

Univerza
v Ljubljani
Fakulteta
*za gradbeništvo
in geodezijo*

*Janova 2
1000 Ljubljana, Slovenija
telefon (01) 47 68 500
faks (01) 42 50 681
fgg@fgg.uni-lj.si*



Univerzitetni program Geodezija,
smer Geodezija

Kandidatka:

Katja Bajec

Modeliranje raznih postopkov v katastrih nepremičnin s pomočjo UML diagramov

Diplomska naloga št.: 693

Mentor:

izr. prof. dr. Radoš Šumrada

Somentor:

viš. pred. dr. Miran Ferlan

Ljubljana, 6. 10. 2006

POPRAVKI

Stran z napako

Vrstica z napako

Namesto

Naj bo

IZJAVA O AVTORSTVU

Podpisana **KATJA BAJEC** izjavljam, da sem avtorica diplomske naloge z naslovom:
**»MODELIRANJE RAZNIH POSTOPKOV V KATASTRIH NEPREMIČNIN S
POMOČJO UML DIAGRAMOV«.**

Izjavljam, da prenašam vse materialne avtorske pravice v zvezi z diplomsko nalogo na UL,
Fakulteto za gradbeništvo in geodezijo.

Ljubljana, 29.09.06

(podpis)

BIBLIOGRAFSKO – DOKUMENTACIJSKA STRAN IN IZVLEČEK

- UDK:** 004.43:528.44(043.2)
- Avtor:** Katja Bajec
- Mentor:** izr. prof. dr. Radoš Šumrada
- Somentor:** viš. pred. dr. Miran Ferlan
- Naslov:** Modeliranje raznih postopkov v katastrih nepremičnin s pomočjo UML diagramov
- Obseg in oprema:** 57 str., 19 pregl., 29 sl.
- Ključne besede:** UML, modeliranje, metodologija modeliranja, postopki v katastrih nepremičnin, urejanje meje, evidentiranje meje

Izvleček

V diplomski nalogi je predstavljena metodologija za izdelavo modelov raznih postopkov v katastrih nepremičnin s standardnim jezikom za modeliranje UML. Poudarek je na izdelavi UML - diagramov, s katerimi grafično prikažemo različne poglede na sistem. Izdelala sem jih s CASE - orodjem Enterprise Architect 5.0.

V teoretičnem delu naloge sem predstavila jezik UML – nastanek, razvoj, pomen, elemente, ki jih definira, in podrobno diagrame, uporabljene v praktičnem delu naloge.

V praktičnem delu naloge sem obravnavala postopke urejanja in evidentiranja meje, ki je dostikrat sestavni del bolj zapletenih postopkov, kot sta npr. parcelacija in komasacija. Namen izdelave modela je bil pregledno prikazati, kdo vse sodeluje v postopkih urejanja in evidentiranja meje, kakšne so vloge sodelujočih ter katere aktivnosti so potrebne za celotno in pravilno izvedbo postopkov. Proces izdelave modela v praktičnem delu naloge je bil osnova za razvoj metodologije za izdelavo modela postopka v katastrih nepremičnin.

BIBLIOGRAPHIC – DOCUMENTALISTIC INFORMATION

UDC: 004.43:528.44(043.2)
Author: Katja Bajec
Mentor: Assoc. Prof. Dr. Radoš Šumrada
Co-mentor: Sen. Lect. Dr. Miran Ferlan
Title: Cadastral procedures modeling with UML diagrams
Obseg in oprema: 57 p., 19 tab., 29 fig.
Ključne besede: UML, modeling, modeling methodology, land cadastral system procedures, definitive boundary approval, definitive boundary registration

Abstract

The present work focuses on the methodology that should be applied when developing cadastral procedures models using the Unified modeling language – UML. The main purpose of cadastral procedure model would be creating UML diagrams, which contain graphical elements arranged to illustrate a particular part or aspect of the system. The diagrams were drawn with CASE tool Enterprise Architect 5.0.

The theoretical part is actually a presentation of UML – its history, development, goals, basic UML elements and a detailed description of diagrams used later in the practical part.

In the practical part I worked on definitive boundary approval and registration procedures, which often are only a part of more complex procedures as for example a subdivision of real property. The purpose of the modeling the definitive boundary approval and registration procedures was to identify the actors that interact with the system, their roles and the activities that make up a complete and correctly executed procedure. When developing a methodology for creating the cadastre procedures models, I relied on the process I used in the practical part.

ZAHVALA

Mentorju, izr. prof. dr. Radošu Šumradi, in somentorju, viš. pred. dr. Miranu Ferlanu, se iskreno zahvaljujem za nasvete in strokovno pomoč pri nastajanju diplomske naloge.

Zahvala gre tudi staršem, predvsem mami, za podporo in vzpodbudo v času mojega študija.

KAZALO VSEBINE

1	UVOD	1
1.1	Predstavitev	1
1.2	Namen diplomske naloge	1
1.3	Metoda dela	2
1.4	Opis vsebine	2
2	POENOTENI MODELIRNI JEZIK UML	3
2.1	Splošno o UML	3
2.2	Nastanek in razvoj UML	4
2.3	Cilji in namen jezika UML	6
2.4	Elementi jezika UML	7
2.4.1	Predmeti	9
2.4.1.1	Strukture	9
2.4.1.2	Vedenja	9
2.4.1.3	Grupiranja	9
2.4.1.4	Oznake	10
2.4.2	Relacije	10
2.4.2.1	Odvisnost	10
2.4.2.2	Asociacija	10
2.4.2.3	Generalizacija (specializacija)	11
2.4.2.4	Realizacija	12
2.4.3	Diagrami	12
2.4.3.1	Statični diagrami	14
2.4.3.2	Dinamični diagrami	14
2.4.4	Razširitveni mehanizmi	15
2.4.4.1	Stereotip	15
2.4.4.2	Etiketna vrednost	15
2.4.4.3	Omejitev	16

2.5	Opis diagramov, uporabljenih v praktičnem delu naloge	17
2.5.1	Diagram primerov uporabe	17
2.5.1.1	Akter	17
2.5.1.2	Primer uporabe	19
2.5.1.3	Odnosi med primeri uporabe in akterji	20
2.5.1.4	Pregledni diagram primerov uporabe	22
2.5.1.5	Podroben opis primerov uporabe	22
2.5.2	Diagram aktivnosti	23
2.5.2.1	Osnovni elementi	24
3	PRIKAZ POSTOPKOV UREJANJA IN EVIDENTIRANJA MEJE (ZEN) Z UML DIAGRAMI	29
3.1	Akterji	29
3.1.1	Opredelevitev akterjev	29
3.1.2	Pregledni diagram primerov uporabe za postopek urejanja in evidentiranja meje	33
3.2	Glavni potek dogodkov za postopek urejanja in evidentiranja meje	34
3.3	Diagram primerov uporabe za postopek urejanja in evidentiranja meje	35
3.3.1	Primer uporabe »priprava na mejno obravnavo«	36
3.3.2	Primer uporabe »izvedba mejne obravnave«	38
3.3.3	Primer uporabe »izdelava elaborata ureditve meje«	42
3.3.4	Primer uporabe »postopek evidentiranja urejene meje«	43
3.3.5	Primer uporabe »ustna obravnavo«	47
4	ZAKLJUČEK	50
	LITERATURA IN VIRI	51

KAZALO PREGLEDNIC

- Preglednica 1: Vrste relacij med primeri uporabe
- Preglednica 2: Oblika podrobnega opisa primera uporabe
- Preglednica 3: Akter geodetsko podjetje
- Preglednica 4: Akter geodet
- Preglednica 5: Akter odgovorni geodet
- Preglednica 6: Akter stranka
- Preglednica 7: Akter naročnik
- Preglednica 8: Akter pasivna stranka
- Preglednica 9: Akter geodetska uprava
- Preglednica 10: Akter sodišče
- Preglednica 11: Glavni potek dogodkov za postopek urejanja in evidentiranja meje
- Preglednica 12: Podroben opis primera uporabe »priprava na mejno obravnavo«
- Preglednica 13: Podroben opis primera uporabe »izvedba mejne obravnave«
- Preglednica 14: Alternativni poteki dogodkov primera uporabe »izvedba mejne obravnave«
- Preglednica 15: Podroben opis primera uporabe »izdelava elaborata ureditve meje«
- Preglednica 16: Podroben opis primera uporabe »postopek evidentiranja urejene meje«
- Preglednica 17: Alternativni poteki dogodkov primera uporabe »postopek evidentiranja urejene meje«
- Preglednica 18: Podroben opis primera uporabe »ustna obravnavo«
- Preglednica 19: Alternativni potek dogodkov primera uporabe »ustna obravnavo«

KAZALO SLIK

- Slika 1: Gradniki UML
- Slika 2: UML diagrami
- Slika 3: Primera različnih notacij za stereotip
- Slika 4: Etiketne vrednosti stereotipa točka
- Slika 5: Omejitve v jeziku OCL za stereotip Točka v obliki komentarja
- Slika 6: Notacija za akterja
- Slika 7: Notacija za generalizacijo akterjev
- Slika 8: Notacija za primer uporabe
- Slika 9: Relacija med akterjem in primerom uporabe
- Slika 10: Primer preglednega diagrama primerov uporabe
- Slika 11: Notacija za akcijo
- Slika 12: Notacija za aktivnost
- Slika 13: Notacija za krmilni tok
- Slika 14: Notacija za začetni element
- Slika 15: Notacija za končni element aktivnosti
- Slika 16: Notacija za končni element toka
- Slika 17: Notacija za element odločitve
- Slika 18: Notacija za element združitve
- Slika 19: Notacija za element cepitve na vzporedne podaktivnosti
- Slika 20: Notacija za element združitve vzporednih podaktivnosti
- Slika 21: Notacija za povezovalni element
- Slika 22: Prikaz uporabe stez
- Slika 23: Pregled notacije elementov diagrama aktivnosti
- Slika 24: Pregledni diagram primerov uporabe za postopek urejanja in evidentiranja meje
- Slika 25: Diagram primerov uporabe za postopek urejanja in evidentiranja meje
- Slika 26: Diagram aktivnosti »priprava na mejno obravnavo«
- Slika 27: Diagram aktivnosti »izvedba mejne obravnave«
- Slika 28: Diagram aktivnosti »postopek evidentiranja urejene meje«
- Slika 29: Diagram aktivnosti »ustna obravnavo«

1 UVOD

1.1 Predstavitev

UML je standarden jezik za modeliranje, ki je nastal predvsem z namenom podpore načrtovanju objektno usmerjenih softverskih sistemov. Vendar pa se njegova uporabnost tu ne konča – z njim lahko opišemo model kateregakoli splošnega sistema. UML definira več diagramskih tehnik, s katerimi grafično prikažemo različne poglede na sistem. V diplomski nalogi sem z diagrami primerov uporabe in diagrami aktivnosti prikazala postopke pri urejanju in evidentiranju meje, kot jih predpisuje Zakon o evidentiranju nepremičnin (ZEN).

1.2 Namen diplomske naloge

Avtorji UML (Booch, Jacobson, Rumbaugh) so razvili poenoteni jezik za modeliranje, razvojni proces izdelave modela pa uporabnik izbere sam glede na lastne potrebe. Namen naloge je razviti primerno metodologijo za modeliranje raznih postopkov v katastrih nepremičnin s pomočjo UML diagramov.

V praktičnem delu naloge sem skušala z ustreznimi UML diagrami pregledno prikazati potek urejanja in evidentiranja meje. Diagrami naj bi dali jasen odgovor na vprašanja:

- Kdo vse sodeluje pri postopkih urejanja in evidentiranja meje?
- Kakšna je vloga lastnika parcele, ki meji ali se dotika meje, ki se ureja?
- Kakšna je vloga geodeta?
- Kakšna je vloga ostalih akterjev?

Rešitve, uporabljene za primer urejanja in evidentiranja meje, bodo podlaga za razvoj metodologije modeliranja poljubnega postopka v katastrih nepremičnin.

1.3 Metoda dela

Pri izdelavi diplomske naloge sem se največ opirala na naslednja dva vira:

- **The Unified Modeling Language Reference Manual Second Edition** avtorjev James Rumbaugh, Ivar Jacobson in Grady Booch, ki je dejansko zgoščen pregled UML 2.0 specifikacije.
- Podatke o postopkih urejanja in evidentiranja meje sem večinoma črpala iz **Zakona o evidentiranju nepremičnin (ZEN)**.

UML diagrame sem izdelala s CASE (Computer Aided Software Engineering) orodjem za načrtovanje in izvedbo softverskih sistemov - **Enterprise Architect 5.0**.

Postopki pri izdelavi diplomske naloge so bili naslednji:

- pregled literature, elektronskih virov in priročnikov s področja UML,
- predstavitev UML,
- pregled ZEN,
- izbira diagramov, ki bi bili primerni za prikaz postopkov pri urejanju in evidentiranju meje,
- predstavitev izbranih diagramov,
- izdelava diagramov ter
- posplošitev postopkov pri modeliranju izbranega primera na metodologijo za modeliranje poljubnega postopka v katastrih nepremičnin.

1.4 Opis vsebine

Diplomsko nalogo sestavljata teoretični in praktični del. V teoretičnem delu sem predstavila jezik UML, njegov nastanek in razvoj ter pomen, razložila pomembne pojme, ki se povezujejo z UML, predstavila diagrame, ki jih definira, in podrobneje opisala tiste, ki sem jih uporabila v praktičnem delu naloge. Praktični del predstavljajo diagrami primerov uporabe in diagrami aktivnosti, ki ponazarjajo postopke pri urejanju in evidentiranju meje. V praktičnem delu sem z diagrami predstavila samo postopke urejanja in evidentiranja meje

(naslov diplomske naloge je zastavljen širše – postopki v katastrih nepremičnin), uporabljena metodologija pa bi bila primerna tudi za opis ostalih postopkov v katastrih nepremičnin. Diplomski nalogi je priložena zgoščanka z elektronskim zapisom dokumenta v obliki pdf in eap datoteko, ki vsebuje vse izdelane diagrame. Eap datoteko odpremo z orodjem Enterprise Architect 5.0.

2 POENOTENI MODELIRNI JEZIK UML

2.1 Splošno o UML

UML (Unified Modeling Language) je objektno usmerjen formalen jezik za analizo, opredelitev, grafično ponazoritev, načrtovanje sestavin, izvedbo ter dokumentacijo softverskih, informacijskih, poslovnih in drugih sistemov (Šumrada, 2005).

Njegov temeljni cilj je zagotoviti uporabnikom standardiziran, konceptualno močan in razširljiv opisni mehanizem za prikaz vseh možnih vidikov obravnavanega sistema (Lahajnar, et al., 2001).

Modeliranje sistema z več zornih kotov služi različnim namenom:

- ugotovitev in natančna opredelitev zahtev, tako da jih lahko vsi, ki bodo sistem uporabljali oz. imeli nanj nek vpliv, razumejo in se z njimi strinjajo,
- pomoč pri načrtovanju sistema,
- pomoč pri izbiri najprimernejše izvedbene možnosti izmed vsemi predlaganimi,
- izdelava produktov, ki bodo uporabni tekom izvedbe (model softverskega sistema se npr. lahko uporabi za izdelavo deklaracij razredov, uporabniških vmesnikov, podatkovnih baz, scenarijev primerov uporabe,...),
- organizacija podatkov, poizvedovanje po njih, prečiščevanje, preiskovanje in urejanje podatkov o obsežnih sistemih,
- pregled različnih rešitev z ekonomskega vidika in
- upravljanje zapletenih sistemov.

S pomočjo UML opišemo statično strukturo in dinamično obnašanje sistema. Ta je modeliran kot nabor diskretnih objektov, ki med seboj sodelujejo z namenom zadovoljiti potrebe zunanjega uporabnika. Statična struktura definira vrste objektnih tipov (razredov), ki imajo pomen za sistem in njegovo izvedbo, pa tudi odnose med njimi. Dinamično obnašanje definira, kaj se z razredi dogaja skozi čas in komunikacijo med njimi tekom doseganja določenih ciljev.

UML definira tudi organizacijske gradnike (*package*), s katerimi programerske ekipe velike sisteme razdelijo v sestavne dele obvladljivih razsežnosti, nadzorujejo odvisnosti med temi sestavnimi deli in upravljajo razvoj enot modela v zapletenem razvojnem okolju (Rumbaugh, et al., 2004).

UML ni vezan na noben določen programski jezik za razvoj objektno usmerjenih sistemov. Tudi ni omejen samo na modeliranje softverskih sistemov, čeprav je bil zasnovan z namenom izboljšanja komunikacije in produktivnosti med razvijalci objektno usmerjenih sistemov, ampak z njim lahko opišemo katerikoli splošen sistem. Zato lahko rečemo, da je UML jezik za modeliranje oz. za opisovanje modelov. Za prikaz teh modelov pa uporablja več vrst diagramov, ki vsebujejo različne gradnike ter medsebojne povezave, odnose in omejitve (Malnarič, 2003, Chonoles, et al., 2003).

2.2 Nastanek in razvoj UML

Od sredine sedemdesetih do poznih osemdesetih so nastajali jeziki za objektno usmerjeno (ang. Object Oriented – v nadaljevanju OO) modeliranje. V tem času se je začelo eksperimentirati z alternativnimi načini analize in projektiranja zapletenih sistemov, izvedenih v OO programskih jezikih. V letih od 1989 do 1994 je število OO metod naraslo od deset do več kot 50, nobena izmed njih pa ni v celoti zadovoljila vseh potreb uporabnikov. Pojavila se je potreba po poenotenju in standardizaciji.

Oktobra 1994 sta James Rumbaugh in Grady Booch začela delati na poenoteni metodologiji, ki sta jo poimenovala Unified Method in ki združuje koncepte njunih lastnih metod – Booch in OMT. Oktobra 1995 je izšla specifikacija UM, verzija 0.8. Približno v istem času se jima je pridružil Ivar Jacobson, ki je enotnemu pristopu dodal principe iz svoje metode OOSE.

Avtorji so ugotovili, da ni potrebno razvijati celotne metodologije, ki poleg jezika (notacija in njena semantika) vključuje tudi proces razvoja softvera. Izbiro razvojnega procesa so prepustili posamezni organizaciji, ki ga naj prilagodi lastnim potrebam, sami pa so se osredotočili na poenoteni jezik, ki so ga poimenovali UML (Miličev, et al., 2001). Odziv javnosti na verzijo 0.8 in Jacobsonov doprinos sta bila upoštevana v naslednjih dveh verzijah (0.9 julija in 0.91 oktobra 1996). Julija 1997 je bila verzija 1.0 predlagana organizaciji OMG (Object Management Group) za standardizacijo. Dodatne izboljšave so bile vsebovane v verziji 1.1 - novembra 1997 je bil UML v okviru OMG standardiziran in je postal standardni jezik za modeliranje sistemov (Vir: Introduction to Visual Modeling with Rational Rose 2002 and UML).

UML je torej nastal z združitvijo več metodologij za OO analizo in načrtovanje, med katerimi so bile najpomembnejše (Slokar, 2005):

- Rumbaugh-eva OMT (Object Modeling Technique),
- Jacobson-ova OOSE (OO Software Ingeneering),
- Booch – eva OO Analysis and Design.

UML se je v času od izdaje UML 1.1 do danes bistveno razvil. Sledilo je več popravkov v verzijah 1.3, 1.4 in 1.5. Aktualni OMG standard je UML 2.0, ki je bil v celoti sprejet konec leta 2005. Trenutno je v pripravi verzija 2.1, ki naj bi izšla v letu 2006 (Vir: Wikipedia).

Verzija 2.0 je zasnovana tako, da uporabniki prejšnjih verzij z njo ne bi smeli imeti težav. Mnogi originalni diagrami (in z njimi povezana notacija) so ohranjeni, nekateri med njimi preoblikovani (diagram komponent) in razširjeni. Poleg tega pa so v njej vsebovani novi tipi diagramov (časovni diagram, pregledni diagram interakcije, diagram strukture sestavin), kar je odziv na trenutne zahteve modeliranja softverskih in drugih sistemov (Hamilton, et al., 2006).

UML 2.0 je doprinesel še naslednje opazne spremembe (Rumbaugh, et al., 2004):

- Da bi diagram zaporedja postal bolj objektno usmerjen, temeljijo njegovi gradniki in notacija v glavnem na ITU standardu za diagram zaporedja sporočil.

- Razdružitev konceptov modeliranja aktivnosti in modeliranja stanj in uporaba notacije, ki je v uporabi pri modeliranju poslovnih sistemov.
- Poenotenje modeliranja aktivnosti z modeliranjem akcij z namenom pridobitve bolj popolnega postopkovnega modela.

Medtem ko večina objektnih metodologij sestoji iz modelirnega jezika in procesa razvoja, UML podaja samo modelirni jezik (notacijo), ki jo metodologije uporabljajo za izražanje modelov. Zato UML imenujemo jezik za objektno modeliranje in ne metodologija. Za komunikacijo o rezultatih razvoja med razvijalci namreč potrebujemo enotno notacijo, ki prikaže rezultate dela in nas ne obremenjuje s postopki, ki so privedli do teh rezultatov. Jezik za objektno modeliranje UML je nastal iz potrebe po poenotenosti in združitvi objektnih razvojnih metod, ki obstajajo v svetu razvoja programskih sistemov. Standardizacija jezika omogoča odločitev za tehnologijo in ne za proizvajalca, kar od razvijalcev podpornih orodij zahteva določeno mero sodelovanja, če želijo doseči medopravilnost orodij. Osnova za to je standarden jezik. Nenazadnje mora jezik biti uporabljen v industrijskem okolju, da bo doživel svoje razširjanje in možnost izkoriščanja. Delo na tem področju danes teče zelo aktivno in pričakujemo lahko, da bodo vsa podporna orodja v prihodnosti podpirala UML (Slokar, 2005).

2.3 Cilji in namen jezika UML

Osnovne cilje jezika UML lahko strnemo v naslednjih točkah (Bennett, et al., 2005.):

- Uporabnikom omogočiti razvoj in izmenjavo modelov v jeziku, ki je neodvisen od programskega jezika in razvojnega procesa.
- Združiti najboljše ideje in izkušnje različnih avtorjev, ki delujejo na področju objektno usmerjenega modeliranja. Zaradi združevanja več različnih pogledov na objektno usmerjeno modeliranje bi UML lahko posledično vseboval odvečno notacijo. V izogib temu je jedro UML zasnovano kar se da enostavno, dodatnim, novim zahtevam pa se ugotovi s pomočjo profilov in ostalih mehanizmov razširjanja jezika.
- Formalna osnova za razumevanje modelirnega jezika naj pomaga uporabniku pri spoznavanju in uporabi jezika.

2.4 Elementi jezika UML

UML definira notacijo, metamodel in razširitvene mehanizme jezika. Notacija je grafična predstavitev jezika (sintaksa), ki sama po sebi ne daje jasnega in nedvoumnega pomena modela.

Avtorji so jezik za objektno modeliranje UML zgradili štiriplastno (Bennett, et al., 2001):

- **meta-metamodel**, ki definira jezik za specifikacijo metamodela in predstavlja infrastrukturo za arhitekturo metamodela,
- **metamodel**, ki je primerek meta-metamodela in definira jezik za specifikacijo modelov,
- **model**, ki je primerek metamodela in definira jezik za opis informacijske domene,
- **uporabniški objekti**, ki so primerki modela in definirajo informacijsko domeno.

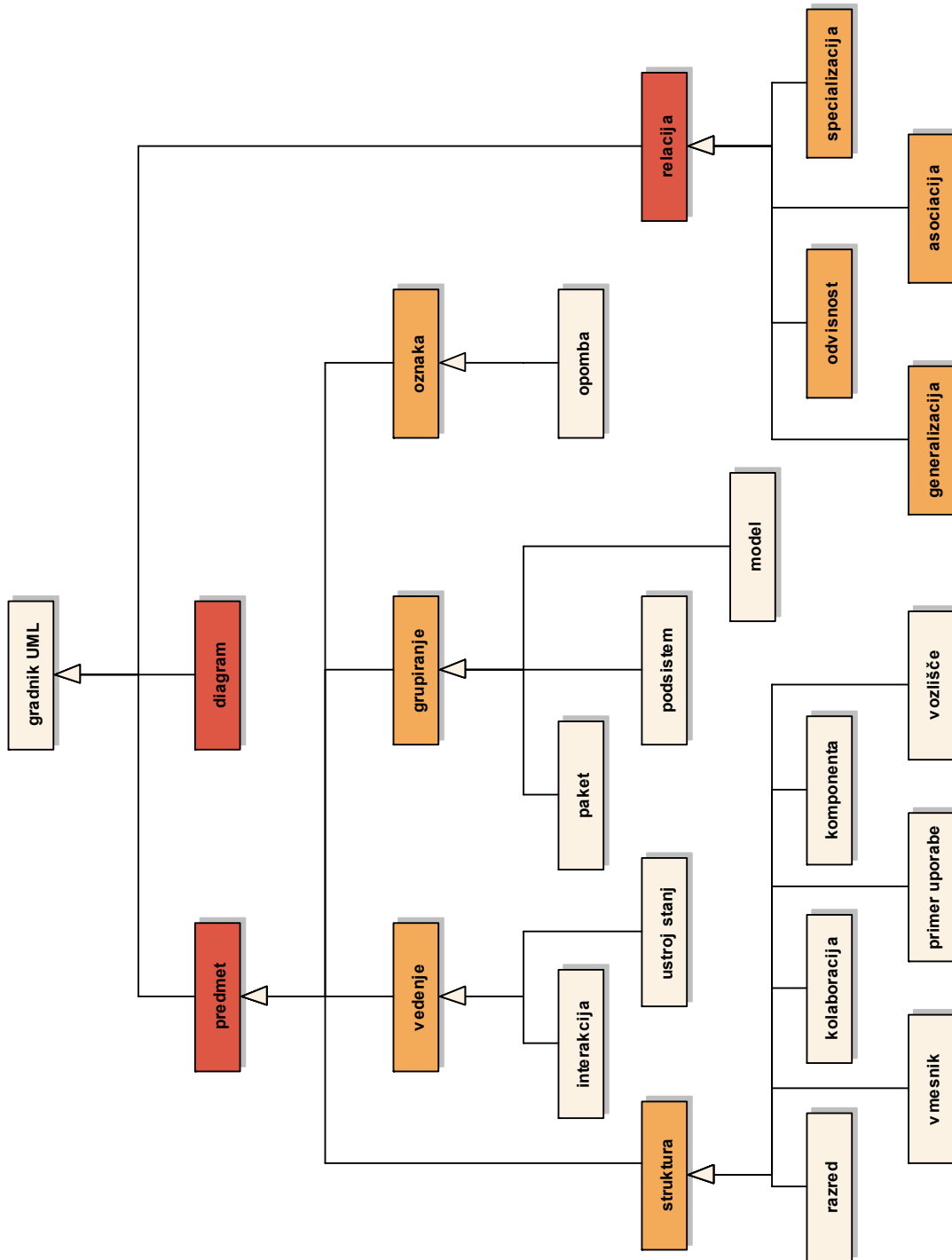
Takšna zasnova jezika omogoča uporabo jezika na vseh področjih, saj vsebuje gradnike, ki omogočajo modeliranje različnih sistemov (Slokar, 2005).

Metamodel UML sestavljajo tri skupine elementov (Šumrada, 2005):

- osnovni gradniki,
- pravila za njihovo uporabo in
- povezovalna načela in razširitveni mehanizmi.

Osnovni gradniki UML so: **predmeti** (*things*), **relacije** (*relationships*) in **diagrami** (*diagrams*) (Šumrada, 2005). Slika 1 prikazuje njihovo podrobnejšo razdelitev na skupine in podskupine.

Gradniki UML, ki v okviru te diplomske naloge niso pomembni, so v nadaljevanju samo naštet, za ostale je podana definicija oz. so podrobneje opisani.



Slika 1: Gradniki UML (Šumrada, 2005)

2.4.1 Predmeti

Predmeti so pomembni modelni objekti, UML jih razvršča v 4 skupine: strukture, vedenja, grupiranja in oznake.

2.4.1.1 *Strukture*

Strukture so abstrakcije, ki tvorijo strukturalni, v glavnem statični del modela. To so (Šumrada, 2005):

- Razred (*class*) je opis enega ali več objektov, ki imajo enak pomen, enak niz atributov, enake relacije in enako vedenje (Objekt je pojav razreda, ki ima identiteto, stanje in obnašanje). Razred ima podatkovno zgradbo (atribute) in procesno vedenje (metode), ki določajo dovoljene operacije nad interno podatkovno sestavo.
- Vmesnik (*interface*)
- Kolaboracija (*collaboration*)
- Primer uporabe (*use case*) je opis niza zaporednih korakov, ki jih sistem izvrši z namenom, da bi proizvedel navzven vidno obnašanje, bistveno za nekega akterja (Rumbaugh, et al., 2004).
- Komponenta (*component*)
- Vozlišče (*node*)

2.4.1.2 *Vedenja*

Vedenja opisujejo obnašanje razredov, t.j. opredeljujejo dinamični del modela (Miličev, 2001):

- Interakcija (*interaction*)
- Ustroj stanj (*state machine*)

2.4.1.3 *Grupiranja*

Grupiranja so organizacijski deli jezika (Rumbaugh, et al., 2004, Miličev, 2001):

- Paket (*package*) je predmet, v katerega se združujejo vsi ostali elementi jezika.

- Podsistem je obsežna enota razstavitve sistema (sistem je zbirka povezanih enot, ki so organizirani, da bi služili nekemu namenu). V UML je modeliran kot stereotip komponente (stereotip je nova oblika obstoječega modelnega elementa, ki ima sicer enako obliko (form), npr. enake attribute in relacije, vendar drug namen).
- Model je paket, ki obsega popoln opis določenega pogleda na sistem.

2.4.1.4 Oznake

Za izpis komentarja o modelu v neformalnem jeziku, dela kode v določenem programskem jeziku, omejitve (*constraint*) ipd. uporabljamo opombe (*note*). Na diagramih jo prikažemo kot pravokotnik z zavihkom v gornjem desnem kotu, ki je s črtkano črto povezan z elementom, ki ga opisuje. Opomba na diagramu brez črtkane črte običajno na nek način opisuje diagram (npr. podrobnosti o avtorju, datum izdelave diagrama, vir podatkov o modelu,...) (Bennett, et al., 2001).

2.4.2 Relacije

Relacija je katerakoli pomenska povezava ali odnos med razredi, ki jo je treba vključiti v podatkovni model. Ločimo štiri vrste možnih relacij ali odnosov med razredi (Šumrada 2005): odvisnost, asociacija, generalizacija ali specializacija in realizacija.

2.4.2.1 Odvisnost

Odvisnost (*dependency*) je katerakoli semantična relacija med razredi, ki označuje, da lahko sprememba enega (neodvisnega) predmeta vpliva na drugi (odvisni) predmet (Slokar, 2005).

2.4.2.2 Asociacija

Asociacija (*association*) je strukturna relacija, ki vsebuje podatke o odnosih med objekti (pojavi razredov) v sistemu. Vsaka asociacija ima urejen seznam dveh ali več krajišč asociacije (dve krajišči se razlikujeta in jih ne smemo zamenjati). Vsako krajišče asociacije definira udejstvovanje razreda v danem položaju v asociaciji (vlogo). Isti razred se lahko

pojavi na več kot eni poziciji. Večino podatkov o asociaciji pripravimo na eno ali obe krajišči: ime (**vlogo**), obveznost (poljubna ali obvezujoča) in nenazadnje **števnost**, ki pove, koliko objektov razreda lahko sodeluje v asociaciji. Najpogosteje bomo definirali števnost na naslednje načine:

- 1 (točno eden)
- 0..1 (eden ali nobeden)
- (poljubno mnogo)

Splošneje števnost definiramo s spodnjo in zgornjo mejo, npr. 2..5. Spodnja meja je lahko katerokoli pozitivno celo število ali 0, zgornja pa katerokoli pozitivno celo število ali * (za neomejen obseg).

Števnost je bolj ko pri **n – kratnih** asociacijah (ki povezujejo več razredov in so zato precej zapletene) bolj uporabna v primeru **binarnih**, ki povezujejo dva razreda. Slednje so tudi večinoma v uporabi.

Posebna oblika asociacije je **povratna** asociacija. Asociacija razreda ne kaže na drugi razred, temveč na samo sebe.

Asociacija ima lahko ime, ki opredeli naravo odnosa med razredoma, smer branja pa določimo s puščico. V osnovi je asociacija dvosmerna – oba razreda vesta eden za drugega. Z usmerjeno povezavo med razredoma pa določimo, da asociacija poteka samo v izbrani smeri. Posebni obliki asociacije sta tudi **agregacija** (*aggregation*) in **kompozicija** (*composition*). Prva označuje nesimetrično povezavo tipa celota/del (*whole/part*), druga pa je strožja oblika agregacije. Označuje odnos, kjer podrejeni razred ne more obstajati brez nadrejenega razreda (Scott, 2001, Fowler, 2004).

2.4.2.3 Generalizacija (specializacija)

Generalizacija ali specializacija (*generalization, specialization*) je relacija, pri kateri lahko objekte generaliziranega elementa (prednika) zamenjajo objekti specializiranega elementa (potomca). Potomec deduje lastnosti in operacije od enega ali več prednikov (Miličev, 2001).

Generalizacija omogoča specializacijo (nad)razreda v nove (pod)razrede. Tako lahko posebne primere ali razširitve obravnavamo kot ločene razrede, pri tem pa ohranimo ključne elemente nadrazreda.

Abstrakten razred je razred, kateremu ne more direktno pripadati noben objekt, ampak samo opisuje skupne attribute in obnašanje drugih razredov. Običajno ima abstraktne operacije – njihova izvedbena metoda ni definirana na mestu njihove specifikacije. Za izvedbo abstraktne operacije poskrbi podrazred, razen če tudi sam ni abstrakten razred.

Podrazred lahko priredi operacije nadrazreda ali pa ima glede na nadrazred dodane nove operacije, attribute in asociacije (Eriksson, et al., 2004).

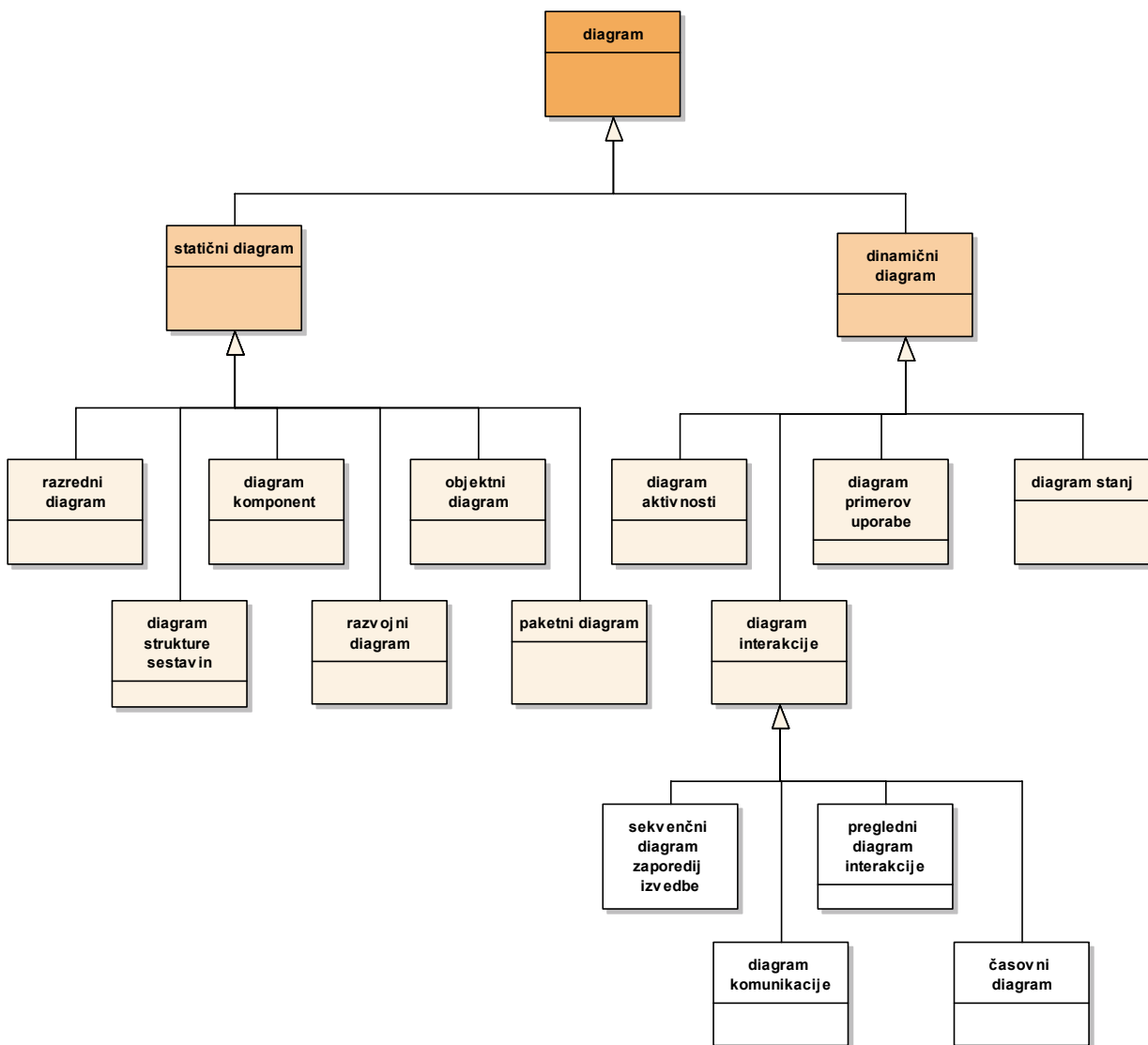
2.4.2.4 Realizacija

Realizacija je semantična povezava med razredoma, kjer prvi opredeljuje funkcijo, ki jo drugi izvede. Srečamo jo večinoma med vmesniki in razredi ter komponentami, ki jih uresničujejo, in med primeri uporabe ter kolaboracijami, ki jih uresničujejo (Slokar, 2005).

2.4.3 Diagrami

Z diagrami grafično in poenostavljeno predstavimo sistem, objekte, dogodke in stanja v različnih pogledih. UML definira 13 vrst diagramov, vendar pa omogoča razvitje tudi novih s poljubnim kombiniranjem predmetov in relacij med njimi (Rumbaugh, et al., 2004).

Glede na vsebino so razdeljeni v dve večji skupini - statični diagrami služijo modeliranju podatkovne sestave sistema, dinamični pa modeliranju procesnih lastnosti oz. funkcionalnosti sistema (Vir: Introduction to OMG's Unified Modeling Language). Na sliki 2 so prikazane vrste diagramov in njihova uvrstitve v skupino statičnih oz. dinamičnih diagramov.



Slika 2: UML diagrami (Vir: Introduction to OMG's Unified Modeling Language)

2.4.3.1 *Statični diagrami*

- **Razredni diagram** (*class diagram*), ki je nekoliko spremenjena diagramska tehnika iz Rumbaugh-ove OMT metodologije, predstavlja osnovo objektno usmerjenega programiranja pri izgradnji softverskih sistemov. Uporabljamo ga za statičen pogled na problemsko domeno, kar pomeni, da nas tu ne zanima časovno odvisno obnašanje sistema, ampak se osredotočimo na sestavo razredov in odnose med njimi (asociacija, generalizacija in različne vrste odvisnosti) (Slokar, 2005).
- **Objektni diagram** (*object diagram*)
- **Diagram komponent** (*component diagram*)
- **Razvojni diagram** (*deployment diagram*)
- **Paketni diagram** (*package diagram*)
- **Diagram strukture sestavin** (*composite structure diagram*)

2.4.3.2 *Dinamični diagrami*

- **Diagram primerov uporabe** (*use case diagram*) izvira neposredno iz Jacobsonove OOSE metodologije in podaja pogled na uporabniške zahteve.
- **Diagram stanj** (*state machine diagram*)
- **Diagram aktivnosti** (*activity diagram*) se osredotoča na izvajanje posameznih dejavnosti znotraj izvedbe določene operacije (metode). V diplomski nalogi sem z njim prikazala aktivnosti, ki sestavljajo primer uporabe.
- **Diagram interakcije** (UML definira 4 vrste diagramov interakcije):
 - **Sekvenčni diagram zaporedij izvedbe** (*sequence diagram*)
 - **Diagram komunikacije ali sodelovanja** (*communication diagram*)
 - **Časovni diagram** (*timing diagram*)
 - **Pregledni diagram interakcije** (*interaction overview diagram*)

V nadaljevanju sta podrobneje opisana diagram primerov uporabe in diagram aktivnosti, ki sem ju uporabila v praktičnem delu diplomskega dela.

Zavedati se je treba, da lahko veliko število diagramskih tehnik zmede uporabnika. Namen jezika UML je uporabiti le tiste diagramske tehnike, ki so pomembne za določeno problemsko področje, nikakor pa ni nujno uporabiti vseh tehnik na vseh področjih. Prav tako sami diagrami ne smejo biti preobsežni oz. morajo vsebovati le elemente, ki so nujno potrebni za preglednost in razumevanje obravnavanega problema (Slokar, 2005).

2.4.4 Razširitveni mehanizmi

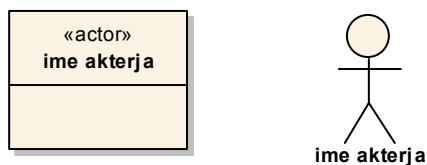
UML vsebuje več razširitvenih mehanizmov. Ti omogočajo ponazarjanje dodatnih vidikov obravnavanega sistema, ki sicer niso zajeti znotraj UML, ne da bi spreminjali metamodel. Običajno se uporabijo za posebna področja uporabe in posebna programerska okolja. Razširitve so organizirane v obliki profilov. Profil je niz medsebojno skladnih razširitev, ki so uporabne na določenem področju oz. za določen namen (Rumbaugh, et al., 2004). Kot primer naj navedem tehnično specifikacijo ISO 19103, ki definira profile za ponazoritev sestave prostorskih podatkov (Šumrada, 2005).

Razširitveni mehanizmi so **stereotipi** (*stereotypes*), **etiketne vrednosti** (*tagged values*) in **omejitve** (*constraints*) (Rumbaugh, et al., 2004).

2.4.4.1 Stereotip

Stereotip je razširitev metamodela, uporabljena z namenom modeliranja posebne vrste modelnega elementa v sistemu. UML sam uporablja stereotipe, npr. <<include>> in <<extend>> za razlikovanje odvisnosti pri diagramih primerov uporabe (Bennet, et al., 2005). Stereotip temelji na obstoječem modelnem elementu. Podatkovna vsebina in oblika stereotipa je enaka kot pri osnovnem modelnem elementu, se pa razlikujeta v pomenu in uporabi. Stereotipiziran element lahko vsebuje seznam omejitev glede uporabe in definicijo etiketnih vrednosti za opis dodatnih lastnosti, ki jih osnovni element ne podpira (Rumbaugh, et al., 2004).

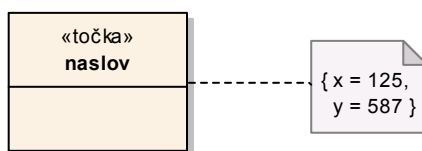
Na sliki 3 je prikazana notacija za stereotip - ime stereotipa v narekovajih znotraj ali zraven simbola osnovnega modelnega elementa, ali pa posebna ikona, ki nadomesti simbol osnovnega elementa.



Slika 3: Primeri različnih notacij za stereotip

2.4.4.2 Etiketna vrednost

Etiketna vrednost je imenovan sklop podatkov, ki se nanaša na modelni element. Z njo prikažemo posebne lastnosti, v katerih se stereotipizirani modelni element razlikuje od osnovnega modelnega elementa. Definirana je znotraj profila. Sestavljata jo etiketa (ime posebne lastnosti stereotipiziranega modelnega elementa) in odgovarjajoča vrednost. Ko uporabimo stereotip pri modelnem elementu, ta privzame etikete, definirane za stereotip. Za vsako etiketo lahko določimo njeno vrednost. Etiketne vrednosti so zlasti uporabne za podajanje podatkov o upravljanju projekta in parametrov za zagon kode. Na diagramih ji prikažemo v zavutih oklepajih kot komentar, pripet na stereotipiziran modelni element (slika 4), če pa na diagramu niso prikazane, so shranjene v obliki pojavnih seznamov ali obrazcev (Bennet, et al., 2005, Rumbaugh, et al., 2004).

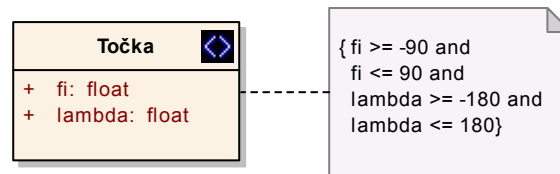


Slika 4: Etiketne vrednosti stereotipa točka

2.4.4.3 Omejitev

Omejitve opredeljujejo semantične pogoje in prepovedi v povezavi z modelnimi elementi, ki so lahko izraženi v formalnem ali neformalnem jeziku. UML vsebuje definicijo standardnega jezika (OCL – Object Constraint Language), ki je primeren (ne pa obvezen) za izražanje omejitev znotraj UML.

Omejitve so definirane v profilu. Na diagramih jih prikažemo v zavutih oklepajih, bodisi znotraj komentarja (slika 5) ali pa pripete na relacijo.



Slika 5: Omejitve v jeziku OCL za stereotip Točka v obliki komentarja

Potrebno se je zavedati, da razširitve predstavljajo odklon od standardne oblike UML, kar lahko otežuje medopravilnost. Zato moramo, preden se odločimo za uporabo razširitev, temeljito preveriti, ali lahko obstoječi mehanizmi zadostijo potrebam, ki se pojavijo pri modeliranju sistema, ali ne (Rumbaugh, et al., 2004).

2.5 Opis diagramov, uporabljenih v praktičnem delu naloge

2.5.1 Diagram primerov uporabe

Obnašanje obravnavanega sistema, t.j. funkcionalnost, ki jo mora sistem nuditi, dokumentiramo z modelom primerov uporabe. Ta vsebuje predvidene funkcije sistema (primeri uporabe), njegovo okolje (akterje) in odnose med primeri uporabe in akterji (diagrami primerov uporabe). Najvažnejša vloga modela primerov uporabe je komunikacija: omogoča izmenjavo mnenj o funkcionalnosti in obnašanju sistema med strankami oz. končnimi uporabniki in tistimi, ki ga načrtujejo. Razvoj modela primerov uporabe se začne z identifikacijo akterjev in osnovnih primerov uporabe sistema, v nadaljevanju obstoječim primerom uporabe dodamo bolj podrobne podatke in model po potrebi dopolnimo z dodatnimi primeri uporabe (Vir: UML Basics – The use case diagram).

2.5.1.1 Akter

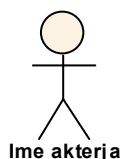
Akter (*actor*) je v splošnem lahko posamezni uporabnik sistema, naprava, podatkovna baza ali drug sistem, ki z opisovanim sistemom komunicira, izmenjuje podatke in iz njega pridobiva rezultate (Slokar, 2005).

Akter predstavlja možno medsebojno vplivanje zunanjih uporabnikov in sistema. Posamezni fizični uporabnik lahko deluje v vlogi različnih akterjev sistema.. Različni uporabniki lahko igrajo vlogo istega akterja in tako predstavljajo več pojavov iste definicije akterja. Na primer: ena oseba je lahko v trgovini blagajnik, v trenutku, ko kupuje artikle v tej trgovini, pa tudi kupec (Rumbaugh, et al., 2004).

V fazi določanja akterjev sistema nam pomaga, če najdemo odgovore na naslednja vprašanja (Vir: UML Basics – The use case diagram):

- Koga zanima določena zahteva, ki jo mora zadovoljiti sistem?
- Kdo v organizaciji bo uporabljal sistem?
- Kdo bo nekaj pridobil z uporabo sistema?
- Kdo bo sistem oskrboval s podatki, kdo uporabljal in kdo jih bo odstranjeval?
- Kdo bo podpiral in vzdrževal sistem?
- Ali sistem uporablja zunanje vire?
- Ali ena oseba igra več različnih vlog?
- Ali več ljudi igra isto vlogo?

Notacija je prikazana na sliki 6: akterja prikažemo z likom možica, pod lik napišemo ime vloge akterja (Rumbaugh, et al., 2004).

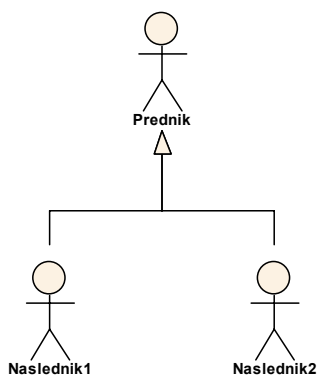


Slika 6: Notacija za akterja

Generalizacija: Situacijo, kjer ima več akterjev v okviru svoje lastne vloge neko skupno (splošnejšo) vlogo, opišemo z generalizacijo. V tem primeru se obnašanje, ki se nanaša na splošno vlogo, prikaže kot vloga splošnejšega akterja – prednika. Specifični akterji – nasledniki podedujejo lastnosti in operacije svojega prednika in jih tudi vsak na svoj način razširijo - dodajo lastne ali (in) preoblikujejo podedovane lastnosti in operacije. Pojav akterja

- naslednika se lahko vedno uporabi v primerih, kjer se pričakuje pojav akterja – prednika (Eriksson, et al., 2004).

Grafično se generalizacija prikaže s puščico z votlo konico, usmerjeno proti predniku (slika 7).

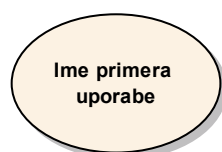


Slika 7: Notacija za generalizacijo akterjev

2.5.1.2 Primer uporabe

Primer uporabe (*use case*) je povezana enota funkcionalnosti, ki jo zagotavlja sistem ali podsistem in je izražena z zaporedji sporočil, izmenjanimi med sistemom in enim ali več uporabniki. Zaključni rezultat, ki ima nek pomen za določenega akterja. Detajlni zaporedni opis primera uporabe vsebuje celotno obnašanje: glavni potek dogodkov, različne variacije normalnega obnašanja in vse izjemne situacije, do katerih lahko pride, vključno s pričakovanimi odzivi akterja na obnašanje sistema. Z uporabnikovega gledišča so izjemne situacije nenormalne, z gledišča sistema pa so to dodatne variacije, ki jih je potrebno opisati in tudi znati obvladati. V modelu je izvršitev vsakega primera uporabe neodvisna od drugih, vendar pa lahko pri izvedbi primerov uporabe pride do implicitnih odvisnosti med njimi zaradi skupnih objektov (Rumbaugh, et al., 2004).

Notacija je prikazana na sliki 8: primer uporabe narišemo kot elipso, znotraj katere se nahaja ime primera uporabe.



Slika 8: Notacija za primer uporabe

Pri določanju primerov uporabe sistema si lahko pomagamo z naslednjimi vprašanji (Vir: UML Basics – The use case diagram):

- Kakšne so naloge vsakega akterja?
- Ali bo kateri od akterjev vnašal, shranjeval, brisal ali pregledoval informacije v sistemu?
- Kateri primer uporabe bo vnašal, shranjeval, brisal ali pregledoval te informacije?
- Ali bo moral kateri od akterjev obvestiti sistem o nenadnih zunanjih spremembah?
- Ali mora biti katerikoli od akterjev obveščen o določenih dogodkih v sistemu?
- Kateri primer uporabe bo podpiral in vzdrževal sistem?
- Ali primeri uporabe podpirajo vse funkcionalne zahteve sistema?

2.5.1.3 Odnosi med primeri uporabe in akterji

Primer uporabe lahko sodeluje v naslednjih vrstah relacij (Rumbaugh, et al., 2004):

Preglednica 1: Vrste relacij med primeri uporabe

Relacija	Pomen	Notacija
asociacija	Komunikacijska povezava med akterjem in primerom uporabe, v kateri sodeluje.	_____
razširja (<i>extend</i>)	Primer uporabe na začetku puščice (lahko) razširi obnašanje primera uporabe na koncu puščice.	«extend» ----->
vsebuje (<i>include</i>)	Primer uporabe na začetku puščice vsebuje obnašanje primera uporabe na koncu puščice.	«include» ----->
generalizacija primera uporabe	Relacija med splošnim in bolj specifičnim primerom uporabe, ki podeduje značilnosti splošnega primera uporabe in mu doda nove.	----->

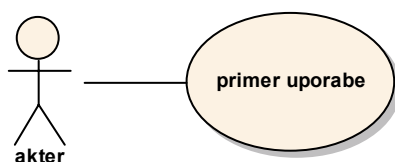
Opis obsežnega primera uporabe lahko razstavimo na druge, bolj enostavne primere uporabe. (podobno kot lahko opis razreda definiramo na podlagi opisa nadrazreda). Primer uporabe lahko vključuje vedenje drugih primerov uporabe kot sestavne dele lastnega vedenja – to je relacija vsebovanja (*include*). Vsebovani primer uporabe ni specializacija osnovnega primera uporabe in osnovnega primera uporabe torej ne more zamenjati.

Primer uporabe lahko definiramo tudi kot razširitev osnovnega primera uporabe – temu rečemo relacija razširitve (*extend*). Obstaja lahko več razširitev osnovnega primera uporabe, ki se lahko tudi hkrati izvajajo (Rumbaugh, et al., 2004).

Črtkani puščici, ki označuje relacijo »extend«, lahko pripnemo oznako, v kateri navedemo potrebne pogoje (*condition*) za »extend« primer uporabe in mesto v osnovnem primeru uporabe, kjer se začne izvajati »extend« primer uporabe (*extension point*) (Chonoles, 2003).

Primer uporabe je lahko specializiran z enim ali več primeri uporabe – generalizacija primera uporabe. Vsak specifični primer uporabe lahko uporabimo v situacijah, v katerih se pričakuje nadrejeni primer uporabe. Puščica povezave, ki ponazarja generalizacijo, je usmerjena k nadrejenemu primeru uporabe (Rumbaugh, et al., 2004).

Vsak primer uporabe ima primarnega akterja, ki od sistema zahteva, naj nudi neko uslugo. Običajno je pobudnik (iniciator) primera uporabe. Primer uporabe skuša doseči cilj, ki ga ima primarni akter. Sekundarni akterji so ostali akterji, s katerimi sistem komunicira tekom izvajanja primera uporabe.



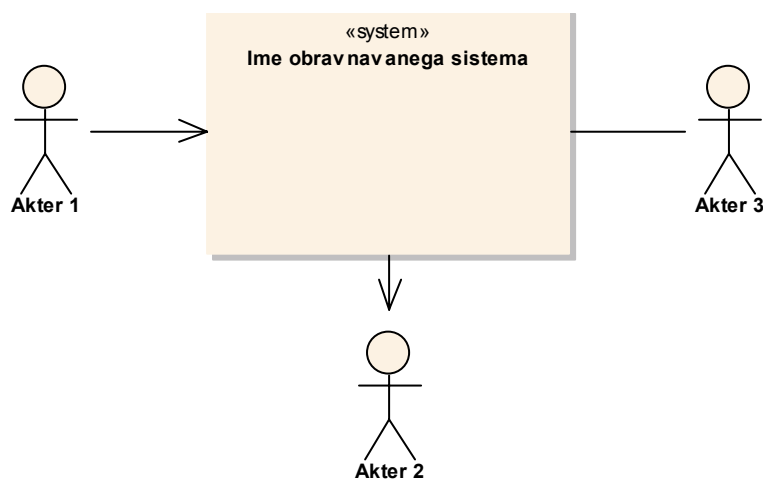
Slika 9: Relacija med akterjem in primerom uporabe

Vsebino primera uporabe lahko opišemo na več načinov: z neformalnimi tekstovnimi opisi, diagrami kolaboracije, diagrami zaporedij izvedbe ali diagrami aktivnosti (Rumbaugh, et al., 2004).

2.5.1.4 Pregledni diagram primerov uporabe

Pregledni diagram primerov uporabe povzame medsebojno vplivanje akterjev in sistema. Pogosto ga imenujemo kontekstualni diagram (njegov namen je splošen prikaz domene obravnavanega sistema). Prikazuje obravnavani sistem in vse njegove akterje, primeri uporabe ostanejo skriti. Ta diagram narišemo takoj, ko smo določili akterje in preden se lotimo določanja primerov uporabe.

Če je akter vedno iniciator primerov uporabe, v katerih sodeluje, ga s sistemom povezuje puščica, usmerjena proti sistemu. Če akter ni nikoli iniciator, naj bo puščica usmerjena proti njemu. Kadar nismo prepričani, puščico izpustimo (slika 10) (Chonoles, 2003).



Slika 10: Primer preglednega diagrama primerov uporabe

2.5.1.5 Podroben opis primerov uporabe

Diagram s primeri uporabe in akterji shematično prikazuje zahteve, ki jih mora sistem zadovoljiti, vendar pa je za dejansko razumevanje delovanja sistema ta nivo preveč splošen. Pomembna dejstva, kot so npr., kdo je najpomembnejši akter ali kateri koraki so vsebovani v primeru uporabe, najlažje navedemo v obliki tekstovnega opisa, ki naj bi se izdelal za vsak primer uporabe. UML ne podaja strogih navodil, kaj točno mora vsebovati (Hamilton, et al., 2006).

V preglednici 2 je prikazano, katere podrobnosti sem v praktičnem delu naloge vključila v podrobne opise primerov uporabe.

Preglednica 2: Oblika podrobnega opisa primera uporabe (Vir: Hamilton, et al., 2006)

Ime primera uporabe	Ime
Cilj	Mesto primera uporabe znotraj sistema in zakaj je primer uporabe pomemben.
Predpogoji	Kaj se mora zgoditi, preden se začne primer uporabe izvrševati.
Uspešni končni pogoji	Kakšni naj bi bili pogoji sistema, če se primer uporabe uspešno izvrši.
Neuspešni končni pogoji	Kakšni naj bi bili pogoji sistema, če se primer uporabe ne izvrši uspešno.
Primarni akterji	Glavni akterji, ki sodelujejo pri primeru uporabe. Večinoma so to akterji, ki sprožijo izvršitev primera uporabe ali pa ob izvršitvi direktno pridobijo podatke.
Sekundarni akterji	Akterji, ki sodelujejo pri primeru uporabe, a nimajo glavne vloge pri izvršitvi primera uporabe.
Glavni potek dogodkov	Pomembni koraki, ko se primer uporabe neovirano izvrši po najkrajši možni poti.
»Extend« primeri uporabe	Primeri uporabe, ki lahko pri določenih pogojih nadgradijo osnovni primer uporabe.
Alternativni poteki dogodkov	Opis katerihkoli alternativnih korakov glede na korake, opisane v glavnem poteku dogodkov (zaradi boljše preglednosti sem alternativne poteke opisala v ločeni tabeli).

Pri glavnem in alternativnih potekih dogodkov sem akterje izpisala poudarjeno svetlo modro, ključne glagole, ki predstavljajo aktivnosti akterjev, pa poudarjeno oranžno, kar naredi podrobne opise primerov uporabe, ki so osnova za izdelavo diagramov aktivnosti, bolj pregledne.

2.5.2 Diagram aktivnosti

Za razliko od razrednega diagrama, ki prikazuje, kdo (kateri razred ali razredi) je povezan (asociacija in generalizacija) s kom (ostali razredi), in kaj lahko vsak naredi (operacije posameznih razredov), se diagram aktivnosti osredotoči na tok obnašanja (ne zanima ga toliko, kdo ga izvaja) (Chonoles, 2003).

Uporabljamo ga lahko za različne namene (Eriksson, et al., 2004):

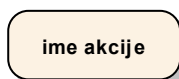
- prikaz zaporedja akcij tekom izvrševanja operacije,
- podroben vpogled v primer uporabe (za ta namen sem ga uporabila pri praktičnem delu diplomske naloge),
- modeliranje poslovnih procesov.

2.5.2.1 Osnovni elementi

Diagram aktivnosti vsebuje naslednje osnovne elemente (Rumbaugh, et al., 2004):

Akcija (*action*) je osnovni gradnik obnašanja. Določimo lahko predpogoje in končne pogoje akcije, ki morajo biti izpolnjeni, preden (oz. potem, ko) se akcija izvede. V smislu programiranja je akcija atomaren in kratek izračun, običajno prireditev vrednosti spremenljivke ali preprost aritmetičen izraz. Lahko pa je tudi pošiljanje signala drugemu objektu, klic operacije, določitev izhodne vrednosti, ustvarjanje ali uničenje objekta (Rumbaugh, et al., 1999). V smislu opisovanja splošnega procesa je akcija aktiven korak v izvrševanju tega procesa (Hamilton, et al., 2006).

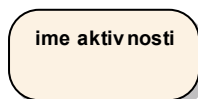
UML notacija za akcijo je zaobljen pravokotnik, znotraj katerega napišemo ime enostavnega obnašanja (slika 11).



Slika 11: Notacija za akcijo

Aktivnost (*activity*) vsebuje zaporedje akcij in/ali drugih aktivnosti – z njo prikažemo niz povezanih akcij. Z aktivnostjo lahko (na nivoju objektno usmerjenega razreda) predstavimo metodo operacije ali pa naloge, ki tvorijo poslovni proces.

Na diagramu jo prikažemo kot zaobljen pravokotnik (slika 12), ki vsebuje ime aktivnosti (enako kot pri akciji), lahko pa tudi kompleksno zaporedje akcij, aktivnosti, objekte in krmilni tok. Prav tako je možen prikaz parametrov, predpogojev (*precondition*), končnih pogojev (*postcondition*) in lastnosti aktivnosti (Rumbaugh, et al., 2004).

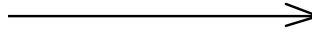


Slika 12: Notacija za aktivnost

Potrebna je pazljivost pri razlikovanju pojmov aktivnost in akcija, zlasti ker se v verziji 2.0 (za razliko od zgodnejših verzij) notaciji bistveno ne razlikujeta. Aktivnost je proces, ki se modelira (npr. pranje avtomobila), akcija pa korak znotraj aktivnosti (npr., miljenje, spiranje sušenje) (Hamilton, et al., 2006).

(Vir za opis sledečih elementov diagrama aktivnosti: Rumbaugh, et al., 2004)

Krmilni tok (*control flow*) je usmerjena povezava med akcijami oz. aktivnostmi, ki s puščico ponazarja potek izvedbe.



Slika 13: Notacija za krmilni tok

Krmilni element (*control node*): krmilni tok vodimo med množico akcij in aktivnosti s krmilnimi elementi, ki jih je več vrst:

- **Začetni element (*initial*)**: z njim začnemo niz aktivnosti in akcij, notacija je polni krog (slika 14).



Slika 14: Notacija za začetni element

- **Končni element aktivnosti (*final activity*)**: z njim zaključimo vse krmilne tokove, prikazan je s koncentričnim krogom (slika 15).



Slika 15: Notacija za končni element aktivnosti

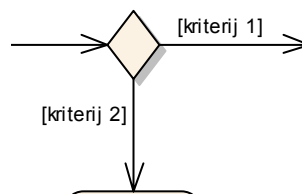
- **Končni element toka (*final flow*)** uporabimo takrat, ko želimo končati enega (in ne vseh) od tokov znotraj aktivnosti. Končni element toka nima nobenega vpliva na ostale tokove. Notacija je X znotraj majhnega kroga (slika 16).



Slika 16: Notacija za končni element toka

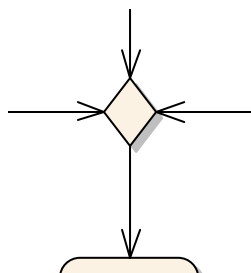
- **Element odločitve (*decision*)** uporabimo takrat, ko želimo izvesti *if-then-else* selekcijo za potek izvedbe. Na ta način naredimo test, ki zagotavlja, da poteka krmilni oz. objektni tok samo po eni poti. Notacija je romb, ki ga povežemo z vsako od

izhajajočih aktivnosti oz. akcij s puščico krmilnega toka. Na linijo krmilnega toka napišemo kriterij odločitve v oglatih oklepajih (slika 17).



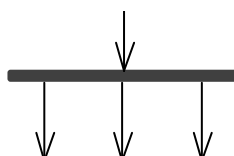
Slika 17: Notacija za element odločitve

- **Element združitve (*merge*):** z njim znova združimo različne izvedbene poti, ki so nastale z elementom odločitve (notaciji sta enaki, s tem da element združitve ne uporablja kriterijev odločitve v oglatih oklepajih – slika 18).



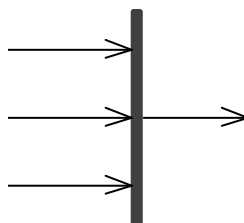
Slika 18: Notacija za element združitve

- **Element cepitve na vzporedne podaktivnosti (*fork*)** razcepi krmilni tok na dva ali več krmilnih tokov, ki se izvajajo neodvisno med seboj in vzporedno. Notacija je poudarjena črta (t.i. sinhronizacijska črta), na katero se na eni strani pripenja ena linija (krmilni tok), na drugi strani pa iz nje izhaja več linij -krmilnih tokov (slika 19).



Slika 19: Notacija za element cepitve na vzporedne podaktivnosti

- **Element združitve vzporednih podaktivnosti (*joins*)** je nasprotje predhodnega elementa – dva ali več krmilnih tokov sinhronizira v enega. Tudi tu je notacija sinhronizacijska črta (slika 20).



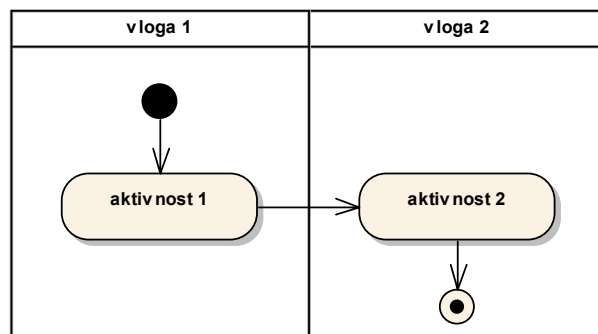
Slika 20: Notacija za element združitve vzporednih podaktivnosti

- **Povezovalni element** uporabimo, ko zmanjka prostora na diagramu in želimo nadaljevati krmilni tok na drugi lokaciji na diagramu ali na naslednji strani. Oznaka znotraj kroga nakazuje, da se tok nadaljuje tam, kjer se nahaja povezovalni element z enako oznako (slika 21).



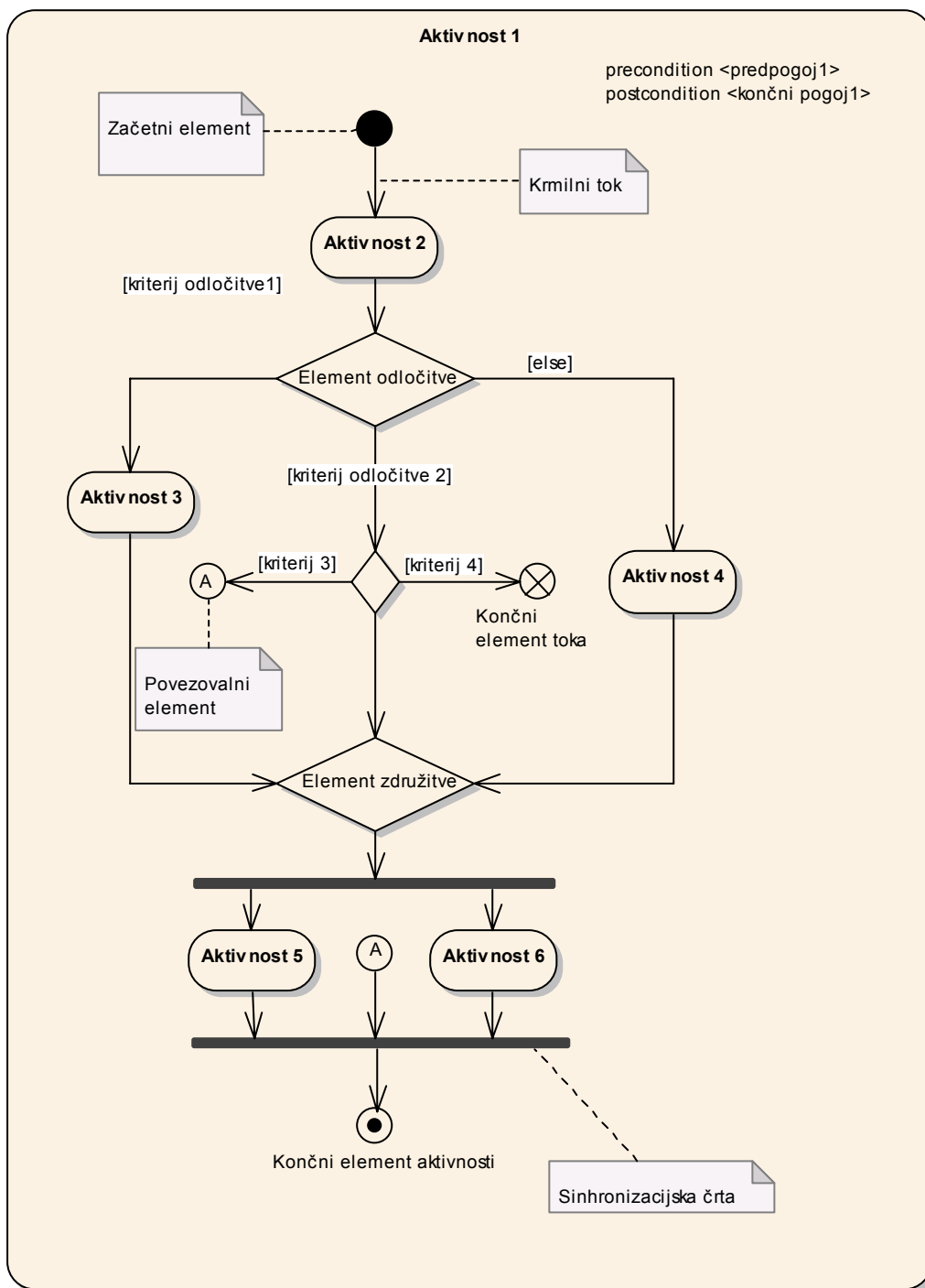
Slika 21: Notacija za povezovalni element

Steze (*swim lanes*) uporabimo, kadar želimo izrecno pokazati, kdo nosi odgovornost za posamezno aktivnost. Diagram aktivnosti vertikalno (lahko tudi horizontalno) razdelimo na vzporedne pasove – steze. Na vrhu (ali ob strani) vsake steze napišemo ime osebe, zaposlitveno vlogo ali organizacijsko enoto, ki je odgovorna za izvajanje aktivnosti na tem pasu. Aktivnosti in akcije pod odgovornostjo ene osebe (vloge, organizacijske enote) postavimo znotraj odgovarjajoče steze (slika 22).



Slika 22: Prikaz uporabe stez

Na sliki 23 je prikazana notacija za večino elementov diagrama aktivnosti.



Slika 23: Pregled notacije elementov diagrama aktivnosti

3 PRIKAZ POSTOPKOV UREJANJA IN EVIDENTIRANJA MEJE (ZEN) Z UML DIAGRAMI

3.1 Akterji

3.1.1 Opredelitev akterjev

Preglednica 3: Akter geodetsko podjetje

Ime akterja	Geodetsko podjetje
Opis	Geodetsko podjetje je samostojni podjetnik posameznik ali gospodarska družba, ki izpolnjuje pogoje za opravljanje geodetskih storitev.
Vloga	<ul style="list-style-type: none"> • svetuje stranki • ima vpogled do podatkov Centralne baze nepremičnin • pridobiva podatke na geodetski upravi za izdelavo elaborata ureditve meje • izvede potrebne meritve in obravnave s strankami na terenu • izdelava elaborat ureditve meje • po pooblastilu stranke vloži zahtevo za uvedbo postopka evidentiranja urejene meje

Preglednica 4: Akter geodet

Ime akterja	Geodet
Opis	Oseba, ki ji je bila izdana geodetska izkaznica v skladu z zakonom, ki ureja geodetsko dejavnost.
Vloga	<ul style="list-style-type: none"> • Dejansko izvaja postopke, ki so naloga geodetskega podjetja.

Preglednica 5: Akter odgovorni geodet

Ime akterja	Odgovorni geodet
Opis	Geodet, ki izpolnjuje pogoje za odgovornega geodeta - določa jih zakon, ki ureja geodetsko dejavnost.
Vloga	<ul style="list-style-type: none">• Lahko izvaja vse postopke, ki so naloga geodetskega podjetja.• S podpisom odgovarja za pravilno izvedbo postopkov v domeni geodetskega podjetja.

Preglednica 6: Akter stranka

Ime akterja	Stranka
Opis	Stranka sodeluje pri mejni obravnavi ter v upravnem postopku, ki ga vodi geodetska uprava. Lahko je naročnik ali pasivna stranka .

Preglednica 7: Akter naročnik

Ime akterja	Naročnik
Opis	<p>Naročnik naroči izvedbo tehničnega postopka in (ali) vložijo zahtevo za izvedbo upravnega postopka. To je lahko:</p> <ul style="list-style-type: none">• Lastnik (lahko tudi katerikoli solastnik ali skupni lastnik) nepremičnine, ki je predmet postopka, kar je najbolj običajno.• Pridobitelj je oseba, ki je na podlagi pravnega posla pridobila pravico, da se kot lastnik vpiše v zemljiško knjigo, in je tudi že vložila predlog za vpis, ta pa še ni izveden.

	<ul style="list-style-type: none"> • Državni organ, organ samoupravnih lokalnih skupnosti in drugih subjektov, če tako določa zakon - pri zemljiščih, katerih lastnik je Republika Slovenija ali samoupravna lokalna skupnost ali so javno dobro, se v zemljiški kataster vpiše, kateri organ oziroma drug subjekt jih v skladu s predpisi ali na drugi pravni podlagi upravlja. (Primer: pri gradnji avtoceste država kot investitor poskrbi in tudi krije stroške ureditve meja.) • Geodetska uprava lahko naroči elaborat zaradi uvedbe postopka evidentiranja meje po uradni dolžnosti. • Zakoniti zastopnik zastopa mladoletne otroke in stranke, ki nimajo procesne sposobnosti. • Skupni predstavnik - v primeru, če dvoje ali več strank nastopa v isti zadevi. • Pooblaščenec - lahko ga določi stranka ali zakoniti zastopnik.
Vloga	<ul style="list-style-type: none"> • vloži zahtevo, ki sproži upravni postopek ali tehnični postopek • dopolni zahtevo • vabljen je k dejanjem upravnega postopka • ima pravico do pritožbe na odločbo oz. sklep

Preglednica 8: Akter pasivna stranka

Ime akterja	Pasivna stranka
Opis	<p>Pasivna stranka je stranka, ki ima pravico udeležiti se postopka, ni pa postopka sprožila. To je lahko:</p> <ul style="list-style-type: none"> • Lastnik (s tem je mišljen tudi solastnik in skupni lastnik) • Pridobitelj • Začasni zastopnik - določi ga geodetska uprava, če je lastnik neznan, če je neznan njegovo prebivališče ali če je umrl, pa dediči niso znani. • Zakoniti zastopnik • Skupni predstavnik

	<ul style="list-style-type: none">• Pooblaščenec• Stranski udeleženec je oseba, ki izkaže pravni interes in zatrjuje, da vstopa v postopek zaradi varstva svojih pravnih koristi (npr. imetnik služnostne pravice). V kolikor stranka upravnemu organu naznani pravni interes in želi vstopiti v postopek o tem upravni organ odloči s sklepom, zoper katerega je možna posebna pritožba. Stranskemu udeležencu oziroma osebi, ki bi morala biti stranski udeleženec mora upravni organ vročiti odločbo.
Vloga	<ul style="list-style-type: none">• vabljen je k dejanjem upravnega in tehničnega postopka• ima pravico do pritožbe na odločbo oz. sklep

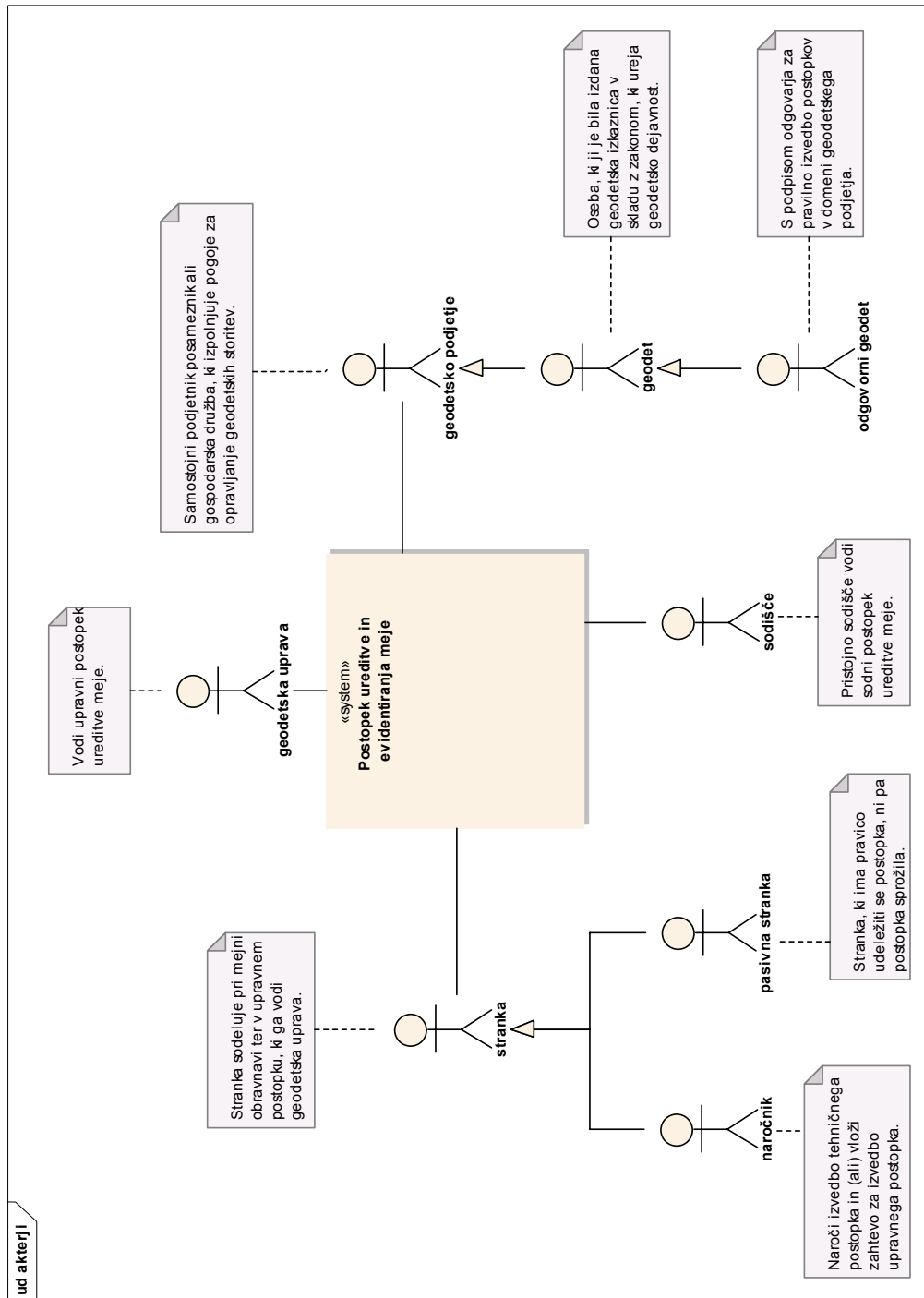
Preglednica 9: Akter geodetska uprava

Ime akterja	Geodetska uprava
Vloga	Vodi upravni postopek ureditve meje.

Preglednica 10: Akter sodišče

Ime akterja	Sodišče
Vloga	Pristojno sodišče vodi sodni postopek ureditve meje.

3.1.2 Pregledni diagram primerov uporabe za postopek urejanja in evidentiranja meje



Slika 24: Pregledni diagram primerov uporabe za postopek urejanja in evidentiranja meje

3.2 Glavni potek dogodkov za postopek urejanja in evidentiranja meje

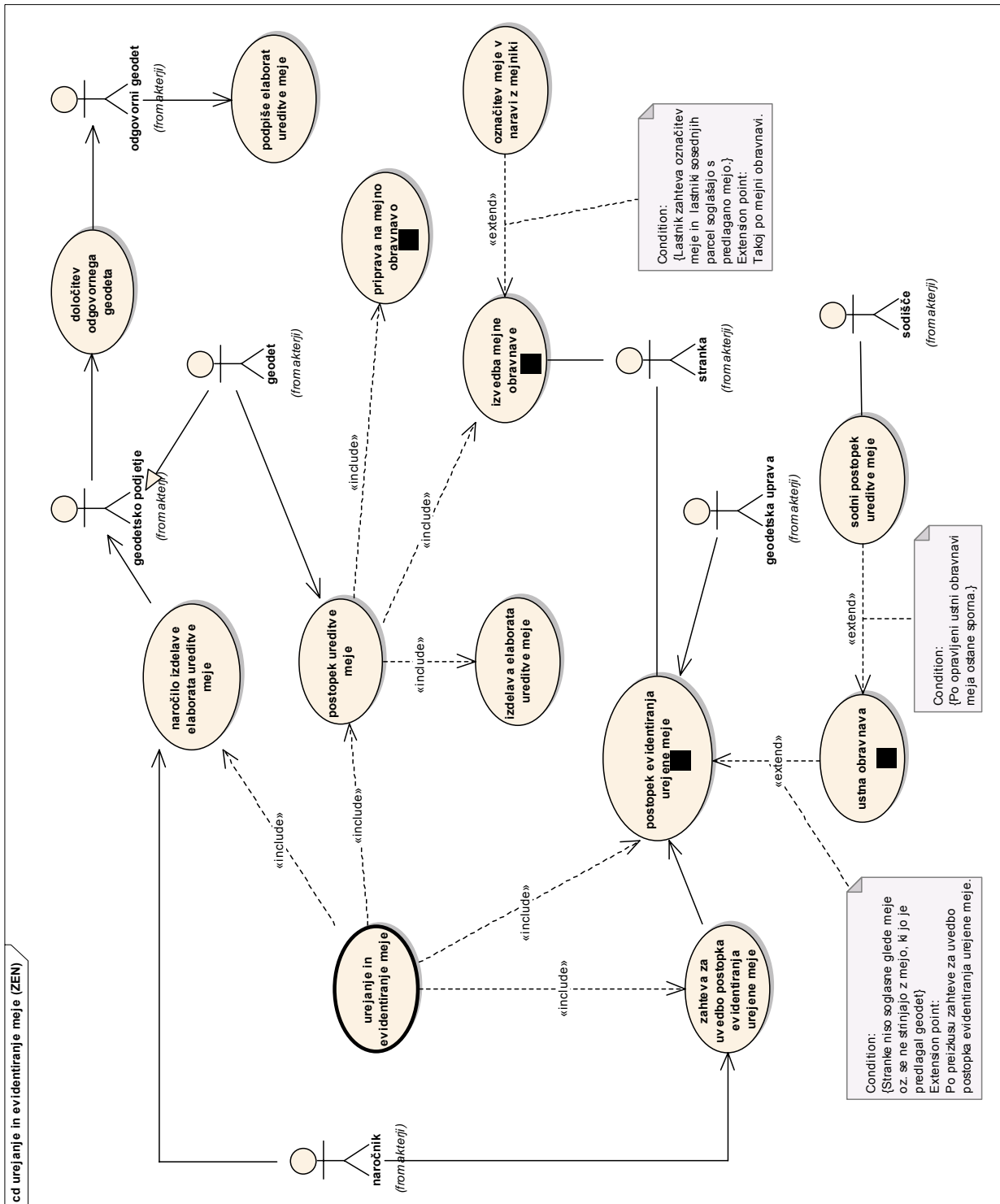
Preglednica 11 vsebuje temeljne korake idealnega (najkrajšega možnega) poteka postopka urejanja in evidentiranja meje.

Preglednica 11: Glavni potek dogodkov za postopek urejanja in evidentiranja meje

Pojavi se potreba po ureditvi meje – npr. lastnik želi prodati parcelo; v prihodnosti se bo izvedel poseg v prostor na območju, ki zajema tudi parcelo, katere meja še ni urejena; prihaja do nesoglasij lastnikov parcel, katerih skupna meja ni urejena ipd.
Naročnik naroči izdelavo elaborata ureditve meje pri geodetskem podjetju.
Geodetsko podjetje imenuje odgovornega geodeta (pogoje za odgovornega geodeta določa zakon, ki ureja geodetsko dejavnost).
Geodet zbere podatke , potrebne za izvedbo mejne obravnave.
Geodet opravi predhodne meritve .
Geodet povabi stranke na mejno obravnavo.
Na mejni obravnavi geodet pokaže mejo po podatkih zemljiškega katastra,
Stranke pokažejo , kje po njihovem poteka meja.
Stranke soglašajo glede poteka predlagane meje, ki se ne razlikuje od meje po podatkih zemljiškega katastra, upoštevajoč natančnost njegovih podatkov.
Geodet izdelava elaborat ureditve meje na podlagi mejne obravnave (elaborat ureditve meje vsebuje predlog meje, ki naj se kot urejena evidentira v zemljiškem katastru, in zapisnik mejne obravnave)
Odgovorni geodet pregleda in podpiše elaborat ureditve meje.
Naročnik oz. geodetsko podjetje (če ga je pooblastil naročnik) vloži zahtevo za uvedbo postopka evidentiranja urejene meje na geodetsko upravo.
Geodetska uprava po skrajšanem ugotovitvenem postopku izda odločbo o evidentiranju urejene meje.
Na podlagi dokončne odločbe geodetska uprava vpiše mejo v zemljiškem katastru kot urejeno.

Preglednica 11 je bila podlaga za izdelavo diagrama primerov uporabe.

3.3 Diagram primerov uporabe za postopek urejanja in evidentiranja meje



Slika 25: Diagram primerov uporabe za postopek urejanja in evidentiranja meje

Zgornji diagram podaja pregleden prikaz primerov uporabe, ki sestavljajo postopek urejanja in evidentiranja meje. Naslednji primeri uporabe so podrobneje razloženi:

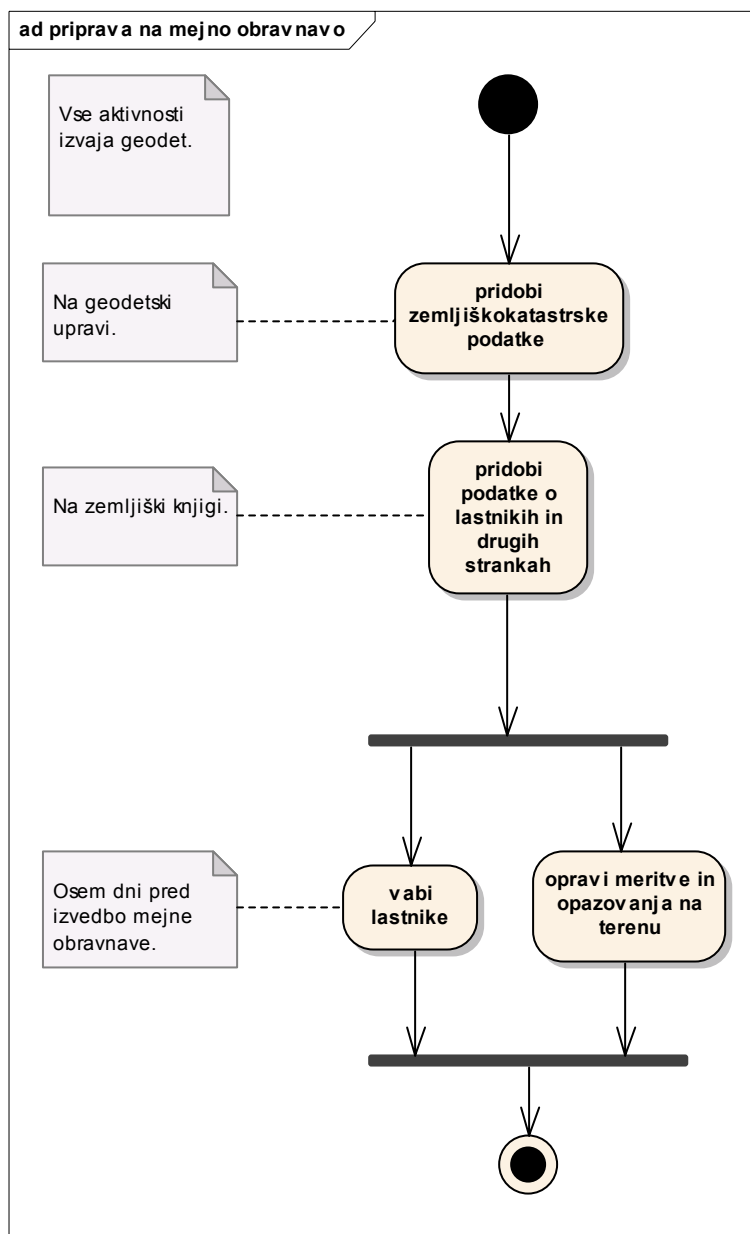
- priprava na mejno obravnavo,
- izvedba mejne obravnave,
- izdelava elaborata ureditve meje,
- postopek evidentiranja urejene meje in
- ustna obravnavo.

3.3.1 Primer uporabe »priprava na mejno obravnavo«

Preglednica 12: Podroben opis primera uporabe »priprava na mejno obravnavo«

Ime primera uporabe	Priprava na mejno obravnavo
Cilj	Pripraviti vse potrebne podatke, da se bo mejna obravnavo lahko tekoče izvedla.
Predpogoji	Naročnik naroči izdelavo elaborata ureditve meje.
Primarni akterji	geodet
Glavni potek dogodkov	1. Geodet pridobi zemljiškokatastrske podatke , potrebne za ureditev meje, pri geodetski upravi.
	2. Geodet pridobi podatke o lastnikih parcel in drugih strankah pri zemljiški knjigi.
	3. Geodet brez sodelovanja lastnikov opravi meritve in opazovanja na terenu z namenom, da ugotovi potek meje po podatkih zemljiškega katastra.
	4. Geodet vabi lastnika parcele , katere meja se ureja, na mejno obravnavo vsaj 8 dni pred njeno izvedbo.
	5. Geodet vabi lastnike sosednjih parcel na mejno obravnavo vsaj 8 dni pred njeno izvedbo.

Na podlagi podrobnega opisa primera uporabe sem izdelala diagram aktivnosti (slika 26).



Slika 26: Diagram aktivnosti »priprava na mejno obravnavo«

3.3.2 Primer uporabe »izvedba mejne obravnave«

Preglednica 13: Podroben opis primera uporabe »izvedba mejne obravnave«

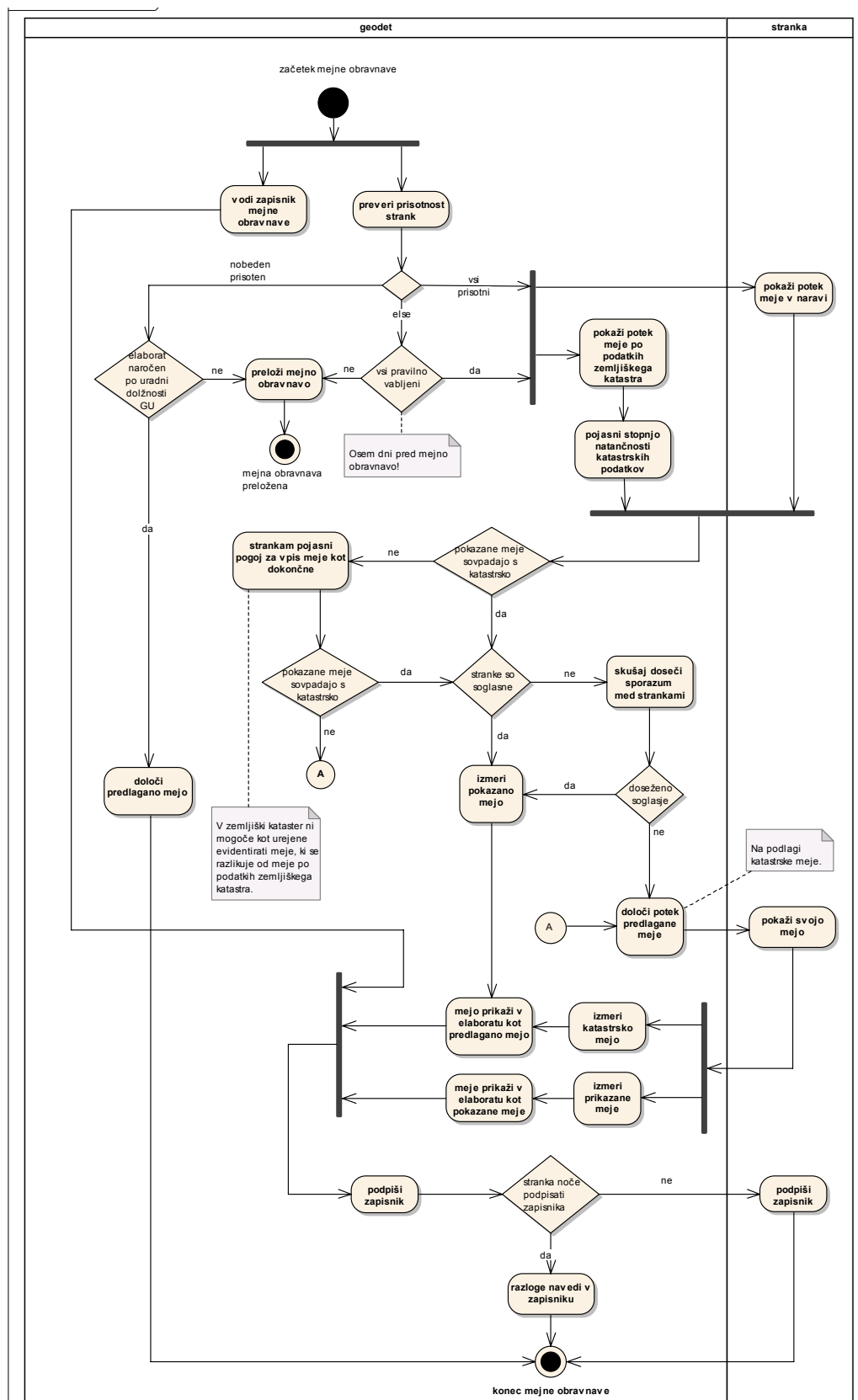
Ime primera uporabe	izvedba mejne obravnave
Cilj	Geodet na terenu določi potek predlagane meje.
Predpogoji	Stranke morajo biti povabljeni na mejno obravnavo vsaj osem dni pred njeno izvedbo.
	V vabilu je treba navesti zakonite posledice izostanka z mejne obravnave.
	Če se stranka udeleži mejne obravnave, čeprav ni bila vabljeni vsaj osem dni prej, se šteje, da je bila vabljeni pravilno.
Uspešni končni pogoji	V zapisniku mejne obravnave je prikazana predlagana meja (in pokazane meje).
Neuspešni končni pogoji	Mejna obravnava se ne izvede.
Primarni akterji	geodet
Sekundarni akterji	stranka
Glavni potek dogodkov	1. Geodet preveri prisotnost strank – vse stranke so prisotne.
	2. Geodet sproti vodi zapisnik mejne obravnave. V njem se navedejo KRAJ in ČAS njene izvedbe, VABLJENI in PRISOTNI udeleženci ter opiše njen POTEK.
	3. Stranke za svoja zemljišča geodetu pokažejo oziroma natančno opišejo potek meje v naravi.
	4. Stranke - lastniki parcel, ki se jih dotika meja, ki se ureja, pokažejo oziroma opišejo samo točke, kjer se zaključita ta meja, ali samo izjavijo, da meja, ki se ureja, ne posega na njihovo zemljišče.
	5. Geodet pokaže potek meje po podatkih zemljiškega katastra.
	6. Geodet stranke opozori na stopnjo natančnosti katastrskih podatkov.
	7. Pokazana meja se ne razlikuje od meje po podatkih zemljiškega katastra, stranke pa soglašajo o poteku meje.

	8. Geodet izmeri pokazano mejo
	9. Geodet prikaže pokazano mejo v zapisniku kot predlagano mejo.
	10. Geodet podpiše zapisnik.
	11. Stranke podpišejo zapisnik.
»extend« primeri uporabe	»označitev meje v naravi z mejniki« Geodet označi mejo v naravi z mejniki takoj po mejni obravnavi, če tako zahteva lastnik parcele, in če lastniki sosednjih parcel soglašajo s predlagano mejo.

Preglednica 14: Alternativni poteki dogodkov primera uporabe »izvedba mejne obravnave«

1. Mejne obravnave se ne udeleži nobena stranka.	
1.1.	Če je bil elaborat naročen zaradi uvedbe postopka evidentiranja meje po uradni dolžnosti geodetske uprave, geodet določi predlagano mejo po podatkih iz zemljiškega katastra.
1.2.	Če postopek ni bil uveden po uradni dolžnosti, geodet preloži mejno obravnavo.
2. Vse strank niso prisotne na mejni obravnavi.	
2.1.	Če se katera od strank ne udeleži mejne obravnave, kljub temu da je bila nanjo pravilno vabljen, se lahko mejna obravnava opravi brez nje – alternativni potek se nadaljuje s korakom 3 glavnega poteka.
2.2.	Če manjkajoča stranka ni bila pravilno vabljen, se mejna obravnava ne opravi.
3. Meja, ki so jo pokazale stranke (pokazana meja), se razlikuje od meje po podatkih zemljiškega katastra.	
3.1.	Geodet strankam pojasni , da v zemljiški kataster ni mogoče kot urejene evidentirati meje, ki se razlikuje od meje po podatkih zemljiškega katastra.
3.2.	Če se po tem pokazana meja ne razlikuje od meje po podatkih zemljiškega katastra, se alternativni potek nadaljuje s korakom 8 glavnega poteka.
3.3.	Če se lastniki ne sporazumejo o meji, se alternativni potek nadaljuje z alternativnim potekom 4.
4. Stranke ne soglašajo s potekom meje.	
4.1.	Geodet skuša doseči sporazum med strankami.
4.2.	Če geodet doseže sporazum, se potek nadaljuje s korakom 8 glavnega poteka.

4.3. Če geodet ne doseže sporazuma, določi potek predlagane meje .
4.4. Stranke , ki ne soglašajo s predlagano mejo geodeta, pokažejo svojo mejo .
4.5. Geodet izmeri pokazane meje in predlagano mejo.
4.6. Geodet v elaboratu ureditve meje prikaže pokazane meje in predlagano mejo.
4.7. Alternativni potek se nadaljuje s korakom 10 glavnega poteka.
5. Stranka noče podpisati zapisnika.
Geodet to dejstvo in razloge za odklonitev podpisa navede v zapisniku.



Slika 27: Diagram aktivnosti »izvedba mejne obravnave«

3.3.3 Primer uporabe »izdelava elaborata ureditve meje«

Preglednica 15: Podroben opis primera uporabe »izdelava elaborata ureditve meje«

Ime primera uporabe	izdelava elaborata ureditve meje
Cilj	Izdelava elaborata ureditve, ki ga je potrebno priložiti zahtevi za uvedbo postopka evidentiranja urejene meje.
Predpogoji	Izvedena mejna obravnava.
Primarni akterji	geodet
Opis	Elaborat ureditve meje vsebuje: <ul style="list-style-type: none">• PREDLOG MEJE, ki naj se kot urejena evidentira v zemljiškem katastru (predlagana meja),• ZAPISNIK MEJNE OBRAVNAVE.
	Predlagana meja se ne sme razlikovati od meje po podatkih zemljiškega katastra, upoštevajoč natančnost njegovih podatkov.
	Če se meja, ki so jo pokazale stranke (pokazana meja), razlikuje od predlagane meje, vsebuje elaborat ureditve meje tudi PRIKAZ POTEKA POKAZANE MEJE.

3.3.4 Primer uporabe »postopek evidentiranja urejene meje«

Preglednica 16: Podroben opis primera uporabe »postopek evidentiranja urejene meje«

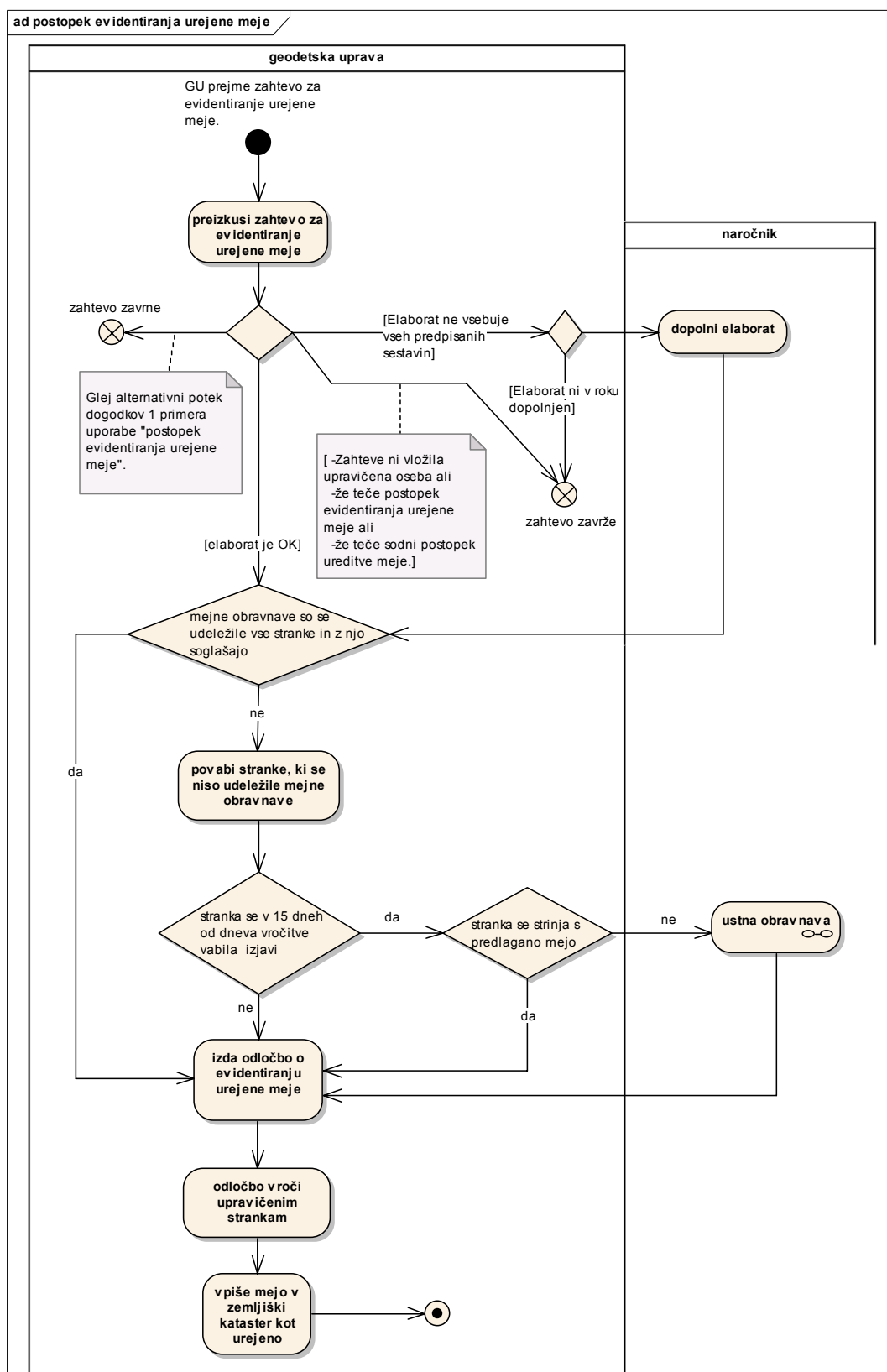
Ime primera uporabe	postopek evidentiranja urejene meje
Cilj	Po podatkih elaborata ureditve meje se v zemljiškem katastru evidentira urejena meja.
Predpogoji	Naročnik (ali pooblaščen geodetsko podjetje) je vložil zahtevo za uvedbo postopka evidentiranja urejene meje na geodetsko upravo.
	Zahtevi je treba priložiti elaborat ureditve meje.
Uspešni končni pogoji	Meja je vpisana v zemljiškem katastru kot urejena.
Primarni akterji	geodetska uprava
Sekundarni akterji	stranka
Glavni potek dogodkov	1. Geodetska uprava preizkusi zahtevo za evidentiranje urejene meje.
	2. Geodetska uprava pri tem ugotovi , da: <ul style="list-style-type: none"> • elaborat ureditve meje izpolnjuje vse pogoje, • so se mejne obravnave udeležile vse stranke, • stranke soglašajo s predlagano mejo.
	3. Geodetska uprava po skrajšanem ugotovitvenem postopku izda odločbo o evidentiranju urejene meje.
	4. Geodetska uprava odloči o evidentiranju urejene meje obvezno priloži grafični prikaz evidentirane meje z označenimi zemljiškokatastrskimi točkami ter vpisanimi parcelnimi številkami.
	5. Geodetska uprava vroči odločbo o evidentiranju urejene meje upravičenim strankam.
	6. Geodetska uprava na podlagi dokončne odločbe vpíše mejo v zemljiškem katastru kot urejeno.

»extend« primeri uporabe	»ustna obravnava« Opravi jo geodetska uprava v uradnih prostorih, če: <ul style="list-style-type: none">• je v elaboratu ureditve meje poleg predlagane prikazana ena ali več pokazanih mej ali• je stranka, ki se ni udeležila mejne obravnave, izjavila, da se ne strinja s predlagano mejo.
--------------------------	---

Preglednica 17: Alternativni poteki dogodkov primera uporabe »postopek evidentiranja urejene meje«

1. Geodetska uprava zavrne zahtevo za evidentiranje urejene meje.
Razlogi so lahko naslednji:
1.1. Meja je urejena po ZEN ali sodnem postopku in se v zahtevi za evidentiranje urejene meje predlaga nova ureditev meje in ne njena natančnejša določitev.
1.2. Elaborata ureditve meje ni izdelalo geodetsko podjetje, ki izpolnjuje pogoje za opravljanje geodetskih storitev, v skladu z zakonom, ki ureja geodetsko dejavnost.
1.3. Elaborat ureditve meje je kot odgovorni geodet podpisala oseba, ki ne izpolnjuje pogojev za odgovornega geodeta geodetskih storitev, v skladu z zakonom, ki ureja geodetsko dejavnost.
1.4. Mejno obravnavo je izvedla oseba, ki ne izpolnjuje z zakonom predpisanih pogojev.
1.5. Elaborat ureditve meje je izdelalo geodetsko podjetje, ki je v postopku nastopalo kot stranka.
1.6. Postopek urejanja meje izvedel geodet, ki je v postopku nastopal kot stranka oz. je s stranko v sorodu (zakonski zvezi).
1.7. Elaborat ureditve meje je potrdil odgovorni geodet, ki je v postopku nastopal kot stranka oz. je s stranko v sorodu (zakonski zvezi).
1.8. Podatki ne omogočajo evidentiranja predlagane meje v zemljiškem katastru.
1.9. Na mejni obrnavi ni bila zagotovljena udeležba vsem upravičnim lastnikom.
1.10. Niso izpolnjeni drugi predpisani pogoji za izdajo odločbe o evidentiranju urejene meje.
2. Geodetska uprava zavrže zahtevo za evidentiranje urejene meje
Razlogi so lahko naslednji:

2.1. Naročnik ni v določenem roku dopolnil elaborata, ki ni vseboval vseh predpisanih sestavin.
2.2. Zahteve ni vložila upravičena oseba.
2.3. Postopek evidentiranja urejene meje že teče.
2.4. Sodni postopek ureditve meje že teče.
3. Vse stranke se niso udeležile mejne obravnave.
3.1. Geodetska uprava povabi stranke, ki se niso udeležile mejne obravnave, da se izjavijo o tem, ali se strinjajo s potekom predlagane meje.
3.2. Vabilo se vroči z osebnim vročanjem na podlagi zakona o splošnem upravnem postopku.
3.3. Geodetska uprava vabilu priloži kopijo skice iz elaborata ureditve meje.
3.4. Geodetska uprava stranke tudi obvesti , kje in kdaj si lahko ogledajo ta elaborat, ter jih opozori na pravne posledice, če se ne bodo izjavile o poteku predlagane meje.
3.5. Če se vabljen stranka v 15 dneh od dneva vročitve ne izjavi, se šteje, da se strinja s potekom predlagane meje – alternativni potek se nadaljuje s korakom 3 glavnega poteka.



Slika 28: Diagram aktivnosti »postopek evidentiranja urejene meje«

Diagram aktivnosti »postopek evidentiranja urejene meje« vsebuje povezavo na diagram aktivnosti »ustna obravnava«.

3.3.5 Primer uporabe »ustna obravnava«

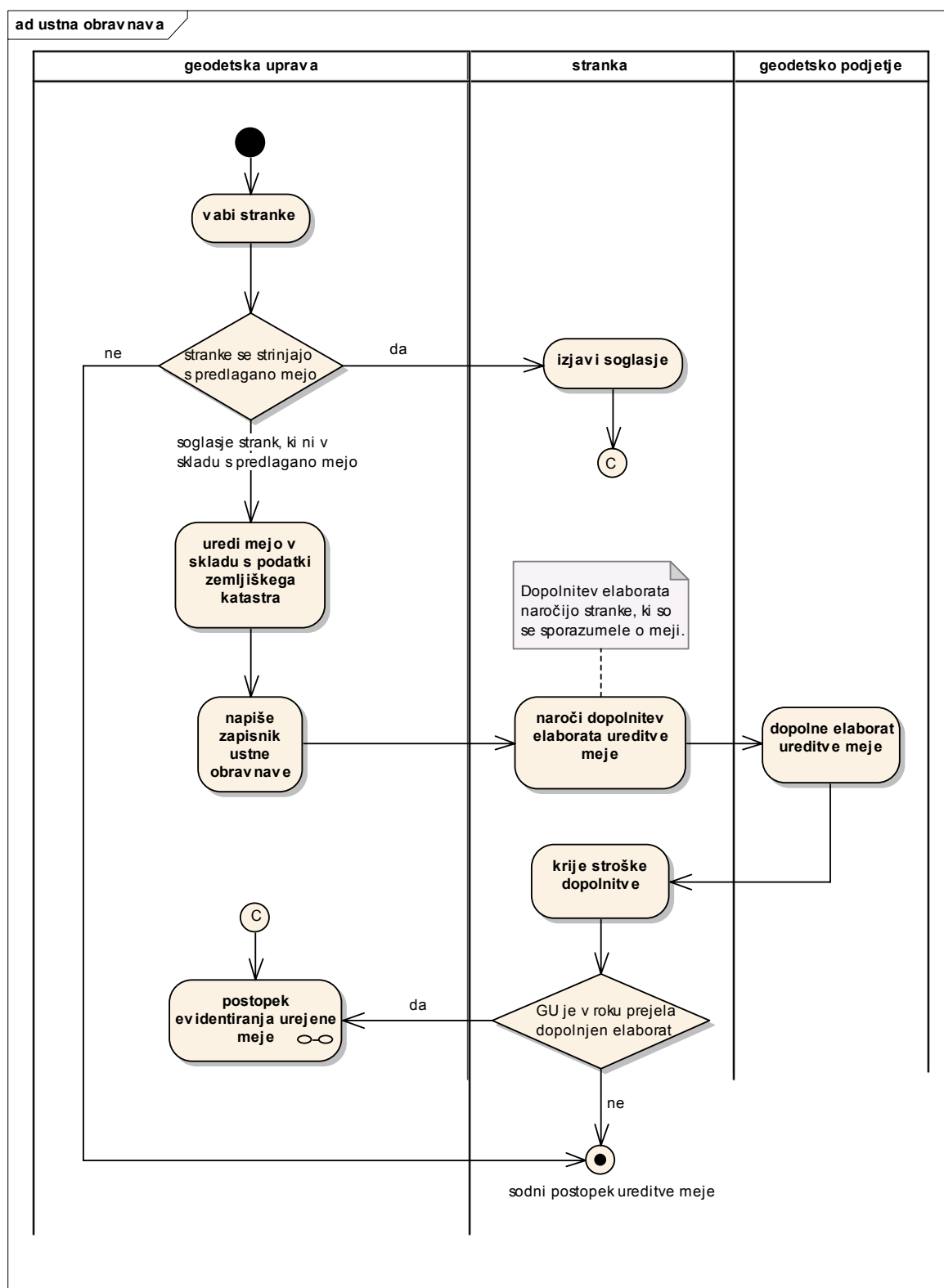
Preglednica 18: Podroben opis primera uporabe »ustna obravnava«

Ime primera uporabe	ustna obravnava
Cilj	Sporazum strank o poteku meje.
Predpogoji	V elaboratu ureditve meje je poleg predlagane prikazana ena ali več pokazanih mej ali pa je
	stranka, ki ni bila prisotna na mejni obravnavi, izjavila, da se ne strinja s predlagano mejo.
Uspešni končni pogoji	Meja, o kateri so se stranke sporazumele, se ne razlikuje od meje po podatkih zemljiškega katastra.
Neuspešni končni pogoji	Po ustni obravnavi meja ostane sporna.
Primarni akterji	geodetska uprava
Sekundarni akterji	stranka, geodetsko podjetje
Glavni potek dogodkov	1. Geodetska uprava vabi na ustno obravnavo stranke, ki so pokazale drugačno mejo od predlagane.
	2. Geodetska uprava vabi na ustno obravnavo stranke, ki niso bile prisotne na mejni obravnavi, in so pisno izjavile, da se s potekom predlagane meje ne strinjajo.
	3. Vabilo se vroči z osebnim vročanjem na podlagi zakona, ki ureja splošni upravni postopek, najmanj osem dni pred ustno obravnavo.
	4. Stranke izjavijo soglasje k poteku predlagane meje.
	5. Potek dogodkov se nadaljuje s korakom 3 glavnega poteka primera uporabe »postopek evidentiranja urejene meje«.

»extend« primeri uporabe	»sodni postopek ureditve meje« Če po opravljeni ustni obravnavi meja ostane sporna, geodetska uprava pozove stranke, ki se ne strinjajo s predlagano mejo, da v 30 dneh od prejema poziva začnejo sodni postopek ureditve meje pred pristojnim sodiščem.
--------------------------	--

Preglednica 19: Alternativni potek dogodkov primera uporabe »ustna obravnava«

Na ustni obravnavi se izkaže, da bi se stranke lahko sporazumele o poteku meje, vendar ne tako, kakor je predlagano v elaboratu ureditve meje.
1. Geodetska uprava uredi mejo na podlagi soglasja strank, pri čemer se meja ne sme razlikovati od meje po podatkih zemljiškega katastra.
2. Geodetska uprava mejo, urejeno v skladu s 1. točko, prikaže v zapisniku ustne obravnave.
3. Stranke , ki so se na ustni obravnavi sporazumele o meji, pri geodetskem podjetju naročijo dopolnitev elaborata ureditve meje.
4. Stranke , ki so se na ustni obravnavi sporazumele o meji, krijejo stroške dopolnitev elaborata.
5. Če geodetska uprava v določenem roku prejme dopolnjeni elaborat ureditve meje, se potek dogodkov nadaljuje s korakom 3 glavnega poteka primera uporabe »postopek evidentiranja urejene meje«.
6. Če geodetska uprava dopolnjenega elaborata ne prejme v roku, se potek dogodkov nadaljuje s primerom uporabe »sodni postopek ureditve meje«.



Slika 29: Diagram aktivnosti »ustna obravnava«

Diagram aktivnosti »ustna obravnava« vsebuje povezavo na diagram aktivnosti »postopek evidentiranja urejene meje«.

4 ZAKLJUČEK

Model je poenostavljena ponazoritev stvarnosti ali pa nečesa neobstoječega oz. načrtovanega. Vsebuje pomembne vidike izbranega dela stvarnosti, poenostavi ali izpusti pa vse, ki so v smislu zahtev, potreb in ciljev, ki jih mora model zadovoljiti, zanemarljivi.

Modeli predstavljajo temelj delovanja inženirskih strok. Načrtovalci modela morajo raziskati potrebe uporabnikov, glavni namen, sestavo in obliko končnega izdelka. Poleg tega, da model predstavlja sistem ali organizacijo, olajša njegovo razumevanje in omogoči predstavo, kakšen bo načrtovani sistem, predstavlja tudi osnovno izhodišče razvoja, saj določa funkcijo, sestavo in obnašanje sistema.

Model sistema se formalno opiše s pomočjo izbranega jezika za modeliranje. UML je objektno usmerjen formalen jezik za analizo, opredelitev, grafično ponazoritev, načrtovanje sestavin, izvedbo ter dokumentacijo softverskih, informacijskih, poslovnih in drugih sistemov. Sicer je bil razvit z namenom podpore softverskemu modeliranju, a je uporaben tudi na drugih področjih informacijske tehnologije, kot so recimo GIS in poslovni sistemi.

Tehnični odbor ISO TC 211, ki razvija mednarodne standarde ISO za področje geografskih informacij in geoinformatike (oznaka ISO 191**), ga je izbral za formalni jezik za modeliranje, podajanje vsebine in sestave standardov.

UML se lahko uporabi tudi za modeliranje zapletenih postopkov, kot so npr. postopki v katastrih nepremičnin. Za ta namen je potrebno razviti ustrezno metodologijo, saj UML definira jezik za opredelitev in formalni zapis razvitih modelov ter grafično notacijo, ne pa tudi tehnik za razvoj modelov.

Metodologija za modeliranje postopkov v katastrih nepremičnin s pomočjo UML diagramov naj torej sestoji iz naslednjih korakov:

- zbiranje podatkov o postopkih, ki jih bomo modelirali, ter uporabniških zahtev, ki jim bo moral izdelani model ugoditi,
- določitev akterjev ter njihova opredelitev (definicija in njihova vloga v podrobno opisanih in razčlenjenih postopkih),
- grafičen prikaz akterjev na preglednem diagramu primerov uporabe,
- določitev temeljnih korakov najkrajšega možnega poteka postopka, ki privede do želenih rezultatov, s čimer se opredelijo glavni cilji opisanega postopka,

- okvirna določitev primerov uporabe,
- izdelava diagrama primerov uporabe za najkrajši možni potek postopka,
- podrobni opisi za zapletene primere uporabe, s katerimi obsežnejše primere uporabe razčlenimo na posamezne aktivnosti in ugotovimo možne alternativne poteke dogodkov,
- grafičen prikaz podrobnih opisov primerov uporabe na diagramih aktivnosti.

Koraki se seveda ne izvajajo neodvisno eden od drugega. Posamezni vidiki sistema so medsebojno povezani in si ne smejo biti v nasprotju, enako velja za korake, ki tvorijo proces izdelave modela sistema. Kronologija razvoja modela je torej postopna in iterativna.

LITERATURA IN VIRI

Bennett, S., Skelton, J., Lunn, K. 2005. Schaum's outline of UML (Second Edition). New York, McGraw – Hill: 398 str.

Bennett, S., Skelton, J., Lunn, K. 2001. Schaum's outline of UML. New York, McGraw – Hill: 360 str.

Eriksson, H.E., Penker, M., Lyons, B., Fado, D. 2004. UML™ 2 Toolkit. Indianapolis, Indiana, Wiley Publishing, Inc.: 511 str.

Chonoles, M.J., Schardt, J.A. 2003. UML 2 for Dummies. New York, Hungry Minds: 412 str.

Fowler, M. 2004. UML distilled: a brief guide to the standard object modeling language. Boston, Addison-Wesley: 175 str.

Hamilton, K., Miles, R. 2006. Learning UML 2.0. Sebastopol, O'Reilly Media: 286 str.

Lahajnar, S., Rožanec, A. 2001. Objektno načrtovanje podatkovnih baz: 9 str.
www.drustvo-informatika.si/dogodki/arhiv/dsi2001/sekcija_a/lahajnar_rozanec.doc.
(17.3.2006)

Malnarič, G. 2003. Zamenjava informacijskega sistema v podjetju Adria Mobil, d.o.o. Diplomsko delo. Univerza v Ljubljani, Ekonomska fakulteta: 65 str.

Miličev, D., Zarić, M., Piroćanac, N. 2001. Objektno orijentisano modelovanje na jeziku UML. Beograd, Mikro knjiga: 254 str.

Rumbaugh, J., Jacobson, I., Booch, G. 2004. The Unified Modeling Language Reference Manual Second Edition. Boston, Addison Wesley: 752 str.

Rumbaugh, J., Jacobson, I., Booch, G. 1999. The Unified Modeling Language Reference Manual. Boston, Addison Wesley: 568 str.

Scott, K. 2001. UML Explained. Boston, Addison-Wesley: 151 str.

Slokar, T. 2005. Objektno orientirano modeliranje informacijskega sistema operativnega procesa v centru vodenja elektrodistribucije. Magistrsko delo. Ljubljana, Univerza v Ljubljani, Ekonomska fakulteta: 132 str.
<http://www.cek.ef.uni-lj.si/magister/slokar546.pdf>. (17.3.2006)

Šumrada, R. 2005. Tehnologija GIS. Ljubljana, Fakulteta za gradbeništvo in geodezijo: 330 str.

Zakon o evidentiranju nepremičnin (ZEN). UL RS št. 47/2006: 001-22-61/06.

Elektronski viri

Introduction to OMG's Unified Modeling Language™ (UML®).
http://www.omg.org/gettingstarted/what_is_uml.htm#12DiagramTypes (7.9.2006).

Introduction to Visual Modeling with Rational Rose 2002 and UML - History of the uml.
<http://www.awprofessional.com/articles/article.asp?p=30432&seqNum=4&rl=1> (7.9.2006).

UML Basics – The use case diagram.
<http://www-306.ibm.com/software/rational/uml/> (20.3.2006).

Wikipedia – Unified Modeling Language.
http://en.wikipedia.org/wiki/Unified_Modeling_Language (7.9.2006).