

Izjavljujem da tijekom izrade ove zadaće neću od drugoga primiti niti drugome pružiti pomoć, te da se neću koristiti nedopuštenim sredstvima. Ove su radnje teška povreda Kodeksa ponašanja te mogu uzrokovati i trajno isključenje s Fakulteta. Također izjavljujem da mi zdravstveno stanje dozvoljava pisanje ove zadaće.

JMBAG

Ime i prezime

Vlastoručni potpis

GRUPA B

1. (1 bod) Programsko inženjerstvo bavi se ____sistematskim__ i ____organiziranim__ pristupom procesu izrade programske potpore.
2. (1 bod) Navedite barem tri načina izražavanja zahtjeva sustava u inženjerstvu zahtjeva:
____strukturiranim prirodnim jezikom, jezikom za opis oblikovanja, grafičkim jezikom (npr. UML), matematičkom (formalnom) notacijom____
3. (1 bod) S obzirom na sadržaj, zahtjeve klasificiramo na: _ funkcionalne_ , _ nefunkcionalne_ i ____ zahtjeve domene primjene____.
4. (1 bod) Kojim tipom veza su aktori povezani s obrascima uporabe?
____pridruživanja (asocijacije)____
5. (1 bod) Sadržaj scenarija uobičajeno obuhvaća:
 - opis početne situacije,
 - opis normalnog/standardnog tijeka događaja,
 - opis što se eventualno može dogoditi krivo (ili opis grešaka ili opis ostalih mogućnosti rada ili opis odstupanja)
 - informaciju o paralelnim aktivnostima,
 - opis stanja gdje scenarij završava.
6. (1 bod) Nacrtajte standardnu grafičku oznaku povratne poruke (engl. *reply message*) UML sekvencijskog dijagrama.



7. (1 bod) U agilnom pristupu razvoja programske potpore zasnovanom na radnom okviru Scrum

osobu koja je jedina odgovorna za upravljanje projektnim dnevnikom zaostataka (engl. *Product Backlog*) nazivamo __ **Vlasnik proizvoda (engl. Product Owner)** ____ .

8. (1 bod) Kod unificiranog procesa razvoja programske potpore, dinamička perspektiva ostvaruje se putem četiri ____ **faze** ____, koje se sastoje od jedne ili više ____ **iteracija** ____, a koje rezultiraju ____ **artefaktima (može se reći i dokumentima ili izdanjima koda)** ____ .
9. (1 bod) Princip oblikovanja objektno usmjerene arhitekture programske potpore koji potpomaže jasno raspoznati rad komponente te ujedno smanjuje utjecaj promjena te komponente na ostale povezane nazivamo ____ **Smanji međuovisnost - engl. Reduce coupling where possible** ____.
10. (1 bod) U objektno usmjerenom oblikovanju pri razradi dijagrama razreda dio vidljivog ponašanja skupa objekata opisuje se ____ **sučeljima, engl. interfaces** ____.
11. (1 bod) Liskovin princip zamjene (engl. *Liskov substitution principle*) kaže da ako se u varijablu tipa ____ **nadrazreda** ____ postavi objekt tipa ____ **podrazreda** ____, program se mora korektno izvoditi.
12. (1 bod) U procesu oblikovanja arhitekture programske potpore strukturiranje sustava obuhvaća definiciju podsustava i mehanizama ____ **komunikacije** ____.
13. (1 bod) U suradničkom alatu GIT novi sadržaji iz udaljenog repozitorija povlače se u ____ **lokalni** ____ repozitorij naredbom ____ **fetch** (priznaje se i pull) ____.

Problemski dio - **Parkirni sustav pametnog grada PSPG**

Za sustav pametnih gradova potrebno je riješiti pronalaženje slobodnih parkirnih mjesta za automobile. Rješenje problema zasnovano je na uporabi masovne podrške (engl. *crowdsourcing*). Parkirni sustav pametnog grada PSPG za svoj rad upotrebljava informacije od registriranih korisnika. Registrirani korisnici uključeni u sustav za komunikaciju s poslužiteljem PSPG -a koriste aplikaciju na pametnom telefonu. Korisnik se registrira prilikom prvog pokretanja aplikacije na pametnom telefonu adresom elektroničke pošte i tada može definirati karakteristike vozila ako je to vozilo u mogućnosti prepoznati slobodno parkirno mjesto tijekom vožnje (vidi sliku). Korisnik može registrirati najviše pet vozila.

PSPG informacije o poziciji slobodnih parkirnih mjesta prikuplja od korisnika modernih vozila tijekom vožnje putem mobilne aplikacije. Aplikacija na pametnom telefonu korisnika povezuje se s vozilom u svrhu prikupljanja podataka. To omogućava odgovarajuće univerzalno sučelje vozila (sastavni dio vozila :)) koje od parkirnih ultrazvučnih osjetnika (engl. *parking sensor*) ili kamera dobiva informaciju o prepoznatom slobodnom mjestu.

Mobilna aplikacija na pametnom telefonu je dobavljač informacija (DI) o:

- poziciji slobodnog parkirnog mjesta - primljeno od osjetnika vozila
- zauzimanju ili napuštanju parkirnog mjesta - primljeno od osjetnika vozila
- zauzimanju ili napuštanju parkirnog mjesta - primljeno od osjetnika pametnog telefona (akceleratori, GPS) za vozila koja nisu opremljena osjetnicima

Dobivene informacije se pohranjuju u središnju bazu podataka PSPG-a.

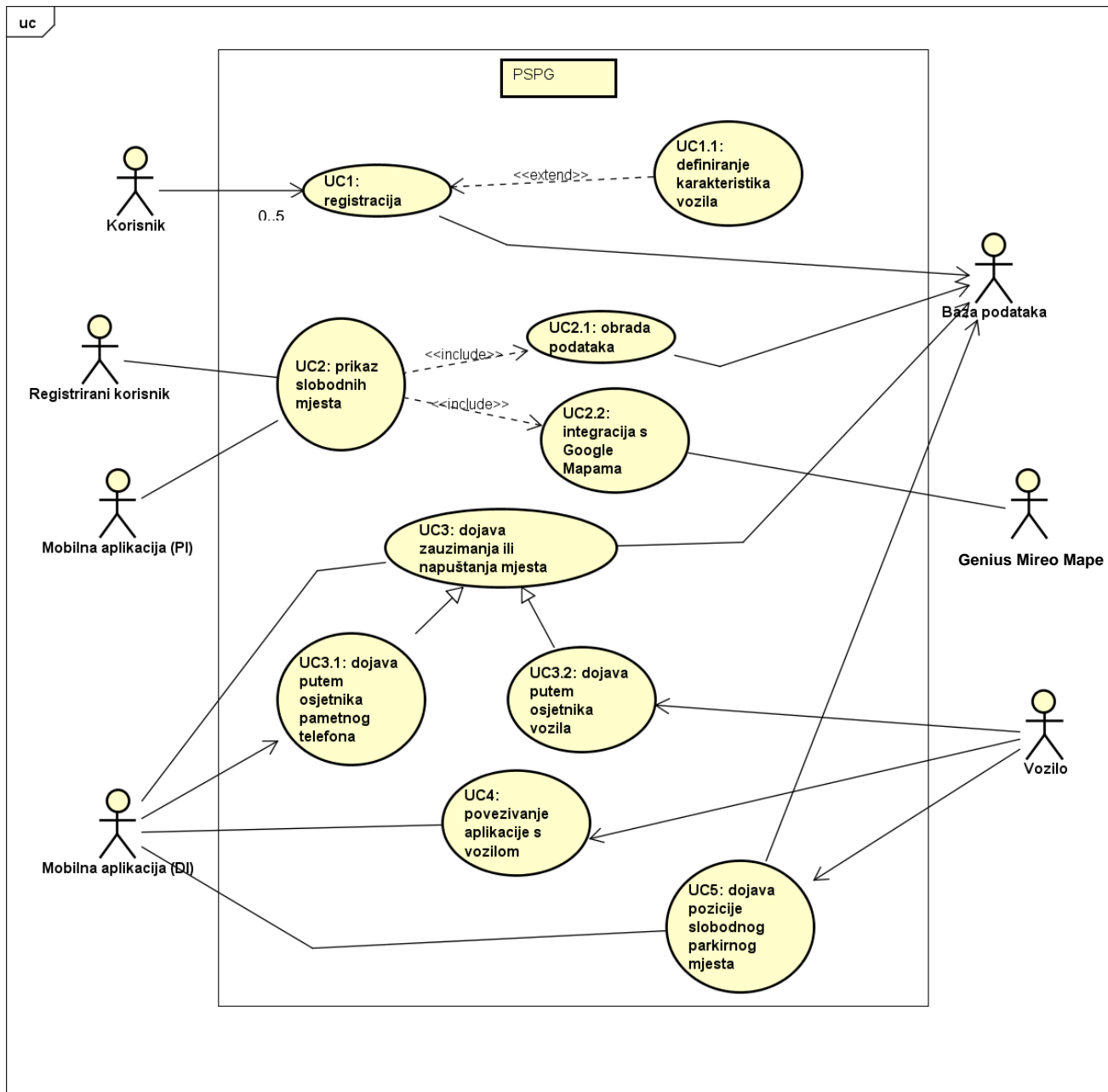
Mobilna aplikacija na pametnom telefonu je primatelj informacija (PI) o:

Oblikovanje programske potpore

- Registrirani korisnici prikaz slobodnih parkirnih mjesta mogu pratiti putem mobilne aplikacije na pametnom telefonu. Prikaz podataka uključuje obradu podataka iz baze podataka i integraciju s Genius Mireo mapama.

14. (4 boda) Dijagram obrazaca uporabe

Rješenje:



Bodovanje:

- 1 bod za ispravno uočavanje svih aktora
- 0.5 boda za ispravno određivanje je li aktor aktivan ili pasivan
- 1 bod za ispravno uočavanje svih obrazaca uporabe (moguće su manje varijacije u prikazanim obrascima)
- 1 bod za ispravno određene veze generalizacije, include i extend
- 0.5 boda za ispravan prikaz granica sustava i imenovanja sustava

Postupak:

1. Uočiti aktore kao imenice koje su povezane s nekom akcijom u PSPG sustavu:

- korisnik,
- registrirani korisnik,
- mobilna aplikacija (DI),
- mobilna aplikacija (PI),
- moderno vozilo (**moguće i rješenje – univerzalno sučelje vozila*),
- baza podataka,
- Genius Maps.

Imenice koje se mogu modelirati kao aktori, ali i ne moraju:

- moguće uvođenje aktora *osjetnici pametnog telefona* i *parkirni osjetnici vozila*

2. Uočiti glagole ili glagolske imenice koje opisuju specifične akcije aktora:

- registracija adresom elektroničke pošte
- definiranje karakteristika vozila
- povezivanje aplikacije s vozilom
- prikupljanje informacija o poziciji parkirnih mjesta
- dobavljanje informacije o poziciji slobodnog parkirnog mjesta
- dobavljanje informacija o zauzimanju ili napuštanju parkirnog mjesta
- pohrana informacija
- primanje informacija o poziciji slobodnih parkirnih mjesta
- prikaz karte (slobodnih mjesta)
- obrada podataka iz baze
- integracija s Genius mapama

Temeljem toga utvrđeni su obrasci uporabe (engl. *use case*) koji opisuju sekvence akcija (uključujući varijante) koje sustav ili drugi entitet obavlja u interakciji s aktorima:

- *Registracija*,
 - Koja nudi mogućnost *definiranja karakteristika vozila* ako je to vozilo moderno (*extend*) za najviše 5 vozila.
- *Prikaz slobodnih mjesta* putem mobilne aplikacije pri čemu se treba
 - *Obraditi podatke* o slobodnim mjestima iz baze (*include*),
 - *Integrirati podatke* s Genius mapama
- *Povezivanje mobilne aplikacije s vozilom*
- *Dojava pozicije slobodnog parkirnog mjesta*
- *Dojava zauzimanja ili napuštanja parkirnog mjesta*
 - Pri čemu su moguće dvije varijante *dojava putem osjetnika vozila* i *dojava putem osjetnika pametnog telefona* (specijalizacija)

Neke od akcija mogu se, ali i ne moraju prikazati kao zaseban obrazac uporabe:

- Pohrana informacija u bazu pri dojavi pozicije slobodnog mjesta
- Pohrana informacija u bazu pri dojavi zauzimanja ili napuštanja mjesta pozicije parkirnog mjesta

3. Povezivanje aktora s obrascima uporabe i određivanje aktivnih i pasivnih aktora.

Pasivni aktori su: baza podataka i Genius mape ,

Aktivni aktori: svi ostali.

- ukoliko su navedeni „osjetnici pametnog telefona“ kao aktor potrebno ih je povezati s obrascem uporabe „Dojava zauzimanja ili napuštanja parkirnog mjesta putem osjetnika pametnog telefona“
- ukoliko su navedeni „parkirni osjetnici vozila“ kao aktor potrebno ih je povezati s obrascima uporabe „Dojava pozicije slobodnog parkirnog mjesta “ i „Dojava zauzimanja ili napuštanja parkirnog mjesta putem osjetnika vozila“

4. Prikazati granice sustava i imenovati sustav.

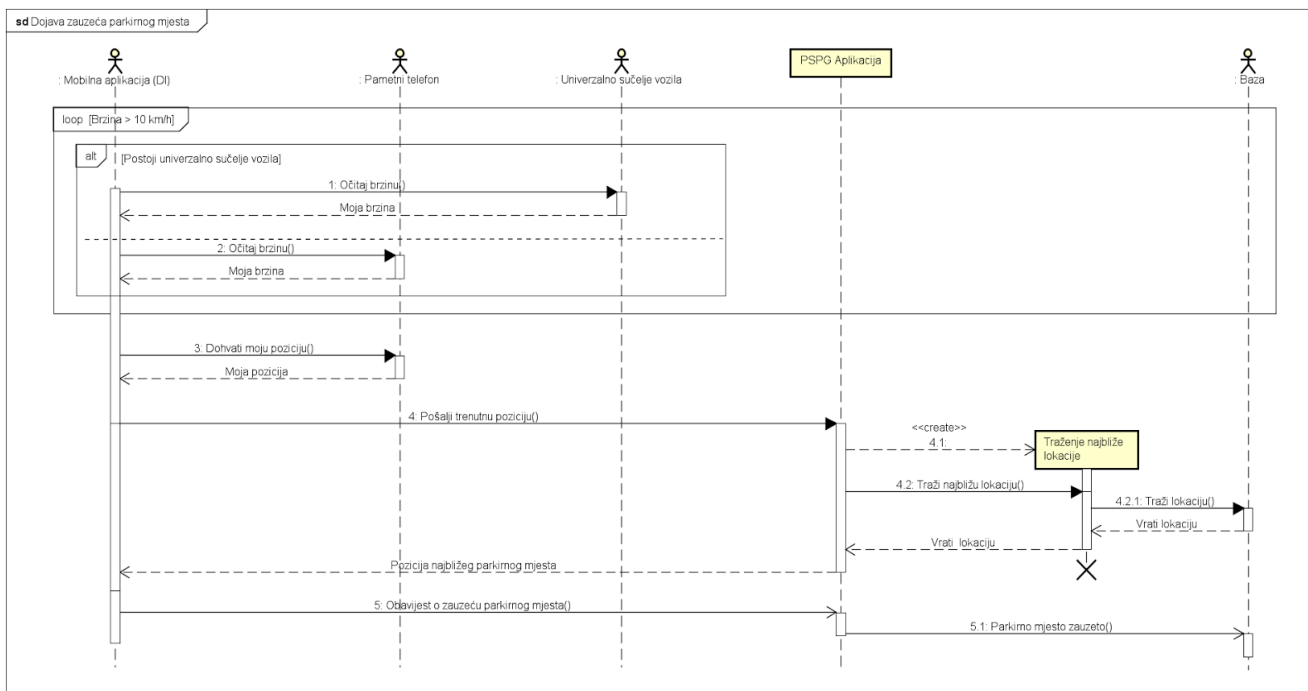
15. (4 boda) Sekvencijski dijagram

Odgovarajućim sekvencijskim dijagramom (engl. *sequential diagram*) modelirajte proces dojava zauzeća parkirnog mjesta.

Prilikom izrade dijagrama uz opći opis uzmite u obzir i sljedeće zahtjeve:

- Proces dojava zauzeća pokreće se tek pri brzinama vozila manjim od 10 km/h.
- Mobilna aplikacija traži provjeru pozicije najbližeg parkirnog mjesta prema GPS lokaciji očitanoj od pametnog telefona. PSPG aplikacija pronalazi lokacije slobodnih mjesta prema primljenoj lokaciji pokretanjem novog procesa pretraživanja informacija iz baze podataka.
- Nakon toga, interni algoritam mobilne aplikacije dojavljuje status zauzeća parkirnog mjesta PSPG-u koji taj podatak pohranjuje u bazu.

Rješenje:



Bodovanje:

- 1 bod za određivanje svih bitnih objekata
- 1 bod za temeljni slijed akcija
 - određivanje pobuda, aktivnosti i formiranje poruka
- Bilo koji detalj koji bitno narušava notaciju i semantiku dijagrama – po -0.5 boda

*priznaje se i rješenje gdje umjesto stvaranja objekta „Traženje najbliže lokacije“ postoji sinkrona poruka za traženje lokacije između PSPG Aplikacije i Baze.

Postupak:

Napomena: postupka prijave zauzeća parkirnog mjesta je potpuno automatska (samostalna) aktivnost koju ne pokreće niti u njoj ne sudjeluje vozač automobila, niti korisnik aplikacije (čovjek)!

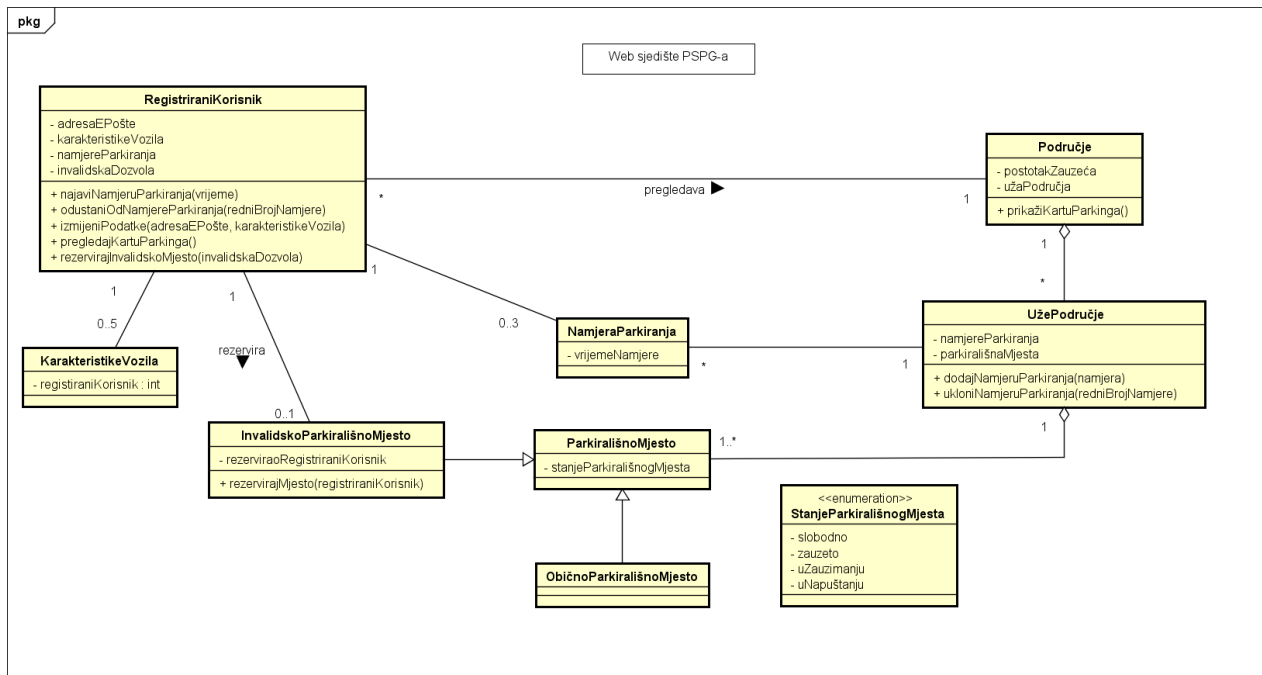
1. Prema aktorima vezanim uz obrasce upotrebe, odrediti elemente:
 - a. Mobilna aplikacija
 - b. Pametni telefon
 - c. Univerzalno sučelje vozila
 - d. PSPG aplikacija
 - e. Baza podataka
2. Odrediti preduvjete za pokretanje postupka prijave zauzeća parkirnog mjesta, a to je brzina manja od 10 km/h
 - a. Ukoliko se radi o "pametnom vozilu" koje ima osjetnike brzine, brzina se očitava s univerzalnog sučelja vozila
 - b. Ukoliko vozilo nema svoje osjetnike brzine, onda se brzina očitava s pametnog telefona, pomoću osjetnika na samom pametnom telefonu (GPS, akcelerometar)
3. Nakon pokretanja procesa, aplikacija prvo utvrđuje vlastitu poziciju, i to pomoću GPS osjetnika za poziciju pametnog telefona
4. Mobilna aplikacija dojavljuje u PSPG sustav trenutnu poziciju. Prema toj trenutnoj poziciji, PSPG sustav pokreće proces pretraživanja baze i traži najbližu lokaciju slobodnog parkirnog mjesta. Moguće je i da PSPG sustav samostalno pretražuje bazu podataka za najbližu lokaciju.

PSPG vraća mobilnoj aplikaciji informaciju o najbližem parkirnom mjestu.
5. Interni algoritam mobilne aplikacije utvrđuje podudarnost trenutne pozicije i pozicije postojećeg slobodnog parkirnog mjesta, i nakon što utvrdi podudarnost ta dva podatka šalje informaciju PSPG sustavu da je parkirno mjesto zauzeto. PSPG sustav taj podatak upisuje u bazu.

16. (4 boda) Dijagram razreda

Odgovarajućim UML-dijagramom razreda modelirajte web sjedište PSPG-a. Navesti attribute i operacije razreda (s vidljivostima) i višestrukosti na vezama. Nije potrebno modelirati tipove podataka.

Rješenje:



powered by Astah

Bodovanje:

- 1 bod za prepoznavanje razreda i veza nasljeđivanja.
- 1 bod za ispravne asocijacije između tih razreda zajedno s višestrukostima (nazivi asocijacija nisu bitni)
- 0.5 boda za ispravno navođenje metoda razreda (ne moraju biti identične ovima iz rješenja, ali moraju pokriti sve funkcionalnosti web sjedišta)
- 0.5 boda za ispravno navođenje atributa razreda
- 0.5 boda za enumeraciju
- 0.5 boda za imenovanje sustava i vidljivosti

Postupak:

1. Identificiranje razreda

- Uočiti bitne imenice iz teksta zadnjeg paragrafa zadatka (koji počinje s: „Web sjedište PSPG-a prikazuje kartu parkinga...“) te ih modelirajte kao razrede.
- Pritom je potrebno zamijetiti da se NE modeliraju oni razredi koji nisu dio web sjedišta (npr. osjetnici, mobilna aplikacija, baza podataka i slično) te da se NE modeliraju općeniti razredi kao što su PSPG sustav ili web sjedište PSPG-a.
- Stanje parkirnog mjesta je enumeracija koju je potrebno prikazati kao zaseban razred.

2.Određivanje odnosa među razredima

- Uočiti da je „najava namjere parkiranja“ pridruženi razred između registriranog korisnika i užeg područja.
- Uočiti generalizaciju između tipa parkirališnog mjesta (obično ili invalidsko) i samog razreda parkirališnog mjesta (koji može, ali i ne mora biti apstraktan) .
- Uočiti asocijacije tipa agregacije između područja i užeg područja, između užeg područja i parkirališnog mjesta te asocijacije između registriranog korisnika i invalidskog mjesta (za rezervaciju) i između registriranog korisnika i područja (za pregledavanje karte).

3.Određivanje višestrukosti pridruživanja

- Svugdje gdje ima više od jednog objekta na nekoj strani pridruživanja potrebno je to označiti s odgovarajućim brojem primjeraka objekata ili oznakom *.

4.Definiranje operacija i atributa za pojedine razrede

- Funkcionalnosti pojedinih razreda koje su navedene u tekstu zadnjeg paragrafa ili osmišljene kao nužne da bi razredi bili povezani potrebno je modelirati kao metode razreda.
- Svojstva pojedinih razreda koje su navedena u tekstu zadnjeg paragrafa ili su izvedena kao nužna da bi razredi bili međusobno povezani vezama pridruživanja potrebno je modelirati kao attribute razreda.
- Načelno, sve metode su javne, a svi atributi privatni, no moguće su manje varijacije po tom pitanju (npr. stanje parkirališnog mjesta je zaštićeni atribut).
- Privatnim atributima pristupa se preko get i set metoda, što nije potrebno prikazivati, kao niti konstruktore
- Neki atributi koji su značajni za provođenje nekih metoda, a ne spominju se u zadnjem paragrafu zadatka trebaju se uključiti u odgovarajuće razrede radi kompletnosti (npr. adresaEPošte, karakteristikeVozila).

5.Imenovanje veza i dijagrama

- Veze pridruživanja mogu se imenovati radi boljeg razumijevanja povezanosti razreda.
- Dijagram je potrebno imenovati (npr. „Web sjedište PSPG-a“).