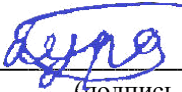


МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «КубГУ»)

**Кафедра информационных технологий**


**КУРСОВАЯ РАБОТА**

**WEB-КОНФИГУРАТОР ДЛЯ ОПРЕДЕЛЕНИЯ ПАРАМЕТРОВ  
СБОРКИ ПК**

Работу выполнил \_\_\_\_\_  \_\_\_\_\_ В.А. Бура  
(подпись, дата)

Факультет компьютерных технологий и прикладной математики 3 курс

Направление 09.03.03– «Прикладная информатика»

Научный руководитель, доц.  
канд. физ.-мат. Наук доц. \_\_\_\_\_  \_\_\_\_\_ О.В.Гаркуша  
(подпись, дата)

Нормоконтролер ст. преп. \_\_\_\_\_  \_\_\_\_\_ А.В.Харченко  
(подпись, дата)

Краснодар 2018

## РЕФЕРАТ

Курсовая работа 35 с., 12 рис., 11 табл., 3 части, 5 источников.

WEB-ПРИЛОЖЕНИЕ, ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР, PHP, БАЗА ДАННЫХ, КОМПЛЕКТУЮЩИЕ.

Цель курсовой работы – разработать web-конфигуратор для определения параметров сборки компьютера.

Для достижения поставленной цели была реализована база данных и начальная версия web-приложения.

В рамках курсовой работы была изучена литература по MySQL и PHP.

В курсовой работе описаны подобные web-конфигураторы, а также технологии, с помощью которых было реализовано web-приложение.

## СОДЕРЖАНИЕ

Введение.....	4
1 Обзор аналогичных web-приложений .....	5
2 Описание проекта и реализация .....	10
2.1 База данных.....	10
2.2 Реализация приложения .....	17
3 Описание интерфейса пользователя.....	25
Заключение .....	33
Список использованных источников .....	34

## ВВЕДЕНИЕ

На сегодняшний день, многие люди желают собрать персональный компьютер сами, тем самым сэкономив деньги за счет самостоятельной сборки и выбрав определенные характеристики, сделав акцент на задачи, для которых этот компьютер будет использоваться. Но прежде чем идти в магазин, людям необходимо изучить данную сферу, узнать какие характеристики комплектующих влияют на общую производительность. Для того чтобы облегчить процесс изучения и принятия решения было разработано WEB-приложение, ориентированное на поставленную задачу.

Данная курсовая работа посвящена изучению средств разработки Web-приложений, а также проектированию и реализации собственного приложения, которое позволяет подобрать оптимальные по стоимости и производительности, комплектующие для сборки персонального компьютера.

Первая глава курсовой работы содержит обзор подобных Web-приложений, представленных на рынке.

Вторая глава посвящена описанию проекта и реализации приложения.

Третья глава содержит описание интерфейса пользователя.

# 1 Обзор аналогичных web-приложений

Анализ предметной области – первостепенная задача любой работы. Перед проектированием и созданием самого web-приложения необходимо выделить отличия от остальных подобных сайтов, для определения вектора направления создания, развития и расширения приложения.

Многие интернет-магазины, ориентированные на цифровую технику, имеют функцию конструктора для персональных компьютеров. Пользователь из данных ему комплектующих может сам подобрать компьютер под интересующие его параметры и задачи.

Примером подобного магазина является интернет-магазин DNS [1] приведенный на рисунке 1.

The screenshot displays the DNS website's PC configuration tool. At the top, there's a search bar and navigation options like 'Сравнение', 'Избранное', and 'Корзина'. The main heading is 'Конфигуратор компьютера'. A yellow box informs that assembly is not available in the user's city. A summary shows a price of 71,247 rubles and a 'Купить без сборки' button. Below, the 'Системный блок' (System Block) is detailed with a table of components:

Component	Price (rubles)	Availability
Процессор * Intel Xeon E5-1650 v4 OEM	47 799	В магазинах: через 6 дней
Материнская плата * ASRock X99E-ITX/ac	19 799	В магазинах: через 6 дней
Корпус * DEXP DC-101B черный	1 199	в 13 магазинах
Видеокарта * Sapphire AMD Radeon R5 230 Silent LP	2 450	В магазинах: <a href="#">послезавтра</a>
Система охлаждения процессора *		+ Развернуть
Оперативная память *		+ Развернуть
Устройство хранения		

Рисунок 1 – DNS-shop сборка ПК онлайн

На представленном рисунке можно наблюдать блоки выбора комплектующих, с кратким указанием их характеристик, которые включают в себя цену. Указывается общая стоимость всех комплектующих. Также имеется функция сборки в магазине за дополнительную плату. Данный конструктор сборки персонального компьютера имеет как свои положительные стороны, так и отрицательные.

К плюсам можно отнести привязанность конструктора к одному интернет-магазину. В конкретном случае, плюсом является то, что любые изменения в товарах магазина, будут тут же применены к конфигуратору, таким образом пользователь может не беспокоиться о различиях цен, наличии товара и т.п. Так же благодаря такой привязанности появляется возможность сборки всех комплектующих в магазине профессионалом, что также является признаком надежности, а также позволяет собирать компьютеры людям, не имеющим навыков по самостоятельной сборке персонального компьютера. Отличительной чертой и еще одним плюсом является наличие показателей совместимости комплектующих. Это позволяет пользователям получить максимальную производительность.

Несмотря на то, что данный конфигуратор имеет много положительных сторон, привязанность к конкретному магазину также плохо отражается на его функционале. Одним из главных минусов – отсутствие выбора. Большое наличие различных комплектующих не может конкурировать с возможностью выбора определенного товара в разных магазинах для сравнения цены и качества. Из этого вытекает отсутствие возможности просмотраборок, основанных на товаре, которыми данный интернет-магазин не торгует, а также не возможность подбора оптимального по цене и качеству персонального компьютера.

В процессе анализа функционала данного web-приложения, необходимо отметить работу с не разбирающимся в компьютерах пользователем. Несмотря на возможность отслеживания совместимости и сборки в магазине, встает вопрос о том, а что именно следует выбрать под конкретные задачи? Какие комплектующие лучше всего под них подойдут, если пользователь не знает в чем они все отличаются?

Существует другое не самостоятельное, привязанное к интернет-магазину, web-приложение, которое позволяет ответить на некоторые из представленных выше вопросов. Сайт [hyperpc.ru](http://hyperpc.ru) [2] – конфигуратор представлен на рисунке 2.

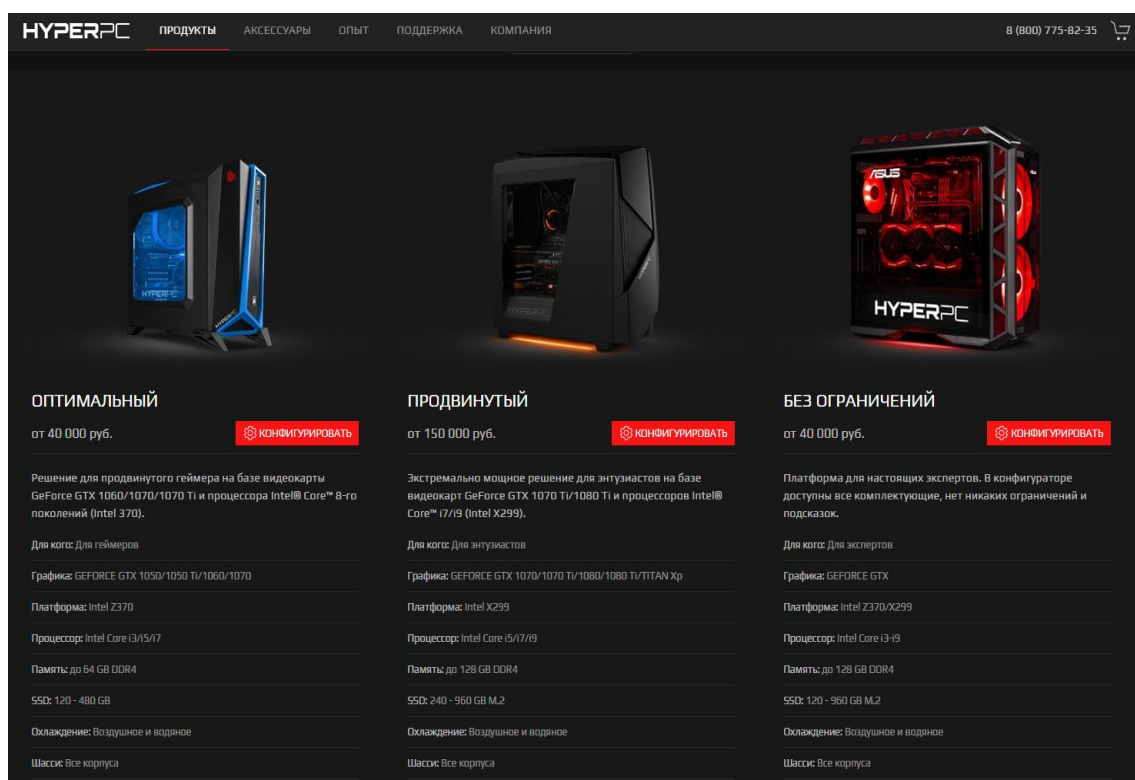


Рисунок 2 – HyperPC – Описание сборки

На данном рисунке показано, как на сайте решен вопрос с использованием конфигуратора неопытным пользователем. Изначально предоставляется три сборки на выбор. Каждая перечисляет свои конфигурации, общую стоимость, а что самое главное – описание, под какие

задачи данный выбор лучше всего подойдет. При нажатии на кнопку конфигурировать, откроется само окно configurатора, представленное на рисунке 3.

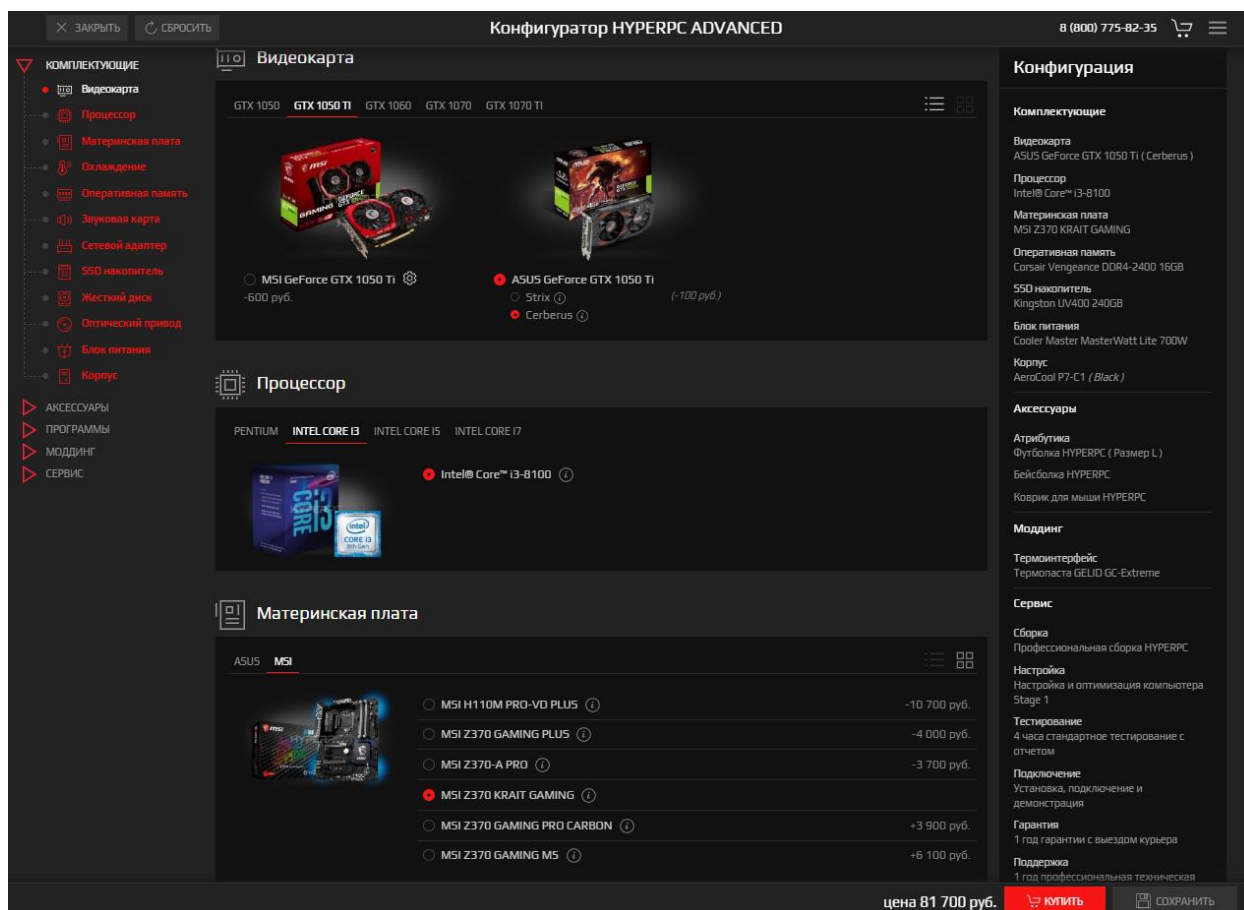


Рисунок 3 НурерPC – Окно configurатора

В правом нижнем углу рисунка 3 указывается общая стоимость сборки, слева располагается меню навигации по видам комплектующих, справа те комплектующие, которые уже выбраны пользователем или программой. В центре располагается блок выбора необходимых конфигураций.

Большой плюс у данного web-приложения – это указание не цены, а разница от текущей стоимости, что упрощает подсчет и наглядность для пользователя.

В остальном присутствуют все те же минусы из-за привязанности к конкретному магазину, что и у DNS-shop. Особенно плохо реализовано



описание комплектующих. Тут даже опытный пользователь не разберется в чем именно преимущество того или иного выбора.

К преимуществам данного конфигуратора можно отнести возможность сохранения конкретной пользовательской сборки.

В заключении необходимо подвести итог анализа подобных web-приложений. Основой для работы должна послужить самостоятельность, не привязанность к конкретному магазину. Это даст пользователю возможность подбора комплектующих и сравнения цен и качества из различных магазинов. Главным же отличием от всех остальных web-приложений послужит понятность для не знакомых с компьютерами и цифровой техникой пользователей. Требуется понятный и простой механизм подбора необходимой сборки, решающую и ориентированную под те задачи, которые будут необходимы. Так же при наличии конфигуратора, комплектующие должны подписывать не только их характеристиками, но и примерным описанием. Текст такого пояснения может содержать задачи, под которые данный выбор больше всего подойдет или список программ, под которые данный вариант лучше всего подойдет.

В итоге, должно получиться приложение удобное и понятное любому пользователю, с возможности подбора оптимальных по цене и качеству конфигураций персонального компьютера.

## 2 Описание проекта и реализация

### 2.1 База данных

База данных состоит из 11 таблиц, которые созданы с помощью СУБД MySQL[]. Был сделан выбор в сторону этой СУБД, т.к. она более простая в использовании, имеется большое наличие документации, в том числе и на русском языке.

Опишем структуру таблиц более подробно.

Для каждого комплектующего создана отдельная таблица со своим уникальным ключом, который передается в таблицу с финальной сборкой.

Таблица `CPU` предназначена для хранения информации о центральных процессорах. Ее структура представлена в таблице 1.

Таблица 1-Таблица БД CPU

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Firma	Varchar(55)	Наименование фирмы
Model	Varchar(55)	Модель
Frequency	Float	Частота процессора в Mhz
Kol-vo yader	Int(11)	Кол-во физических ядер
Img	Varchar(55)	Путь к директории с изображением процессора
description	Text	Описание возможностей ЦПУ
Price	Float	Средняя цена процессора

Таблица `video\_card` – предназначена для хранения данных о видеокарте.

Таблица 2 – Таблица БД video\_card

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Firma	Varchar(55)	Наименование фирмы видеокарты
Seria	Varchar(55)	Серия видеокарты
Model	Varchar(55)	Модель
Memory	Int(11)	Кол-во памяти видеокарты в Гб
Memory_type	Varchar(55)	Тип видеопамяти
Img	Varchar(55)	Путь к директории с изображением видеокарты
description	Text	Описание видеокарты
Price	Float	Средняя цена видеокарты

Таблица motherboard – предназначена для хранения данных о материнской плате.

Таблица 3 – Таблица БД motherboard

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Firma	Varchar(55)	Наименование фирмы материнской платы

Таблица 3 - Таблица БД motherboard (Продолжение)

Model	Varchar(55)	Модель
Img	Varchar(55)	Путь к директории с изображением материнской платы
description	text	Описание материнской платы
Price	Float	Средняя цена материнской платы

Таблица motherboard\_slots – предназначена для хранения всех имеющихся слотов различных видеокарт.

Таблица 4 – Таблица БД motherboard\_slots

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Name_slot	Varchar(55)	Наименование слота

Таблица Соотнош\_Мат\_Разъем – предназначена для взаимосвязи таблицы 3 и таблицы 4, т.е., хранит соотношение материнской платы, уникальный ключ, который передается и различных слотов, уникальный ключ которых так же передается и их количество.

Таблица 5- Таблица БД Соотнош\_Мат\_Разъем

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ

Таблица 5- Таблица БД Соотнош\_Мат\_Разъем (Продолжение)

Id_mat	Varchar(55)	Передается уникальный ключ материнской платы
Id_slot	Varchar(55)	Передается уникальный ключ слота
Kol	Varchar(55)	Количество слотов с данным ключом

Таблица RAM – предназначена для хранения данных об оперативной памяти. Название таблицы в базе данных – `RAM`.

Таблица 6 - Таблица БД RAM

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Firma	Varchar(55)	Наименование фирмы
Model	Varchar(55)	Модель
Memory	Int(11)	Кол-во памяти в Гб
Img	Varchar(55)	Путь к директории с изображением оперативной памяти
description	Text	Описание ОЗУ
Price	Float	Средняя цена оперативной памяти

Таблица power\_supply – предназначена для хранения данных о блоке питания.

Таблица 7 – Таблица БД power\_supply

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Firma	Varchar(55)	Наименование фирмы
Model	Varchar(55)	Модель
Kol_vt	Float	Количество Вт в БП
Img	Varchar(55)	Путь к директории с изображением блока питания
Description	Text	Описание БП
Price	Float	Средняя цена блока питания

Таблица hard\_drive – предназначена для хранения данных о жестком диске.

Таблица 8 – Таблица БД hard\_drive

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Firma	Varchar(55)	Наименование фирмы
Model	Varchar(55)	Модель
volume	Float	Объем памяти в Мб
Img	Varchar(55)	Путь к директории с изображением
Description	Text	Описание HDD
Price	Float	Средняя цена процессора

Таблица body – предназначена для хранения данных о корпусе.

Таблица 9 – Таблица БД body

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Firma	Varchar(55)	Наименование фирмы
Model	Varchar(55)	Модель
Img	Varchar(55)	Путь к директории с изображением корпуса
description	Text	Описание Системного блока
Price	Float	Средняя цена корпуса

Таблица BuildPC – предназначена для хранения данных о финальной сборке.

Таблица 10 – Таблица БД BuildPC

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Body_id	Int(11)	Передается уникальный ключ корпуса
Video_card_id	Int(11)	Передается уникальный ключ видеокарты
CPU_id	Int(11)	Передается уникальный ключ процессора
RAM_id	Int(11)	Передается уникальный ключ оперативной памяти

Таблица 10 – Таблица БД BuildPC (Продолжение)

Power_supply_id	Int(11)	Передается уникальный ключ блока питания
Motherboard_id	Int(11)	Передается уникальный ключ материнской платы
Hard_drive_id	Int(11)	Передается уникальный ключ жесткого диска
Img	Varchar(55)	Путь к директории с изображением корпуса(Используется и для общей сборки)
Description	Text	Описание возможностей сборки для пользователя
Price	Float	Общая средняя цена сборки

Таблица build\_position – предназначена для хранения всех имеющихся сборок с их координатами.

Таблица 11- Таблица БД build\_position

Наименование поля	Тип	Назначение
Id	Int(11)	Уникальный ключ
Build_id	Int(11)	Передается уникальный ключ сборки
x	Int(11)	Координата по оси x
y	Int(11)	Координата по оси Y



## 2.2 Реализация приложения

Для реализации были использованы язык: php[4] ,js[5].

В директории сайта имеется файл “index.php” в котором описана верстка сайта.

По нажатию на одну из клавиш “Отправить” (в зависимости от способа подбора, который выбрал пользователь), данные отправляются методом POST на страницу “result.php”, на которой происходит взаимосвязь с базой данных и обработка данных которые отправил пользователь. Основные строчки кода представлены ниже.

В данной ситуации есть 2 варианта развития событий:

- 1) Пользователь выбрал фирму, которую он предпочитает, и цену, которую считает приемлемой.
- 2) Пользователь ответил на вопросы, и подбор будет происходить исходя из его ответов.

Рассмотрим первый вариант:

```
//      Если была послана форма с Ценой и фирмами.
if ( isset($_POST['firmsAndModels']) ) {
//      Определяем какая серия видеокарты выбрана
$postFirms = $_POST['choose-firms-video'];
if ( $postFirms == 1 ) {
    $seria = 'any';
} else if ( $postFirms == 2 ) {
    $seria = 'NVIDIA';
} else if ( $postFirms == 3 ) {
    $seria = 'AMD';
};
```

После чего те же самые действия проделываем для определения фирмы производителя процессора.

```
$cpuFirm = $_POST['choose-firms-CPU'];  
if ( $cpuFirm == 1 ) {  
    $cpu = 'any';  
} else if ( $cpuFirm == 2 ) {  
    $cpu = 'Intel';  
} else if ( $cpuFirm == 3 ) {  
    $cpu = 'AMD';  
};
```

После чего делается SQL-запрос к базе данных, который находит все видеокарты данной серии. Если же был выбран селектор на значении «Любая», то выбираются все имеющиеся видеокарты.

```
if ( $seria != 'any' ) {  
    $query = $db->query("SELECT `id` FROM `video_card` WHERE  
`seria` = '$seria'");  
} else {  
    $query = $db->query("SELECT `id` FROM `video_card`");  
}
```

Следующий блок кода, заносит данные об первичном ключе выбранной/ых видеокарт в массив. Это сделано для того, чтобы лишний раз не обращаться к базе данных.

```
$i = 0;  
while($row = $query->fetch()) {  
    $videoList[$i] = $row['id'];  
    $i++;  
}
```

Далее происходит подбор по такой же системе, можно выделить только проверку на существование сборки в данном ценовом диапазоне.

Так в следующими строчками кода задается диапазон цены, выбранной пользователем. Где EPS – заранее заданная константа, которая определяет величину интервала между минимальной и максимальной ценой.

```
$price = $_POST['pred_price'];  
$minPrice = $price - EPS;  
$maxPrice = $price + EPS;
```

И уже используя найденный интервал, используем его для нахождения сборки с заданной ценой, делаем это с помощью следующего запроса.

```
$query = $db->query("SELECT * FROM `BuildPC` WHERE  
`Summary` < $maxPrice AND `Summary` > $minPrice;");
```

Так как таблица к которой мы обращались содержит номера ключей комплектующих, необходимо записать интересующие ключи в массив, как было показано ранее.

После произведенных действий, необходимо соотнести все условия и найти такую сборку, которая бы всем им удовлетворяла. Для этого необходимо пройти по всем 3 условиям. Для этого был использован цикл while.

```
// проверка на наличие такой сборки  
$i = 0;  
$build = array();  
while( $i <= count($priceList) - 1 ) {  
    $j = 0;  
    while ($j <= count($cpuList) - 1) {  
  
        $l = 0;  
        while ($l <= count($videoList) - 1) {
```

После чего необходимо записать условие, что данный процессор, установлен в сборку, которая находится в выбранном ценовом диапазоне. Аналогичные условия и для видеокарты.

```
if ( $cpuList[$j] == $priceList[$i]['CPU_id'] and $videoList[$i] == $priceList[$i]['video_card_id'] ) {
```

После чего закрываются все циклы, увеличивается значение переменных, отвечающие за номер шага. И так пока не будут просмотрены все варианты.

Во втором случае код будет выглядеть так, т.е. если была послана форма с вопросами:

```
} else if ( isset($_POST['quest']) ) {
```

Необходимо установить координаты. В данном приложении координатами являются ответы на вопросы, где номер вопросы – это Ось, а номер ответа – числовое значение.

```
$x1 = $_POST['forWhat'];
```

```
$y1 = $_POST['cost'];
```

Вариант просматривать координаты сразу через SQL-запрос, является не самым удобным. Проще собрать все координаты сразу одним запросом:

```
$query = $db->query("SELECT `build_id`,`x`,`y` FROM `build_position`");
```

Запишем их в массив:

```
$i = 0;
```

```
while($row = $query->fetch()) {
```

```
$pos[$i]['build_id'] = $row['build_id'];
```

```
$pos[$i]['x'] = $row['x'];
```

```
$pos[$i]['y'] = $row['y'];
```

```
$i++;
```

```
};
```

И работая с массивом, можно легко найти минимальное значение расстояния между точками. Для этого достаточно сначала присвоить минимальное значение первому элементу:

```
$min = sqrt( pow(($x1-$pos[0]['x']),2) + pow(($y1-$pos[0]['y']),2) );  
$resPosX = $pos[0]['x'];  
$resPosY = $pos[0]['y'];  
$buildId = $pos[0]['build_id'];
```

После чего достаточно в цикле просмотреть все оставшиеся и, в случае необходимости, пере присвоить и само значение, и координаты, которые ему соответствуют. Пример реализации показан следующим циклом.

```
$i = 1;  
while ( $i <= count($pos) - 1) {  
    $length = sqrt( pow(($x1-$pos[$i]['x']),2) + pow(($y1-$pos[$i]['y']),2) );  
    if ( $length < $min) {  
        $min = $length;  
        $resPosX = $pos[$i]['x'];  
        $resPosY = $pos[$i]['y'];  
        $buildId = $pos[$i]['build_id'];  
    }  
    $i++;  
}
```

Далее достаточно просто выбрать из базы данных первичный ключ сборки, который был найден в предыдущем блоке кода. Делает выборку следующий запрос:

```
$query = $db->query("SELECT * FROM `BuildPC` WHERE `id` = '$buildId'");
```

Можно заметить, что для обоих случаев наименование переменных одинаково, это требуется для того, чтобы не вводить лишние условия и проверки, а также не акцентировать внимание на том, какой способ подбора был выбран. Далее происходит вывод информации с базы данных, все делается практическим одинаковыми запросами. Код представлен ниже.

Например, программа ничего не смогла найти по заданным значениям, тогда будет выведено советующее сообщение. Код описан ниже.

```
<?php if (empty($build)) { ?>
```

```
<h3 class="norm-title result center" style="color: #D2515C;" >К  
сожалению по данным требованиям мы ничего не смогли собрать</h3>
```

```
<?php }
```

Если была найдена хоть одна сборка, то начинается целый блок кода, который сначала выдает сообщение:

```
else {
```

```
?>
```

```
<h3 class="norm-title result center">Подобранная сборка ПК</h3>
```

Далее, характеристики каждого комплектующего выводятся на экран. Для этого необходимо исполнить следующий запрос, который выводит характеристики процессора.

```
<?php $cpuQ = $db->query("SELECT * FROM `CPU` WHERE `id` = " .  
$build[$i]['CPU_id'] . " ");
```

```
while($row = $cpuQ->fetch()) { //Расшифровка
```

```
?>
```

Для остальных будут меняться только названия переменных и таблиц.

Также существует отдельный конструктор персонального компьютера, который позволяет изменять конфигурации, если пользователю это необходимо.

За исключением выгрузки всех комплектующих из бахы данных на страницу, все остальные действия осуществляются с помощью языка javascript, а именно в файле «construct.js».

При нажатии на комплектующие, необходимо понять, какого именно оно вида, для этого используется следующий блок кода.

```
var src = $(this).attr('src');
var clas = $(this).attr('class');
clas = clas.split(" ");
if ( jQuery.inArray("cpu",clas) != -1 ) { ... }
```

В данном условном блоке производится присвоение комплектующему активного класса, что необходимо для выделения изображения. Далее идет пример присвоения активного класса и добавление изображения в блок конструктора.

```
$('.cpu').removeClass('active_comp');
$(this).addClass("active_comp");
$('#activePhoto').attr('src',src);
$("#cpuActiveImg").attr("src",src);
```

А также добавляются все необходимые значения в следующий ассоциативный массив

```
var select = {
    cpu : [ "", "", 0, "" ],
    body : [ "", "", 0, "" ],
    hdd : [ "", "", 0, "" ],
    video : [ "", "", 0, "" ],
    op : [ "", "", 0, "" ],
    pow : [ "", "", 0, "" ],
    mat : [ "", "", 0, "" ],
}
```

Данный блок определяет, какое именно комплектующее выбрано, чтобы имелась возможность быстро обратиться к нему, а также к его характеристикам, например к цене.

И в конце выполнения условного блока, идет присвоение всех необходимых значений в ассоциативный массив представленный выше. А само действие имеет следующую реализацию

```
var s = "";
s += "Фирма: " + cpuArr[number].firma + "; ";
s += "Модель: " + cpuArr[number].model + "; ";
s += "Частота: " + cpuArr[number].frequency + "; ";
s += "Кол-во ядер: " + cpuArr[number].yader + "; ";
$('.body_description').text(s);
$('#active_cost').text(cpuArr[number].price + " руб.");
select.cpu[0] = number;
select.cpu[1] = s;
select.cpu[2] = cpuArr[number].price;
```

Примерно так выглядит реализация данного приложения.



### 3 Описание интерфейса пользователя

Разработанное Web-приложение позволяет подобрать сборку ПК по различным критериям различным группам пользователей. Главная страница имеет вид, представленный на Рисунок 4.

На всех страницах у пользователя есть навигация с помощью двух ссылок вверху, а именно «Главная», «О проекте» и «Конструктор».

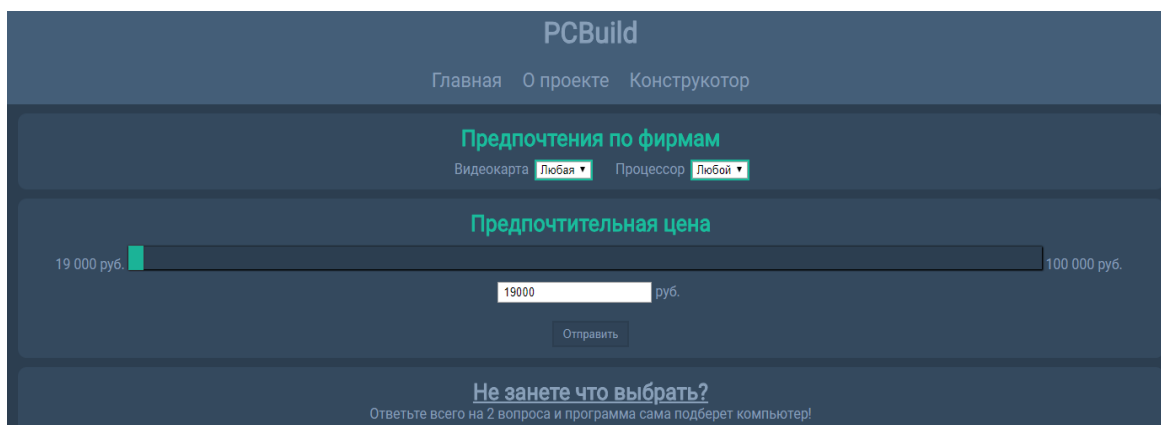


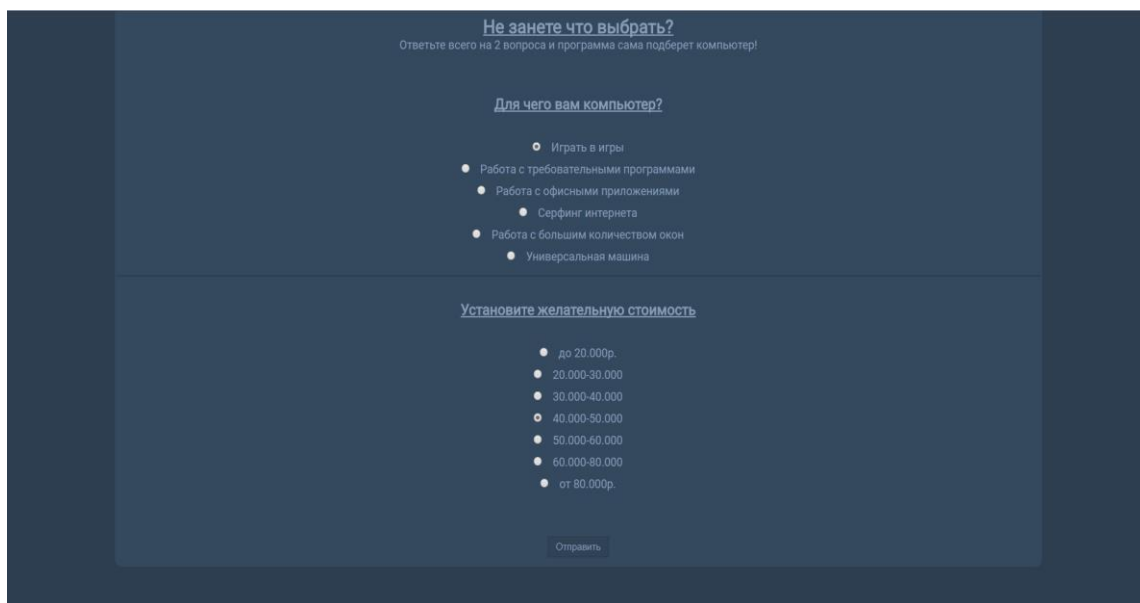
Рисунок 4 – Главная страница Web-приложения

На данной странице пользователю доступно несколько способов, с помощью которых программа будет подбирать самую оптимальную сборку комплектующих для персонального компьютера:

- 1) Самому выбрать интересующие его фирмы процессора и серии видеокарт, для этого присутствует два выпадающих списка, которые находятся в блоке с заголовком «Предпочтения по фирмам», при нажатии на них будет раскрыт весь список фирм или серий, в зависимости от выбранного списка. А также имеется возможность выбрать оптимальную цену, которая изначально равна цене минимальной сборке, которая присутствует в нашей базе данных. Указать цену возможно с помощью ползунка, который находится сразу под заголовком «Предпочтительная цена», где требуется с помощью зажатой левой кнопки мыши, переместить ползунок, в, интересующий

пользователя, ценовой диапазон. Второй вариант, просто вписать приемлемую для посетителя сайта цену в окно для ввода, которое располагается немного ниже ползунка. После того, как пользователь указал всё, что считал необходимым, ему требуется нажать на кнопку «Отправить», находящуюся ниже текстового окна.

2) Второй способ, если пользователь не имеет никаких предпочтений, связанных с фирмой процессора или серии видеокарты. Для этого на сайте предусмотрена опросная система, которая подберет компьютер, в зависимости от того, чего именно хочет пользователь. Для того чтобы перейти к этому способу, посетителю сайта необходимо нажать на блок, в заголовке которого находится надпись: «Не знаете, что выбрать?». После чего раскроется блок с вопросами и вариантами ответа на них. Данный блок представлен на рисунке 5.



The image shows a dark-themed web form titled "Не знаете что выбрать?" (Don't know what to choose?). Below the title is a subtitle: "Ответьте всего на 2 вопроса и программа сама подберет компьютер!" (Answer just 2 questions and the program will choose the computer for you!). The first question is "Для чего вам компьютер?" (What do you need the computer for?). It has five radio button options: "Играть в игры" (Play games), "Работа с требовательными программами" (Work with demanding programs), "Работа с офисными приложениями" (Work with office applications), "Серфинг интернета" (Internet surfing), and "Работа с большим количеством окон" (Work with a large number of windows). The second question is "Установите желательную стоимость" (Set the desired price). It has six radio button options representing price ranges in Russian rubles: "до 20.000р.", "20.000-30.000", "30.000-40.000", "40.000-50.000", "50.000-60.000", "60.000-80.000", and "от 80.000р.". At the bottom of the form is a button labeled "Отправить" (Send).

Рисунок 5 – Блок с вопросами и вариантами ответов на них

В блоке имеется 2 вопроса, с разными вариантами ответов. После выбора ответов на вопросы необходимо нажать на кнопку «Отправить», которая располагается ниже.

После нажатия на эту кнопку, вне зависимости от выбора способа подбора, пользователь попадает на новую страницу, вид которой зависит от того, смогла ли программа найти подходящие комплектующие. К примеру, если этого не произошло, то страницу будет выглядеть, как представлено на рисунке 6 и сообщает пользователю о том, что по его запросу нет сведений.

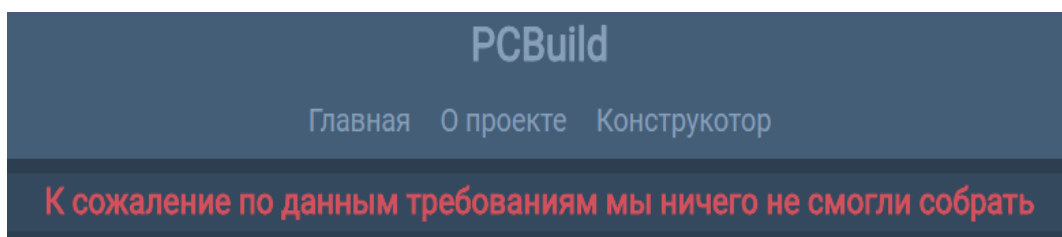


Рисунок 6 – Сборок не найдено

Второй вариант того, как может выглядеть страница, представлен на рисунке 7 и рисунке 8. На рисунках показано, что в базе данных нашлась одна сборка, и комплектация этой сборки расписана на странице.

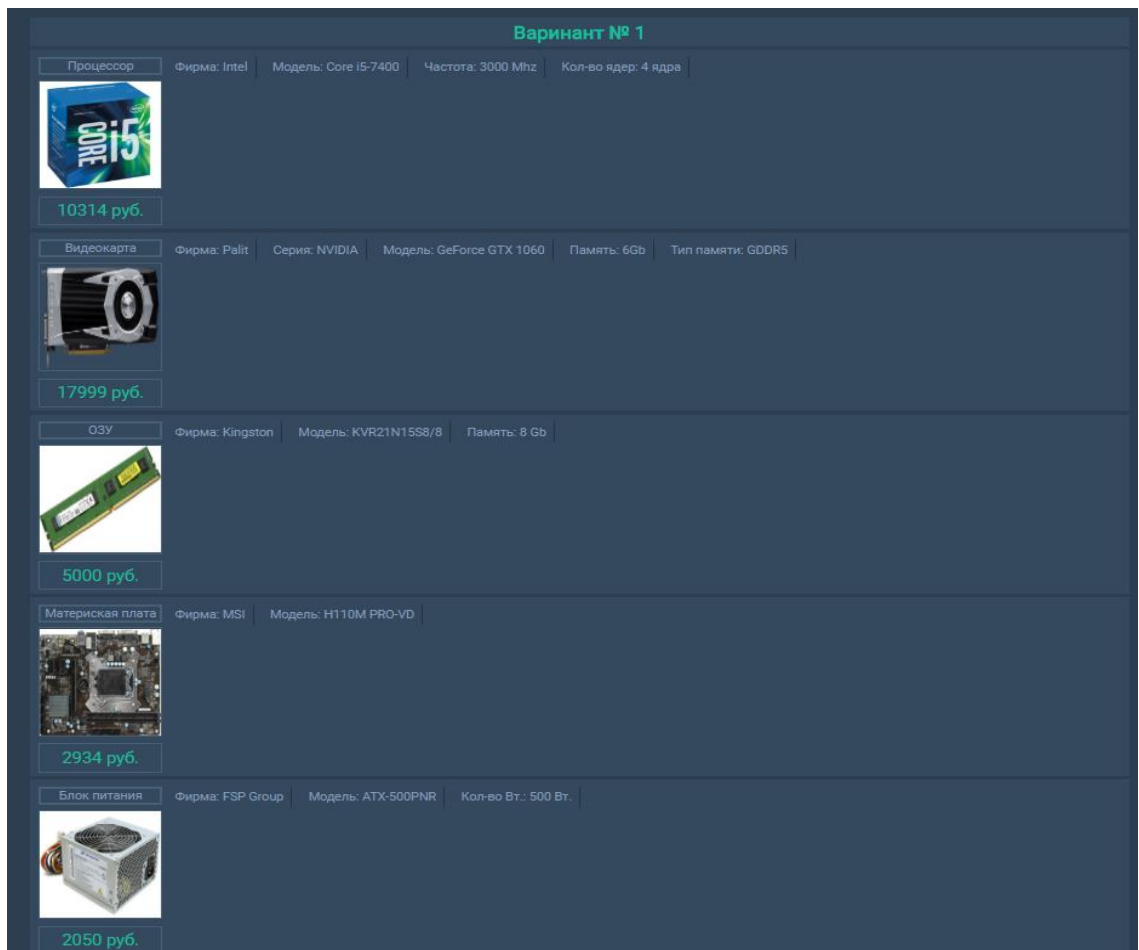


Рисунок 7 – Найдена одна сборка

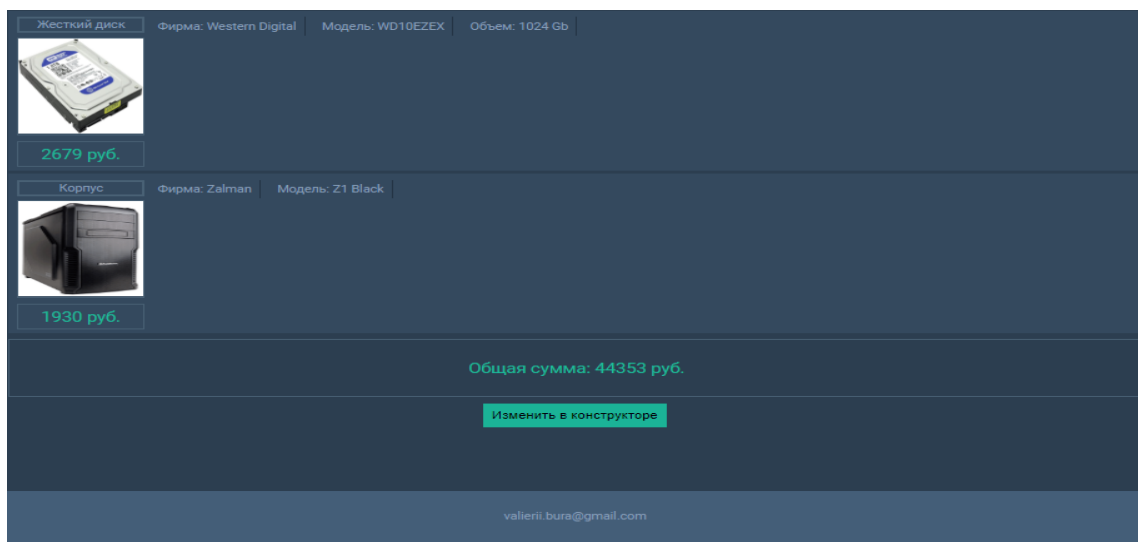


Рисунок 8 – Найдена одна сборка (продолжение)

На рисунке 9 представлен вариант результатов поиска, когда найдено несколько сборок. Сборки перечисляются последовательно с полным

описанием всех комплектующих, с их картинками, характеристиками и ценой. В конце сборки пишется общая сумма всех комплектующих.

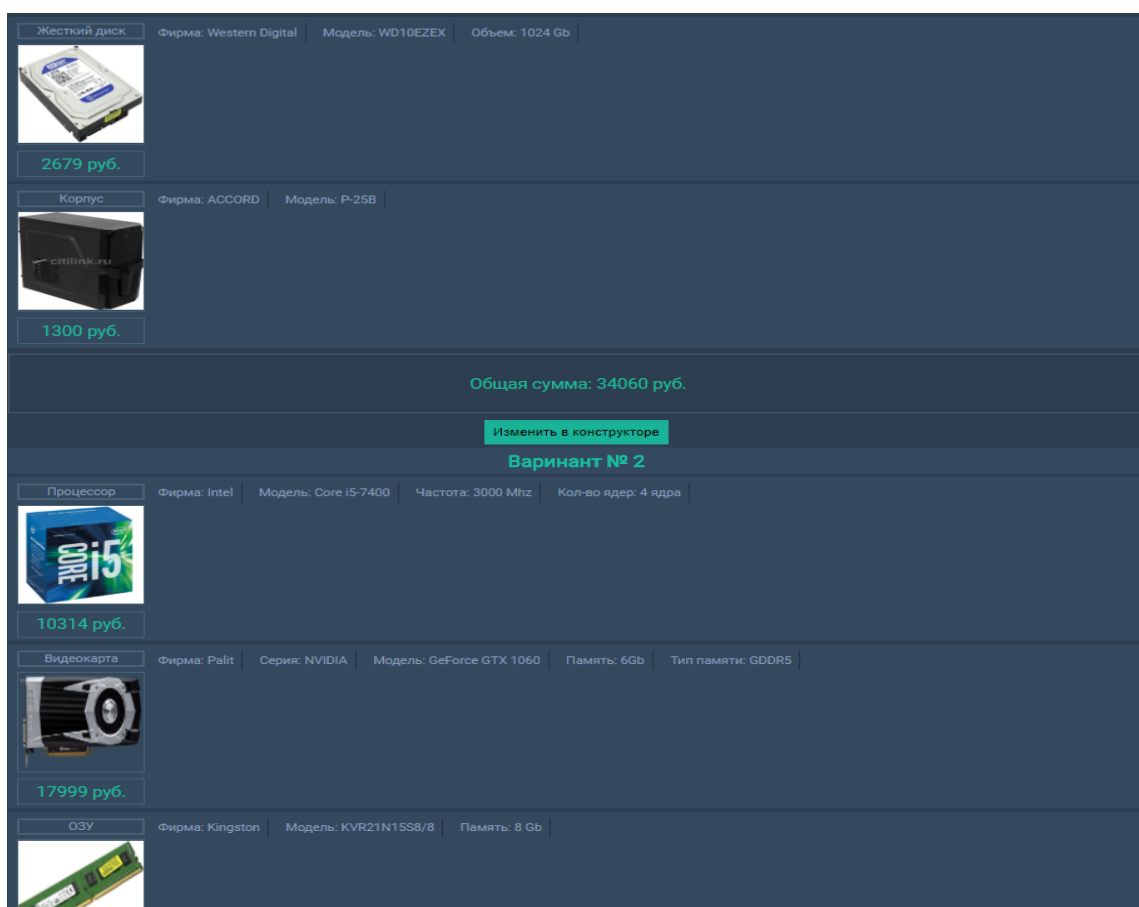


Рисунок 9 – Алгоритм подобрал несколько сборок

Чтобы перейти на окно конструктора, необходимо нажать на советующую ссылку в верхнем меню – «Конструктор». Откроется страница, представленная на рисунке 10.

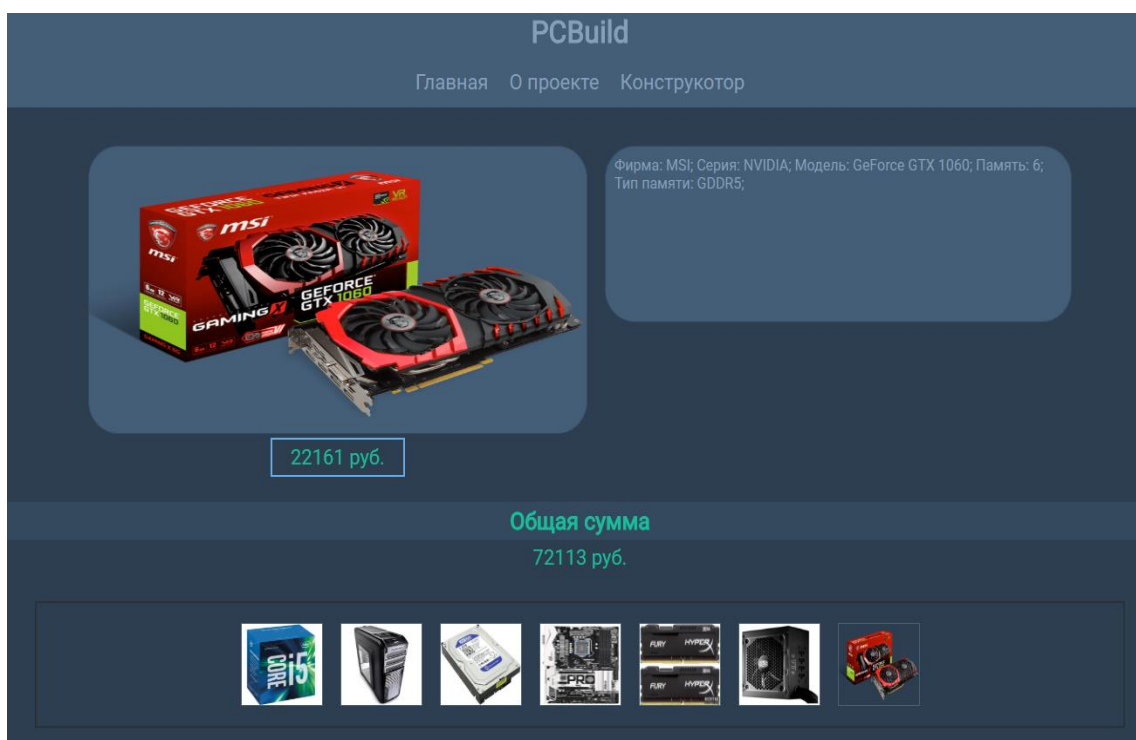


Рисунок 10 – Конструктор

Под верхним меню располагается блок активного, т.е. выбранного в данный момент, комплектующего. Слева находится его изображение, а справа описание характеристик. Ниже изображения располагается цена активного товара.

Ниже и по центру под надписью «общая сумма» располагается суммарное значение цен, всех выбранных конфигураций. Ниже изображения всех выбранных комплектующих, пользователь может переключаться между ними нажатием по изображению левой кнопки мыши, для просмотра их характеристик.

Ниже на странице представлены все имеющиеся комплектующие, на которые он может заменить выбранные до этого. Все товары в сборке подсвечиваются зеленым светом, для наглядности.



Рисунок 11 – Выбор комплектующего

В случае наведения мышкой на интересующий пользователя товар, высвечивается подсказка с описанием комплектующего.

Так же может  
Процессор

Процессор подойдет для средних программ. Работа с Photoshop 2007, программы видео монтажа того же года. Использование браузера с несколькими вкладками. Поддержки средне-требуемых игр




Рисунок 12 – Описание товара



## ЗАКЛЮЧЕНИЕ

В процессе выполнения курсовой работы были изучены этапы создания web-приложений, получены навыки работы с языком разметки HTML и каскадной таблицей стилей CSS, с помощью которых были созданы каркасы страниц. Так же были выработаны навыки работы с СУБД MySQL, создание таблиц, первичных ключей. Был изучен язык программирования PHP, и язык SQL-запросов.

На основе полученных знаний было создано web-приложение, которое позволяет пользователю подобрать сборку персонального компьютера двумя способами, либо по его предпочтениям касательно фирм производителей процессора и видеокарт, либо сборка будет подобрана по результатам ответов на представленные вопросы. При втором варианте все сборки, представляются в виде объектов в  $n$ -мерном пространстве, где  $n$  – это количество вопросов. И самая близкая точка к той что «создал» пользователь – является сборка, наиболее подходящая под его ответы.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 DNS-Shop – интернет-магазин цифровой и бытовой техники. URL: <https://www.dns-shop.ru/configurator> (Дата обращения 15.05.2018)
- 2 Hyperpc.ru – Конфигуратор. URL: [https://hyperpc.ru/gaming-  
pc/configurator](https://hyperpc.ru/gaming-pc/configurator) (Дата обращения 15.05.2018)
- 3 PHPClub - клуб разработчиков PHP. URL: <https://phpclub.ru/mysql/doc/>  
(Дата обращения 12.12.2017)
- 4 PHP – Manual Руководство. URL: <http://php.net/manual/ru/> (Дата обращения: 20.05.2018)
- 5 JavaScript | MDN Руководство по языку JavaScript. URL: <https://developer.mozilla.org/ru/docs/Web/JavaScript> (Дата обращения 20.05.2018)