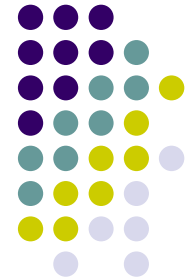


Analiza i dizajn korištenjem UML

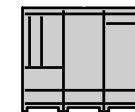


- Prednosti vizualnog modeliranja
- Povijesni razvoj UML-a
- Vizualno modeliranje korištenjem UML-a
- Iterativni razvojni proces

Vizualno modeliranje



- **Modeliranje** ima zadataku prepoznati bitne dijelove sustava
- **Vizualno modeliranje** je modeliranje koje koristi standardno grafičko označavanje



Računalni sustav

Vizualno modeliranje prepoznaje ključne dijelove procesa

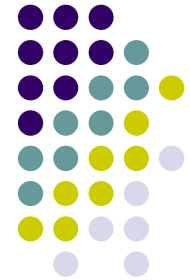


- Analiza slučajeve korištenja – sagledavanje poslovnog procesa iz perspektive korisnika
- Vizualno modeliranje
 - prepoznaje poslovne objekte i logiku
 - analizira i oblikuje programsku podršku
 - savladava se kompleksnost
- Model sustava preslikava se u fizičku arhitekturu
- Vizualnim modeliranjem potiče se višestruka upotrebljivost komponenti

Unified Modeling Language - UML

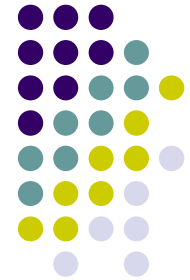


- UML objedinjava
 - koncepte modeliranja podataka (dijagram entiteta i odnosa među njima)
 - modeliranje poslovanja – radnog procesa
 - modeliranje objekata
 - modeliranje komponenti
- standardni jezik za vizualizaciju, specificiranje, konstruiranje i dokumentiranje složenih softverskih sustava
- Može se koristiti tijekom cijelog razvojnog ciklusa i neovisno o tehnikama implementacije



Povijest UML-a

- Tijekom 1990-ih, tri jezika za modeliranje su se unificirali (Booch, OOSE, OMT)
- Ciljevi unificiranja:
 - modeliranje sustava, od koncepata do izvršnih programa, korištenjem objektno-orijentiranih tehnika
 - dati učinkovit alat za manipuliranje kompleksnim sustavima
 - stvoriti jezik za modeliranje podjednako razumljiv ljudima i računalima



Razlozi za korištenje UML

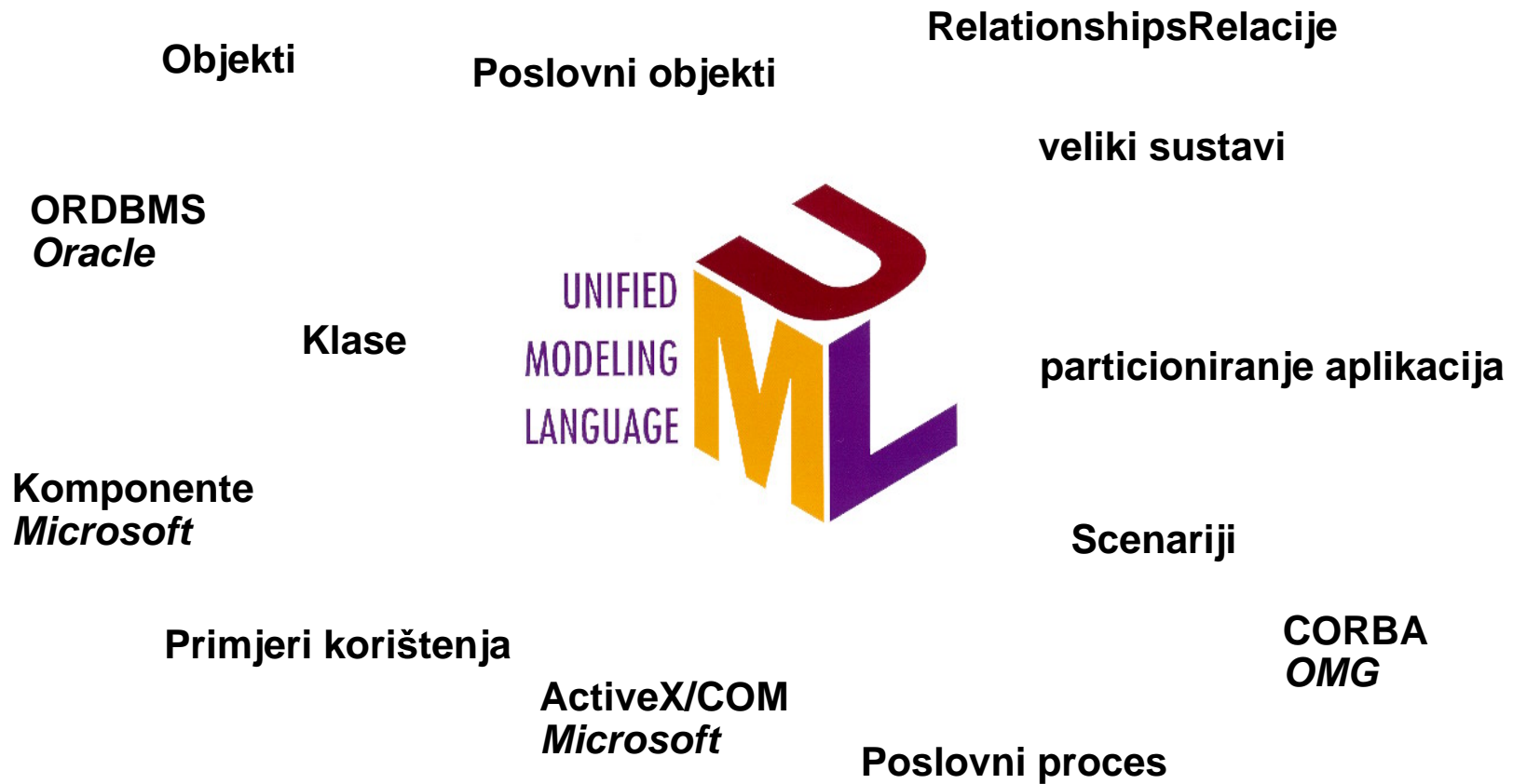
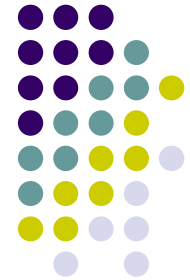
- Želja nam je razvijati softver koji će biti
 - kvalitetan
 - koristan
 - omogućavati održavanje
- Za trajno osiguravanje ovih zahtjeva potrebno je razviti svima razumljiv i jednostavan model (gradnja pseće kućice i nebodera)

Tri osnovne kategorije u UML gramatici

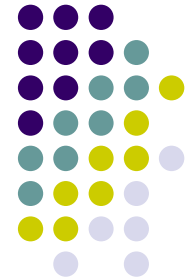


- Stvari (objekti)
- Odnosi među njima
- Dijagrami – pogodno složeni objekti i odnosi među njima

UML podržava razvoj mnogih aplikacija



Korištenje UML-a



- prikaz dosega sustava i njegovih glavnih funkcija korištenjem slučajevea i sudionika
- prikaz konkretnih slučajevea s dijagramima međudjelovanja
- Prikaz statičke strukture sustava korištenjem dijagrama klasa
- modeliranje ponašanja objekata pomoću tranzicijskih dijagrama.
- prikaz arhitekture fizičke implementacije pomoću dijagrama komponenti i distribucije

Primjer korištenja UML

Rad studentske službe

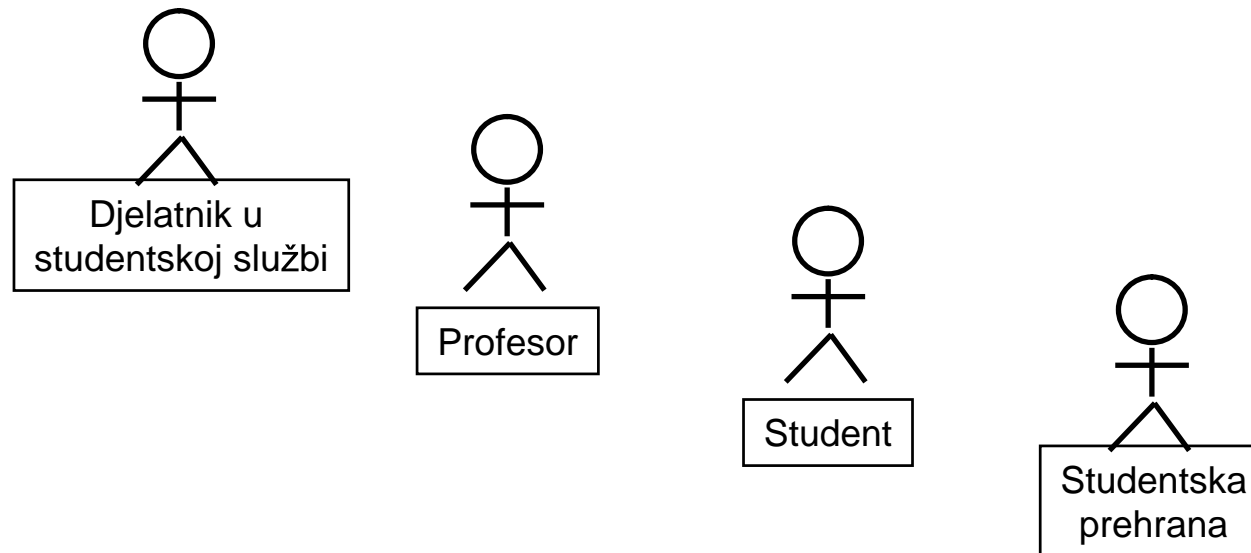


- Djelatnik u studentskoj službi upisuje kolegije koji se drže tijekom semestra
 - Na jednom smjeru nudi se više izbornih kolegija
- Student upisuje 4 obavezna i 2 izborna kolegija
- Nakon upisa semestra, student ostvaruje studentska prava (sustav prehrane itd.)
- Studenti mogu koristiti sustav za promjenu izbornih kolegija neko vrijeme nakon upisa
- Profesori koriste sustav za izradu popisa slušača svog kolegija
- Za pristup sustavu svi korisnici sustava imaju korisničko ime i lozinku

Sudionici



- **Sudionik** je svaka osoba odnosno sustav koji na neki način koristi sustav odnosno s njim komunicira



Slučajevi korištenja (use cases)



- **Slučaj korištenja** (use case) je uzorak ponašanja sustava, podsustava ili klase što se ogleda u razmjeni poruka između sustava i jednog ili više vanjskih sudionika.
- Sudionici moraju specificirati svoje zahtjeve
 - Djelatnik u studentskoj službi: Održavanje popisa kolegija
 - Profesor: Zahtijeva popis studenata
 - Student: Odabire kolegije s obzirom na satnicu
 - Studentska prehrana: dobiva popis studenata koji imaju pravo na prehranu

Dokumentiranje slučajeva korištenja



- Za svaki slučaj korištenja definira se slijed događaja – gledano sa strane sudionika
- Opisuje se što sustav mora pružiti sudioniku u određenom slučaju
- Tipični dijelovi:
 - Početak i kraj slučaja korištenja
 - Uobičajeni slijed događaja
 - Slijed događaja koji se rjeđe javlja
 - Slijed događaja koji se javlja izuzetno rijetko

Održavanje popisa kolegija

Slijed događaja

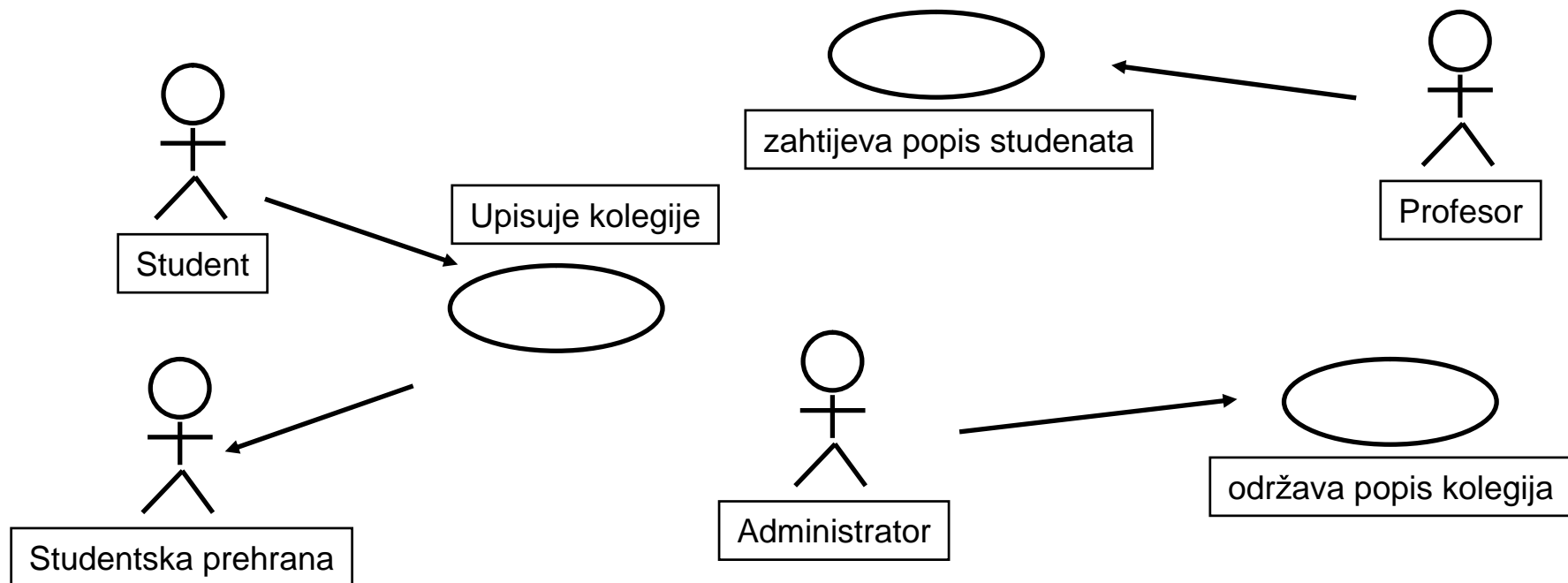


- Administrator (djelatnik studentske službe) se prijavljuje u sustav putem lozinke. Sustav provjerava ispravnost lozinke (E-1) i postavlja pitanje za odabir semestra (E-2). Administrator odabire semestar. Nakon toga sustav postavlja pitanje o željenoj akciji: DODAJ, IZBRIŠI, IZMIJENI i IZLAZ.
- Ako se odabere aktivnost DODAJ, izvodi se procedura opisana u S-1: Dodavanje kolegija.
- Ako se odabere aktivnost IZBRIŠI, izvodi se procedura opisana u S-2: Brisanje kolegija.
- Ako se odabere aktivnost IZMIJENI, izvodi se procedura opisana u S-3: Izmjena kolegija.
- Ako se odabere IZLAZ, slučaj korištenja završava

Dijagram slučajeja korištenja



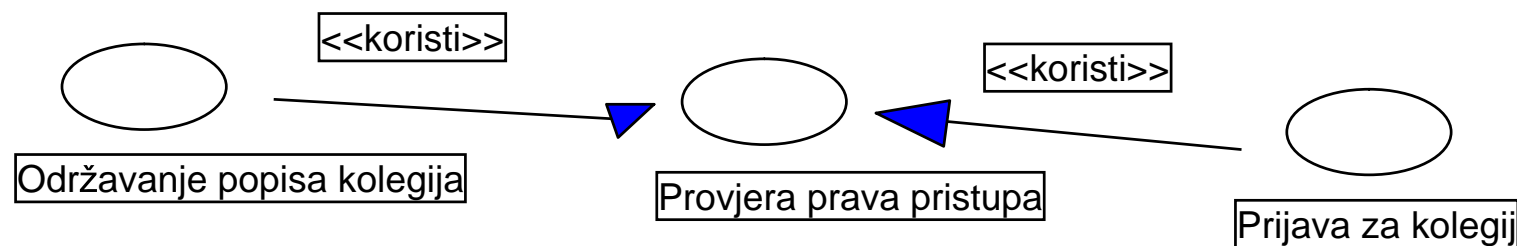
- Za vizualizaciju odnosa između sudionika i slučajeja korištenja crtaju se dijagrami slučajeja korištenja



Relacija korištenja



- Nakon popisivanja slučajeva korištenja, mogu se otkriti novi odnosi među pojedinim slučajevima korištenja
- Veza putem korištenja pokazuje ponašanje koje je zajedničko u više slučajeva korištenja



Realizacije slučajeveva korištenja



- Dijagram slučajeveva korištenja predstavlja vanjski pogled na sustav
- Interakcijski dijagrami opisuju kako se slučajevevi korištenja realiziraju putem interakcije među objektima
- Dvije vrste interakcijskih dijagrama:
 - Slijedni dijagram (*sequence diagram*)
 - Dijagram međudjelovanja (*collaboration diagram*)
- ako imamo jedan dijagram onda iz njega možemo konstruirati drugi dijagram jer pokazuju iste informacije na različit način



Slijedni dijagram

- opisuje dinamičku suradnju između objekata
- prikazuje **vremenski slijed poruka** koje objekti razmjenjuju (najčešće se pokazuje primjer u kojem se vide sve poruke koje objekti razmjenjuju)
- pokazuje vrijeme života objekta (kreiranje i uništavanje)
- objekti se prikazuju u formatu:
naziv_objekta:naziv_klase

Primjer slijednog dijagrama

