

**Федеральное государственное бюджетное образовательное  
учреждение высшего образования "Белгородский государственный  
технологический университет им. В.Г. Шухова"**

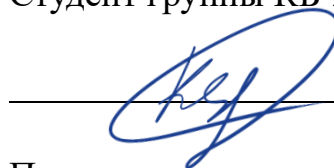
Кафедра программного обеспечения вычислительной техники и  
автоматизированных систем.

**Курсовая работа**

Разработка клиентской части веб-приложения сети магазинов продуктов.

Выполнил:

Студент группы КБ-211



Коренев Д.Н.

Принял:

Панченко М.В.

## Оглавление

Введение .....	3
1. Основная часть .....	4
1.1. Постановка задачи.....	4
1.2. Описание решения задачи .....	4
1.3. Ожидаемый результат.....	4
1.4. Техническая информация.....	4
1.5. Описание и демонстрация интерфейса .....	5
1.6. NGINX .....	7
1.7. Тестирование .....	8
Заключение .....	11
Список использованной литературы.....	12
Приложения .....	13

## **Введение**

В современном мире сети магазинов сталкиваются с растущей конкуренцией и необходимостью присутствия в онлайн-среде для удовлетворения потребностей клиентов. Создание веб-сайта для сети магазинов становится важным шагом для расширения клиентской базы и увеличения объема продаж. В данной курсовой работе будет рассмотрена тема разработки веб-сайта для сети магазинов с использованием базы данных.

Целью данной работы является проектирование веб-сайта, обеспечивающей эффективное пользование серверной части приложения, удовлетворяя потребности как клиентов, так и владельцев магазинов. Для достижения этой цели необходимо провести анализ требований, разработать функционал веб-сайта, интегрировать базу данных с веб-сайтом.

С развитием интернет-технологий и изменением потребительского поведения важно, чтобы веб-сайт сети магазинов предоставлял удобный и привлекательный интерфейс для пользователей, а также обеспечивал эффективное управление товарами и другими аспектами бизнеса. Поэтому данное исследование имеет практическую значимость для бизнеса и академического сообщества в области информационных технологий.

Результаты данной работы могут быть использованы в качестве руководства при создании веб-сайтов для сетей магазинов с целью повышения их эффективности и конкурентоспособности. Таким образом, данное исследование представляет актуальную тему для изучения и может привлечь внимание специалистов в области информационных технологий и бизнеса.

## **1. Основная часть**

### **1.1. Постановка задачи**

Целью данной курсовой работы является разработка и реализация веб-сайта, который обеспечивает оптимальное использование серверной части приложения для удовлетворения потребностей различных заинтересованных сторон, включая клиентов и владельцев магазинов. Для успешного выполнения этой задачи необходимо выполнить следующие этапы: анализ требований к веб-сайту, проектирование функционала веб-сайта, интеграция веб-сайта с базой данных.

### **1.2. Описание решения задачи**

Целью данного проекта является создание веб-сайта, который позволяет просматривать и редактировать данные из таблиц базы данных, связанных с магазинами, товарами, работниками, группами товаров и чеками. Для обеспечения удобства и эффективности работы пользователей необходимо разработать интуитивно понятный и функциональный интерфейс веб-сайта. Для реализации данного проекта необходимо изучить различные технологии и методы разработки веб-сайтов, провести сравнительный анализ и выбрать наиболее оптимальное решение с учетом требований и ограничений.

### **1.3. Ожидаемый результат**

В качестве итогового результата данного проекта, предполагается получение следующих продуктов:

- 1) Веб-сайт, соответствующий требованиям и спецификациям, разработанный с использованием современных технологий и методов.
- 2) Программный код веб-сайта, написанный с использованием выбранного фреймворка, документированный и отформатированный в соответствии с принятыми стандартами.
- 3) Проведение тестов веб-сайта, включая функциональные, нагрузочные, безопасностные и юзабилити тесты, а также анализ и устранение обнаруженных ошибок и недостатков.

### **1.4. Техническая информация**

В рамках данного проекта я принял решение использовать язык программирования JavaScript для разработки веб-сайта. JavaScript обладает большим количеством фреймворков, которые значительно упрощают и ускоряют процесс создания веб-сайтов, таких как React, Angular, Vue.js,

Ember.js, Backbone.js и jQuery. Из них я выбрал React, поскольку он является самым популярным и востребованным фреймворком на сегодняшний день, а также отличается своей простотой и гибкостью. Для обработки входящих и исходящих соединений, а также запросов от пользователей, я решил использовать NGINX. NGINX позволяет эффективно распределять нагрузку на серверную часть приложения и обеспечивать высокую производительность и надежность при работе с большим количеством запросов одновременно. Для обмена данными между веб-сайтом и Django REST API я буду использовать библиотеку axios. Axios является одной из самых популярных и удобных библиотек для работы с HTTP-запросами в JavaScript. С помощью axios я буду получать данные для таблиц из REST API, а также отправлять туда результаты изменений, внесенных пользователем.

## 1.5. Описание и демонстрация интерфейса

На главной странице веб-сайта пользователь может выбрать одну из таблиц базы данных для загрузки и просмотра (см. рисунок 1). На каждой странице веб-сайта также имеется нижняя панель, которая содержит различную информацию, а также ссылки на другие разделы веб-сайта.

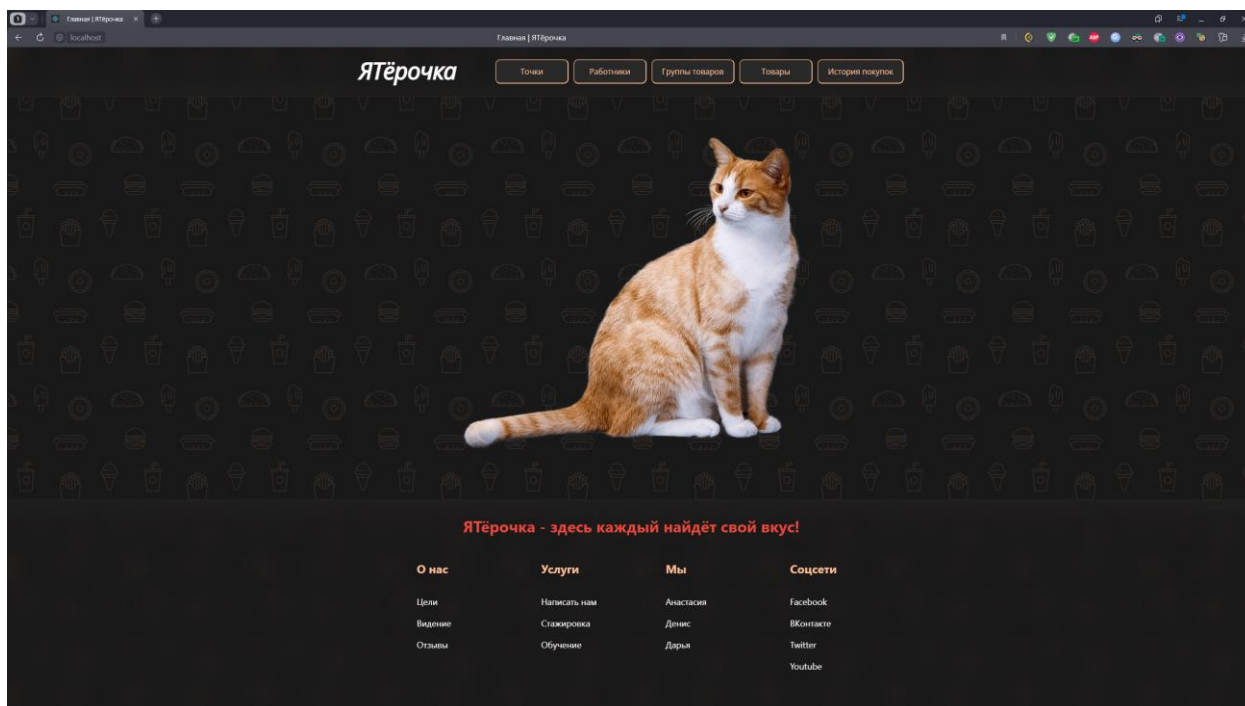


Рисунок 1. Главная страница сайта

После того, как пользователь выберет одну из таблиц базы данных на главной странице, ему будет представлена соответствующая таблица (см. рисунок 2). На этой странице также расположены кнопки, которые позволяют пользователю войти или выйти из своего аккаунта, а также загрузить данные из таблицы в форматы JSON или CSV.

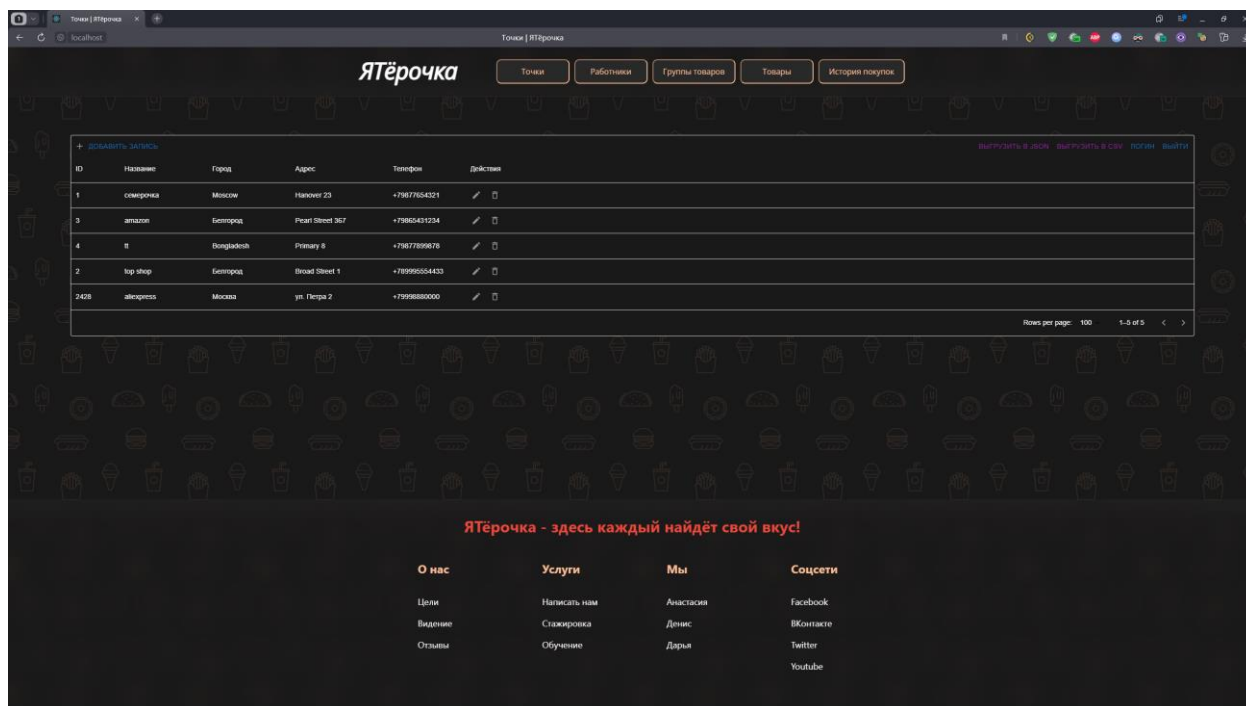


Рисунок 2. Таблица «Точки»

На этих страницах пользователь может редактировать (см. рисунки 3-4), удалять и добавлять строки в таблицах базы данных (при условии, что пользователь имеет соответствующие права доступа). Кроме того, пользователь может применять различные фильтры и сортировки к данным внутри таблиц, чтобы найти нужную информацию или упорядочить данные по определенному критерию.

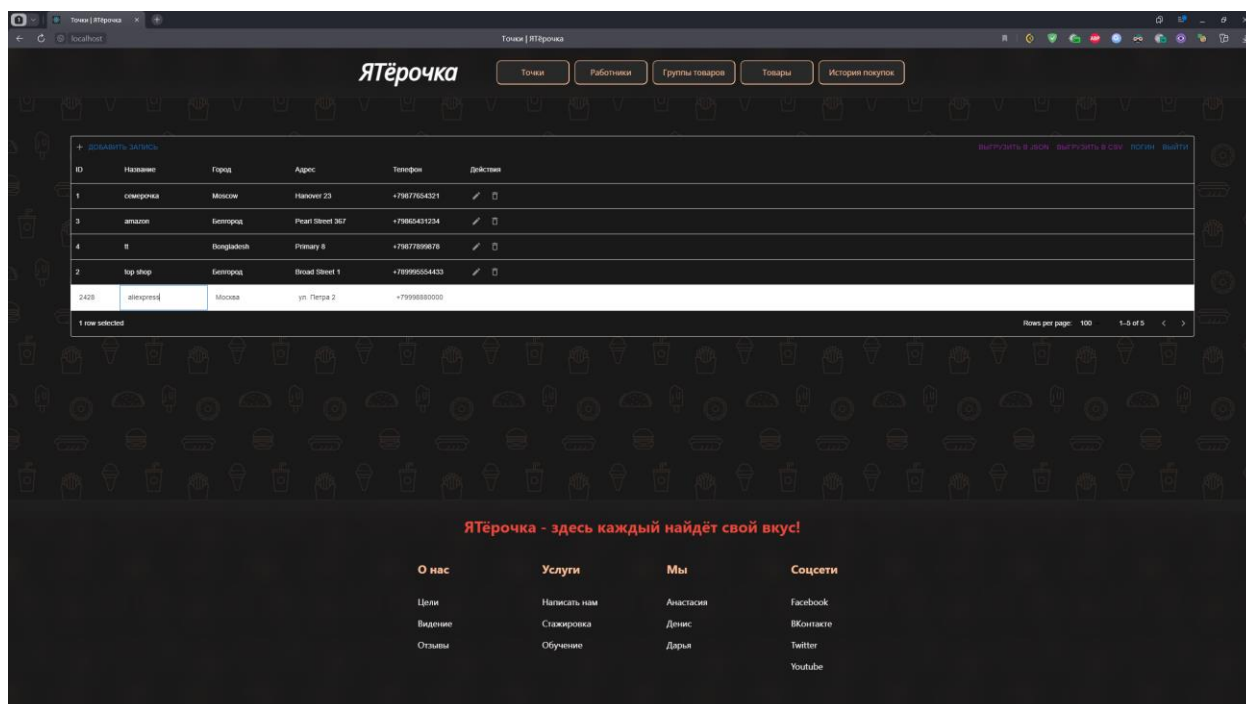


Рисунок 3. Изменение таблицы

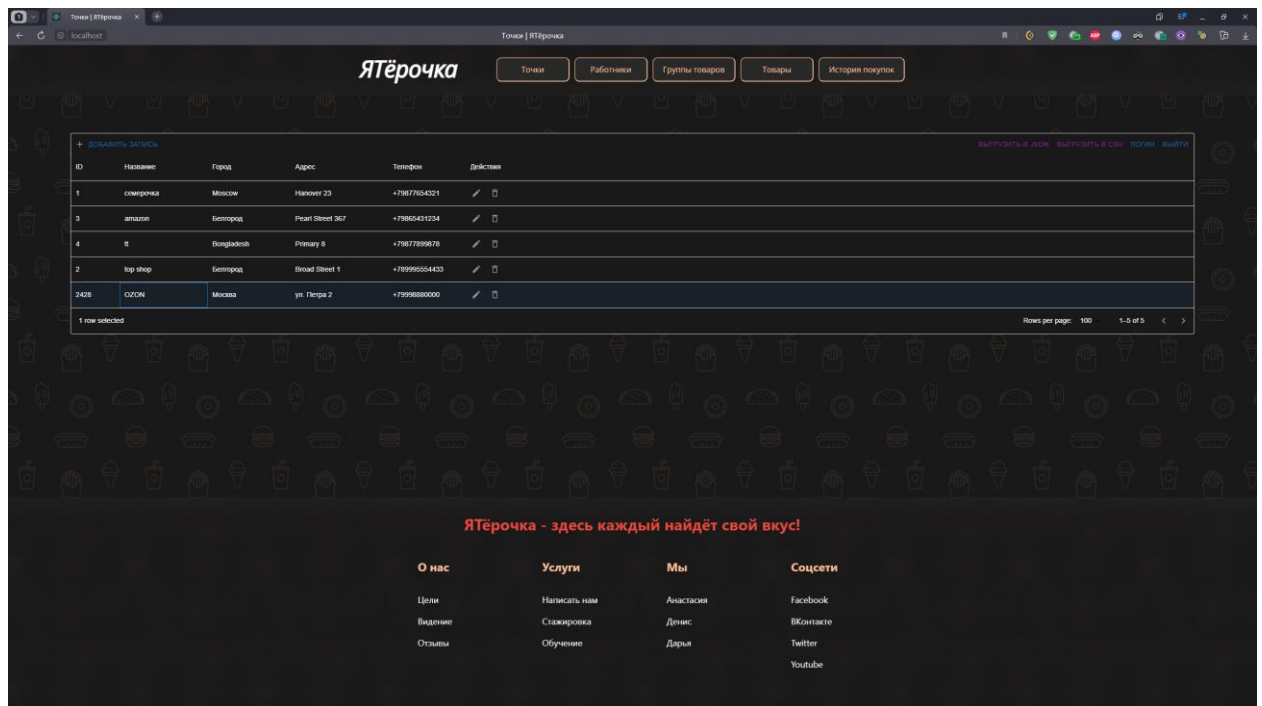


Рисунок 4. Измененная таблица

## 1.6. NGINX

В рамках данного проекта мы приняли решение использовать NGINX для управления входящими и исходящими соединениями, а также для обработки запросов от пользователей к веб-сайту и API. NGINX позволяет настроить перенаправление трафика с одного порта на другой, что упрощает работу с приложением. Для того, чтобы реализовать такое перенаправление с порта 80, который является стандартным портом для HTTP-запросов, на порты 3000 и 8000, на которых запущены веб-сайт и API соответственно, мы написали следующий файл конфигурации для NGINX.

```
upstream client {
    server yaterochka-web;
}

server {
    listen 80;

    location / {
        proxy_pass http://yaterochka-web:3000/;
    }
    location /api/product/ {
        proxy_pass http://yaterochka-back:8000/product/?format=json;
    }
    location /api/store/ {
        proxy_pass http://yaterochka-back:8000/store/?format=json;
    }
    location /api/employee/ {
        proxy_pass http://yaterochka-back:8000/employee/?format=json;
    }
    location /api/group/ {
        proxy_pass http://yaterochka-back:8000/group/?format=json;
    }
    location /api/check/ {
        proxy_pass http://yaterochka-back:8000/check/?format=json;
    }
}
```

```
}  
}
```

Для запуска и работы NGINX мы будем использовать официальный Docker-контейнер NGINX, который предоставляет все необходимые настройки и зависимости. Поэтому нам не потребуется выполнять дополнительных действий для конфигурации NGINX, кроме того, что мы уже сделали.

## 1.7. Тестирование

На страницах с таблицами базы данных пользователь имеет возможность изменять значения любых полей (за исключением поля ID), которые отображаются в рисунках 3-4. Также пользователь может добавлять новые строки в таблицы, используя формы, представленные в рисунках 5-6, или удалять существующие строки, нажимая на кнопку “Удалить”, как показано в рисунке 7.

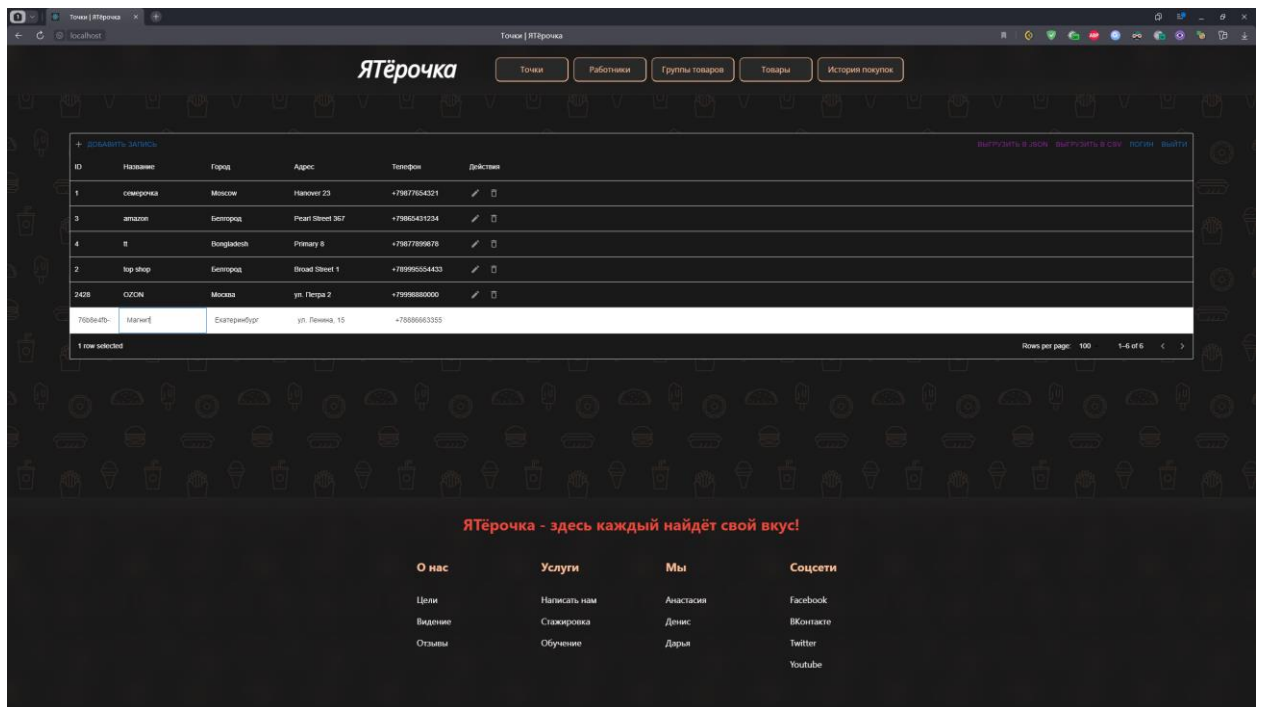


Рисунок 5. Добавление новой строки в таблицу



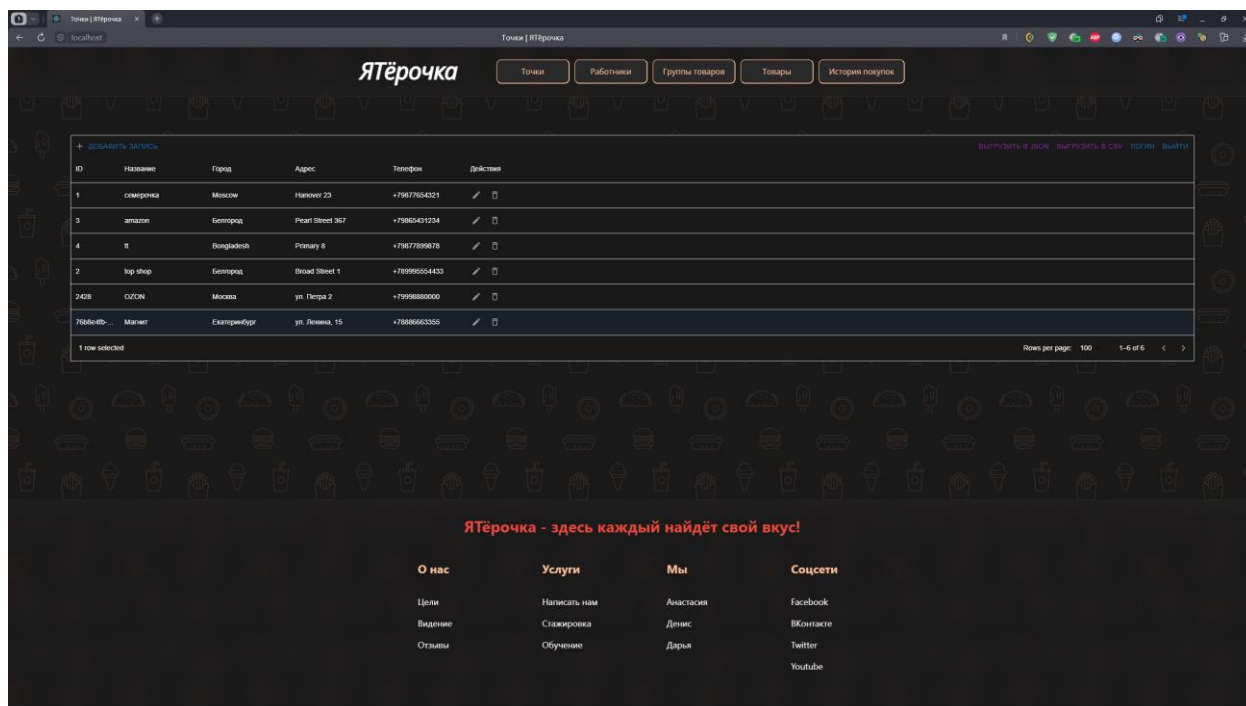


Рисунок 6. Измененная таблица

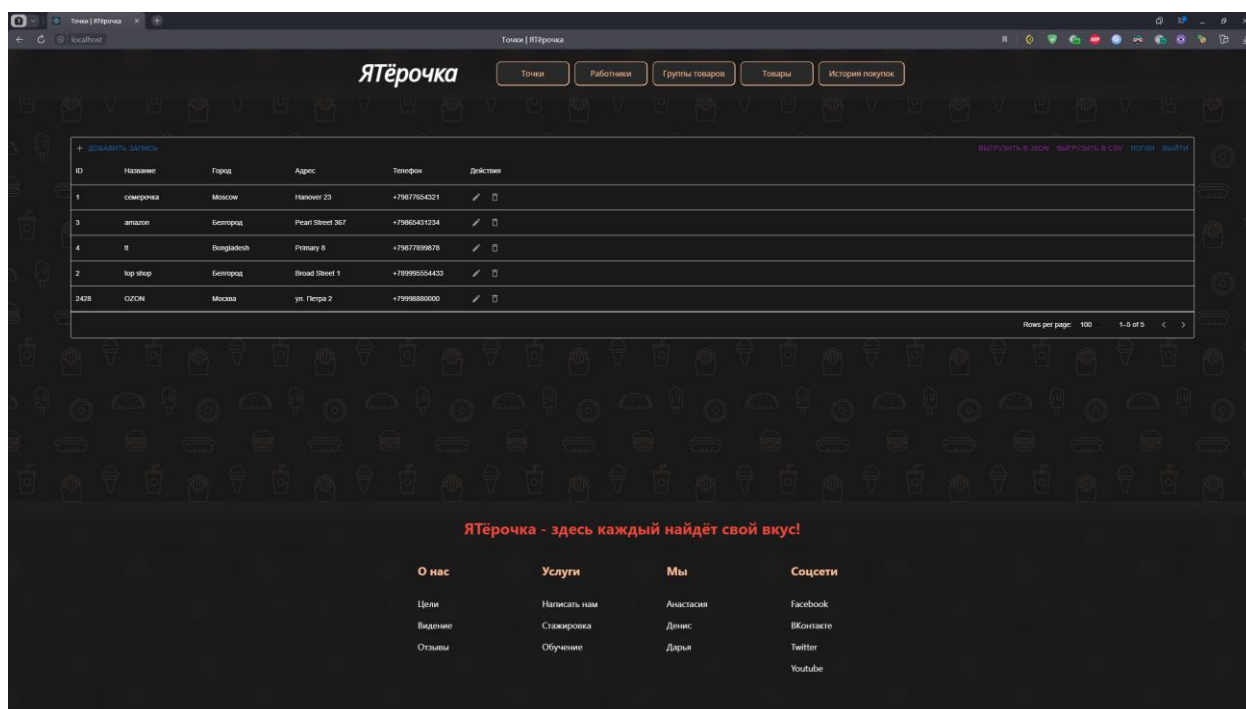


Рисунок 7. Измененная таблица после удаления строки

В рамках данного проекта также была реализована функция выгрузки данных из таблиц базы данных в форматы JSON и CSV. Эта функция позволяет пользователю сохранить данные в удобном для дальнейшей обработки или анализа формате. Для того, чтобы воспользоваться этой функцией, пользователь должен нажать на соответствующую кнопку на странице с таблицей. После этого наш сайт сгенерирует и вернет файл с данными в выбранном формате для текущей таблицы. Примеры таких файлов для таблицы “Магазины” приведены в листингах 1-2.



## Заключение

В ходе выполнения данной курсовой работы была решена задача разработки веб-сайта для сети магазинов с использованием базы данных. Для этого были выполнены следующие этапы:

- Анализ требований к веб-сайту, включая определение целевой аудитории, функциональных и нефункциональных требований, сценариев использования и диаграммы вариантов использования.
- Проектирование веб-сайта, включая разработку структуры, навигации, дизайна и контента веб-сайта.
- Реализация веб-сайта, включая выбор технологий для разработки веб-сайта и базы данных, написание кода для реализации функционала веб-сайта, интеграция базы данных с веб-сайтом, тестирование и отладка веб-сайта.

В результате разработки веб-сайта были достигнуты следующие результаты:

- Веб-сайт сети магазинов предоставляет удобный и привлекательный интерфейс для пользователей.
- Веб-сайт сети магазинов обеспечивает эффективное управление товарами и другими аспектами бизнеса, позволяя владельцам магазинов добавлять, редактировать и удалять товары, просматривать статистику продаж, а также настраивать параметры веб-сайта.
- Веб-сайт сети магазинов использует базу данных для хранения и обработки информации о товарах, магазинах, покупках, клиентах и других сущностях, а также обеспечивает безопасность и надежность данных.
- Веб-сайт сети магазинов соответствует функциональным и нефункциональным требованиям, а также удовлетворяет потребности как клиентов, так и владельцев магазинов.

Таким образом, цель данной работы была достигнута. Однако, в процессе разработки и оценки веб-сайта были выявлены некоторые недостатки и проблемы, которые могут быть устранены в будущем. Например, веб-сайт может быть улучшен с точки зрения дизайна, интерактивности, адаптивности, доступности, оптимизации и безопасности. Кроме того, веб-сайт может быть расширен с помощью добавления новых функций. В заключение, можно сказать, что разработка веб-сайта для сети магазинов с использованием базы данных является актуальной и перспективной темой для исследования и практики, которая может способствовать развитию бизнеса и удовлетворению потребностей клиентов.

## Список использованной литературы

1. React URL: <https://react.dev/> (дата обращения 12.12.2023).
2. Учебное пособие: Введение в React URL: <https://ru.react.js.org/tutorial/tutorial.html> (дата обращения 12.12.2023).
3. React.js: понятное руководство для начинающих // Хабр URL: <https://habr.com/ru/companies/ruvds/articles/428077/> (дата обращения 15.12.2023).
4. nginx URL: <https://nginx.org/ru/> (дата обращения 16.12.2023).
5. Архитектура open source-приложений: Как работает nginx // Хабр URL: <https://habr.com/ru/companies/latera/articles/273283/> (дата обращения 17.12.2023).
6. Django URL: <https://www.djangoproject.com/> (дата обращения 19.12.2023).
7. Что же такое Django? // Хабр URL: <https://habr.com/ru/articles/747234/> (дата обращения 20.12.2023).
8. Введение в REST API — RESTful веб-сервисы // Хабр URL: <https://habr.com/ru/articles/483202/> (дата обращения 20.12.2023).
9. Django Rest Framework для начинающих: создаём API для чтения данных // Хабр URL: [https://habr.com/ru/companies/yandex\\_praktikum/articles/561696/](https://habr.com/ru/companies/yandex_praktikum/articles/561696/) (дата обращения 20.12.2023).
10. Django REST framework URL: <https://www.django-rest-framework.org/> (дата обращения 20.12.2023).

## Приложения

### Приложение 1. Файл «package.json»

```
{
  "name": "yaterochka-web",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "@emotion/react": "^11.11.1",
    "@emotion/styled": "^11.11.0",
    "@mui/icons-material": "^5.14.19",
    "@mui/material": "^5.14.20",
    "@mui/x-data-grid": "^6.18.4",
    "@mui/x-data-grid-generator": "^6.18.4",
    "@tanstack/react-query": "^5.13.4",
    "@testing-library/jest-dom": "^5.17.0",
    "@testing-library/react": "^13.4.0",
    "@testing-library/user-event": "^13.5.0",
    "angular": "^1.8.3",
    "axios": "^1.6.2",
    "material-react-table": "^2.0.5",
    "react": "^18.2.0",
    "react-bootstrap": "^2.9.1",
    "react-dom": "^18.2.0",
    "react-router-dom": "^6.20.1",
    "react-scripts": "^3.0.1",
    "styled-components": "^6.1.1",
    "web-vitals": "^2.1.4"
  },
  "scripts": {
    "start": "react-scripts --openssl-legacy-provider start",
    "build": "react-scripts --openssl-legacy-provider build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },
  "eslintConfig": {
    "extends": [
      "react-app",
      "react-app/jest"
    ]
  },
  "browserslist": {
    "production": [
      ">0.2%",
      "not dead",
      "not op_mini all"
    ],
    "development": [
      "last 1 chrome version",
      "last 1 firefox version",
      "last 1 safari version"
    ]
  }
}
```

### Приложение 2. Файл «App.js»

```
import './App.css';
import './Header.css';

import { Routes, Route } from 'react-router-dom'

import React from 'react';
```

```

import Home from "./Home";
import Places from "./Places";
import Workers from "./Workers";
import Groups from "./Groups";
import Goods from "./Goods";
import History from "./History";

function App() {
  return (
    <div>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/places" element={<Places />} />
        <Route path="/workers" element={<Workers />} />
        <Route path="/groups" element={<Groups />} />
        <Route path="/goods" element={<Goods />} />
        <Route path="/history" element={<History />} />
      </Routes>
    </div>
  );
}

export default App;

```

### Приложение 3. Файл «App.css»

```

.App {
  text-align: center;
}

.content {
  min-height: 49vh;
  padding-top: 5rem;
}

```

### Приложение 4. Файл «Footer.jsx»

```

import React from "react";
import styled from "styled-components";
import './Header.css';

export const Box = styled.div`
  padding: 0;
`;

export const GlassBox = styled.div`
  inset: 0;
  z-index: -1;
  /* Extend the backdrop to the bottom for it to "collect the light"
  outside of the nav */
  --extended-by: -100px;
  bottom: calc(-1 * var(--extended-by));

  --filter: blur(20px);
  -webkit-backdrop-filter: var(--filter);
  backdrop-filter: var(--filter);
  // pointer-events: none;

  /* Cut the part of the backdrop that falls outside of <nav /> */
  --cutoff: calc(100% - var(--extended-by));
  -webkit-mask-image: linear-gradient(to bottom,
    black 0,
    black var(--cutoff),
    transparent var(--cutoff));
  transform: rotate(180deg);

```

```

`;

export const FooterContainer = styled.div`
  display: flex;
  flex-direction: column;
  justify-content: center;
  max-width: 1000px;
  margin: 0 auto;
  padding-bottom: 50px;
`;

export const Column = styled.div`
  display: flex;
  flex-direction: column;
  text-align: left;
  margin-left: 60px;
`;

export const Row = styled.div`
  display: grid;
  grid-template-columns: repeat(
    auto-fill,
    minmax(185px, 1fr)
  );
  grid-gap: 20px;

  @media (max-width: 1000px) {
    grid-template-columns: repeat(
      auto-fill,
      minmax(200px, 1fr)
    );
  }
`;

export const FooterLink = styled.a`
  color: #fff;
  margin-bottom: 20px;
  font-size: 18px;
  text-decoration: none;
  text-style: none;

  &:hover {
    color: #ffbe98;
    transition: 200ms ease-in;
  }
`;

export const Heading = styled.p`
  font-size: 24px;
  color: #ffbe98;
  margin-bottom: 40px;
  font-weight: bold;
`;

export const GlassBorder = styled.div`
  position: relative;
  z-index: -1;
  left: 0;
  right: 0;

  --extended-by: 80px;
  --offset: 20px;
  --thickness: 2px;
  height: calc(var(--extended-by) + var(--offset));
  /* Offset is used to snuck the border backdrop slightly under the main backdrop
  for smoother effect */
  top: calc(100% - var(--offset) + var(--thickness));

```

```

/* Make the blur bigger so that the light bleed effect spreads wider than blur on
the first backdrop */
/* Increase saturation and brightness to fake smooth chamfered edge reflections */
--filter: blur(90px) saturate(160%) brightness(1.3);
-webkit-backdrop-filter: var(--filter);
backdrop-filter: var(--filter);
pointer-events: none;

-webkit-mask-image: linear-gradient(
to bottom,
black 0,
black var(--offset),
transparent var(--offset)
);
margin-bottom: -18px;
transform: rotate(180deg);
`;

export const FlippedBox = styled.div`
transform: rotate(180deg);
`;

const Footer = () => {
  return (
    <Box>
      <GlassBorder />
      <GlassBox>
        <FlippedBox>
          <h1>
            style={{
              color: "#ed493c",
              textAlign: "center",
              paddingTop: "30px",
              paddingBottom: "50px",
            }}
          >
            ЯТёручка - здесь каждый найдёт свой вкус!
          </h1>
          <FooterContainer>
            <Row>
              <Column>
                <Heading>0 нас</Heading>
                <FooterLink href="#">
                  Цели
                </FooterLink>
                <FooterLink href="#">
                  Видение
                </FooterLink>
                <FooterLink href="#">
                  Отзывы
                </FooterLink>
              </Column>
              <Column>
                <Heading>Услуги</Heading>
                <FooterLink href="#">
                  Написать нам
                </FooterLink>
                <FooterLink href="#">
                  Стажировка
                </FooterLink>
                <FooterLink href="#">
                  Обучение
                </FooterLink>
              </Column>
              <Column>
                <Heading>Мы</Heading>
                <FooterLink href="#">
                  Анастасия
                </FooterLink>
              </Column>
            </Row>
          </FooterContainer>
        </FlippedBox>
      </GlassBox>
    </Box>
  );
};

```



```

        <FooterLink href="https://github.com/Kseen715">
            Денис
        </FooterLink>
        <FooterLink href="https://github.com/kotikgriga27">
            Дарья
        </FooterLink>
    </Column>
    <Column>
        <Heading>Соцсети</Heading>
        <FooterLink href="#">
            Facebook
        </FooterLink>
        <FooterLink href="#">
            ВКонтакте
        </FooterLink>
        <FooterLink href="#">
            Twitter
        </FooterLink>
        <FooterLink href="#">
            Youtube
        </FooterLink>
    </Column>
</Row>
</FooterContainer>
</FlippedBox>
</GlassBox>
</Box >
    );
};
export default Footer;

```

## Приложение 5. Файл «Goods.js»

```

import React from "react";

import Header from './Header';
import Footer from './Footer';
import Title from './Service';
import TableV from './TableView';

const cols = [
    {
        field: 'id',
        headerName: 'ID',
        width: 100,
        editable: true,
    },
    {
        field: 'expiration_date',
        headerName: 'Срок годности',
        width: 180,
        editable: true,
    },
    {
        field: 'price',
        headerName: 'Цена',
        type: 'number',
        align: 'left',
        headerAlign: 'left',
        width: 150,
        editable: true,
    },
]

function Goods() {
    return (

```

```

    <div className="App">
      {Title('Товары | ЯТёручка')}
      <Header />
      <div class="content" style={
        {
          color: '#ffffff',
        }
      }>
        <div className="table-v">
          {TableV(cols, 'sel/product')}
        </div>
      </div>
      <Footer />
    </div >
  );
};

export default Goods;

```

## Приложение 6. Файл «Groups.js»

```

import React from "react";

import Header from './Header';
import Footer from './Footer';
import Title from './Service';
import TableV from './TableView';

const cols = [
  {
    field: 'id',
    headerName: 'ID',
    width: 100,
    editable: true,
  },
  {
    field: 'type',
    headerName: 'Тип',
    width: 180,
    editable: true,
  },
  {
    field: 'number',
    headerName: 'Количество',
    width: 170,
    editable: true,
  },
  {
    field: 'product_id',
    headerName: 'ID товара',
    type: 'number',
    align: 'left',
    headerAlign: 'left',
    width: 150,
    editable: true,
  },
  {
    field: 'store_id',
    headerName: 'ID магазина',
    width: 150,
    editable: true,
  },
],

function Groups() {
  return (

```

```

    <div className="App">
      {Title('Группы товаров | ЯТёручка')}
      <Header />
      <div class="content" style={
        {
          color: '#ffffff',
        }
      }>
        <div className="table-v">
          {TableV(cols, 'sel/group')}
        </div>
      </div>
      <Footer />
    </div >
  );
};

export default Groups;

```

## Приложение 7. Файл «Header.jsx»

```

import React from 'react'
import { useNavigate } from "react-router-dom";
import Button from "react-bootstrap/Button";

function Header() {
  const navigate = useNavigate();
  return (
    <nav className="bien-nav">
      <div className="bien-glass"></div>
      <div className="bien-glass-edge"></div>
      <div className="header">
        <a className="logo-text" href="/">
          ЯТёручка
        </a>
        <Button className="button-header" id="buttons-places" onClick={() =>
          navigate('/places')}>Точки</Button>
        <Button className="button-header" id="buttons-workers" onClick={() =>
          navigate('/workers')}>Работники</Button>
        <Button className="button-header" id="buttons-groups" onClick={() =>
          navigate('/groups')}>Группы товаров</Button>
        <Button className="button-header" id="buttons-goods" onClick={() =>
          navigate('/goods')}>Товары</Button>
        <Button className="button-header" id="buttons-history" onClick={() =>
          navigate('/history')}>История покупок</Button>
      </div>
    </nav >
  )
}

export default Header

```

## Приложение 8. Файл «Header.css»

```

@import url('https://fonts.googleapis.com/css2?family=Open+Sans:ital@1&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Open+Sans&display=swap');

nav {
  position: sticky;
  left: 0;
  right: 0;
  top: 0;
  height: 100px;
  z-index: 99999;
}

```

```

.bien-glass {
  position: absolute;
  inset: 0;
  z-index: -0;
  /* Extend the backdrop to the bottom for it to "collect the light"
  outside of the nav */
  --extended-by: 100px;
  bottom: calc(-1 * var(--extended-by));

  --filter: blur(20px);
  -webkit-backdrop-filter: var(--filter);
  backdrop-filter: var(--filter);
  pointer-events: none;

  /* Cut the part of the backdrop that falls outside of <nav /> */
  --cutoff: calc(100% - var(--extended-by));
  --mask-uni: linear-gradient(to bottom,
    black 0,
    black var(--cutoff),
    transparent var(--cutoff));
  -webkit-mask-image: var(--mask-uni);
  mask-image: var(--mask-uni);
}

.bien-glass-edge {
  position: absolute;
  z-index: -1;
  left: 0;
  right: 0;

  --extended-by: 80px;
  --offset: 20px;
  --thickness: 2px;
  height: calc(var(--extended-by) + var(--offset));
  /* Offset is used to snuck the border backdrop slightly under the main
  backdrop for smoother effect */
  top: calc(100% - var(--offset) + var(--thickness));

  /* Make the blur bigger so that the light bleed effect spreads wider
  than blur on the first backdrop */
  /* Increase saturation and brightness to fake smooth chamfered edge
  reflections */
  --filter: blur(90px) saturate(160%) brightness(1.3);
  -webkit-backdrop-filter: var(--filter);
  backdrop-filter: var(--filter);
  pointer-events: none;

  --mask-img: linear-gradient(to bottom,
    black 0,
    black var(--offset),
    transparent var(--offset));
  mask-image: var(--mask-img);
  -webkit-mask-image: var(--mask-img);
}

/* ***** */

* {
  margin: 0;
  padding: 0;
}

@keyframes blink {
  0% {
    opacity: 0;
  }

  25% {

```

```

        opacity: 1;
    }

    75% {
        opacity: 1;
    }

    100% {
        opacity: 0;
    }
}

.logo-text {
    font-size: 3rem;
    font-weight: 900;
    color: #fff;
    font-style: italic;
    letter-spacing: -1px;
    padding-left: 10px;
    font-family: 'Open Sans', sans-serif;
    margin-right: 5rem;
    text-decoration: none;
}

.header {
    position: sticky;
    grid-template-columns: 1fr 1fr;
    grid-template-rows: 1fr 1fr;
    padding-top: 15px;
}

.button-header {
    font-family: 'Open Sans', sans-serif;
    font-size: 1rem;
    color: #ffbe98;

    margin-top: 0px;
    margin-bottom: 10px;
    margin-left: 0;
    margin-right: 10px;

    border-width: 2px;
    border-radius: 10px;
    border-color: #ffbe98;
    border-style: inset;

    padding-top: 7px;
    padding-bottom: 7px;
    padding-left: 20px;
    padding-right: 20px;

    min-width: 150px;
    max-width: 300px;
    min-height: 50px;
    max-height: 50px;

    text-align: center;

    background-color: #292929;
    box-shadow: #ffbe98;
    transform: translateY(-10px);
}

.button-header:hover {
    background: #ffbe98;
    color: #292929;
}

```

## Приложение 9. Файл «History.js»

```
import React from "react";

import Header from './Header';
import Footer from './Footer';
import Title from './Service';
import TableV from './TableView';

const cols = [
  {
    field: 'id',
    headerName: 'ID',
    width: 100,
    editable: true,
  },
  {
    field: 'date',
    headerName: 'Дата',
    width: 180,
    editable: true,
  },
  {
    field: 'time',
    headerName: 'Время',
    width: 170,
    editable: true,
  },
  {
    field: 'total',
    headerName: 'Итого',
    type: 'number',
    align: 'left',
    headerAlign: 'left',
    width: 200,
    editable: true,
  },
  {
    field: 'product_group_id',
    headerName: 'Группа',
    width: 150,
    editable: true,
  },
  {
    field: 'store_id',
    headerName: 'ID магазина',
    width: 150,
    editable: true,
  },
  {
    field: 'number_sold_group',
    headerName: 'Продано',
    width: 150,
    editable: true,
  },
]

function History() {
  return (
    <div className="App">
      {Title('История покупок | ЯТёрочка')}
      <Header />
      <div className="content" style={
        {
          color: '#ffffff',
        }
      }>
        <div className="table-v">
```

```

        {TableV(cols, 'sel/check')}}
      </div>
    </div>
    <Footer />
  </div >
);
};
export default History;

```

## Приложение 10. Файл «Home.js»

```

import React from "react";

import Header from './Header';
import Footer from './Footer';
import Title from './Service';

import Cat from './imgs/cat.png';

const Home = (props) => {
  return (
    <div className="App">
      {Title('Главная | ЯТёручка')}
      <Header />

      <div class="content">
        <img
          width={700}
          src={Cat}
          alt='cat'
          class='cat'
        />
      </div>

      <Footer />
    </div >
  );
};

export default Home;

```

## Приложение 11. Файл «Places.js»

```

import React from "react";

import Header from './Header';
import TableV from './TableView';
import Footer from './Footer';
import Title from './Service';

const cols = [
  {
    field: 'id',
    headerName: 'ID',
    width: 100,
    editable: true,
  },
  {
    field: 'store_name',
    headerName: 'Название',
    width: 180,
    editable: true,
  },
  {

```

```

        field: 'city',
        headerName: 'Город',
        width: 170,
        editable: true,
    },
    {
        field: 'address',
        headerName: 'Адрес',
        width: 200,
        editable: true,
    },
    {
        field: 'phone',
        headerName: 'Телефон',
        width: 150,
        editable: true,
    },
]

function Places() {
    return (
        <div className="App">
            {Title('Точки | ЯТёпочка')}
            <Header />
            <div className="content" style={
                {
                    color: 'ffffff',
                }
            }>
                <div className="table-v">
                    {TableV(cols, "sel/store")}
                </div>
            </div>
            <Footer />
        </div >
    );
};

export default Places;

```

## Приложение 12. Файл «Service.jsx»

```

import { /*useState,*/ useEffect } from "react";
import axios from "axios";

export function Title(ttl) {
    useEffect(() => {
        document.title = ttl;
    }, [ttl]);
}

function getCookie(name) {
    const cookieArr = document.cookie.split(";");

    for (let i = 0; i < cookieArr.length; i++) {
        let cookiePair = cookieArr[i].split("=");

        if (name === cookiePair[0].trim()) {
            return decodeURIComponent(cookiePair[1]);
        }
    }

    return null;
}

export const apiURL = "http://localhost:8000/";
export const endURL = "?format=json";

```



```

export const headers = {
  "Content-type": "application/json",
  "Access-Control-Allow-Origin": "*",
  "Access-Control-Allow-Methods": "DELETE, POST, GET, OPTIONS",
  "Access-Control-Allow-Headers": "Content-Type, Authorization, X-Requested-With",
  'X-CSRFToken': getCookie('csrftoken')
};

export async function getData(midURL) {
  const baseURL = `${apiURL}${midURL}${endURL}`;
  const response = await axios.get(`${baseURL}`, {
    headers: {
      headers,
    },
    withCredentials: true
  });
  return response.data;
}

export default Title;

```

### Приложение 13. Файл «Workers.js»

```

import React from "react";

import Header from './Header';
import Footer from './Footer';
import Title from './Service';
import TableV from './TableView';

const cols = [
  {
    field: 'id',
    headerName: 'ID',
    width: 100,
    editable: true,
  },
  {
    field: 'name',
    headerName: 'Имя',
    width: 180,
    editable: true,
  },
  {
    field: 'job',
    headerName: 'Должность',
    width: 170,
    editable: true,
  },
  {
    field: 'phone',
    headerName: 'Телефон',
    // type: 'number',
    align: 'left',
    headerAlign: 'left',
    width: 200,
    editable: true,
  },
  {
    field: 'store_id',
    headerName: 'ID магазина',
    width: 150,
    editable: true,
  },
]

function Workers() {

```

```

    return (
      <div className="App">
        {Title('Работники | ЯТёпочка')}
        <Header />
        <div class="content" style={
          {
            color: '#ffffff',
          }
        }>
          <div className="table-v">
            {TableV(cols, 'sel/employee')}
          </div>
        </div>
        <Footer />
      </div >
    );
  };
export default Workers;

```

## Приложение 14. Файл «TableView.js»

```

import * as React from 'react';
import Box from '@mui/material/Box';
import Button from '@mui/material/Button';
import AddIcon from '@mui/icons-material/Add';
import EditIcon from '@mui/icons-material/Edit';
import DeleteIcon from '@mui/icons-material/DeleteOutlined';
import SaveIcon from '@mui/icons-material/Save';
import CancelIcon from '@mui/icons-material/Close';
import {
  GridRowModes,
  DataGrid,
  GridToolbarContainer,
  GridActionsCellItem,
  GridRowEditStopReasons,
} from '@mui/x-data-grid';
import {
  randomId,
} from '@mui/x-data-grid-generator';
import './TableView.css';

import { getData } from './Service';
import { useEffect } from 'react';
import axios from 'axios';

let json_link;
let csv_link;

function EditToolbar(props) {
  const { setRows, setRowModesModel } = props;

  const handleClick = () => {
    const id = randomId();
    setRows((oldRows) => [...oldRows, { id, isNew: true }]);
    setRowModesModel((oldModel) => ({
      ...oldModel,
      [id]: { mode: GridRowModes.Edit, fieldToFocus: 'name' },
    }));
  };
  return (
    <GridToolbarContainer>
      <Button color="primary" startIcon={AddIcon} onClick={handleClick}>
        Добавить запись
      </Button>
      <div style={
        {

```

```

        marginLeft: 'auto',
      }
    }>
    <Button color="secondary" onClick={
      () => {
        window.location.href = json_link;
      }
    }>
      Выгрузить в JSON
    </Button>
    <Button color="secondary" onClick={
      () => {
        window.location.href = csv_link;
      }
    }>
      Выгрузить в CSV
    </Button>
    <Button color="primary" onClick={
      () => {
        window.location.href = 'http://localhost:8000/log/login/';
      }
    }>
      Логин
    </Button>
    <Button color="primary" onClick={
      () => {
        window.location.href = 'http://localhost:8000/log/logout/';
      }
    }>
      Выйти
    </Button>
  </div>
</GridToolbarContainer >
);
}

function ConvertHashToInt(hash) {
  let res = 0;
  for (let i = 0; i < hash.length; i++) {
    res += hash.charCodeAt(i);
  }
  return res;
}

export default function TableV(columns, subURL) {
  const [rows, setRows] = React.useState([]);

  useEffect(() => {
    json_link = 'http://localhost:8000/get-json-data/' + subURL + '/';
    csv_link = 'http://localhost:8000/get-csv-data/' + subURL + '/';
    getData(subURL)
      .then(res => {
        // console.log(res);
        setRows(res);
      })
      .catch(err => console.error(err));
  }, []); // Empty array means this effect runs once on component mount
  // const [rows, setRows] = React.useState(initialRows);
  const [rowModesModel, setRowModesModel] = React.useState({});

  const handleRowEditStop = (params, event) => {
    if (params.reason === GridRowEditStopReasons.rowFocusOut) {
      event.defaultMuiPrevented = true;
    }
  };

  const handleEditClick = (id) => () => {
    setRowModesModel({ ...rowModesModel, [id]: { mode: GridRowModes.Edit } });
  };
}

```

```

function getCookie(name) {
  const cookieArr = document.cookie.split(";");

  for (let i = 0; i < cookieArr.length; i++) {
    let cookiePair = cookieArr[i].split("=");

    if (name === cookiePair[0].trim()) {
      return decodeURIComponent(cookiePair[1]);
    }
  }

  return null;
}

const handleSaveClick = (id) => async () => {
  setRowModesModel({ ...rowModesModel, [id]: { mode: GridRowModes.View } });
};

const handleDeleteClick = (id) => () => {
  setRows(rows.filter((row) => row.id !== id));

  const response = axios({
    method: 'destroy',
    url: 'http://localhost:8000/' + subURL + '/' + id + '/',
    headers: {
      'Content-Type': 'application/json',
      'X-CSRFToken': getCookie('csrftoken')
    },
    withCredentials: true
  });
};

const handleCancelClick = (id) => () => {
  setRowModesModel({
    ...rowModesModel,
    [id]: { mode: GridRowModes.View, ignoreModifications: true },
  });

  const editedRow = rows.find((row) => row.id === id);
  if (editedRow.isNew) {
    setRows(rows.filter((row) => row.id !== id));
  }
};

const processRowUpdate = (newRow) => {
  const updatedRow = { ...newRow, isNew: false };
  setRows(rows.map((row) => (row.id === newRow.id ? updatedRow : row)));
  console.log(updatedRow);
  let row = updatedRow

  const kal = {
    'sel/employee': {
      id: row.id,
      job: row.job,
      name: row.name,
      phone: row.phone,
      store_id: row.store_id
    },
    'sel/group': {
      id: row.id,
      type: row.type,
      number: row.number,
      product_id: row.product_id,
      store_id: row.store_id
    },
    'sel/store': {
      id: row.id,

```

```

        store_name: row.store_name,
        city: row.city,
        address: row.address,
        phone: row.phone
    },
    'sel/check': {
        id: row.id,
        date: row.date,
        time: row.time,
        total: row.total,
        product_group_id: row.product_group_id,
        store_id: row.store_id,
        number_sold_group: row.number_sold_group
    },
    'sel/goods': {
        id: row.id,
        expiration_date: row.expiration_date,
        price: row.price,
    }
}

const kal2 = {
    'sel/employee': {
        id: ConvertHashToInt(row.id),
        job: row.job,
        name: row.name,
        phone: row.phone,
        store_id: row.store_id
    },
    'sel/group': {
        id: ConvertHashToInt(row.id),
        type: row.type,
        number: row.number,
        product_id: row.product_id,
        store_id: row.store_id
    },
    'sel/store': {
        id: ConvertHashToInt(row.id),
        store_name: row.store_name,
        city: row.city,
        address: row.address,
        phone: row.phone
    },
    'sel/check': {
        id: ConvertHashToInt(row.id),
        date: row.date,
        time: row.time,
        total: row.total,
        product_group_id: row.product_group_id,
        store_id: row.store_id,
        number_sold_group: row.number_sold_group
    },
    'sel/goods': {
        id: ConvertHashToInt(row.id),
        expiration_date: row.expiration_date,
        price: row.price,
    }
}

let response;
if (!row.isNew && !isNaN(row.id)) {
    response = axios({
        method: 'update',
        url: 'http://localhost:8000/' + subURL + '/' + row.id + '/',
        data: kal[subURL],
        headers: {
            'Content-Type': 'application/json',
            'X-CSRFToken': getCookie('csrftoken')
        },

```

```

        withCredentials: true
      });
    } else {
      response = axios({
        method: 'post',
        url: 'http://localhost:8000/' + subURL + '/',
        data: kal2[subURL],
        headers: {
          'Content-Type': 'application/json',
          'X-CSRFToken': getCookie('csrftoken')
        },
        withCredentials: true
      });
    }

    return updatedRow;
  };

  const handleRowModesModelChange = (newRowModesModel) => {
    setRowModesModel(newRowModesModel);
  };

  const columns_end = [
    {
      field: 'actions',
      type: 'actions',
      headerName: 'Действия',
      width: 100,
      cellClassName: 'actions',
      getActions: ({ id }) => {
        const isInEditMode = rowModesModel[id]?.mode === GridRowModes.Edit;

        if (isInEditMode) {
          return [
            <GridActionsCellItem
              icon={<CancelIcon />}
              label="Отменить"
              className="textPrimary"
              onClick={handleCancelClick(id)}
              color="inherit"
            />,
          ];
        }

        return [
          <GridActionsCellItem
            icon={<EditIcon />}
            label="Изменить запись"
            className="textPrimary"
            onClick={handleEditClick(id)}
            color="inherit"
          />,
          <GridActionsCellItem
            icon={<DeleteIcon />}
            label="Удалить запись"
            onClick={handleDeleteClick(id)}
            color="inherit"
          />,
        ];
      },
    },
  ];

  columns = columns.concat(columns_end);

  return (
    <Box className="table-v-box">
      <DataGrid
        className="table-v-grid"

```

```

        rows={rows}
        columns={columns}
        editMode="row"
        rowModesModel={rowModesModel}
        onRowModesModelChange={handleRowModesModelChange}
        onRowEditStop={handleRowEditStop}
        processRowUpdate={processRowUpdate}
        slots={{
          toolbar: EditToolbar,
        }}
        slotProps={{
          toolbar: { setRows, setRowModesModel },
        }}
      />
    </Box>
  );
}

```

## Приложение 15. Файл «TableView.css»

```

:root {
  --table-color-text: #ffffff;
  --table-color-white-border: #ffffff88;
  --table-color-icons: #ed493c;
  --table-color-background: #191919;
}

.table-v {
  width: 90vw;
  margin: 0 auto;
  color: var(--table-color-text);
}

.table-v-box {
  height: 500;
  width: 100%;
  color: var(--table-color-text);
}

.MuiDataGrid-root {
  background-color: var(--table-color-background);
}

.MuiDataGrid-main,
.MuiToolbar-gutters,
.MuiDataGrid-selectedRowCount {
  color: var(--table-color-text);
}

.MuiDataGrid-withBorderColor {
  border-color: var(--table-color-background);
}

.MuiSvgIcon-fontSizeMedium,
.MuiSelect-iconStandard,
.MuiSvgIcon-fontSizeSmall,
.MuiSelect-icon,
.MuiTablePagination-select,
.MuiTablePagination-selectIcon,
.MuiSvgIcon-root,
.css-pqjvzy-MuiSvgIcon-root-MuiSelect-icon {
  color: var(--table-color-white-border);
}

.css-ptiqhd-MuiSvgIcon-root {
  color: var(--table-color-icons);
}

```

## Приложение 16. Файл «index.js»

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import { BrowserRouter } from "react-router-dom";

import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <BrowserRouter>
      <App />
    </BrowserRouter>
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

## Приложение 17. Файл «index.css»

```
body {
  background-color: #191919;
  background-image: url("data:image/svg+xml,%3Csvg xmlns='http://www.w3.org/2000/svg'
width='260' height='260' viewBox='0 0 260 260'%3E%3Cg fill-rule='evenodd'%3E%3Cg
fill='%23ffbe98' fill-opacity='0.1'%3E%3Cpath d='M24.37 16c.2.65.39 1.32.54 2H21.17l1.17
2.34.45.9-.24.11V28a5 5 0 0 1-2.23 8.94l-.02.06a8 8 0 0 1-7.75 6h-20a8 8 0 0 1-7.74-6l-
.02-.06A5 5 0 0 1-17.45 28v-6.76l-.79-1.58-.44-.9.9-.44.63-.32H-20a23.01 23.01 0 0 1
44.37-2zm-36.82 2a1 1 0 0 0-.44.11-3.1 1.56.89 1.79 1.31-.66a3 3 0 0 1 2.69 0l2.2 1.1a1
1 0 0 0 .9 0l2.21-1.1a3 3 0 0 1 2.69 0l2.2 1.1a1 1 0 0 0 .9 0l2.21-1.1a3 3 0 0 1 2.69
0l2.2 1.1a1 1 0 0 0 .86.02l2.88-1.27a3 3 0 0 1 2.43 0l2.88 1.27a1 1 0 0 0 .85-.02l3.1-
1.55-.89-1.79-1.42.71a3 3 0 0 1-2.56.06l-2.77-1.23a1 1 0 0 0-.4-.09h-.01a1 1 0 0 0-
.4.09l-2.78 1.23a3 3 0 0 1-2.56-.06l-2.3-1.15a1 1 0 0 0-.45-.11h-.01a1 1 0 0 0-.44.11l.9
19.22a3 3 0 0 1-2.69 0l-2.2-1.1a1 1 0 0 0-.45-.11h-.01a1 1 0 0 0-.44.11l-2.21 1.1a3 3 0
0 1-2.69 0l-2.2-1.1a1 1 0 0 0-.45-.11h-.01zm0-2h-4.9a21.01 21.01 0 0 1 39.61 0h-2.09l-
.06-.13-.26.13h-32.31zm30.35 7.68l1.36-.68h1.3v2h-36v-1.15l.34-.17 1.36-
.68h2.59l1.36.68a3 3 0 0 0 2.69 0l1.36-.68h2.59l1.36.68a3 3 0 0 0 2.69 0l2.26
23h2.59l1.36.68a3 3 0 0 0 2.56.06l1.67-.74h3.23l1.67.74a3 3 0 0 0 2.56-.06zM-13.82
27l16.37 4.91l18.93 27h-32.75zm-.63 2h.34l16.66 5 16.67-5h.33a3 3 0 1 1 0 6h-34a3 3 0 1
1 0-6zm1.35 8a6 6 0 0 0 5.65 4h20a6 6 0 0 0 5.66-4H-13.1z'/%3E%3Cpath id='path6_fill-
copy' d='M284.37 16c.2.65.39 1.32.54 2H281.17l1.17 2.34.45.9-.24.11V28a5 5 0 0 1-2.23
8.94l-.02.06a8 8 0 0 1-7.75 6h-20a8 8 0 0 1-7.74-6l-.02-.06a5 5 0 0 1-2.24-8.94v-6.76l-
.79-1.58-.44-.9.9-.44.63-.32H240a23.01 23.01 0 0 1 44.37-2zm-36.82 2a1 1 0 0 0-.44.11-
3.1 1.56.89 1.79 1.31-.66a3 3 0 0 1 2.69 0l2.2 1.1a1 1 0 0 0 .9 0l2.21-1.1a3 3 0 0 1
2.69 0l2.2 1.1a1 1 0 0 0 .9 0l2.21-1.1a3 3 0 0 1 2.69 0l2.2 1.1a1 1 0 0 0 .86.02l2.88-
1.27a3 3 0 0 1 2.43 0l2.88 1.27a1 1 0 0 0 .85-.02l3.1-1.55-.89-1.79-1.42.71a3 3 0 0 1-
2.56.06l-2.77-1.23a1 1 0 0 0-.4-.09h-.01a1 1 0 0 0-.4.09l-2.78 1.23a3 3 0 0 1-2.56-.06l-
2.3-1.15a1 1 0 0 0-.45-.11h-.01a1 1 0 0 0-.44.11l-2.21 1.1a3 3 0 0 1-2.69 0l-2.2-1.1a1 1
0 0 0-.45-.11h-.01a1 1 0 0 0-.44.11l-2.21 1.1a3 3 0 0 1-2.69 0l-2.2-1.1a1 1 0 0 0-.45-
.11h-.01zm0-2h-4.9a21.01 21.01 0 0 1 39.61 0h-2.09l-.06-.13-.26.13h-32.31zm30.35
7.68l1.36-.68h1.3v2h-36v-1.15l.34-.17 1.36-.68h2.59l1.36.68a3 3 0 0 0 2.69 0l1.36-
.68h2.59l1.36.68a3 3 0 0 0 2.69 0l1.36-.68h2.59l1.36.68a3 3 0 0 0 2.56.06l1.67-
.74h3.23l1.67.74a3 3 0 0 0 2.56-.06zM246.18 27l16.37 4.91l278.93 27h-32.75zm-.63
2h.34l16.66 5 16.67-5h.33a3 3 0 1 1 0 6h-34a3 3 0 1 1 0-6zm1.35 8a6 6 0 0 0 5.65 4h20a6
6 0 0 0 5.66-4H246.9z'/%3E%3Cpath d='M159.5 21.02A9 9 0 0 0 151 15h-42a9 9 0 0 0-8.5
6.02 6 6 0 0 0 .02 11.96A8.99 8.99 0 0 0 109 45h42a9 9 0 0 0 8.48-12.02 6 6 0 0 0 .02-
11.96zM151 17h-42a7 7 0 0 0-6.33 4h54.66a7 7 0 0 0-6.33-4zm-9.34 26a8.98 8.98 0 0 0
3.34-7h-2a7 7 0 0 1-7 7h-4.34a8.98 8.98 0 0 0 3.34-7h-2a7 7 0 0 1-7 7h-4.34a8.98 8.98 0
```



0 0 3.34-7h-2a7 7 0 0 1-7 7h-7a7 7 0 1 1 0-14h42a7 7 0 1 1 0 14h-9.34zM109 27a9 9 0 0 0-  
7.48 4H101a4 4 0 1 1 0-8h58a4 4 0 0 1 0 8h-.52a9 9 0 0 0-7.48-4h-42z'/%3E%3Cpath d='M39  
115a8 8 0 1 0 0-16 8 8 0 0 0 16zm6-8a6 6 0 1 1-12 0 6 0 0 1 12 0zm-3-29v-2h8v-6H40a4  
4 0 0 0-4 4v10H22L1.33 4-.67 2h2.19L26 130h26L3.81-40H58L-.67-2L56 84H42v-6zm-4-  
4v10h2V74h8v-2h-8a2 2 0 0 0-2 2zm2 12h14.56L.67 2H22.77L.67-2H40zm13.8 4H24.2L3.62  
38h22.36L3.62-38z'/%3E%3Cpath d='M129 92h-6v4h-6v4h-6v14h-3L.24 2 3.76 32h36L3.76-32  
.24-2h-3v-14h-6v-4h-6v-4h-8zm18 22v-12h-4v4h3v8h1zm-3 0v-6h-4v6h4zm-6 6v-16h-  
4v19.17c1.6-.7 2.97-1.8 4-3.17zm-6 3.8V100h-4v23.8a10.04 10.04 0 0 0 4 0zm-6-.63V104h-  
4v16a10.04 10.04 0 0 0 4 3.17zm-6-9.17v-6h-4v6h4zm-6 0v-8h3v-4h-4v12h1zm27-12v-4h-  
4v4h3v4h1v-4zm-6 0v-8h-4v4h3v4h1zm-6-4v-4h-4v8h1v-4h3zm-6 4v-4h-4v8h1v-4h3zm7 24a12 12 0  
0 0 11.83-10h7.92L-3.53 30h-32.44L-3.53-30h7.92A12 12 0 0 0 130 126z'/%3E%3Cpath d='M212  
86v2h-4v-2h4zm4 0h-2v2h2v-2zm-20 0v.1a5 5 0 0 0-.56 9.65L.06.25 1.12 4.48a2 2 0 0 0 1.94  
1.52h.01L7.02 24.55a2 2 0 0 0 1.92 1.45h4.98a2 2 0 0 0 1.92-1.45L7.02-24.55a2 2 0 0 0  
1.95-1.52L224.5 96L.06-.25a5 5 0 0 0-.56-9.65V86a14 14 0 0 0-28 0zm4 0h6v2h-9a3 3 0 1 0  
0 6H223a3 3 0 1 0 0-6H220v-2h2a12 12 0 1 0-24 0h2zm-1.44 14L-1-4h24.88L-1 4h-22.88zm8.95  
26L-6.86-24h18.7L-6.86 24h-4.98zM150 242a22 22 0 1 0 0-44 22 22 0 0 0 0 44zm24-22a24 24  
0 1 1-48 0 24 24 0 0 1 48 0zm-28.38 17.73L2.04-.87a6 6 0 0 1 4.68 0L2.04.87a2 2 0 0 0  
2.5-.82L1.14-1.9a6 6 0 0 1 3.79-2.75L2.15-.5a2 2 0 0 0 1.54-2.12L-.19-2.2a6 6 0 0 1  
1.45-4.46L1.45-1.67a2 2 0 0 0 0-2.62L-1.45-1.67a6 6 0 0 1-1.45-4.46L.2-2.2a6 2 0 0 0-  
1.55-2.13L-2.15-.5a6 6 0 0 1-3.8-2.75L-1.13-1.9a2 2 0 0 0-2.5-.8L-2.04.86a6 6 0 0 1-4.68  
0L-2.04-.87a2 2 0 0 0-2.5.82L-1.14 1.9a6 6 0 0 1-3.79 2.75L-2.15.5a2 2 0 0 0-1.54  
2.12L.19 2.2a6 6 0 0 1-1.45 4.46L-1.45 1.67a2 2 0 0 0 0 2.62L1.45 1.67a6 6 0 0 1 1.45  
4.46L-.2 2.2a2 2 0 0 0 1.55 2.13L2.15.5a6 6 0 0 1 3.8 2.75L1.13 1.9a2 2 0 0 0  
2.5.8zm2.82.97a4 4 0 0 1 3.12 0L2.04.87a4 4 0 0 0 4.99-1.62L1.14-1.9a4 4 0 0 1 2.53-  
1.84L2.15-.5a4 4 0 0 0 3.09-4.24L-.2-2.2a4 4 0 0 1 .97-2.98L1.45-1.67a4 4 0 0 0 0-5.24L-  
1.45-1.67a4 4 0 0 1-.97-2.97L.2-2.2a4 4 0 0 0-3.09-4.25L-2.15-.5a4 4 0 0 1-2.53-1.84L-  
1.14-1.9a4 4 0 0 0-5.162L-2.03.87a4 4 0 0 1-3.12 0L-2.04-.87a4 4 0 0 0-4.99 1.62L-1.14  
1.9a4 4 0 0 1-2.53 1.84L-2.15.5a4 4 0 0 0-3.09 4.24L.2 2.2a4 4 0 0 1-.97 2.98L-1.45  
1.67a4 4 0 0 0 5.24L1.45 1.67a4 4 0 0 1 .97 2.97L-.2 2.2a4 4 0 0 0 3.09 4.25L2.15.5a4  
4 0 0 1 2.53 1.84L1.14 1.9a4 4 0 0 0 5 1.62L2.03-.87zM152 207a1 1 0 1 1 2 0 1 1 0 0 1-2  
0zm6 2a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm-11 1a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm-6 0a1 1 0 1 1 2  
0 1 1 0 0 1-2 0zm3-5a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm-8 8a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm3  
6a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm0 6a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm4 7a1 1 0 1 1 2 0 1 1 0  
0 1-2 0zm5-2a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm5 4a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm4-6a1 1 0 1 1  
2 0 1 1 0 0 1-2 0zm6-4a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm-4-3a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm4-  
3a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm-5-4a1 1 0 1 1 2 0 1 1 0 0 1-2 0zm-24 6a1 1 0 1 1 2 0 1 1  
0 0 1-2 0zm16 5a5 5 0 1 0 0-10 5 5 0 0 0 10zm7-5a7 7 0 1 1-14 0 7 7 0 0 1 14 0zm86-  
29a1 1 0 0 0 0 2h2a1 1 0 0 0 0-2h-2zm19 9a1 1 0 0 1 1-1h2a1 1 0 0 1 0 2h-2a1 1 0 0 1-1-  
1zm-14 5a1 1 0 0 0 0 2h2a1 1 0 0 0 0-2h-2zm-25 1a1 1 0 0 0 0 2h2a1 1 0 0 0 0-2h-2zm5 4a1  
1 0 0 0 0 2h2a1 1 0 0 0 0-2h-2zm9 0a1 1 0 0 1 1-1h2a1 1 0 0 1 0 2h-2a1 1 0 0 1-1-1zm15  
1a1 1 0 0 1 1-1h2a1 1 0 0 1 0 2h-2a1 1 0 0 1 0 1-1-1zm12-2a1 1 0 0 0 0 2h2a1 1 0 0 0 0-2h-  
2zm-11-14a1 1 0 0 1 1-1h2a1 1 0 0 1 0 2h-2a1 1 0 0 1 0 1-1-1zm-19 0a1 1 0 0 0 0 2h2a1 1 0 0  
0 0-2h-2zm6 5a1 1 0 0 1 1-1h2a1 1 0 0 1 0 2h-2a1 1 0 0 1 0 1-1-1zm-25 15c0-.47.01-.94.03-  
1.4a5 5 0 0 1-1.7-8 3.99 3.99 0 0 1 1.88-5.18 5 5 0 0 1 3.4-6.22 3 3 0 0 1 1.46-1.05 5 5  
0 0 1 7.76-3.27A30.86 30.86 0 0 1 246 184c6.79 0 13.06 2.18 18.17 5.88a5 5 0 0 1 7.76  
3.27 3 3 0 0 1 1.47 1.05 5 5 0 0 1 3.4 6.22 4 4 0 0 1 1.87 5.18 4.98 4.98 0 0 1-1.7  
8c.02.46.03.93.03 1.4v1h-62v-1zm.83-7.17a30.9 30.9 0 0 0-.62 3.57 3 3 0 0 1-.61-  
4.2c.37.28.78.49 1.23.63zm1.49-4.61c-.36.87-.68 1.76-.96 2.68a2 2 0 0 1-.21-  
3.71c.33.4.73.75 1.17 1.03zm2.32-4.54c-.54.86-1.03 1.76-1.49 2.68a3 3 0 0 1-.07-4.67 3 3  
0 0 1 1.56 1.99zm1.14-1.7c.35-.5.72-.98 1.1-1.46a1 1 0 0 1 1.45zm5.34-5.77c-1.03.86-  
2 1.79-2.9 2.77a3 3 0 0 0-1.11-.77 3 3 0 0 1 4-2zm42.66 2.77c-.9-.98-1.87-1.9-2.9-2.77a3  
3 0 0 1 4.01 2 3 3 0 0 0-1.1.77zm1.34 1.54c.38.48.75.96 1.1 1.45a1 1 0 1 0-1.1-  
1.45zm3.73 5.84c-.46-.92-.95-1.82-1.5-2.68a3 3 0 0 0 1.57-1.99 3 3 0 0 1-.07 4.67zm1.8  
4.53c-.29-.9-.6-1.8-.97-2.67.44-.28.84-.63 1.17-1.03a2 2 0 0 1-.2 3.7zm1.14 5.51c-.14-  
1.21-.35-2.4-.62-3.57.45-.14.86-.35 1.23-.63a2.99 2.99 0 0 1-.6 4.2zM275 214a29 29 0 0  
0-57.97 0h57.96zM72.33 198.12c-.21-.32-.34-.7-.34-1.12v-12h-2v12a4.01 4.01 0 0 7.09  
2.54c.57-.69.91-1.57.91-2.54v-12h-2v12a1.99 1.99 0 0 1-2 2 2 2 0 0 1-1.66-.88zM75  
176c.38 0 .74-.04 1.1-.12a4 4 0 0 0 6.19 2.4A13.94 13.94 0 0 1 84 185v24a6 6 0 0 1-6 6h-  
3v9a5 5 0 1 1-10 0v-9h-3a6 6 0 0 1-6-6v-24a14 14 0 0 1 14-14 5 5 0 0 0 5 5zm-17  
15v12a1.99 1.99 0 0 0 1.22 1.84 2 2 0 0 0 2.44-.72c.21-.32.34-.7.34-1.12v-12h2v12a3.98  
3.98 0 0 1-5.35 3.77 3.98 3.98 0 0 1-.65-.3V209a4 4 0 0 0 4 4h16a4 4 0 0 0 4-4v-24c.01-  
1.53-.23-2.88-.72-4.17-.43.1-.87.16-1.28.17a6 6 0 0 1-5.2-3 7 7 0 0 1-6.47-4.88A12 12 0  
0 0 58 185v6zm9 24v9a3 3 0 1 0 6 0v-9h-6z'/%3E%3Cpath d='M-17 191a1 1 0 0 0 0 2h2a1 1 0  
0 0 0-2h-2zm19 9a1 1 0 0 1 1-1h2a1 1 0 0 1 0 2H3a1 1 0 0 1-1-1zm-14 5a1 1 0 0 0 0 2h2a1  
1 0 0 0 0-2h-2zm-25 1a1 1 0 0 0 0 2h2a1 1 0 0 0 0-2h-2zm5 4a1 1 0 0 0 0 2h2a1 1 0 0 0 0-  
2h-2zm9 0a1 1 0 0 1 1-1h2a1 1 0 0 1 0 2h-2a1 1 0 0 1 0 1-1-1zm15 1a1 1 0 0 1 1-1h2a1 1 0 0 1  
0 2h-2a1 1 0 0 1 0 1-1-1zm12-2a1 1 0 0 0 0 2h2a1 1 0 0 0 0-2H4zm-11-14a1 1 0 0 1 1-1h2a1 1 0  
0 1 0 2h-2a1 1 0 0 1 0 1-1-1zm-19 0a1 1 0 0 0 0 2h2a1 1 0 0 0 0-2h-2zm6 5a1 1 0 0 1 1-1h2a1  
1 0 0 1 0 2h-2a1 1 0 0 1 0 1-1-1zm-25 15c0-.47.01-.94.03-1.4a5 5 0 0 1-1.7-8 3.99 3.99 0 0 1

```

1.88-5.18 5 5 0 0 1 3.4-6.22 3 3 0 0 1 1.46-1.05 5 5 0 0 1 7.76-3.27A30.86 30.86 0 0 1-
14 184c6.79 0 13.06 2.18 18.17 5.88a5 5 0 0 1 7.76 3.27 3 3 0 0 1 1.47 1.05 5 5 0 0 1
3.4 6.22 4 4 0 0 1 1.87 5.18 4.98 4.98 0 0 1-1.7 8c.02.46.03.93.03 1.4v1h-62v-lzm.83-
7.17a30.9 30.9 0 0 0-.62 3.57 3 3 0 0 1-.61-4.2c.37.28.78.49 1.23.63zm1.49-4.61c-.36.87-
.68 1.76-.96 2.68a2 2 0 0 1-.21-3.71c.33.4.73.75 1.17 1.03zm2.32-4.54c-.54.86-1.03 1.76-
1.49 2.68a3 3 0 0 1-.07-4.67 3 3 0 0 1.56 1.99zm1.14-1.7c.35-.5.72-.98 1.1-1.46a1 1 0
1 0-1.1 1.45zm5.34-5.77c-1.03.86-2 1.79-2.9 2.77a3 3 0 0 0-1.11-.77 3 3 0 0 1 4-2zm42.66
2.77c-.9-.98-1.87-1.9-2.9-2.77a3 3 0 0 1 4.01 2 3 3 0 0 0-1.1.77zm1.34 1.54c.38.48.75.96
1.1 1.45a1 1 0 1 0-1.1-1.45zm3.73 5.84c-.46-.92-.95-1.82-1.5-2.68a3 3 0 0 0 1.57-1.99 3
3 0 0 1-.07 4.67zm1.8 4.53c-.29-.9-.6-1.8-.97-2.67.44-.28.84-.63 1.17-1.03a2 2 0 0 1-.2
3.7zm1.14 5.51c-.14-1.21-.35-2.4-.62-3.57.45-.14.86-.35 1.23-.63a2.99 2.99 0 0 1-.6
4.2zM15 214a29 29 0 0 0-57.97 0h57.96z'/%3E%3C/g%3E%3C/g%3E%3C/svg%3E");
margin: 0;
font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
sans-serif;
-webkit-font-smoothing: antialiased;
-moz-osx-font-smoothing: grayscale;
}

code {
font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
monospace;
}

```