

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Алтайский государственный университет»  
Факультет математики и информационных технологий  
Кафедра теоретической кибернетики и прикладной математики

РАЗРАБОТКА АИС СОПРОВОЖДЕНИЯ ПРОЦЕССОВ  
ТЕХНИЧЕСКОГО ОБСЛУЖИВАНИЯ  
Магистерская диссертация

Выполнил:  
студент 458М-ПИ группы  
Приколота Евгений  
Владимирович

---

Научный руководитель:  
д.т.н., профессор  
Оскорбин Николай  
Михайлович

---

Допустить к защите:  
зав. кафедрой: к.т.н.,  
доцент Л.А. Хворова

« \_\_\_ » \_\_\_\_\_ 2017 г.

---

Работа защищена:  
« \_\_\_ » \_\_\_\_\_ 2017 г.  
Оценка: \_\_\_\_\_

Председатель ГЭК:

---

Барнаул – 2017

## Введение

На сегодняшний день автоматизация деятельности становится неотъемлемой частью практически любого предприятия. Управление различными процессами при помощи компьютера позволяет добиться более высокой производительности труда и повысить эффективность использования рабочего времени. Высококачественная автоматизация технологических процессов значительно облегчает работу предприятия и производства в целом.

Данная АИС была создана для упрощения отношений «заказчик - исполнитель». В информационной системе будет отображаться вся информация, ранее доступная только человеку, принимающему в ремонт оборудование.

В рамках данного проекта на сайте сервисного центра создается специальный раздел для клиента «Личный кабинет», который будет особо актуален для оптовых клиентов и клиентов из районов края. Зная только номер заказа, клиент сможет узнать полную информацию о переданном в ремонт оборудовании, а так же всю историю заказов, выполненных для него ранее.

Для корректного отображения всей информации были применены стили и классы, которые использовались на сайте, в структуре которого проектируется личный кабинет.

Элементы АИС были написаны на платформе PhpStorm, на языке программирования PHP. PhpStorm представляет собой интеллектуальный редактор для PHP, HTML и JavaScript с возможностями анализа кода на лету, предотвращения ошибок в коде и автоматизированными средствами рефакторинга для PHP и JavaScript. Для написания структуры веб-приложения использовался фреймворк symfony2. Он предлагает быструю разработку и управление веб-приложениями, позволяет легко решать рутинные задачи веб-программиста, а так же содержит некоторые готовые классы, для облегчения работы с базой данных.

Целью работы является разработка web-интерфейса отображения статуса заказа и процесса его обслуживания.

Объект исследования – процесс приема на ремонт цифровой техники и состояние ее обслуживания.

Предмет исследования – создание web-приложения, позволяющего клиенту самостоятельно проверять статус заказа и процесс его выполнения.

Система спроектирована с использованием технологий баз данных MySQL и таблиц стилей CSS. Алгоритмы решения задач написаны на языке программирования PHP и JavaScript. Выбор средств программирования обусловлен их низкой стоимостью, достаточными техническими возможностями и высоким уровнем освоенности специалистами.

Объем работы 44 страницы, количество рисунков – 18, использованных источников литературы – 14.

# 1. ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ ПРОЕКТИРОВАНИЯ АИС С ИСПОЛЬЗОВАНИЕМ WEB-ТЕХНОЛОГИЙ

## **1.1. Основы проектирования информационной системы.**

Информационной системой называют взаимосвязанную совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации для достижения поставленной цели.

По сфере применения информационные системы делятся на технические, экономические, гуманитарные и другие. Они обеспечивают сбор, хранение, поиск, обработку выдачу информации, необходимой в процессе принятия решений задач из любой области.

Современное понимание информационной системы предполагает использование в качестве основного средства переработки информации компьютера. Кроме того, особая роль отводится человеку, для которого предназначена информация и без которого невозможно ее представление и получение.

Необходимо понимать разницу между компьютерами и информационными системами. Компьютеры являются технической базой и инструментом для информационных систем. Информационная система немыслима без персонала, взаимодействующего с компьютерами и телекоммуникациями.

## **1.2. Этапы создания и развития информационной системы**

Вопросы проектирования, разработки и внедрения автоматизированных информационных систем (АИС) рассмотрены, например, в следующей литературе [1-12]. Этапы развития практики разработки и внедрения АИС представлены в таблице 1.

Таблица 1. Этапы развития информационных систем

Период времени	Концепция использования информации	Вид информационных систем	Цель использования
1950-1960 гг.	Бумажный поток расчетных документов	Электромеханические бухгалтерские машины	Упрощение процедуры обработки счетов и расчетов зарплаты
1960-1970 гг.	Помощь в подготовке отчетов	Управленческие информационные системы для производственной информации	Ускорение процесса подготовки отчетности
1970-1980 гг.	Управленческий контроль процессов	Системы поддержки принятия решений	Выработка рациональных решений
С 1980 года по н/в	Информация - стратегический ресурс, обеспечивающий конкурентное преимущество	Стратегические информационные системы. автоматизированные офисы	Выживание и процветание организаций

Развития информационных систем можно рассматривать с позиции развития самой техники и с точки зрения совершенствования автоматизированных информационных систем.

Первый аспект состоит из двух этапов:

- 1) до появления ЭВМ;
- 2) с развитием ЭВМ.

Рассмотрим второй аспект. Главной тенденцией развития автоматизированных информационных систем является постоянное совершенствование технических и программных средств.

Поколение автоматизированных информационных систем делится на:

- первое поколение (1960-1970 гг.). Оно строилось на базе вычислительных центров;

- второе поколение (1970-198 гг.). В этот период информационные технологии проникают в отделы, службы предприятия. Появились пакеты и децентрализованные базы данных, начали внедряться двух, трехуровневые модели организации систем обработки данных;

- третье поколение (1980 - начало 1990 гг.). Для него характерен массовый переход к распределенной сетевой обработке на базе персональных компьютеров с объединением разрозненных рабочих мест в единую информационную систему;

- четвертое поколение. Здесь наблюдается тенденция к возврату на крупных и средних предприятиях к использованию в информационных системах электронно-вычислительных машин в качестве центрального узла системы и дешевых сетевых терминалов.

### **1.3. Организация внедрения и функционирования автоматизированных информационных систем**

Процессы, обеспечивающие работу информационной системы, позволяют выполнять функции автоматизации рутинных операций. Приведем перечень основных процессов автоматизированных информационных систем:

- 1) Ввод информации из внешних и внутренних источников.
- 2) Обработка входной информации, представления ее в удобном виде.
- 3) Вывод информации для предоставления потребителям или передачи в другую систему.
- 4) Обратная связь - это информация, переработанная людьми данной организации для коррекции входной информации.

Перечислим основные свойства современных информационных систем:

- информационная система является динамичной и развивающейся;
- информационная система может быть подвергнута анализу, построена и управляема на основе общих принципов построения систем;
- при построении информационной системы необходимо использовать системный подход;
- информационную систему следует воспринимать как человеко-компьютерную систему обработки информации;
- выходной продукцией информационной системы является информация, на основе которой принимаются решения.

Информационные системы в настоящее время разрабатываются и внедряются с целью достижения следующих результатов:

- получить более рациональный вариант решения задач;
- освободить работников от рутинной работы за счет ее автоматизации;
- обеспечить достоверность информации;
- совершенствовать структуру потоков информации и системы документооборота в фирме;
- уменьшить затраты на производство продуктов и услуг.

## 2. ПРОЕКТИРОВАНИЕ И СОЗДАНИЕ WEB-ПРИЛОЖЕНИЯ ПРИ РЕМОНТЕ ЦИФРОВОЙ ТЕХНИКИ

### 2.1. Обоснование функционала и структуры информационной системы

Структуру информационной системы составляет совокупность отдельных ее частей, называемых подсистемами.

Подсистема - это часть системы, выделенная по какому-либо признаку.

Общую структуру информационной системы можно рассматривать как совокупность подсистем независимо от сферы применения. В этом случае говорят о структурном признаке классификации, а подсистемы называют обеспечивающими. Структура любой информационной системы может быть представлена совокупностью обеспечивающих подсистем (рис. 1).



Рис. 1. Структура информационной системы, как совокупность обеспечивающих подсистем

Назначение подсистемы состоит в своевременном формировании и выдаче достоверной информации для принятия решений.

Информационное обеспечение - совокупность единой системы классификации и кодирования информации, унифицированных систем документации, схем информационных потоков, циркулирующих в организации, а также методология построения баз данных.



Унифицированные системы документации создаются на государственном, республиканском, отраслевом и региональном уровнях. Главная цель - обеспечение сопоставимости показателей различных сфер общественного производства.

Но при исследовании организаций выделяется целый комплекс типичных недостатков:

- чрезвычайно большой объем документов для ручной обработки;
- одни и те же показатели часто дублируются в разных документах;
- имеются показатели, которые создаются, но не используются;
- работа с большим количеством документов отвлекает специалистов от решения непосредственных задач, и др.

Устранение указанных недостатков является одной из задач, стоящих при создании информационного обеспечения.

Схемы информационных потоков отражают маршруты информации и ее объемы, места возникновения первичной информации и использования резульатной информации. За счет анализа структуры подобных схем можно выработать меры по совершенствованию всей системы управления.

Построение схем информационных потоков обеспечивает:

- исключение неиспользуемой и дублирующей информации;
- классификацию и рациональное представление информации.

Методология построения баз данных базируется на теоретических основах их проектирования, состоящих из двух этапов:

1-й этап - обследование всех функциональных подразделений с целью: понять специфику и структуру его деятельности, построить схему информационных потоков, проанализировать существующую систему документооборота, определить информационные объекты и соответствующий состав реквизитов, описывающих их свойства и назначение;

2-й этап - построение концептуальной информационно-логической модели данных для обследованной на 1-м этапе сферы деятельности. В этой

модели должны быть установлены и оптимизированы все связи между объектами и их реквизитами. Информационно-логическая модель является фундаментом, на котором будет создана база данных.

Техническое обеспечение - это комплекс технических средств, предназначенных для работы информационной системы, а также соответствующая документация на эти средства и технические процессы.

Комплекс технических средств составляют: компьютеры любых моделей, устройства сбора, накопления, обработки, передачи и вывода информации, передачи данных и линий связи, оргтехника и устройства автоматического съема информации и др.

Математическое и программное обеспечение - совокупность математических методов, моделей, алгоритмов и программ для реализации целей и задач информационной системы, а также нормального функционирования комплекса технических средств.

Средства математического обеспечения делятся на:

- средства моделирования процессов управления;
- типовые задачи управления;
- методы математического программирования, математической статистики, теории массового обслуживания и др.

Программное обеспечение состоит из общесистемных и специальных программных продуктов и технической документации.

Организационное обеспечение - совокупность методов и средств, регламентирующих взаимодействие работников с техническими средствами и между собой в процессе разработки и эксплуатации информационной системы. Функции организационного обеспечения:

- анализ существующей системы управления организацией и выявление задач, подлежащих автоматизации;

- подготовка задач к решению на компьютере, включая техническое задание на проектирование информационной системы и технико-экономическое обоснование ее эффективности;

- разработка решений по составу и структуре организаций, методологии решения задач, направленных на повышение эффективности системы управления.

Правовое обеспечение - совокупность правовых норм, определяющих создание, юридический статус и функционирование информационных систем, регламентирующих порядок получения, преобразования и использования информации. Его главная цель - укрепление законности.

Правовое обеспечение этапов функционирования информационной системы включает:

- статус информационной системы;
- права, обязанности и ответственность персонала;
- порядок создания и использования информации и др.

## **2.2. Выбор типовых решений при создании информационной системы по признаку структурирования задач и по степени автоматизации.**

При создании или при классификации информационных систем возникают проблемы, связанные с математическим и алгоритмическим описанием решаемых задач.

Различают три типа задач, для которых создаются информационные системы: структурированные (формализуемые), неструктурированные (неформализуемые), частично структурированные.

Структурированная задача - это задача, где известны все ее элементы и взаимосвязи между ними.

В данной задаче удается выразить ее содержание в форме математической модели, имеющей точный алгоритм решения. Эти задачи обычно приходится решать многократно. Целью использования информационной систе-

мы для решения структурированных задач является полная автоматизация их решения.

Неструктурированная задача - это задача, в которой невозможно выделить элементы и установить между ними связь.

Решение неструктурированных задач связано с большими трудностями из-за невозможности создания математического описания и разработки алгоритма. Решение в таких случаях принимается человеком на основе опыта и косвенной информации из различных источников.

Неструктурированных или полностью структурированных задач на практике существует немного. В большинстве задач часть элементов и их взаимосвязей известна. Такие задачи называются частично структурированными. Для таких задач можно создать информационную систему, получаемая в ней информация анализируется человеком. Такие информационные системы называются автоматизированными.

Пути решения частично структурированных задач:

- выделение дополнительного финансирования на увеличение численности работающих;
- отнесение срока окончания работы на более позднюю дату и др.

Информационные системы, используемые для решения частично структурированных задач, делятся на два вида (рис. 2):

- создающие управленческие отчеты (ориентированы на обработку данных и создание управленческих отчетов);
- разрабатывающие альтернативы решений.

Информационные системы, создающие управленческие отчеты, обеспечивают информационную поддержку пользователя.

Информационные системы, разрабатывающие альтернативы решений, могут быть модельными и экспертными. Модельные системы предоставляют пользователю математические, статические, финансовые и другие модели, использование которых облегчает выработку и оценку альтернатив решения.

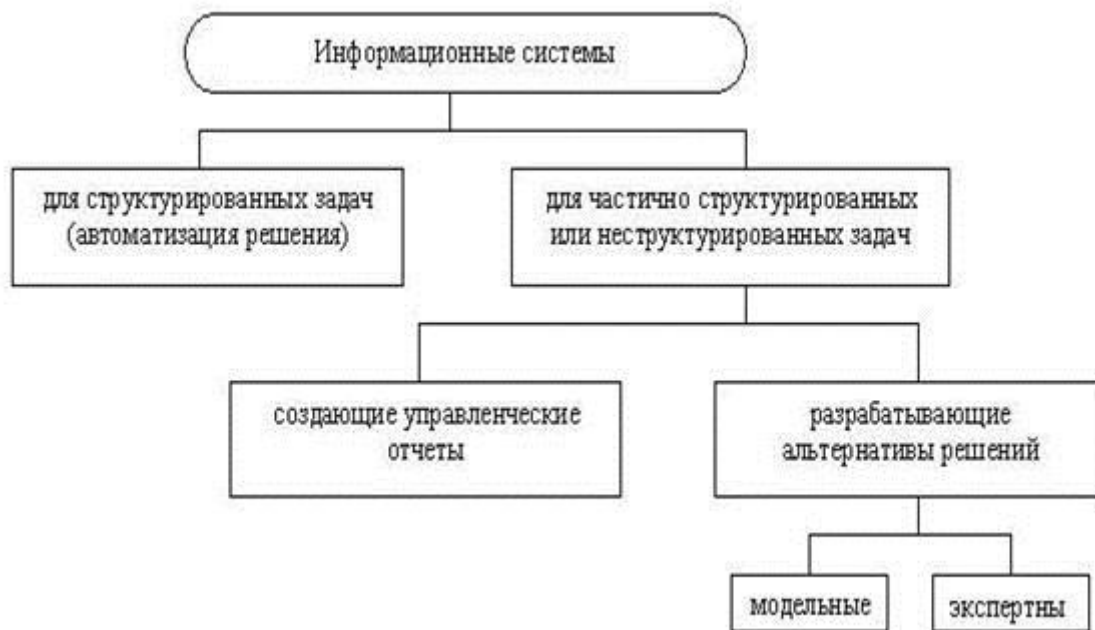


Рис. 2. Классификация информационных систем по признаку структурированности задач

Основными функциями этой системы являются:

- возможность работы в среде типовых математических моделей;
- достаточно быстрая и адекватная интерпретация результатов моделирования;
- оперативная подготовка и корректировка входных параметров и ограничений модели;
- возможность графического отображения динамики моделей;
- возможность объяснения пользователю необходимых шагов формирования и работы модели.

Экспертные информационные системы обеспечивают выработку и оценку возможных альтернатив пользователем за счет создания экспертных систем, связанных с обработкой знаний.

Ручные информационные системы характеризуются отсутствием современных технических средств обработки информации и выполнением всех операций человеком по заранее разработанным методикам (рис. 3).

Автоматизированные информационные системы - это человеко-машинные системы, обеспечивающие автоматизированный сбор, обработку и передачу информации, необходимой для принятия управленческих решений в организациях различного типа.



Рис.3. Классификация информационной системы по степени автоматизации

Автоматические информационные системы характеризуются выполнением всех операций по обработке информации автоматически, без участия человека, но оставляют за человеком контрольные функции.

Автоматизированные информационные системы имеют различные модификации и могут быть классифицированы по характеру использования информации и по сфере применения.

### 2.3. Техническая реализация элементов web-приложения при ремонте цифровой техники в условиях конкретной организации

В данном разделе рассматривается техническая реализация сайта в условиях конкретного предприятия. В начале реализации проектом было принято решение зарегистрировать хостинг, скопировав туда полноценно функционирующий сайт сервисного центра. Весь сайт сделан по принципу MVC (modelviewcontroller), это схема разделения данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер — таким образом, что модификация каждого компонента может осуществляться независимо. Перечислим составляющие этого разделения:

**Модель (Model)** предоставляет данные и реагирует на команды контроллера, изменяя свое состояние.

**Представление (View)** отвечает за отображение данных модели пользователю, реагируя на изменения модели.

**Контроллер (Controller)** интерпретирует действия пользователя, оповещая модель о необходимости изменений.

В итоге у нас получилось 3 папки, представленных на рисунке 4.



Name	Date modified	Type	Size
app	08.06.2017 22:24	File folder	
src	08.06.2017 22:23	File folder	
web	08.06.2017 22:26	File folder	

Рис. 4. Схема разделения данных приложения, пользовательского интерфейса и управляющей логики

В первой папке у нас лежит все необходимое для Symfony. Во второй весь код. В третьей папке — стили, картинки и т.п.

Как уже говорилось ранее, что используется MVC и в третьей папке есть подкаталог CSS, тут и хранятся исходные данные (рис. 5).

Controller	08.06.2017 22:28	File folder	
Entity	08.06.2017 22:23	File folder	
Resources	08.06.2017 22:23	File folder	
AppBundle.php	27.03.2016 17:24	JetBrains PhpStorm	1 KB

Рис. 5. Исходные данные подкаталога css папки 3

Для нашего проекта не будем подгружать реальную базу данных, а создадим свою – проверочную базу с идентичными полями, но с гораздо меньшим количеством записей. На рисунке бпоказан фрагмент кода, который создает нужные нам поля в базе данных. В языке Symfony есть свой способ задания названия поля. Просто вводишь в командной строке команду и база создается сама исходя из того, что прописали в модели.

```

private $id;

/**
 * @var string
 *
 * @ORM\Column(name="device", type="string", length=255)
 * @JMS\Expose
 */
private $device;

/**
 * @var string
 *
 * @ORM\Column(name="number", type="string", length=255)
 * @JMS\Expose
 */
private $number;

/**
 * @var string
 *
 * @ORM\Column(name="serial", type="string", length=255)
 * @JMS\Expose
 */
private $serial;

/**
 * @var string
 *

```



Рис. 6. Фрагмент кода программы создания полей в базе данных  
Рассмотрим более подробно приведенное ниже описание (рис. 7):

```
*  
* @ORM\Column(name="number", type="string", length=255)  
* @JMS\Expose
```

Рис. 7. Анализ фрагмента кода программы создания полей

В данном фрагменте кода описано как в базе данных будет колонка называться, то что она будет являться строкой, длина строки, то есть как происходит весь процесс.

Ссылка на нужную нам страницу будет генерироваться с помощью контроллера. Запросы обрабатываются в файле defaultController, и он генерирует нужную нам страницу, но при этом сами страницы лежат в папке resources, то есть контроллер получил запрос, и вернул страницу. Исходя из того, что в строке было он возвращает нужную страницу в адресной строке.

Вот эти страницы, сам внешний вид (рис. 8):

site > src > AppBundle > Resources > views > default

Name	Date modified	Type
history.html.php	08.06.2017 22:15	JetBra
info.html.php	08.06.2017 22:04	JetBra
test1.html.php	08.06.2017 19:47	JetBra
test2.html.php	08.06.2017 21:13	JetBra

Рис. 8. Коды формирования страницы в адресной строке

Напишем функцию, отвечающую за историю (рис. 9):

```
/**  
 * @Route("/history/id", name="get_history")  
 * @Method({"GET", "POST"})  
 * @return Response  
 */  
public function getHistory($id)  
{  
    $order = $this->getDoctrine()->getManager()->getRepository('Order')->findOneBy([  
        'historyId' => $id  
    ]);  
    return $this->render('AppBundle::default/history.html.php', [  
        'order' => $order  
    ]);  
}
```

Рис. 9. Коды формирования истории

В этой функции считывается id, поэтому id достается из базы данных нужная запись, и уже все записи складываются и на сайте в личном кабинете отображается вся история выбранного пользователя.

Для каждой странички есть метод в контроллере который отдает страницу саму, но иногда отдает страницу передавая в нее данные, которые взяты из базы данных. Контроллеру нужно принять запрос из адресной строки браузера, в котором прописаны все маршруты.

Для изменения главной страницы сайта добавим следующие элементы (рис. 10):

```
199 <div class="header-block">
200 <div class="header-block_icon-wrapper"><em class="fa fa-sign-in"> </em></div>
201 <span class="header-block_title"><a href="http://api.e-accident.eu/sign-in">Личный
кабинет</a></span>
202 <p class="header-block_small-text"></p>
203 </div>
204 <!-- MODULE Block cart -->
```

Рис. 10. Коды изменения главной страницы

Этот скрипт добавляет иконку личного кабинета на домашний экран, а при нажатии на кнопку он выполняется (рис. 11):

```
475
476 <input class="inputNew form-control grey newsletter-input" id="numberZakaz" type="text" size="18"
value="Введите номер заказа" onblur="if (this.value == '') {this.value = 'Введите номер заказа'}"
477 onfocus="if (this.value == 'Введите номер заказа') {this.value = ''}" />
478 <button id="zakazButton" class="btn btn-default button button-small">
479 <span>OK</span>
480 </button>
481 <script>
482 $('#zakazButton').on("click", "button", function() {
483 console.log("asd");
484 var number = $('#numberZakaz').html();
485 window.location.href = "http://api.e-accident.eu/information/" + number;
486 })
487 </script>
488 </div>
489
```

Рис. 11. Формирование иконки личного кабинета на домашний экран

Далее создается скрипт с историей заказов (рис. 12):

```
475
476 <h4>Заказ номер: <?php echo $order->getNumber() ?></h4>
477 <hr>
478 <?php echo $order->getGeneral() ?>
479
480 <table>
481 <tr>
482 <td>Номер заказа</td>
483 <td>Дата приема</td>
484 <td>Оборудование</td>
485 <td>Статус заказа</td>
486 </tr>
487 </table>
488
489 <table>
490 <?php foreach ($order as $o){?>
491 <tr>
492 <td><?php echo $o->getNumber() ?></td>
493 <td><?php echo $o->getDate() ?></td>
494 <td><?php echo $o->getDevice() ?></td>
495 <td><?php echo $o->getStatus() ?></td>
496 </tr>
497 <?php } ?>
498 </table>
499 </div>
```

Рис. 12. Формирование истории заказа

Скрипт с информацией о заказе (рис. 13):

```
475
476 <h4>Заказ номер: <?php echo $order->getNumber() ?></h4>
477 <hr>
478 <?php echo $order->getGeneral() ?>
479 <table>
480 <tr>
481 <td>Наименование, характеристика, артикул материалов заказчика</td>
482 </tr>
483 <tr>
484 <td>Оборудование: <?php echo $order->getDevice() ?></td>
485 </tr>
486 <tr>
487 <td>Серийный номер: <?php echo $order->getSerial() ?></td>
488 </tr>
489 <tr>
490 <td>Оборудование: <?php echo $order->getDate() ?></td>
491 </tr>
492 <tr>
493 <td>Неисправность: <?php echo $order->getDefect() ?></td>
494 </tr>
495 <tr>
496 <td>Состояние заказа: <?php echo $order->getStatus() ?></td>
497 </tr>
498 <tr>
499 <td>Стоимость ремонта: <?php echo $order->getPrice() ?></td>
500 </tr>
501 <tr>
502 <td>Оборудование: <?php echo $order->getComment() ?></td>
503 </tr>
504 </table>
505 </div>
```

Рис. 13. Формирование информации по истории заказа

Начальная страница сайта сервисного центра приведена на рисунке 14:

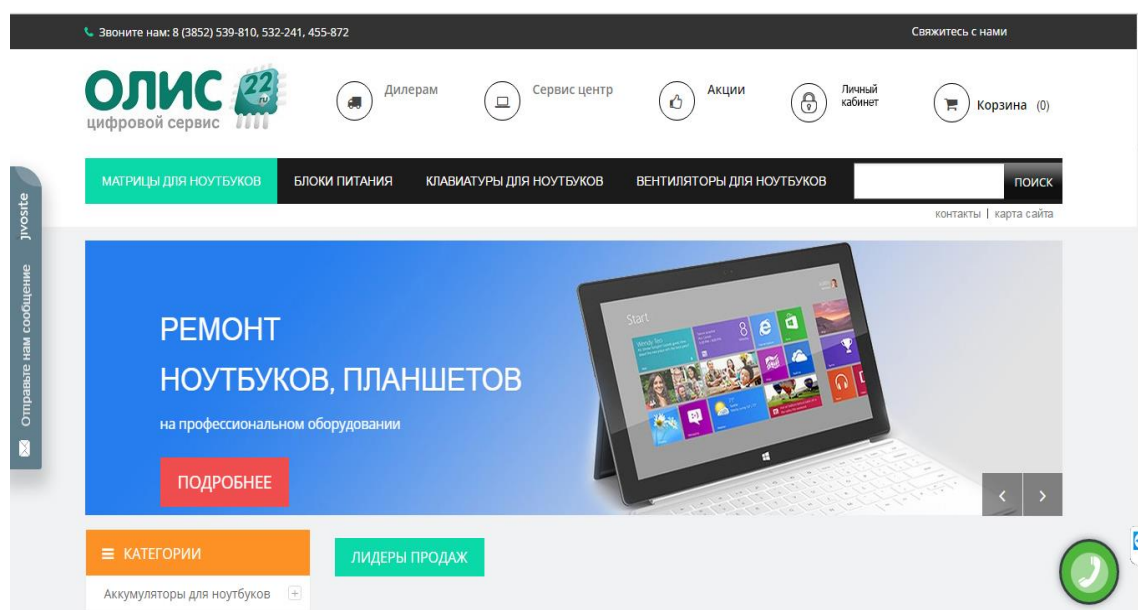


Рис. 14. Главная страница сайта сервисного центра

Начальная страница личный кабинет приведена на рисунке 15:

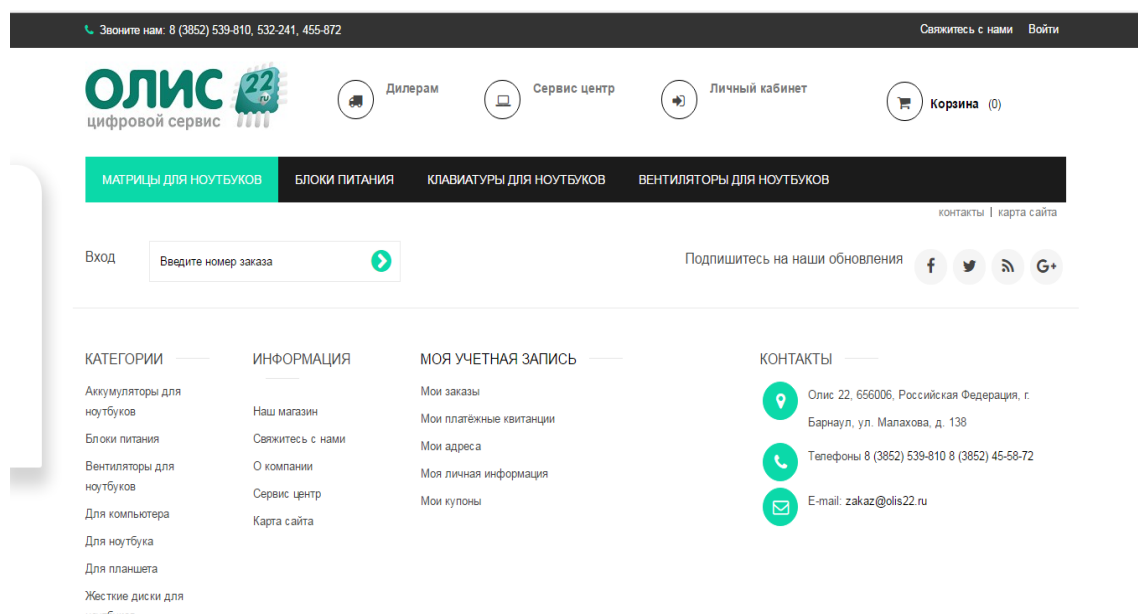


Рис. 15. Вход пользователя в личный кабинет

Страница информации о текущем заказе пользователя (рис. 16):

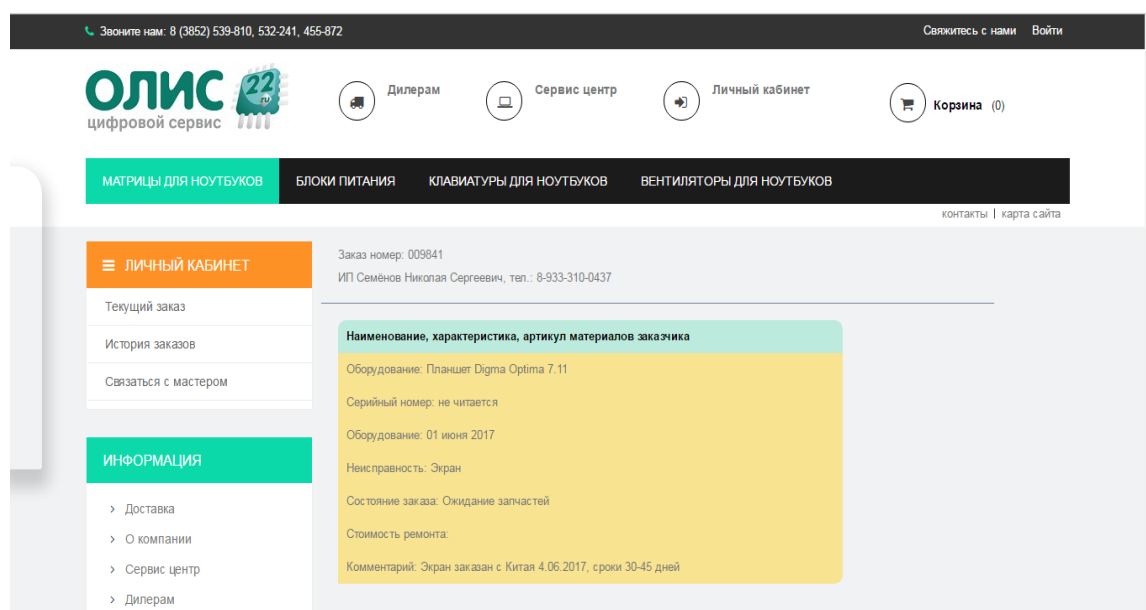


Рис. 16. Оформление и информация по текущему заказу

История заказов пользователя в личном кабинете представлена на рисунке 17.

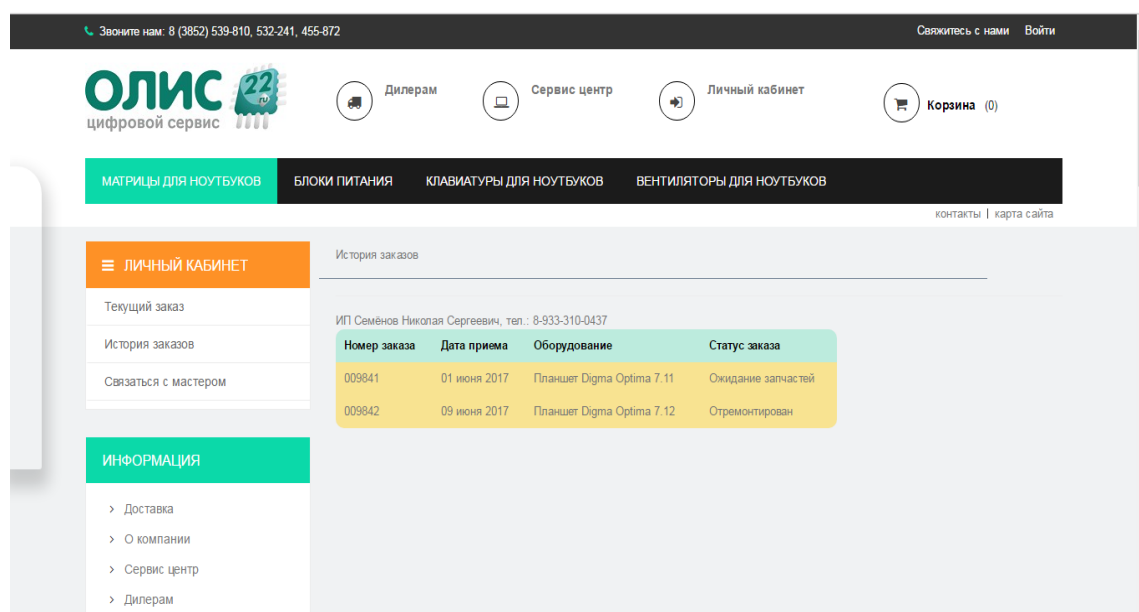


Рис. 17. Страница сайта «История заказов пользователя»

Возможность и технические решения поддержки связи с мастером сервисного центра представлены на рисунке 18.

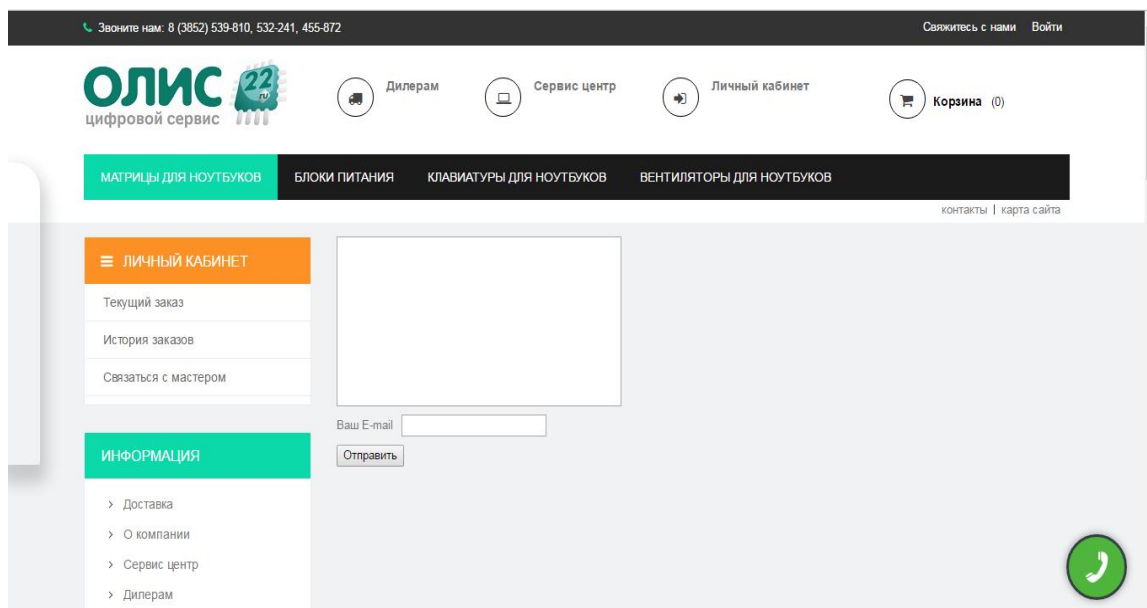


Рис. 18. Страница сайта «Связь с мастером сервисного центра»

Представленные проектные решения и программные коды оформлены как автоматизированная информационная система, которая сопряжена с информационной системой сервисного центра. Проведено тестирование программных средств и согласование графики сайта со специалистами сервисного центра.

В настоящее время ведется комплекс работ по внедрению разработки на конкретном предприятии.

## ЗАКЛЮЧЕНИЕ

В процессе проделанной работы были достигнуты цели и решены задачи исследования.

А именно, изучены теоретические основы проектирования автоматизированных информационных систем и комплекс типовых проектных решений по их созданию. Выполнен обзор и проведено сравнение существующих web-технологий для информационного сопровождения процессов технического обслуживания и ремонта цифровой техники сервисного центра.

Результатом работы стал сайт, позволяющий информировать клиентов о процессе технического обслуживания и состоянии выполнения их заказа.

Сайт оформлен с использованием проектных решений создания личных кабинетов клиентов, вход для работы в котором выступает номер заказа.

В настоящее время ведется комплекс работ по внедрению разработки на конкретном предприятии.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. JetBrainsPhpStorm[Электронный ресурс]. – Загл. с экрана. – URL:<https://ru.wikipedia.org/wiki/PhpStorm>.– 10.05.2017
2. Model-View-Controller[Электронный ресурс]. – Загл. с экрана. – URL:<https://ru.wikipedia.org/wiki/Model-View-Controller>.–08.05.2017
3. Symfony – свободный фреймворк[Электронный ресурс]. – Загл. с экрана. – URL:<https://ru.wikipedia.org/wiki/Symfony>.– 04.05.2017
4. Web технологии [Электронный ресурс] / Кожеватов П. Р., Шишкин В. И. –Москва: МГТУ МИРЭА, 2013. – Загл. с титул. Экрана
5. Web-технологии [Текст] / И. Н. Васильева, Д. Ю. – Санкт-Петербург: Изд-во Санкт-Петербургского гос. экономического ун-та, 2014. – 67 с.: ил., табл. – Библиогр.: с. 64-67
6. Web-технологии и Web-дизайн [Электронный ресурс]. – Тюмень: Издательство Тюменского государственного университета, 2014
7. Баранов, Иван Викторович Веб-технологии [Электронный ресурс] / Баранов И. В., Краюшкин В. В., Лесников Г. И. – Москва: НИУ МЭИ, 2013. – Загл. с титул. Экрана
8. Бузюков, Лев Борисович WEB-технологии [Текст] / Л. Б. Бузюков. – Санкт-Петербург: СПбГУТ, 2011. – 149, [2] с.: ил., табл. – Библиогр.: с. 151
9. Васильева, Ирина Николаевна Web-технологии [Текст] / И. Н. Васильева, Д. Ю. – Санкт-Петербург: Изд-во Санкт-Петербургского гос. экономического ун-та, 2014. – 67 с.: ил., табл. – Библиогр.: с. 64-67
- 10.Веб-технологии [Электронный ресурс] / Баранов И. В., Краюшкин В. В., Лесников Г. И.; Нац. исследовательский ун-т МЭИ. – Москва: НИУ МЭИ, 2013. – Загл. с титул.экрана
- 11.Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания:



- ГОСТ 34.601-90 [Электронный ресурс]. – Загл. с экрана. – URL:  
<http://base.garant.ru/187735/>. – 10.05.2017
12. Методы и средства проектирования информационных систем [Текст] /  
А. А. Басаргин. – Новосибирск : СГУГиТ, 2015. – 235 с.
13. Олис. Цифровой сервис [Электронный ресурс]. – Загл. с экрана. – URL:  
<http://olis22.ru>. – 12.05.2017
14. Проектирование информационных систем [Электронный ресурс] /  
[сост.: Макаров К. В.]. – Муром: МИ ВлГУ, 2016

## ПРИЛОЖЕНИЕ

```
<div class="header-block">
    <div class="header-block_icon-wrapper"><em class="fa fa-sign-
in">&nbsp;</em></div>
    <span class="header-block_title"><a href="http://api.e-
accident.eu/sign-in">Личный кабинет</a></span>
    <p class="header-block_small-text">&nbsp;</p>
</div>
<div id="center_column" class="center_column col-xs-12 col-sm-9">
    <?php if($order == null){ ?>
    <h5>Такого заказа не существует</h5>
    <?php } ?>
    <?php if($order != null){ ?>
    <h5>Заказ номер: <span id="numberZakaz"> <?php echo $order-
>getNumber() ?> </span> </h5>
    <?php echo $order->getGeneral() ?>
    <hr style="border-color: lightslategray; margin-left: -20px"
width="800">
    <table border="1">
        <tr>
            <th>Наименование, характеристика, артикул материалов за-
казчика</th>
        </tr>
        <tr>
            <td>Оборудование: <?php echo $order->getDevice() ?></td>
```

```

</tr>
<tr>
    <td>Серийный номер: <?php echo $order->getSerial() ?></td>
</tr>
<tr>
    <td>Оборудование: <?php echo $order->getDate() ?></td>
</tr>
<tr>
    <td>Неисправность: <?php echo $order->getDefect() ?></td>
</tr>
<tr>
    <td>Состояние заказа: <?php echo $order->getStatus() ?></td>
</tr>
<tr>
    <td>Стоимость ремонта: <?php echo $order->getPrice()
?></td>
</tr>
<tr>
    <td>Комментарий: <?php echo $order->getComment() ?></td>
</tr>
</table>
</div>
<div class="block_content" style="">
    <ul class="tree dynamized" style="display: block;">

```

```

</li>
<span onclick="
    var url = 'http://api.e-accident.eu/current/' + '<?php echo
$order->getNumber() ?>';
    window.location.href = url"><a href="#" title=""> Теку-
щий заказ </a> </span>
</li>
<li>
<span onclick="
    var url = 'http://api.e-accident.eu/history/' + '<?php echo
$order->getHistoryId() ?>' + '/' + '<?php echo $order->getNumber() ?>';
    window.location.href = url"><a href="#" title=""> Исто-
рия заказов </a> </span>
</li>
<li>
<span onclick="var url = 'http://api.e-accident.eu/mail/' +
'<?php echo $order->getNumber() ?>';
    window.location.href = url"><a href="#" title=""> Свя-
заться с мастером </a> </span>
</li>
</ul>
</div>
</div>
<div id="center_column" class="center_column col-xs-12 col-sm-9">
    <p><textarea id="text" rows="10" cols="45"
name="text"></textarea></p>

```

```
<p>Ваш E-mail<input id="mail" type="text" value="" style="margin-left: 10px"></p>
```

```
<p><input type="submit" value="Отправить" onclick="
    var mail = $('#mail').val();
    var text = $('#text').val();
    var url = 'http://api.e-accident.eu/send_mail/' + '<?php echo $order->getNumber() ?>' + '/' + mail + '/' + text;
    window.location.href = url"><a href="#" title=""></p>
</div>
```

```
<div id="page">
```

```
<div class="header-container">
```

```
<header id="header">
```

```
<div class="banner">
```

```
<div class="container">
```

```
<div class="row">
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="nav">
```

```
<div class="container">
```

```
<div class="row">
```

```
<nav><!-- Block user information module NAV -->
```

```
<div class="header_user_info">
```

```
<a class="login" href="http://olis22.ru/my-account"
rel="nofollow" title="Войти в учетную запись">
```

Войти

```
</a>
</div>
<!-- /Block usmodule NAV -->
<!-- Block languages module -->
<!-- /Block languages module -->
<div id="contact-link">
    <a href="http://olis22.ru/contact-us" title="Свяжитесь с
нами">Свяжитесь с нами</a>
</div>
<span class="shop-phone">
    <i class="icon-phone"></i>Звоните нам: <strong>8 (3852) 539-810,
532-241, 455-872</strong>
</span>
</nav>
</div>
</div>
</div>
<div>
    <div class="container">
        <div class="row">
            <div id="header_logo">
                <a href="http://olis22.ru/" title="Олис 22">
                    
                </a>
```

```
</div>

<div class="header-block">

  <div class="header-block_icon-wrapper"><em class="fa fa-truck">&nbsp;</em></div>

  <span class="header-block_title"><a href="http://olis22.ru/content/7-dileram">Дилерам</a></span>

  <p><span class="header-block_small-text">&nbsp;</span></p>

</div>

<div class="header-block">

  <div class="header-block_icon-wrapper"><em class="fa fa-laptop">&nbsp;</em></div>

  <span class="header-block_title"><a href="http://olis22.ru/content/6-uslugi">Сервис центр</a></span>

  <p class="header-block_small-text">&nbsp;</p>

</div>

<div class="header-block">

  <div class="header-block_icon-wrapper"><em class="fa fa-sign-in">&nbsp;</em></div>

  <span class="header-block_title"><a href="http://api.e-accident.eu/sign-in">Личный кабинет</a></span>

  <p class="header-block_small-text">&nbsp;</p>

</div>

<!-- MODULE Block cart -->

<div class="col-md-2 col-sm-12 shopping_cart-wrapper clearfix">

  <div class="shopping_cart">

    <a href="http://olis22.ru/quick-order" title="Просмотр корзины" rel="nofollow">
```

```
<b>Корзина</b>
<span class="ajax_cart_quantity invisible">0</span>
<span class="ajax_cart_product_txt invisible"></span>
<span class="ajax_cart_product_txt_s invisible"></span>
<span class="ajax_cart_total invisible">
    </span>
<span class="ajax_cart_no_product">(0)</span>
</a>
<div class="cart_block block exclusive" style="">
    <div class="block_content">
        <!-- block list of products -->
        <div class="cart_block_list">
            <p class="cart_block_no_products">
                Нет товаров
            </p>
            <div class="cart-prices">
                <div class="cart-prices-line first-line">
                    <span class="price
cart_block_shipping_cost ajax_cart_shipping_cost invisible">
                        To be determined
                    </span>
                    <span class="invisible">
                        Доставка
                    </span>
                </div>
            </div>
        </div>
    </div>
</div>
```



```

        </div>
        <div class="cart-prices-line last-line">
            <span class="price cart_block_total
ajax_block_cart_total">0,00 руб</span>
            <span>Итого, к оплате:</span>
        </div>
    </div>
    <p class="cart-buttons">
        <a id="button_order_cart" class="btn btn-default
button button-small" href="http://olis22.ru/quick-order" title="Оформить заказ"
rel="nofollow">
            <span>
                Оформить заказ<i
class="icon-chevron-right right"></i>
            </span>
        </a>
    </p>
</div>
</div>
</div><!-- .cart_block -->
</div>
</div>

<div id="layer_cart">
    <div class="clearfix">
        <div class="layer_cart_product col-xs-12 col-md-6">

```

```

<span class="cross" title="Close window"></span>
<h2>
  <i class="icon-check"></i>Товар добавлен в корзину
</h2>
<div class="product-image-container layer_cart_img">
</div>
<div class="layer_cart_product_info">
  <span id="layer_cart_product_title" class="product-
name"></span>
  <span id="layer_cart_product_attributes"></span>
  <div>
    <strong class="dark">Quantity</strong>
    <span id="layer_cart_product_quantity"></span>
  </div>
  <div>
    <strong class="dark">Итого, к оплате:</strong>
    <span id="layer_cart_product_price"></span>
  </div>
</div>
</div>
<div class="layer_cart_cart col-xs-12 col-md-6">
  <h2>
    <!-- Plural Case [both cases are needed because page may
be updated in Javascript] -->
    <span class="ajax_cart_product_txt_s invisible">

```

There are 0 items in your cart.

</span>

<!-- Singular Case [both cases are needed because page may be updated in Javascript] -->

<span class="ajax\_cart\_product\_txt ">

There is 1 item in your cart.

</span>

</h2>

<div class="layer\_cart\_row">

<strong class="dark">

Total products

</strong>

<span class="ajax\_block\_products\_total">

</span>

</div>

<div class="layer\_cart\_row">

<strong class="dark invisible">

Total shipping&nbsp;

</strong>

<span class="ajax\_cart\_shipping\_cost invisible">

To be determined

</span>

</div>

<div class="layer\_cart\_row">

<strong class="dark">

```

        Итого, к оплате:
    </strong>
    <span class="ajax_block_cart_total">
</span>
    </div>
    <div class="button-container">
        <span class="continue btn btn-default button ex-
clusive-medium" title="Продолжить покупки">
            <span>
                <i class="icon-chevron-left
left"></i>Продолжить покупки
            </span>
        </span>
        </span>
        <a class="btn btn-default button button-medium"
href="http://olis22.ru/quick-order" title="Оформить заказ" rel="nofollow">
            <span>
                Оформить заказ<i class="icon-
chevron-right right"></i>
            </span>
        </a>
    </div>
</div>
<div class="crossseling"></div>
</div> <!-- #layer_cart -->
<div class="layer_cart_overlay"></div>

```

```

<!-- /MODULE Block cart -->

<!-- Menu -->

<div id="block_top_menu" class="sf-contener clearfix col-lg-12">
  <div class="cat-title">Личный кабинет</div>
  <ul class="sf-menu clearfix menu-content container sf-js-enabled
sf-arrows">
    <li><a href="http://olis22.ru/65-matricy-dlya-noutbukov" title
    <li class="sf-search noBack" style="float:right">
      <div id="search_block_top" style="display: block;">
        <form method="get" action="http://olis22.ru/search"
id="searchbox" style="display: block;">
          <label for="search_query_top"><!-- image on back-
ground --></label>
          <input type="hidden" name="controller" val-
ue="search">
          <input type="hidden" name="orderby" val-
ue="position">
          <input type="hidden" name="orderway" value="desc">
          <input class="search_query ac_input" type="text"
id="search_query_top" name="search_query" value="" autocomplete="off">
          <input type="submit" name="submit_search" val-
ue="Поиск" class="button">
        </form>
      </div></li>
    </ul>
  </div>
<!-- / Menu -->

```

```

<!-- BelVG block search module TOP -->
<!-- BelVG block search js-block -->
<script type="text/javascript">
    // <![CDATA[
    function tryToCloseInstantSearch() {
        if ($('#old_center_column').length > 0)
        {
            $('#center_column').remove();
            $('#old_center_column').attr('id', 'center_column');
            $('#center_column').show();
            return false;
        }
    }
    instantSearchQueries = new Array();
    function stopInstantSearchQueries(){
        for(i=0;i<instantSearchQueries.length;i++) {
            instantSearchQueries[i].abort();
        }
        instantSearchQueries = new Array();
    }
    $("#search_query_top").keyup(function(){
        if($(this).val().length > 0){
            stopInstantSearchQueries();
            instantSearchQuery = $.ajax({
                url: 'http://olis22.ru/search',

```

```

data: {
    instantSearch: 1,
    id_lang: 1,
    q: $(this).val()
},
dataType: 'html',
type: 'POST',
success: function(data){
    if($("#search_query_top").val().length > 0)
    {
        tryToCloseInstantSearch();
        $('#center_column').attr('id', 'old_center_column');
        $('#old_center_column').after('<div
id="center_column" class="" + $('#old_center_column').attr('class') +
">'+data+'</div>');

        $('#old_center_column').hide();

        $("#instant_search_results a.close").click(function()
{
        $("#search_query_top").val("");
        return tryToCloseInstantSearch();
    });
        return false;
    }
    else
        tryToCloseInstantSearch();
}

```

```

    });
    instantSearchQueries.push(instantSearchQuery);
  }
  else
    tryToCloseInstantSearch();
});
// ]]>
</script>
<script type="text/javascript">
  // <![CDATA[
  $('document').ready( function() {
    $("#search_query_top")
      .autocomplete(
        'http://olis22.ru/search', {
          minChars: 3,
          max: 10,
          width: 500,
          selectFirst: false,
          scroll: false,
          dataType: "json",
          formatItem: function(data, i, max, value, term) {
            return value;
          },
          parse: function(data) {
            var mytab = new Array();

```



```

        for (var i = 0; i < data.length; i++)

            mytab[mytab.length] = { data: data[i], value: '<div
class = "belvg_search_container"><div class =
"belvg_search_container_left"><img src="" + data[i].id_image + "" title="" + da-
ta[i].pname + "" width="58" height="58"/></div><div class =
"belvg_search_container_right"><p class="category_name">' + data[i].cname +
'</p><p class="product_name">' + data[i].pname + '</p><p
class="product_description">' + data[i].description_short + '</p><p
class="product_price">' + data[i].price + ' ' + data[i].currency_sign +
'</p></div></div>' };

        //console.log(mytab);

        return mytab;

    },

    extraParams: {

        ajaxSearch: 1,

        id_lang: 1

    }

}

)

.result(function(event, data, formatted) {

    $('#search_query_top').val(data.pname);

    document.location.href = data.product_link;

})

});

// ]]>

</script>

<!-- /BelVG block search module TOP -->

```

```
<!-- Block permanent links module HEADER -->
```

```
<ul id="header_links">
```

```
  <li id="header_link_contact"><a href="http://olis22.ru/contact-  
us" title="контакты">контакты</a></li>
```

```
  <li id="header_link_sitemap"><a href="http://olis22.ru/sitemap"  
title="карта сайта">карта сайта</a></li>
```

```
  <li id="header_link_bookmark">
```