



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
«Электронные вычислительные машины»

Программа для автоматизации деятельности фирмы по закупке оборудования

Руководитель работы: к.т.н.,
доцент каф. ЭВМ, В.А. Парасич

Автор работы: студент группы
КЭ-406, Д.Р. Аскарлов

Актуальность

- Современные предприятия постоянно нуждаются в обновлении ИТ-оборудования
- Закупки вручную — медленные, подвержены ошибкам и требуют больших затрат
- Автоматизация сокращает время, снижает издержки и повышает прозрачность
- Разработанный комплекс обеспечивает быстрый подбор и оформление заказов
- Тема особенно актуальна в условиях цифровой трансформации бизнеса

Цель и задачи

Цель работы – разработка программы для автоматизации деятельности фирмы по закупке оборудования.

Для достижения данной цели необходимо решить следующие задачи:

1. Провести анализ предметной области
2. Спроектировать программу
3. Реализовать программу
4. Протестировать программу

Обзор аналогов

Параметр	ERP-системы	SRM-системы	ЭТП	Конфигураторы	Моя программа
Автоматический парсинг цен	✗	✗	✗	✓	✓
Подбор ПК по компонентам	✗	✗	✗	✓	✓
Интеграция с поставщиками	✓	✓	✓	✗	✓
Простота внедрения	✗	✗	✓	✓	✓
Гибкость и масштабируемость	✗ / ограничено	✓	✗	✗	✓

Требования к программе

Функциональные требования:

- автоматический сбор данных о товарах с различных торговых площадок (Citilink, DNS, Регард, OnlineTrade, ТоргРС);
- представление результатов в виде ссылок на товары;
- возможность создания конфигурации целого персонального компьютера и выборочных комплектующих;
- возможность создания шаблонов и составление заявок готовых к заказу.

Требования к программе

Нефункциональные требования:

- поддержка расширяемости (возможность добавления новых поставщиков);
- минимальные системные требования к оборудованию и возможность локального развертывания;
- интуитивно понятный интерфейс;
- простота в обслуживании и сопровождении проекта;
- использование свободного программного обеспечения и открытых стандартов.

Выбор технологий

- **Язык программирования:** Python — простота, универсальность, обширная экосистема
- **Веб-фреймворк:** Django — встроенная ORM, админ-панель
- **База данных:** PostgreSQL — надёжность, масштабируемость
- **Сбор данных:** BeautifulSoup, Selenium и Requests — легкость реализации парсинга

Проектирование системы

- **Модульная структура:** Отдельные модули парсинга, обработки данных, формирования заказов и шаблонов
- **База данных:** PostgreSQL с нормализованной структурой — товары, цены, поставщики, конфигурации
- **Модели Django:** использование ORM, защита целостности данных
- **Пользовательский интерфейс:** админ-панель Django и HTML-страницы
- **Архитектура:** Монолит — простота устранения неполадок и внедрения



Диаграмма работы парсеров

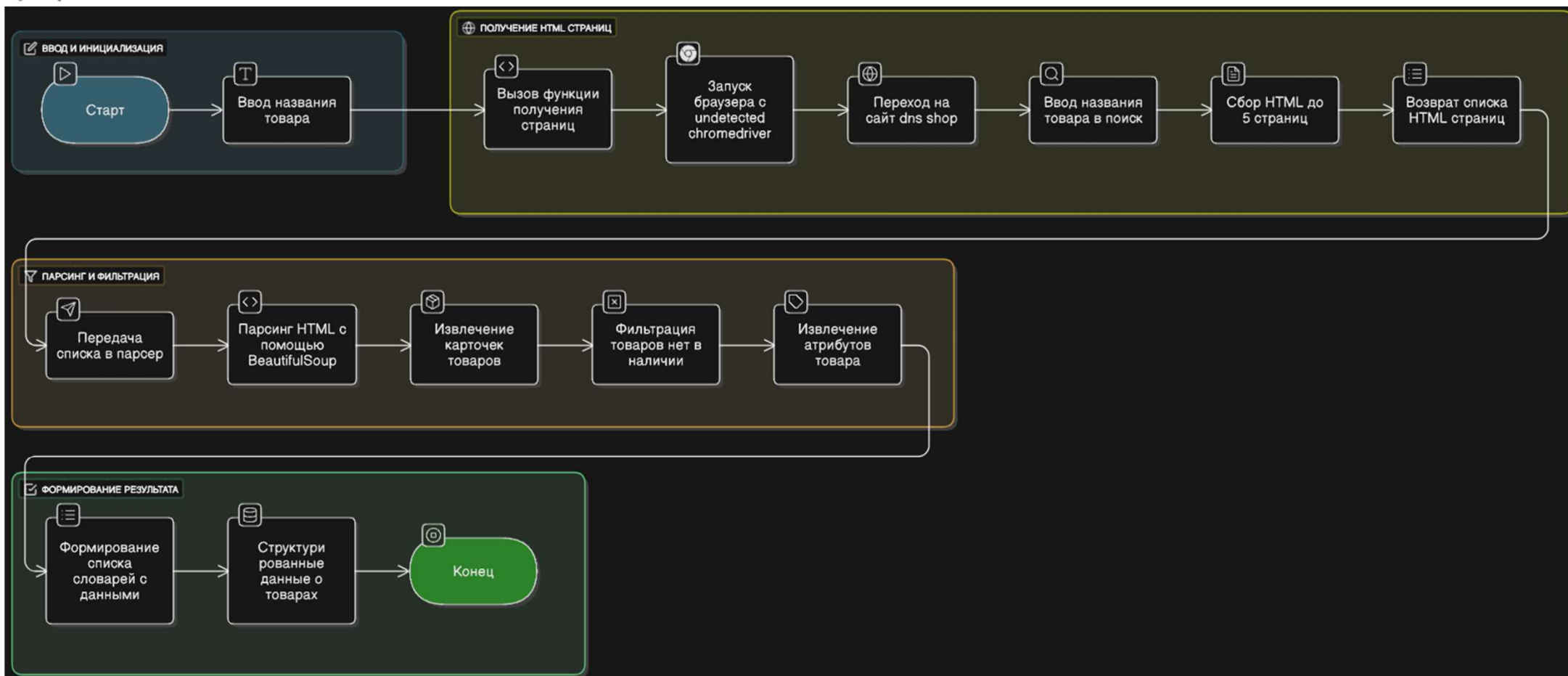
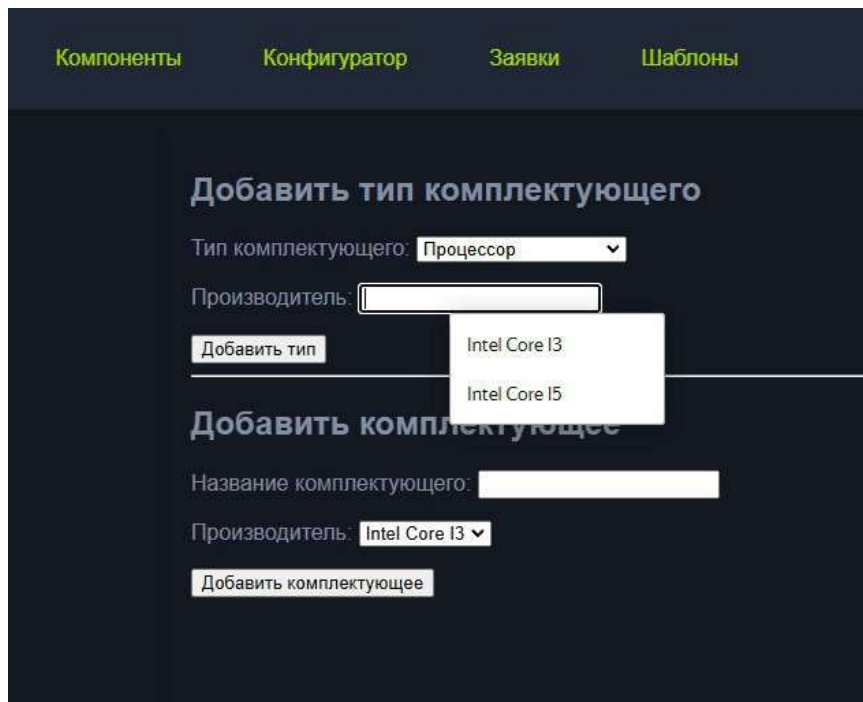


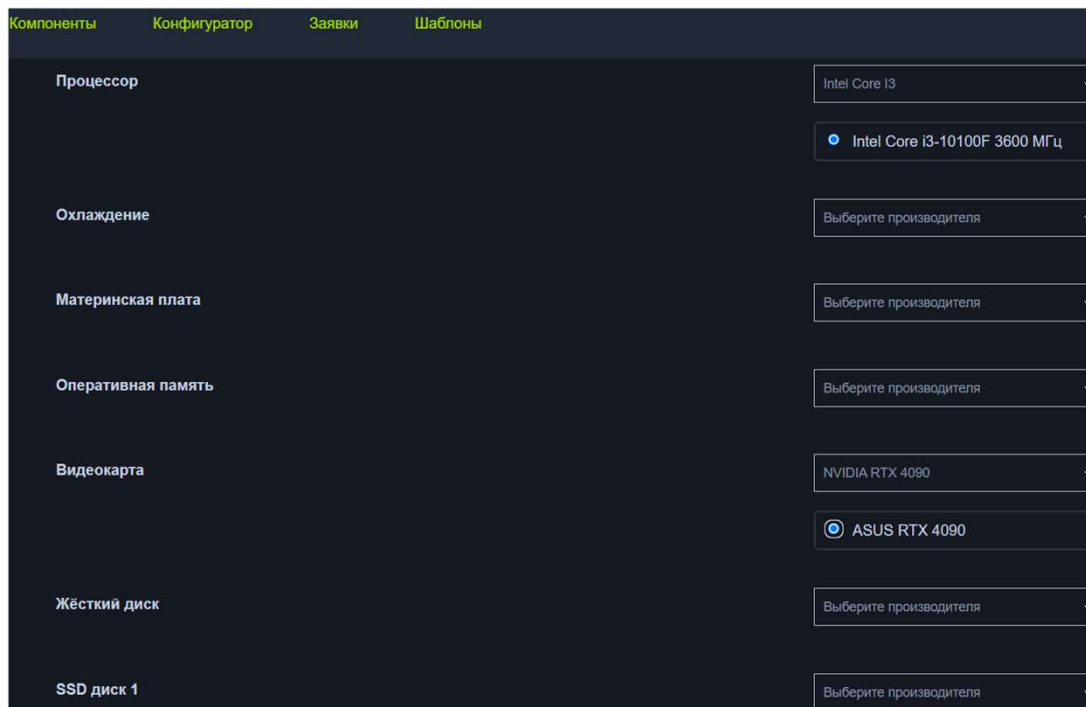
Рисунок 1 –алгоритм работы парсера сайта «dns-shop.ru»

Интерфейс программы



The screenshot shows a web interface with a dark blue header containing four tabs: 'Компоненты', 'Конфигуратор', 'Заявки', and 'Шаблоны'. The 'Конфигуратор' tab is active. The main content area has a title 'Добавить тип комплектующего'. Below it, there is a form with the following fields: 'Тип комплектующего:' with a dropdown menu showing 'Процессор'; 'Производитель:' with an empty text input field; a 'Добавить тип' button; a dropdown menu showing 'Intel Core i3' and 'Intel Core i5'; 'Добавить комплектующее' button; 'Название комплектующего:' with an empty text input field; 'Производитель:' with a dropdown menu showing 'Intel Core i3'; and another 'Добавить комплектующее' button.

Рисунок 2 – добавление комплектующих



The screenshot shows a web interface with a dark blue header containing four tabs: 'Компоненты', 'Конфигуратор', 'Заявки', and 'Шаблоны'. The 'Конфигуратор' tab is active. The main content area displays a list of components with their configuration options: 'Процессор' (Intel Core i3, Intel Core i3-10100F 3600 МГц), 'Охлаждение' (Выберите производителя), 'Материнская плата' (Выберите производителя), 'Оперативная память' (Выберите производителя), 'Видеокарта' (NVIDIA RTX 4090, ASUS RTX 4090), 'Жёсткий диск' (Выберите производителя), and 'SSD диск 1' (Выберите производителя).

Рисунок 3 – конфигуратор с выбором комплектующих

Интерфейс программы

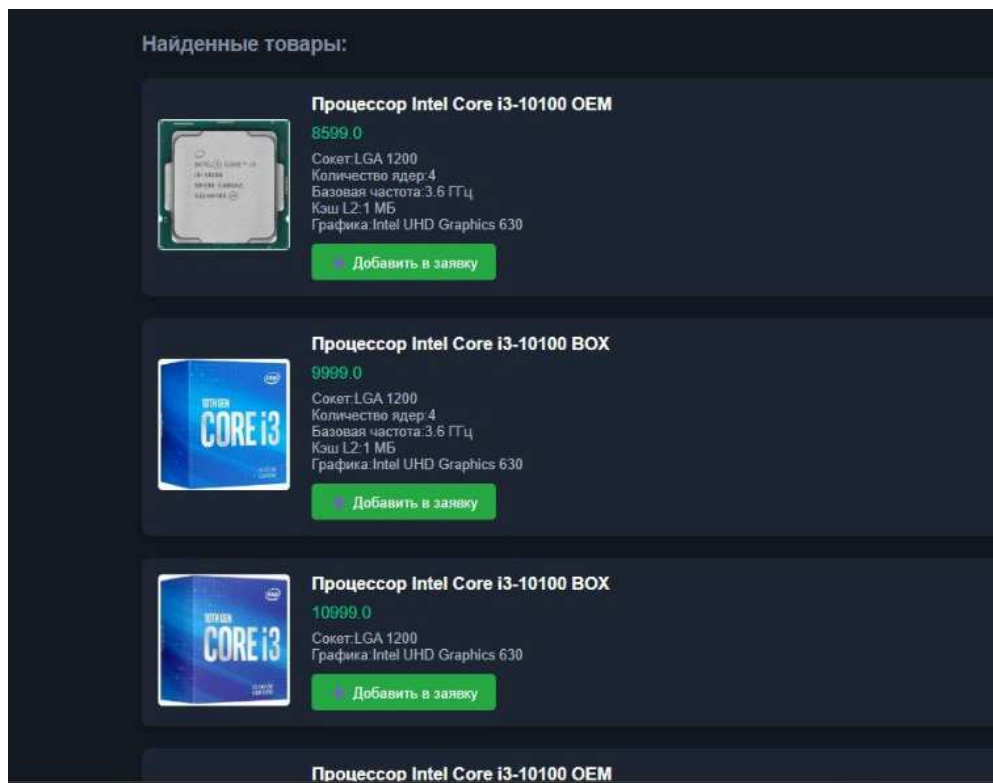


Рисунок 4 – найденные по запросу процессоры с
возможностью добавить их в заявку

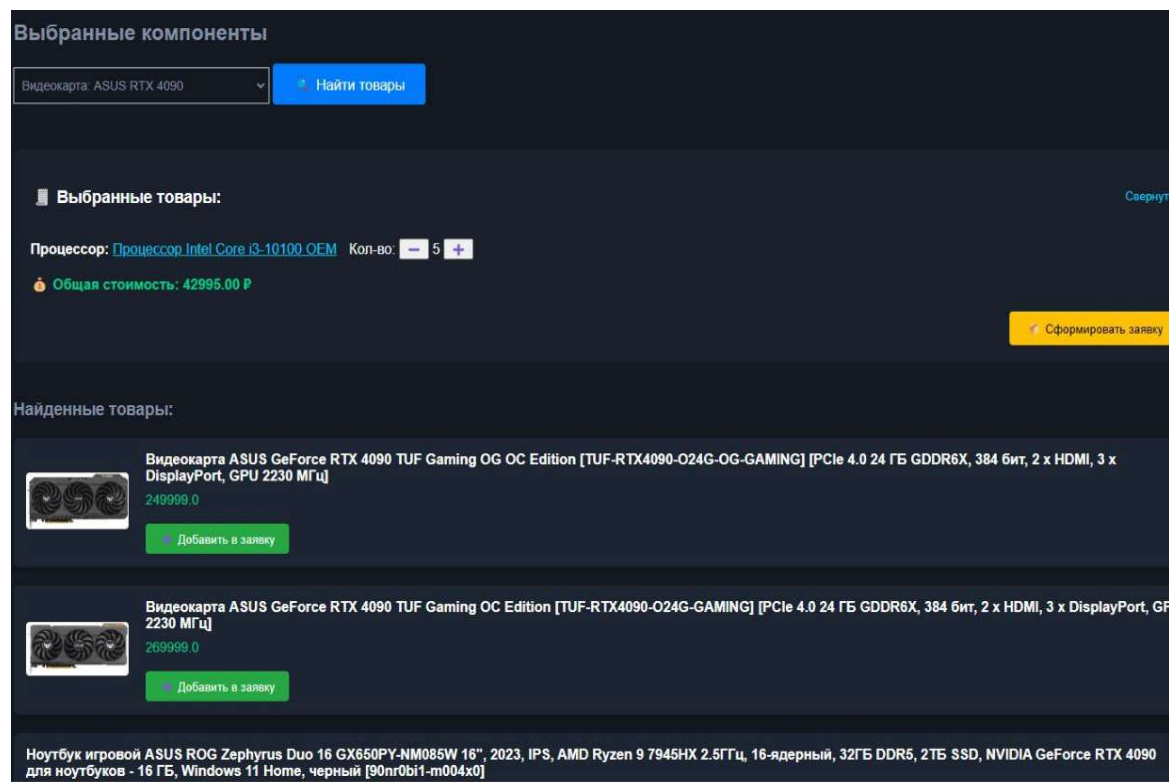


Рисунок 5 – найденные по запросу видеокарты с
возможностью добавить их в заявку

Интерфейс программы

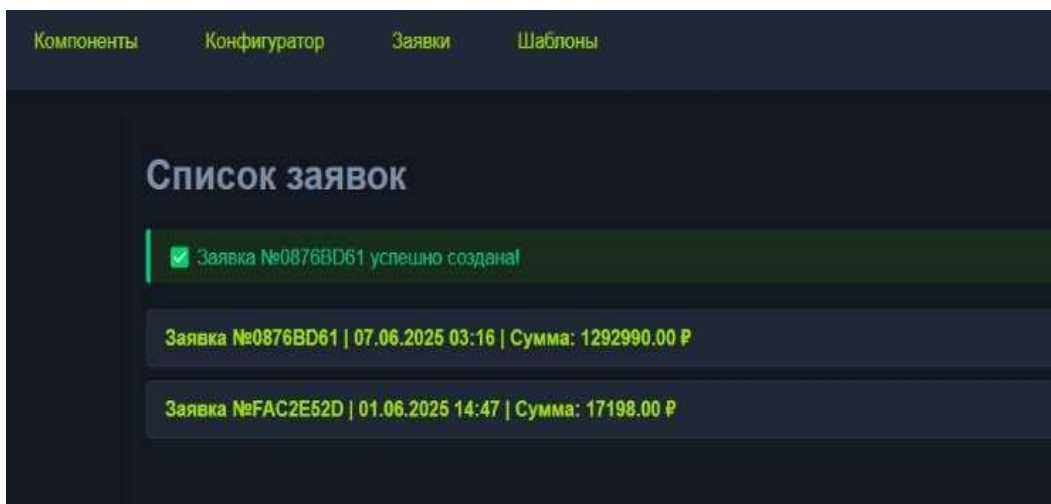


Рисунок 6 – список созданных заявок

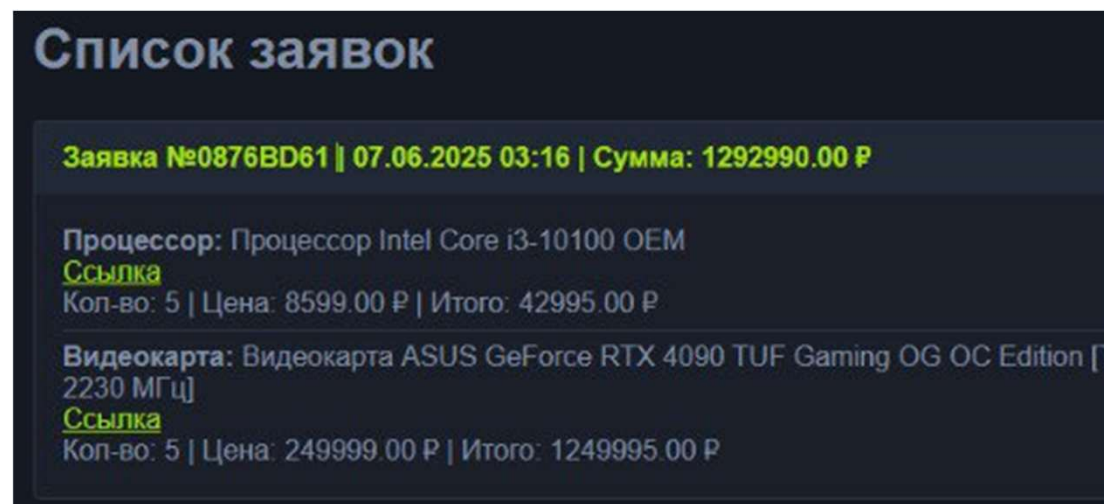


Рисунок 7 – конкретная заявка

Интерфейс программы

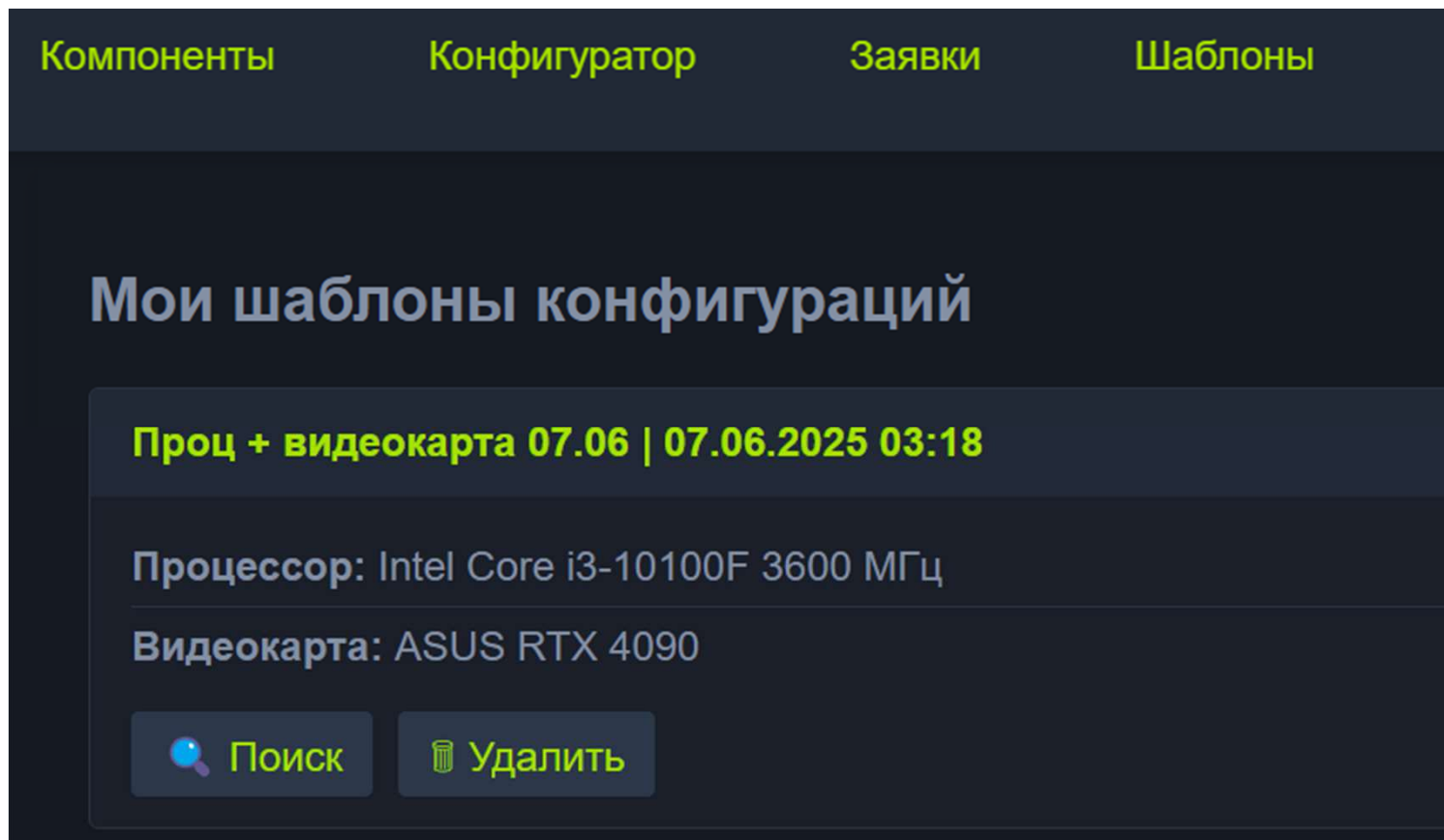


Рисунок 8 – сохраненные шаблоны

Основные результаты

1. Проведён анализ предметной области и существующих решений в сфере автоматизации закупок.
2. Сформулированы функциональные и нефункциональные требования к системе.
3. Разработана архитектура программы на языке программирования Python и основе фреймворка Django, с подключением базы данных PostgreSQL и ряда необходимых библиотек.
4. Реализованы основные модули: парсинг, конфигуратор, оформление заявок, база данных и веб-интерфейс.
5. Проведено тестирование системы — подтверждена работоспособность и соответствие требованиям.
6. Обоснованы направления дальнейшего расширения и интеграции системы.

Выводы

Данная программа позволяет значительно сократить трудозатраты, оперативно реагировать на изменение цен и быстрее формировать заказы. Благодаря модульной структуре и гибкой архитектуре, программа легко масштабируется под новые категории товаров и может легко и недорого интегрироваться в малый и средний бизнес.



Южно-Уральский
государственный
университет

Благодарю за внимание!