

Но, несмотря на эти недостатки, многие люди используют скрытые криптоконтейнеры для хранения важных данных, наивно веря в их превосходящую надёжность и в то, что их невозможно обнаружить. Следовательно, работа в этом направлении важна: она позволяет определить, является ли техника сокрытия одного контейнера в другой надёжной с точки зрения стеганографии, и если нет, то стоит рассматривать варианты, как сделать скрытую часть ещё более трудно обнаруживаемой.

**Н.С. Седых** (УО «ГГУ им. Ф. Скорины», Гомель)  
Науч. рук. **В.Н. Кулинченко**, старший преподаватель

### **РАЗРАБОТКА ПРОЕКТА ВИЗУАЛИЗАЦИИ СТАТИСТИКИ ПОСЕЩЕНИЙ WEB-СЕРВИСОВ ДЛЯ ПРЕДПРИЯТИЯ «ЭПАМ СИСТЕМЗ»**

Интернет-реклама – реклама, размещаемая в сети Интернет. В тоже время, это современное направление маркетинга, которое открывает организациям иные каналы привлечения новых пользователей и расширяет площадь их влияния. Здесь появляются следующие задачи: хранение, отображение и управление статистикой посещений для последующего использования.

Для хранения данных используется СУБД MySQL. Предполагается, что приложение будет использоваться большим количеством пользователей, следовательно, база данных будет находиться под нагрузкой, что может привести к снижению скорости или отказу в обслуживании. Для решения этой проблемы необходимо распределить нагрузку на базы данных с помощью синхронизации данных, которую также называют репликацией. На рисунке 1 показана схема распределения нагрузки базы данных.

Проект визуализации статистики основан на архитектуре клиент-сервер с использованием REST-подхода. REST – метод взаимодействия компонентов распределённого приложения в сети Интернет, при котором вызов удаленной процедуры представляет собой обычный HTTP-запрос (обычно GET или POST; такой запрос называют REST-запрос), а необходимые данные передаются в качестве параметров запроса. Этот способ является альтернативой более сложным методам, таким как SOAP, CORBA и RPC. На рисунке 2 показана схема работы REST.

Серверная часть приложения реализована с использованием Python фреймворка Django, использующий шаблон проектирования MVC. Для работы с базой данных Django использует собственный ORM, в котором

модель данных описывается классами Python, и по ней генерируется схема базы данных. Также можно указать дополнительные настройки, например имя таблицы. При необходимости можно переопределить методы сохранения, удаления, обновления записей таблицы и т. д.

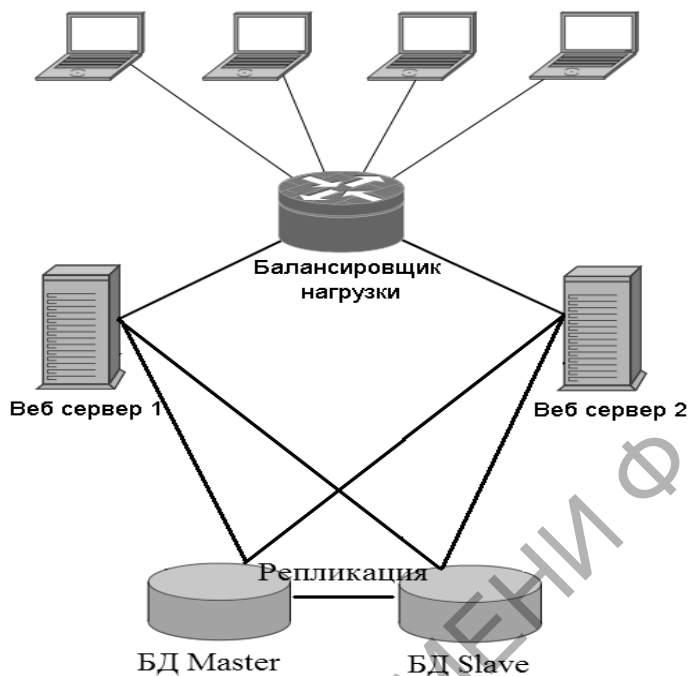


Рисунок 1 – Схема балансировки нагрузки БД

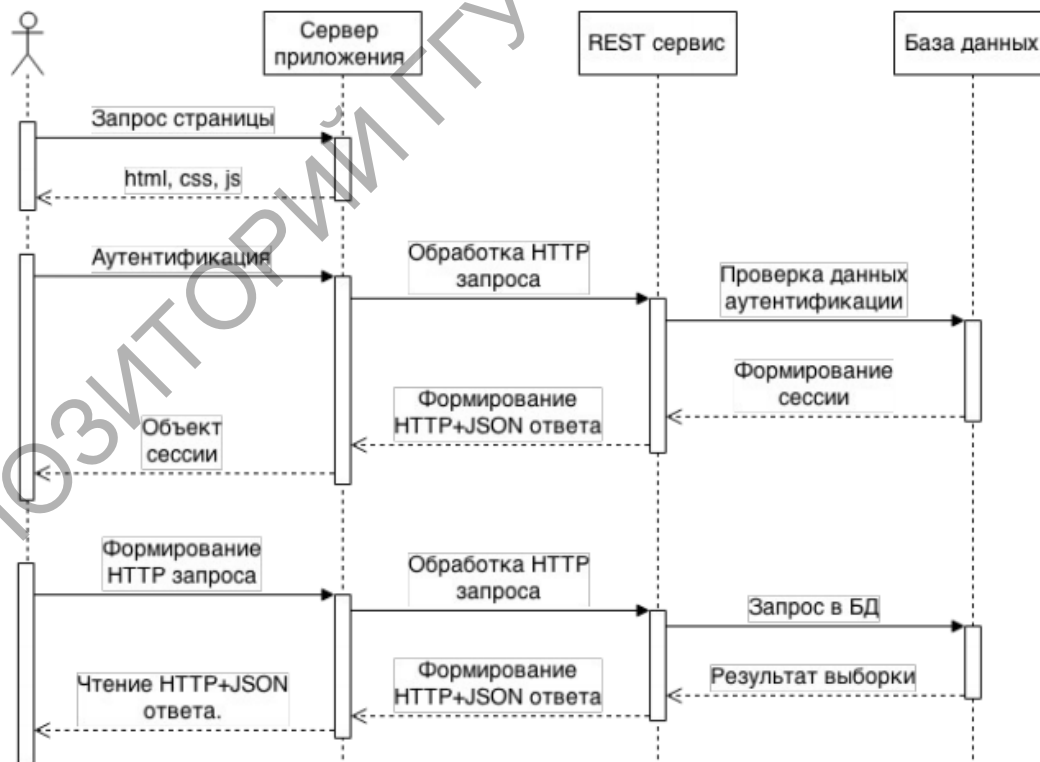


Рисунок 2 – Схема работы REST API

Для реализации REST API используется фреймворк для Django – Tastypie. Чтобы предоставить клиенту API для работы с моделью или другой частью приложения, необходимо описать класс ресурса с именем, по которому клиенты будут обращаться к серверу. При этом ресурс предоставляет 5 HTTP методов: GET, POST, PUT, DELETE, PATCH. На рисунке 3 показана структура приложения клиента.

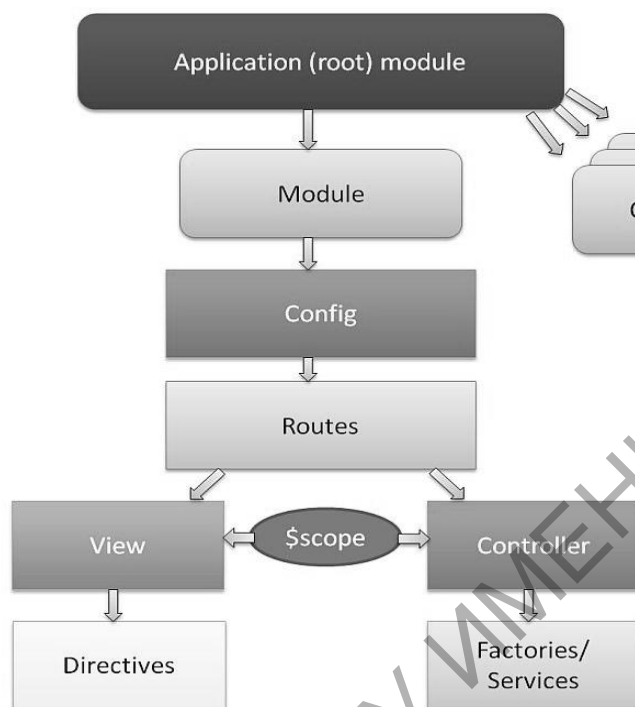


Рисунок 3 – Структура приложения клиента

Клиент – single page application, реализованный при помощи фреймворка AngularJS. Все приложение разбито на модули, имеющие собственную конфигурацию и маршрутизацию. Обычно, под маршрутом понимается адрес, по которому пользователь получает страницу.

**Н.С. Седых** (УО «ГГУ им. Ф. Скорины», Гомель)  
 Науч. рук. **В.Н. Кулинченко**, старший преподаватель

**РЕАЛИЗАЦИЯ ПРОЕКТА ВИЗУАЛИЗАЦИИ  
 СТАТИСТИКИ ПОСЕЩЕНИЙ WEB-СЕРВИСОВ  
 ДЛЯ ПРЕДПРИЯТИЯ «ЭПАМ СИСТЕМЗ»**

Одной из задач DMP-системы является визуализация статистики посещения Web-сервисов. Это позволяет пользователям проводить анализ данных для осуществления оптимизации медиазакупок и планирования