

ЗАДАНИЕ 8

Для предыдущего задания реализовать веб-сервис, принимающий данные формы в формате JSON или XML. Для не авторизованного пользователя веб-сервис возвращает логин и пароль созданного пользователя и адрес профиля нового пользователя. Для авторизованных пользователей веб-сервис позволяет менять все ранее отправленные данные, кроме логина и пароля. При реализации веб-сервиса валидация параметров и логика работы полностью аналогичны предыдущему заданию.

Реализовать отправку формы в предыдущем задании помощью JavaScript (XMLHttpRequest или Fetch) через разработанный веб-сервис без перезагрузки страницы, если в браузере включен JavaScript. В противном случае форма отправляется как обычно.

В задании можно использовать JQuery или другие библиотеки на клиенте. Можно переиспользовать код фронтэнда с прошлого семестра. Использование сторонних фреймворков на сервере (бекэнде) не предполагается. Веб-сервис реализуется средствами выбранного вами языка программирования.

СДАЧА РАБОТЫ И ОЦЕНИВАНИЕ

В Moodle на проверку отправляется ссылка на сайт и ссылка на репозиторий GIT.

Время выполнения задания до 16 часов, включая самостоятельную работу.

За задание выставляется **до 16 баллов**:

- 4 балла: корректная реализация веб-сервиса для сохранения новых данных форм и изменения отправленных ранее данных при авторизации;
- 4 балла: корректная реализация с использованием JavaScript, включая валидацию на клиенте и отображение ошибок при валидации на сервере, отправку новых данных и отображение логина и пароля, авторизацию и редактирование введенных ранее данных;
- 4 балла: фоллбек на корректно работающий вариант реализации предыдущей задачи в случае если JavaScript в браузере отключен;
- 4 балла: реализация веб-сервиса с учетом принципов REST, единая точка входа, использование методов POST для сохранения новых данных и PUT по динамическому адресу для изменения ранее введенных данных.

Допускается не полное выполнение задания. Не допускается просрочка выполнения задания (просроченное задание оценивается в 0 баллов).

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

Ознакомьтесь с содержанием лекции 11-12 и учебным фреймворком. Проведите рефторинг кода и используйте учебный фреймворк для организации модулей или разработайте свой фреймворк для сопоставления URL-страниц модулям, шаблонизации. Код валидации и работы с базой данных для веб-сервиса и формы на бекэнде не должен дублироваться. Обработчик JavaScript назначайте на форму динамически при отображении страницы чтобы изменить поведение формы, в случае если в браузере включен JavaScript.

Если вы выполняете задание не на PHP, то в данном задании предполагается самостоятельная разработка фреймворка средствами выбранного языка программирования (без использования высокоуровневых фреймворков типа Django на Python или Express на NodeJS), для обеспечения единой точки входа, модульности, разделения шаблонов HTML/CSS, файлов JS и бизнес-логики бекэнда.