

## **UML (Unified Modeling Language) - резюме**

**UML** е език за уточняване, визуализиране, изграждане и документ. на софт. с-ми. **Модел/ абстракция** на с-мата от разл. гледни точки

**Обектът = Идентичност + Поведение + Състояние**

**Системата** - съвкупност от взаимодействащи си обекти

**Цели:** 1/Осигуряване изразителен визуален моделен език  
2/Независимост от езици за програмиране 3/Концепции на високо ниво 4/Интегриране на най-добри практики

**Архитектура.** Базира се на четирислойна метамодел архитектура:

1/ Мета-метамодел 2/Метамодел 3/Модел 4/User Object; User data

**Видове UML диаграми – поведенчески и структурни**

**Структурни:**

- **Клас** - статичната структура на софтуера. Показва йерархични връзки между класове и обекти
- **Composite Structure** - вътрешна структура на класа и неговите взаимодействия с други части на системата
- **Object** - снимка на instances of classes в системата и връзката, които съществува между тях
- **Компонентни** – показват на високо ниво конфигурации на софтуерната система в едно от нейните deploy-състояния
- **Deployment** - system hardware and its software
- **Пакетни** – показва групи от класове или случаи на употреба и описва като цяло поведението на системата

**Поведенчески:**

1. **State Machine** , **Диаграми на състоянието** - представяне на състоянието на системата или част от системата в крайни моменти от

време. Представя поведението, използвайки преходи с крайни състояния. Показват как състояния на обекти се променят при възникване на събития

**2.Activity** – описват последователност от действия- workflow. Поддържат условно и паралелно поведение. За анализ, работен процес, паралелни процеси

**3.Use Case**– поведението на с-мата, без да уточнява как работи и взаимодействието ѝ с околната среда. За спец.и анализ на изисквания

**4.Sequence** - изобразява взаимодействието между обектите в последователен ред

**5.Communication (Collaboration)** - показва последователни съобщения, обменени между обекти

**6.Timing** - диаграми на последователности, които се използват за изобразяване на поведението на обекти през времева рамка. Показват ограничения във времето и продължителността

**7.Interaction Overview** - илюстрира потока от взаимодействия между различни елементи в система или процеси.

**UML - Архитектурни изгледи:** 1/Потребителски (**use cases**) 2/ Структурен (static, **logical**)– class diagram 3/ **Имплементационен** - component diagram 4/ **Поведенчески** - dynamic – statechart, activity, interactions, sequence, component 5/ Environment (**deployment**)