

Минобрнауки России

Юго-Западный государственный университет

Кафедра программной инженерии

КУРСОВАЯ РАБОТА (ПРОЕКТ)

по дисциплине «Проектирование и архитектура программных систем»
наименование дисциплины

на тему Разработка библиотеки для чтения изображений формата JPEG

Направление подготовки (специальность) 09.03.04
(код, наименование)

Программная инженерия

Автор работы (проекта) Лукьянчиков А. Е.
(инициалы, фамилия) (подпись, дата)

Группа ПО-33б

Руководитель работы (проекта) А. А. Чаплыгин
(инициалы, фамилия) (подпись, дата)

Работа (проект) защищена
(дата)

Оценка

Члены комиссии

подпись, дата	фамилия и. о.
подпись, дата	фамилия и. о.
подпись, дата	фамилия и. о.

Курск 2025 г.

Минобрнауки России

Юго-Западный государственный университет

Кафедра программной инженерии

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ (ПРОЕКТ)

Студента Лукьянчикова А.Е., шифр хх-хх-хххх, группа ПО-33б

1. Тема «Разработка библиотеки для чтения изображений формата JPEG» .

2. Срок предоставления работы к защите

3. Исходные данные для создания программной системы:

3.1. Перечень решаемых задач:

- 1) проанализировать ИТ-инфраструктуру предприятия;
- 2) разработать концептуальную модель системы управления ИТ-инфраструктурой предприятия на основе подхода к управлению и организации ИТ-услуг ITSM;
- 3) спроектировать программную систему управления ИТ-инфраструктурой предприятия;
- 4) сконструировать и протестировать программную систему управления ИТ-инфраструктурой предприятия.

3.2. Входные данные и требуемые результаты для программы:

1) Входными данными для программной системы являются: данные справочников комплектующих, конфигураций, ПО, критериев качества SLA, ИТ-услуг, департаментов компании; технические данные ИТ-ресурсов; данные входящих заявок на ИТ-ресурсы; данные запросов поставщикам на комплектующие.

2) Выходными данными для программной системы являются: сформированные заявки на обслуживание ИТ-ресурсов; сформированные запросы на закупку комплектующих; сведения о выполненных работах по заявкам; статусы заявок; выходные отчеты (инфографика) – по качеству услуг, по со-

стоянию ИТ-ресурсов, по деятельности ИТ-отдела, по стоимости обслуживания ИТ-ресурсов, воронка заявок.

4. Содержание работы (по разделам):

4.1. Введение.

4.1. Анализ предметной области.

4.2. Техническое задание: основание для разработки, назначение разработки, требования к программной системе, требования к оформлению документации.

4.3. Технический проект: общие сведения о программной системе, проект данных программной системы, проектирование архитектуры программной системы, проектирование пользовательского интерфейса программной системы.

4.4. Рабочий проект: спецификация компонентов и классов программной системы, тестирование программной системы, сборка компонентов программной системы.

4.5. Заключение.

4.6. Список использованных источников.

5. Перечень графического материала:

Руководитель работы (проекта)

(подпись, дата)

А. А. Чаплыгин

(инициалы, фамилия)

Задание принял к исполнению

(подпись, дата)

Лукьянчиков А. Е.

(инициалы, фамилия)

РЕФЕРАТ

Объем работы равен 22 страницам. Работа содержит 0 иллюстраций, 4 таблицы, 0 библиографических источников и 0 листов графического материала. Количество приложений – 2. Графический материал представлен в приложении А. Фрагменты исходного кода представлены в приложении Б.

Перечень ключевых слов: коммерческий сайт, Система, CMS, Битрикс, Joomla, аддитивные технологии, 3D-принтеры, услуги, сервисы, информатизация, автоматизация, информационные технологии, веб-форма, Apache, классы, база данных, средства защиты информации, подсистема, компонент, модуль, сущность, информационный блок, метод, контент-редактор, администратор, пользователь, web-сайт.

Объектом разработки является web-сайт компании, занимающейся производством 3D-принтеров, выпуском оборудования для создания порошков, разработкой программного обеспечения и организацией центров аддитивного производства.

Целью выпускной квалификационной работы является привлечение клиентов, увеличение заказов, информирование о продукции и услугах путем создания сайта компании.

В процессе создания сайта были выделены основные сущности путем создания информационных блоков, использованы классы и методы модулей, обеспечивающие работу с сущностями предметной области, а также корректную работу web-сайта, разработаны разделы, содержащие информацию о компании, ее деятельности, производимой продукции и услугах, разработан сервис по заказу 3D-деталей.

При разработке сайта использовалась система управления контентом «1С-Битрикс: Управление сайтом».

Разработанный сайт был успешно внедрен в компании.

ABSTRACT

The volume of work is 22 pages. The work contains 0 illustrations, 4 tables, 0 bibliographic sources and 0 sheets of graphic material. The number of applications is 2. The graphic material is presented in annex A. The layout of the site, including the connection of components, is presented in annex B.

List of keywords: commercial website, System, CMS, Bitrix, Joomla, additive technologies, 3D printers, services, services, informatization, automation, information technology, web form, Apache, classes, database, component, module, entity, information block, method, content editor, administrator, user, web site.

The object of the research is the analysis of information technologies for the development of a production company's website.

The object of the development is the website of a company engaged in the production of 3D printers, the production of equipment for the creation of powders, software development and the organization of additive manufacturing centers.

The purpose of the final qualifying work is to attract customers, increase orders, inform about products and services by creating a company website.

In the process of creating the site, the main entities were identified by creating information blocks, classes and methods of modules were used to ensure work with the entities of the subject area, as well as the correct operation of the website, sections containing information about the company, its activities, products and services were developed, a service for ordering 3D parts was developed.

When developing the site, the content management system «1C – Bitrix: Site Management» was used.

The developed website was successfully implemented in the company.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	8
1 Анализ предметной области	11
1.1 Введение и актуальность	11
1.2 Типы растровых изображений	11
1.3 Библиотеки для работы с изображениями	12
1.4 Формат JPEG	12
2 Техническое задание	15
2.1 Основание для разработки	15
2.2 Цель и назначение разработки	15
2.3 Требования пользователя к интерфейсу web-сайта	15
2.4 Моделирование вариантов использования	15
2.5 Требования к оформлению документации	16
ПРИЛОЖЕНИЕ А Фрагменты исходного кода программы	17

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

БД – база данных.

ИС – информационная система.

ИТ – информационные технологии.

КТС – комплекс технических средств.

ОМТС – отдел материально-технического снабжения.

ПО – программное обеспечение.

РП – рабочий проект.

СУБД – система управления базами данных.

ТЗ – техническое задание.

ТП – технический проект.

UML (Unified Modelling Language) – язык графического описания для объектного моделирования в области разработки программного обеспечения.

ВВЕДЕНИЕ

Аддитивные технологии (АТ) начали активно развиваться со времени получения первых трехмерных изображений изделий на дисплеях компьютеров. Начало положила стереолитография, затем довольно многочисленные новые принципы стали называть технологиями быстрого прототипирования, затем укоренилось название «Аддитивные технологии». Интенсивность развития данных технологий не имеет аналогов. АТ изменили процессы проектирования и конструирования изделий, превратив их в процессы непрерывного создания изделий. Современные проектирование и производство изделий невозможно представить без данного рода технологий. 3D-принтеры стали такими же распространенными, как и персональные компьютеры. С помощью 3D-принтеров получают ткани, обувь, продукты питания, а также выращивают человеческие органы. Во многих отраслях, например, в космической отрасли, альтернативы аддитивным технологиям нет.

АТ предполагают изготовление детали методом послойного нанесения материала, в отличие от традиционных методов формирования детали, за счёт удаления материала из массива заготовки.

При использовании АТ все стадии реализации проекта от идеи до материализации находятся в единой технологической цепи, в которой каждая технологическая операция выполняется в цифровой CAD/CAM/CAE-системе.

Современные компании, видя, как развиваются информационные технологии, пытаются использовать их выгодно для своего бизнеса, поэтому запускают свой web-сайт. С его помощью предприятие может заявить о себе, проинформировать потенциального заказчика об услугах или продуктах, которые предоставляет, а также позволяет пользователям сделать с помощью сайта онлайн-заказ, произвести покупку или оплатить счета.

Сайт считается лицом компании и может существенно повысить ее имидж. Любой пользователь сети Интернет сможет получить необходимую информацию о компании в любой момент, появляется возможность найти контактные телефоны, адрес и e-mail, чтобы связаться с компанией. Сейчас

большинство клиентов узнают о ее существовании именно через сайт. Поэтому сайт можно назвать самой лучшей рекламой.

Главной задачей профессионально построенного сайта является превращение посетителя, зашедшего на сайт, в потенциального клиента.

Цель настоящей работы – разработка web-сайта компании для привлечения новой аудитории, увеличения заказов, рекламы продукции и услуг компании. Для достижения поставленной цели необходимо решить *следующие задачи*:

- провести анализ предметной области;
- разработать концептуальную модель web-сайта;
- спроектировать web-сайт;
- реализовать сайт средствами web-технологий.

Структура и объем работы. Отчет состоит из введения, 4 разделов основной части, заключения, списка использованных источников, 2 приложений. Текст выпускной квалификационной работы равен 22 страницам.

Во введении сформулирована цель работы, поставлены задачи разработки, описана структура работы, приведено краткое содержание каждого из разделов.

В первом разделе на стадии описания технической характеристики предметной области приводится сбор информации о деятельности компании, для которой осуществляется разработка сайта.

Во втором разделе на стадии технического задания приводятся требования к разрабатываемому сайту.

В третьем разделе на стадии технического проектирования представлены проектные решения для web-сайта.

В четвертом разделе приводится список классов и их методов, использованных при разработке сайта, производится тестирование разработанного сайта.

В заключении излагаются основные результаты работы, полученные в ходе разработки.

В приложении А представлен графический материал. В приложении Б представлены фрагменты исходного кода.

1 Анализ предметной области

1.1 Введение и актуальность

В современном цифровом мире растровые изображения являются фундаментальным типом данных. Они повсеместно используются в веб-разработке, мобильных приложениях, научных исследованиях (медицина, астрономия), машинном обучении (компьютерное зрение), дизайне и фотографии.

Обработка изображений — ресурсоемкая задача, требующая эффективных и специализированных инструментов. Существует множество библиотек (OpenCV, Pillow, ImageMagick, libpng, libjpeg-turbo), однако их зависимости, сложность интеграции или лицензионные ограничения часто создают препятствия для разработчиков.

1.2 Типы растровых изображений

Библиотеки обрабатывают растровые изображения, читают и записывают их, для работы с ними необходимо знать каких типов бывают растровые изображения и в чем их разница. Основное отличие типов растровых изображений в их хранении и записи, некоторые форматы сжимают данные и или удаляют ненужные куски данных, в зависимости от предназначения, выбираются различные форматы изображений.

Наиболее популярные из них:

1. PNG, этот формат использует алгоритм сжатия без потерь (LZ77 - Deflate), в особенности этого формата входит поддержка алфа-канала
2. JPEG, этот формат использует алгоритм сжатия с потерями (Дискретное косинусное преобразование)
3. PPM, особенность этого формата в том, что он не использует алгоритмы сжатия, и хранит данные, как они есть, каждый пиксель в виде числа

1.3 Библиотеки для работы с изображениями

Уже существует огромное множество библиотек предназначенных для различных языков и различных задач для работы с изображениями. Они отличаются открытостью кода, удобством разным подходом к представлению данных. Каждый выбирает одну из них на основе поставленной задачи, например, для работы с компьютерным зрением подойдет OpenCV, а для простой обработки изображений на Python хватит и Pillow.

Основные библиотеки и их особенности:

1. OpenCV, мощная библиотека для обработки изображений, использующаяся для работы с компьютерным зрением, существует почти для любого языка программирования, но очень большая по размеру
2. Pillow, библиотека для работы с изображениями, их чтением, обработкой, записью, но существует только для языка Python
3. ImageMagick, библиотека с открытым исходным кодом, включает в себя широкий набор функций, подходит для работы с широким списком форматов

Несмотря на различия в функциональности этих библиотек, они имеют ряд общих черт, которые должны быть у любой библиотеки для работы с изображениями. Основные функции:

1. Поддержка различных форматов изображений(PNG, JPEG, TIFF и др.)
2. Чтение и запись изображений
3. Возможность обрабатывать или редактировать изображение

1.4 Формат JPEG

JPEG — один из популярных растровых графических форматов, применяемый для хранения фотографий и подобных им изображений. Файлы, содержащие данные JPEG, обычно имеют расширения (суффиксы) .jpg (самое популярное), .jfif, .jpe или .jpeg. MIME-тип — image/jpeg.

Алгоритм JPEG позволяет сжимать изображение как с потерями, так и без потерь (режим сжатия lossless JPEG). Поддерживаются изображения с линейным размером не более 65535×65535 пикселей.

Процесс сжатия JPEG изображения происходит в несколько этапов, сначала изображение преобразуется из RGB цветового пространства в YCbCr, после этого оно проходит дискретное косинусное преобразование, а в конце кодируются длины серий, делты, Хаффмана.

Стандарт JPEG предусматривает два основных способа представления кодируемых данных.

Наиболее распространённым, поддерживаемым большинством доступных кодеков, является последовательное (sequential JPEG) представление данных, предполагающее последовательный обход кодируемого изображения разрядностью 8 бит на компоненту (или 8 бит на пиксель для чёрно-белых полутоновых изображений) поблочно слева направо, сверху вниз. Над каждым кодируемым блоком изображения осуществляются описанные выше операции, а результаты кодирования помещаются в выходной поток в виде единственного «скана», то есть массива кодированных данных, соответствующего последовательно пройденному («просканированному») изображению. Основной или «базовый» (baseline) режим кодирования допускает только такое представление (и хатфмановское кодирование квантованных коэффициентов ДКП). Расширенный (extended) режим наряду с последовательным допускает также прогрессивное (progressive JPEG) представление данных, кодирование изображений разрядностью 12 бит на компоненту/пиксель (сжатие таких изображений спецификацией JFIF не поддерживается) и арифметическое кодирование квантованных коэффициентов ДКП.

В случае progressive JPEG сжатые данные записываются в выходной поток в виде набора сканов, каждый из которых описывает изображение полностью с всё большей степенью детализации. Это достигается либо путём записи в каждый скан не полного набора коэффициентов ДКП, а лишь какой-то их части: сначала — низкочастотных, в следующих сканах — высокочастотных (метод «spectral selection» то есть спектральных выборок), либо путём

последовательного, от скана к скану, уточнения коэффициентов ДКП (метод «successive approximation», то есть последовательных приближений). Такое прогрессивное представление данных оказывается особенно полезным при передаче сжатых изображений с использованием низкоскоростных каналов связи, поскольку позволяет получить представление обо всём изображении уже после передачи незначительной части JPEG-файла.

Алгоритм JPEG наиболее эффективен для сжатия фотографий и картин, содержащих реалистичные сцены с плавными переходами яркости и цвета. Наибольшее распространение JPEG получил в цифровой фотографии и для хранения и передачи изображений с использованием Интернета. Формат JPEG в режиме сжатия с потерями малоприспособлен для сжатия чертежей, текстовой и знаковой графики, где резкий контраст между соседними пикселями приводит к появлению заметных артефактов. Такие изображения целесообразно сохранять в форматах без потерь, таких как JPEG-LS, TIFF, GIF, PNG, либо использовать режим сжатия Lossless JPEG. JPEG (как и другие форматы сжатия с потерями) не подходит для сжатия изображений при многоэтапной обработке, так как искажения в изображения будут вноситься каждый раз при сохранении промежуточных результатов обработки. JPEG не должен использоваться и в тех случаях, когда недопустимы даже минимальные потери, например при сжатии астрономических или медицинских изображений. В таких случаях может быть рекомендован предусмотренный стандартом JPEG режим сжатия Lossless JPEG (который, однако, не поддерживается большинством популярных кодеков) или стандарт сжатия JPEG-LS.

2 Техническое задание

2.1 Основание для разработки

Основанием для разработки является задание на курсовую работу «Разработка библиотеки для чтения изображений формата JPEG».

2.2 Цель и назначение разработки

Основной задачей курсовой работы является разработка библиотеки для чтения и декодирования JPEG изображений.

Задачами данной разработки являются:

- Создание функции для чтения файла формата JPEG;
- Создание функции для декодирования маркеров файла JPEG;
- Создание функций для декодирования сжатого набора данных JPEG файла.

2.3 Требования пользователя к интерфейсу web-сайта

Т.к. создаваемый проект является библиотекой, то пользовательский интерфейс будет программным, то есть реализован через прямое взаимодействие пользователя с функциями библиотеки.

Библиотека должна включать в себя следующие функции:

- Функция для чтения данных изображения;
- Функция определения метаданных изображения по маркерам;
- Функция нахождения DC, AC коэффициентов;
- Функция обратная квантованию;
- Функция обратного дискретно-косинусного преобразования;
- Функция преобразования изображения в другую цветовую палитру.

2.4 Моделирование вариантов использования

Для разрабатываемого сайта была реализована модель, которая обеспечивает наглядное представление вариантов использования библиотеки.

Она помогает в физической разработке и детальном анализе взаимосвязей объектов. При построении диаграммы вариантов использования применяется унифицированный язык визуального моделирования UML.

Диаграмма вариантов описывает функциональное назначение разрабатываемой системы. То есть это то, что система будет непосредственно делать в процессе своего функционирования. Она является исходным концептуальным представлением системы в процессе ее проектирования и разработки. Проектируемая система представляется в виде ряда прецедентов, предоставляемых системой актерам или сущностям, которые взаимодействуют с системой. Актером или действующим лицом является сущность, взаимодействующая с системой извне (например, человек, техническое устройство). Прецедент служит для описания набора действий, которые система предоставляет актеру.

На основании анализа предметной области в программе должны быть реализованы следующие прецеденты:

1. Открытие файла формата JPEG.
2. Чтение файла формата JPEG.
3. Декодирование файла формата JPEG.

2.5 Требования к оформлению документации

Разработка программной документации и программного изделия должна производиться согласно ГОСТ 19.102-77 и ГОСТ 34.601-90. Единая система программной документации.

ПРИЛОЖЕНИЕ А

Фрагменты исходного кода программы

main.tex

```
1 \input{setup.tex}
2
3 % Режим шаблона (должен быть включен один из трех)
4 %\VKPtrue
5 %\Практикаtrue
6 \Курсоваяtrue
7
8 \newcommand{\Дисциплина}{<<Проектирование и архитектура программных систем>>}
9 % для курсовой
10 \newcommand{\КодСпециальности}{09.03.04} % Курсовая
11 \newcommand{\Специальность}{Программная инженерия} % Курсовая
12 \newcommand{\Тема}{Разработка библиотеки для чтения изображений формата JPEG}
13 % ВКР Курсовая
14 \newcommand{\ТемаВтораяСтрока}{}
15 \newcommand{\ГдеПроводитсяПрактика}{Юго-Западном государственном университете}
16 % для практики
17 \newcommand{\РуководительПрактПредпр}{Куркина А. В.} % для практики
18 \newcommand{\ДолжнРуководительПрактПредпр}{директор} % для практики
19 \newcommand{\РуководительПрактУнивер}{Чаплыгин А. А.} % для практики
20 \newcommand{\ДолжнРуководительПрактУнивер}{к.т.н. доцент} % для практики
21 \newcommand{\Автор}{Лукьянчиков А. Е.}
22 \newcommand{\АвторРод}{Лукьянчикова А.Е.}
23 \newcommand{\АвторПолностьюРод}{Иванова Ивана Ивановича} % для практики
24 \newcommand{\Шифр}{xx-xx-xxxx}
25 \newcommand{\Курс}{3} % для практики
26 \newcommand{\Группа}{ПО-336}
27 \newcommand{\Руководитель}{А. А. Чаплыгин} % для ВКР и курсовой
28 \newcommand{\Нормоконтроль}{А. А. Чаплыгин} % для ВКР
29 \newcommand{\ЗавКаф}{А. В. Малышев} % для ВКР
30 \newcommand{\ДатаПриказа}{«07» апреля 2023~г.} % для ВКР
31 \newcommand{\НомерПриказа}{1505-с} % для ВКР
32 \newcommand{\СрокПредоставления}{} % для ВКР, курсового
33
34 \begin{document}
35 \maketitle
36 \ifПрактика{}\else{
37   \input{ЛистЗадания}
38   \input{Реферат}}\fi
39 \tableofcontents
40 \input{Обозначения}
41 \ifПрактика{}\else{\input{Введение}}\fi
42 \input{Анализ}
43 \input{ТехЗадание}
44 %\input{ТехПроект}
45 \ifПрактика{}\else{
46   %\input{РабочийПроект}
47   %\input{Заключение}
48 } \fi
49 %\input{СписокИсточников}
```

```

47 \ifBKP{\input{Плакаты}}\fi
48 \ifПрактика{}\else{\input{Код}}\fi
49 \end{document}

```

ТехПроект.tex

```

1 \section{Технический проект}
2 \subsection{Общая характеристика организации решения задачи}
3
4 Необходимо спроектировать и разработать сайт, который должен способствовать
   продвижению компании на рынке.
5
6 Интернет-сайт представляет собой набор взаимосвязанных электронных страниц,
   которые сгруппированы по разделам, содержащие текстовую, графическую, а
   также мультимедийную информацию (изображения, видеоролики и пр.). Сайт
   располагается в Интернете по определенному адресу – доменному имени сайта
   в виде www.имя\_сайта.ru. Каждая страница web-сайта – это текстовый
   документ, написанный на языке программирования (HTML, CSS, JavaScript и т.
   д.).
7
8 \subsection{Обоснование выбора технологии проектирования}
9
10 На сегодняшний день информационный рынок, поставляющий программные решения в
   выбранной сфере, предлагает множество продуктов, позволяющих достигнуть
   поставленной цели – разработки web-сайта.
11
12 \subsubsection{Описание используемых технологий и языков программирования}
13
14 В процессе разработки web-сайта используются программные средства и языки
   программирования. Каждое программное средство и каждый язык
   программирования применяется для круга задач, при решении которых они
   необходимы.
15
16 \subsubsection{Язык программирования PHP}
17
18 PHP – язык для написания сценариев, исполняемых на компьютере web-приложения
   посредством интерпретации исходного кода \cite{php}. Основное
   предназначение данного языка – это выполнение на сервере сценариев,
   создающих динамические web-страницы.
19
20 \subsubsection{Язык программирования JavaScript}
21
22 \paragraph{Достоинства языка JavaScript}
23
24 JavaScript – объектно-ориентированный язык программирования для написания
   сценариев \cite{javascript}. Чаще всего JavaScript используется для
   написания сценариев работы с web-страницами, отображаемыми web-браузером.
   Web-бра\у\зер интерпретирует код сценария языка JavaScript, и на основе
   описанных в сценарии действий производит манипуляции с разметкой web-
   страницы. Посредством языка JavaScript реализуется возможность
   программирования на стороне клиента. Предоставляет возможность доступа к
   элементам разметки web-страницы посредством объектов. При создании
   сценариев на языке JavaScript приходится сталкиваться с трудностями,
   связанными с тем, что различные web-браузеры могут по-разному
   интерпретировать эти сценарии. Серьезные трудности возникают, если какой-

```

либо из браузеров не поддерживает тот или иной объект, метод или свойство. Наиболее практичным способом решения данной проблемы является использование библиотеки jQuery. Данная библиотека реализована на языке JavaScript и расширяет возможности данного языка, нивелируя различия между браузерами.

```

25
26 \paragraph{Недостатки языка Javascript}
27
28 0 == []\ // -> true
29
30 0 == \textquotedbl0\textquotedbl\ // -> true
31
32 \textquotedbl0\textquotedbl\ == [] // -> false
33

```

И еще:

```

34
35
36 2 + \textquotedbl 2\textquotedbl\ // -> 22
37
38 2 - \textquotedbl 2\textquotedbl\ // -> 0
39

```

Это был пример раздела с нумерацией вида 1.1.1.1

```

40
41
42 \subsection{Диаграмма компонентов и схема обмена данными между файлами
    компонента}
43

```

Диаграмма компонентов описывает особенности физического представления разрабатываемой системы. Она позволяет определить архитектуру системы, установив зависимости между программными компонентами, в роли которых может выступать как исходный, так и исполняемый код. Основными графическими элементами диаграммы компонентов являются компоненты, интерфейсы, а также зависимости между ними. На рисунке \ref{comp:image} изображена диаграмма компонентов для проектируемой системы. Она включает в себя сервер с операционной системой, на которой установлена система управления содержимым, включающая в себя базу данных и интерфейс. Помимо этого на диаграмме изображен клиентский компьютер с операционной системой, на которой установлен браузер.

```

44
45
46 \begin{figure}[ht]
47 \center{\includegraphics[width=1\linewidth]{comp}}
48 \caption{Диаграмма компонентов}
49 \label{comp:image}
50 \end{figure}
51

```

Любой компонент должен быть вызван в сценарии страницы web-сайта. Web-страница передает данные компоненту в момент вызова последнего.

На рисунке \ref{data:image} представлена схема обмена данными между сценариями компонента при вызове компонента на странице сайта.

```

52
53
54
55
56 \begin{figure}[ht]
57 \center{\includegraphics[width=1\linewidth]{data}}
58 \caption{Диаграмма компонентов}
59 \label{data:image}
60 \end{figure}

```

61

62 При вызове компонента в сценарии web-страницы указываются значения параметров компонента, которые далее посредством массива `\$arParams` передаются в сценарий файла `component.php`.

63

64 В сценарии файла `component.php` посредством метода `\linebreak` `IncludeComponentTemplate` класса `CBitrixComponent` происходит вызов одного из шаблонов компонента. `Id` шаблона также определяется в сценарии страницы web-приложения и неявно для разработчика передается указанный выше метод. Подключается сценарий файла `template.php` одного из шаблонов, в который передается, возможно, измененный в сценарии `component.php` массив `\$arParams` и, также, сформированный в сценарии `component.php` массив `\$arResult`. Оба этих массива доступны также и в файле `result_modifier.php`, который подключается перед подключением файла `template.php`.

65

66 Работа компонента заканчивается в момент завершения работы сценария файла `component.php`, т.е. возможно выполнить действия уже после подключения шаблона. Однако, если массив `\$arResult` будет изменен в сценарии шаблона, в сценарий файла компонента `component.php` измененные данные переданы не будут.

67

68 `\subsection{Диаграмма размещения}`

69

70 Диаграмма размещения (рис.~\ref{place:image}) отражает физические взаимосвязи между программными и аппаратными компонентами системы.

71

72 `\vspace{-8mm}` % чтобы убрать пустую строку, которая осталась после переноса рисунка на следующую страницу

73 `\begin{figure}[ht]`

74 `\center{\includegraphics[width=0.57\linewidth]{place}}`

75 `\caption{Диаграмма размещения. Не помещается на страницу. Очень длинный заголовок}`

76 `\label{place:image}`

77 `\end{figure}`

78

79 Она является хорошим средством для показа маршрутов перемещения объектов и компонентов в распределенной системе.

80

81 В таблице `\ref{ssevsws:table}` приведен пример использования пакета `xltabular` с автоматическим расчетом ширины столбца.

82

83 `\begin{xltabular}{\textwidth}{|c|X|X|}`

84 `\caption{Сравнение протоколов SSE и WebSocket\label{ssevsws:table}}\ \\`
`hline`

85 `~ & \centrow SSE & \centrow WebSocket \\ \hline`

86 `\endfirsthead`

87 `\continuecaption{Продолжение таблицы \ref{ssevsws:table}}`

88 `~ & \centrow SSE & \centrow WebSocket \\ \hline`

89 `\finishhead`

90 Направленность &

91 Однонаправленный, полудуплексный: данные посылает только сервер &

92 Двухнаправленный, полнодуплексный: и сервер, и клиент могут обмениваться сообщениями \\ \hline

93 Соединение & HTTP & WS \\ \hline

```

94 Тип данных & Только текст & Бинарные и текстовые данные \\ \hline
95 Доп. возможности & Встроенный механизм идентификаторов событий и
    переподключения & Переподключение и идентификация события реализуются на
    стороне приложения
96 \end{xltabular}
97
98 \subsection{Содержание информационных блоков. Основные сущности}
99
100 Проанализировав требования, можно выделить шесть основных сущностей:
101 \begin{itemize}
102 \item "<Новости">;
103 \item "<Продукция">;
104 \item "<Услуги">.
105 \end{itemize}
106
107 В состав сущности "<Новости"> можно включить атрибуты, представленные в
    таблице \ref{news:table}.
108
109 \begin{xltabular}{\textwidth}{|l|l|l|p{1.7cm}|X|}
110 \caption{Атрибуты сущности "<Новости">\label{news:table}}\\ \hline
111 \centrow Поле & \centrow Тип & \centrow Обяза\~тельное & \centrow Описание
    \\ \hline
112 \thead{1} & \thead{2} & \centrow 3 & \centrow 4 \\ \hline
113 \endfirsthead
114 \continuecaption{Продолжение таблицы \ref{news:table}}
115 \thead{1} & \thead{2} & \centrow 3 & \centrow 4 \\ \hline
116 \finishhead
117 \_id & ObjectId & true & Уникальный идентификатор \\ \hline
118 head & String & true & Заголовок новости \\ \hline
119 short & String & false & Аннотация к новости \\ \hline
120 createdAt & Date & true & Время создания новости \\ \hline
121 author & String & false & Автор новости \\ \hline
122 content & String & true & Текст новости \\ \hline
123 views & Integer & true & Количество просмотров новости зарегистрированными
    пользователями
124 \end{xltabular}
125
126 Пример использования различных типов столбцов представлен в таблице \ref{prod
    :table}. Рекомендуется использовать пакет xltabular для создания таблиц.
127
128 \begin{xltabular}{\textwidth}{|R|C{2.5cm}|l|T|}
129 \caption{Атрибуты сущности "<Новости разметки в LaTeX"> с использованием
    различных типов столбцов и многострочным заголовком\label{prod:table}}\\ \hline
130 \centrow Поле & \centrow Тип & \centrow Обязательное & \centrow Описание \\ \hline
131 \centrow 1 & \centrow 2 & \thead{3} & \centrow 4 \\ \hline
132 \endfirsthead
133 \continuecaption{Продолжение таблицы \ref{prod:table}}
134 \centrow 1 & \centrow 2 & \thead{3} & \centrow 4 \\ \hline
135 \finishhead
136 \_id & ObjectId & true & Уникальный идентификатор \\ \hline
137 head & String & true & Заголовок новости \\ \hline
138 short & String & false & Аннотация к новости \\ \hline

```

```

139   createdAt & Date & true & Время создания новости \\ \hline
140   author & String & false & Автор новости \\ \hline
141   content & String & true & Текст новости \\ \hline
142   views & Integer & true & Количество просмотров новости зарегистрированными
      пользователями
143 \end{xltabular}
144
145 В системе предусмотрен внутренний механизм связи между разделами и элементами
      информационных блоков, поэтому введения дополнительных идентификаторов
      при реализации связей между сущностями не предполагается.
146
147 Экземпляры сущностей реализуются в информационных блоках посредством
      элементов, атрибуты сущности – посредством полей и свойств элемента.

```