

# Model primerov uporabe

Osnovni namen modelov primerov uporabe je učinkovito zajemanje in predstavitev zahtev uporabnika. V osnovi predstavlja model primerov uporabe množico opisanih primerov uporabe – je model funkcionalnosti sistema in okolja. Primeri uporabe so tekstovni dokumenti (ne diagrami) in modeliranje primerov uporabe predstavlja pisanje besedila (ne risanje diagramov). Seveda za zajem zahtev model primerov uporabe ni edini izdelek v procesu razvoja. Med izdelke za zajem zahtev sodijo še dodatne specifikacije, slovarji, vizija in opisi poslovnih pravil.

Model primerov uporabe vsebuje tudi diagram primerov uporabe (diagram UML), kjer so prikazana imena primerov uporabe, akterjev in njihovih povezav. Diagrami primerov uporabe povežejo uporabnike sistema in procese, ki v sistemu tečejo. Takšen diagram omogoča hitro in enostavno razumevanje seznama vseh primerov uporabe.

Modeli primerov uporabe so osnova za objektno modeliranje, kjer je omogočeno sodelovanje uporabnika sistema skozi celoten razvojni cikel. Model primerov uporabe služi kot osnova pri:

- definiranju funkcionalnih zahtev
- identificiranju in določanju lastnosti objektov
- definiranju interakcije med objekti in vmesniki objektov
- testiranju
- komunikaciji z uporabnikom.

## 1. Primeri uporabe

Primeri uporabe so dober način zapisa zahtev, ki je dovolj enostaven, da ga lahko uporabijo tudi domenski eksperti (ki ne poznajo OO konceptov, OO analize in načrtovanja) za zapis in je kot takšen zapis dovolj dober za analitike in načrtovalce rešitev. Primeri uporabe sami po sebi niso objektno-orientirani, saj se ne izvaja OO analiza, ko jih pišemo. So pa osnova v OO analizo in načrtovanje.

Druga vrednost primerov uporabe je, da izpostavijo cilje in perspektive uporabnika. Pri pisanju primerov uporabe nas zanimajo predvsem odgovori na vprašanja kot so:

1. kdo uporablja sistem
2. kakšni so njihovi tipični scenariji uporabe
3. kakšni so njihovi cilji uporabe

Primeri uporabe so zahteve, v osnovi zahteve funkcionalnosti in obnašanja, ki nakazujejo, kaj bo sistem delal. Lahko rečemo tudi, da so primeri uporabe pogodbe, v katerih je definirano, kako se bo sistem obnašal.

Akter je zunanja entiteta, ki na kakršenkoli način komunicira s sistemom. Akter lahko predstavlja vlogo posameznika, sistema ali stroja izven sistema. V opisih primerov uporabe se pojavijo različni tipi akterjev:

- **primarni akter** – akter, ki sproži posamezne primere uporabe. Je akter z dejanskimi cilji uporabe sistema.
- **sekundarni akter** – zagotavlja določeno storitev. Pogosto je to primer kakšnega informacijskega sistema, s katerim med izvajanjem komunicira naš sistem.

## 2. Opisi primerov uporabe

Za vsak primer uporabe za natančnejše razumevanje izdelamo opise potekov oz. scenarijev.

Opis primera uporabe je lahko v obliki:

- **strukturiranega besedila** - primer uporabe vsebuje določen tok dogodkov skozi naš sistem. Potek dogodkov opišemo s pomočjo strukturiranega besedila, kjer navedemo zaporedje dogodkov. Pri opisu scenarija izvedbe primera uporabe se lahko pojavijo alternative - odstopanja od osnovnega toka podatkov. Za alternative natančno opišemo, kaj se je zgodilo in kako se tok dogodkov spremeni.
- **ustrezne diagramске tehnike** (diagram sodelovanja, diagram zaporedja, diagram aktivnosti) - namesto strukturiranega besedila lahko za lažje razumevanje uporabimo diagram zaporedja: identificiramo "subjekte", ki so odgovorni za izvajanje scenarija in izrišemo časovno zaporedje dogodkov, ki tečejo med udeleženi "subjekti"

Za tekstovni opis primerov uporabe se lahko uporabijo različne predloge. Primer takšne predloge je:

<b>PU#</b>	< enolično ime, ki podaja cilj primera uporabe >	
<b>Cilj in kontekst</b>	<daljši opis cilja in konteksta izvajanja primera uporabe >	
<b>Sistem in nivo</b>	<Kateri sistem opisuje primer uporabe > <na kakšnem nivoju je primer uporabe >	
<b>Predpogoji</b>	<pričakovanja stanja sistema pred pričetkom izvajanja primera uporabe >	
<b>Pogoj uspešnega zaključka</b>	<Stanje sistema po uspešnem zaključku primera uporabe >	
<b>Pogoj neuspešnega zaključka</b>	<Stanje sistema, če primer uporabe ni uspešno zaključen >	
<b>Primarni in sekundarni akterji</b>	<vloga in opis primarnega akterja >. <drugi sistemi, PU, ki so potrebni za izvajanje tega PU >	
<b>Prožilec</b>	<akcija, na podlagi katere sistem sproži PU >	
<b>SCENARIJ</b>	<b>korak</b>	<b>Akcija</b>
	1	<opis akcije sistema >
	2	<... >
	3	
<b>Razširitve</b>	<b>korak</b>	<b>Vejitvena akcija</b>
	1a	<pogoj, ki sproži vejitev > : <akcija ali izvajanje odvisnega primera uporabe >
<b>Podvariacije</b>		<b>Vejitvena akcija</b>
	1	<seznam variacij >

<b>Druge informacije</b>	<Ime primera uporabe>
<b>Pririteta:</b>	<kako pomemben je PU za sistem/oganzizacijo>
<b>Performance</b>	<predvidena zgornja časovna meja izvajanja PU>
<b>Frekvenca</b>	<predvidena frekvenca izvajanja PU>
<b>Komunikacija</b>	<načini komunikacije – interaktivni vmesnik, PB...>
<b>Rok</b>	<predviden datum implementacije PU>
<b>Podrejeni PU</b>	<seznam vključenih PU>
<b>Nadrejeni PU</b>	<seznam PU, ki ta primer uporabe uporabljajo >

### 3. Diagrami primerov uporabe

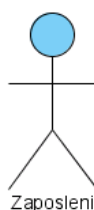
Diagrami primerov uporabe pomagajo pri definiranju systemskega in kontrolnega načrtovanja kot sredstvo za zajemanje in sledenje zahtevam, komunikacijo s končnim uporabnikom in strankami, razgradnji sistema in ocenjevanju velikosti projekta in potrebnih virov. S pomočjo diagramov primerov uporabe modeliramo vplive okolja na sistem. Sistem modeliramo kot množico procesov, ki uporabniku ponujajo usluge (storitve) sistema. Hkrati okolje sistema modeliramo kot množico akterjev, ki vplivajo na sistem (oz. od sistema zahtevajo merljiv rezultat).

Gradniki diagrama primerov uporabe so:

- Akter ("actor")
- Primer uporabe ("use case")
- Povezave med akterjem in primerom uporabe
- Povezave med primeri uporabe
- Sistem ali podsistem («subject»)

#### Akter ("actor")

Akter definira vlogo posameznika ali drugega sistema, ki je v interakciji s sistemom, ki ga opisujemo. Interakcija med sistemom in akterjem pomeni izmenjavo signalov, podatkov,... Akter je zunanja entiteta, ki komunicira s sistemom. Akterje lahko predstavljajo vloge, ki jih igrajo človeški uporabniki, informacijski sistemi, zunanje strojne naprave, ali drugi predmeti. Pomembno je tudi dejstvo, da akter ni točno določena fizična entiteta znotraj sistema, ampak predstavlja razred uporabnikov z isto vlogo.



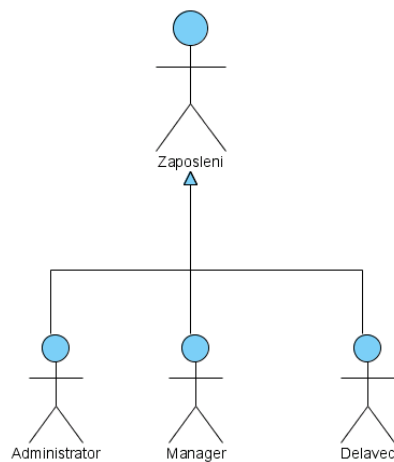
Slika 1: Simbol za akterja v diagramu primerov uporabe

Slika 1 prikazuje simbol za notacijo akterja. Akter mora imeti definirano ime, ki je ponavadi pod simbolom akterja. Akter ima lahko povezavo asociacije s primeri uporabe, komponentami in razredi. Za prikaz akterjev na diagramu primerov uporabe je sicer dovoljena tudi uporaba drugih simbolov (ikone, ki so bolj skladne s tem, kakšne vrste je akter).



**Slika 2: Drug simbol za prikaz akterja**

Slika 3 prikazuje primer povezave generalizacije med akterji. S pomočjo generalizacije lahko izpostavimo splošnega akterja oz. vlogo, ki jo predstavljajo vsi izpeljani akterji. Vsi izpeljani akterji pa lahko komunicirajo še z drugimi primeri uporabe, s katerimi ostali ne komunicirajo.



**Slika 3: Generalizacija akterjev**

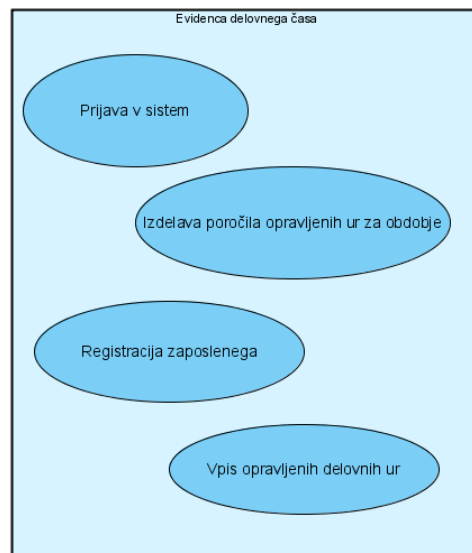
## Primer uporabe ("use case")

Primer uporabe predstavlja zaporedje transakcij v sistemu, ki akterju vrača merljive rezultate. Opisuje možen potek interakcije med sistemom in enim ali več akterji. Vsak primer uporabe vsebuje množico možnih scenarijev. Za natančno razumevanje sistema je potrebno naštetih vse primere uporabe. Primeri uporabe so primerni tako za definiranje zunanjih zahtev sistema (torej funkcionalnosti, ki jih sistem potrebuje za izvajanje posameznih funkcionalnosti) kot tudi za definiranje funkcionalnosti, ki jih sam sistem omogoča.



**Slika 4: Nekaj primerov uporabe**

Primer uporabe narišemo v obliki eklipse (Slika 4), kjer je ime primera uporabe napisano ali v eklipsi ali pod simbolom. Primere uporabe lahko narišemo znotraj okvirja, ki predstavlja sistem.



**Slika 5: Okvir, ki predstavlja sistem**

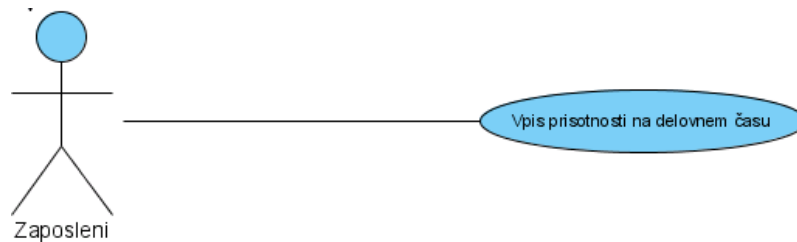
Obnašanje oz. izvajanje primera uporabe se lahko opiše s specifikacijo (tekstovno, diagramске tehnike, ...).

## Povezave med akterji in primeri uporabe

Edina povezava med akterjem in primerom uporabe je asociacija, ki pomeni udeležnost oz. komunikacijo med primerkom akterja in primerkom primera uporabe. Sodelovanje akterja v primeru uporabe prikazemo z neprekinjeno povezavo med simbolom akterja in simbolom primera uporabe. Asociacijo med akterjem in primerom uporabe imenujemo relacija "komunicira" ("communicate"). Priporočljivo je uporabljati eno-smerne povezave, ker na tak način lahko določimo:

- **Aktivne akterje** - Akter se smatra za aktivnega, če v paru (akter, primer uporabe) sproži izvajanje primera uporabe. Puščica relacije komunikacija je usmerjena v primer uporabe.

- **Pasivne akterje** - Akter je pasiven, če je primer uporabe tisti, ki sproži komunikacijo med primerom uporabe in akterjem. Pasivni akterji so ponavadi zunanji sistemi ali naprave, s katerimi komunicira naš sistem. Puščica je usmerjena od primera uporabe v akterja.



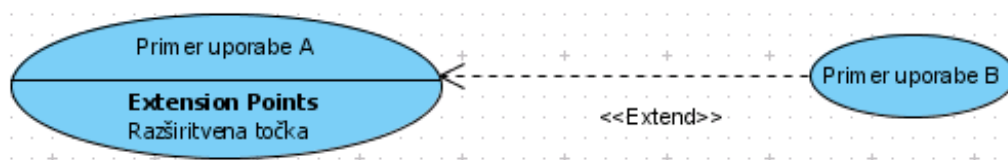
Slika 6: Primer povezave "komunicira" (zaposleni je aktivni akter, ki sproži proces vpisa ur prisotnih na delovnem mestu)

## Povezave med primeri uporabe

### 1. Povezava "razširja" ("extend")

Povezava razširja od primera uporabe A do primera uporabe B pomeni, da izvajanje primerka primera uporabe B lahko vsebuje (ni pa obvezno) obnašanje, ki je vsebovano v primeru uporabe A. To pomeni, da primer uporabe A lahko razširja tok dogodkov primera uporabe B (A extend B). Primer uporabe, ki ga razširjamo (B), je opisan kot samostojna enota in se lahko izvede tudi brez izvajanja primera uporabe, ki ga razširja (A).

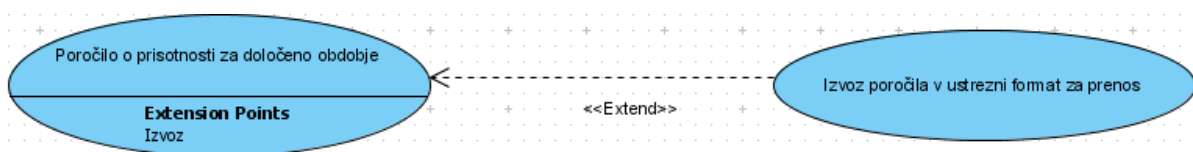
Povezavo razširja prikazemo z usmerjeno črtkano povezavo od primera uporabe, ki zagotavlja razširitev, do osnovnega primera uporabe. Puščica se označi s ključno besedo <<Extend>>. Pogoji za izvajanje razširitev se lahko opcijsko prikažejo v simbolu za opombo.



Slika 7: Ponazoritev povezave Extend

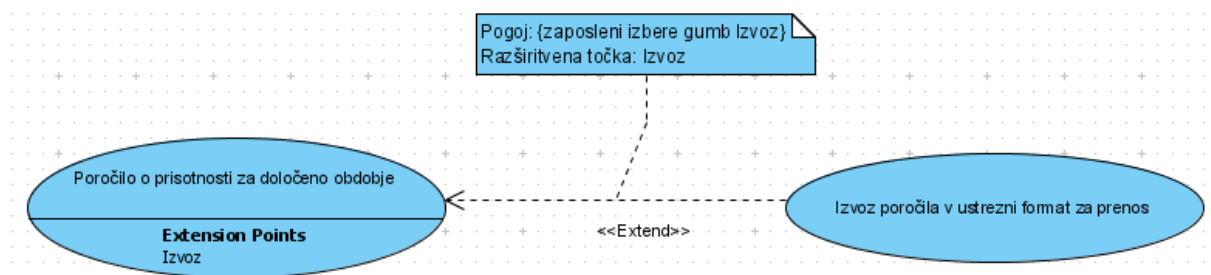
### Razširitvena točka (iz primerov uporabe)

Oz. z drugimi besedami, je lastnost primera uporabe, ki identificira točko, kjer se obnašanje primera uporabe lahko poveča z elementi drugega (razširitvenega) primera uporabe. Razširitvena točka predstavlja torej referenco na lokacijo znotraj primera uporabe, kamor so lahko vključeni elementi obnašanja drugega primera uporabe.



Slika 8: Primer razširitvene točke

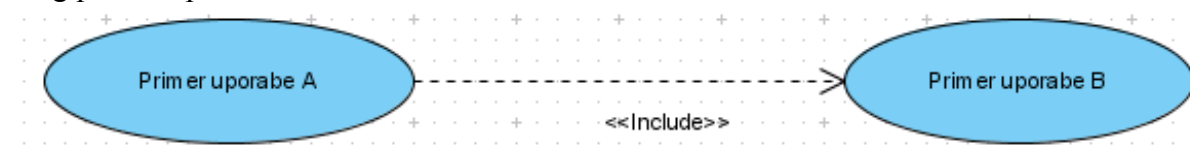
Slika 8 prikazuje primer razširitvene točke, kjer primer uporabe »Poročilo o prisotnosti za določeno obdobje« vsebuje razširitveno točko »Izvoz«. Ta primer uporabe je razširjen skozi razširitveno točko »Izvoz« s primerom uporabe »Izvoz poročila v ustrezni format za prenos« vedno, ko se izvajanje PU »Poročilo o prisotnosti za določeno obdobje« nahaja v točki, ki je označena z »Izvoz« in uporabnik pritisne na gumb »Izvoz«. Oba primera uporabe sta opisana ločeno. Razširitveno točko lahko na diagramu podrobneje opišemo tudi s pomočjo simbola za opombo, v katerem podrobneje definiramo korak, ki je potreben, da se izvede razširitveni primer uporabe. Primer takšne notacije prikazuje



Slika 9: Podrobneje opisan pogoj razširitvene točke

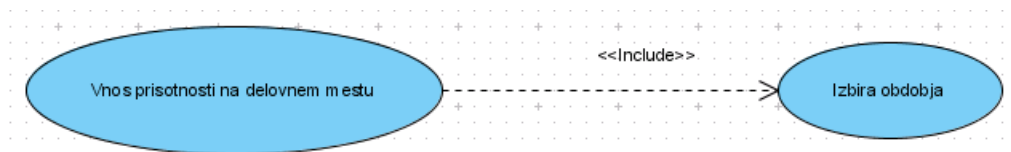
## 2. Povezava "vključuje" ("include")

Povezava vključuje med primeroma uporabe A in B določa, da primerka primera uporabe A v svojem delovanju vsebuje tudi delovanje primerka primera uporabe B (A include B). Povezavo prikažemo kot usmerjeno povezavo od primera uporabe, ki vključuje (uporablja) drug primer uporabe.



Slika 10: Ponazoritev povezave vključuje

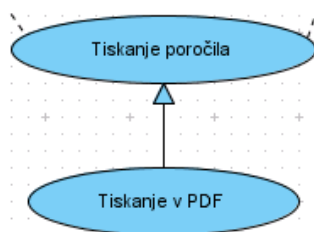
Ta povezava se uporablja na primer, ko imamo dva ali več primerov uporabe, ki vsebujejo skupno obnašanje, ki ga izločimo, opišemo v ločenem primeru uporabe in ga lahko vključijo vsi osnovni primeri uporabe, ki imajo ta skupni del. Glede na to, da je povezava vključuje namenjena za ponovno uporabo skupnih delov, to, kar ostane v osnovnem primeru uporabe, ponavadi ni celota in je odvisno od vključenega dela. Tudi sama usmeritev povezave nakazuje, da je osnovni primer uporabe odvisen od dodatnega primera uporabe, kar pa ne velja obratno. Izvajanje vključenega primera uporabe je analogno klicu neke metode ali funkcije znotraj neke metode ali funkcije, kar pomeni, da je vso obnašanje vključenega primera uporabe izvede na eni lokaciji osnovnega primera uporabe, preden se nadaljuje osnovni primer uporabe.



Slika 11: Primer povezave vključuje

### 3. Povezava generalizacija

Generalizacija med primeroma uporabe A in B določa, da je primer uporabe A specializacija primera uporabe B.



**Slika 12: Primer povezave generalizacija med primeru uporabe**

Primer povezave generalizacija prikazuje Slika 12.