Государственное бюджетное профессиональное образовательное учреждение Республики Марий Эл

«ЙОШКАР-ОЛИНСКИЙ ТЕХНОЛОГИЧЕСКИЙ КОЛЛЕДЖ»

|  |
| --- |
| Специальность: 09.02.07 Информационные системы и программирование  Группа: А-31  Дисциплина: Технология разработки ПО |

ОТЧЕТ ПО

«Диаграмма компонентов»

Руководитель: Пинешкин Ю.С

Выполнил(а): студентки А-31

Николаева С.В, Мочалова К.Е.

Йошкар-Ола

2025

**Предметная область**

**Некоторые примеры компонентов для определённой предметной области**: клиентское приложение, сервер приложения, база данных, платёжный шлюз в случае интернет-магазина.

**Диаграмма компонентов** — это структурная диаграмма языка унифицированного моделирования (UML), которая показывает разбиение программной системы на структурные компоненты и связи (зависимости) между ними.

**Некоторые цели создания таких диаграмм**:

* визуализация общей структуры исходного кода программной системы;
* спецификация исполнимого варианта программной системы;
* обеспечение многократного использования отдельных фрагментов программного кода;
* представление концептуальной и физической схем баз данных.



Пример диаграммы компонентов



Пример диаграммы компонентов

**Вывод:** Диаграмма компонентов применяется в разработке программного обеспечения для визуального представления структуры системы, демонстрации её различных компонентов и их взаимодействий.

Некоторые области применения диаграммы компонентов:

* Проектирование и архитектура системы. Отображая части (компоненты), их соединения и любые зависимости между ними, эти диаграммы помогают архитекторам и дизайнерам понять структуру системы.
* Анализ требований. Диаграммы помогают клиентам и разработчикам понять функциональные (что должна выполнять система) и нефункциональные (производительность, безопасность и т. д.) требования к системе.
* Системная документация. Диаграммы на основе компонентов служат записями о том, как построена система, отражая общую картину проектирования и архитектурные решения для дальнейшего использования.
* Разработка программного обеспечения. Эти диаграммы служат руководством для разработчиков на этапе сборки, чётко очерчивая границы компонентов и то, как должны взаимодействовать различные части программного обеспечения.
* Генерация и реализация кода. Иногда эти диаграммы могут быть отправной точкой для автоматической генерации кода, ускоряя процесс создания программных компонентов.
* Обслуживание и эволюция системы. По мере роста или изменений системы эти диаграммы полезны для понимания текущей архитектуры, делая обновления более простыми и организованными.

|  |  |
| --- | --- |
| **Плюсы** | **Минусы** |
| * предоставляют общую картину архитектуры системы; * помогают разработчикам и архитекторам лучше понять структуру и взаимосвязи; * являются полезным инструментом для коммуникации и документирования архитектурных решений; * позволяют получить высокоуровневое представление о компонентах системы; * могут описывать программные системы, реализованные на любом языке программирования или стиле. | * на первый взгляд могут показаться сложными; * упрощённая модель, не включает некоторые детали, например аккаунты пользователей. |