**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КАЛУЖСКОЙ ОБЛАСТИ**

**ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ профессиональное образовательное учреждение Калужской области «Калужский ТЕХНИЧЕСКИЙ КОЛЛЕДЖ»**

**ГАПОУ КО «КТК»**

**ПРАКТИЧЕСКАЯ РАБОТА №\_\_**

**Тема: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

по **МДК.05.01 «Проектирование и дизайн ИС»**

специальность: **09.02.07 «Информационные системы и программирование»**

**I курс**

Выполнил:

студент группы 1ИСиП4

Поповичев Д.С.

Подпись\_\_\_\_\_\_\_\_\_\_\_

Руководитель:

Метаки Л.Г.

Дата защиты\_\_\_\_\_\_\_

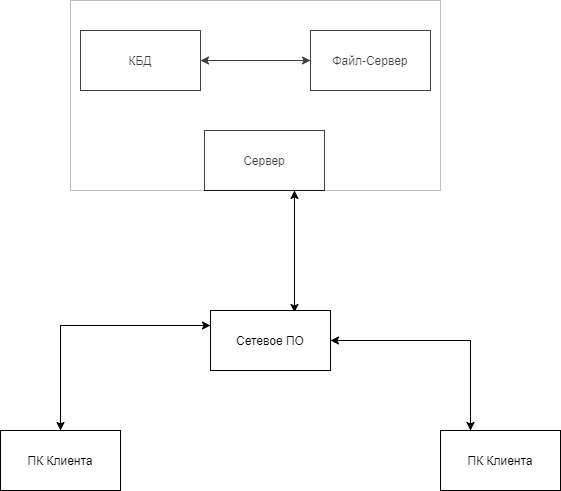
Оценка\_\_\_\_\_\_\_\_\_\_\_\_

Подпись\_\_\_\_\_\_\_\_\_\_\_

**Калуга, 2020 год**

**Цель работы:** получение навыков разработки подели архитектуры ИС

**Файл-серверная архитектура**

Файл-серверные приложения — приложения, схожие по своей структуре с локальными приложениями и использующие сетевой ресурс для хранения данных в виде отдельных файлов. Функции сервера в таком случае обычно ограничиваются хранением данных (возможно также хранение исполняемых файлов), а обработка данных происходит исключительно на стороне клиента. Количество клиентов ограничено десятками ввиду невозможности одновременного доступа на запись к одному файлу. Однако клиентов может быть значительно больше, если они обращаются к файлам исключительно в режиме чтения.****

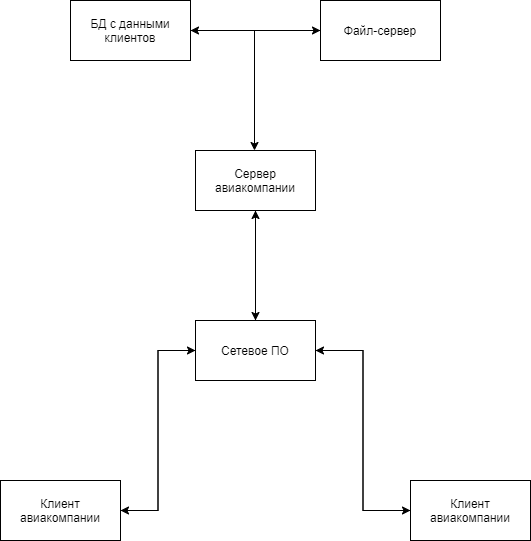
Достоинства:

* низкая стоимость разработки;
* невысокая стоимость обновления и изменения ПО.

Недостатки:

* рост числа клиентов резко увеличивает объём трафика и нагрузку на сети передачи данных;
* высокие затраты на модернизацию и сопровождение сервисов бизнес-логики на каждой клиентской рабочей станции;
* низкая надёжность системы.

**Задание 1**

****

**Клиент-серверная**

**«Клиент — сервер»** (англ. *client–server*) — вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами.

Преимущества:

* Отсутствие дублирования кода программы-сервера программами-клиентами.
* Так как все вычисления выполняются на сервере, то требования к компьютерам, на которых установлен клиент, снижаются.
* Все данные хранятся на сервере, который, как правило, защищён гораздо лучше большинства клиентов. На сервере проще организовать контроль полномочий, чтобы разрешать доступ к данным только клиентам с соответствующими правами доступа.

Недостатки

* Неработоспособность сервера может сделать неработоспособной всю вычислительную сеть. Неработоспособным сервером следует считать сервер, производительности которого не хватает на обслуживание всех клиентов, а также сервер, находящийся на ремонте, профилактике и т. п.
* Поддержка работы данной системы требует отдельного специалиста — [системного администратора](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B0%D0%B4%D0%BC%D0%B8%D0%BD%D0%B8%D1%81%D1%82%D1%80%D0%B0%D1%82%D0%BE%D1%80).
* Высокая стоимость оборудования.

**Задание 2**

****

**Трёхуровневая архитектура**

[архитектурная модель](https://ru.wikipedia.org/wiki/%D0%90%D1%80%D1%85%D0%B8%D1%82%D0%B5%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D1%8B) программного комплекса, предполагающая наличие в нём трёх компонентов: [клиента](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B8%D0%BA%D0%BB%D0%B0%D0%B4%D0%BD%D0%BE%D0%B5_%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), [сервера приложений](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) (к которому подключено клиентское приложение) и [сервера баз данных](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%80%D0%B2%D0%B5%D1%80_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) (с которым работает сервер приложений).

****

По сравнению с двухзвенной [клиент-серверной](https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B8%D0%B5%D0%BD%D1%82-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) архитектурой или [файл-серверной](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80) архитектурой трёхуровневая архитектура обеспечивает, как правило, большую [масштабируемость](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%81%D1%88%D1%82%D0%B0%D0%B1%D0%B8%D1%80%D1%83%D0%B5%D0%BC%D0%BE%D1%81%D1%82%D1%8C) (за счёт горизонтальной масштабируемости сервера приложений и мультиплексирования соединений), большую конфигурируемость (за счёт изолированности уровней друг от друга). Реализация приложений, доступных из [веб-браузера](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%B1%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80) или из [тонкого клиента](https://ru.wikipedia.org/wiki/%D0%A2%D0%BE%D0%BD%D0%BA%D0%B8%D0%B9_%D0%BA%D0%BB%D0%B8%D0%B5%D0%BD%D1%82), как правило, подразумевает развёртывание программного комплекса в трёхуровневой архитектуре. При этом обычно разработка трёхзвенных программных комплексов сложнее, чем для двухзвенных, также наличие дополнительного связующего программного обеспечения может налагать дополнительные издержки в администрировании таких комплексов.

**Задание 3**

