

Лабораторная работа №4

Тема: Построение архитектуры программного средства

Цель: Ознакомиться с процессом построения архитектуры программного средства

Оборудование: IBM PC

Программное обеспечение: Microsoft Word

СОДЕРЖАНИЕ

Задание	3
Требования к содержанию и оформлению отчета по лабораторной работе.....	3
Теоретические сведения.....	4
Пример построения архитектуры приложения.....	7
Контрольные вопросы.....	10

ЗАДАНИЕ

1. Ознакомиться с теоретическими сведениями и примерами, приведенными в задании к лабораторной работе.
2. Описать назначение системы. Проанализировать требования к функциональным характеристикам приложения.
3. Выделить основные функции, модули программы и объекты интерфейса, которые должны быть реализованы. Построить структуру элементов приложения.
4. Выбрать вид архитектуры, которая будет лежать в основе построения приложения.
5. Построить модель архитектуры, расположив на ней функциональные модули системы.
6. Написать отчет по лабораторной работе.
7. Ответить на контрольные вопросы.

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОФОРМЛЕНИЮ ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ.

Отчет по лабораторной работе должен состоять из:

1. Титульный лист.
2. Тема, цель, оборудование, ПО (то что содержится в шапке лабораторной работы).
3. Содержание.
4. Постановка задачи.
5. Список функций системы и перечень модулей программы.
6. Схема архитектуры приложения.
7. Ответы на контрольные вопросы.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Архитектура информационной системы – концепция, определяющая модель системы, структуру, выполняемые её функции и взаимосвязь между её компонентами, слоями и звеньями.

Архитектура программной системы во многом зависит от предметной области, для которой разрабатывается система. Поэтому часто архитектуры систем, разрабатываемых для одной и той же предметной области, имеют много общего.

По физическому размещению компонентов информационной системы и логическому распределению их функционала на серверах и рабочих станциях различают звенья архитектуры информационной системы.

По выполняемым функциям компоненты архитектуры ИС можно разделить на три связанных друг с другом слоя: слой представления, слой бизнес-логики и слой доступа к данным.

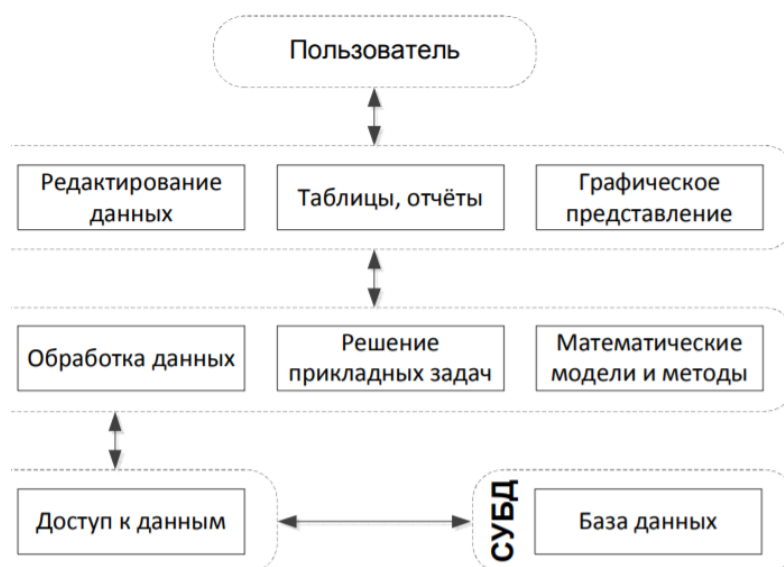


Рис. 1. Трёхслойная архитектура информационной системы

1. Слой представления представляет собой реализацию пользовательского интерфейса, в том числе – функций для организации взаимодействия с пользователем: работа с окнами и формами, нажатие кнопок, движение мыши, построение изображений, ввод данных в формах ввода, вывод данных в форме таблиц и т.д. Функции слоя представления отвечают только за визуальное представление и форматирование данных и не выполняют анализ и обработку данных, включая поиск, отбор, сортировку и т.д. Функции слоя представления оперируют понятиями и элементами пользовательского интерфейса ИС.

2. Слой бизнес-логики содержит правила и алгоритмы реакции приложения на действия

пользователя или на внутренние события, правила анализа и обработки данных. Функции слоя решают задачи управления данными на абстрактном уровне, в общем виде, без привязки к конкретной реализации пользовательского интерфейса и к конкретному способу хранения данных ИС. Функции слоя бизнес-логики оперируют концептуальными сущностями – объектами предметной области.

3. Слой доступа к данным отвечает за создание, хранение, выборку, модификацию и удаление данных, связанных с решаемой прикладной задачей. На данном слое реализуются функции, обеспечивающие взаимодействие с базой данных, с учётом специфики конкретной СУБД и выбранных технологий доступа к данным, используемых при разработке программного обеспечения ИС. Слой доступа к данным обеспечивает переход от логической структуры реляционной базы данных (таблиц) к концептуальным сущностям – объектам предметной области.

Архитектура «клиент-сервер»

В настоящее время при создании ИС широко используется двухзвенная архитектура типа «клиент-сервер».

Сервером является компьютер, на котором установлена и постоянно функционирует СУБД, которая обеспечивает непрерывный доступ к базе данных ИС с любого узла локальной сети. На сервере ИС реализован слой доступа к данным.

Клиентом является компьютер (рабочая станция) пользователя, на котором выполняются прикладные программы. Клиентские приложения реализуют функции слоя представления (пользовательский интерфейс, ввод и вывод данных) и слоя бизнес-логики (операции анализа обработки данных, формирование отчётов).

Вся бизнес-логика и все механизмы формирования пользовательского интерфейса реализованы на стороне клиента с помощью клиентского приложения, установленного на компьютере пользователя. Сервер используется только для работы СУБД и физического хранения файлов БД.

Недостатком описанной реализации клиент-серверной архитектуры является то, что сервер фактически рассматривается только как удалённая база данных, а экземпляры клиентских приложений, реализующие бизнес-логику анализа и обработки данных, выполняются на каждом отдельном компьютере клиента. Следствием этого является высокая нагрузка на локальную вычислительную сеть (ЛВС), так как для анализа и

обработки данных между клиентом и сервером БД необходимо перемещать большой объём данных.

Для сокращения нагрузки на ЛВС и централизации анализа и обработки данных рекомендуется использовать другой способ реализации клиент-серверной архитектуры, который предполагает перенос реализации слоя бизнес-логики и слоя представления на отдельный сервер ИС – сервер приложений.

В общем случае сервер слоёв бизнес-логики и представления и сервер базы данных может представлять собой два разных сервера. В таком случае происходит переход от двухзвенной архитектуры к трёхзвенной архитектуре, которая включает в себя:

- 1) клиентское приложение на рабочей станции конечного пользователя;
- 2) сервер приложений, на котором реализованы слой бизнес-логики и слой представления;
- 3) сервер базы данных, который обеспечивает функционирование СУБД и физическое хранение данных информационной системы

По сравнению с двухзвенной архитектурой трёхзвенная архитектура объединяет действия по формированию пользовательского интерфейса, анализу и обработке данных, формированию отчётов не в клиентском приложении, выполняющемся на рабочей станции пользователя, а в приложении, выполняющемся на сервере. При этом сервер БД по-прежнему решает задачи только управления информацией, хранящейся в базе данных, а также физического хранения файлов ИС.

Трёхзвенная архитектура является эталонной с точки зрения компонентного физического распределения двух различных функциональных компонентов системы (сервер приложений и сервер базы данных) по разным технологическим узлам (серверам локальной вычислительной сети). При отсутствии возможности реализовать архитектуру системы в таком виде допускается объединение функций сервера приложений и сервера базы данных на одном сервере.

ПРИМЕР ПОСТРОЕНИЯ АРХИТЕКТУРЫ ПРИЛОЖЕНИЯ.

Описать назначение системы. Проанализировать требования к функциональным характеристикам приложения. Выделить основные функции и объекты интерфейса программы, которые должны быть реализованы.

Система предназначена для автоматизации работы сотрудника отдела маркетинга малых и средних предприятий, связанной со сбором и анализом маркетинговой информации.

Программа предназначена для сбора, хранения и анализа маркетинговой информации. Для ее использования пользователи обязательно должны пройти регистрацию, введя все необходимые данные о себе.

Составим список объектов интерфейса и необходимых действий, реализуемых в программе.

Список объектов интерфейса и необходимых действий

Объект интерфейса	Действия
Главная страница web-сайта	Получить информацию о ресурсе
Форма регистрации и входа	Ввести данные о пользователе с целью последующей авторизации на сайте
Форма справки	Изучить справочную информацию
Форма ввода данных	Заполнить поля стандартизированных полей
Форма выбора метода анализа	Выбрать необходимый метод для анализа
Управляющие элементы сохранения изменений	Сохранить изменения
Перечень дружественных ресурсов	Представить список ресурсов однонаправленной тематики

Таким образом, структура приложения может выглядеть следующим образом

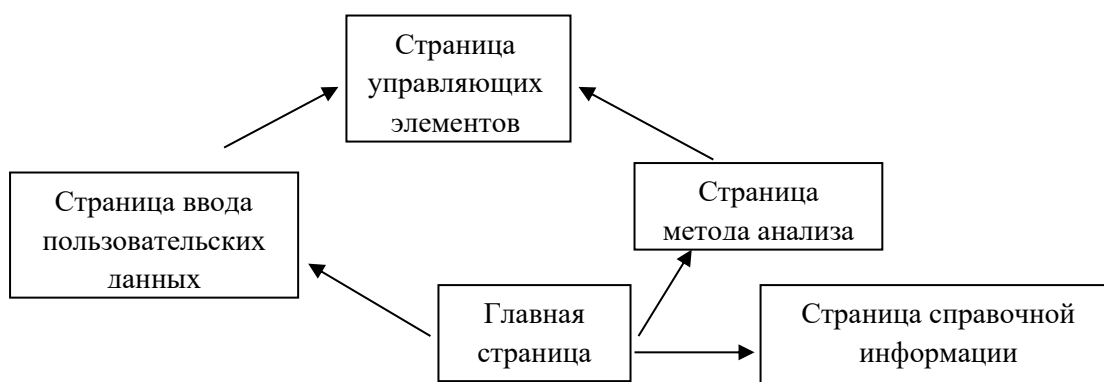


Рис.1. Структура элементов приложения

Выбрать вид архитектуры, которая будет лежать в основе построения приложения.

В основу разработки автоматизированной системы положена клиент-серверная архитектура.

Клиентская часть системы обращается по сети к серверной части. Клиентские задачи выполняет браузер, а серверные – web-сервер. Запросы пользователя интерпретируются PHP сценариями и направляются базе данных.

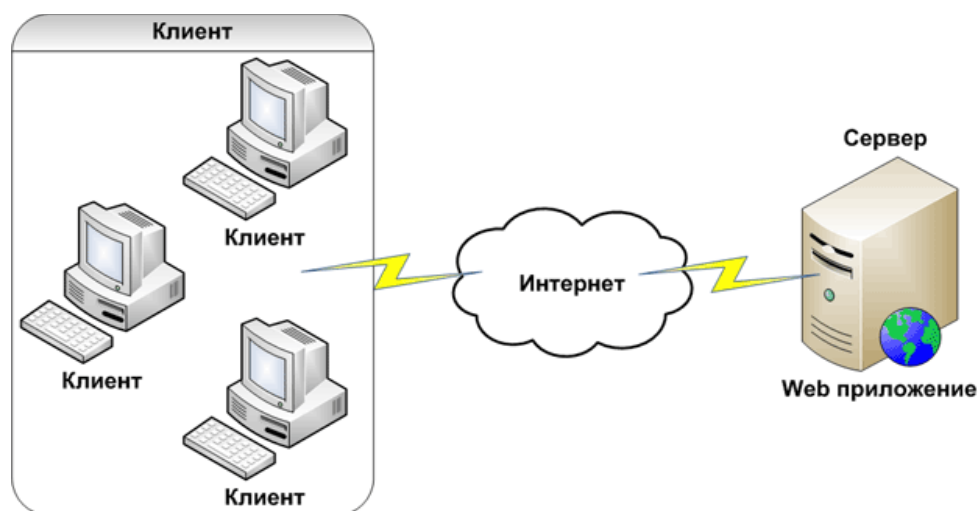


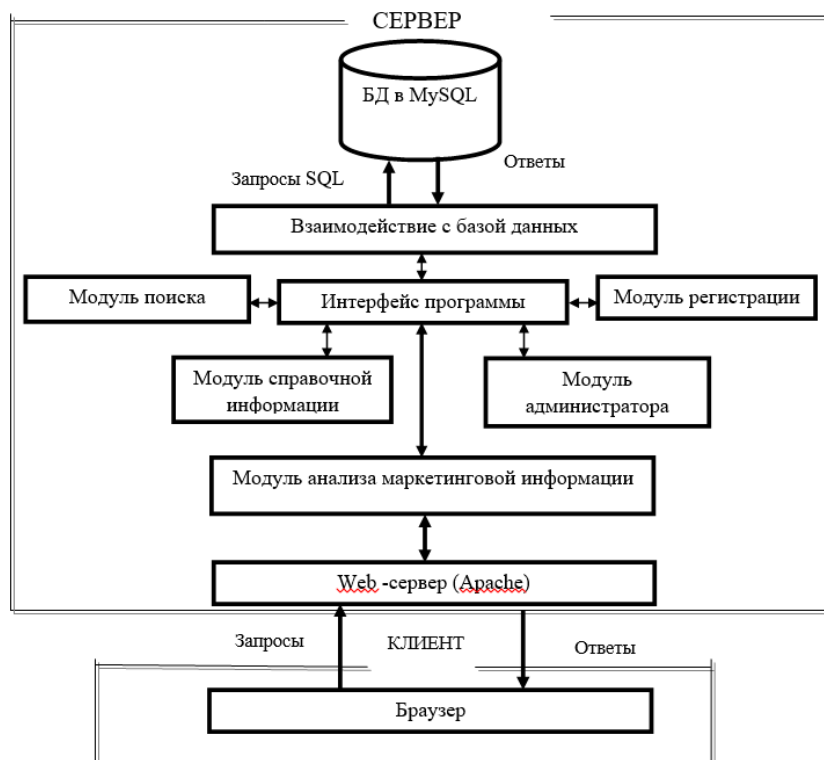
Рис.2. Клиент-серверная модель архитектуры приложений

Выделить основные функции и модули программы, которые должны быть реализованы.

Для удобства обновления и расширений программного обеспечения программу решено сделать модульной. Таким образом можно выделить следующие модули программы:

- Модуль взаимодействия с БД. В зависимости от прав доступа, предоставляет маркетинговую информацию из базы данных.
- Модуль анализа маркетинговой информации. Реализует возможность сбора, анализа, обработки и хранения маркетинговой информации.
- Модуль администратора. Помимо базовых функций, реализует возможности добавления пользователей, новых функций, отслеживать работу сервера и пользование ресурса.
- Модуль регистрации. Проверяет корректность ввода логина и пароля.
- Модуль поиска. Реализует поиск информации по полям базы данных.
- Модуль справочной информации. Предоставляет возможность получения справочной информации по методам анализа маркетинговой информации.
- Интерфейс программы.

Построить модель архитектуры, расположив на ней функциональные модули системы.



КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое архитектура приложения.
2. На какие компоненты можно разделить архитектуру ИС по выполняемым функциям?
3. Что подразумевает архитектура «Клиент-сервер»?
4. Что представляет собой двухзвенная и трехзвенная архитектура «Клиент-сервер»?