Внедрение и поддержка компьютерных систем.

Лабораторная работа №6. Васильев Максим 4438.

Проектирование архитектуры программной системы.

Архитектура программного обеспечения — совокупность важнейших решений об организации программной системы. Архитектура включает:

* выбор структурных элементов и их интерфейсов, с помощью которых составлена система, а также их поведения в рамках сотрудничества структурных элементов;
* соединение выбранных элементов структуры и поведения во всё более крупные системы;
* архитектурный стиль, который направляет всю организацию — все элементы, их интерфейсы, их сотрудничество и их соединение.

[Документирование](https://ru.wikipedia.org/wiki/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F_%D0%BD%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) архитектуры [программного обеспечения](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) (ПО) упрощает процесс коммуникации между разработчиками, позволяет зафиксировать принятые проектные решения и предоставить информацию о них эксплуатационному персоналу системы, [повторно использовать](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B2%D1%82%D0%BE%D1%80%D0%BD%D0%BE%D0%B5_%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BA%D0%BE%D0%B4%D0%B0) [компоненты](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D0%BD%D0%B0%D1%8F_%D0%B0%D1%80%D1%85%D0%B8%D1%82%D0%B5%D0%BA%D1%82%D1%83%D1%80%D0%B0) и [шаблоны](https://ru.wikipedia.org/wiki/%D0%A8%D0%B0%D0%B1%D0%BB%D0%BE%D0%BD_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F) проекта в других.

Жизненный цикл программы состоит из этапа разработки программы и этапа её эксплуатации и сопровождения.

**Технология разработки ПО** – совокупность приёмов, позволяющих создать безошибочную программу в течение заданного времени. Состоит из четырёх этапов:

* Формулировка задачи на естественном языке и создание математической модели. Целью данной разработки является создание функционирующих модулей учета товара, прибыли и выручки, предназначенных для благоприятного введения бизнеса предпринимателем.
* Разработка нового или выбор существующего метода численного решения математической задачи (алгоритма). Разработка нового приложения в дальнейшем способствует уменьшению трудоемкости работы предпринимателя.
* Написание программы на языке программирования. Разрабатываемая система будет реализована при помощи языка программирования C#.
* Тестирование и отладка программ.

Технология программирования – это совокупность методов и средств, используемых в процессе разработки ПО.

Существует различное множество подходов программирования. Для реализации системы товарного учета выберем объектный подход.

Объектно-ориентированное программирование (ООП) определяется как технология создания сложного ПО, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса. Классы образуют иерархию с наследованием свойств. Взаимодействие программных объектов в такой системе осуществляется путем передачи сообщений.

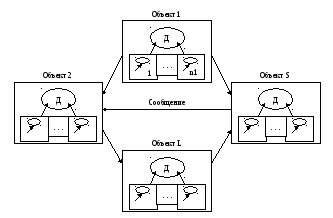


Рисунок 1 – Структура программы при ООП.

Для разработки desktop приложения, выбрана платформа WPF - система для построения клиентских приложений Windows с визуально привлекательными возможностями взаимодействия с пользователем, графическая подсистема в составе .NET Framework, использующая язык XAML.

Преимущества платформы WPF:

* Широкая интеграция;
* Независимость от разрешения экрана;
* Аппаратное ускорение;
* Декларативное программирование;
* Богатые возможности композиции и настройки;

Для реализации всех главных потребностей системы, необходимо разработать и реализовать связь программного обеспечения с базой данных. Чтобы каждый пользователь мог с легкостью использовать разрабатываемую систему, нужно осуществить взаимодействие клиента и базы данных через сервер. Тогда конечному клиенту (предпринимателю) придется лишь только установить программу на рабочий компьютер и авторизоваться в системе, без лишних действий по развертыванию локальной базы данных.

На основе описанных особенностей, наиболее подходящей для реализации данной системы будет клиент-серверная архитектура.

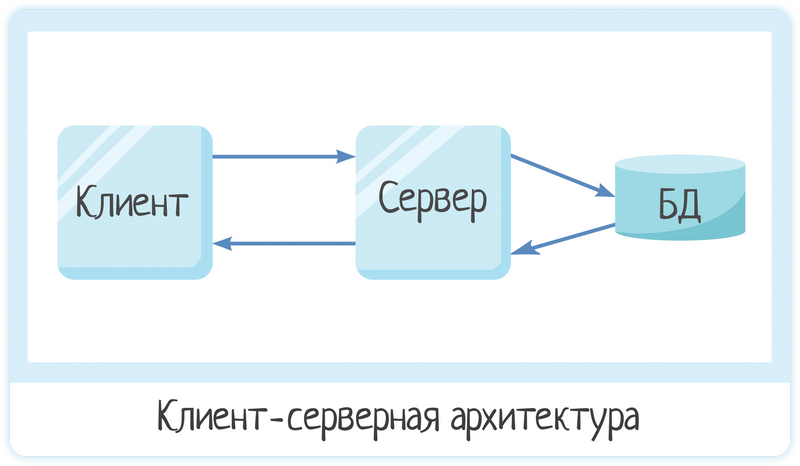


Рисунок 2 – Архитектура Клиент-сервер.

В качестве клиента в клиент-серверной архитектуре выступает конечный пользователь, он использует систему и выполняет различные действия, которые представляют собой запросы, отправляемые на сервер. Сервер принимает все запросы клиента и отвечает на них, обращаясь к базе данных, которая находится на нем, тем самым сервер является некоторым посредником между клиентом и базой данных. Благодаря такому подходу, систему могут использовать сразу большое количество пользователей, а ее быстродействие никак на этом не скажется.

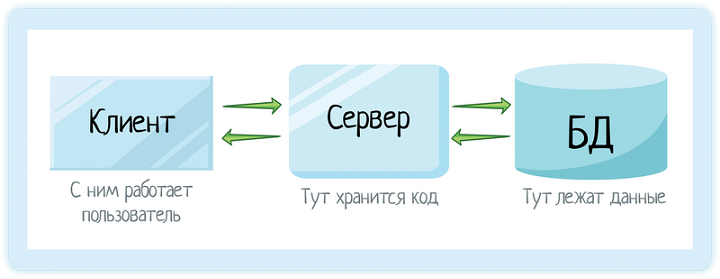


Рисунок 3 – Взаимодействия компонентов архитектуры.

Для исполнения компонентов системы определим набор инструментальных средств:

* IDE (Интегрированная среда разработки) Visual Studio - Это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Помимо стандартного редактора и отладчика, которые есть в большинстве сред IDE, Visual Studio включает в себя компиляторы, средства автоматического завершения кода, графические конструкторы и многие другие функции для улучшения процесса разработки.
* СУБД (Система управления базами данных) PhpMyAdmin - это программный инструмент, написанный на [PHP](https://php.net/), предназначенный для администрирования [MySQL](https://www.mysql.com/) через Интернет. PhpMyAdmin поддерживает широкий спектр операций с MySQL. Часто используемые операции могут выполняться через пользовательский интерфейс.
* Сервер MAMP – это комплекс серверного программного обеспечения для Windows.
* Figma - Онлайн-сервис для разработки интерфейсов и прототипирования с возможностью организации совместной работы в режиме реального времени.

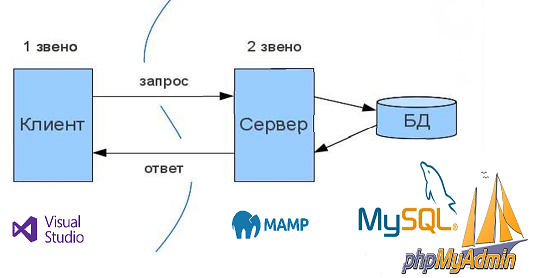


Рисунок 4 – Проектирование Архитектуры системы.

Для реализации сетевого взаимодействия клиента и сервера будет использоваться MySQL Connector/ODBC.

ODBC (Open Database Connectivity) – программный интерфейс прикладного программирования (API) для доступа к системам управления базами данных (СУБД).

С помощью ODBC или открытого механизма взаимодействия с базами данных можно разрабатывать приложения для использования одного интерфейса доступа к данным, не беспокоясь о тонкостях взаимодействия с несколькими источниками.

Это достигается благодаря тому, что поставщики различных баз данных создают драйверы, реализующие конкретное наполнение стандартных функций из ODBC API с учётом особенностей их продукта.

MySQL Connector – это название семейства драйверов MySQL ODBC, которые обеспечивают доступ к базе данных MySQL с использованием стандартного API ODBC.