**Задание**

Написать на PHP небольшое приложение для импорта контента из файлов.

**1)** Приложение будет содержать 2 страницы.

На первой странице своего веб-приложения размещаем:

- поле для загрузки файла (только CSV формат, размер не более 1 Мб),

- кнопку «Import»,

- кнопку «Clear all records»,

- ссылку «View results», ведущую на вторую страницу.

На второй странице:

- ссылку «Import data», ведущую на первую страницу.

- таблицу результатов с загруженным контентом.

**2)** Сами файлы для импорта представляют собой csv-таблицы, на основе которых и нужно будет создавать контент. Они лежат в папке import.

Файлы users-1.csv и users-2.csv одинаковые по структуре (имеют одинаковый набор колонок и заголовки), но содержат разный контент. Нужно будет создать базу данных и таблицу в ней, с колонками, соответствующими заголовкам этих csv-таблиц.

Файл users-3.csv — неправильный по структуре.

**3)** При нажатии на кнопку «Import» происходит загрузка файла.

При этом содержимое загружается в БД в соответствии с колонками в csv-файле.

Колонка UID — уникальна для записи в csv-файле, поэтому если в БД нет записи с данным UID, то она создаётся, а если уже есть, то данные в БД для этой строки обновляются.

Для тестирования этого и предназначены файлы users-1.csv и users-2.csv.

Если же файл (users-3.csv) имеет неправильную структуру, то данные в БД не пишутся, а пользователю выдаётся ошибка о том, что файл не корректный.

**4)** При нажатии на кнопку «Clear all records» все записи в БД очищаются.

**5)** На странице с результатами выводим в виде html-таблицы всё содержимое, которое было загружено в БД.

Если записей в БД ещё нет, то выводим вместо таблицы сообщение об этом.

**Дополнительно:**

**6)** Под таблицей результатов разместить кнопку «Export to CSV», нажав на которую можно будет скачать в виде csv-файла всё содержимое БД.

Структура этого файла аналогична users-1.csv и users-2.csv.

**7)** Над таблицей результатов размещаем фильтры по всем колонкам из таблицы. Поле Gender делаем выпадающим списком, остальные поля — текстовые.

**8)** Сделать так, что при клике на заголовки таблицы результатов будет производится сортировка данных по выбранному заголовку. Первый клик — сортировка по возрастанию, второй — по убыванию.

**Требования к заданию:**

1) В приложении нужно с помощью чистого объектно-ориентированного PHP реализовать модель MVC (PHP-фреймворки использовать не рекомендуется).

2) Проект можно опубликовать в собственном публичном github репозитории и прислать ссылку на него, либо же прислать в виде ZIP архива на электронную почту.

3) Обязательно наличие файла README с описанием того, как запустить проект локально.

4) Допускается обоснованное использование дополнительных PHP и JS библиотек, решающих мелкие задачи.

5) Приветствуется хорошо структурированный и задокументированный код и использование паттернов проектирования.

6) Внешнее оформление страниц остаётся на усмотрение кандидата, однако будет плюсом наличие аккуратной темизации, корректных сообщений об ошибках, валидации и др.