

Выпускная квалификационная работа на тему:

РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ИНТЕРНЕТ-МАГАЗИНОМ ПО ПРОДАЖЕ ПОСУДЫ КОМПАНИИ ООО «ГРАНД- ЛЮКС»

Дикий Г. П.
Руководитель: Бондарев Е. С.

Цель работы

Разработка системы управления интернет-магазином по продаже посуды компании ООО «Гранд-Люкс».

Разработка двух интернет-магазинов на основе данной системы управления. Один из них с поддержкой оптовой и розничной торговли, а второй, только розничной торговли.

Основные задачи

- Возможность работы с интернет-магазином как оптовым, так и розничным пользователям
- Возможность синхронизации с 1С:Предприятием
- Возможность фильтрации товаров
- Возможность поиска товаров
- Возможность каталогизации товаров
- Разработка модуля акций
- Разработка модуля купонов
- Разработка модуля поступлений
- Разработка клиентской части с поддержкой адаптивного дизайна
- Внедрение

Этапы работы

- **Обзор существующих аналогов и выбор программной платформы**
 - *Постановка задачи*
 - *Анализ ПрО*
 - *Обзор популярных CMS*
 - *Обзор популярных фреймворков*
- **Проектирование системы управления**
 - *Проектирование схемы базы данных*
 - *Проектирование серверной части*
 - *Проектирование интерфейса*
- **Реализация**
 - *Реализация серверной части*
 - *Реализация клиентской части*
 - *Тестирование*
 - *Документация*
- **Внедрение**
 - *Конфигурация серверного оборудования и ПО*
 - *HTTPS, Celery и кэширование*
 - *Результат внедрения*

Анализ ПрО

Интернет-магазин – это сайт:

- реализующий торговлю посредством сети интернет;
- предоставляющий информацию о товаре или услуге;
- дающий возможность сформировать заказ из выбранных товаров или услуг;
- предоставляющий покупателю возможность доставки;
- выписывающий счет на оплату, который и является подтверждением заказа;

Преимущества интернет-магазина

- экономия времени и денег;
- возможность доставки точно по адресу покупателя;
- возможность избежать очередей и суеты в торговых центрах;
- возможность получить интересующий товар или услугу людям с ограниченными возможностями;
- снижение нагрузка на транспортную инфраструктуру.

Популярные CMS

	Wordpress & Woocommerce	1С-Битрикс	OpenCart
Стоимость	Бесплатно	Платно	Бесплатно
Возможность поддержки оптовых и розничных клиентов	Требуем изменения базовой части CMS	Требуем изменения базовой части CMS	Требуем изменения базовой части CMS
Возможность синхронизации с 1С:Предприятием	Требуется разработка	Присутствует	Требуется разработка
Возможность поиска товаров	Присутствует	Присутствует	Присутствует
Возможность добавления акций	Требуется разработка	Требуется разработка	Требуется разработка
Возможность добавления купонов	Присутствует	Присутствует	Присутствует
Возможность учета поступлений	Требуется разработка	Требуется разработка	Требуется разработка
Возможность фильтрации	Присутствует	Присутствует	Присутствует
Возможность каталогизации товаров	Присутствует	Присутствует	Присутствует

Популярные веб-фреймворков

	Django	Laravel	Ruby on Rails
Используется шаблон проектирования MVC	+	+	+
Использование ORM для работы с БД	+	+	+
Встроенная система шаблонов	+	+	+
Встроенная система миграций	+	+	+
Автоматическая генерация форм с автоматической валидацией	+	-	-
Встроенная система событий, с возможностью создания собственных событий	+	+	+
Встроенная система тестирования	+	-	-
Маршрутизация на основе регулярных выражений	+	+	-
Автогенерация панели администратора	+	-	-

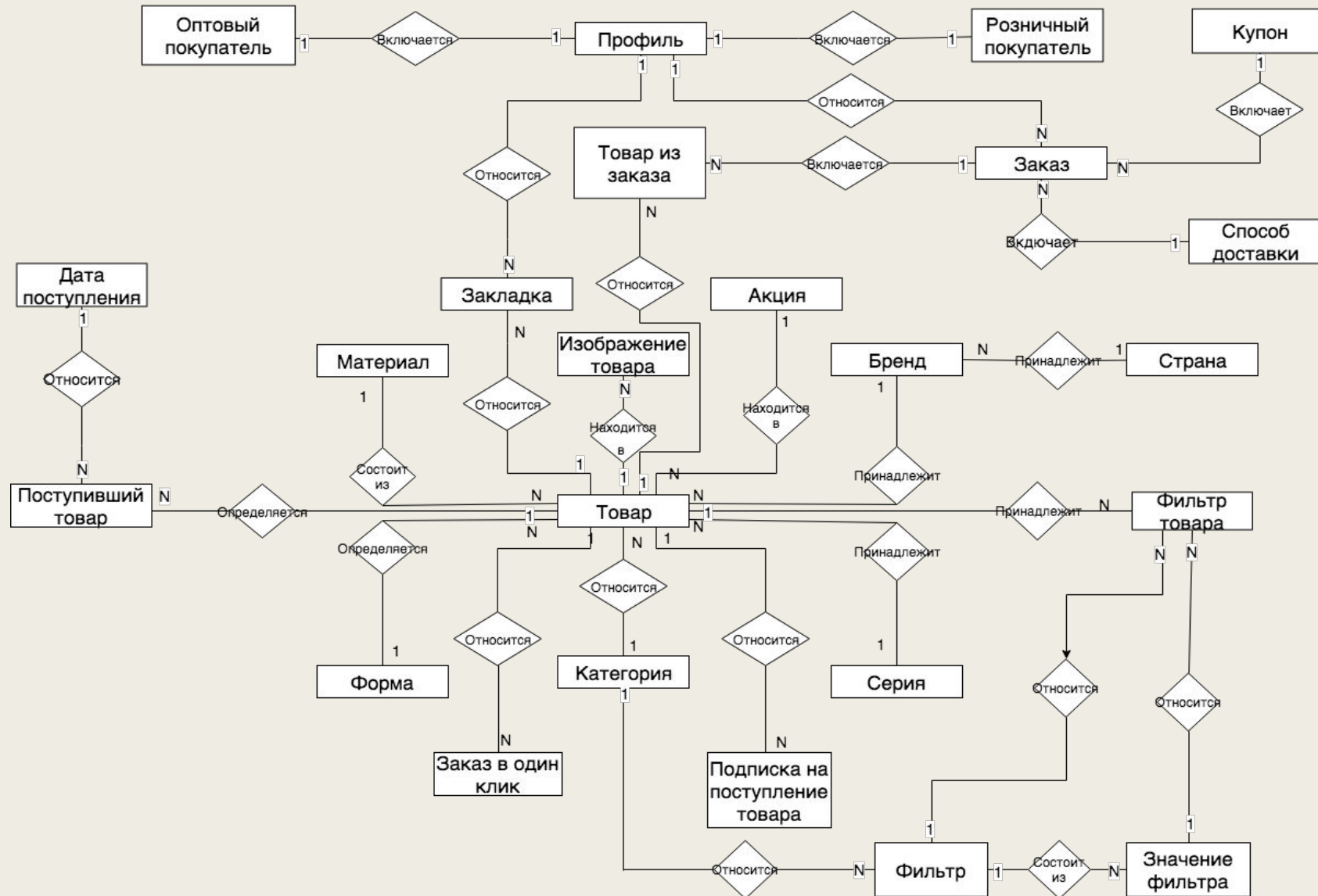
Django написан на языке Python, что позволяет использовать его для реализации поддерживающих бизнес-процессов.

Веб-фреймворк Django

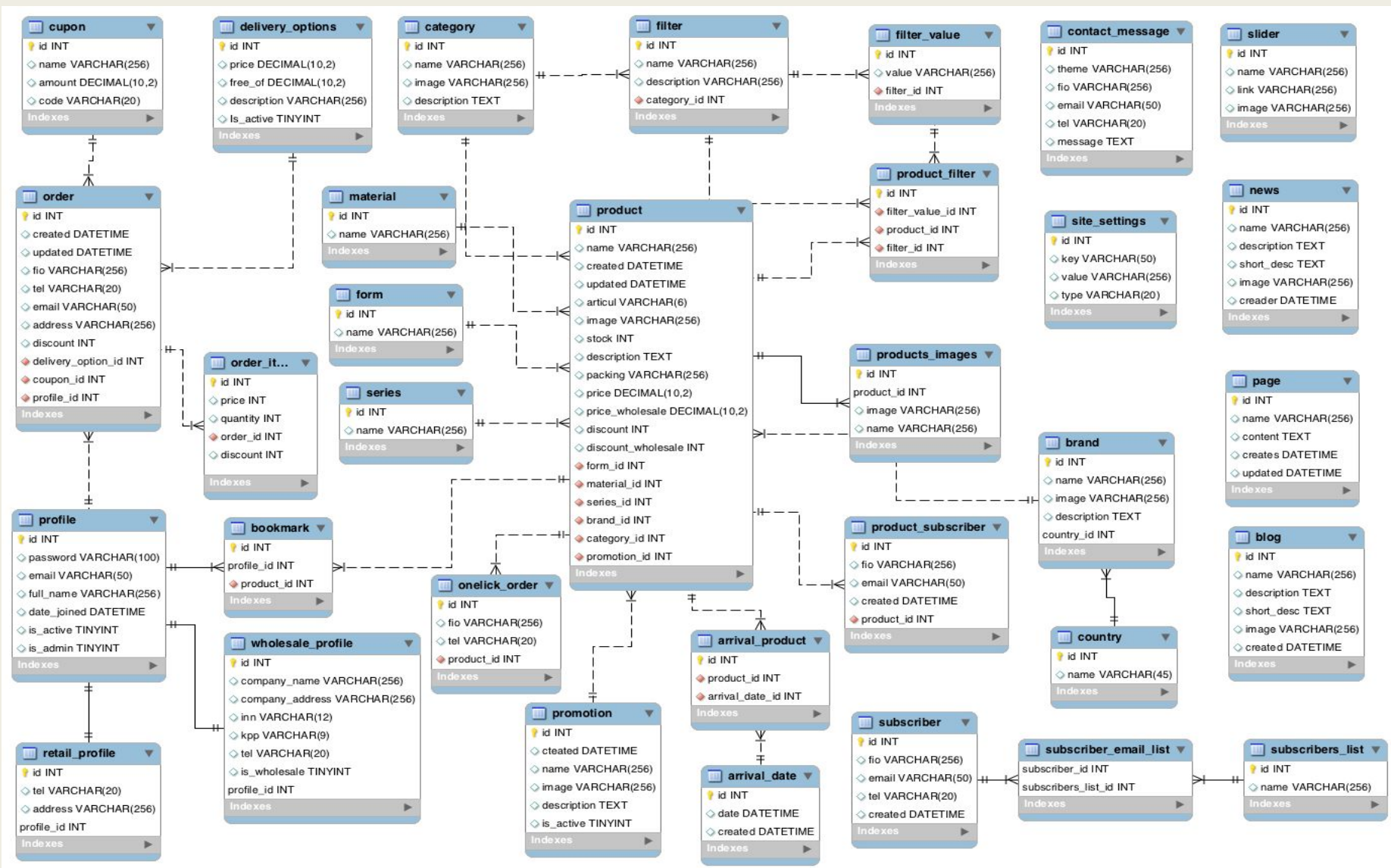
Сильные стороны:

- Автоматическая генерация панели администрирования на основании моделей;
- Автоматическая генерация форм с валидацией на основе моделей;
- Встроенная система кэширования;
- Поддержка системы событий, позволяющей подписываться на эти события и выполнять определенные действия;
- Возможность создания собственных событий;
- Встроенные функции сериализации объектов в формат JSON или XML;
- Встроенная система тестирования, основанная на библиотеке unittest;
- Встроенная возможность генерации файлов карт сайта sitemaps;
- Встроенная система защиты от базовых веб-уязвимостей;
- Расширяемая система регистрации и авторизации.

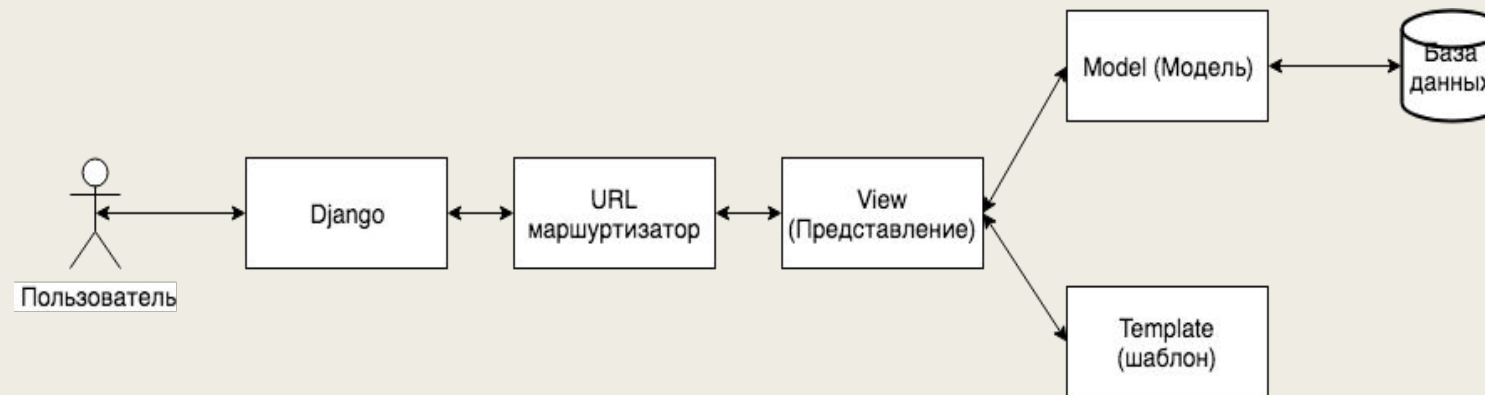
Инфологическая модель



Даталогическая модель



Django MVC (MVT)



Model-View-Controller (MVC, «Модель-Представление-Контроллер») — схема разделения данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо

- Django использует паттерн MVT (Model-View-Template)
- View – отвечает за получение данных от моделей и отрисовка шаблона (Template)
- Model – отвечает за работу с данными
- Template – отвечает за представление данных в HTML разметке

Django ORM и миграции

ORM (англ. Object-Relational Mapping) — технология программирования, которая связывает отношения БД с классами ООП, предоставляя возможность работы с данными, как с классами.

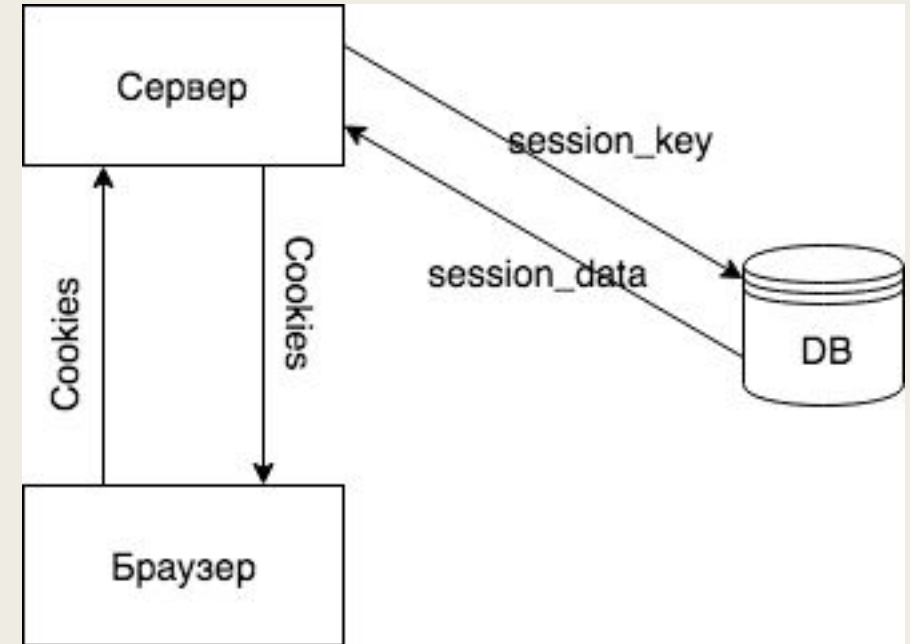
- + Скорость разработки возрастает
- + Возможность повторного использования текста программы с другими СУБД
- + Встроенные механизмы защиты от SQL – инъекций
- + Нет зависимости от определенной СУБД
- + Нет зависимости от языка запросов SQL
- сложные запросы к БД не всегда возможно реализовать с использованием ORM;
- ORM не всегда генерирует самые оптимальные запросы

Миграции – файлы, позволяющие синхронизировать изменения моделей с отношениями в БД.

- Django создает файлы миграции на языке Python
- Изменения обнаруживаются за счет создания виртуальной модели, созданной по файлам миграции и сравнения ее с реальной моделью

Сессии в Django

- Django при помощи сессий позволяет работать как с анонимными, так и с зарегистрированными пользователями
- По умолчанию сессии хранятся в БД
- Сессии используются для аутентификации и авторизации пользователей
- Автоматически загружаются из БД при каждом запросе
- Позволяют хранить дополнительную информацию о пользователе
- Подходят для создания корзины пользователей



Авторизация и аутентификация в Django

- По умолчанию компонент аутентификации и авторизации поставляется вместе с Django
- Возможности стандартного компонента:
 - Система регистрации и авторизации
 - Система сброса паролей
 - Поддержка системы прав пользователей
 - Поддержка групп пользователей
- Предоставляется возможность расширить стандартный компонент
- Django не хранит пароли в открытом виде, пароль в БД представлен в виде:
`<алгоритм>$<итерации>$<соль>$<хэш>`

Доступные алгоритмы хэширования паролей по умолчанию:

- `pbkdf2_sha256` (по умолчанию);
- `pbkdf2_sha1`;
- `crypt`;
- `argon2`;
- `bcrypt_sha256`;
- `bcrypt`;
- `sha1`;
- `md5`;
- `unsalted_sha1`;
- `unsalted_md5`

Проектирование интерфейса

Проектирование и реализация интерфейса производилась сторонним веб-дизайнером, на основании выявленных функциональных требований и возможностей

Требования к дизайнеру:

- использование белых, черных и серых цветов;
- «легкость» и простота интерфейсов;
- разработка с учетом адаптивности верстки;
- формат шаблона PSD.

Задача прототипирования и разработки дизайна следующих страниц:

- главная страница;
- страница товара;
- страница категории;
- страница корзины;
- страница новостей.

Синхронизация 1С:Предприятия с сайтом



Компания «ООО Гранд-Люкс» для ведения бухгалтерского учета всех товаров пользуется программным обеспечением 1С:Предприятие

- Синхронизация базы товаров происходит посредством таблиц формата XLS
- Синхронизация товаров инициируется 1С предприятием, который формирует POST запрос
- Используется заголовок HTTP Authorization для проверки прав доступа
- Предоставлена возможность загрузки XLS файла из панели администратора

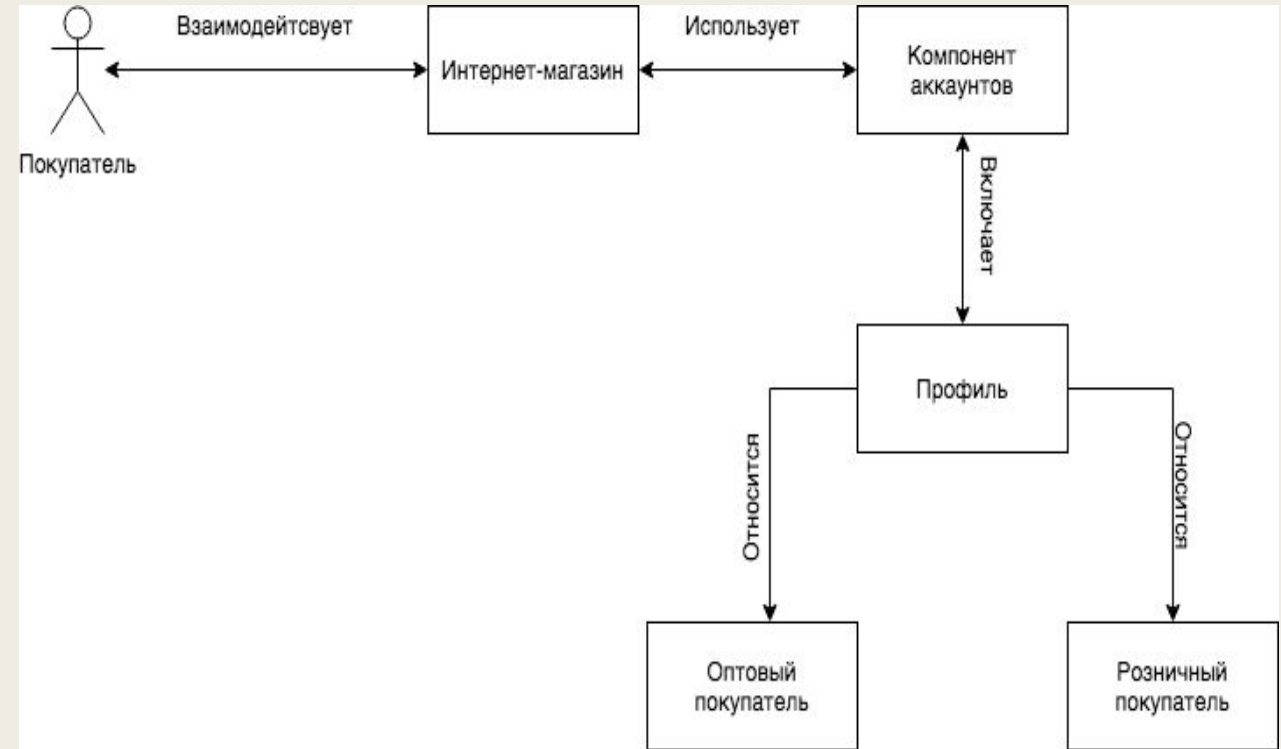
Недостаток:

- Интернет-магазин не может выступать в роли инициатора синхронизации

Модели профилей пользователей

Для поддержки работы как оптовых, так и розничных покупателей, были созданы следующие модели в Django: WholeSale, Retail, Profile.

- **Profile** – общая модель для розничных и оптовых пользователей, Реализует авторизацию пользователей.
- **WholeSale** – модель для хранения информации об оптовом покупателе
- **Retail** – модель для хранения информации об розничном покупателе



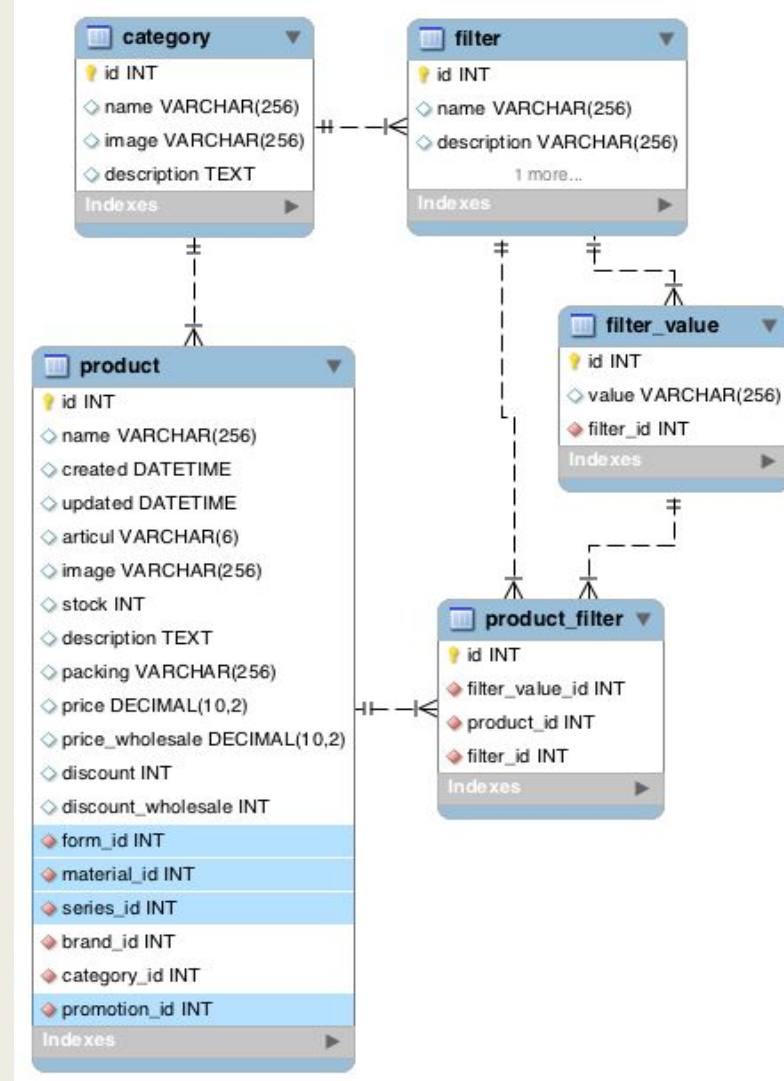
Фильтрация товаров

Для фильтрации товаров используется адресная строка параметрами, вида:

- `<filter_id>=<filter_value1>,<filter_value2>,<filter_valueN>`
- **filter_id** – идентификатор фильтра
- **filter_value** – значение фильтра

Алгоритм фильтрации:

- Получение параметров адресной строки
- Определение фильтров и их значений
- Поиск товаров в БД с соответствующими фильтром и значениями



НАЛИЧИЕ НОЖКИ

- ☐ На ножке
- ☐ Без ножки

ФОРМА

- ☐ Круглое
- ☐ Овальное
- ☐ Квадратное
- ☐ Прямоугольное
- ☐ Фигурное

РАЗМЕР

- ☐ До 26 см
- ☐ От 26 см до 36 см
- ☐ От 36 см

Каталог товаров

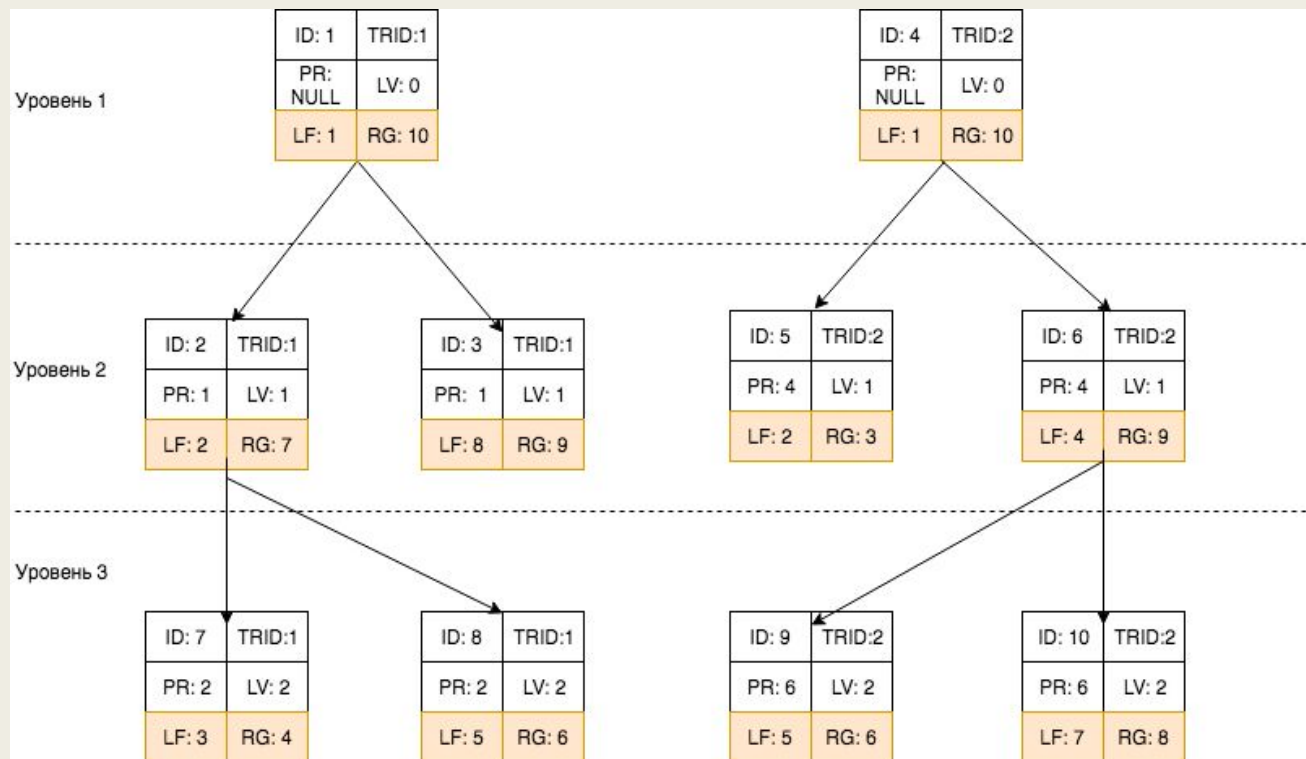
Меню сайта представлено в виде иерархической структуры со множеством категорий и подкатегорий.

- Для решения данной задачи был использован сторонний компонент **Django-MPTT**
- Для хранения иерархических структур используется **алгоритм вложенных множеств**

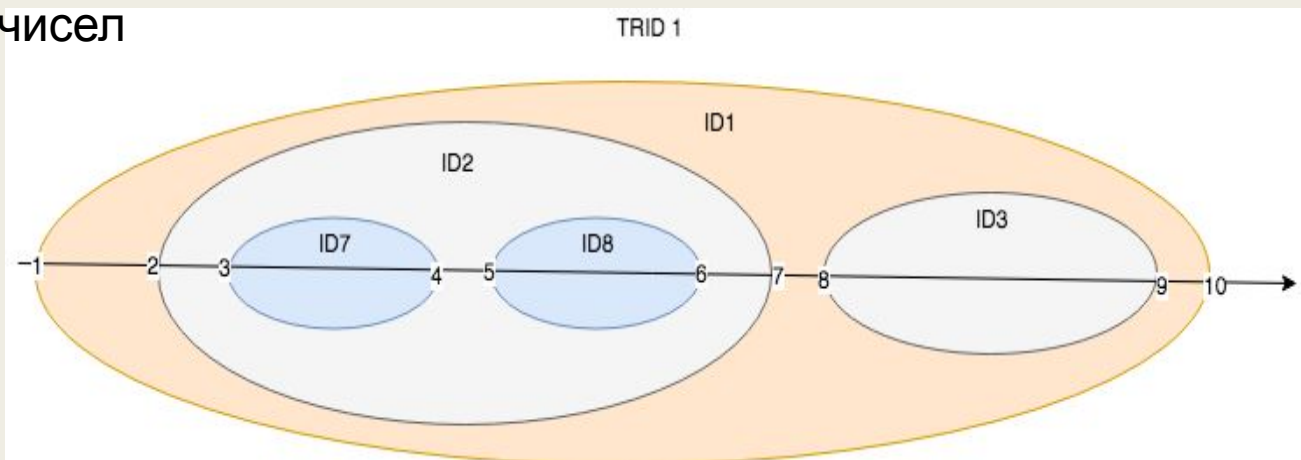
Обозначения представленные на рисунке:

- ID – идентификатор записи;
- TRID – идентификатор иерархической структуры;
- PR – идентификатор родительской записи;
- LV – уровень вложенности;
- LF – левый ключ;
- RF – правый ключ.

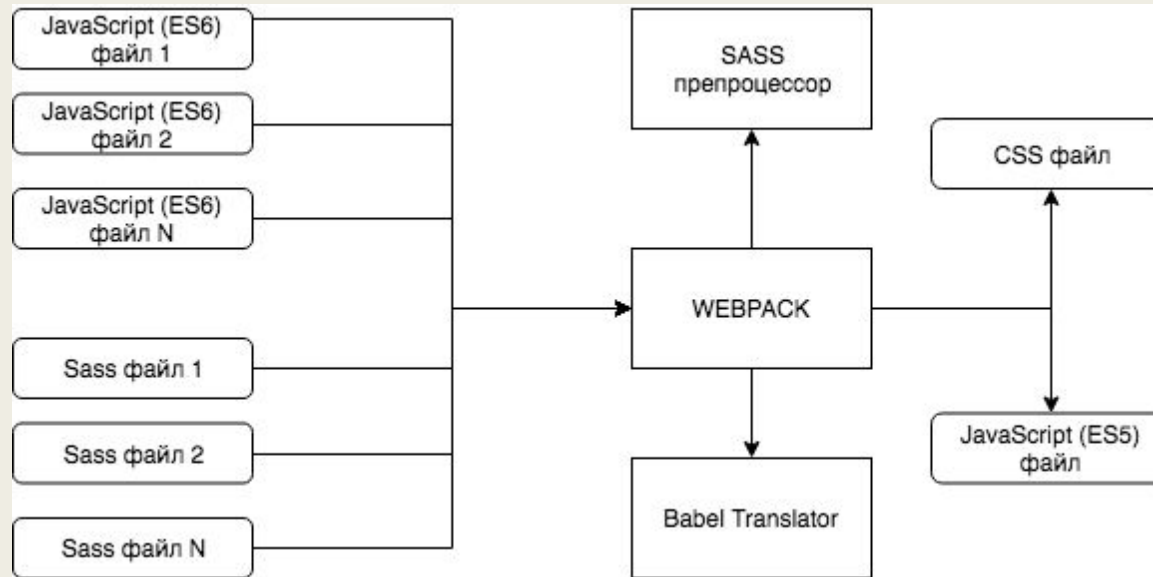
Санкт-Петербург, 2018



Представление дерева на оси натуральных чисел



Реализация клиентской части



Webpack – позволяет совершать сборку JS, CSS файлов, а также управляет зависимостями.

Babel translator – транслятор текста программы на языке JavaScript со стандарта ES6 в ES5

Sass препроцессор – транслятор с языка SCSS в язык стилей CSS.

- Дизайн основных страниц был получен от дизайнера в формате PSD
- Множество JavaScript и CSS файлов в результате преобразования Webpack объединились в один файл
- CSS и JS файлы минимизировались, уменьшая размер файлов передаваемых по сети

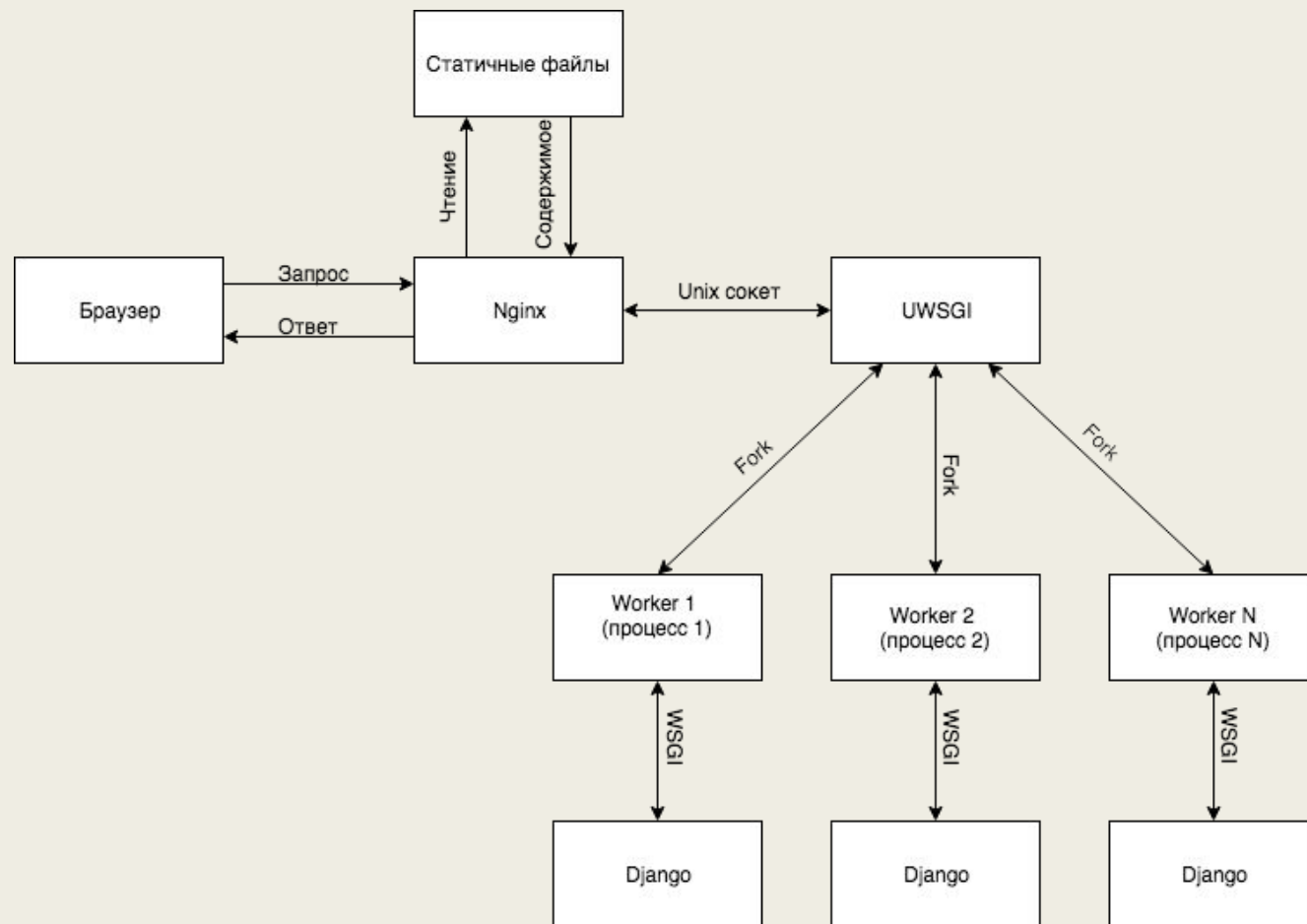
Конфигурация серверного ПО

Nginx – прокси-сервер, построен на Event-Driven (событийно-ориентированной) архитектуре

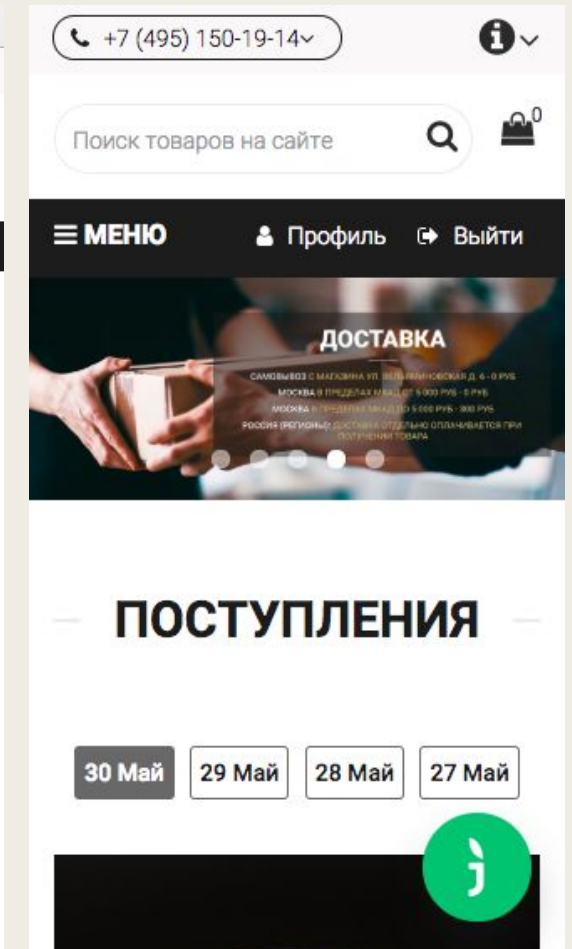
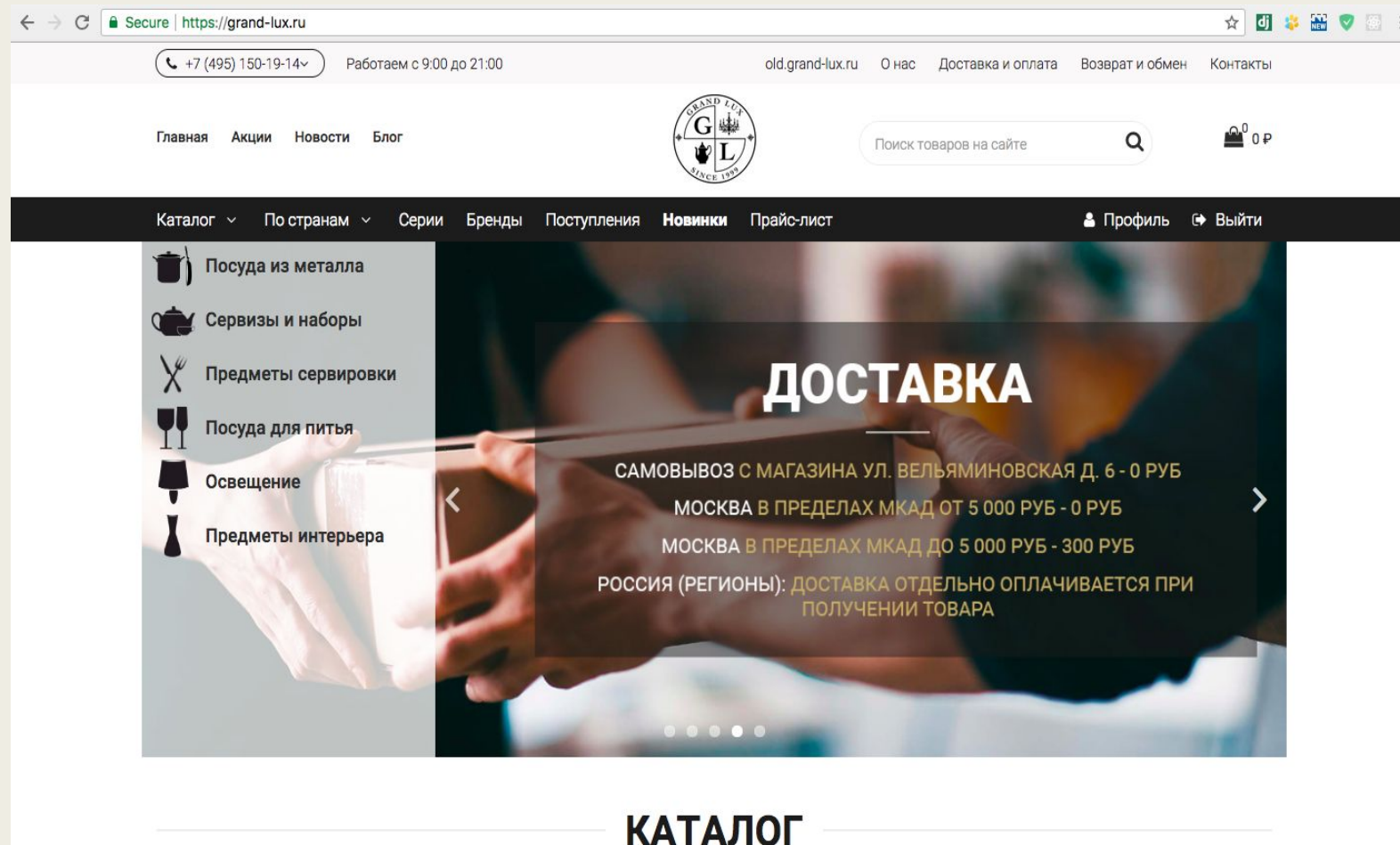
UWSGI – веб-сервер приложений

Схема работы

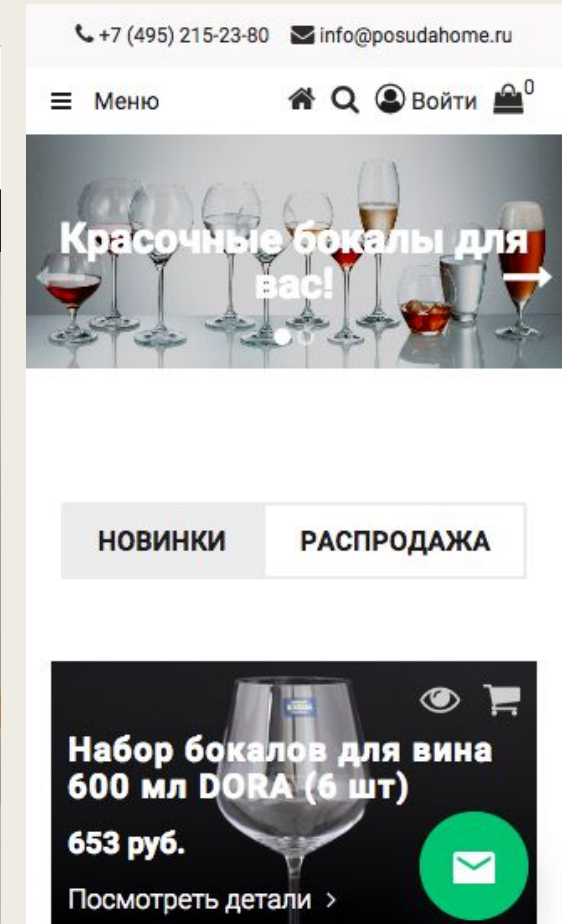
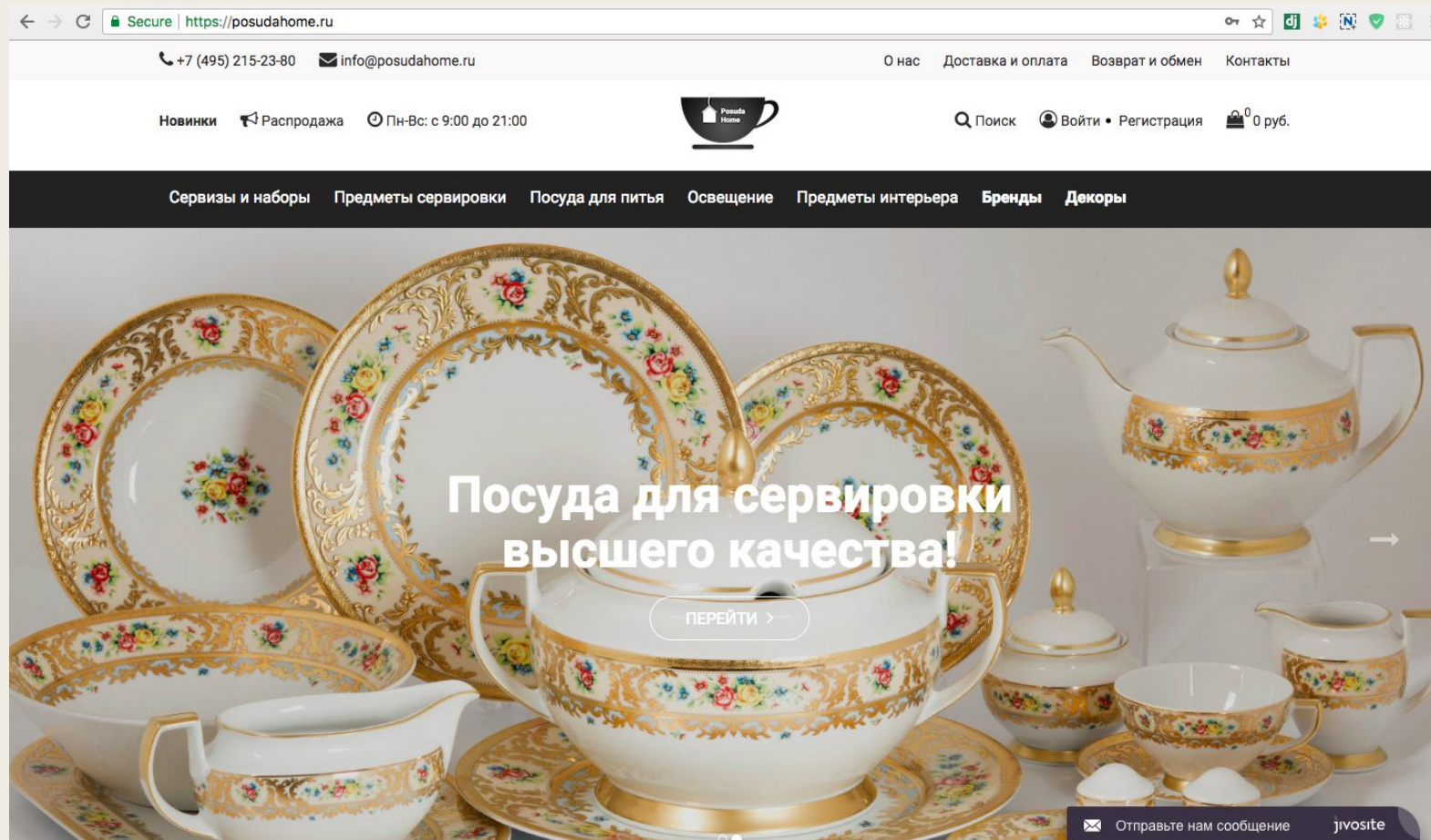
- Статические файлы отдаются Nginx
- Запросы на динамические документы передаются от Nginx к UWSGI
- UWSGI работает с Django по протоколу WSGI
- Django получает запрос от UWSGI и в зависимости от запроса генерирует HTTP ответ
- Ответ передается UWSGI, от него к Nginx и от него пользователю



Сайт Grand-lux.ru



Сайт Posudahome.ru



Заключение

В результате выполнения квалификационной работы была разработана система управления интернет-магазином в соответствии с требованиями заказчика, и на ее основе были разработаны интернет-магазины grand-lux.ru и posudahome.ru.

СПАСИБО ЗА ВНИМАНИЕ!