# Задание к курсовой работе «Разработка прототипа web-приложения» по курсу «Современные средства программирования» (ПРОЕКТ)

Для выполнения данной лабораторной работы необходимо создать и защитить набор моделей (диаграмм), описывающий заданную в варианте предметную область (ER, eEPС, IDEF0, диаграммы UML (use case и др.), блок-схемы и/или др.).

Необходимо подготовить описание реализуемого прототипа (идеальный вариант – все спроектированное реализовано в web-приложении). Необходимо предоставить отчет в произвольной форме, который бы описывал структуру и назначение реализованных web-компонентов, их связь с компонентами web-приложения, список пользователей, работающих с системой, набор и структуру данных.

Web-приложение должно включать:

1. набор динамических страниц PHP с использованием технологий HTML, хотя бы одна страница должна включать CSS , JavaScript, AJAX;
2. несколько таблиц в MySql (в одной или более БД), кроме MySql можно дополнительно использовать другую СУБД (необходимо предварительно согласовать), логическая и физическая модели БД прототипа;
3. home-страницу с несколькими пунктами меню (можно обычные ссылки) и авторизацией;
4. два типа пользователей с разными правами (например, администратор, который может записывать в БД, и неавторизованный пользователь, который только считывает информацию);
5. страницу (страницы) с вводом (изменением, добавлением) данных как минимум в 2 связанные таблицы БД одновременно (предполагается, что данные связаны, например, в одну таблицу записываются данные о персоне, а в связанную – список контактных телефонов);
6. вывод данных по разным запросам (возможно, один запрос будет представлен графиком), как минимум один запрос должен быть по нескольким «связанным» таблицам БД; организация «постраничного» вывода таблиц – по желанию.
7. Асинхронный запрос (AJAX) хотя бы к одной таблице БД.

Требования:

1. Создать и скрипт на SQL для создания и начального заполнения БД. Начальное заполнение БД должно позволить без дополнительных настройки и ввода данных проверить все функции системы.
2. Должен быть реализован хотя бы один триггер (хранимая процедура в БД).
3. Осуществить разбиение скрипта на модули php.
4. Организовать сессию для авторизованного пользователя (2 мин), после 2-х минут неактивности предупреждать об окончании сессии и продолжать работу как с незарегистрированным пользователем.
5. Осуществить запись одного или нескольких cookie, сохраняемых несколько минут
6. Использовать как минимум по 1 разу методы Get и Post.
7. Приложение должно обрабатывать ошибки пользователя (тест-план не требуется для предъявления).
8. Необходимо использовать один из общеупотребимых стандартов интерфейса по элементам управления, цветовой гамме и т.п. (например, на основе IBM Common User Access). Все решения по интерфейсу, ограничениям данных должны быть продуманы и представлены в отчете.
9. «Украшательства» HTML приветствуются, но не обязательны.
10. Подготовить демонстрацию одновременной работы пользователей с базой данных (блокировка чтения/записи): при отсутствии блокировок – ошибка, при наличии – верно
11. Подготовить демонстрацию изменения на стороне клиента кода страницы (кода, ответственного за проверку вводимых данных на стороне клиента на допустимость), которое «нейтрализуется» дублирующей проверкой на стороне сервера.
12. Подготовить план (сценарий) работы с приложением, демонстрирующий все функции и требования.

.