

Лабораторна робота № 7

Тема: Діаграми фізичного рівня

Мета: набуття практичних навичок щодо побудови та використання діаграм компонентів та розгортання.

Час: 2 години.

Література: конспект лекцій; Л2; Л6.

I. НАВЧАЛЬНІ ПИТАННЯ ТА РОЗРАХУНОК ЧАСУ

№ з/п	НАВЧАЛЬНІ ПИТАННЯ	Час (хв)
1.	Вступна частина	5
2.	Основна частина	90
2.1.	Перевірка готовності студентів до заняття	20
2.2.	Розробка діаграми компонентів та розгортання системи	70
3.	Заклучна частина	5

II. МЕТОДИЧНІ ВКАЗІВКИ

1 Вступна частина.

Перевірка наявності студентів, оголошення теми та мети заняття, інструктаж з правил техніки безпеки, ходу проведення заняття.

2 Основна частина.

2.1. Перевірка готовності студентів до заняття

Проводиться у вигляді письмового опитування.

Питання до опитування:

1. Графічне зображення об'єктів на діаграмах компонентів

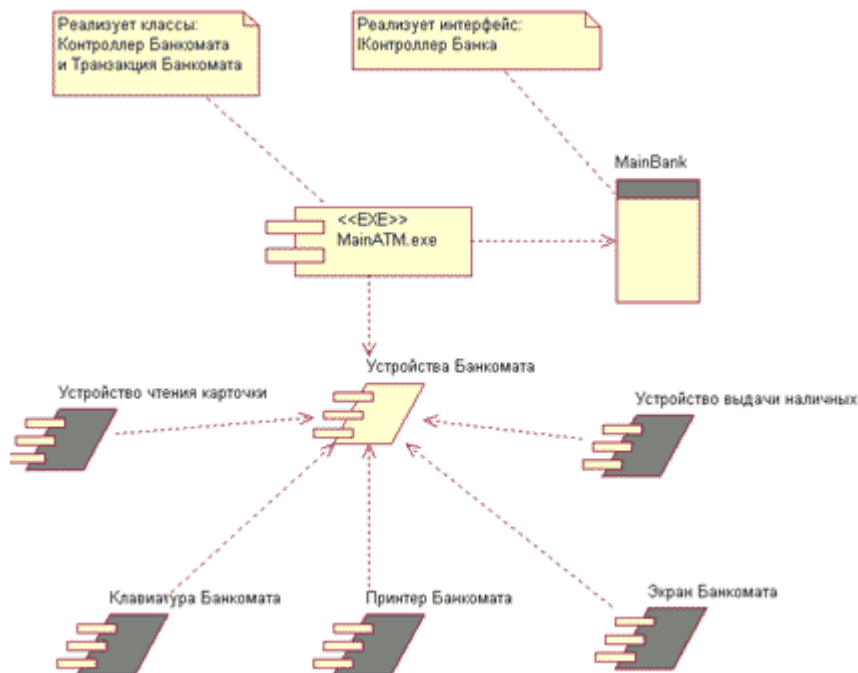
2. Зв'язки на діаграмах розгортання

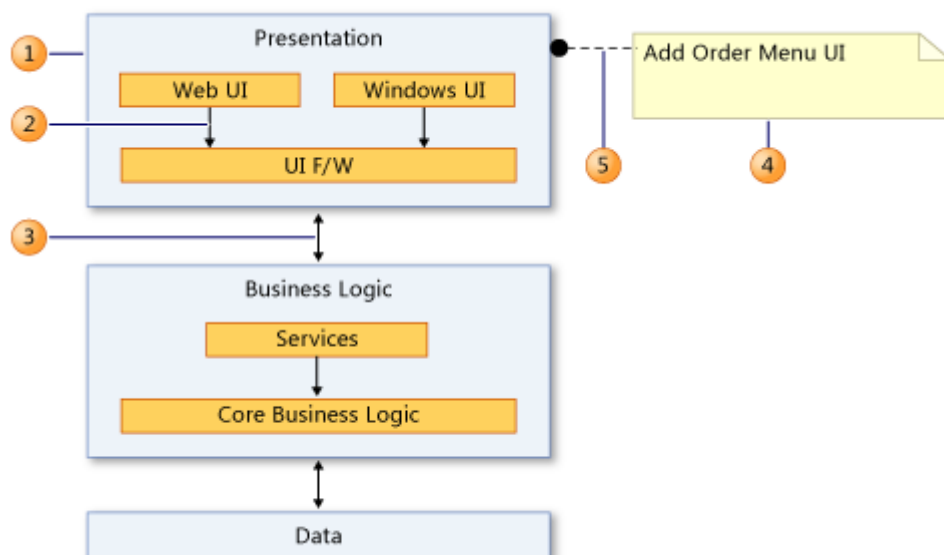
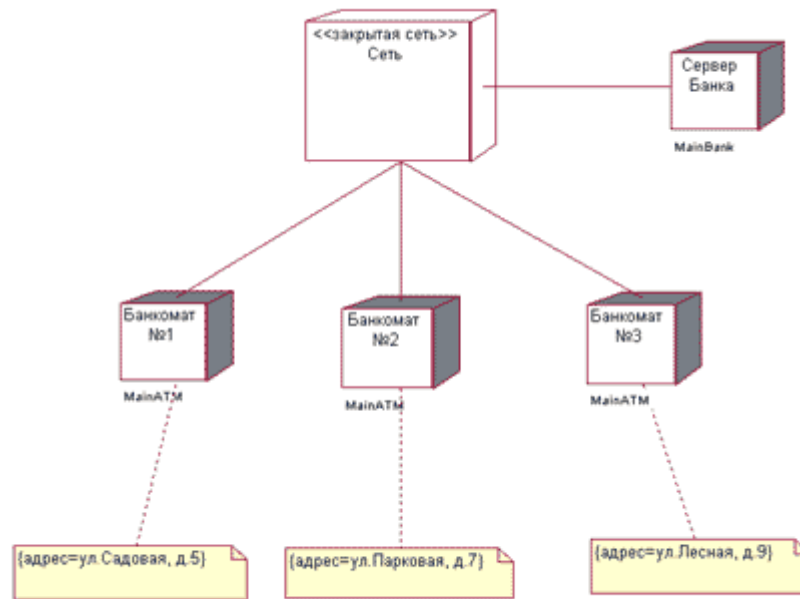
1. Графічне зображення об'єктів на діаграмах розгортання

2. Зв'язки на діаграмах компонентів

Завдання:

Завдання 1. Розглянути наведний приклад та проаналізувати порядок взаємодії та методи опису компонентів на діаграмах UML у відповідності до об'єктної структури системи.





Завдання 2. Побудувати діаграму компонентів та розгортання для класів системи за індивідуальним завданням. Для цього:

1. Визначити структуру програмного комплексу системи (виконуємі файли, бібліотеки, драйвери та ін.).
2. Визначити топологію та технологію реалізації комунікацій в системі.
3. Реалізувати діаграму компонентів.
4. Визначити типові вузли розгортання системи та з'єднання між ними.
5. Реалізувати діаграму розгортання.

4.a Визначити слої, їх типи та артефакти.

5.a Реалізувати схему слоїв.

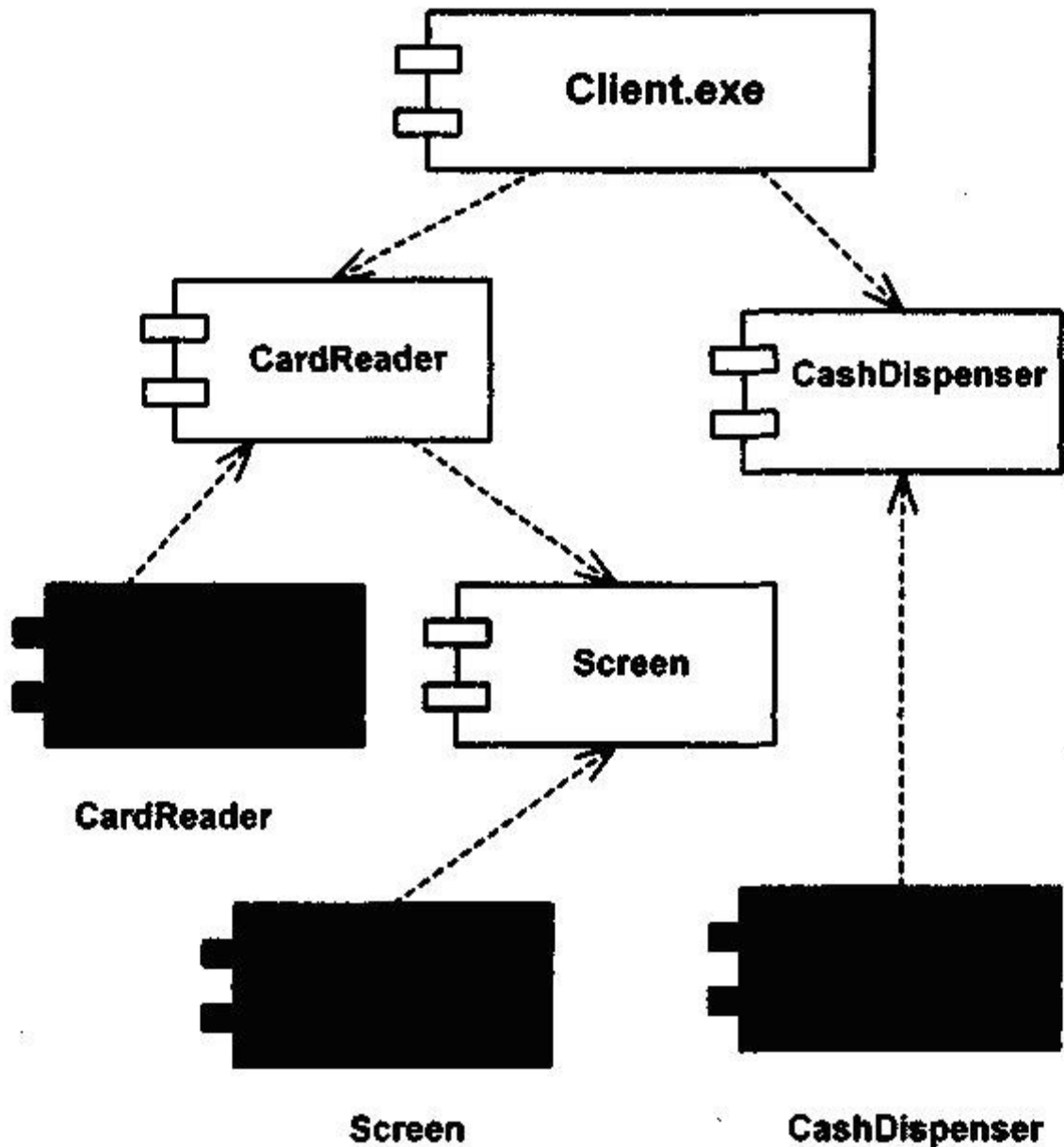
3 Заключна частина

Підведення підсумків заняття та оголошення оцінок, відповіді на запитання.

<http://msdn.microsoft.com/ru-ru/library/dd409393.aspx>

<http://msdn.microsoft.com/ru-ru/library/dd409390.aspx>

На рисунке изображена одна из диаграмм компонентов для некоторой системы обслуживания банкоматов архитектуры клиент-сервер.



На этой диаграмме показаны компоненты клиента системы. В данном случае система строится с помощью языка C++. У каждого класса имеется свой собственный заголовочный файл и файл с расширением .CPP, так что каждый класс преобразуется в свои собственные компоненты на диаграмме. Например, некоторый класс Screen преобразуется в два компонента,

представляющие тело и заголовок класса Screen. Выделенный темным компонент называется спецификацией пакета (package specification) и соответствует файлу тела класса Screen на языке C++ (файл с расширением .CPP). Невыделенный компонент также называется спецификацией пакета, но соответствует заголовочному файлу класса языка C++ (файл с расширением .H). Компонент Client.exe является исполняемой программой.

Компоненты соединены штриховой линией, что соответствует зависимостям между ними. Например, класс CardReader зависит от класса Screen. Это означает, что, для того чтобы класс CardReader мог быть скомпилирован, класс Screen должен уже существовать. После компиляции всех классов может быть создан исполняемый файл Client.exe.

В данном примере система включает два исполняемых файла. Один из них – это клиент Client.exe, он содержит компоненты CashDispenser, CardReader и Screen. Второй файл – это сервер, включающий в себя компонент Account.

Диаграммы компонентов применяются теми участниками проекта, кто отвечает за компиляцию системы. Из нее видно, в каком порядке надо компилировать компоненты, а также какие исполняемые компоненты будут созданы системой. На такой диаграмме показано соответствие классов реализованным компонентам. Она нужна там, где начинается генерация кода.

<http://msdn.microsoft.com/ru-ru/library/dd409462.aspx> - схема слоев

<http://msdn.microsoft.com/ru-ru/library/dd418995.aspx> справочные

материалы схема слоев

На рисунке изображен персональный компьютер, связанный с UNIX-сервером посредством протокола TCP/IP (Transmission Control Protocol/Internet Protocol – протокол управления передачей – протокол Интернет). Связи между узлами показывают коммуникационные каналы, с помощью которых осуществляются системные взаимодействия. Компоненты на диаграмме размещения представляют собой физические модули программного кода. Как правило, они в точности соответствуют компонентам на диаграмме компонентов. Таким образом, диаграмма размещения отражает выполнение каждого компонента в системе.

На данной диаграмме Пользовательский Интерфейс Отделения Заболеваний Печени зависит от Клиентской Части Отделения Заболеваний Печени, поскольку он обращается к конкретным методам клиентской части. Хотя коммуникация является двунаправленной в том смысле, что Клиентская Часть возвращает данные, Клиентская Часть не знает, кто ее вызывает, и поэтому не зависит от Пользовательского Интерфейса. Что касается коммуникаций между двумя компонентами Медицинской Помощи, каждый из них знает, что передается другому компоненту, поэтому коммуникационная зависимость является двунаправленной.

Компонент может иметь более одного интерфейса, в этом случае видно, какие компоненты взаимодействуют с каждым интерфейсом. На рисунке ПК включает два компонента: пользовательский интерфейс и клиентскую часть приложения. Клиентская часть приложения обращается к прикладному интерфейсу серверной части приложения. Отдельный компонент конфигурации выполняется только на сервере. Приложение взаимодействует с локальным компонентом Медицинская Помощь, который может взаимодействовать с другими компонентами Медицинской Помощи в сети.

Факт использования множества компонентов Медицинской Помощи скрыт от данного приложения. Каждый компонент Медицинской Помощи имеет свою локальную базу данных.



Для 8 лабораторной работы и курсовой

<http://msdn.microsoft.com/ru-ru/library/dd409448.aspx> - проверка системы в ходе разработки

<http://msdn.microsoft.com/ru-ru/library/ff657795.aspx> - создание кода

<http://msdn.microsoft.com/ru-ru/library/ff657806.aspx> - создание классов

<http://msdn.microsoft.com/ru-ru/library/dd409365.aspx> - визуализация кода