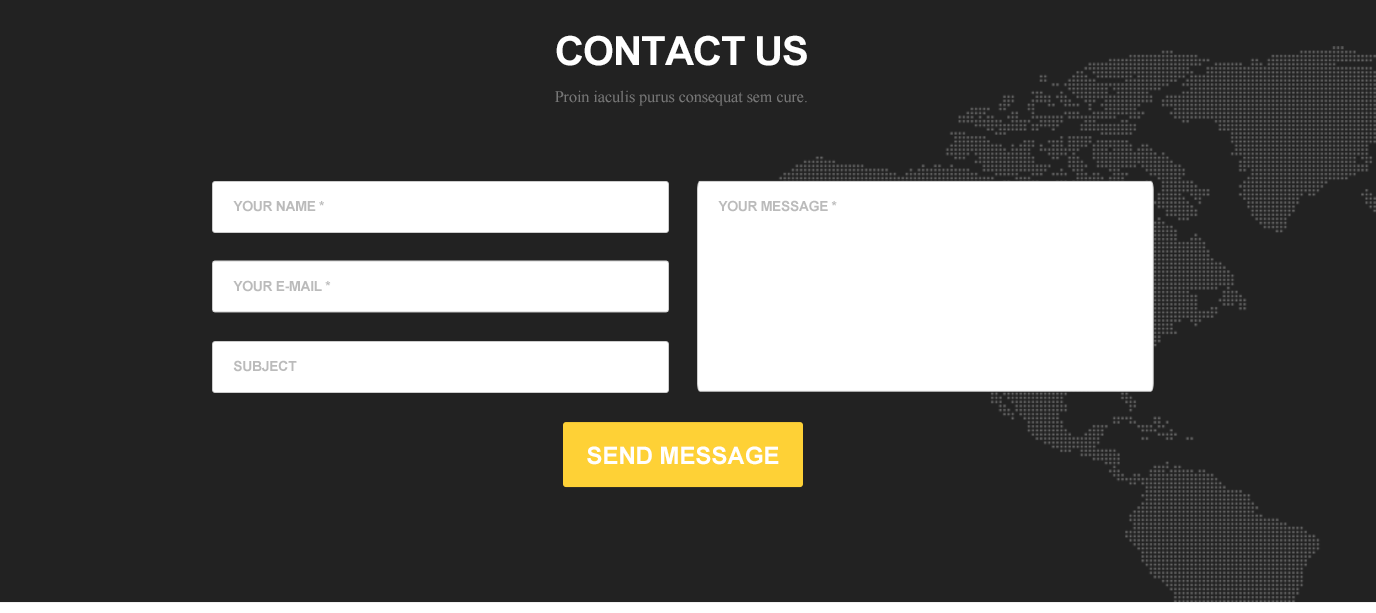
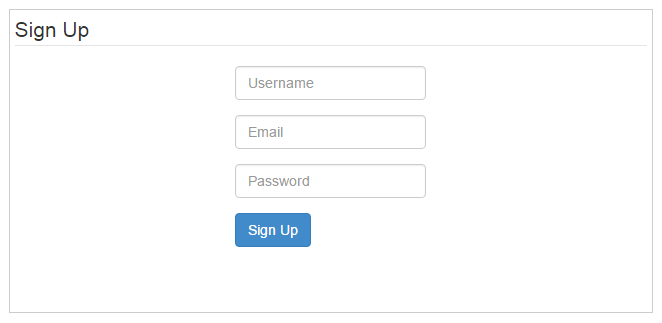
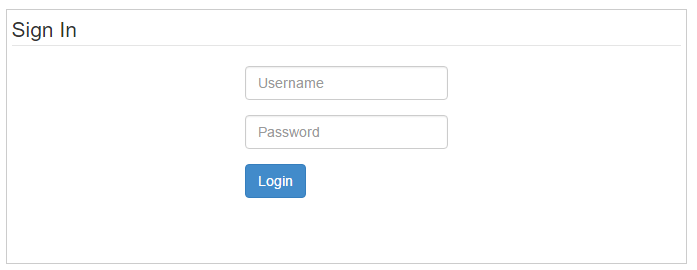
1. Диаграмма сущностей:

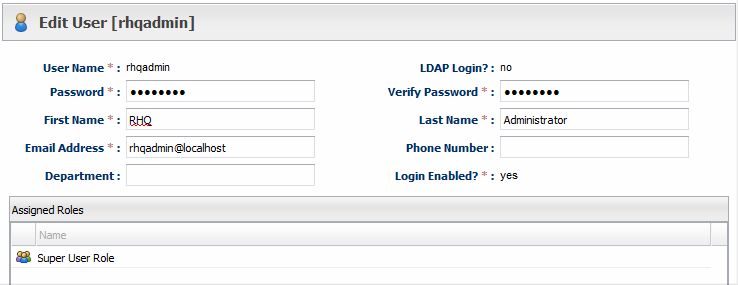


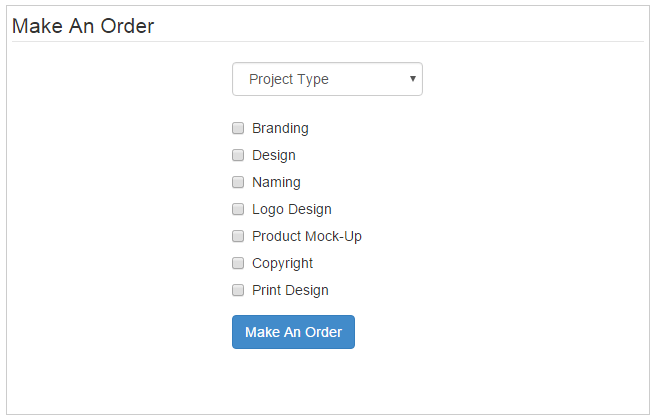
1. Прототипы форм:











Phone NotificationPhoneDocument

1. Описание действий в системе (название функции (входные данные): выходные данные // описание):

createClient(data): clientData // создать пользователя (заказчика)

getClient(IdClient): clientData // получить данные пользователя

updateClient(IdClient, data): clientData // обновить информацию о пользователе

deleteClient(IdClient): void // удалить пользователя

createProjectRequest(data): projectRequestData // создать заказ

getProjectRequest (IdProjectRequest): projectRequestData // получить данные о заказе

updateProjectRequest (IdProjectRequest, data): projectRequestData // обновить информацию о заказе

deleteProjectRequest (IdProjectRequest): void // удалить заказ

createDocument (data): DocumentData // создать документ

getDocument (IdDocument): DocumentData // получить данные о документе

updateDocument (IdDocument, data): Document Data // обновить информацию о документе

deleteDocument (IdDocument): void // удалить документ

createProjectDocument (data): ProjectDocumentData // создать проектную документацию

getProjectDocument (IdProjectDocument): ProjectDocumentData // получить данные о проектной документации

updateProjectDocument (IdProjectDocument, data): ProjectDocumentData // обновить проектную документацию

deleteProjectDocument (IdProjectDocument): void // удалить проектную документацию

createContact (data): ContactData // создать контакт

getContact (IdContact): ContactData // получить данные о контакте

updateContact (IdContact, data): ContactData // обновить контакт

deleteContact (IdContact): void // удалить контакт

createCalendar (data): CalendarData // создать календарь

getCalendar (IdCalendar): CalendarData // получить данные о календаре

updateCalendar (IdCalendar, data): CalendarData // обновить календарь

deleteCalendar (IdCalendar): void // удалить календарь

createProjectCostCenter (data): ProjectCostCenterData // создать смету

getProjectCostCenter (IdProjectCostCenter): ProjectCostCenterData // получить данные о смете

updateProjectCostCenter (IdProjectCostCenter, data): ProjectCostCenterData // обновить смету

deleteProjectCostCenter (IdProjectCostCenter): void // удалить смету

1. Иерархическая структура работ:
2. Создание логики

1.1) Формулирование задач

1.2) Утверждение ТЗ

1.3) Создание структуры кода

1. Разработка БД

2.1) Выбор БД

2.2) Установка БД

2.3) Создание структуры БД

2.4) начальное наполнение БД

1. Создание интерфейса

3.1) Создание прототипов форм

3.2) Выбор единой стилистики

3.3) Создание интерфейса сайта

3.3.1) Создание главной страницы

3.3.2) Создание страницы документа

3.3.3) Создание страницы администратора

3.3.4) Создание страницы входа

3.3.5) Создание страницы пользователя

1. Разработка Back-End’а сайта

4.1) Написание модулей

4.2) Модульное тестирование

4.3) Объединение модулей

1. Разработка Front-End’а сайта  
   5.1) Создание дизайна сайта  
   5.2) Верстка макета сайта
2. Разработка тестов

6.1) Выбор стратегии тестирования (серый, белый, чёрный ящики)

6.2) Создание тестов ручного тестирования

6.3) Создание тестов автоматического тестирования

1. Тестирование

7.1) Проведение ручного тестирования

7.1.1) Тестирование функциональности

7.1.2) Тестирование ограничений

7.1.3) Приёмные испытания

7.1.4) Инсталляционное тестирование

7.2) Проведение автоматического тестирования

1. Документирование

8.1) Создание сопровождающей литературы

8.2) Документирование функциональных возможностей программы

1. Внедрение на сервер

9.1) Выделение сервера

9.2) Установка ПО

9.3) Запуск работы