**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Сибирский государственный университет науки и технологий**

**имени академика М.Ф. Решетнева»**

**Кафедра информационных экономических систем**

**Отчет по лабораторной работе №8**

**«Проектирование физической реализации разрабатываемого приложения с помощью UML Deployment»  
Вариант №22**

Руководитель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.В. Василенко

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка, дата)

Выполнил:

студент группы БПЦ23-01

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.А. Швалов

(подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_28.10.2025\_\_\_\_\_\_\_\_\_\_

**Цель работы:** спроектировать физическую реализацию разрабатываемого приложения с помощью UML Deployment Diagram.

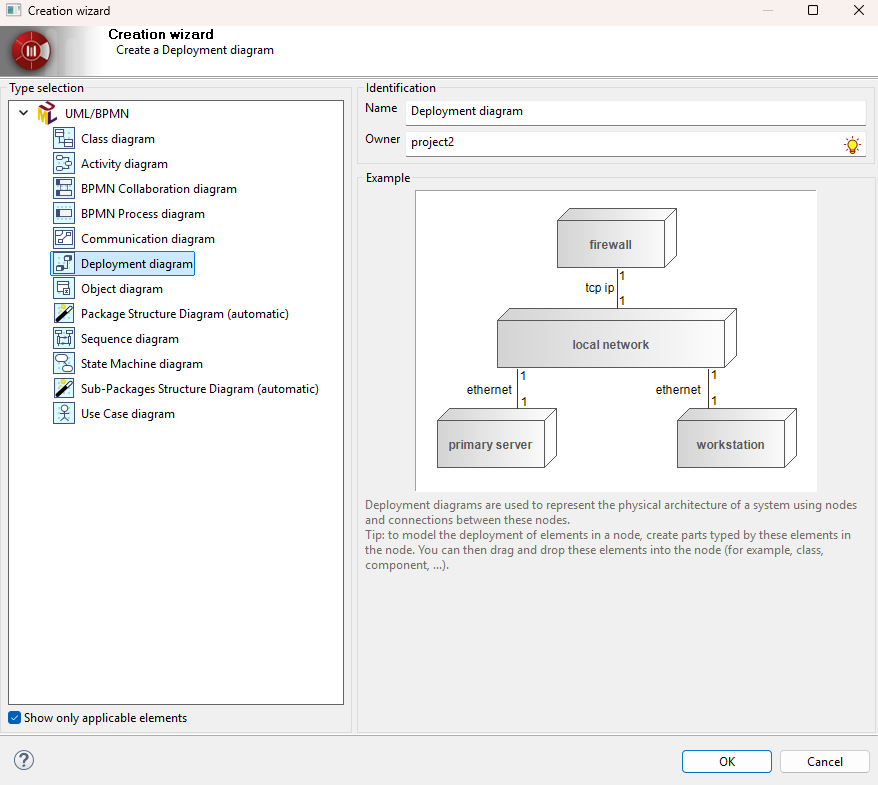
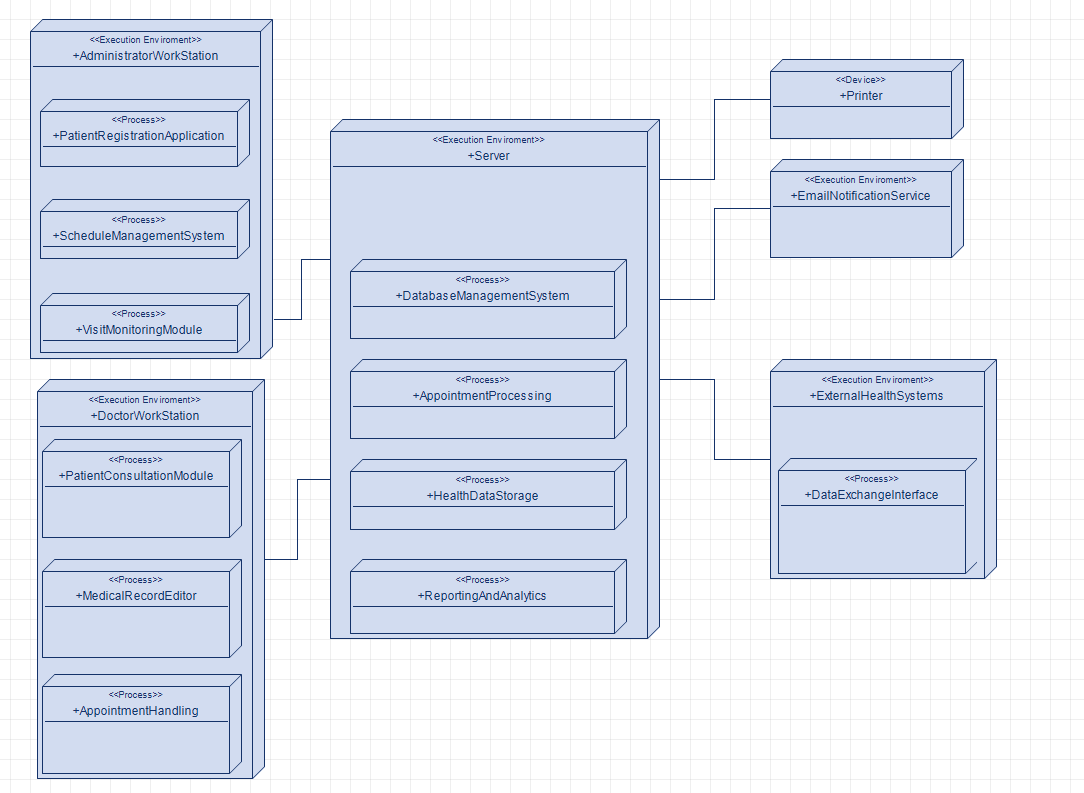
Создадим диаграмму развертывания.

Рисунок 1 – Создание диаграммы

Создадим узлы, в которых будут находится определенные артефакты.

Рисунок 2 – Диаграмма развертывания

**Administrator (администратор / регистратура):**

1. Регистрирует новых пациентов и создает медицинские карты.
2. Вносит и корректирует данные пациентов в системе.
3. Записывает пациентов на прием к специалистам.
4. Контролирует выполнение расписания и посещаемость.
5. Взаимодействует с врачами и пациентами по вопросам записи и расписания.

**Doctor (врач):**

1. Получает список записанных пациентов и время приема.
2. Проводит осмотр и вносит данные в медицинскую карту.
3. Назначает лечение, анализы и дополнительные обследования.
4. Вносит результаты осмотров, диагнозы и рекомендации.
5. Формирует заключения для дальнейшего наблюдения пациента.

**Patient (пациент):**

1. Записывается на прием к врачу через личный кабинет или терминал.
2. Просматривает расписание, результаты обследований и назначения.
3. Получает электронные рекомендации и заключения врача.
4. Оплачивает медицинские услуги онлайн.
5. Является потребителем медицинских услуг учреждения.

**Server (сервер):**

1. Хранит базу данных с информацией о пациентах, врачах, записях и медицинских картах.
2. Обрабатывает бизнес-логику — запись, учет посещений, обновление расписания.
3. Обеспечивает безопасность данных и контроль прав доступа.
4. Интегрируется с внешними системами (финансовыми, лабораторными и страховыми).
5. Предоставляет веб-интерфейсы для администраторов, врачей и пациентов.

**Database (база данных):**

1. Содержит таблицы классов: «Пациент», «Врач», «Запись», «Расписание», «Медицинская карта», «Администратор».
2. Обеспечивает целостность и доступность данных.
3. Сохраняет историю посещений, диагнозов и назначений.
4. Поддерживает резервное копирование и восстановление данных.

**PaymentSystem (платежная система):**

1. Обеспечивает прием онлайн-платежей от пациентов.
2. Интегрируется с банковскими и страховыми сервисами.
3. Формирует электронные чеки и квитанции об оплате.
4. Передает информацию о платежах в сервер медицинских услуг.

**Вывод:** в данной лабораторной работе была спроектирована диаграмма развертывания для разрабатываемого приложения в Modelio Open Source.