Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Кубанский государственный технологический университет»

Кафедра информационных систем и программирования

**Развертывание и интеграция программных систем**

Отчет по лабораторной работе №9

**SOA-интеграция. Проектирование корпоративной ИС с применением SOA**

Выполнил:

студент 4 курса

группы 21-КБ-ПР1

Агниев С.В.

Проверил:

проф.кафедры ИСП

Зайков В.П.

Краснодар

2025

**Цель лабораторной работы:**

- Разработать компонентную бизнес-модель на основе сервис-ориентированной архитектуры информационной системы предприятия;

- Определить состав сервисов и выполнить их программную

реализацию.

**Вариант** **№01:** Информационная система предприятия оптово-розничной торговли комплектующими монтажных систем и отопления.

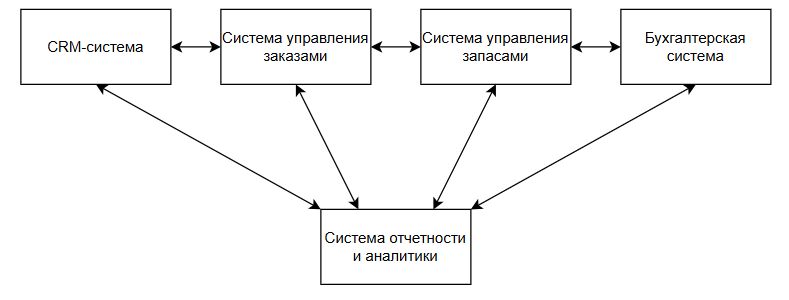
**Выполнение работы**

1. Определить основные (программные) компоненты информационной системы для своего варианта предприятия (лабораторная работа № 7).

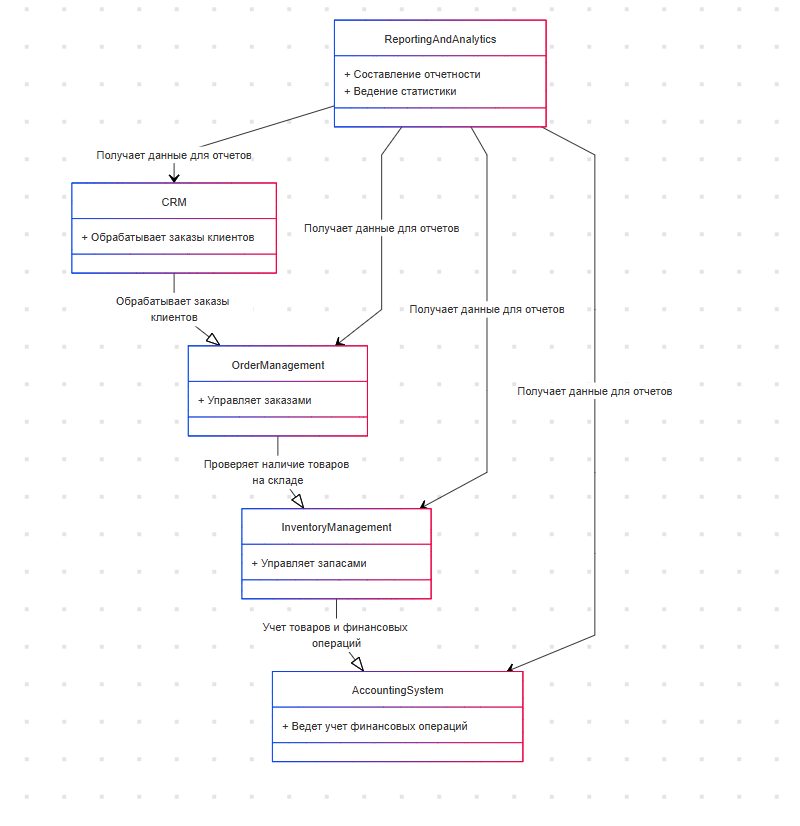
Изобразить компоненты условной схемой, с отражением связи между ними. Число компонентов: от 3 до 5.

Для предприятия оптово-розничной торговли комплектующими монтажных систем и отопления можно выделить следующие основные программные компоненты:

* Система управления запасами (Inventory Management System) – отвечает за учет и управление товарными запасами на складе.
* CRM-система (Customer Relationship Management) – управляет взаимоотношениями с клиентами, учет заказов, продаж и обратной связи.
* Бухгалтерская система (Accounting System) – ведение бухгалтерского учета, расчеты с поставщиками и клиентами.
* Система управления заказами (Order Management System) – обработка заказов от клиентов, управление поставками и отгрузками.
* Система отчетности и аналитики (Reporting and Analytics System) – формирование отчетов и аналитических данных для принятия управленческих решений.



2. Отобразить модель полученной системы с помощью диаграммы компонентов на языке UML.



3. Для каждого компонента определить группировки сервисов.

Для каждого компонента можно выделить следующие группировки сервисов:

CRM-система:

* Сервисы для управления клиентами (добавление, редактирование, удаление клиентов).
* Сервисы для обработки заказов (создание, изменение, отмена заказов).
* Сервисы для анализа продаж (отчеты по продажам, анализ клиентской базы).

Система управления заказами:

* Сервисы для обработки заказов (создание, изменение, отмена заказов).
* Сервисы для управления поставками (планирование поставок, отслеживание статуса поставок).
* Сервисы для интеграции с CRM и системой управления запасами.

Система управления запасами:

* Сервисы для учета товаров (добавление, редактирование, удаление товаров).
* Сервисы для управления складскими операциями (приемка, отгрузка, перемещение товаров).
* Сервисы для интеграции с бухгалтерской системой.

Бухгалтерская система:

* Сервисы для учета финансовых операций (учет доходов и расходов).
* Сервисы для расчета налогов и формирования отчетов.
* Сервисы для интеграции с системой управления запасами.

Система отчетности и аналитики:

* Сервисы для формирования отчетов (финансовые отчеты, отчеты по продажам).
* Сервисы для анализа данных (анализ продаж, анализ запасов).
* Сервисы для интеграции с другими компонентами.

4. Определить состав сервисов по каждой группировке:

Сервисы для бизнес-инноваций и оптимизации процессов:

* Сервис анализа продаж (CRM-система).
* Сервис прогнозирования спроса (Система управления запасами).
* Сервис оптимизации заказов (Система управления заказами).

Инфраструктурные сервисы:

* Сервис управления пользователями (CRM-система).
* Сервис управления правами доступа (Бухгалтерская система).
* Сервис интеграции данных (Система отчетности и аналитики).

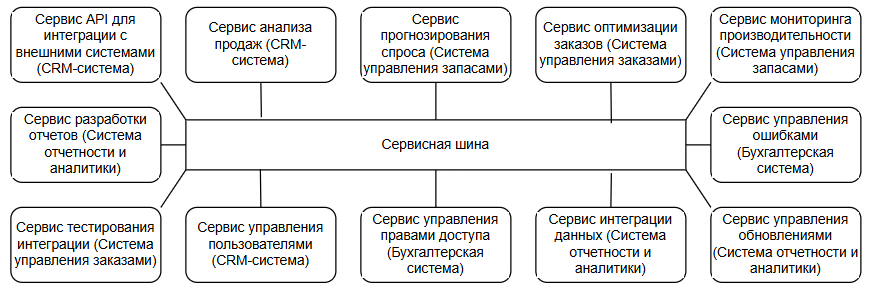
Сервисы для разработки:

* Сервис API для интеграции с внешними системами (CRM-система).
* Сервис разработки отчетов (Система отчетности и аналитики).
* Сервис тестирования интеграции (Система управления заказами).

Управляющие сервисы:

* Сервис мониторинга производительности (Система управления запасами).
* Сервис управления ошибками (Бухгалтерская система).
* Сервис управления обновлениями (Система отчетности и аналитики).

5. Построить модель SOA, взяв за основу рис.9.1, указав в ней выделенные в предыдущем пункте сервисы.



6. Cформировать каталог сервисов. Результат отобразить скриншотом.

|  |  |  |  |
| --- | --- | --- | --- |
| **Название сервиса** | **Описание** | **Группа** | **Компонент** |
| Сервис анализа продаж | Анализ данных о продажах | Сервисы для бизнес-инноваций и оптимизации процессов | CRM-система |
| Сервис прогнозирования спроса | Прогнозирование будущего спроса | Сервисы для бизнес-инноваций и оптимизации процессов | Система управления запасами |
| Сервис оптимизации заказов | Оптимизация процесса обработки заказов | Сервисы для бизнес-инноваций и оптимизации процессов | Система управления заказами |
| Сервис управления пользователями | Управление учетными записями пользователей | Инфраструктурные сервисы | CRM-система |
| Сервис управления правами доступа | Управление правами доступа к функциям и данным | Инфраструктурные сервисы | Бухгалтерская система |
| Сервис интеграции данных | Обеспечение интеграции данных между компонентами | Инфраструктурные сервисы | Система отчетности и аналитики |
| Сервис API для интеграции | Предоставление API для интеграции с внешними системами | Сервисы для разработки | CRM-система |
| Сервис разработки отчетов | Разработка и кастомизация отчетов | Сервисы для разработки | Система отчетности и аналитики |
| Сервис тестирования интеграции | Тестирование интеграции между компонентами | Сервисы для разработки | Система управления заказами |
| Сервис мониторинга производительности | Мониторинг производительности системы | Управляющие сервисы | Система управления запасами |
| Сервис управления ошибками | Управление ошибками и исключительными ситуациями | Управляющие сервисы | Бухгалтерская система |
| Сервис управления обновлениями | Управление обновлениями программного обеспечения | Управляющие сервисы | Система отчетности и аналитики |

7. В качестве примеров, реализовать программно веб-сервисы, работающие в синхронном и асинхронном режиме (вопрос - ответ).

Пример реализации веб-сервиса для обработки заказов (синхронный режим):

python

from flask import Flask, request, jsonify

app = Flask(\_\_name\_\_)

orders = []

@app.route('/orders', methods=['POST'])

def create\_order():

data = request.json

orders.append(data)

return jsonify({"message": "Order created", "order": data}), 201

@app.route('/orders/<int:order\_id>', methods=['GET'])

def get\_order(order\_id):

if order\_id < len(orders):

return jsonify(orders[order\_id])

else:

return jsonify({"error": "Order not found"}), 404

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

Пример реализации веб-сервиса для анализа продаж (асинхронный режим):

python

import asyncio

from aiohttp import web

async def analyze\_sales(request):

data = await request.json()

# Асинхронная обработка данных

await asyncio.sleep(2) # Имитация длительной обработки

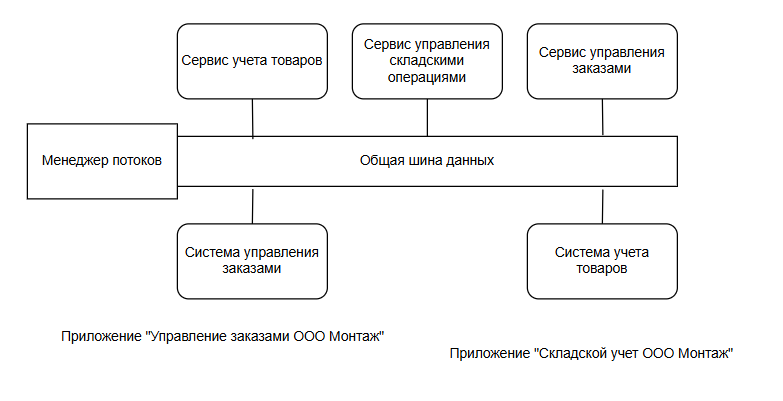
return web.json\_response({"message": "Sales analyzed", "data": data})

app = web.Application()

app.router.add\_post('/analyze', analyze\_sales)

web.run\_app(app)

8. Отобразить схемой пример интеграции нескольких бизнес-процессов (рис.9.2):



Например, в системе управления заказами приложением будет служить «Заказы ООО Монтаж»