The PHP file contains part of a code for the backend of a TV Web application which I developed a year ago.

The EventController class is a Laravel controller responsible for managing events in the TV web application. Below are the use of some of the functions I created:

1. Managing Events:

getEvent: Fetches all events, paginates them, and includes details about deleted events (trash). It prepares the data for rendering the event listing view.

addEvent: Displays the form for creating a new event.

postAddEvent: Handles event creation. It saves the event details, processes and stores uploaded videos and images, and logs the activity.

2. Handling Event Deletions:

deleteEvent: Soft deletes an event by moving it to an "EventTrash" model and logs the action.

deletedEvent: Displays all soft-deleted events.

restoreEvent: Restores a soft-deleted event from the trash and logs the action.

permanentDelete: Permanently deletes an event from the trash and logs the action.

3. Editing Events:

getEditEvent: Retrieves event details (including associated images and videos) for editing.

postEditEvent: Updates event details, uploads new videos/images if provided, and logs the changes. It also removes old images before adding new ones.

4. Media Management:

getMedia: Fetches and paginates all media (images) associated with events for display in the media management view.

**Why This Approach Was Taken**

* Data Validation: This ensures the user enters the right value in the form. Validation is enforced during video uploads to ensure compatibility.
* Logging: Activities like event creation, updates, deletion, and restoration are logged for accountability and tracking user actions.
* Soft Deletion: Events are initially moved to “trash" to allow recovery, safeguarding against accidental deletions.
* File Management: The code handles video and image uploads efficiently, including resizing images and storing them in a structured manner.
* User Experience: Pagination for events and media ensures better performance and usability for large datasets.