  
Start at: 2025-02-01 13:10:00  
Entry: free entry

1

After a simple account creation process a user can become registered and access further features such as directly creating events advertisements, which will be placed among others for everyone to see. Each registered user can view their own events on a separate "My Events" page with the ability to remove old and create new once.

Registered users are able to further modify their account by adding/changing user image, username and email.



## User Info



username

ivgh

email

ivgh@ivgh.iv

Add/Change Profile Image

Choose File

No file chosen

save

## Create Event

Title

Description

Category

Movie

Start Date and Time

dd.mm.yyyy, --:--

Location

Entry Price

Cover Image

Choose File

No file chosen

create event

## Technical documentation

The “Event manager” project is a full-stack web application developed with a tech stack of:

vanilla PHP, HTML, CSS, JS and MySQL.

Key features:

- Routing in the application is implemented using so called **Page Routing** by referencing the actual names of the scripts, representing each individual page.

- User management in the application covers account creation, signin, signup and logout processes. Passwords are hashed and are never stored in the plain format. Authentication is implemented making use of PHP sessions. Unregistered, registered and administrator users are present in the application's workcycle. There are publicly, registration required and admin only available pages in the app, with the access rights checks accordingly.
- Authentication is managed with the (session based) `Auth` service, featuring methods:
  - `checkAuth()` with redirect to signin
  - `checkAdmin(): bool`
  - `login($userId, $userName)`
  - `logout()`
  - `isLoggedIn(): bool`
- There is a Crypto service, responsible for hashing and verifying passwords, making use of the `password_hash()` and `password_verify()` methods.
- Data Management in the application is implemented with the:
  - Entities types, representing business logic entities (Users, Events);
  - Database communication layer is represented by the `DataContext` class, containing reusable `executeSQL()` and `executeFromSQL()` methods (checkout PHPDoc comments for more info);
  - Users and Events Repositories containing CRUD methods;
  - Categories Enum enhancing current categories consistency among the application;
  - CreateTable Scripts.
- Events grid and Admin page data display are implemented making use of pagination with partial data fetching according to the current page, page is determined by the optional `?page=` url parametrs.
- Image processing: (allowed: png, jpg), limit by `php_ini_maxfilesize`, `imageUri = entityId_user/eventimg`. Replacing images on change, deleting images on entities deletion, **no artifacts left**.

- Admin page

Users management: make/reduce admin rights, prevent reduce own rights, delete user (cascade delete events, delete related images

```
( deleteEventImages($eventId) , deleteUserImages($userId) );
```

Events management: delete event ( deleteEventImages(\$eventId) );

- MISC

- @media print styles for Admin page;
- JS Filter by Categories script.