1. Архитектура web-приложений, ее особенности и виды

Архитектура веб-приложения в основном представляет отношения и взаимодействия между такими компонентами, как пользовательские интерфейсы, мониторы обработки транзакций, базы данных и другие. Основная цель - убедиться, что все элементы правильно работают вместе.

Логика довольно проста: когда пользователь вводит URL-адрес в браузере и нажимает «ввод», браузер делает запрос к серверу. Сервер отвечает, а затем показывает требуемую веб-страницу. Все эти компоненты создают архитектуру веб-приложения.

Все приложения состоят из двух частей - клиентской (front-end) и серверной (back-end).

Интерфейс - это визуальная часть приложения. Пользователи могут видеть интерфейс и взаимодействовать с ним. Клиентский код реагирует на действия пользователей. Серверная часть не визуальна для пользователей, но заставляет их запросы работать. Он обрабатывает бизнес-логику и отвечает на HTTP-запросы.

Поэтому, когда вы вводите свои учетные данные в регистрационную форму, вы имеете дело с внешним интерфейсом, но как только вы нажимаете «ввод» и регистрируетесь - это серверная часть заставляет его работать.

При правильной работе клиентская и серверная стороны составляют архитектуру программного обеспечения веб-приложения.

**Базовые компоненты архитектуры веб-приложений**

Веб-архитектура имеет компоненты пользовательского интерфейса и структурные компоненты. Последние также делятся на клиентские и серверные.

Компоненты пользовательского интерфейса обозначают все элементы интерфейса, такие как журналы активности, информационные панели, уведомления, настройки и многое другое. Они являются частью макета интерфейса веб-приложения.

**Структурные компоненты состоят из клиентской и серверной сторон:**

Клиентский компонент разработан с HTML, CSS или JavaScript. Веб-браузеры запускают код и преобразуют его в интерфейс, поэтому нет необходимости в настройке операционной системы.

Что касается серверного компонента, он построен на Java, .Net, Node.JS, Python и других языках программирования. Сервер состоит из двух частей - логики приложения и базы данных. Логика приложения - это центр управления веб-приложением. База данных отвечает за хранение информации (например, ваших учетных данных).

Существует четыре общих уровня веб-приложений:

Уровень представления (PL)

* Уровень обслуживания данных (DSL)
* Уровень бизнес-логики (BLL)
* Уровень доступа к данным (DAL)

Уровень представления

PL отображает пользовательский интерфейс и упрощает взаимодействие с пользователем. Уровень представления имеет компоненты пользовательского интерфейса, которые визуализируют и показывают данные для пользователей. Также существуют компоненты пользовательского процесса, которые задают взаимодействие с пользователем. PL предоставляет всю необходимую информацию клиентской стороне. Основная цель уровня представления - получить входные данные, обработать запросы пользователей, отправить их в службу данных и показать результаты.

Слой бизнес-логики

BLL несет ответственность за надлежащий обмен данными. Этот уровень определяет логику бизнес-операций и правил. Вход на сайт - это пример уровня бизнес-логики.

Уровень службы данных

DSL передает данные, обработанные уровнем бизнес-логики, на уровень представления. Этот уровень гарантирует безопасность данных, изолируя бизнес-логику со стороны клиента.

Уровень доступа к данным

DAL предлагает упрощенный доступ к данным, хранящимся в постоянных хранилищах, таких как двоичные файлы и файлы XML. Уровень доступа к данным также управляет операциями CRUD - создание, чтение, обновление, удаление.

Можно выделить несколько типов архитектуры веб-приложений, в зависимости от того, как логика приложения распределяется между клиентской и серверной сторонами. Наиболее распространенные архитектуры веб-приложений:

* Одностраничные веб-приложения
* Многостраничные веб-приложения
* Архитектура микросервисов
* Бессерверная архитектура
* Прогрессивные веб-приложения

Одностраничное приложение или SPA

SPA - это веб-сайт или веб-приложение, которое загружает всю необходимую информацию при входе на страницу. Одностраничные приложения имеют одно существенное преимущество - они обеспечивают потрясающий пользовательский интерфейс, поскольку пользователи не испытывают перезагрузки веб-страниц. Одностраничные веб-приложения часто разрабатываются с использованием фреймворков JavaScript, таких как Angular, React и других.

Известные СПА : Gmail, Facebook, Twitter, Slack.

Многостраничное приложение или MPA

Многостраничные приложения более популярны в Интернете, так как в прошлом все веб-сайты были MPA. В наши дни компании выбирают MPA, если их веб-сайт довольно большой (например, eBay). Такие решения перезагружают веб-страницу для загрузки или отправки информации с / на сервер через браузеры пользователей.

Известные MPA: eBay, Amazon.

Традиционная монолитная архитектура веб-приложения состоит из трех частей - базы данных, клиентской и серверной сторон. Это означает, что внутренняя и внешняя логика, как и другие фоновые задачи, генерируются в одной кодовой базе. Чтобы изменить или обновить компонент приложения, разработчики программного обеспечения должны переписать все приложение.

Что касается микросервисов, этот подход позволяет разработчикам создавать веб-приложение из набора небольших сервисов. Разработчики создают и развертывают каждый компонент отдельно.

Архитектура микросервисов выгодна для больших и сложных проектов, поскольку каждый сервис может быть изменен без ущерба для других блоков. Поэтому, если вам нужно обновить логику оплаты, вам не придется на время останавливать работу сайта.

Известные проекты: Netflix, Uber, Spotify, PayPal.

Прогрессивные веб-приложения или PWA

Одна из основных тенденций в разработке веб-приложений последних лет - это прогрессивные веб-приложения. Это веб-решения, которые работают как собственные приложения на мобильных устройствах. PWA предлагают push-уведомления, автономный доступ и возможность установить приложение на домашний экран.

Для создания PWA разработчики используют «языки веб-программирования», такие как HTML, CSS и JavaScript. Если приложению требуется доступ к функциям устройств, разработчики используют дополнительные API - NFC API, API геолокации, Bluetooth API и другие. Известные PWA: Uber, Starbucks, Pinterest.

1. Практическая часть

Создано 3 HTML-страницы, связанных между собой ссылками. На главной странице в шапке сайта размешены ссылки: главная/производственные показатели/регистрация и осуществлено отображение даты и времени с применением javascript. На странице «производственные показатели» размещены данные информационного характера. На странице «регистрация» размещена форма регистрации, на которой использовались текстовые поля, флажки, меню select, кнопки «Зарегистрироваться» и «Очистить».

Код CSS-файла:

@charset "utf-8";

.body {

background-color: rgb(0, 0, 0);

color: white;

}

.headre{

padding-left: 200px;

padding-right: 200px;

}

.hcont{

width:100%;

}

.hcont div{

width:49%;

padding-top: 30px;

padding-bottom: 30px;

display:inline-block;

font-size: 20px;

}

.h3 {

color: #dededf;

}

.menu{

text-align: left;

}

.timer{

text-align: right;

color: #dededf;

}

.content {

margin-left: 200px;

margin-right: 200px;

font-size: medium;

font-style: normal;

color: #dededf;

background-color: #0A012F;

overflow: hidden;

height: auto;

}

.divc{

text-align: center;

}

.divp{

text-align: left;

padding-left: 5%;

padding-right: 5%;

color: #dededf;

}

.mimg{

width: 90%;

}

.footre{

margin-left: 200px;

margin-right: 200px;

}

.table{

width: 100%;

margin-bottom: 20px;

border: 1px solid #dddddd;

border-collapse: collapse;

}

.thead{

background-color: rgb(140, 167, 218);

color: rgb(68, 63, 63);

font-size: x-large;

}

.table th {

font-weight: bold;

padding: 5px;

background: #efefef;

border: 1px solid #dddddd;

}

.table td {

border: 1px solid #dddddd;

padding: 5px;

}

.fcont div{

padding-top: 10px;

padding-bottom: 10px;

font-size: 20px;

}

.divbot{

height: 100px;

margin-left: 200px;

margin-right: 200px;

font-size: medium;

font-style: normal;

color: #dededf;

text-align: center;

}

Код главной HTML-страницы

<html><head>

<title>БАЭС - Главаня</title>

<link href="KR.css" rel="stylesheet" type="text/css"></head>

<body><header>

<div class="headre">

<div class="hcont"{>

<div class="menu">

<a href="index.html" class="h3">Главная</a>&nbsp;|

<a href="news.html" class="h3">Производственные показатели</a>&nbsp;|

<a href="registration.html" class="h3">Регистрация</a></div>

<div class="timer">

<script type="text/javascript">

var d=new Date();

var day=d.getDate();

var month=d.getMonth() + 1;

var year=d.getFullYear();

document.write("Дата: " + day + "." + month + "." + year + " Время: " + d.toLocaleTimeString());

</script></div></div></div></div></header>

<main class="content">

<div class="divc">

<h2>САЙТ БАЛАКОВСКОЙ АЭС</h2>

<img src="img\bal10.jpg" alt="" class="mimg" ></div>

<div class="divp">

<p><b>Место расположения: вблизи г. Балаково (Саратовская обл.)</b></p>

<p><b>Тип реактора: ВВЭР-1000</b></p>

<p><b>Количество энергоблоков: 4</b></p>

<p><b>Балаковская АЭС – одно из крупнейших предприятий атомной энергетики России. На станции эксплуатируются реакторы типа ВВЭР-1000 (проект В-320). В настоящее время она ежегодно вырабатывает порядка 30 миллиардов кВтч электроэнергии. Доля Балаковской АЭС в общей генерации электроэнергии, которая вырабатывается в Саратовской области, – более 75%. Ее электроэнергия поступает потребителям Поволжья, Центральной России, Урала и Сибири.</b></p>

<p><b>Балаковская АЭС – признанный лидер атомной энергетики России по многим показателям. Станция 17 раз удостаивалась звания «Лучшая АЭС России» (по итогам работы в 1995, 1999, 2000, 2003, 2005–2009 и 2011, 2012, 2013, 2014, 2016 и 2017, 2018 и 2019 гг.).</b></p>

<p><b>В десятый раз – по итогам 2020 года – Балаковская АЭС признана лучшей среди российских атомных станций по культуре безопасности.</b></p>

<p><b>Одним из приоритетных направлений деятельности Балаковской АЭС, соответствующим общемировой тенденции в атомной энергетике, является продление срока эксплуатации энергоблоков. В 2015 г. станция получила лицензию на продление срока эксплуатации энергоблока №1 еще на 30 лет, в 2017 г. – энергоблока №2 на 26 лет, в 2018 г. – энергоблока №3 на дальнейшую эксплуатацию сроком на 30 лет. Этому предшествовала масштабная работа по модернизации систем и оборудования, в т.ч. в области безопасности.</b></p>

<p><b>Одним из приоритетных направлений деятельности Балаковской АЭС, соответствующим общемировой тенденции в атомной энергетике, является продление срока эксплуатации энергоблоков. В 2015 г. станция получила лицензию на продление срока эксплуатации энергоблока №1 еще на 30 лет, в 2017 г. – энергоблока №2 на 26 лет, в 2018 г. – энергоблока №3 на дальнейшую эксплуатацию сроком на 30 лет. Этому предшествовала масштабная работа по модернизации систем и оборудования, в т.ч. в области безопасности.</b></p>

<p><b>Важной сферой инновационной деятельности Балаковской АЭС является внедрение производственной системы Росатома (ПСР). Она концентрируется на непрерывных улучшениях производственных процессов при одновременном снижении затрат. Балаковская АЭС является признанным лидером отрасли в сфере внедрения ПСР.</b></p>

<p><b>За все 35 лет работы Балаковской АЭС негативного воздействия на окружающую среду не отмечено. Радиационная обстановка в районе расположения Балаковской АЭС не изменилась и находится на уровне фоновых значений, характерных для европейской части России, которые наблюдались здесь до начала строительства станции. Это – свидетельство высокого уровня ее экологической безопасности. </b></p>

<p><b>Ежегодно на атомной станции совершенствуется система экологического менеджмента, совершенствуются технологии, персонал непрерывно повышает знания в области радиационной безопасности. В 2020 году продолжается работа по обеспечению экологической безопасности и рационального использования природных ресурсов. Среди запланированных мероприятий ключевым является ввод в эксплуатацию узла по сортировке отходов производства и потребления.</b></p>

<p><b>Балаковская АЭС признана победителем Всероссийского конкурса «Лидер природоохранной деятельности в России-2020» в номинации «Экологическая культура в промышленности и энергетике». Это звание присваивается предприятию уже в 13-й раз.</b></p></div></main>

<footer><div clss="footre">

<div class="fcont">

<div class="divbot">

<p>(C) BMS (Buggs making studio). Все права защищены.</p>

<p>Разработчик: Краснов В.А., ИФСТ-4з</p></div></div></footer></body>

</html>

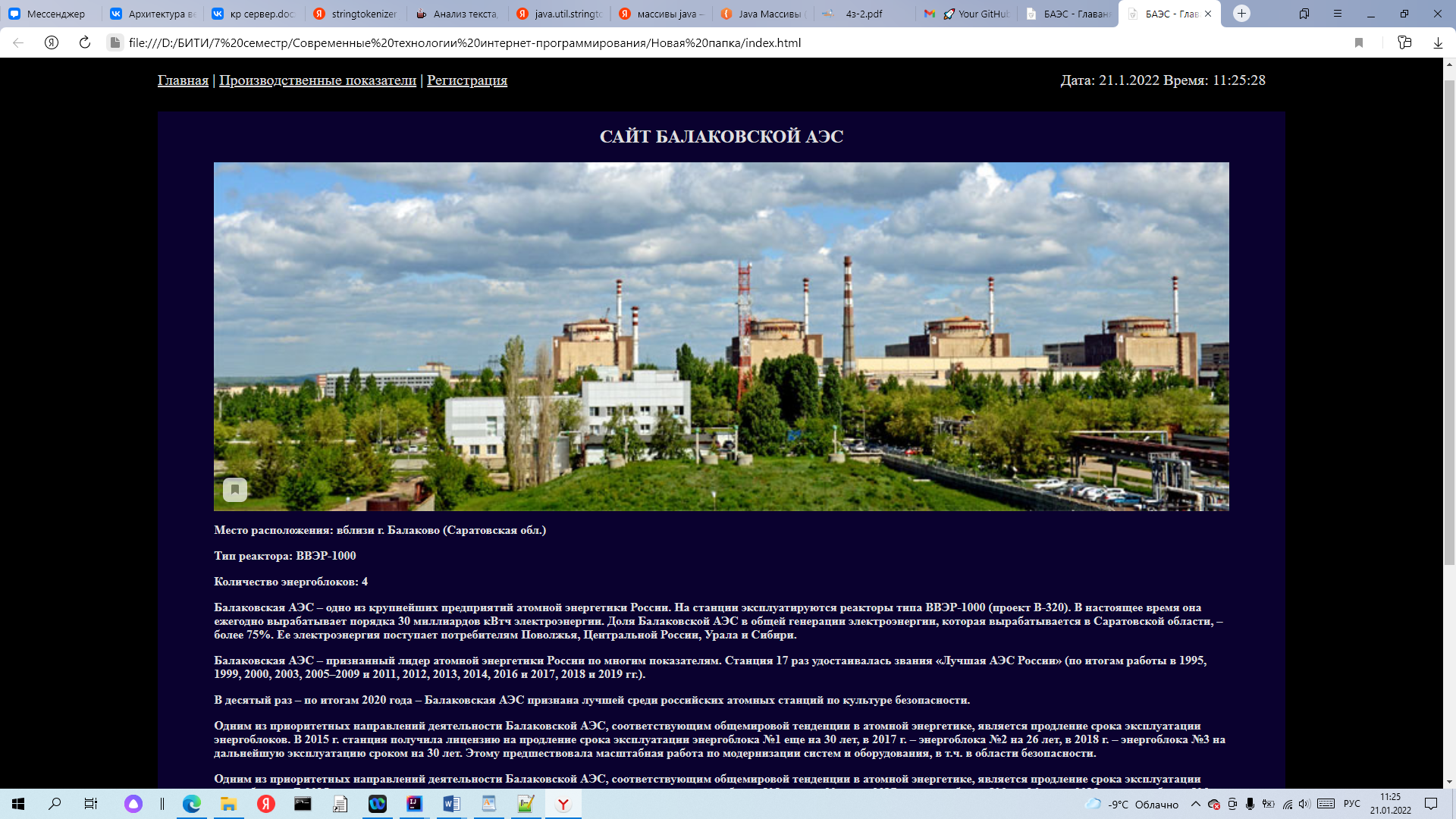


Рисунок 1 – Главная страница

Код страницы производственные показатели:

<html><head>

<title>БАЭС - Производственные показатели</title>

<link href="KR.css" rel="stylesheet" type="text/css"></head><body><header>

<div class="headre">

<div class="hcont"{>

<div class="menu">

<a href="index.html" class="h3">Главная</a>&nbsp;|

<a href="news.html" class="h3">Производственные показатели</a>&nbsp;|

<a href="registration.html" class="h3">Регистрация</a></div>

<div class="timer">

<script type="text/javascript">

var d=new Date();

var day=d.getDate();

var month=d.getMonth() + 1;

var year=d.getFullYear();

document.write("Дата: " + day + "." + month + "." + year + " Время: " + d.toLocaleTimeString());

</script></div></div></div></div></header>

<main class="content">

<div class="divc">

<h2>ПРОИЗВОДСТВЕННЫЕ ПОКАЗАТЕЛИ</h2>

<img src="img\bal9.jpg" alt="" class="mimg" ></div>

<div class="divp"><p>

<b>ПРОИЗВОДСТВЕННЫЕ ПОКАЗАТЕЛИ БАЛАКОВСКОЙ АЭС ЗА 2020 Г.

</b></p><p>

<table class="table">

<thead class="thead"><tr><td>

ПОКАЗАТЕЛИ

</td><td>

ЕД. ИЗМ.

</td><td>

ЗНАЧЕНИЯ

</td></tr></thead><tbody><tr><td>

Выработано электроэнергии

</td><td>

млн кВт\*ч

</td><td>

30 627,397

</td></tr><tr><td>

Выполнение балансового задания

</td><td>

%

</td><td>

98,7

</td></tr><tr><td>

КИУМ

</td><td>

%

</td>

<td>

87,17

</td></tr></tbody></table></p><p>

<b>ПРОИЗВОДСТВЕННЫЕ ПОКАЗАТЕЛИ БАЛАКОВСКОЙ АЭС ЗА 2019 Г.</b></p>

<p><table class="table">

<thead class="thead"><tr><td>

ПОКАЗАТЕЛИ

</td><td>

ЕД. ИЗМ.

</td><td>

ЗНАЧЕНИЯ

</td></tr></thead><tbody><tr><td>

Выработано электроэнергии

</td><td>

млн кВт\*ч

</td><td>

29 995,241

</td></tr><tr><td>

Выполнение балансового задания

</td><td>

%

</td><td>

103,4

</td></tr><tr><td>

КИУМ

</td><td>

%

</td><td>

85,6

</td></tr></tbody></table></p>

<p><b>ПРОИЗВОДСТВЕННЫЕ ПОКАЗАТЕЛИ БАЛАКОВСКОЙ АЭС ЗА 2018 Г.</b></p>

<p>

<table class="table">

<thead class="thead"><tr><td>

ПОКАЗАТЕЛИ

</td><td>

ЕД. ИЗМ.

</td><td>

ЗНАЧЕНИЯ

</td></tr></thead><tbody><tr><td>

Выработано электроэнергии

</td><td>

млн кВт\*ч

</td><td>

31861,39

</td>

</tr>

<tr>

<td>

Выполнение балансового задания

</td><td>

%

</td><td>

103,5

</td></tr><tr><td>

КИУМ

</td><td>

%

</td><td>

90,93

</td></tr></tbody></table></p></div></main><footer>

<div clss="footre">

<div class="fcont">

<div class="divbot">

<p>(C) BMS (Buggs making studio). Все права защищены.</p>

<p>Разработчик: Краснов В.А., ИФСТ-4з</p> </div></div></footer></body></html>

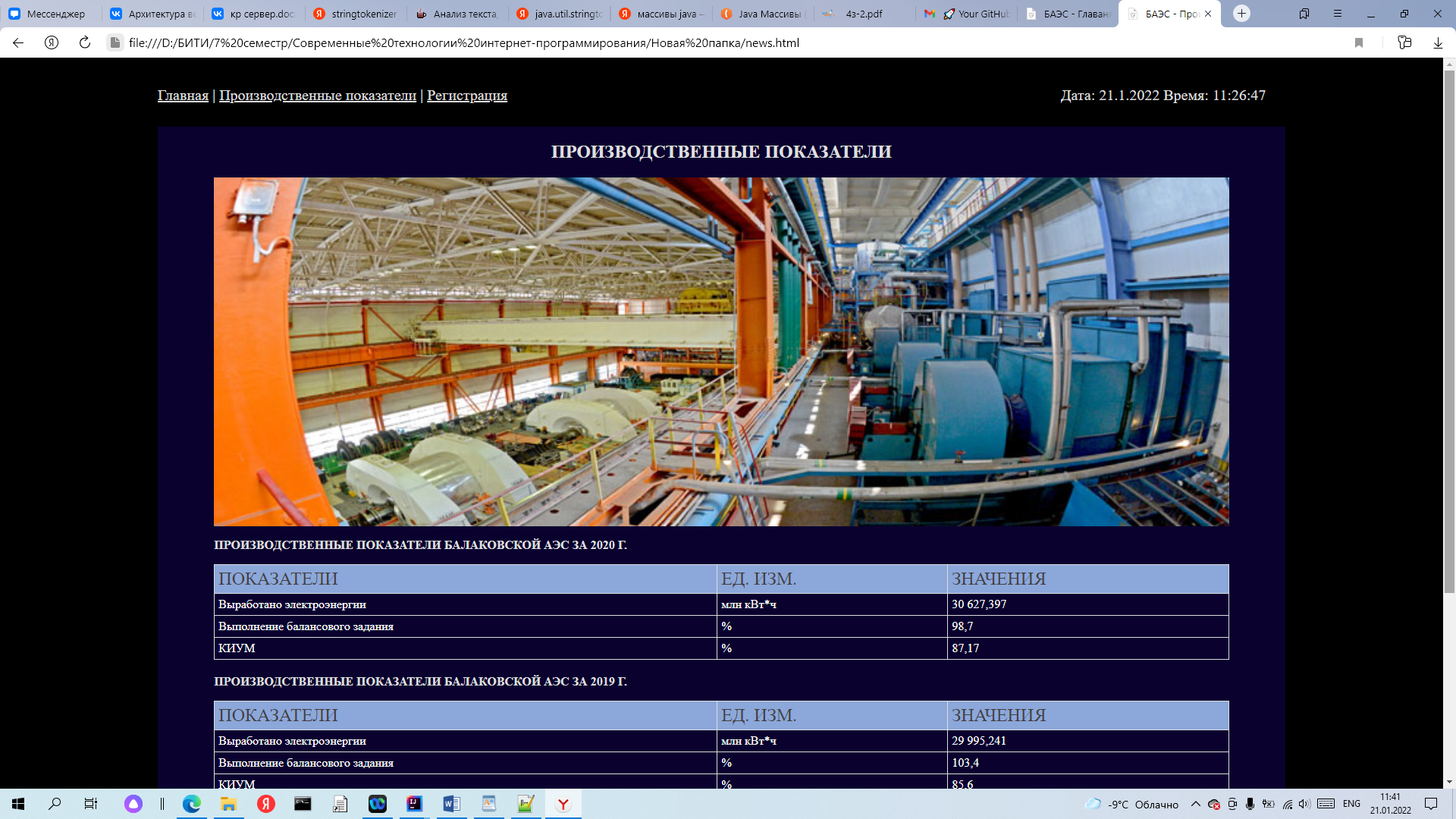


Рисунок 2 – Страница производственные показатели

Код страницы регистрации:

<html><head><title>БАЭС - Регистрация</title>

<link href="KR.css" rel="stylesheet" type="text/css"></head>

<body><header><div class="headre">

<div class="hcont"{>

<div class="menu">

<a href="index.html" class="h3">Главная</a>&nbsp;|

<a href="news.html" class="h3">Производственные показатели</a>&nbsp;|

<a href="registration.html" class="h3">Регистрация</a>

</div>

<div class="timer">

<script type="text/javascript">

var d=new Date();

var day=d.getDate();

var month=d.getMonth() + 1;

var year=d.getFullYear();

document.write("Дата: " + day + "." + month + "." + year + " Время: " + d.toLocaleTimeString());</script></div></div></div></div></header>

<main class="content">

<div class="divc">

<h2>РЕГИСТРАЦИЯ</h2>

<img src="img\bal12.jpg" alt="" class="mimg" ></div>

<div class="divp">

<script language="JavaScript">

function start(form) {

if (form.login.value == "" , form.pass.value == "" ,form.pass2.value == "" ,form.name.value == "" ,form.email.value == "")

alert("Пожалуйста, заполните все поля!")

else {

alert("Спасибо за регистрацию, "+form.name.value+"!");

const user = {

login: form.login.value,

password: form.pass.value,

name: form.name.value,

email: form.email.value,

post: form.region.value};

var messege = "login:"+form.login.value+";password:"+form.pass.value;

const request = new XMLHttpRequest();

request.open("POST", "post.php", true);

request.setRequestHeader("registration","localhost:4444");

request.send(messege);

document.getElementById('out\_arr').innerHTML =

"login: " + user.login + "<br>"+

"password: " + user.password + "<br>"+

"name: " + user.name + "<br>"+

"email: " + user.email + "<br>"+

"post: " + user.post + "<br>"}}

</script>

<p><b>Для регистрации заполните все поля.</b></p>

<p><form name="registration" method="get" action="http://mysite.faraway.com">

<p>Логин:<br>

<input name="login" type="text" size="20" maxlength="40"></p>

<p>Пароль:<br>

<input name="pass" type="password" size="20" maxlength="40"></p>

<p>Повторите пароль:<br>

<input name="pass2", type="password" size="20" maxlength="40"> </p>

<p>Имя, видимое другим (никнейм):<br>

<input name="name" type="text" size="20" maxlength="40"></p>

<p>Почта:<br>

<input name="email" type="text" size="20" maxlength="40"></p>

<p>Принадлежность к БАЭС:<br>

<select name="region" size="1">

<option value="0" selected>Сотрудник

<option value="1">Хочу устроиться

<option value="2">Хочу пройти практику(студент)

<option value="3">Не имею принадлежности</select></p>

<p>Я согласен с правилами регистрации:<input type="checkbox" name="confirm"><br></p>

<p><input type="button" value="Зарегистрироваться" onclick="start(this.form)">

<input type="button" value="Очистить все поля" onclick="this.form.reset()"></p> </form></p>

<p id="out\_arr"></p></div></main>

<footer><div clss="footre">

<div class="fcont">

<div class="divbot">

<p>(C) BMS (Buggs making studio). Все права защищены.</p>

<p>Разработчик: Краснов В.А., ИФСТ-4з</p></div></div></footer></body></html>

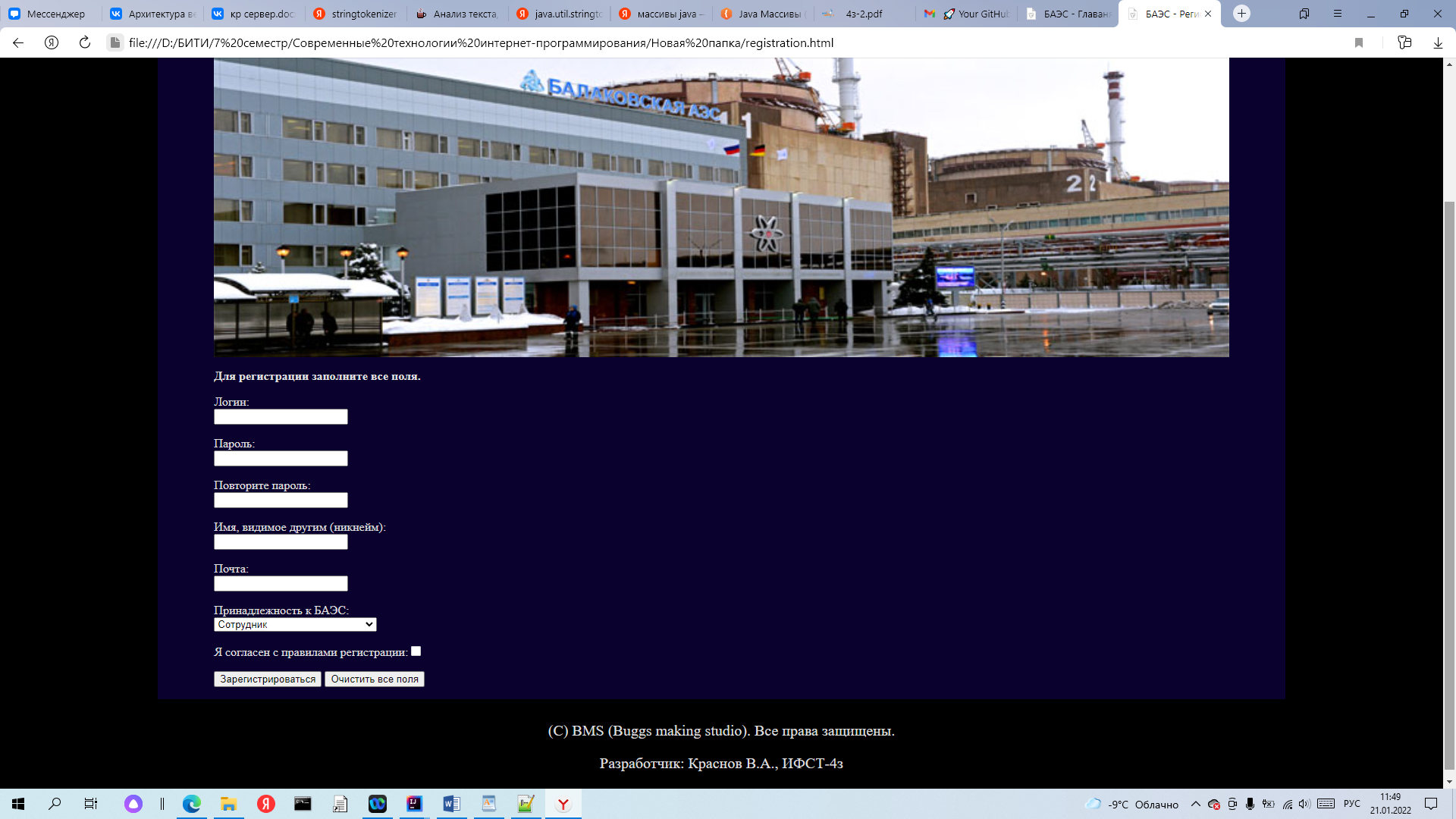


Рисунок 3 – Страница регистрации

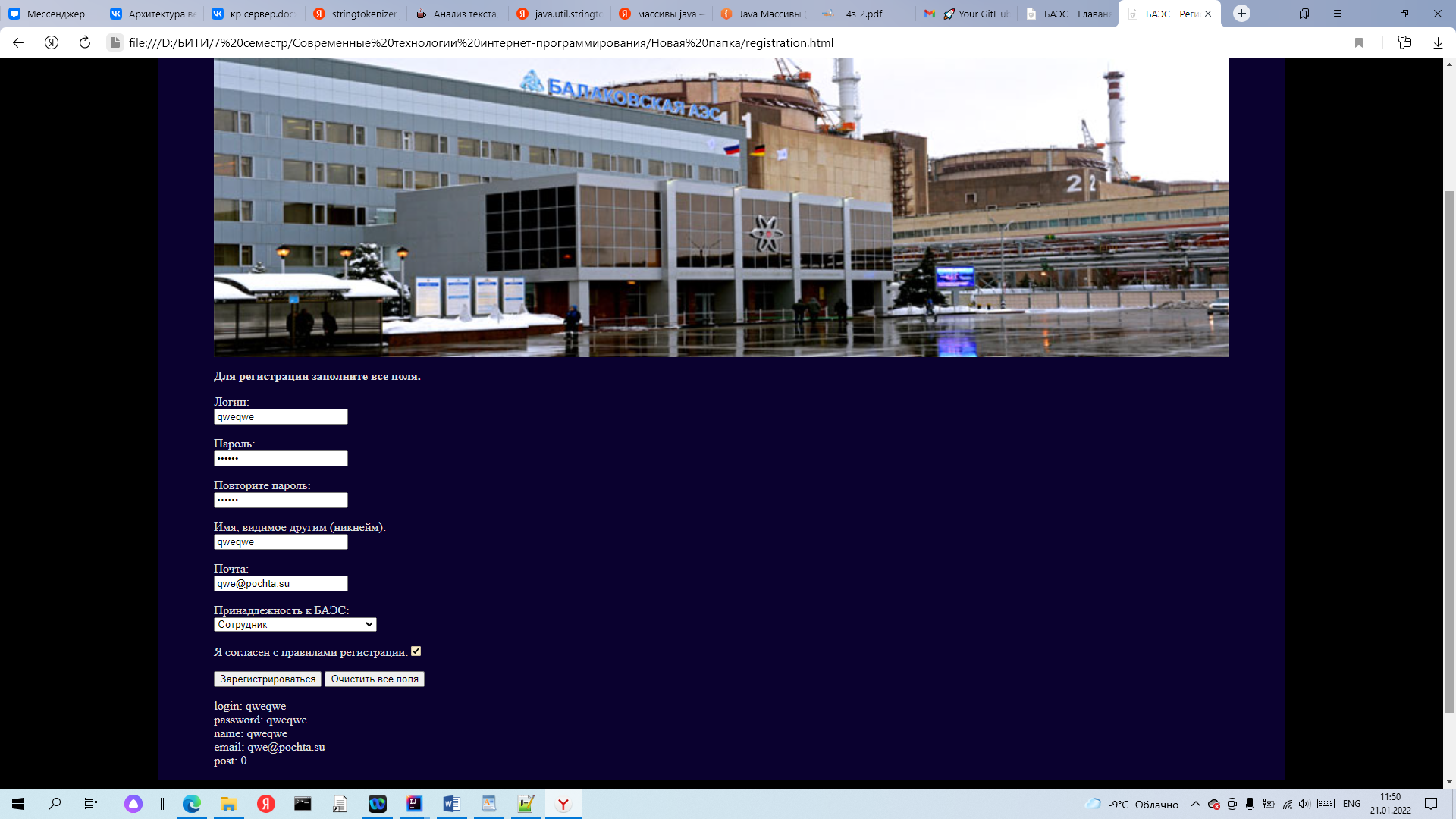


Рисунок 4 – Страница регистрации после обработки формы