Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего профессионального образования

‹‹Уральский федеральный университет имени первого Президента России

Б.Н.Ельцина››

Кафедра ‹‹Интеллектуальных информационных технологий››

дисциплина ‹‹WEB программирование››

**Курсовая работа**

**«Сайт найма детективов»**

Преподаватель: Свинцов Д. В.

Студент: Шульц М.А.

Группа: ФО-340001

Екатеринбург

2016

# **Введение**

На данный момент прогресс в области веб-программирования достиг очень многого. Тяжело перечислить все возможности интернета. Но как ни странно, и по сей день, благодаря веб-разработке, можно реализовать много интересных новых идей.

Как известно, существует огромное множество сайтов, где люди могут найти себе работу. Обычно там опубликовывают вакансии, касающиеся различных сфер. И в какой-то момент мне стало интересно, а найду ли я вакансию частного сыщика. Ведь это довольно-таки узкая специальность. Как я и ожидал, мне очень часто выпадало сообщение “Не найдено ни одной вакансии”

Именно по этой причине я решил разработать сайт, где люди смогли бы найти себе подобную работу. И в то же время наниматели увеличили вероятность решения их проблем.

**Описание**

Сайт содержит следующий функционал:

* Авторизация
* Блог, в котором будут размещены объявления людей
* Возможность предложить свою услугу на объявление

**Обоснование**

Разумеется, мне пришла в голову мысль, что частного детектива можно нанять и другими способами. Например, многие детективные агентства имеют свои информационные порталы, где можно узнать и телефоны, и расценки. Но я увидел в моей идее преимущества:

* Возможность найти работу, как детективным агентствам, так и людям
* Возможность детективам выбирать дела, не отказывая людям
* Возможность нанимателям выбрать сразу несколько детективов
* Нанимателям нужно выбирать, а не искать
* Возможность анонимного объявления

Но существует большой недостаток: Могут найтись люди, которые будут опубликовать несерьезные проблемы.

**Инструкция пользователя**

Структура сайта состоит из семи страниц:

**Главная страница (index.jinja2)**

Данная страница содержит все посты добавленные пользователями. Также она позволяет:

* Перейти на страницу регистрации
* Войти зарегистрированному пользователю
* Добавить пост, как зарегистрированному пользователю, так и незарегистрированному
* Прочитать пост любому человеку
* Отправить зарегистрированному как детектив человеку в качестве ответа на пост предложение услуги

**Страница поста (read.jinja2)**

Содержит информацию о посте:

* Заголовок
* Описание

**Страница добавления поста (add.jinja2)**

Содержит форму для заполнения поста

**Страница регистрации (register.jinja2)**

Содержит форму для заполнения данных пользователе

* Логин (Отслеживается уникальность)
* Пароль (Для безопасности он хэшируется)
* Почта
* Выпадающий список, позволяющий выбрать роль пользователя

**Страница просмотра заявки(readpropose.jinja2)**

Страница показывает данные заявки. Более того, если пост был сделан анонимно, то заявки могут просматривать всею Но если пост был сделан зарегистрированным пользователем, то предложения просматривать может только он.

**Страница предложения услуги (propose.jinja2)**

Это страница содержит форму для заполнения заявки. Перейти на эту страницу может только зарегистрированный как детектив пользователь.

**Инструкция программиста**

**База данных**

Чтобы впервые начать работать с проектом, необязательно иметь заполненную базу данных, так как функционал позволяет ее заполнить в любой момент. Более того, она после каждого использования сохраняется.

База данных состоит из следующих сущностей:

* Пользователи (Users)
* Дела (Posts)
* Роли пользователей (Roles)
* Связь пользователей и ролей (UsersRoles)
* Заявки (Request)

На рисунке 1 вы можете посмотреть, как взаимосвязаны сущности

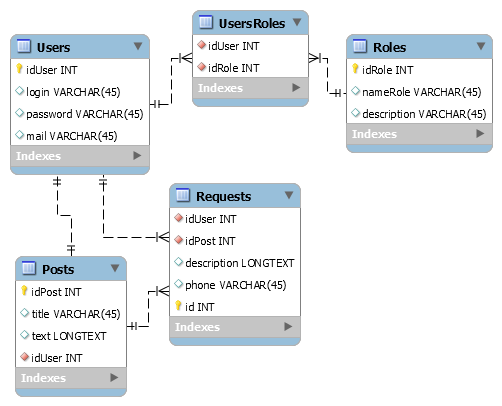


Рисунок 1 Диаграмма связей базы данных

**Необходимые библиотеки**

В ходе работы были использованы следующие библиотеки и модули:

1. Для запуска сервера:

* wsgiref (сам запуск)
* pyramid.config (все настройки, пути и представления для сервера)
* pyramid\_jinja2 (работа с шаблонизатором jinja2)
* pyramid\_tm (менеджер транзакций)

1. Для работы с базой данных

* sqlalchemy (создание и весь функционал базы данных)
* pyramid\_sqlalchemy (удобное создание сессий)
* passlib (хэширование паролей)

1. Для работы с формами

* wtforms ( содержит готовые формы)

1. Для авторизации и аутентификации:

* Pyramid.authorization
* Pyramid.authetication

**Описание классов**

* **\_\_init\_\_.py**

Содержит все необходимое для запуска сервера: представление jinja2 в виде каталога адресов, views для каждого пути

* **models.py**

Содержит структуру каждой сущности базы данных

* **views.py**

Содержит функционал каждой страницы сайта

**Выводы:**

В данной работе были получены следующие результаты:

* Был сделан сайт, построенный на фреймворке pyramid
* Была спроектирована база данных
* Был построен веб-сервер, на котором сайт держится
* Был освоен паттерн MVC

В ходе работы были получены следующие навыки:

* Работа с базой данных
* Верстка
* Умение работать с языком Python
* Понимание работы web-фреймворков и паттернов

**Список используемой литературы:**

1. Основы веб программирования [Электронный ресурс] URL: <http://lectures.uralbash.ru/index.html>[http://lectureswww.readthedocs.io/6.www.sync/3.framework/pyramid/0.introduction.htmlhttp://lectureswww.readthedocs.io/6.www.sync/3.framework/pyramid/0.introduction.htmlhttp://lectureswww.readthedocs.io/](http://lectureswww.readthedocs.io/6.www.sync/3.framework/pyramid/0.introduction.htmlhttp:/lectureswww.readthedocs.io/6.www.sync/3.framework/pyramid/0.introduction.htmlhttp:/lectureswww.readthedocs.io/6.www.sync/3.framework/pyramid/0.introduction.html)
2. Документация pyramid, sqlalchemy [Электронный ресурс] URL:

<https://readthedocs.org/>