Архитектура – базовая организация системы, воплощенная в ее компонентах, … и окружении, а также принципы, определяющие проектирование и развитие системы.

**ЛЕКЦИЯ 2**

**Разновидности архитектуры информационных систем**

1)Клиент-сервер

Распределенная система, состоящая из клиента и сервера и соединяющей сети

Описывает отношения между клиентом и 1 и более серверами, где клиент инициирует 1 и более запросов, ожидает ответы и обрабатывает их при получении

1.1)клиент очередь клиент

1.2)Peer-to-Peer(torrent)

Позволяет обмениваться ролями для распределения

1.3)Серверы приложений

Приложения на сервере

Тонкий клиент доступ через браузер или спец приложение

3+)простота обслуживания

2) N- уровневая архитектура

Стиль развертывания пр- ний, описывающий разделение функциональностей на сегменты, физически расположенные на разных узлах сети.

Узлы называют уровнями.

Для увеличения надежности и производительности сети

Рек к исп 1)когда требования по разработке включают перекос распределения ресурсов(на одной стадии обработки нагрузка выше, чем на других)

2) разные требования по безопасности к отдельным уровням системы

3)нужно обеспечивать возможность совместного использования либо бизнес логики либо данных разными приложениями и имеется достаточное количество оборудования

3)Объектно-ориентированная архитектура

— Парадигма проектирования Основанная на разделении ответственности приложения на самостоятельные пригодные для повторного использования объекты, каждая из которых содержит данные и их поведение

Объекты обособленны, независимы и слабо связаны

Обмен данными между объектами происходит через интерфейсы путем вызова методов и свойств др объектов и отправки/приема сообщений

Принципы:

Абстракция – сложная операция в обобщение, сохраняющая основные х-ки(сложное описываем как совокупность простого и доводим до некоего интерфейса)

Композиция – объект может состоять из других объектов

Наследование

Полиморфизм – переопределение базового типа

Исп когда есть объекты или их легко получить; ЯП на ООП;

**Лекция 3**

*Компонентная архитектура*

Множество удаленных компонентов

Преимущества компонентной архитектуры:

1. Простота развертывания
2. Меньшая стоимость
3. Простота разработки
4. Возможность повторного использования

Применять архитектуру, если

1)уже имеются похожие компоненты

2) можем получить готовые компоненты откуда либо

Подойдет для создания дополнений приложений

-:

1) снижаются показатели собственной эффективно работы за счет использования универсальных компонентов

*Многослойная архитектура*

Обеспечивает группировку связанной функциональности в разных слоях, выстраиваемых вертикально поверх друг друга. Функциональность каждого слоя объединена своей ролью, слои слабо связаны и между ними ос-ся явный обмен данными.

Представляет собой некоторую пирамиду

Пользователь видит только верхний слой пирамиды(пример mvc)

Рек. К исп.

Уже есть какой то слой

*Архитектура, основанная на шине сообщений*

Описывает принципе исп-ния программной системы, которая может принимать и отправлять сообщения по одному или более каналам связи, обеспечивая приложениям возможность взаимодействия без необходимости знания конкретных деталей друг о друге.

Вз-е ос-ся путем передачи сообщений через общую шину

Шина предприятий

Шина интернет

Преимущества:

1. Расширяемость( добавлять и удалить приложения к шине
2. Невысокая сложность
3. Гибкость
4. Слабое связывание
5. Возможность подключения к шине множества экземпляров одного и тоже …

Применяется, если имеется существующее приложение, выполняющее задачи путем взаимодействия друг с другом

Сервис ориентированная архитектура

Обеспечивает возможность предоставлять функциональность приложения в виде набора сервисов и создавать приложения, использующие программные сервисы

Сервисы слабо связаны между собой, тк используют основанные на стандартах интерфейсы. МБ вызваны, опубликованы и обнаружены

Осн задача сервисов – предоставление схемы и вз-я с приложением посредством приложений через интерфейсы

**Лекция 4**

XML-RPC

Xmlrpc.scripting.com

c-sharpcorner.com/article/an-xml-rpc-package-for-net

**Клиент серверная архитектура**

Файл сервер(толстый клиент)

Сохраняется автономность ПО на каждом клиенте