

## VEŽBA 2.

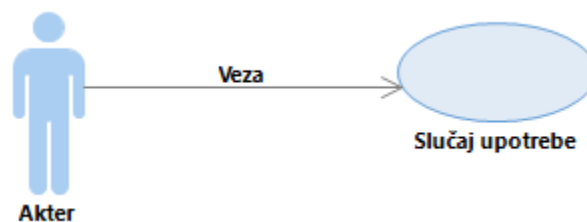
U ovoj vežbi upoznaćete se sa UseCase - *Dijagrami slučajeve korišćenja*. UseCase dijagrami biće objašnjeni kroz uvodni teorijski deo a zatim biće praktično prikazani u drugom delu vežbe.

### UseCase Dijagram

Služi kao “ugovor” između klijenta, korisnika i developera gde oni proveravaju da će sistem postati ono što su očekivali. Sastoji se od aktera (*učesnika*) i slučajeve korišćenja. Svaki slučaj korišćenja se detaljno opisuje, prikazujući korak po korak kakva je interakcija sistema sa akterima i šta sistem radi u slučajevima korišćenja. UseCase Dijagram omogućava krajnjim korisnicima da razumeju sistem, pogled korisnika na funkcionisanje sistema (*šta sistem radi, a ne kako funkcioniše*). Razvoj dijagrama slučajeve korišćenja definiše se sledećim aktivnostima:

- Definisanjem učesnika
- Definisanjem slučajeve korišćenja (*upotrebe*)
- Definisanjem tipova veza između učesnika i slučajeve korišćenja
- Izradom dijagrama slučajeve korišćenja

Elementi dijagrama:



**Učesnik/Akter** – osoba ili veštački entitet (*softver ili sistem*) koji učestvuje u slučaju upotrebe. Korisnik je čovek koji koristi sistem, dok je učesnik specifična uloga koju korisnik ima u komunikaciji sa sistemom. Učesnika je moguće identifikovati na osnovu odgovora na sledeća pitanja:

- Ko će koristiti osnovnu funkcionalnost sistema (primarni učesnici)?
- Ko treba da upravlja, administrira i održava sistem (sekundarni učesnici)?
- Kome će biti potrebna podrška sistema u obavljanju dnevnih zadataka?
- Kojim hardverskim uređajima sistem treba da upravlja?
- Sa kojim drugim sistemima dotični sistem treba da bude u vezi?
- Ko ili šta je zainteresovan za rezultate koje sistem proizvodi?

**Slučaj korišćenja/Slučaj upotrebe** – definiše funkcionalnost sistema sa stanovništva učesnika – šablon ponašanja delova sistema. Pitanja za učesnika – identifikuju slučajeve upotrebe:

- Koje funkcije učesnik zahteva od sistema – šta učesnik treba da radi?
- Da li učesnik treba da čita, kreira, briše, izmeni ili da unese neke informacije u sistem?
- Da li učesnik treba da bude obavešten o događajima u sistemu?
- Da li svakodnevni rad učesnika može da se pojednostavi kroz nove funkcije sistema?

### **Prednosti UseCase modela**

Komunikacija sa krajnjim korisnicima i ekspertima:

- Osigurava obostrano razumevanje zahteva
- Omogućava odobravanje projekta u ranim fazama razvoja sistema

Identifikacija korisnika sistema i onoga šta sistem treba da radi:

- Zahtevi interfejsa sistema

Verifikacija svih prikupljenih zahteva:

- Osigurava da razvojni tim razume zahteve

### **Tipovi veza**

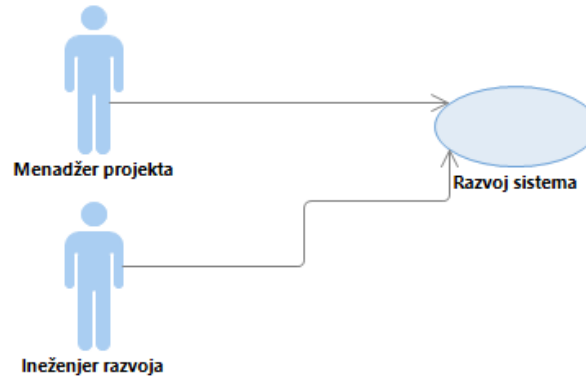
Veze koje se uspostavljaju u dijagramu slučajeve upotrebe su:

- Veza pridruživanja (*asocijacije*)
- Veza uključivanja <<include>>
- Veza proširivanja <<extend>>
- Veza generalizacije

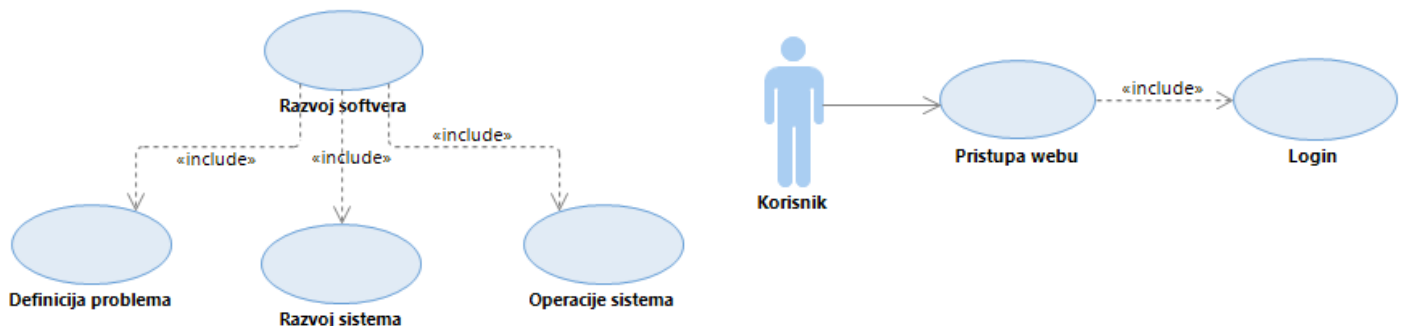
Uočavaju se relacije između:

- Aktera i slučaja upotrebe
- Relacije između samih slučajeve upotrebe
- Relacije između aktera

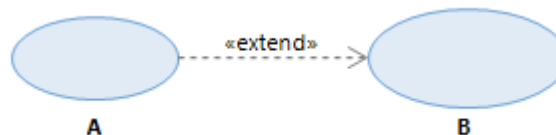
**Veza** – linija koja spaja učesnike i slučajeve upotrebe. **Asocijacija** između samih učesnika ili slučajeve upotrebe, definiše povezanost tih elemenata.



**Veza <<include>>** opisuje odnos između slučajeve upotrebe u kojem jedan slučaj upotrebe koristi usluge drugog

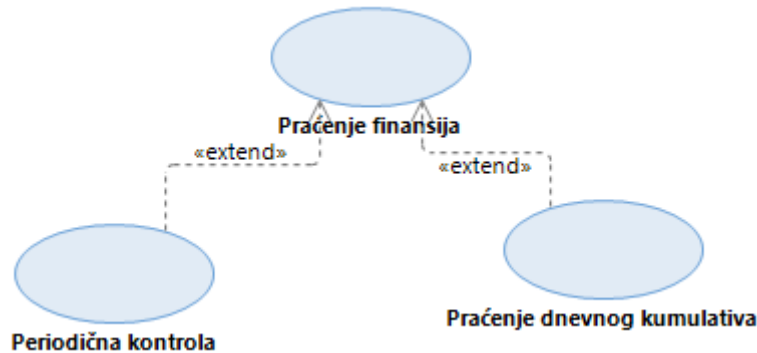


**Veza <<extend>>**, proširivanjem jednog slučaja upotrebe opisuje se neka složenija funkcija sistema. Proširivanje se vrši sa jednim ili više drugih postojećih slučajeve upotrebe



Ako slučaj **A** proširuje slučaj **B**:

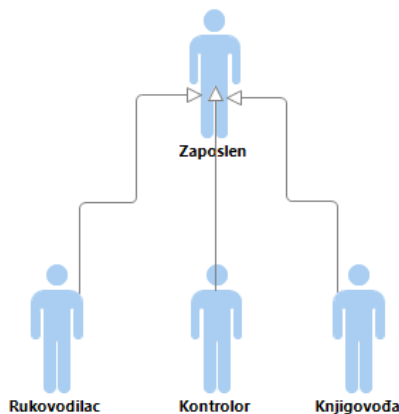
- I slučaj **A** i slučaj **B** mogu da postoje sami
- Slučaj **B** može (*a ne mora*) da bude proširen slučajem **A**



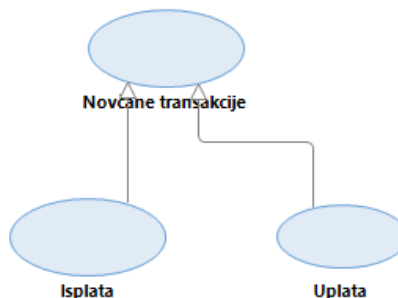
## Generalizacija

Predstavlja vezu između roditelja i deteta, vezana je pojmom nasleđivanja, dete nasleđuje osobine roditelja.

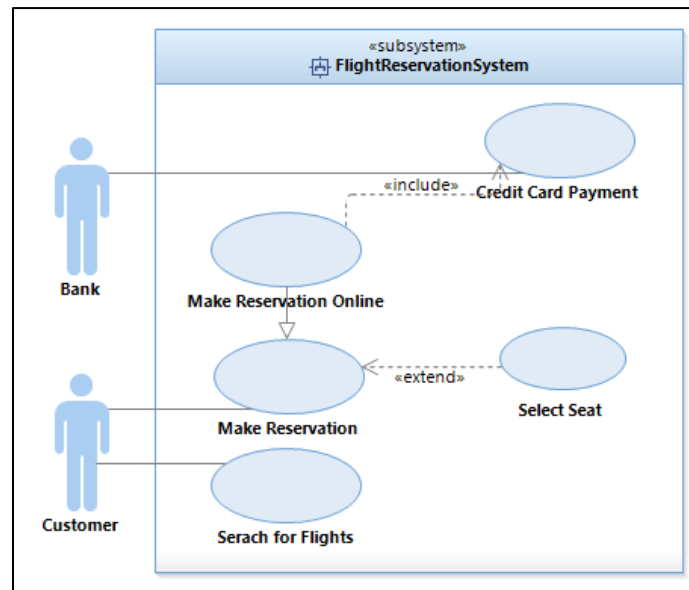
**Generalizacija učesnika** – izvedeni učesnik ima sve osobine i ponašanje osnovnog (*apstraktong*) učesnika, ali može dodati osobine ili redefinisati ponašanje




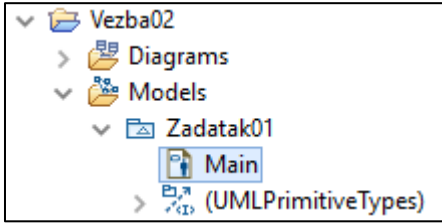




**Generalizacija slučajeja korišćenja** – izvedeni slučaj korišćenja ima sve osobine i ponašanje apstraktnog slučaja upotrebe, ali može dodati osobine ili redefinisati ponašanje.

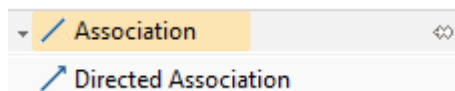
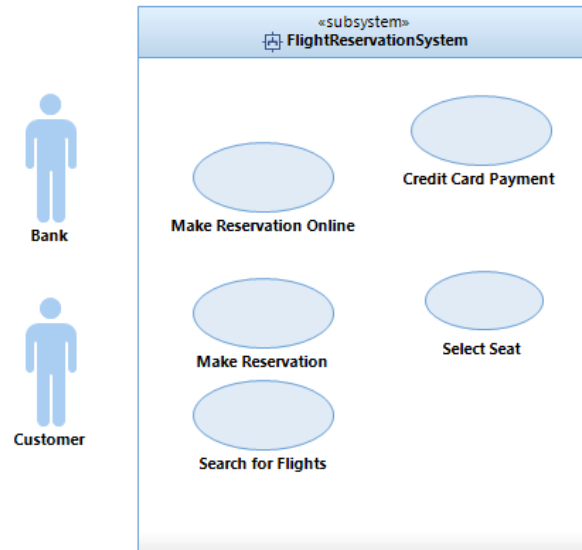


**Zadatak 1: Pomoću Rational Software Architect nacrtati sledeći UseCase dijagram**



**Rešenje:**

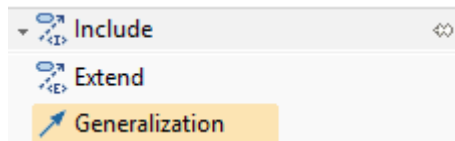
1. Izaberite **File > New > Project**
  2. Proširite folder **Modeling** i izaberite  **UML Project**
  3. Klikom na **Next** nazovite projekat **Vežba02, Next**
  4. Izaberite sledeće:
    - **Categories:** Requirements
    - **Templates:** Blank Use Case Package
    - **File name:** Zadatak01
- 
5. Klikom na **Finish** otvoriće vam se prazna radna površina **Main** prozoru
  6. Iz  **Palette** (desna strana ekrana) izaberite  **Actor** i kliknite na radnoj površini u **Main** prozoru
  7. Unesite dva **Korisnika (Actor)** i jednog nazovite **Bank**, a drugog **Customer**
  8. Zatim izaberite  **Subsystem** i kliknite na radnu površinu, **Subsystem** nazovite **FlightReservationSystem**
  9. Možete da resize prozor **Subsystema** tako što ga selektujete na radnoj površini i raširite na odgovarajuću veličinu
  10. Izaberite  **Use Case** i prevucite unutar kontejnera **Subsystem**, ovo ponovite pet puta
  11. Nazovite svaki slučaj korišćenja (*Use Case*) odgovarajućim imenom (*Make Reservation Online, Make Reservation, Search for Flights, Select Seat, Credit Card Payment*)



12. Izaberite **Association** pa **Association**

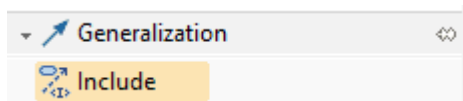
13. Mišem povežite **Bank** i **Credit Card Payment**

14. Povežite **Customer** sa **Make Reservation** i **Search for Flights**



15. Izaberite **Generalization** pa birajte **Generalization**

16. Povežite **Make Reservation Online** sa **Make Reservation**



17. Birajte **Include** i izaberite **Include**

18. Povežite **Make Reservation Online** sa **Credit Card Payment**

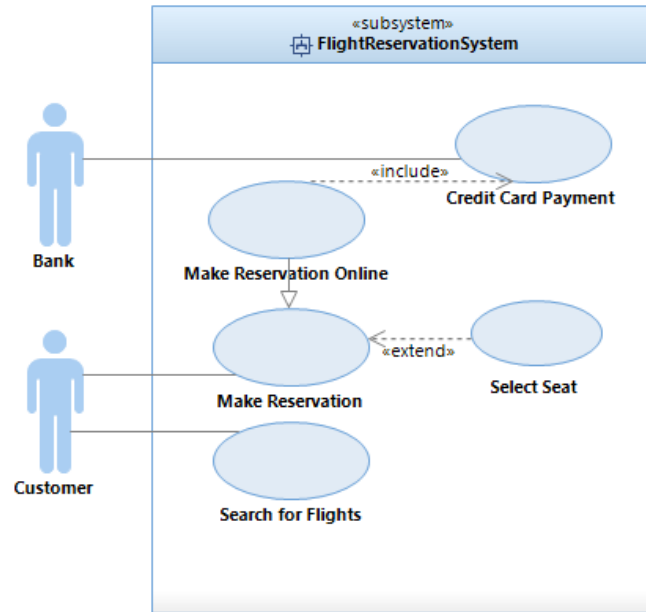


19. Izaberite **Extend** i birajte **Extend**

20. Povežite **Select Seat** sa **Make Reservation**

21. Sačuvajte zadatak, **File > Save**

Po završetku zadatka trebali biste da imate sledeći UseCase dijagram:





## Zadatak 2:

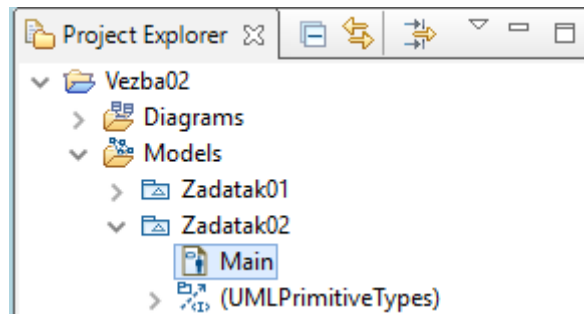
Menadžeri osiguravajuće kompanije, u cilju povećanja broja auto kasko osiguranja su odlučili da ponude kasko osiguranje na Web sajtu.

Analizom zahteva ustanovljeno je da se budući sistem koji obezbeđuje izračunavanje premije (*glavni Use Case*) mora sastojati od sledećih funkcionalnih celina:

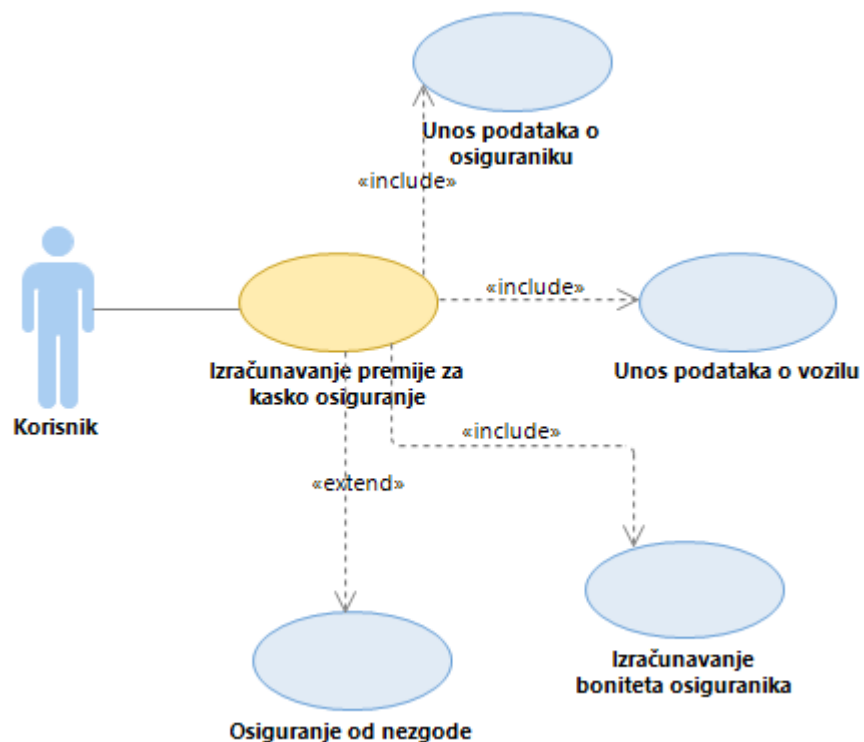
- Unosa podataka o osiguraniku
- Unosa podataka o vozilu
- Izračunavanje boniteta osiguraniku
- Izbor opcije za osiguranje za slučaj nezgode vozača i saputnika u vozilu

## Rešenje:

1. U  **Project Explorer** desnim klikom na  **Models** izaberite **Create Model**
2. Izaberite **Standard Template** (*default*) i izaberite **Next**
3. Kao pri kreiranju **Zadatak01** ostavite **Categories** i **Template** da budu isti, nazovite u **File name:** Zadatak02
4. Klikom na **Finish** otvara se nov prozor **Main**



5. Iz predhodno stečenog znanja u **Zadatak01** kreirati sledeći UseCase dijagram:



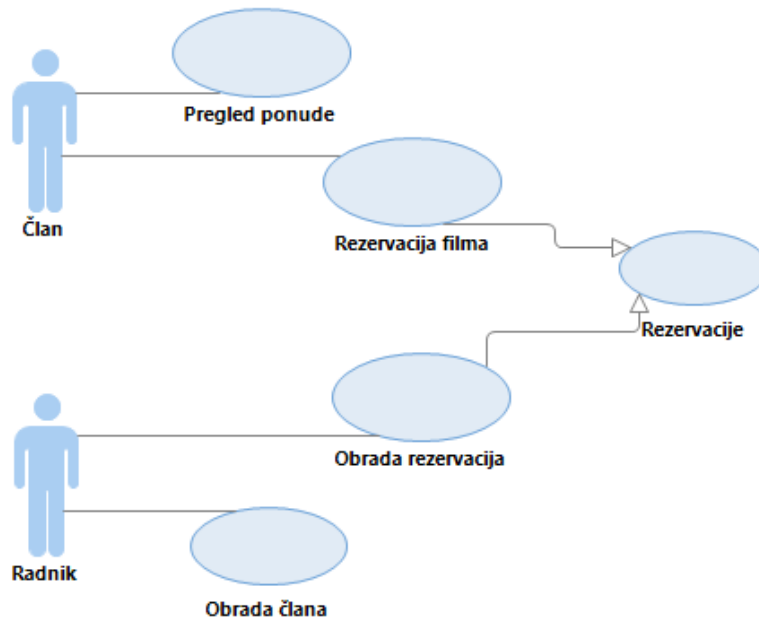


### Zadatak 3: Video Klub

Nacrtati dijagram slučajeve korišćenja koji će prikazati funkcionisanje video kluba, posmatramo učesnike **Član** i **Radnik** u video klubu.

#### Rešenje:

Kreirati sledeći UseCase dijagram:



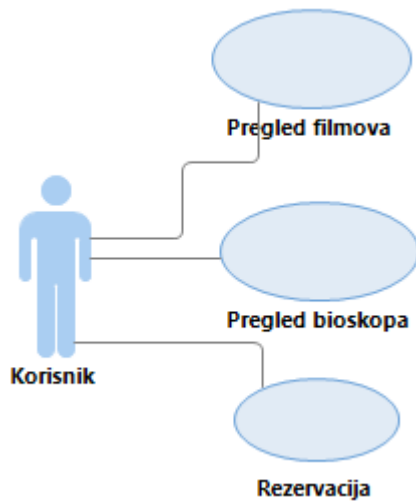
### Zadatak 4: Rezervacija karata putem interneta

Nacrtati dijagram slučajeve korišćenja koji će omogućiti korisniku da preko Interneta rezerviše bioskopske ulaznice za željene projekcije. Takođe je potrebno omogućiti korisniku stalan uvid u bioskopski repertoar i informacije o bioskopu u kojem željeni film igra.

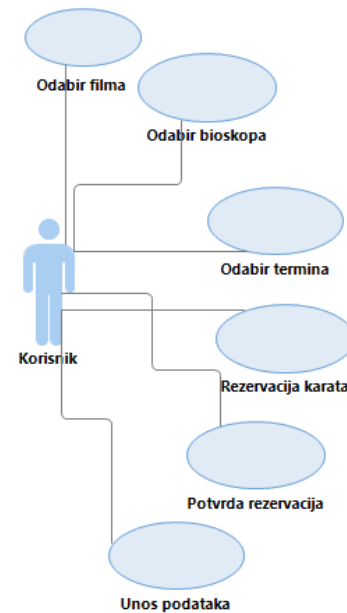
Nacrtati još jedan dijagram slučajeve upotrebe procesa rezervacija.

Rešenje:

Dijagram slučajeja upotrebe

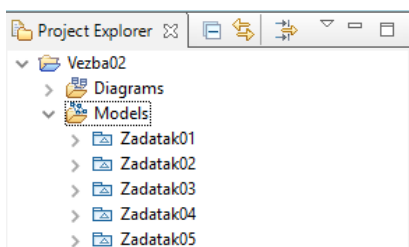


Dijagram slučajeja upotrebe procesa rezervacija



### Zadatak 5: Po izboru

Nacrtati dijagram slučajeja upotrebe za aktivnosti po izboru. Minimum 2 učesnika i minimum 4-5 slučajeva upotrebe.



Po završetku sačuvati projekat, **Save all**.