

Use case dijagrami

# Scenario

- Scenario je sekvenca akcija koja opisuje interakciju između korisnika i Sistema
- Primjer on-line prodavnica, scenario Kupovina proizvoda
  - *Kupac pretražuje katalog proizvoda i željene proizvode dodaje u korpu. Prilikom plaćanja, kupac unosi potrebne informacije o platnoj kartici i potvrđuje kupovinu. Sistem provjerava unijete parametre i prikazuje poruku kojom se obavještava kupac o uspješno obavljenoj kupovini, nakon čega se generiše i šalje odgovarajuće obavještenje na mejl adresu kupca.*
- Šta se dešava ukoliko autorizacija kupca nije izvršena uspješno?

# Slučaj korišćenja

- Slučaj korišćenja je skup povezanih scenarija koji se odnose na isti cilj, gledano sa strane korisnika sistema
- Tipično, use case sadrži jedan osnovni ili primarni scenario (all-goes-well case) i nekoliko alternativnih scenarija
- „primary elements in software development“, Jacobson (1994)

# Prikaz slučajevea korišćenja

- Zbog nepostojanja formalnog standarda moguće su razne varijante, npr. dozvoljeno je definisati preuslove koji moraju biti zadovoljeni da bi otpočeo slučaj upotrebe

## Buy a Product

1. Customer browses through catalog and selects items to buy
2. Customer goes to check out
3. Customer fills in shipping information (address; next-day or 3-day delivery)
4. System presents full pricing information, including shipping
5. Customer fills in credit card information
6. System authorizes purchase
7. System confirms sale immediately
8. System sends confirming email to customer

### *Alternative: Authorization Failure*

At step 6, system fails to authorize credit purchase

Allow customer to re-enter credit card information and re-try

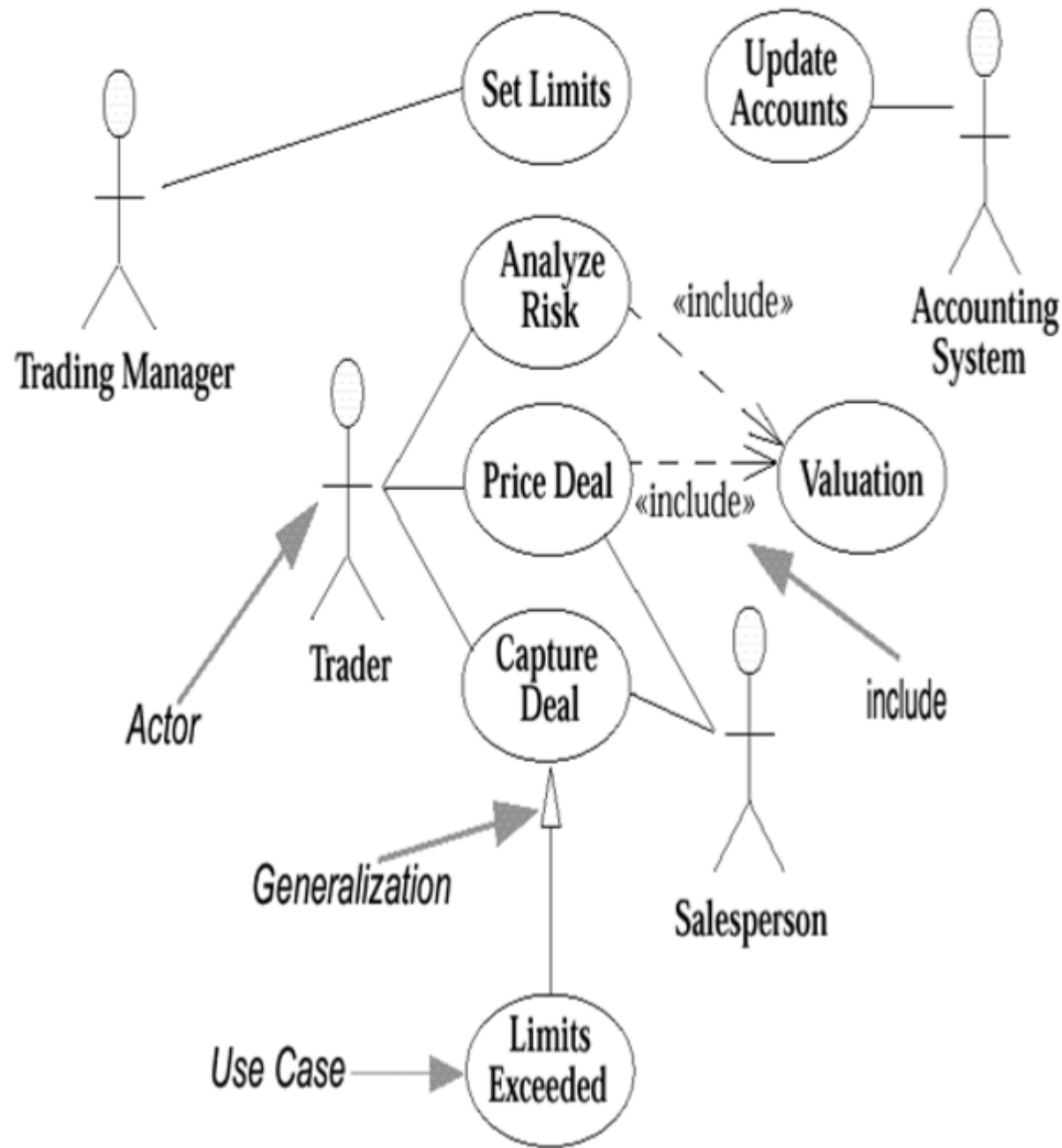
### *Alternative: Regular Customer*

3a. System displays current shipping information, pricing information, and last four digits of credit card information

3b. Customer may accept or override these defaults

Return to primary scenario at step 6

# Dijagrami slučajeve korišćenja

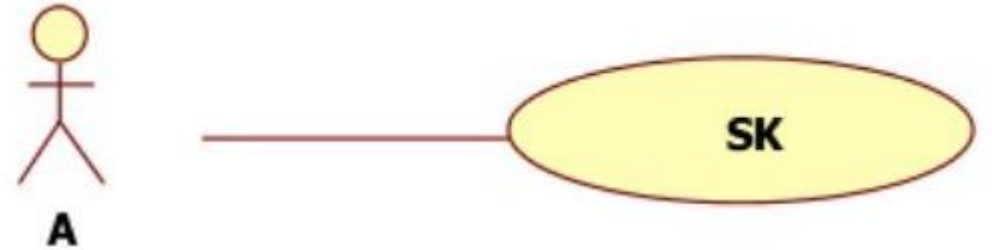


# Akteri

- Akter je uloga koji korisnik ostvaruje gledano sa strane sistema
  - Više korisnika može da ima istu ulogu
  - Jedan korisnik može da ostvaruje više uloga
  - *When dealing with actors, it is important to think about roles rather than people or job titles (Fowler)*
- Akteri realizuju slučajeve korišćenja
  - Jedan akter može da bude uključen u više slučajeva korišćenja
  - Jedan slučaj korišćenja može da obuhvati više aktera
  - Da li je akter obavezno čovjek?
- Preporuka za velike sisteme: prvo identifikovati aktere pa onda slučajeve korišćenja za svakog aktera

# Relacija komunikacije

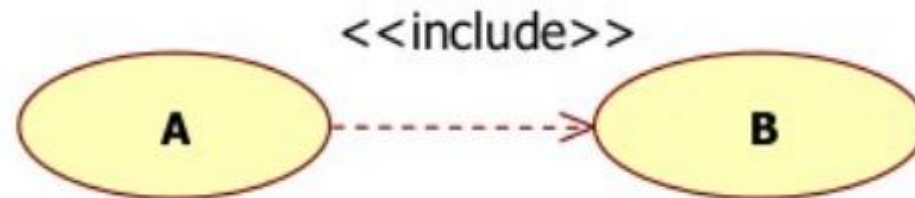
- Prikazuje se punom linijom (asocijacija)



- Komunikaciju može inicirati akter ili slučaj korišćenja (bidirekciona veza)
- Relacija je dozvoljena između:
  - Aktera i slučaja korišćenja
  - Dva slučaja korišćenja koja se ne odnose na isti subjekat
- Multiplikativnost  $>1$  na strani aktera
  - Za pokretanje slučaja korišćenja potrebno je više aktera (konkurentno ili sekvencijalno)

# Relacije između slučajeva korišćenja

- Relacija uključivanja *include*
  - Postoji niz koraka koji su zajednički za više slučajeva korišćenja

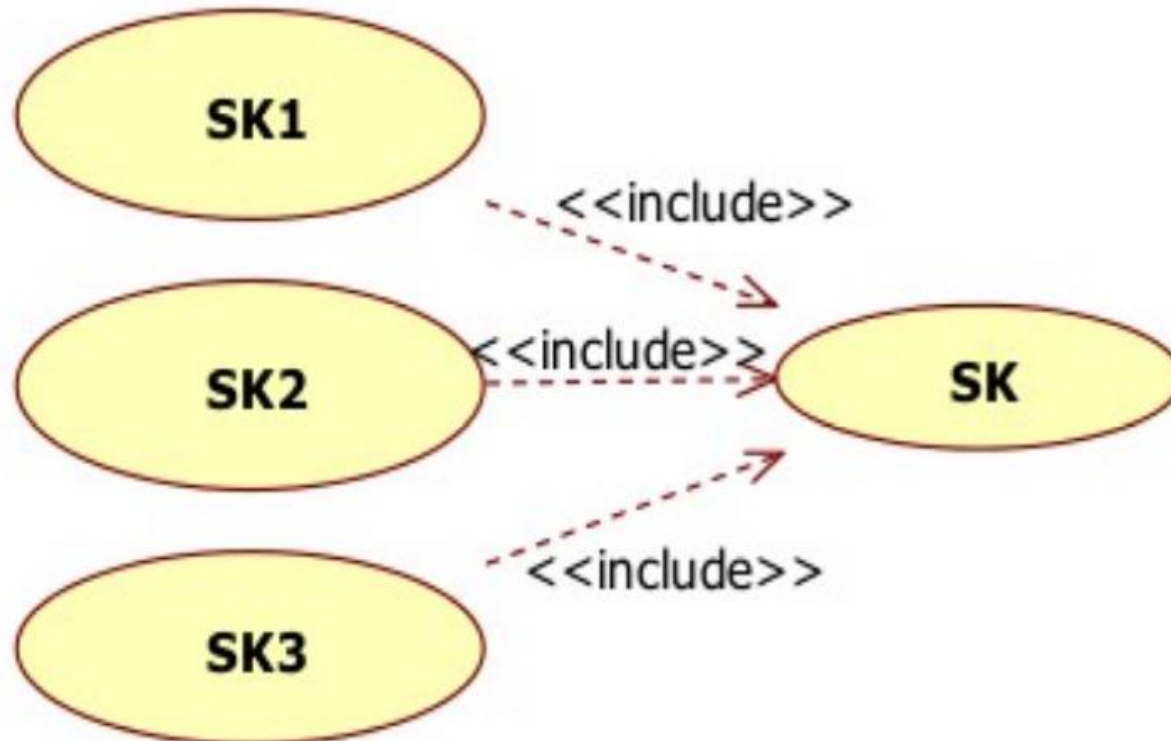


- Relacija uključivanja od slučaja korišćenja A prema slučaju korišćenja B ukazuje da će slučaj korišćenja A uključiti i ponašanje slučaja korišćenja B
- Ponašanje opisano u B obavezno je za A

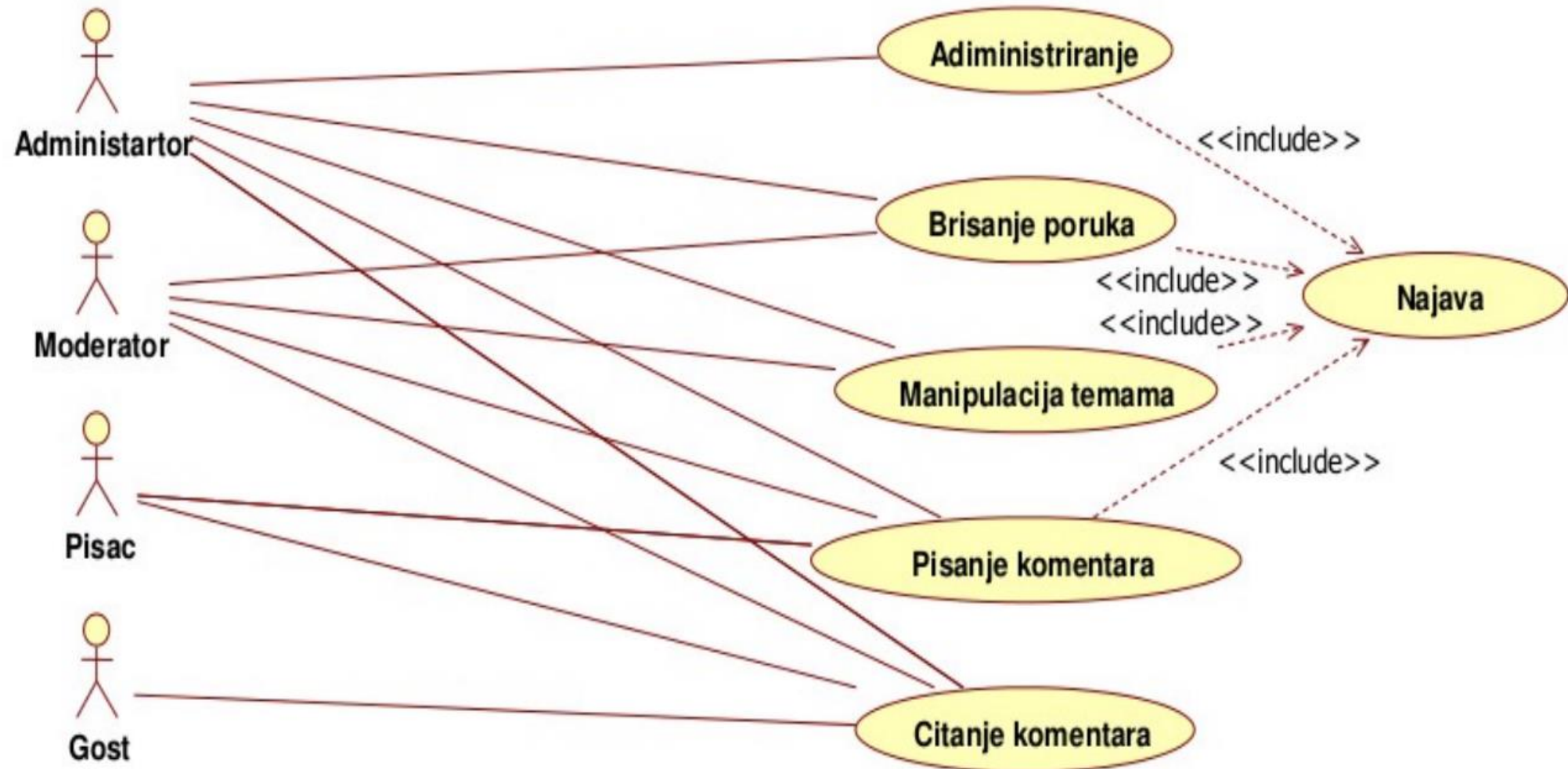


# Primjer za relaciju include

- Slučajevi korišćenja SK1, SK2 i SK3 uključuju ponašanje SK

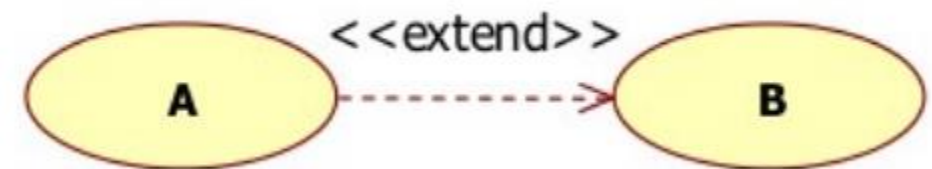


# Primjer za relaciju include



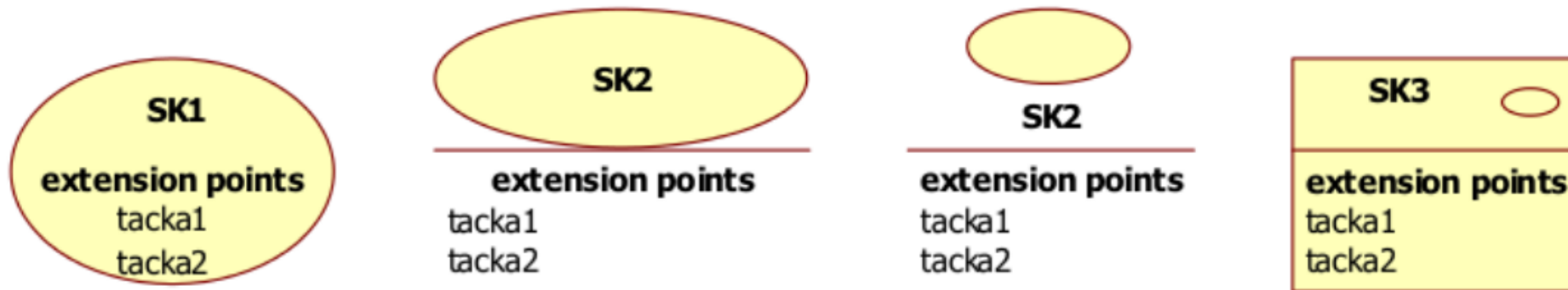
# Relacije između slučajeva korišćenja

- Relacija proširivanja *extend*
- Relacija proširivanja od slučaja korišćenja A prema slučaju korišćenja B ukazuje da B može obuhvatiti ponašanje specificirano u A
- Praktično B može da se proširi i ispolji cjelokupno ponašanje opisano u A
- Koristi se da se izrazi opciono ponašanje osnovnog slučaja korišćenja
  - ponašanje opisano u A je opciono, a ono u B osnovno
- Problem sa terminom stereotipa <<extend>>
  - sličnost sa ključnom riječju extends jezika Java
  - sasvim različito značenje



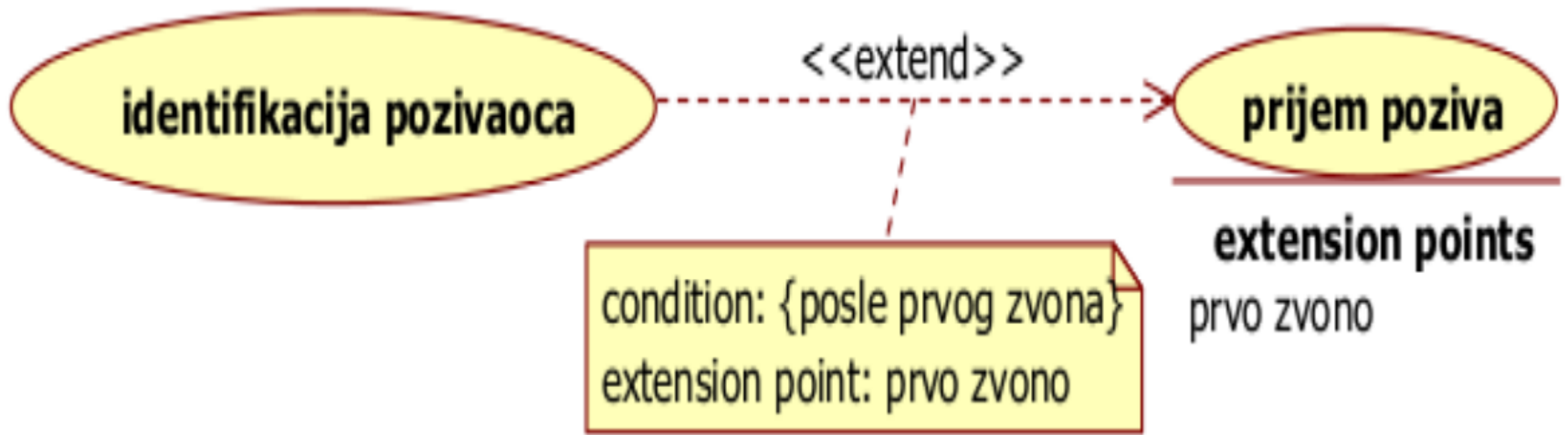
# Tačke proširenja

- Osnovni slučaj korišćenja se proširuje u određenim tačkama ponašanja
  - tačka se naziva tačkom proširenja (ekstenzije)
- Tačka se navodi po sintaksi: ime [: objašnjenje]
- Alternativne grafičke notacije:



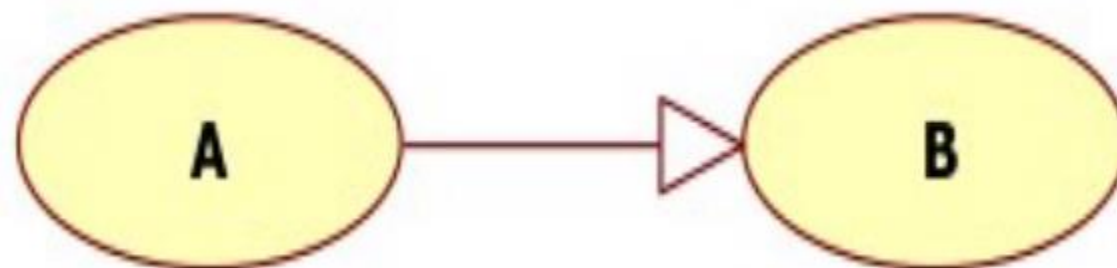
# Primjer za relaciju extend

- Identifikacija pozivaoca je opciona funkcija telefona



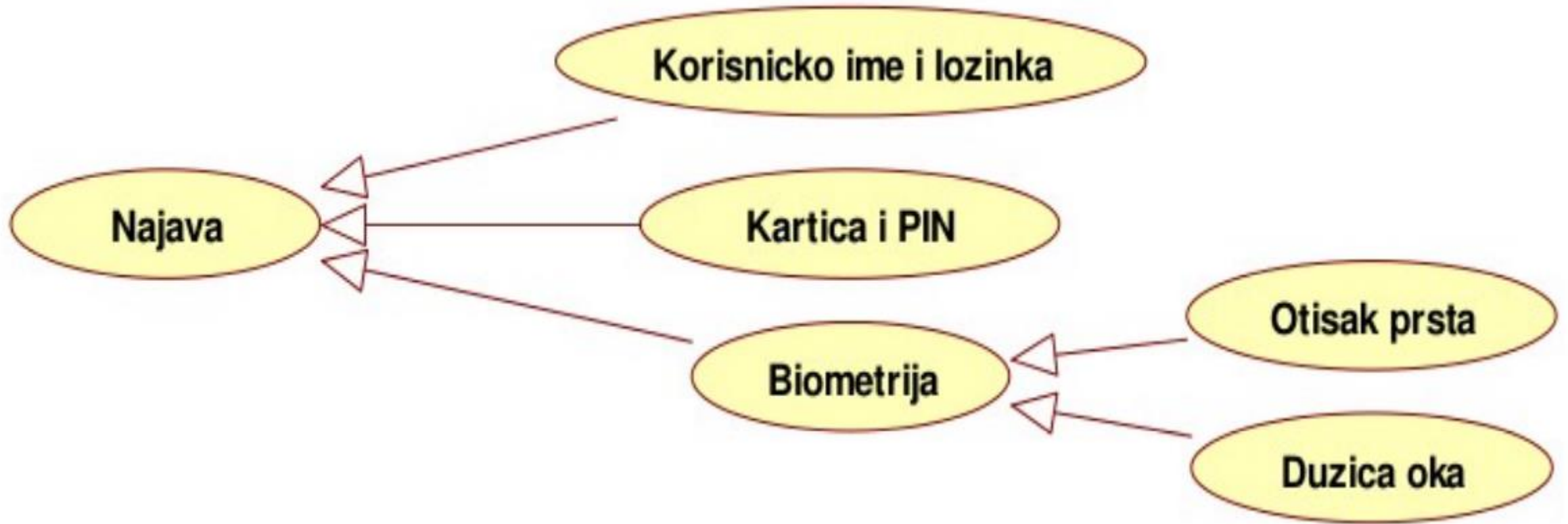
# Relacije između slučajeva korišćenja

- Relacija generalizacije



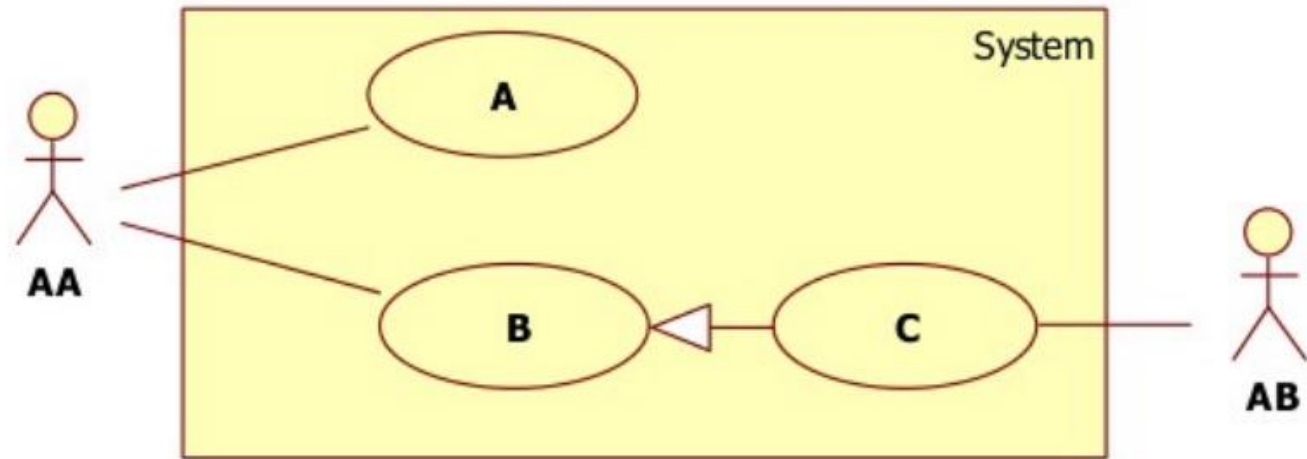
- Relacija generalizacije od slučaja korišćenja A prema slučaju korišćenja B ukazuje da je slučaj korišćenja A specifičan slučaj opštijeg slučaja B
  - Implementira alternativne scenarije

# Primjer za relaciju generalizacije



# Okvir subjekta

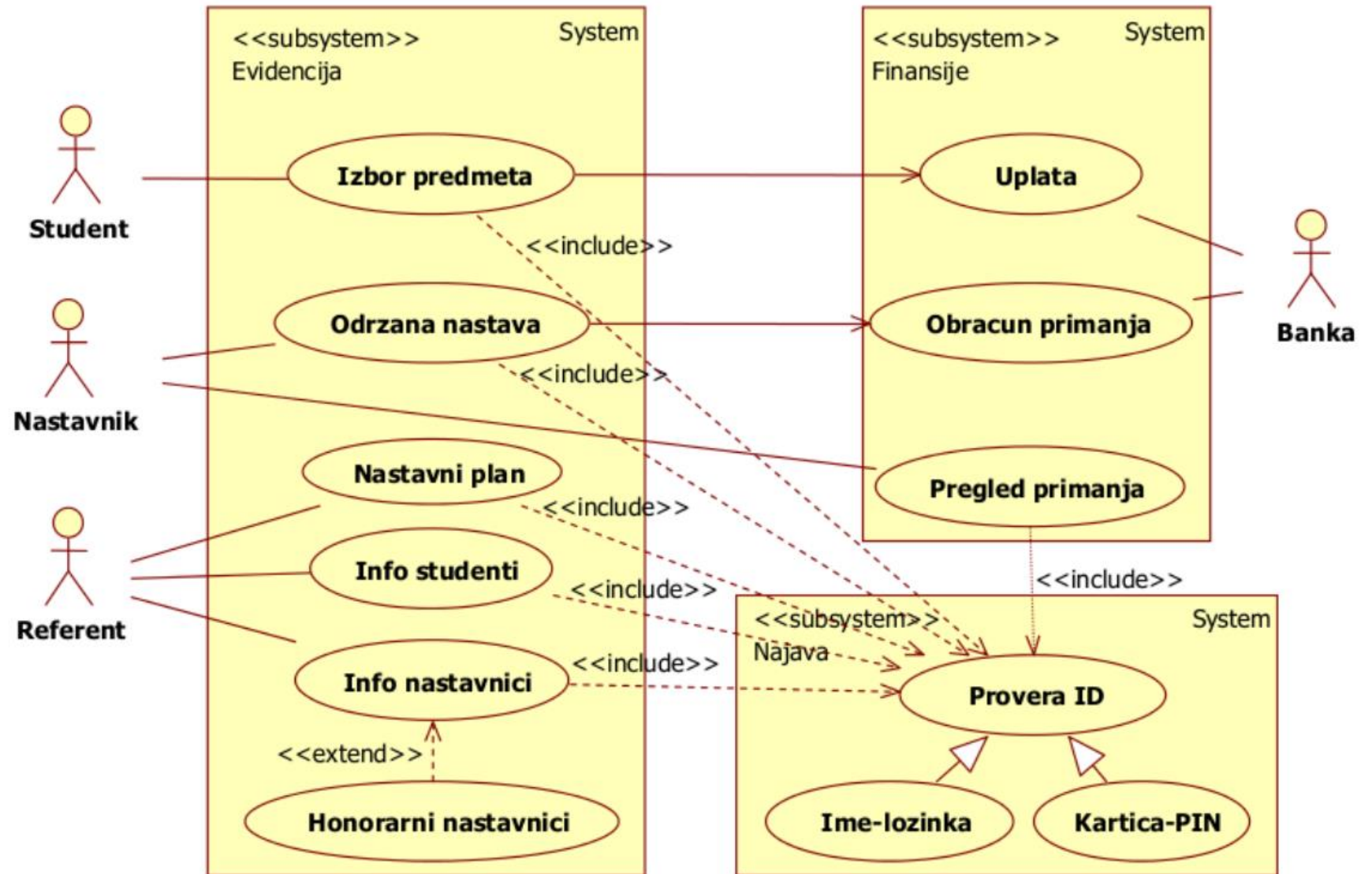
- Slučajevi korišćenja su unutar, a akteri izvan subjekta modeliranja
- Grafička reprezentacija



- Subjekt nije vlasnik slučajeva korišćenja koji predstavljaju njegove funkcionalnosti
  - Vlasnik može biti klasa, paket ili model



# Primjer – IS fakulteta



# Biznis i sistem slučajeви korišćenja

- Fowler:

- *In my work, I focus on business use cases first, and then I come up with system use cases to satisfy them. By the end of the elaboration period, I expect to have at least one set of system use cases for each business use case I have identified-at minimum, for the business use cases I intend to support in the first delivery*
- *do use cases and conceptual modeling at the same time*
- *use cases represent an external view of the system. As such, don't expect any correlations between use cases and the classes inside the system.*