|  |
| --- |
|  |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |
| Институт информационных технологий (ИТ) |
| Кафедра ИиППО |

|  |  |
| --- | --- |
| **ОТЧЕТ**  **ПО ПРАКТИЧЕСКОЙ РАБОТЕ №5** | |
| **по дисциплине** |  |
| **«Архитектура клиент-серверных приложений»**  **Тема: Введение в многослойные клиент-серверные архитектуры.** | |
| Выполнил студент группы ИКБО-24-21 | Новиков А.А. |
| Принял преподаватель | Волков М.Ю. |

|  |  |  |
| --- | --- | --- |
| Практические работы выполнены | « » 2023г. |  |
| «Зачтено» | « » 2023 г. |  |

Москва 2023

**Теоретическое введение**

В предыдущих практических работах мы неявно познакомились с одной из архитектур распределенных систем: клиент-серверная архитектура.

‎ Тонкий клиент спроектирован так, чтобы основная часть обработки данных происходила на сервере. Тонкий клиент как правило без жесткого диска: действуют как простой терминал к серверу и требует постоянной связи с сервером. ‎

‎ Толстый клиент выполняет основную часть обработки. У толстых клиентов нет необходимости в непрерывной связи с сервером, поскольку они в основном передают информацию на сервер.

Тонкие клиенты обеспечивают работу рабочего стола в средах, где конечный пользователь имеет четко определенное и регулярное количество задач, для которых используется система. Тонких клиентов можно найти в медицинских офисах, авиабилетах, школах, правительствах, производственных предприятиях и даже колл-центрах. Наряду с простотой установки, тонкие клиенты также предлагают более низкую общую стоимость владения по сравнению с толстыми клиентами.

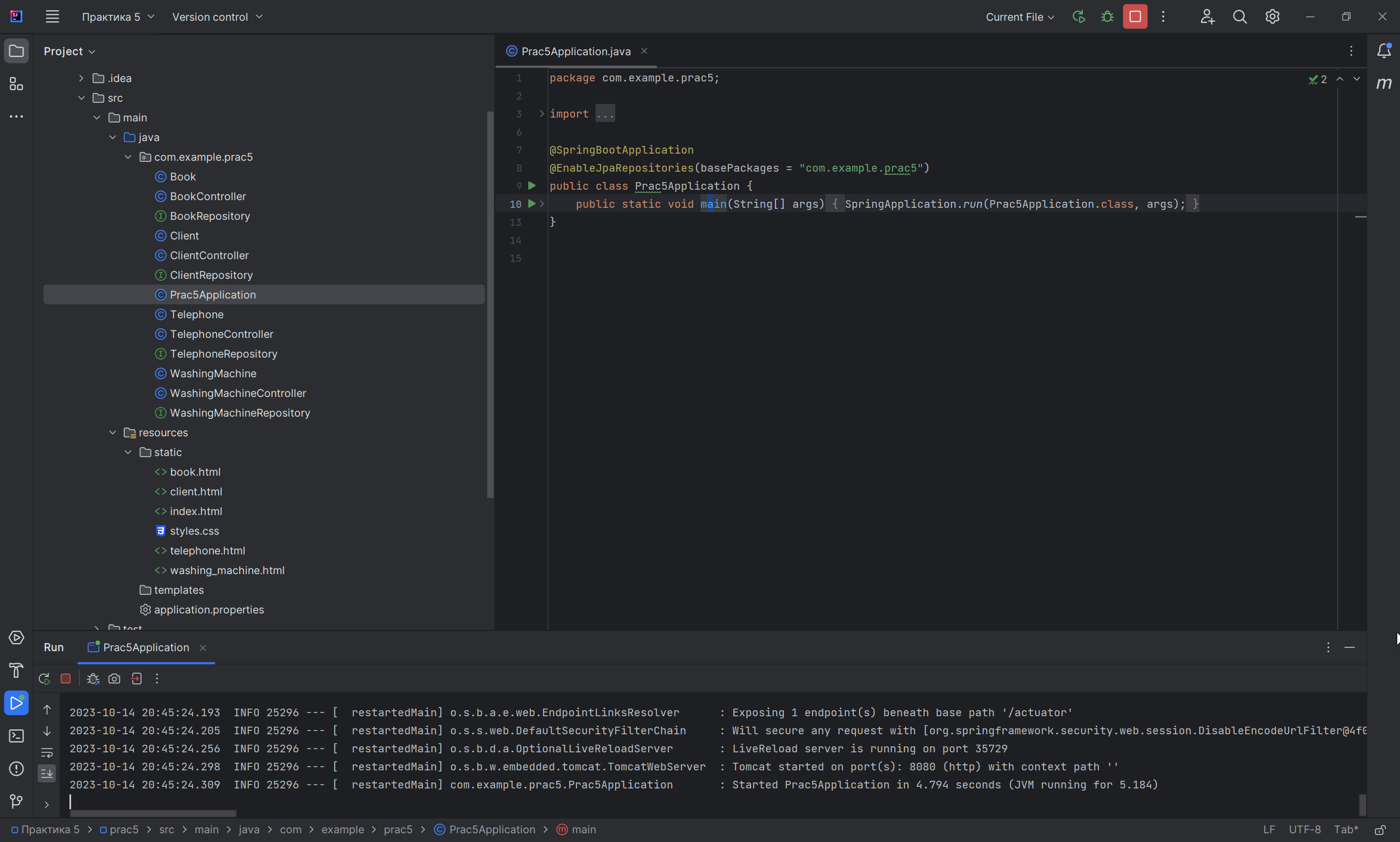
Если вашим приложениям требуются мультимедийные компоненты или которые интенсивно используют пропускную способность, стоит присмотреться к разработке толстых клиентов. Одно из самых больших преимуществ толстых клиентов – некоторые операционные системы и программное обеспечение не могут работать на тонких клиентах. Толстые клиенты могут справиться с ними, поскольку у них есть свои собственные ресурсы.

**Постановка задачи**

Поскольку для трёхуровневой архитектуры необходимо физическое разделение подсистем, то предлагается разработать трёхслойное приложение: БД, сервер, приложение. В качестве приложения можно использовать: запросы в postman/insomnia/testmace, простой сайт, десктопное приложение, мобильное приложение. В качестве БД можно использовать MySQL.

15й вариант: реализовать синхронное взаимодействие через API.

**Программный код**



**Рисунок 1 – Структура проекта**

**Вывод программы**

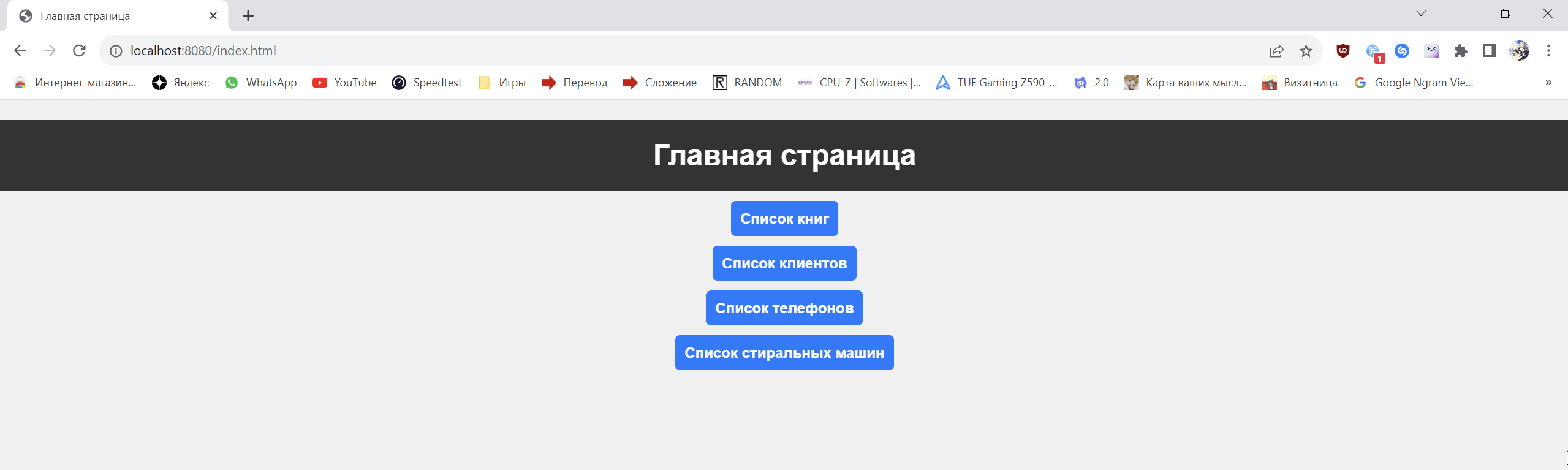


Рисунок 2 – Рабочее приложение

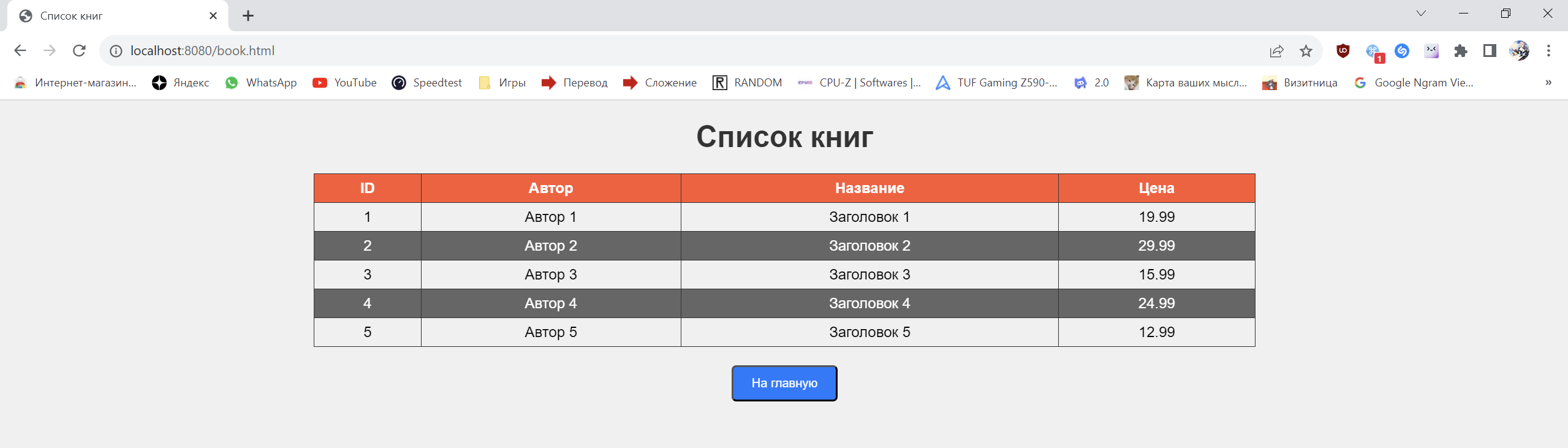


Рисунок 3 – Рабочее приложение

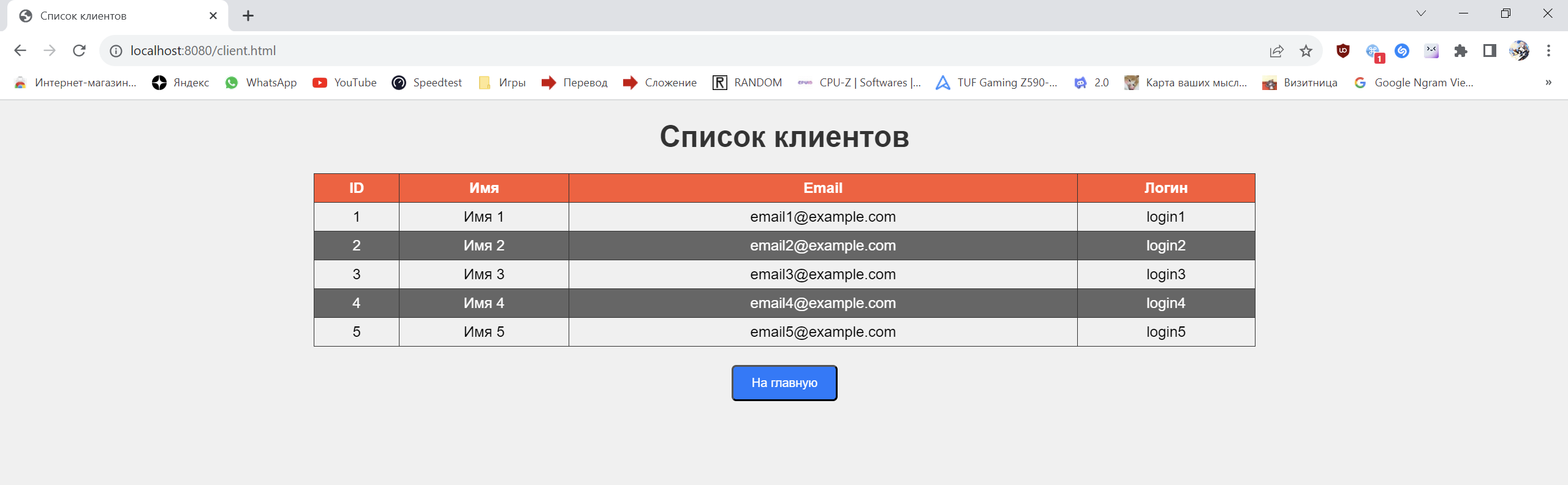


Рисунок 4 – Рабочее приложение

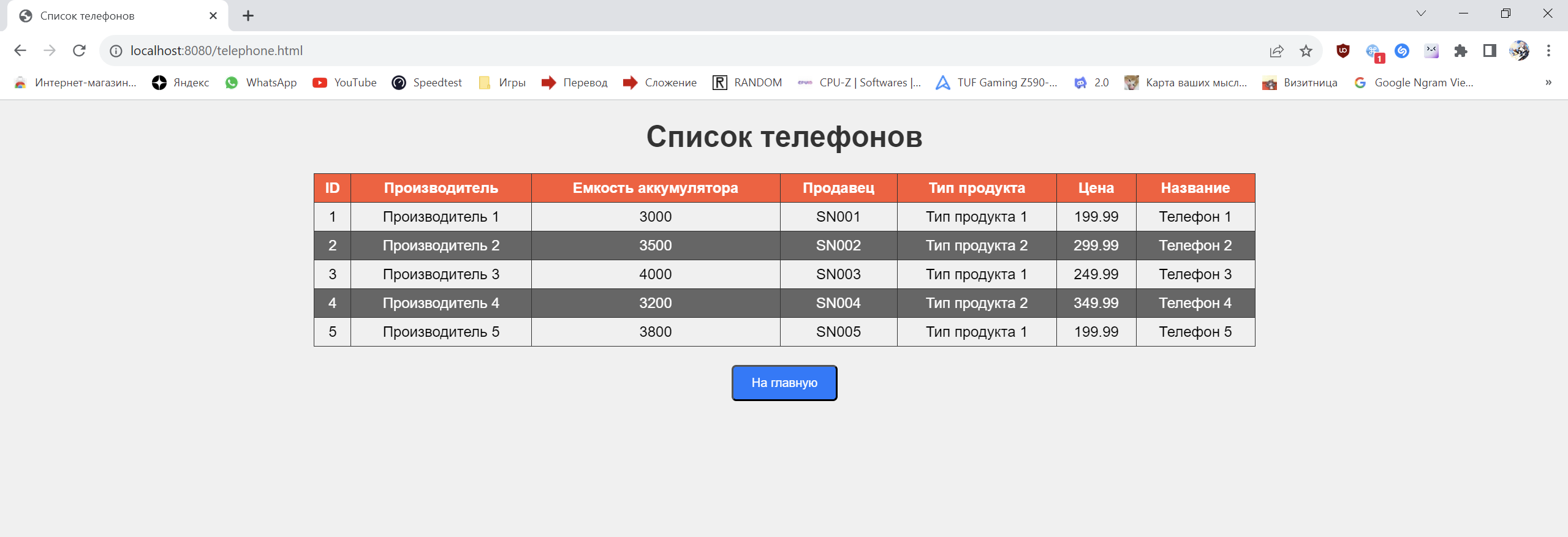


Рисунок 5 – Рабочее приложение

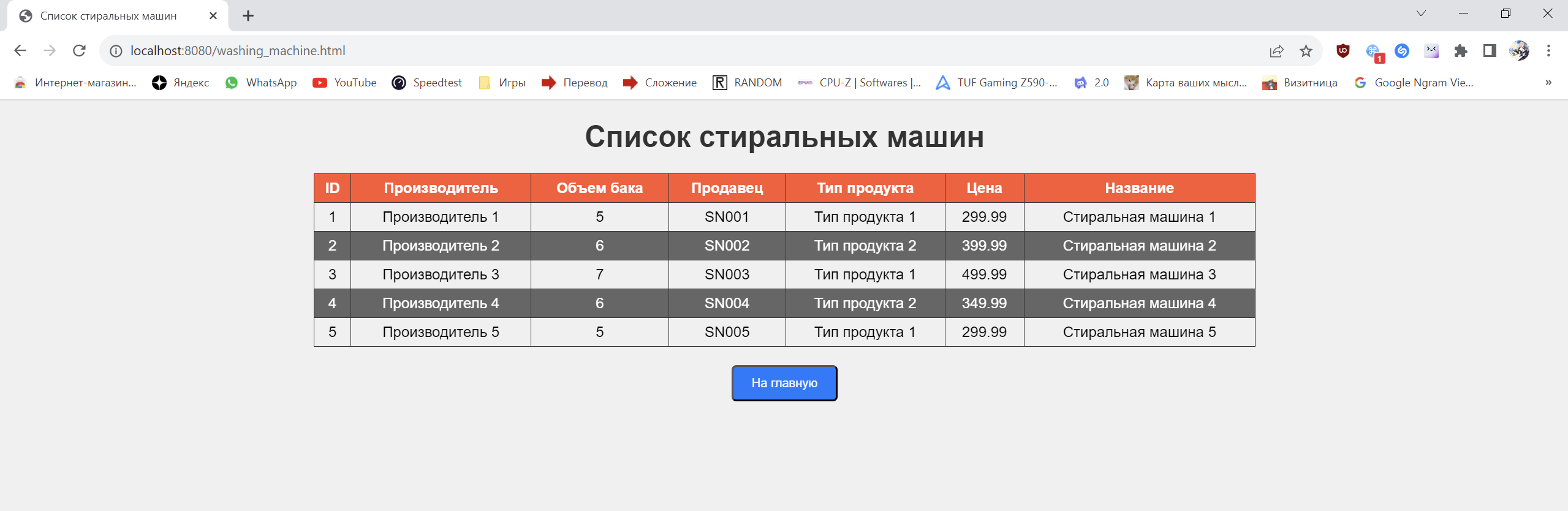


Рисунок 6 – Рабочее приложение

**Вывод**

В результате выполнения этой задачи по созданию трёхслойного приложения на языке JAVA. Было создано приложение, соответствующее этим требованиям.

**Список информационных источников**

1. Конспект лекций по дисциплине «Архитектура клиент-серверных приложений», РТУ МИРЭА (дата обращения 14.10.2023)
2. Документация Java – URL: https://docs.oracle.com/en/java/ (дата обращения 14.10.2023)
3. Курс практик по дисциплине «Архитектура клиент-серверных приложений», РТУ МИРЭА (дата обращения 14.10.2023)