

# Обектно-ориентиран анализ и проектиране на софтуерни системи с UML

Зимен семестър 2020/2021

проф. Боян Бончев ([bbontchev@fmi.uni-sofia.bg](mailto:bbontchev@fmi.uni-sofia.bg)) и  
ас. Явор Данков ([yavor.dankov@fmi.uni-sofia.bg](mailto:yavor.dankov@fmi.uni-sofia.bg))

Всяко упражнение от тази дисциплина се състои от три части. Някои от тях са за групово решаване. За да Ви бъдат най-полезни те, е необходимо да се разделите на групи от 3-5 човека. Първата част е за групово дискусия, втората е за самостоятелно усвояване на инструменти и техники, а третата е за работа в екип и развитие на уменията за анализ и проектиране.

Материалите за този курс са достъпни от  
<https://learn.fmi.uni-sofia.bg/course/view.php?id=6734>

## Упражнение Модул 8: Диаграма на компонентите. Диаграма на разгръщане.

### Съдържание

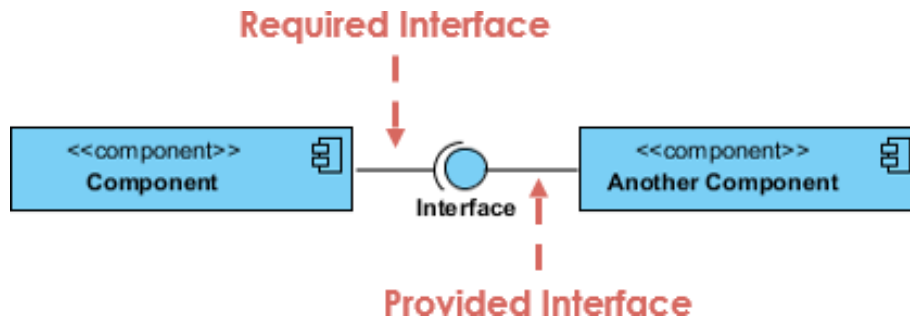
Диаграма на компонентите. Специфики на диаграма на компонентите.  
Диаграма на разгръщане. Употреби.

### Част А: Групово дискусия

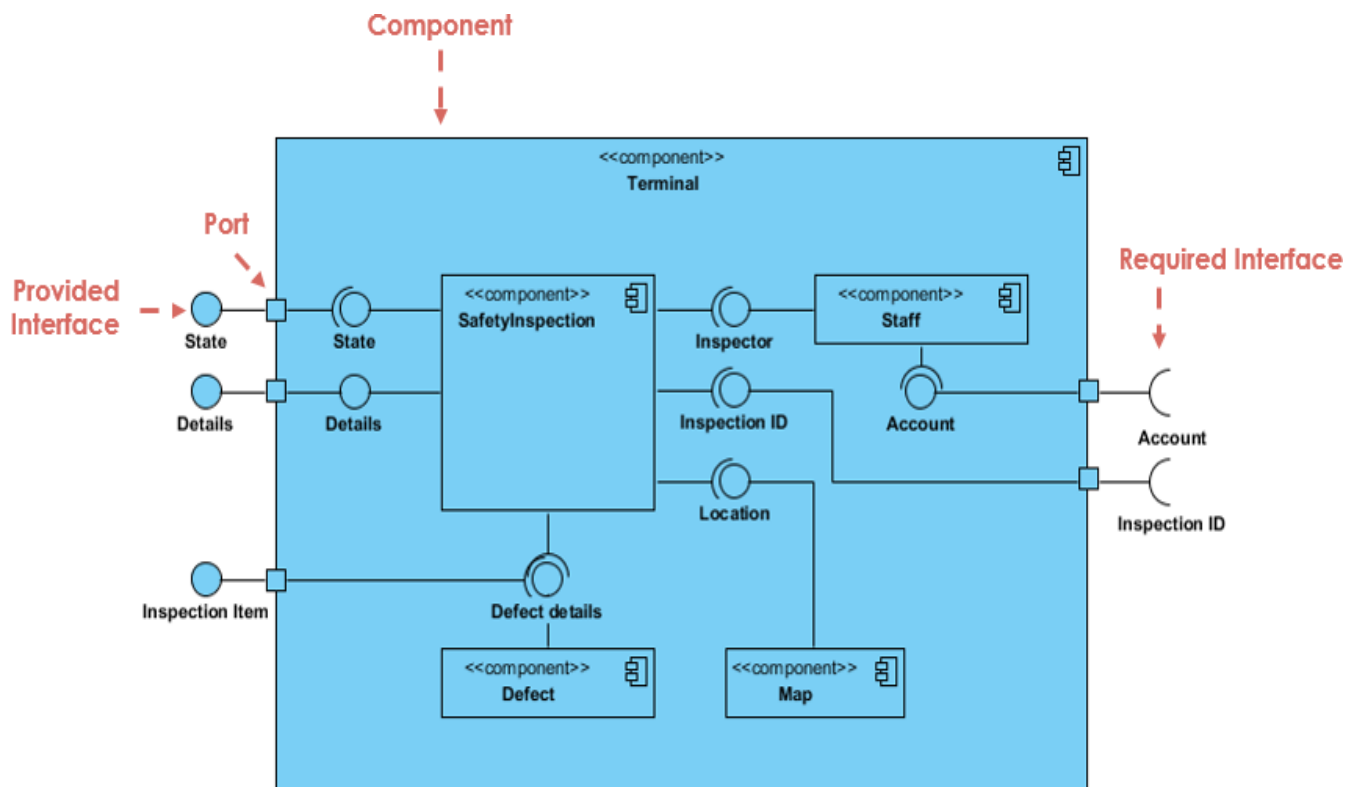
1. Кое е най-високото ниво на абстракция, което може да се пресъздаде чрез изучените диаграми?
2. Подсистеми, пакети, компоненти. Какви са приликите и разликите между тях? Къде другаде имаме пакети и какво обединяват те?
3. Какви типове компоненти познавате?
4. Освен между компоненти и пакети, къде другаде можем да имаме релация зависимост?
5. Чрез коя диаграма може да се опише инсталацията на едно приложение?
6. Какви детайли трябва да включва диаграмата на разгръщането?

7. Коментирайте следните примерни диаграми (Visual Paradigm), тяхното значение и характеристики?. Коментирайте приликите и разликите?

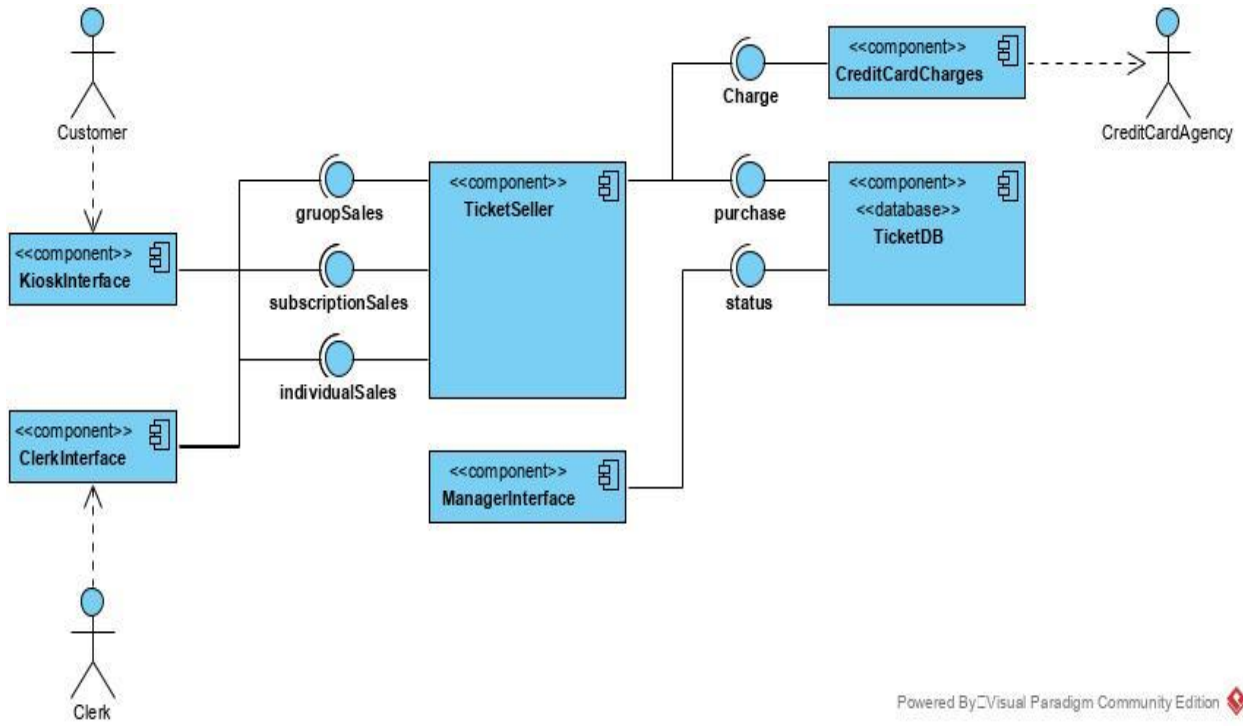
a)



b)

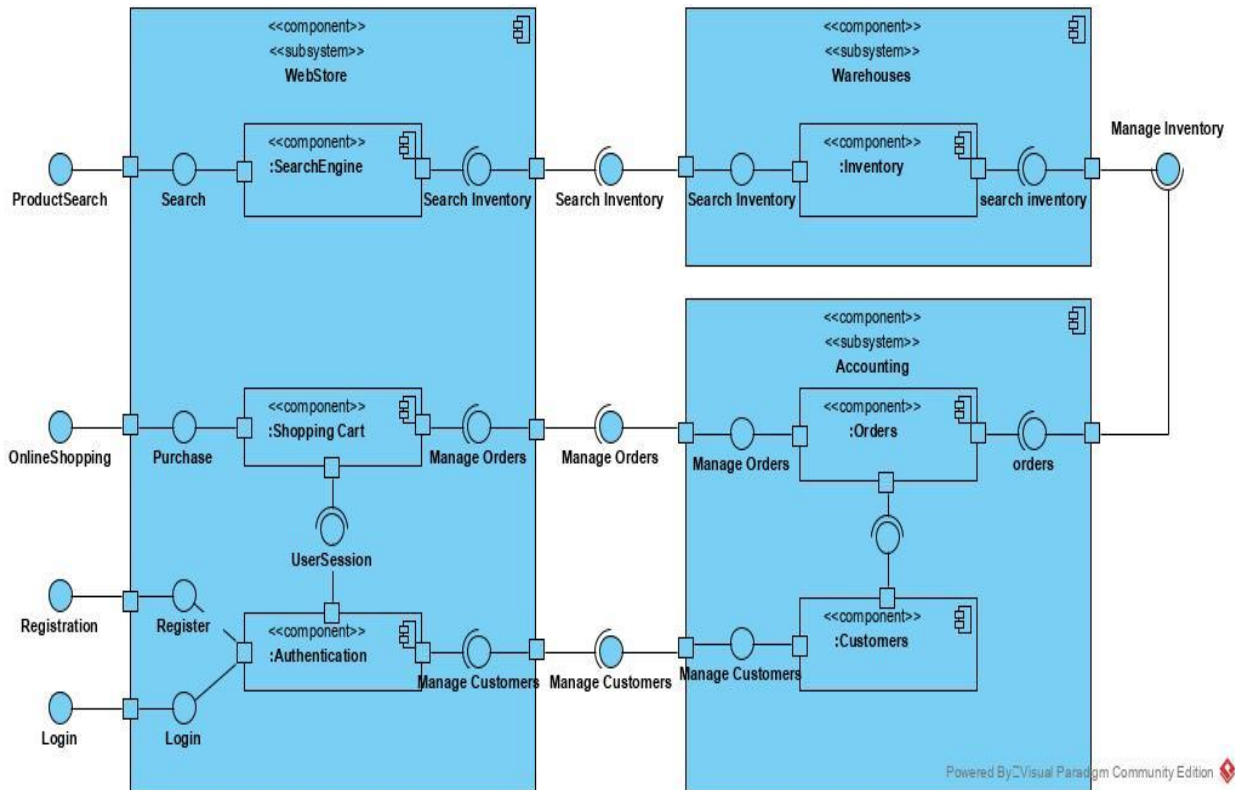


B)



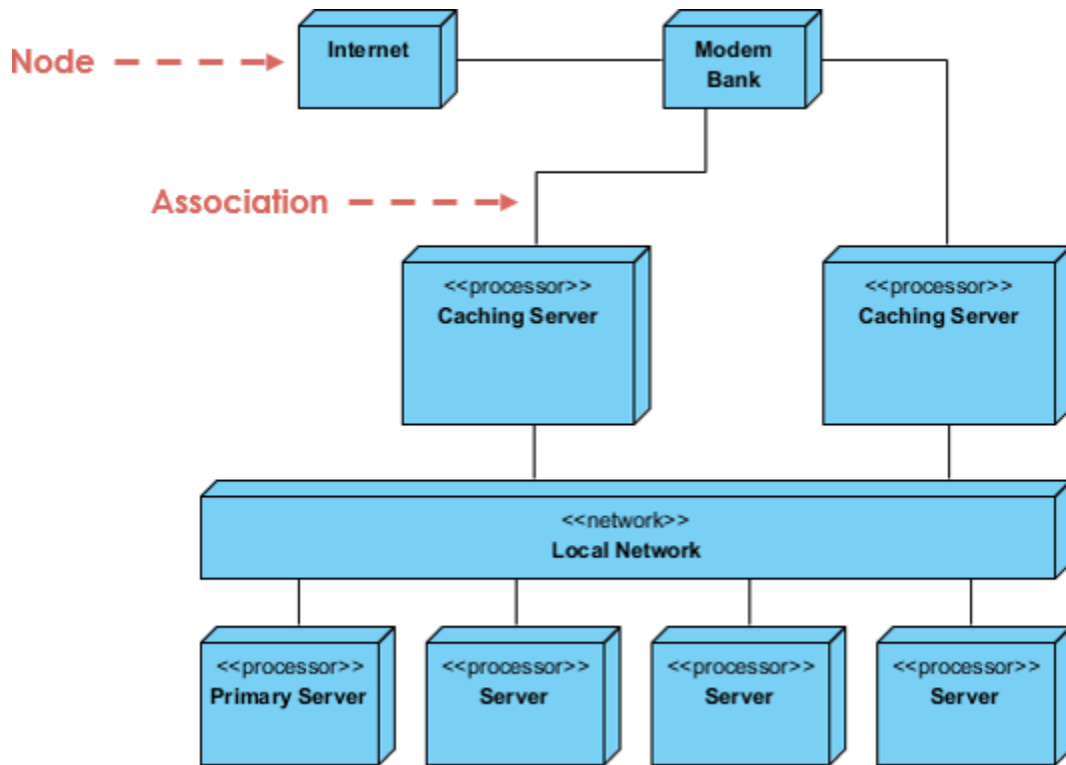
Powered By Visual Paradigm Community Edition

r)

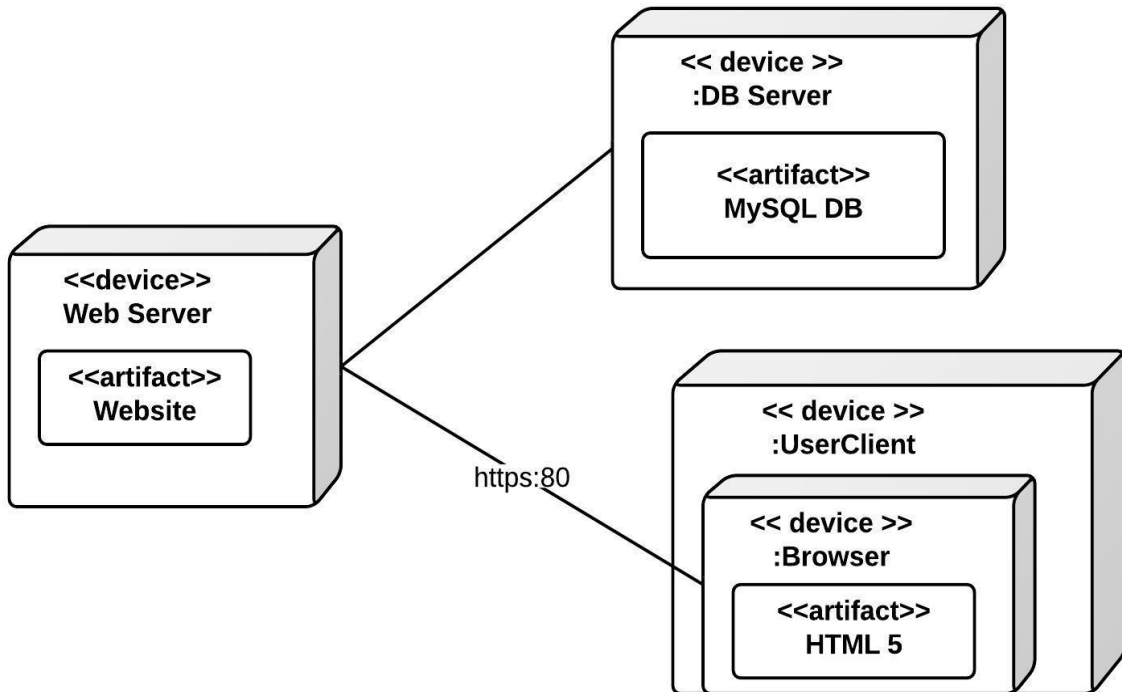


Powered By Visual Paradigm Community Edition

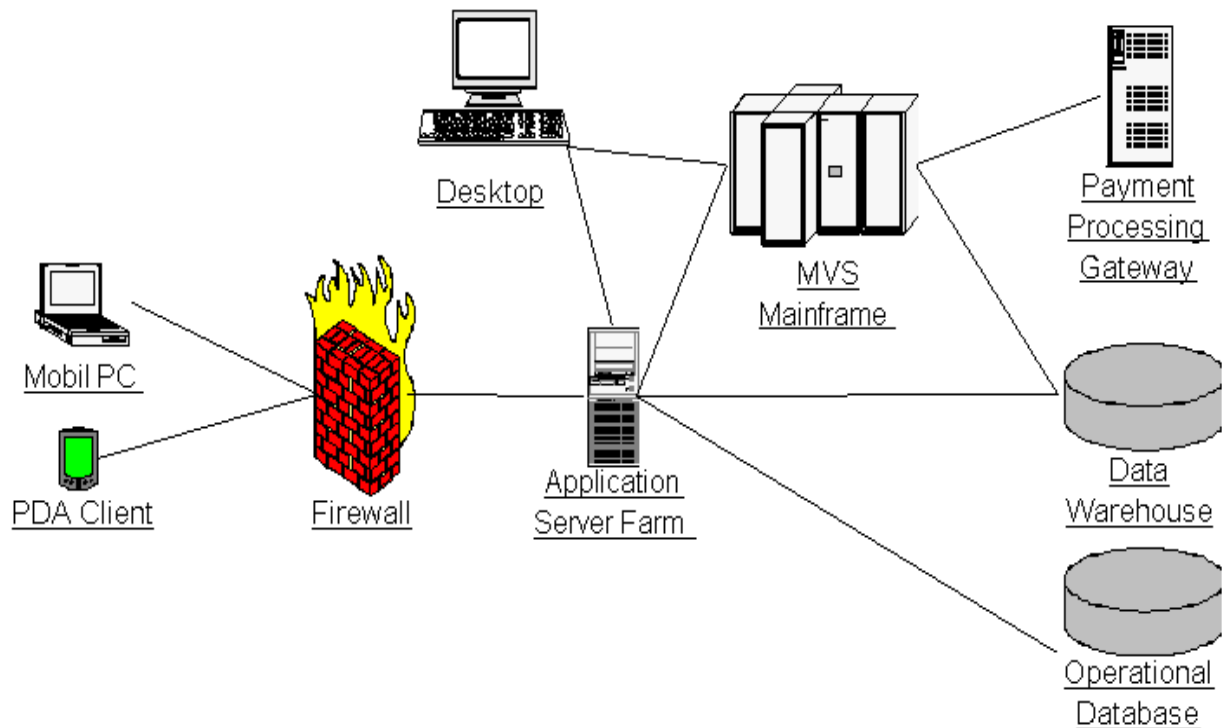
д)



е)



Ж)



## Част Б: Инструменти и техники

1. Създайте диаграма на компонентите и добавете един компонент. Задайте му стереотип <<subsystem>>.
2. Добавете на компонента интерфейс, който той да предоставя (provided interface).
3. Нека същият интерфейс се използва от втори компонент (required interface). Ще имаме ли имплицитна зависимост и ако да, в коя посока ще е тя?
4. Добавете трети компонент и го направете зависим от първия.
5. На третият компонент добавете един порт, който да предоставя един интерфейс и да зависи от друг.
6. Поставете първия и втория компонент в отделни пакети.
7. Добавете зависимост в правилната посока между пакетите.
8. Добавете стереотип <<import>> на зависимостта. Обърнете внимание, че такъв стереотип няма и ще трябва ръчно да го създадете.
9. Реализиране на компонент. За да добавите класове, които реализират един компонент:
  - a. Създайте нова диаграма на класовете (Може да я създадете и като под-диаграма на компонента)
  - b. Добавете желаните класове, заедно с техните релации към новосъздадената диаграма на класовете.

- c. С drag-and-drop поставете класовете върху компонента, който те реализират.
- d. В диаграмата на класовете създайте интерфейс и заменете с него интерфейса на първия компонент. Нека един от класовете в този компонент реализира интерфейса.

Заб.: при необходимост от помощ разгледайте описанието на адрес [https://www.visual-paradigm.com/support/documents/vpuserguide/94/2581/7292\\_drawingcompo.html](https://www.visual-paradigm.com/support/documents/vpuserguide/94/2581/7292_drawingcompo.html) .

## **Част В: Анализ и проектиране**

Социална мрежа.

Разгледайте продукта на Google - Google Plus. На какви подсистеми и компоненти може да се раздели. Каква е връзката му с Picasa, YouTube, Event Calendar, Google Maps, Gmail, etc. Създайте диаграма на компонентите като дефинирайте отделните елементи. Задайте подходящи стереотипи, определете интерфейсите, които предпоставят и използват. Свържете по подходящ начин различните елементи.