

# Упражнение по Софтуерни архитектури

## “4+1 архитектурен изглед”

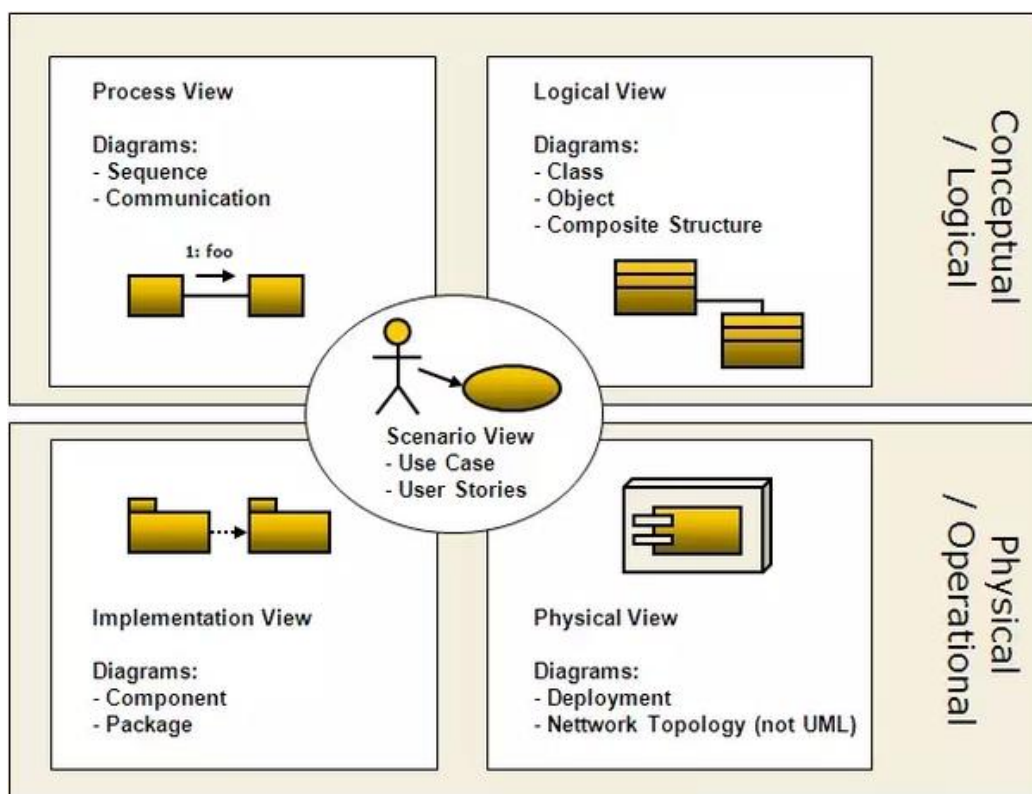
### Въведение

“4+1 архитектурен изглед” представя архитектурата на една програмна система погледната през призмата на четири различни гледни точки:

- ✓ на потребителя;
- ✓ на разработчика;
- ✓ на системния инженер;
- ✓ на системния интегратор.

Всеки изглед съдържа определени архитектурни структури, представени чрез съответните UML диаграми, показани на фигурата по-долу.

- В центъра на изгледа се намират Use case сценариите, определени от функционалните изисквания към системата.
- Логическият изглед (Logical) включва клас диаграми, диаграми на състоянията.
- Изгледът за разработка (Development, Implementation view) включва package диаграми, в които са разпределени основните класове на отделните модули.
- Физическият изглед включва диаграми на внедряването, показващи къде са разположени отделните модули на системата – на работната станция на потребителя, на уеб сървър, на сървър с бази данни. Каква е комуникационната среда между тези модули (LAN, VPN, Internet). Основен вид диаграма тук е deployment диаграмата.
- Process view – описва отделните изпълними процеси. UML диаграмите за представяне на изглед на процеса включват диаграмата на последователностите, комуникационната диаграма, диаграмата на активността (Activity диаграма).



Фигура 1. 4+1 Архитектурен изглед

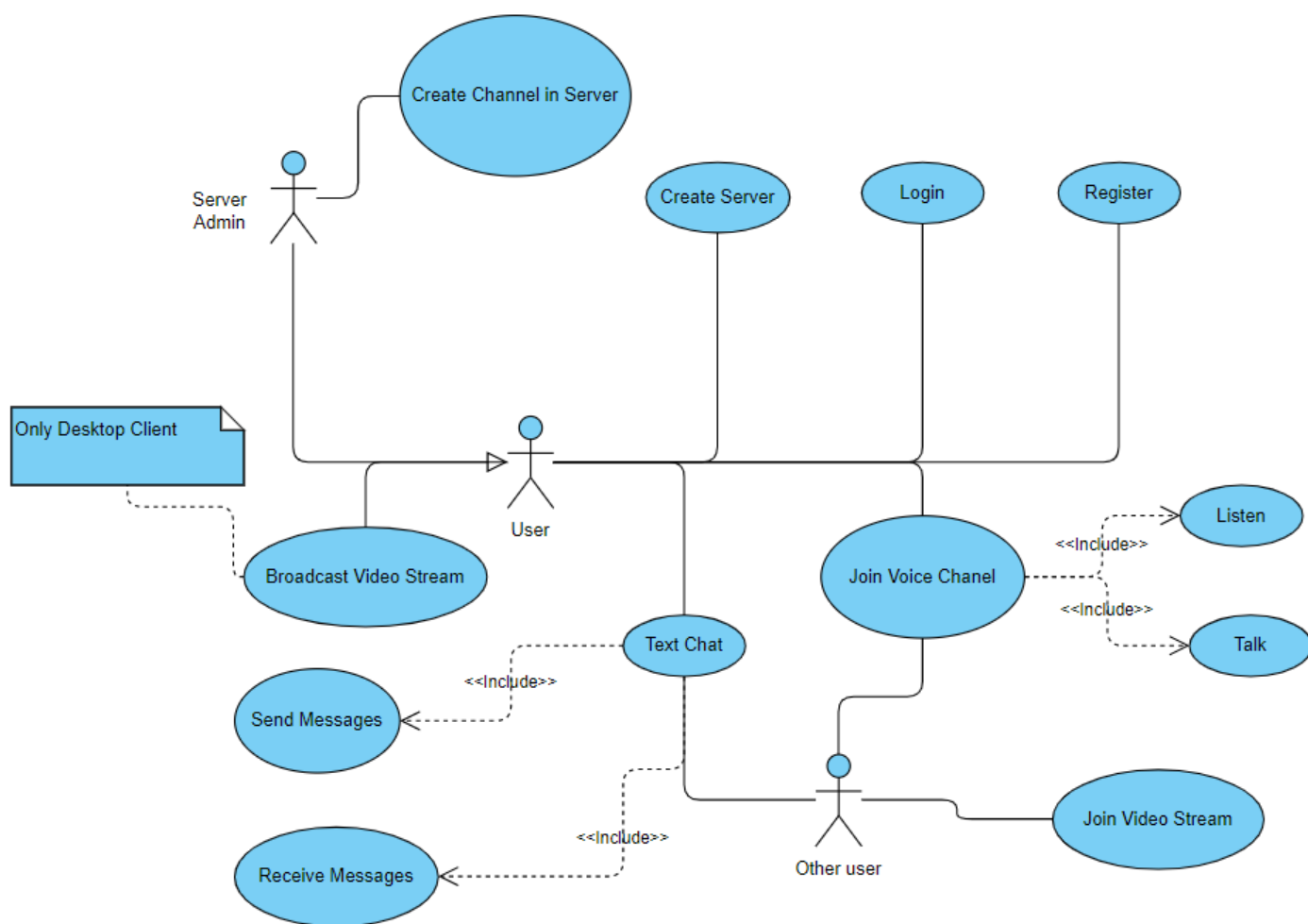
## Задача

На фигура 2 е представена UML Use case диаграма на приложение от вида на Discord (приложение за онлайн комуникация, онлайн конферентни връзки и споделяне на видео стрийм). Следвайки представената Use case диаграма реализирайте останалите 4 архитектурни изгледа, използвайки съответните UML диаграми, използвайки инструкциите по-долу.

## Допълнителни нефункционални изисквания

Достъпът до чат и конферентен разговор може да се осъществи чрез специализирано приложение за съответната платформа (Windows, Linux, Android, MacOS) или чрез уеб браузър (уеб базиран потребителски интерфейс). В архитектурата на системата трябва ясно да са разграничени отделните компоненти:

- Платформено зависимо приложение (десктоп базирано за персонални компютри по управлението на Windows, Linux, MacOS);
- Уеб сайт за достъп през браузър до услугите на системата (без функцията Broadcast video stream);
- Уеб сървър, играещ централна роля, приемащ и съхраняващ отделните съобщения. Бизнес логиката на системата е съсредоточена.

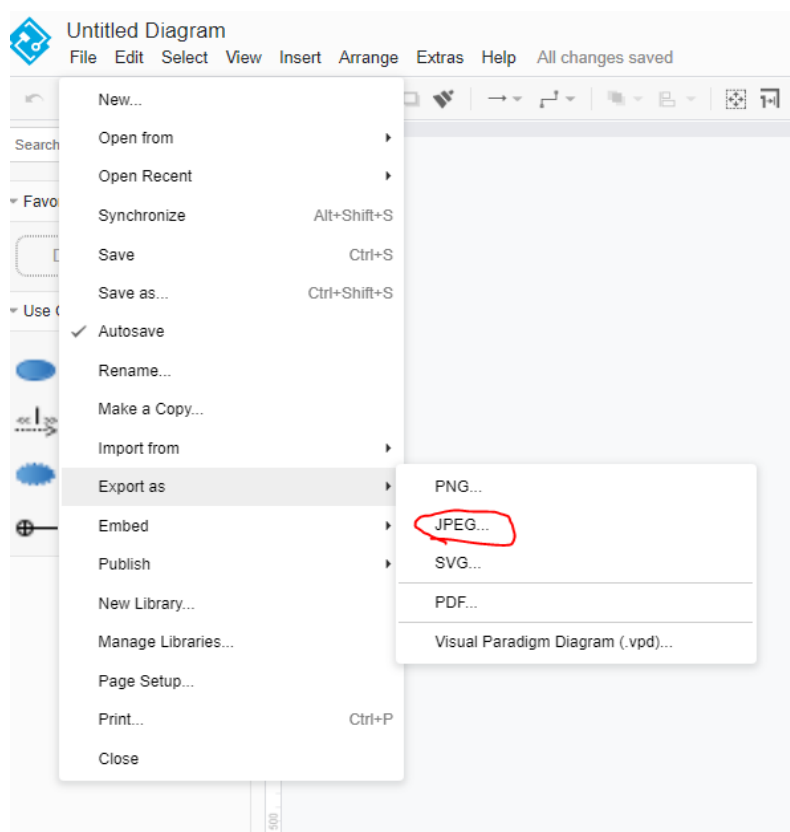


Фигура 2. Use case диаграма.

## Инструкции за реализация

Реализирайте четирите архитектурни изгледи, следвайки напътствията за всеки един от тях. Обединете всички UML диаграми (реализирани чрез <https://online.visual-paradigm.com/>) в един Word или PDF Документ.

Можете да направите експорт на моделираните UML диаграми като самостоятелни картинки чрез следната опция на онлайн-редактора



Всяка от картинките – UML диаграми, включете в един Word или PDF документ, добавяйки и текстово описание към всяка UML диаграма, описвайки най-важните особености.

## Логически изглед

В логическия изглед включете поне една клас диаграма. Основните класове трябва да бъдат “Потребител”, “Сървър”, “Канал” и други каквито сметнете, че са необходими. Създайте и поне една диаграма на състоянията – например описваща състоянията и подсъстоянията на даден потребител (офлайн, онлайн, в конферентен разговор, със заглушен микрофон, активен микрофон)

## Изглед за разработка

Опишете основните модули на системата – потребителски интерфейс, бизнес логика, база данни, разпределени в слоеве посредством UML package диаграми, подобно на дадената по-долу (източник [uml-diagrams.org](http://uml-diagrams.org))

