Изображение выглядит как символ, логотип, круг, Графика

Контент, сгенерированный ИИ, может содержать ошибки.МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Пензенский государственный технологический университет»

(ПензГТУ)

Колледж технологический ПензГТУ

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 8

на тему «РАЗРАБОТКА ДИАГРАММЫ КОМПОНЕНТОВ».

Выполнил: студент гр. 23ИП3т

Сухов М. А.

Проверил: преподаватель   
 Рябчиков Р.В.

Пенза, 2025г

Цель работы: научиться разрабатывать диаграмму компонентов,

проектировать и визуализировать структуры программной системы на

уровне компонентов, научиться выделять ключевые модули, их

взаимодействие через интерфейсы, а также понимать роль компонентов в

архитектуре приложения.

Ход работы:

1) изучить краткие теоретические сведения;

2) ответить на контрольные вопросы;

3) выполнить практическое задание;

4) оформить отчет.

Индивидуальный вариант предметной области:

6. Система Автобаза. Диспетчер распределяет Заявки на Рейсы между

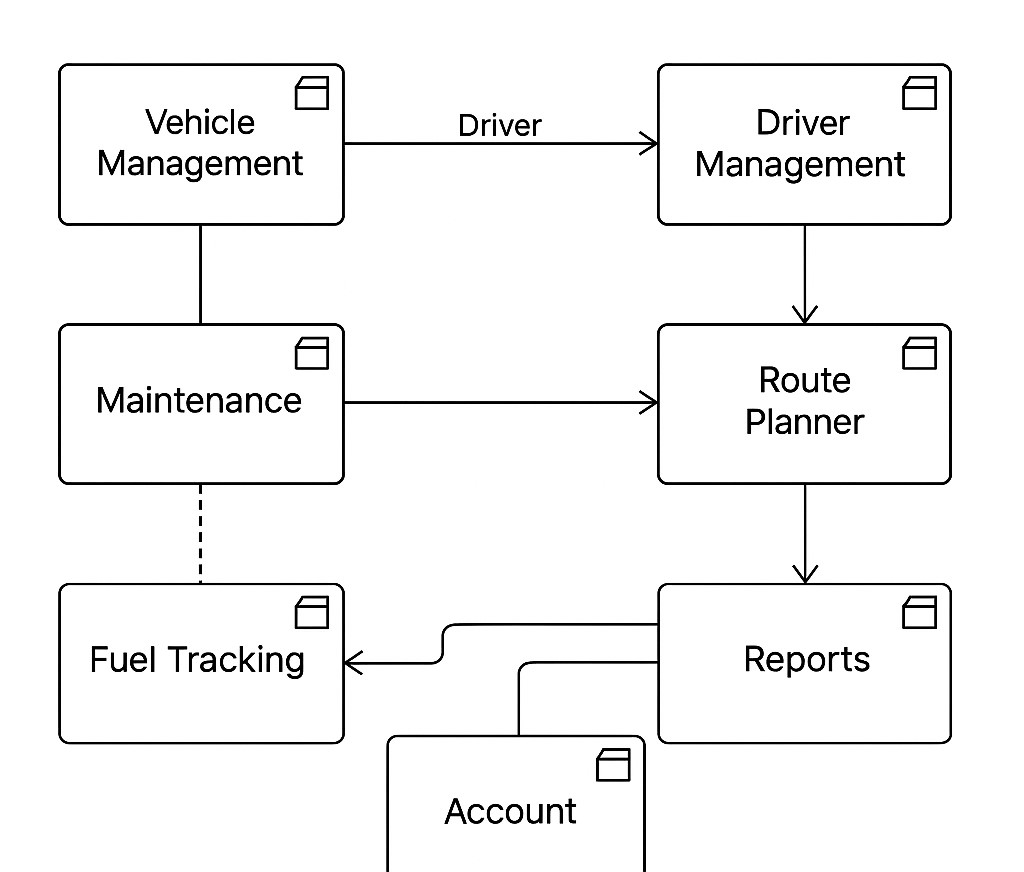
Водителями, за каждым из которых закреплен свой Автомобиль. На Рейс

может быть назначен Автомобиль, находящийся в исправном состоянии и

характеристики которого соответствуют Заявке. Водитель делает отметку о

выполнении Рейса и состоянии Автомобиля.

Диаграмма компонентов



Контрольные вопросы:

1. Какие цели преследуются при разработке диаграммы компонентов?  
Диаграмма компонентов создаётся с целью:

* Визуализировать архитектуру системы на уровне компонентов.
* Определить структуру программных модулей и их взаимодействие.
* Обеспечить прозрачность при разработке, тестировании и сопровождении системы.
* Способствовать раздельной разработке и повторному использованию модулей.
* Показывать зависимости и интерфейсы между частями системы.

2. Чем отличается «предоставляемый интерфейс» от «требуемого интерфейса»?

* Предоставляемый интерфейс (provided interface) — это интерфейс, который компонент реализует и делает доступным другим компонентам. Он показывает, что компонент умеет делать.
* Требуемый интерфейс (required interface) — это интерфейс, необходимый компоненту для функционирования. Он обозначает, что компоненту нужно от других.

Например:  
Компонент «Обработка платежей» может предоставлять интерфейс processPayment() и требовать интерфейс validateAccount() от компонента учёта.

3. Какие типы зависимостей между компонентами существуют? Приведите примеры.

Основные типы зависимостей:

* Статическая зависимость — компонент использует другой на этапе компиляции.  
  *Пример: модуль A импортирует библиотеку B.*
* Динамическая зависимость — взаимодействие происходит во время выполнения.  
  *Пример: модуль вызывает внешнюю службу через API.*
* Зависимость через интерфейс — один компонент использует интерфейс, предоставляемый другим.  
  *Пример: модуль отчётов использует интерфейс базы данных для получения информации.*

4. Почему при проектировании диаграммы компонентов важно минимизировать связи между исполняемыми модулями?

Минимизация связей важна, потому что:

* Уменьшается сопряжённость компонентов — это облегчает изменение и сопровождение системы.
* Повышается модульность и гибкость архитектуры.
* Упрощается тестирование и повторное использование компонентов.
* Снижается риск каскадных ошибок при изменении одного из компонентов.

Чем меньше зависимостей — тем проще масштабировать, заменять и обновлять отдельные части системы.

Вывод: Я научился разрабатывать диаграмму компонентов,

проектировать и визуализировать структуры программной системы на

уровне компонентов, научиться выделять ключевые модули, их

взаимодействие через интерфейсы, а также понимать роль компонентов в

архитектуре приложения.