



Рисунок 3 – Страница просмотра компании

Н.А. Навныко (УО «ГГУ им Ф. Скорины», Гомель)

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

АРХИТЕКТУРА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РЕКРУТИНГА СОТРУДНИКОВ

Разработанное приложение основано на CMS фреймворке Django. Выбор был сделан именно в сторону Django из-за ряда преимуществ. Во-первых, Django написан на языке Python – одном из наиболее распространённых интерпретируемых языков. Python позволяет выполнять большое количество разнообразных задач, поскольку является объектно-ориентированным языком программирования. Также он отличается динамичностью и кроссплатформенностью, то есть без проблем переносится на другую платформу (реализован практически на всех ОС и платформах). Во-вторых, Django предоставляет возможности использования ORM (Object-relationship mapper), что позволяет в большинстве случаев обходиться без SQL-синтаксиса в выражениях, что автоматически снижает риск появления SQL-injection уязвимости. Достоинствами

фреймворка являются автоматически генерируемая административная часть, множество классов для форм, представлений и моделей, позволяющие быстро и эффективно создать качественный сайт.

Django строит приложение на основе архитектуры MTV (Модель-Шаблон-Представление), которая соответствует шаблону MVC, где часть представлений представлена шаблонами, а часть контроллеров – представлениями django.

Шаблон Django – текстовый файл, содержащий html теги и `templatetags` Django. Шаблоны принято наследовать от базового. В нашем приложении базовый шаблон включает в себя основное меню сайта и баннеры (header) и ссылки на статьи, контактные данные администраторов сайта, ссылки на социальные сети (footer). Остальные шаблоны расширяют базовый, наполняя его контентом. Шаблон – это просто текстовый файл. Шаблон содержит переменные, которые будут заменены значениями при выполнении шаблона, и теги, которые управляют логикой шаблона.

Представления в Django – это контроллеры, обработчики запросов. Каждое представление закрепляется за одним или несколькими URL. В сам контроллер запрос поступает с дополненными Django атрибутами. Каждое представление должно возвращать http ответ (HTTPResponse), принимаемый браузером. Представления в Django можно описать функциями или классами, расширяющими базовые (ModelView, TemplateView и т. д.).

Веб-приложение подразделяется на две составные части: административную и основную.

Административная часть в Django Framework создается автоматически (если проект был создан командой `python manage.py startproject`). Административная часть приложения предназначена для управления данными. При регистрации модели администратору сайта доступен список объектов этой модели, которые он может редактировать или создавать новые объекты. Административная часть в Django Framework может быть изменена под любые нужды. В текущем приложении она используется для управления данными из базы данных, назначения прав пользователям.

Основная часть приложения (интерфейс) создается непосредственно шаблонами, представлениями и формами. Описывая формы Django, программист уменьшает количество кода HTML и получает возможность легко составлять структуру формы и методы ее обработки. Формы Django также позволяют автоматически осуществлять валидацию данных через описанные поля форм и их классы.

Детали реализации компонентов приложения обсуждаются в докладе.