

Федеральное государственное образовательное бюджетное учреждение
высшего образования
**«Финансовый университет при Правительстве
Российской Федерации»
(Финансовый университет)**

Колледж информатики и программирования

ПМ.08 Разработка кода
информационных систем

Группа: ЗИСИП-622

УТВЕРЖДАЮ

Председатель предметно-цикловой
комиссии информационных систем
и программирования

_____ Т.Г. Аксёнова

« ____ » _____ 2025 г.

ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №4

Построение диаграммы компонентов и генерация кода

Преподаватель

_____ Р. Р. Абзалимов

Исполнитель

_____ Л. Д. Слепцов

Оценка: _____

« ____ » _____ 2025 г.

Москва

2025

Цель работы:

Ознакомиться с методологией моделирования информационных систем на основе языка UML.

Тема: Охота.

Ход работы:

Диаграмма компонентов — это один из типов диаграмм в языке моделирования UML, который используется для визуализации структуры системы на уровне компонентов. Она показывает, как система разделена на отдельные модули (компоненты), как эти компоненты взаимодействуют друг с другом и какие интерфейсы они предоставляют или используют.

На рисунке 1 представлена диаграмма компонентов корзины интернет-магазина для охоты HuntPlanet.

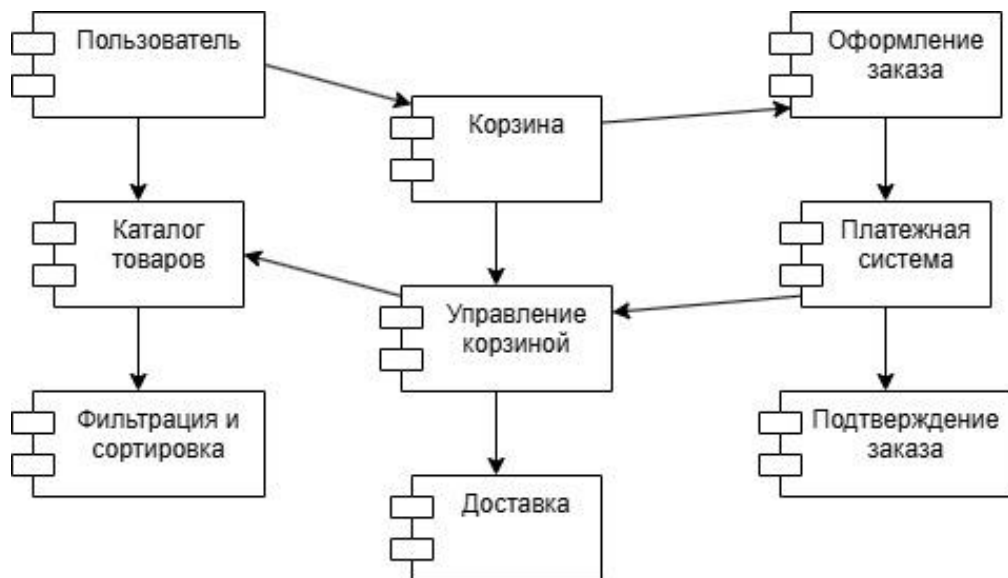


Рисунок 1 –диаграмма компонентов корзины интернет-магазина для охоты HuntPlanet

Вопросы:

1. Основные цели и задачи диаграммы компонентов в UML:

- Диаграмма компонентов используется для визуализации структуры системы на уровне компонентов.
- Она помогает определить, как компоненты системы взаимодействуют друг с другом.

- Позволяет показать модульность системы и разделение ответственности между компонентами.
- Используется для проектирования и документирования архитектуры системы.

2. Элементы диаграммы компонентов:

- **Компоненты** — основные строительные блоки системы, представляющие собой модули или части системы (например, библиотеки, классы, сервисы).
- **Интерфейсы** — определяют точки взаимодействия между компонентами.
- **Зависимости** — показывают, как один компонент зависит от другого.
- **Порты** — точки подключения компонентов к внешним интерфейсам.
- **Соединители** — линии, которые связывают компоненты через интерфейсы или порты.

3. Определение зависимостей между компонентами:

- Зависимости обозначаются пунктирной стрелкой, направленной от зависимого компонента к компоненту, от которого он зависит.
- Например, если компонент А использует функциональность компонента В, то стрелка будет направлена от А к В.

4. Использование диаграммы компонентов для архитектурного проектирования:

- Диаграмма компонентов помогает разделить систему на модули, что упрощает проектирование и разработку.
- Она позволяет определить ключевые компоненты системы и их взаимодействие.
- Используется для анализа и оптимизации архитектуры системы, а также для документирования решений.

5. Генерация кода на основе диаграммы компонентов:

- Современные инструменты UML (например, Enterprise Architect, Visual Paradigm, IBM Rational Software Architect) поддерживают генерацию кода на основе диаграмм компонентов.
- Код генерируется в соответствии с определенными компонентами, интерфейсами и зависимостями.
- Например, для каждого компонента может быть создан класс или модуль, а зависимости преобразуются в вызовы методов или импорты.

6. Преимущества генерации кода на основе диаграмм UML:

- Ускорение процесса разработки за счет автоматизации.
- Снижение количества ошибок, так как код генерируется на основе проверенных моделей.
- Упрощение поддержки и модификации системы благодаря четкой документации и структуре.
- Улучшение понимания архитектуры системы разработчиками.

7. Определение необходимых компонентов и их связей:

- На основе анализа требований к системе выделяются ключевые функциональные блоки.
- Определяются зависимости между этими блоками (например, какие компоненты используют другие).
- Используются интерфейсы для описания взаимодействия между компонентами.

8. Роль диаграммы компонентов в понимании архитектуры и модульности:

- Диаграмма компонентов наглядно показывает, как система разделена на модули.
- Она помогает понять, какие компоненты отвечают за конкретные функции.
- Упрощает анализ и рефакторинг системы, так как зависимости и взаимодействия четко определены.

9. Оценка важности генерации кода на основе диаграмм

компонентов:

- Генерация кода на основе диаграмм компонентов значительно ускоряет процесс разработки, особенно для крупных проектов.
- Она позволяет сосредоточиться на проектировании архитектуры, а не на рутинном написании кода.
- Это особенно полезно в agile-подходах, где важно быстро создавать прототипы и итеративно улучшать систему.

10. Основные компоненты в проекте и их взаимодействие:

- В проекте могут быть идентифицированы такие компоненты, как:
 - **Пользовательский интерфейс (UI).**
 - **Бизнес-логика** (сервисный слой).
 - **Хранение данных** (база данных или репозиторий).
 - **Внешние API** (интеграция с другими системами).
- На диаграмме компонентов эти элементы связаны через интерфейсы и зависимости. Например, UI зависит от сервисного слоя, а сервисный слой зависит от репозитория данных.

Вывод

Данная работа помогла глубже понять как проектирование UML-диаграмм способствует разработке эффективных и понятных систем, и как они могут быть использованы для оптимизации процесса управления корзиной в интернет-магазине.