

Сопроводительный текст к 1му этапу проекта «Система управления заказами»

1. Проект разрабатывается на ZF3 + Twitter Bootstrap 4 + DataTables + jQuery.

На данный момент полностью реализована функциональность п.п. 3-9 по ТЗ («Система управления заказами - 1 этап (1).docx» - прилагается в директории doc архива). Порядок выполнения пунктов оговорен с заказчиком. Функциональное тестирование уже реализованных пунктов заказчиком еще не проводилось. Тестовое размещение проекта на время реализации: <http://shop.yura.dp.ua>

2. Т.к. время и финансирование на разработку ограничено, а система предназначена для использования самим заказчиком, и в будущем его нанятыми работниками, я максимально придерживался принципа YAGNI. В частности, не обращал внимания на XSS уязвимости (кроме StripTags в фильтрах), а валидаторы написаны только для тех полей, которые вводятся вручную, а не реализованы методом выбора (т.е. я подразумеваю, что заказчик или его нанятые работники не станут намеренно ломать свою систему XSS атаками, и не станут подменять значения полей в формах). Так же, я не обрабатываю дополнительно некорректно введенные данные в URL (к примеру, если в URL подставить значение сущности, которую выбираем из базы, а такой сущности с данным ID не существует, получим просто Exception, а не 404). Считаю такие обработки излишними для текущей задачи. Но я вполне понимаю, что такие обработки необходимы, если система будет открыта для общего доступа.

3. Структуры двух БД предоставлена заказчиком (это указано в ТЗ, и судя по всему, с этими БД уже работает другое ПО заказчика), я не могу менять уже существующие поля, только добавлять новые поля согласно ТЗ, либо по моему усмотрению. Структуру БД привожу в дампах («core.sql» - основная БД, «db2.sql» - дополнительная БД, в директории doc архива) для того, чтобы можно было проверить корректность полей по умолчанию при добавлении/изменении данных в БД. А также проверить корректность валидаторов (тип, и длину строковых полей). Что касается валидаторов, были учтены не только типы полей в таблицах БД, а и логику данных. Т.е. если такие поля, как ИНН, ОКПО и т.д. имеют в БД тип int, но не могут принимать отрицательных значений или значений с разделителями, то я проверяю в валидаторах такие поля не на IsInt, а на Digits (таких проверок подавляющее большинство, если вообще не все).

4. Пагинация реализована только средствами библиотеки datatables (это описано в ТЗ). Заказчик уже понял ошибку такого требования реализации задачи в ТЗ, но оценка по времени и стоимости на 1й этап проекта уже дана по такому описанию ТЗ, страничная подгрузка данных в таблицы на данном этапе не предусмотрена и будет реализована на следующем этапе.

5. Есть некоторые небольшие несоответствия ТЗ и реализации. Но все они согласованы с заказчиком. К примеру, заказчик по ТЗ хотел реализацию таблиц библиотекой bootstrap-table, я же его убедил в том, что datatables библиотека имеет больший функционал и возможности. Заказчик, исследовав возможности datatables согласился на использование именно datatables. Так же, в выпадающих полях с поиском в ТЗ описана древовидная структура с отступами, но была обговорена и реализована функциональность вывода таких данных через слеш одной строкой (путь к категории и конечная категория). Так же, в некоторых формах отсутствуют плейсхолдеры. Изначально они были на всех формах, но на некоторых заказчик попросил их убрать, т.к. они мешают визуальному просмотру. Так же есть таблица, в которой галка «выбрать все» выбирает только строки текущей страницы, а не всей таблицы. Это тоже просьба заказчика, т.к. ему удобно пользоваться такой галкой только для текущей страницы. Так же отображение категорий помечаются не разноцветно, как указано в ТЗ, а соответственно статусу (включена, либо выключена), что тоже одобрил заказчик.

6. К внешнему виду лишь одно требование по ТЗ: адаптивность к мобильным устройствам, поэтому дополнительное использование стилей минимальное.