Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ

УНИВЕРСИТЕТ»

«Проектирование программного обеспечения»

Отчёт по лабораторной работе №6

Объектно-ориентированное моделирование. ФИЗИЧЕСКИЕ ДИАГРАММЫ UML.   
Цель: Изучение методологии объектно-ориентированного моделирования средствами UML. Ознакомление с основными принципами объектно-ориентированного проектирования программного обеспечения, получение навыков проектирования архитектуры информационной системы с применением методологии UML.

Выполнил: Драч К.А.

ФИТ 3 курс 1 группа

Преподаватель: Панченко О.Л.

1. Описание программных средств

Как и в прошлой лабораторной работе, для построения диаграмм будем использовать Draw.io –бесплатное онлайн-приложение для создания диаграмм и схем.

Название: Draw.io

Версия: Веб-приложение

Разработчик: JGraph Ltd.

Адрес: https://www.draw.io/

Режим использования: Онлайн

Доступность на платформах: любой веб-браузер.

Draw.io позволяет создавать различные типы диаграмм, такие как блок-схемы, организационные диаграммы, UML-диаграммы, сетевые диаграммы и многое другое.

1. Описание практического задания

Диаграмма развёртывания для нашего проекта будет следующая – рисунок 1.

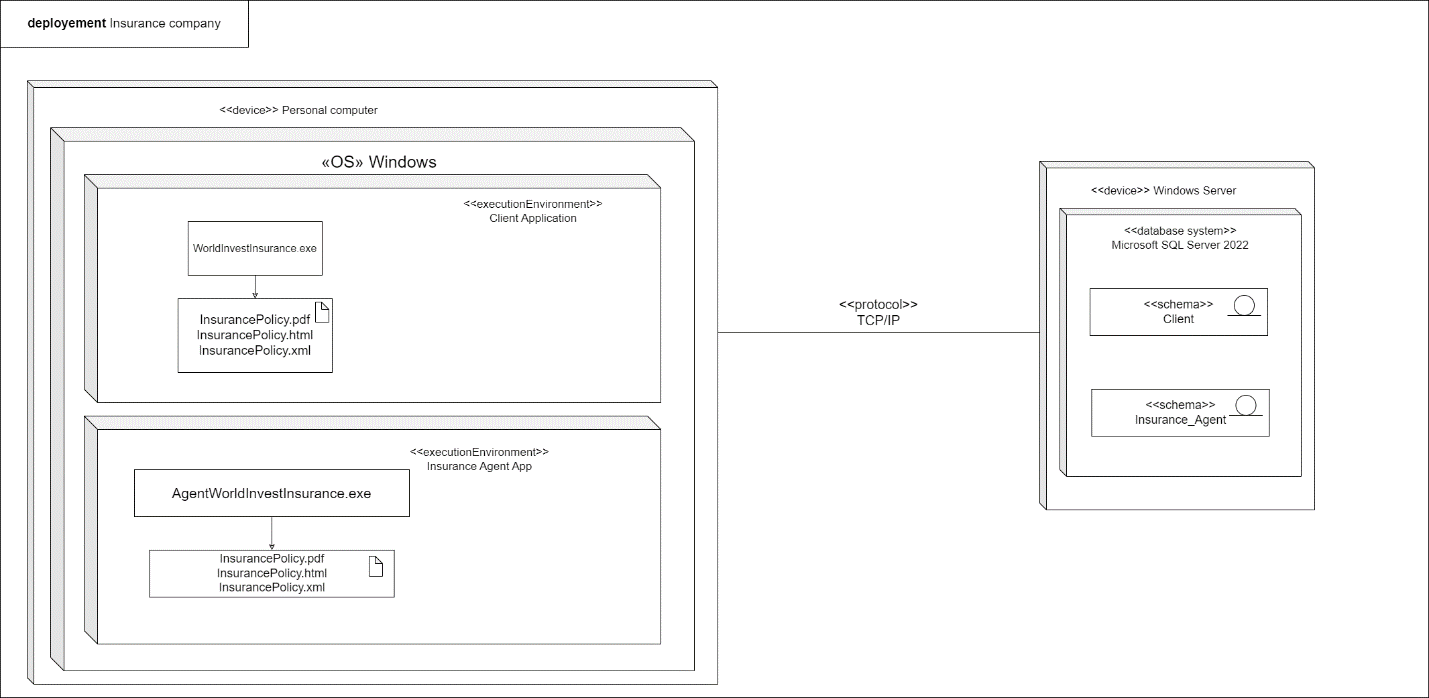


Рисунок 1 – Диаграмма развёртывания и компонентов

Диаграмма показывает схему развёртывания приложения страховой компании, взаимодействие сред и артефактов системы.

**Ответы на теоретические вопросы**

1. **Назначение физических диаграмм:**
   * **Диаграмма компонентов:** Используется для визуализации высокоуровневой структуры системы, показывая её компоненты (или модули) и их взаимосвязи. Отражает архитектурные блоки и их взаимодействие.
   * **Диаграмма развертывания:** Описывает физическое размещение компонентов системы на аппаратном обеспечении. Позволяет увидеть, как компоненты системы развернуты на узлах и как они взаимодействуют в среде выполнения.
2. **Определения:**
   * **Узел:** Узел представляет собой физическое или вычислительное устройство, на котором выполняются компоненты системы. Примеры узлов включают серверы, компьютеры, мобильные устройства и т.д.
   * **Артефакт:** Артефакт представляет собой физический объект или продукт, производимый или используемый процессом разработки. Это может включать в себя файлы, библиотеки, базы данных, исполняемые файлы и другие элементы.
   * **Интерфейс:** Интерфейс представляет собой точку взаимодействия между двумя компонентами, определяя методы или соглашения, по которым они обмениваются информацией.
3. **Нотации для компонентов:**
   * **UML-диаграмма компонентов:** Используется в языке моделирования UML. Компоненты представлены прямоугольниками, а связи между ними показывают взаимодействие.
   * **IDEF0 (Integration Definition for Function Modeling):** Используется для функционального моделирования компонентов и их взаимосвязей.
4. **Нотации для диаграмм развертывания:**
   * **UML-диаграмма развертывания:** Компоненты представлены узлами, а связи показывают размещение компонентов на узлах.
   * **Archimate:** Используется для моделирования архитектурных видов, включая диаграммы развертывания.
5. **Основные виды связей:**
   * **Зависимость (Dependency):** Показывает, что изменения в одном компоненте могут повлиять на другой.
   * **Связь агрегации/композиции:** Показывает отношение "часть-целое" между компонентами.
   * **Использование (Usage):** Показывает, что один компонент использует функциональность другого.
   * **Взаимодействие:** Показывает обмен данными между компонентами.
   * **Связь развертывания:** Используется на диаграммах развертывания для показа размещения компонентов на узлах.