



SE322 - INŽENJERSTVO ZAHTEVA

Primena slučajeva korišćenja

Lekcija 06

PRIRUČNIK ZA STUDENTE

SE322 - INŽENJERSTVO ZAHTEVA

Lekcija 06

PRIMENA SLUČAJEVA KORIŠĆENJA

- ✓ Primena slučajeva korišćenja
- ✓ Poglavlje 1: Slučajevi korišćenja i priče korisnika
- ✓ Poglavlje 2: Slučajevi korišćenja
- ✓ Poglavlje 3: Utvrđivanje slučajeva korišćenja i zahteva u CTS
- ✓ Poglavlje 4: Vežba
- ✓ Poglavlje 5: Domaći zadatak
- ✓ Poglavlje 6: Projektni zadatak
- ✓ Zaključak

Copyright © 2017 – UNIVERZITET METROPOLITAN, Beograd. Sva prava zadržana. Bez prethodne pismene dozvole od strane Univerziteta METROPOLITAN zabranjena je reprodukcija, transfer, distribucija ili memorisanje nekog dela ili čitavih sadržaja ovog dokumenta., kopiranjem, snimanjem, elektronskim putem, skeniranjem ili na bilo koji drugi način.

Copyright © 2017 BELGRADE METROPOLITAN UNIVERSITY. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, without the prior written permission of Belgrade Metropolitan University.

▼ Uvod

UVOD

Uvodne napomene

U ovoj lekciji detaljnije radimo slučajeve korišćenja koji su osnov za razumevanje potreba korisnika, za utvrđivanje njegovih zahteva, za definisanje scenarija, tj. tokova akcija koje se moraju izvršiti u nekom slučaju korišćenja, za utvrđivanje funkcionalnih zahteva i definisanja testova prihvatanja sistema. Ako se projektuju i razvijaju objektno-orijentisani (OO) sistemi, onda se lako mogu iz slučajeve korišćenja da definišu model objekata i sekvencijalni dijagrami, iz kojih se daljim detaljisanjem razvijaju klase, sa svojim atributima i metodima.

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

▼ Poglavlje 1

Slučajevi korišćenja i priče korisnika

VIDEO PREDAVANJE ZA OBJEKAT "SLUČAJEVI KORIŠĆENJA I PRIČE KORISNIKA"

Trajanje video snimka: 19min 29sek

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

OBLASTI PRIMENE SLUČAJEVA KORIŠĆENJA I PRIČA KORISNIKA

Slučajevi korišćenja i priče korišćenja služe da se utvrde šta korisnici žele da sistem radi. Korisne su za poslovne aplikacije, razvoj veb sajtova, i interakciju

Preduslov za projektovanje softvera koji zadovoljava potrebe svojih korisnika je da se razume šta korisnik namerava da radi sa softverom. Neko razvija softver tako da definiše sam šta sve softver može da radi nadajući se da je to potrebno korisnicima ("**product-centric approach**"). Drugi polaze od utvrđenih potreba korisnika i na osnovu toga specificiraju zahteve korisnika i prema tome razviju softver ("**user-centric approach**"). Usmerenje ka korisniku i njegovom planu korišćenja softvera pomaže otkrivanju potrebne funkcionalnosti softvera, izostavljanju svojstava softvera koje se neće koristiti, i olakšava određivanje prioriteta u razvoju softvera.

Zahtevi korisnika, kao što je poznato, se nalaze između poslovnih ciljeva projekta i funkcionalnih zahteva koji opisuju šta inženjer razvoja treba da projektuje i izvrši. U ovoj lekciji izložićemo dve najčešće korišćene metode za istraživanje zahteva korisnika: slučajeve korisnika i priče korisnika.

Priče korisnika su počele da se koriste sa primenom agilnih metoda razvoja softvera. One na koncizan način opisuju šta je potreba korisnika i služi za početak razgovora radi utvrđivanja detalja.

Obe metode služe da se utvrdi šta korisnici žele da sistema radi. Opisuju šta korisnik treba da radi sa sistemom, opisuju njihovu interakciju. Svaki slučaj korisnika opisuje više scenarija korišćenja (jedan obezbeđuje potrebnu funkcionalnost, a ostali opisuju razne posebne situacije i izuzetke).

Biznis analitičar treba da na bazi slučajeva korišćenja i njihovih scenarija definiše funkcionalne zahteve i da testira da li sistem, na osnovu njih, izvršava opisane scenarije.

Slučajevi korišćenja i priče korisnika su vrlo korisne za istraživanje zahteva poslovnih aplikacija, veb-sajtova, kioska, i sistema u kojima korisnik kontroliše aplikaciju. Međutim, ove dve metode nisu zgodne za određene aplikacije, kao što je paketna obrada, računarski intenzivne aplikacije, biznis analitiku, a skladištenje podataka (**data warehousing**) može da ima samo nekoliko slučajeva korišćenja.

Pored ovih računarski intenzivnih aplikacija, postoje i druge aplikacije u kojima slučajevi korišćenja i priče korisnika nisu odgovarajuće. To su sistemi sa ugrađenim softverom i drugi sistemi u realnom vremenu. Zahtevi za ovakve sisteme su liste koje povezuju događaje i odgovarajuće odgovore sistema.

UPOREĐENJE SLUČAJEVA KORIŠĆENJA I PRIČA KORISNIKA

Slučaj korišćenja opisuje sekvencu interakcija između sistema i spoljnih aktera. Priča korisnika (user story) je kratak opis svojstva ispričan iz perspektive osobe koja želi novu sposobnost

Slučaj korišćenja ili upotrebe (**use case**) opisuje sekvencu interakcija između sistema i spoljnih aktera koja dovodi akterima ostvarenje rezultata od značaja za njih. Nazivi slučajeva korišćenja se uvek pišu u formatu "glagol + objekat". Birajte snažne opisne nazive koje pokazuju vrednost koju obezbeđuju nekom korisniku. Na slici 1 prikazani su nazivi pojedinih slučajeva korišćenja za slučaj nekoliko aplikacija.

Aplikacija	Neki slučajevi korišćenja
Chemical Tracking System (Sistem za praćenje hemikalija)	Zahtevaj himikaliju Štampaj list sa podatkima o bezbednom materijalu Promeni zahtev za hemikalijom Promeni status narudbenice Generiše kvartalni izveštaj o upotrebi hemikalije
Sistem prijavljivanja na aerodromu	Prijava za let Štampaj kartu za ukrcajanje Promeni sedište Proveri prtljag Kupi nadogradnju
Računovodstveni sistem	Kreiraj fakturu Uskladi izvod iz računa Unesi transakciju sa kreditnom karticom Odštampaj poreski formular prodavca Traži specifičnu transakciju
Onlajn prodavnica	Dopuni profil kupca Traži jednu stavku Kupi jednu stavku Prati paket sa isporukom Poništi neposlat nalog

Izvor: Karl Wiegers, Joy Beatty, Software Requirements, 3rd ed., Microsoft, 2013

Slika 1.1 Neki slučajevi upotrebe ili korišćenja nekoliko aplikacija

Priča korisnika (**user story**) je kratak opis svojstva ispričan iz perspektive osobe koja želi novu sposobnost, koja je obični ili korisnik ili kupac. Priča korisnika se obično piše u sledećem formatu:

Kao <tip korisnika> želim <neki cilj> tako da <neki razlog>

Upotreba ovog formata obezbeđuje prednost u odnosu i na kraći naziv slučaja korišćenja zbog toga što priča korisnika, pored cilja korisnika, daje informaciju i o klasi korisnika i o razlozima zahteva. Klasa korisnika odgovara primarno akteru u slučaju korišćenja. Razlog se mora navesti u kratkom opisu slučaja korišćenja. Na slici 2 se prikazuje se pojedini slučajevi korišćenja sa slike 1 mogu da predstave kratkim pričama

Aplikacija	Slučaj korišćenja	Priča korisnika
Chemical Tracking System	Zahtevaj himikaliju	Kao hemičar, ćelim da yahtevam hemikaliju da bih izvršio eksperiment
Sistem prijavljivanja na aerodromu	Prijava za let	Kao putnik, ćelim da seprijavim ya let kako bi leteo u moju destinaciju
Računovodstveni sistem	Kreiraj fakturu	Kao mali biznismen, ćelim da kreiram fakturu, kako bi ispostavio račun kupcu.
Onlajn prodavnica	Dopuni profil kupca	Kao kupac, ćelim da dopunim moj profil kupca, tako da buduće kupovine idu na račun kartice sa novim brojem

Izvor: Karl Wiegers, Joy Beaty, Software Requirements, 3rd ed., Microsoft, 2013

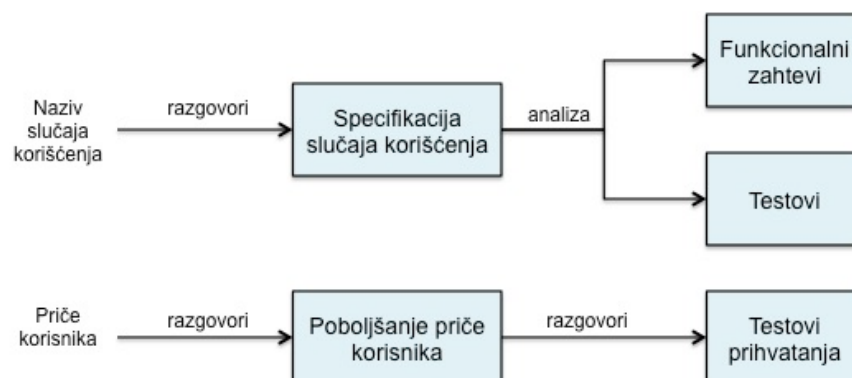
Slika 1.2 Slučajevi korišćenja (upotrebe) i odgovarajuće priče korisnika

RAZLIKE IZMEĐU SLUČAJEVA KORIŠĆENJA I PRIČA KORISNIKA.

Na osnovu specifikacije slučaja korišćenja, analitičar onda dobija funkcionalne zahteve. Priče korisnika se detaljišu i dele na manje, i vode dob

Na prvi pogled, slučajevi korišćenja i priče korisnika izgledaju slično. Obe metode pokušavaju da opišu šta različiti tipovi korisnika žele da urade interakcijama sa sistemom. Međutim, njihovi procesi nadalje idu različitim pravcima, kao što se vidi na slici 3. pristupa proizvode rezultate, kao što su vizuelni modeli analize.

U slučaju primene slučajeva korišćenja, analitičar mora u razgovoru sa korisnikom da razume kako on zamišlja dijalog kupca i sistema pri izvršenju slučaja korišćenja. Analitičar daje strukturu dobijenim informacijama u skladu sa formularom za definisanje slučaja korišćenja. On sadrži prostor za upisivanje razumevanja slučaja korišćenja.



Izvor: Karl Wiegers, Joy Beaty, Software Requirements, 3rd ed., Microsoft, 2013

Slika 1.3 Kako zahtevi korisnika vode do funkcionalnih zahteva i testova, preko slučajeva korišćenja i priča korisnika

Slučaj korišćenja pomaže analitičaru i korisniku da otkriju sve neophodne informacije vezane za slučaj korišćenja. Na osnovu specifikacije slučaja korišćenja, analitičar onda dobija funkcionalne zahteve koje inženjer razvoja treba da primeni, a tester treba da pripremi testove radi provere da li je slučaj korišćenja pravilno primenjen. Inženjer razvoja može da primeni u celini slučaj korišćenja i jednom izdanju ili iteraciji, ili da samo deo slučaja korišćenja, a ostale delova - u narednim izdanjima ili iteracijama.

Kod agilnih projekata, priča korisnika služi za unos budućih razgovora koji će se voditi u budućnosti, kada to zatreba, između inženjera razvoja, predstavnika kupca, i biznis analitičara. Ovi razgovori otkrivaju dodatne informacije koje inženjeri razvoja moraju da znaju da bi primenili priču. Na taj način oni poboljšavaju priče korisnika, što dovodi do kolekcije manjih pojedinačnih komada nove funkcionalnosti. Male priče korisnika su veličine koje se mogu realizovati u jednoj iteraciji.

I JOŠ NEŠTO O RAZLIKAMA

Priče korisnika nude prednost u konciznosti i jednostavnosti, a slučajevi korišćenja daju učesnicima strukturu i kontekst, što pričama korisnika nedost

Umesto funkcionalnih zahteva, timovi agilnih projekta sa detaljnijim pričama korisnika, dobijaju "zadovoljavajuće uslove", koji su osnov za definisanje testova prihvatanja. Razmišljanje o testovima pomaže vam da utvrdite promene osnovne priče korisnika (ili slučaja korišćenja), uslove izuzetaka, i nefunkcionalne zahteve, kao što su performanse i mere bezbednosti. Ako programer primeni potreban kod da bi zadovoljio testove prihvatanja, time će zadovoljiti i uslove zadovoljenja, a to onda znači da je priča korisnika korektno primenjena (implementirana).

Priče korisnika daju koncizan iskaz o potrebama korisnika. Slučajevi korišćenja idu dublje u opisivanje kako korisnik zamišlja interakciju sa sistemom da bi zadovoljio svoje ciljeve. Slučajevi korišćenja ne bi trebalo da ulaze u aspekte projektovanja. Priče korisnika nude prednost u konciznosti i jednostavnosti, a slučajevi korišćenja daju učesnicima strukturu i kontekst, što pričama korisnika nedostaje. Slučajevi korišćenja omogućavaju analitičaru da na organizovani način vodi izazivanje zahteva, što je više nego samo prikupljanje onoga što korisnici žele i šta im je potrebno, kao kod priča korisnika.

Postoje mišljenja da priče korisnika ne nude rešenje za slučaj velikih i zahtevnijih projekata. One ne nude strukturu informacija o potrebama korisnika i može se desiti da se neki test prihvatanje ne definiše. Zato, analitičar treba da ima iskustva sa pričama korisnika da ne bi isпустиu neku funkcionalnost.

▼ Poglavlje 2

Slučajevi korišćenja

VIDEO PREDAVANJE ZA OBJEKAT "SLUČAJEVI KORIŠĆENJA"

Trajanje video snimka: 32min 9sek

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

PRIMER ODREĐIVANJA AKTERA U CHEMICAL TRACKING SYSTEM

Slučajevi korišćenja obezbeđuju apstraktnu vizualno predstavljanje zahteva korisnika.

Slučaj korišćenja opisuje redosled interakcija između sistema i spoljnog aktera dovodi do rezultata koji obezbeđuje akteru određenu vrednost. Slučajevi korišćenja obezbeđuju apstraktno vizualno predstavljanje zahteva korisnika.

Akter je osoba (ali može biti i drugi sistem, ili uređaj) koji je u interakciji sa sistemom i izvršava slučaj korišćenja. U slučaju koji razmatramo na predmetu, projekta Chemical Tracking System (CTS), ulogu aktera obavlja **Podnosilac zahteva** (**Requestor**) koji je predstavnik klase korisnika **Requestor**. Članovi ove klase mogu biti i hemičari i zaposleni u skladištu hemikalija, jer i jedni i drugi mogu da zahtevaju isporuku hemikalija.

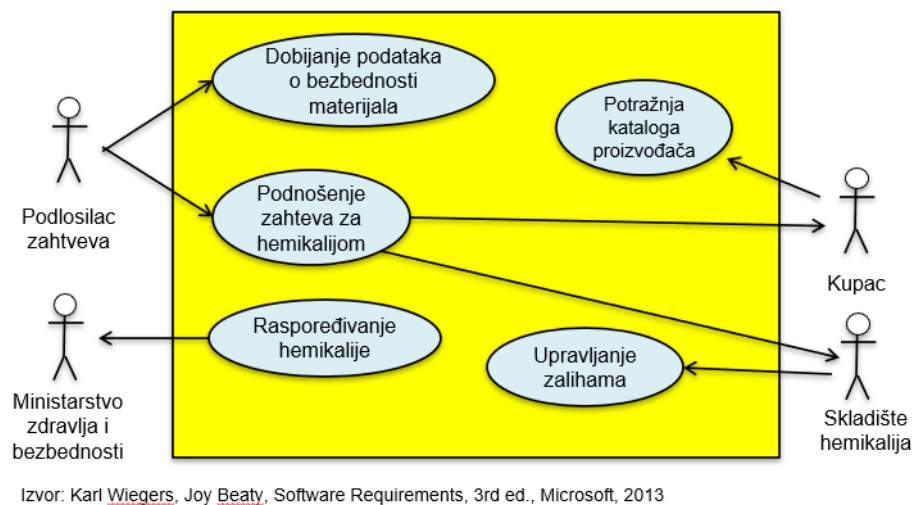
Da bi lakše utvrdili ko je akter sistema, možete predstavniku korisnika sistema postaviti sledeća pitanja:

- Ko (ili šta) se obaveštava kada se nešto desi u sistemu?
- Ko (ili šta) obezbeđuje informaciju ili servise sistemu?
- Ko (ili šta) pomaže da sistem odgovori i da završi zadatak?

Strelica od aktera ka sistemu ukazuje na primarnog aktera. Strelica koja od slučaja korišćenja vodi ka akteru, pokazuje ko je sekundarni akter, koji ne inicira interakciju sa slučajem korišćenja (kao primarni akter), ali u njoj učestvuje. Ostali softverski sistema obično imaju ulogu sekundarnog aktera, jer pomažu izvršenju slučaja korišćenja (na primer, baza za obuku)

Sistem se aktivira kada Podnosilac zahteva opasnu hemikaliju koja zahteva da je podnosilac zahteva obučan da je koristi. Strelice u dijagramu korišćenja pokazuju interakcije između

aktera i slučajeva korišćenja. Za razliku od dijagrama konteksta, one ne predstavljaju tok podataka, kontrolnih signala ili materijala.



Slika 2.1 Deo slučaja korišćenja projekta Chemical Tracking System

SLUČAJEVI KORIŠĆENJA I SCENRIJI KORIŠĆENJA

Scenario je opis jednog primera upotrebe sistema. Jedan slučaj korišćenja ima kolekciju scenarija.

Slučaj korišćenja predstavlja jednu diskretnu, samostalnu aktivnost koju jedan akter izvršava da bi dobio rezultat koji predstavlja dodatnu vrednost. Sam slučaj korišćenja (aktiviran jednom akcijom aktera), u sebi sadrži niz aktivnosti koji imaju isti cilj. **Scenario je opis jednog primera upotrebe sistema. Jedan slučaj korišćenja je predstavlja kolekciju scenarija upotrebe sistema, a scenario je jedan primer (instanca) slučaja korišćenja.** Kada analizirate zahteve korisnika, možete početi od nekog uopštenog slučaja korišćenja, pa da razvijete više specijalizovane slučajeve korišćenja. Možete da radite i obrnuto, možete generalizovati prethodne razvijene specijalizovane scenarije.

Slika 2 prikazuje primer jednog slučaj korišćenja, "Zahtev hemikalije", u projektu Chemical Tracking sistem koji radimo u svim lekcijama ovog predmeta. Ne mora uvek da se popuni ceo formular. To zavisi od specifičnosti slučaja korišćenja. Ako neka informacija iz formulara negde postoji u dokumentaciji, dovoljno je sam dati referencu na tu informaciju

Bitni elementi jednog slučaja korišćenja su:

- jedinstveni identifikacioni broj slučaja korišćenja i njegov naziv.
- Kratki tekstualni opis svrhe slučaja korišćenja
- uslov početka koji pokreće izvršenje slučaja korišćenja
- nula ili više preduslova za početak slučaja korišćenja
- jedan ili više postuslova koji opisuju stanje sistema posle uspešnog završetka slučaja korišćenja.

UC-4 Zahtev himikalije		
Miloš Ilić	Datum Ireiranja	22.3.2013
Podnosilac zahteva	Sekundarni akter:	Kupac, Sladiše hemikalija, Baza za obuku
Podnosilac zahteva specificira željene hemikaliju u zahtev unošenjem njenog imena ili ID broja ili unošenjem njene strukture iz alata za crtanje hemijske strukture. Sistem ili nudi podnosiocu zahteva kontejner sa hemikalijom iz skladišta hemikalija ili dozvoljava zahtevaocu hemikalija da naruči kontejner od prodavca.		
Podnosioc zahteva naynačuje da želi da yahteva hemikaliju.		
PRE-1. Identitet korisnika je utvrđen.		
PRE-2. Korisnik je ovlašćen da yahteva henikaliju.		
PR-3 Baza hemikalija je onlajn.		
POST-1. Zahtev je uskladišćen in CST.		
POST-2. Zahtev je poslat u Sladište hemikalija ili Kupcu		
4.0 Zahtev za hemikalijom iz Skladišta hemikalija.		
1. Podnosilac zahteva specificira željenu hemikaliju.		
2. Sistem izlistava kontejnere sa željenom hemikalijom		
3. Sistem daje Podnosiocu zahteva opciju na Pogled na istoriju kontejnera za bilo koji kontejner.		
4. Podnosilac zahteva bira određeni kontejner ili pita da postavi zahtev isporučiocu hemikalija (vidi 4.1).		
5. Podnosilac zahteva unosi druge informacije da bi kompletirao zahtev.		
6. Sistem skladišti zahtev i obaveštava Stovarište hemikalija.		
4.1 Zahtev za hemikalijom od Isporučiova		
1. Podnosilac zahteva traži katalog isporučiova za henikaliju (vidi 4.1E)		
2. Sistem prikazuje listu prodavaca hemikalije sa raspoloživim veličinama kontejnera, kvalitetom i cenama.		
3. Podnosilac zahteva bira isporučioa hemikalije, veličinu kontejnera, njegov kvalitet i broj kontejnera.		
4. Podnosilac yahteva unosi druge informacije radi kompletiranja yahteva.		
5. Sistem skladišti zahtev i obaveštava Kupca.		
4.1.E1 Hemikalija nije komercijalno na raspolaganju		
1. Sistem prikazuje poruku «nema isporučioa za tu hemikaliju».		
2. Sistem pita Podnosioca yahteva da li želi podnese yahtev ya neju drugu hemikaliju (3a) ili da iyače iy sistema (4a).		
3a. Podnosilac yahteva traži d apodnese yahtev ya drugu hemikaliju		
3b. Sistem počinje sa novim normalnim tokom.		
4a. Podnosilac zahteva traži izlaz.		
4b. Sistem završaba slučaj korišćenja.		
Visoka		
Otrprike 5 puta nedeljno po hemičaru, 200 puta nedeljno od zaposlenih u Sladištu hemikalija		
BR-28, BR-31		
Sistem mora da omogući unos hemijske strukture u standarnom obliku od bilo koga podržanog paketa sa crtežima hemijskih struktura.		
Pretpostavlja se da su proverene unete strukture hemikalije		

Izvor: Karl Wieggers, Joy Beaty, Software Requirements, 3rd ed., Microsoft, 2013

Slika 2.2 Delimična specifikacija slučaja korišćenja "Zahtev za hemikaliju" za CTS

PREDUSLOVI I POSTUSLOVI

Preduslovi definišu šta mora da bude ispunjeno da bi sistem počeo da izvršava slučaj korišćenja. Postuslovi opisuju stanje sistema posle uspešnog izvršenja slučaja korišćenja.

Preduslovi (**preconditions**) definišu šta mora da bude ispunjeno da bi sistem počeo da izvršava slučaj korišćenja. Sistem mora da ima testove za utvrđivanje ispunjenosti preduslova. Preduslovi mogu da opišu stanja sistema ("ATM mora da u sebi ima novac, da bi mogao da pokrenem slučaj povlačenje novca iz automata"), ali ne opisuju nameru korisnika ("korisniku je potreban novac").

Kada sistem dobije događaj koji ga pokreće na izvršenje nekog uslova korišćenja, on onda ide na proveru preduslova, i ako su zadovoljeni, počinje sa izvršavanjem slučaja korišćenja. Pokretač nije preduslov.

Postuslovi (**postconditions**) opisuju stanje sistema posle uspešnog izvršenja slučaja korišćenja. Postuslovi mogu da obuhvate:

- nešto što korisnik može da opazi (na primer, balas računa u banci).
- fizički rezultat (ATM je isporučio novac i izveštaj),

- promenjeno je unutrašnje stanje sistema (odbijena sa bilnas stanja računa povećena količina novca i taksa za transakciju).

Svi izlazi iz slučaja korišćenja bi trebalo da obezbede određenu vrednost korisniku, te su svi oni opazljivi korisniku. Ono što je neophodno da bude postuslov, a što korisnik ne mora da opazi, to mora biznis analitičar da stavi u postuslove, ako je to uslov da sistem radi ispravo,

Koristite konvenciju o označavanju malih i malih slova

Specifikacije slučajeva upotrebe sastoje se od brojnih malih paketa informacija: normalni i alternativni tokovi, izuzeci, preduslovi i postuslovi, i tako dalje. Primer na slici2 pokazuje jednostavnu konvenciju o označavanju koja vam može pomoći da ovi elementi budu ravni. Svaki slučaj upotrebe ima redni broj i smisleno ime koje odražava korisnikov cilj: UC-4 Zahtevaj hemikaliju. Identifikator normalnog protoka za ovaj slučaj upotrebe je 4.0. Alternativni tokovi se identifikuju povećanjem broja desno od decimalnog značaja, tako da je prvi alternativni tok 4.1, drugi bi bio 4.2, i tako dalje. I normalan i alternativni protok mogu imati svoje izuzetke. Prvi izuzetak za slučaj normalnog protoka upotrebe broj 4 bio bi označen sa 4.0.E1. Drugi izuzetak za prvi alternativni tok za ovaj slučaj upotrebe bio bi 4.1.E2.

NORMALNI I ALTERNATIVNI TOKOVI (SCENARIJI) I IZUZECI

Jedan scenario predstavlja jedan normalan tok događaja za slučaj korišćenja. Može imati i alternativne tokove (scenarije).

Jedan scenario predstavlja jedan normalan tok događaja za slučaj korišćenja. On se takođe naziva i glavnim tokom događaja, ili osnovnim tokom, ili primarnim scenarijem, glavnim uspešnim scenarijem, i dr. Na slici 2 je bio opisan glavni tok događaja, tj. primarni scenario za slučaj korišćenja "Zahtev za hemikaliju". Kao što se vidi, dat je redosled koraka koje aktor i sistem treba da izvrše. Za svaki korak se navodi ko izvršava taj korak.

Drugi uspešni tokovi su alternativni tokovi ili sekundarni scenariji. Alternativni tokovi isporučuju rezultat kao i normalni tok, ali predstavljaju manje uobičajen rezultat ili neki specifičan rezultat. Normalni tok može imati grananje u neki alternativni tok u nekoj tački odlučivanja u sekvenciji dijaloga. Kasnije se može spojiti sa glavnim tokom.

Priče korisnika najčešće pokrivaju jedan scenario ili alternarivni tok.

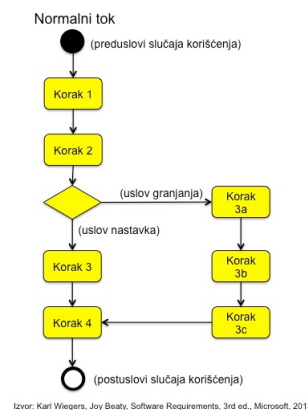
Uslovi koji mogu da spreče uspeh slučaja korisnika su izuzeci. Izuzeci (exceptions) opisuju uslove pri kojima se javljaju greške prilikom izvršenja slučajeva korišćenja, U nekim slučajevima, slučaj korišćenja se može opraviti od greške i uspešno završiti, međutim, u nekim, ne može, i prekida izvršenje. Ako ne definišete akciju opravka sistema u uslovima kada dođe da izuzetka, onda sistem prekida rad.

Neke greške mogu da izazovu posledice i po više slučaja korišćenja, ili u više koraka normalnog toka događaja u jednom slučaju korišćenja. Na primer, gubitak veze u mreži, greška u bazi podataka, i dr. U ovakvim slučajevima ovo treba uzeti kao dodatne funkcionalne

zahteve, a ne kao izuzetke i ponavljati ih u svim slučajevima korišćenja u kojim može da dođe do ovakve greške

Morate izvršiti sve izuzetke ako želite da sprečite da vam tokovi stanu. Kodiranje u dobrom delu se odnosi na izvršenje izuzetaka. Međutim to je nužno da bi dobili robustan softver.

Pored tekstualnog opisa tokova, mogu se koristiti i grafičke metode, kao što su karte tokova ili UML dijagrami aktivnosti (slika 3).



Slika 2.3 UML dijagram aktivnosti sa glavnim i alternativnim tokom (scenarijom)

EXTEND I INCLUDE

Veza extend može povremeno da proširi osnovni slučaj korišćenja. Veza include omogućuje da se jedan slučaj korišćenja koristi u više slučajeva korišćenja.

Postoje dve veze između slučajeva korišćenja:

- **extend** - slučaj korišćenja koji u nekim slučajevima može da proširi drugi, osnovni slučaj korišćenja
- **include** - slučaj korišćenja koji je obavezan deo obično više drugih slučajeva korišćenja.

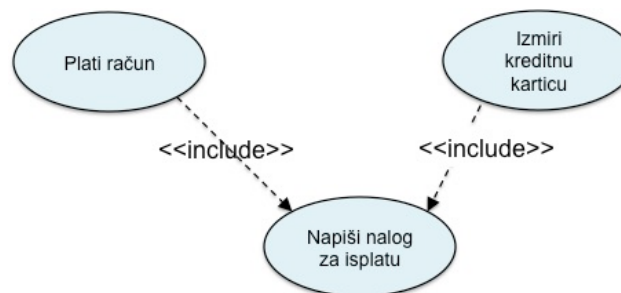
Navešćemo prvo primer veze extend u dijagramu slučajeva korišćenja projekta Chemical Tracking Request koji je dat ranije na slici 1. Na tom dijagramu se vidi da akter **Kupac** koristi slučaj korišćenja "**Traži katalog proizvođača**". Ako se pogleda specifikacija slučaja korišćenja "**Zahtev za hemikalijom**" na slici 2, vidi se da sistem daje **Podnosiocu zahteva za hemikalijom** izveštaj o kontejnerima sa tom hemikalijom (ako je poseduje). Sistem daje i opciju Podnosiocu zahteva da koristi slučaj korišćenja "**Traži katalog proizvođača**", Kada Podnosilac zahteva bira ovu opciju? Kada je **Skladište hemikalija** nema, ili nema dovoljno tražene hemikalije. To se povremeno dešava. Tada tok događaja prelazi na alternativni tok događaja (alternativni scenario) i ide se u izvršenje slučaja korišćenja "**Traži katalog proizvođača**" koje je u vezi sa **Kupcem**, tj. agentom koji predstavlja nabavnu službu organizacije. Kako se slučaj korišćenja "**Traži katalog proizvođača**" ponekad koristi, onda se može on povezati sa vezom **extend** (slika 4).



Izvor: Karl Wieggers, Joy Beaty, Software Requirements, 3rd ed., Microsoft, 2013

Slika 2.4 Primer korišćenja veze extend između dva slučaja korišćenja u CTS.

Za vezu include dajemo sledeći primer. Zamislimo da softver za računovodstvo koristi dva slučaja korišćenja **"Plati račun"** i **"Izmiri kreditnu karticu"**. Kako oba slučaja korišćenja podrazumevaju da korisnik treba da nalog za isplatu, onda se može uvesti treći slučaj korišćenja **"Daj nalog za isplatu"**, koji bi koristila oba navedene slučaja korišćenja (slika 5).



Izvor: Karl Wieggers, Joy Beaty, Software Requirements, 3rd ed., Microsoft, 2013

Slika 2.5 Primer korišćenja veze include u softveru za računovodstvo

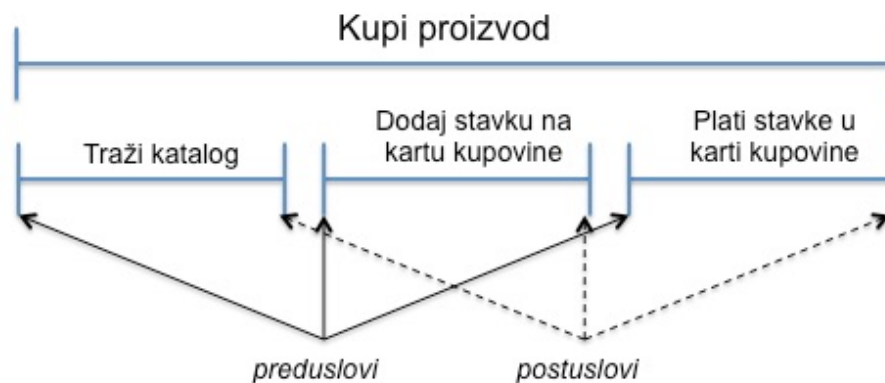
POVEZIVANJE PREDUSLOVA I POSTUSLOVA

Izjednačavanje postuslova sa prethodnim i preduslovima narednog slučaja korišćenja, mogu se povezati više sekvencijalnih slučajeva korišćenja u jedan, makro slučaj korišćenja.

Često je poželjno povezati nekoliko slučajeva korišćenja u jedan "makro" slučaj korišćenja koji sadrži više drugih slučajeva korišćenja. Time se jednom transakcijom pokreće izvršenje više drugih slučajeva korišćenja koje obuhvata "makro" slučaj korišćenja.

Navešćemo jedan primer, Veb sajt e-trgovine koi može koristiti sledeće slučajeve korišćenja: **"Traži katalog"**, **"Dodaj stavku u kartu kupovine"** i **"Plati stavke u karti kupovine"**. Ova tri slučaja korišćenja se uvek sekvencijalno izvršavaju. Zato je jednostavnije da se oni povežu u jedan "makro" slučaj korišćenja **"Kupi proizvod"**, pa kada se krene u njegovo izvršavanje, onda se krene u sekvencijalno izvršavanje sva tri slučaja korišćenja, u okviru samo jedne transakcije.

Kako ovo specificirati? Da bi opisani proces radio, svaki od tri navedena slučaja korišćenja treba po svom izvršenju da napuste sistem koji treba da bude u stanju koje je neophodno za izvršenje sledećeg slučaja korišćenja, bez odlaganja. Da bi se to ostvarilo, potrebno je da postuslovi prethodnog slučaja korišćenja budu identični sa preduslovima narednog slučaja korišćenja. Na slici 6 je prikazano ovo povezivanje postuslova i preduslova navedena tri slučaja korišćenja.



Izvor: Karl Wiegers, Joy Beatty, Software Requirements, 3rd ed., Microsoft, 2013

Slika 2.6 Primer izjednačavanja postsulova i preduslova radi sekvencijalnog povezivanja tri slučaja korišćenja u jedan slučaj korišćenja "Kupi proizvod".

SLUČAJEVI KORIŠĆENJA I POSLOVNA PRAVILA

Poslovna pravila mogu da utiču na naredne korake u normalnom scenariju definišući ispravne ulazne vrednosti ili određujući kako će se izvršavati obračun.

Slučajevi korišćenja i poslovna pravila su isprepletani. Neka poslovna pravila ograničavaju uloge svih ili pojedinih delova nekog slučaja korišćenja. Na primer, pravilo da samo korisnici sa posebnom dozvolom mogu da izvršavaju alternativne scenarije. To znači da ovo pravilo zahteva da se postavi dati preduslov i da se on testira pre nego što se korisniku dozvoli dalje izvršavanje. **Poslovna pravila mogu da utiču na naredne korake u normalnom scenariju definišući ispravne ulazne vrednosti ili određujući kako će se izvršavati obračuni.**

Pretpostavimo da jedan avio prevoznik naplaćuje dodatak na kartu putnika ako on želi da rezerviše posebno sedište u avionu. Kad putnik izvrši slučaj korišćenja u kome on rezerviše sedište u avionu, odgovarajuća poslovna pravila se aktiviraju i menjaju cenu karte putnika, ako je on izabrao određena sedišta sa višom tarifom.

Kada specificirate neko poslovno pravilo, zapišite identifikator onoga ko to poslovno pravilo koje utiče na slučaj korišćenja postavio i zapišite koji deo slučaja korišćenja svako poslovno pravilo zahvata.

Kada istražujete slučajeve korišćenja, možete otkriti značajna poslovna pravila. Kada hemičar koji učestvuje u izazivanju zahteva za Chemist Tracking System, bilo je zahteva da jedan korisnik ne bi trebalo da vidi pravila drugog, i obrnuto. Tako su došli do novog pravila: "Korisnik može da vidi samo pravila za zahteve koje je on postavio".

Ponekad, vi otkrijete pravila za vreme izazivanja zahteva i njihove analize, ponekad unapredite postojeća pravila, a ponekad vi već znate postojeća pravila koja će sistem morati da poštuje.

▼ Poglavlje 3

Utvrđivanje slučajeva korišćenja i zahteva u CTS

UTVRĐIVANJE SLUČAJEVA KORIŠĆENJA

Daje se preporuka postupka za utvrđivanje slučajeva korišćenja.

Možete utvrditi slučajeve korišćenja na nekoliko načina:

1. Prvo utvrdite aktere , onda postavite poslovne procese koje sistem treba da podrži, i onda definišite slučajeve korišćenja za aktivnosti u kojima raspodelu dolazi do interakcija sistema i aktera.
2. Kreirajte specifične scenarije za svaki poslovni proces, onda uopštite scenario za svaki slučaj korišćenja, i utvrdite aktere koji učestvuju u njima.
3. Upotrebom opisa poslovnih procesa, postavite sebi pitanje "Koje zadatke sistem treba da izvrši da bi ostvario do kraja ovaj poslovni proces, ili da konvertuje ulaze u izlaze. Ovi zadaci mogu da postanu slučajevi korišćenja.
4. Utvrdite spoljne događaje na koje sistem mora da odgovori, a onda povežite te događaje sa odgovarajućim akterima, i specifičnim slučajevima korišćenja.
5. Upotrebite CRUD analizu da bi pronašli unose podataka koje zahtevaju slučajevi korišćenja da bi kreirali, učitali, promenili, obrisali ili nešto drugo radili sa njima.
6. Ispitajte dijagram konteksta i pita se: "Koje ciljeve imaju svi spoljni entiteti, koji žele da ih ostvare, uz pomoć sistema.

CTS je primenio prvi pristup, upotrebljavajući poslovni proces koji je prethodni definisao. Tri biznis analitičara su održali su seriju dvočasovnih radionica, koje su se održavale dva puta nedeljno radi izazivanja zahteva.

Paralelno su održavane radionice sa drugim analitičarima, za različite grupe korisnika, To je moglo da se realizuje jer samo mali slučajeva korisnika zahtevalo učešće korisnika iz različitih grupa (klasa) korisnika.

Pre početka radionice, analitičari su tražili od učesnika da razmisle o zadacima koje bi trebalo da realizuju sa sistemom. Svaki zadatak je postao kandidat za jedan slučaj korišćenja. Dobijena lista mogućih slučajeva korišćenja sprečava da se neki od slučajeva korišćenja previde.

Neki slučajevi korišćenja su izbačeni jer su van okvira, a neki su spojeni u jedan veći i uopšteniji slučaj korišćenja. Grupe su takođe nalazile i nove slučajeve korišćenja koje prethodni nisu utvrdile iz zadataka.

Nazivi slučajeva korišćenja ukazuju na ciljeve koje imaju slučajevi korišćenja, te zato počinju sa glagolom. Ponekad analitičar mora pitanjem korisnika da utvrdi njihove ciljeve sa pojedinim slučajevima korišćenja, da bi im dao odgovarajući naziv.

Pre ulaženja u detaljnu analizu svakog slučaja korišćenja, prvo ih utvrdite da bi imali celinu pred očima. To vam olakšava određivanje prioriteta i početnu raspodelu slučajeva korišćenja po budućim izdanjima sistem, odn. iteracijama. Onda počnite da analizirate slučajeve korišćenja sa najvećim prioritetom.

ISTRAŽIVANJE SLUČAJEVA KORIŠĆENJA

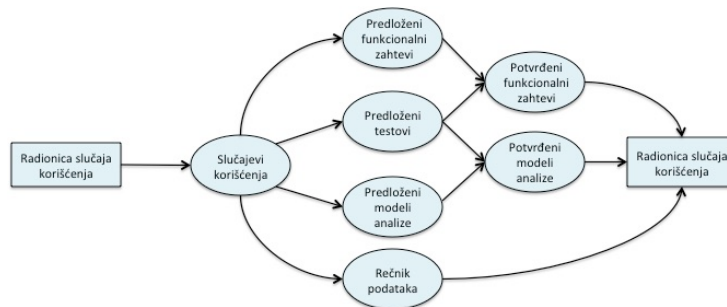
Na radionicama se utvrđuju svi elementi slučajeva korišćenja. Ili se koriste lepljive beleške, ili se projektuje uzorak specifikacije slučajeva korišćenja

Učesnici CTS radionica počinjali su rad na svakom slučaju korišćenja diskusijom s ciljem da utvrde aktera koji će imati korist od izvršenja slučaja korišćenja. Procenom frekvencije korišćenja slučaja korišćenja (SK), dobili su prvi indikator istovremenog korišćenja i zahteva kapaciteta.

Zatim su prešli na definisanje preduslova i postuslova, na granicama slučajeva korišćenja. Svi koraci u slučaju korišćenja se nalaze između ovih granica. Tokom diskusije vršile su se popravke ovih uslova. Zatim, analitičari su pitali učesnike kako zamišljaju interakciju sa sistemom za vreme izvršenja zadatka. Dobijeni redosled akcija aktera odredio je normalni scenario slučaja korišćenja. Analitičari su beležili predloge akcija aktera na lepljivim beleškama koje su lepili na tablu ili papir. Njihovim premeštanjem lako se vrši njihovo grupisanje ili zamena. Drugi način je da se projektuje formular za specifikaciju slučaja korišćenja i onda da ga grupa kolektivno popunjava. Sličan postupak je primenjen i za alternativne scenarije i izuzetke. Mnogi izuzeci su otkriveni kada je analitičar pitao "Šta bi se desilo ako bi u jednom trenutku prestala veza sa bazom podataka?" ili "Šta bi se desilo ako hemikalija nije komercijalno raspoloživa". Rad u grupi omogućava i da se govori i o očekivanom kvalitetu sistema, u vidu vremena odziva, raspoloživosti sistema, zahteva bezbednosti, i projektnih ograničenja korisničkog interfejsa.

Radionice su prvo radili sa SK sa većim prioritetom, i u više sesija.

Slika 1 prikazuje redosled dobijenih rezultata rada za vreme procesa utvrđivanja slučajeva korišćenja za slučaj CTS. Posle radionice, analitičar je na osnovu rezultata radionice, popunio specifikaciju slučajeva korišćenja, procenjujući potreban nivo detaljnosti specifikacije SK. . Proces na slici 1 pokazuje da su nakon svake radionice BA iz sistema za praćenje hemikalija proizveli funkcionalne zahteve softvera iz slučajeva upotrebe. BA su takođe izradili neke modele analize, kao što je dijagram prelaska stanja koji je pokazao sve moguće statuse hemijskih zahteva i dozvoljene promene statusa. Slučajevi višestruke upotrebe mogu manipulirati hemijskim zahtevom, pa dijagram objedinjuje informacije i operacije koje obuhvataju nekoliko slučajeva upotrebe



Izvor: Karl Wiegiers, Joy Beatty, Software Requirements, 3rd ed., Microsoft, 2013

Slika 3.1 Rezultati utvrđivanja slučajeva korišćenja na radionicama

POTVRĐIVANJE SLUČAJEVA KORIŠĆENJA

Umesto jednog načina predstavljanja zahteva dobijenih iz slučaja korišćenja, treba koristiti više načina predstavljanja, jer se tada otkrivaju više grešaka.

Dan ili dva nakon svake radionice, BA je dao slučajeve upotrebe i funkcionalne zahteve učesnicima radionice, koji su ih pregledali pre sledeće radionice. Ovi neformalni pregledi otkrili su mnogo grešaka: prethodno neotkriveni alternativni tokovi, novi izuzeci, pogrešni funkcionalni zahtevi i nedostajući koraci dijaloga. Tim je brzo naučio da dozvoli bar jedan dan između uzastopnih radionica. Mentalno opuštanje koje dolazi nakon dan ili dva omogućava ljudima da pregledaju svoj raniji rad iz sveže perspektive. Jedan BA koji je održavao svakodnevne radionice utvrdio je da su učesnici imali poteškoća da uoče greške u pregledanim materijalima, jer su im informacije bile previše sveže. Mentalno su recitirali nedavnu diskusiju i nisu videli greške. Početkom razvoja potreba, testni sistem sistema za praćenje hemijskih proizvoda počeo je kreirati konceptualne testove - nezavisno od specifičnosti primene i korisničkog interfejsa - od slučajeva primene. Ovi testovi su pomogli tim da postigne zajedničko razumevanje kako se sistem treba ponašati u određenim scenarijima. Testovi omogućavaju BA-ima da provere jesu li izvukli funkcionalnost koja je potrebna kako bi korisnici izvršili svaki slučaj upotrebe. Tokom završne radionice o izlasku, učesnici su zajedno prolazili testove kako bi bili sigurni da su se dogovorili o tome kako treba da rade slučajevi upotrebe.

Rano konceptualno probno razmišljanje poput ovog mnogo je jeftinije i brže od pisanja koda, izgradnje dela sistema, izvođenja testova i tek tada otkrivanja problema sa zahtevima. To je analogno agilnom pristupu izmišljanja korisničkih priča sa testovima prihvatanja, ali CTS tim je napisao i funkcionalne zahteve i testove. Izvršeno je upoređivanje dve otkrivene greške pre nego što je bilo koji kod napisan. CTS tim stvorio je višestruko predstavljanje zahteva koje su identifikovali: spisak funkcionalnih zahteva, skup odgovarajućih testova i modele analize, a sve zasnovano na slučajevima korišćenja. Upoređivanje ovih alternativnih stavova sa zahtevima je moćna tehnika kvaliteta. Tim je koristio testove za verifikaciju funkcionalnih zahteva, tražeći testove koji se ne mogu „izvršiti“ sa setom zahteva i zahtevima koji nisu obuhvaćeni testovima.

Ako stvorite samo jedno predstavljanje ili jedan pogled na zahteve, morate joj verovati. Nemate s čim da ga uporedite radi traženja grešaka, praznina i različitih interpretacija.

Agilni projektni timovi obično ne dokumentuju funkcionalne zahteve, radije stvaraju testove prihvatanja. Iako je razmišljanje o testiranju tokom istraživanja zahteva odlična ideja za svaki projekat, ipak vam ostavlja samo jedan prikaz zahteva za koje morate verovati da su tačni. Slično tome, tradicionalni projektni timovi koji stvaraju samo niz funkcionalnih zahteva i testiranje ostave za kasnije u projektu imaju samo jedno predstavljanje. Dobićete najbolje rezultate razumnom kombinacijom pismenih zahteva, testova, modela analize i prototipa.

SLUČAJI KORIŠĆENJA I FUNKCIONALNI ZAHTEVI

Slučajevi upotrebe opisuju perspektivu korisnika, sagledavajući spoljno vidljivo ponašanje sistema. Oni ne sadrže sve informacije potrebne programeru.

Programeri softvera ne primenjuju poslovne zahteve ili zahteve korisnika. Oni implementiraju funkcionalne zahteve, specifične bitove ponašanja sistema. Neki praktičari smatraju da su slučajevi upotrebe funkcionalni zahtevi. Međutim, videli smo da mnoge organizacije upadaju u probleme kada jednostavno prenesu svoje primene namenjene programerima na implementaciju. Slučajevi upotrebe opisuju perspektivu korisnika, sagledavajući spoljno vidljivo ponašanje sistema. Oni ne sadrže sve informacije potrebne programeru da bi napisao softver. Korisnik bankomata ne zna za bilo kakvu pozadinsku obradu, poput komunikacije sa računarom banke. Ovaj detalj je korisniku nevidljiv, ali programer mora znati za njega. Programeri koji primaju čak i potpuno opisane slučajeve upotrebe često imaju mnogo pitanja. Da biste smanjili ovu nesigurnost, razmislite o tome da BA ima izričito specificiranje funkcionalnih zahteva neophodnih za implementaciju svakog slučaja upotrebe.

Mnogi funkcionalni zahtevi ispadaju iz dijaloških koraka između aktera i sistema. Neke su očigledni, kao što je „Sistem će svakom zahtevu dodeliti jedinstveni redni broj.“ Nema smisla da ih duplirate negde drugde ako su to jasni zahtevi iz slučaja upotrebe. Ostali funkcionalni zahtevi se ne prikazuju u opisu slučaja upotrebe. Na primer, način na koji se dokumentiraju slučajevi upotrebe obično ne precizira šta sistem treba da radi ako nije ispunjen preduslov.

Ovo je primer kako slučajevi upotrebe često ne pružaju sve potrebne informacije programeru da zna šta da gradi. BA mora izvući te nedostajuće zahteve i dostaviti ih programerima i ispitivačima. Analiza dobijanja pogleda programera na zahteve na osnovi pogleda korisnika na zahteve, treba da izvrši BA i time da doda vrednost projektu

Sistem praćenja hemikalija koristio je slučajeve upotrebe prvenstveno kao alat za otkrivanje potrebnih funkcionalnih potreba. Analitičari su napisali samo povremene opise manje složenih slučajeva upotrebe. Zatim su izvukli sve funkcionalne zahteve koji bi, kada se primene, omogućili akteru da izvrši slučaj upotrebe, uključujući alternativne tokove i rukovaoce izuzetaka. Analitičari su ove funkcionalne zahteve dokumentovali u SRS-u, koji je organizovan kao svojstvo proizvoda.

Funkciju povezanu sa slučajem korišćenja možete dokumentovati na više načina. Nijedna od sledećih metoda nije savršena, pa odaberite pristup koji najbolje odgovara načinu na koji želite da dokumentujete i upravljate softverskim zahtevima vašeg projekta.

NAČINI DOKUMENTOVANJA ZAHTEVA NA OSNOVU SLUČAJEVA UPOTREBE

Postoji nekoliko načina dokumentovanja funkcionalnih zahteva koji su proizašli iz slučajeva korišćenja.

Koristite samo slučajeve upotrebe

Ovo je jedna od mogućnosti je da se uključe funkcionalni zahtevi uz svaku specifikaciju slučaja upotrebe, ako već nisu očigledni. I dalje ćete morati da dokumentujete nefunkcionalne zahteve i bilo koju funkciju koja nije povezana sa slučajem korišćenja. Pored toga, može d se desi da nekoliko slučajeva upotrebe koristi isti funkcionalni zahtev. Ako pet slučajeva upotrebe zahteva da se identitet korisnika potvrdi, ne želite da pišete pet različitih blokova koda u tu svrhu. Umesto da ih duplira, funkcionalni zahtevi koji se pojavljuju u višestrukim slučajevima koriste se unakrsnim referencama. Slučajevi upotrebe mogu se prikupiti u dokumentu o potrebama korisnika.

Koristite slučajeve upotrebe i funkcionalne zahteve

Druga opcija je pisanje prilično jednostavnih slučajeva korišćenja i dokumentovanje funkcionalnih zahteva proisteklih iz svakog u SRS-u ili skladištu zahteva. U ovom pristupu trebalo bi da uspostavite sledljivost između slučajeva upotrebe i njihovih povezanih funkcionalnih potreba. Na taj način, ako se slučaj upotrebe promeni, brzo možete pronaći pogođene funkcionalne zahteve. Najbolji način za upravljanje sledljivošću je korišćenje alata za upravljanje zahtevima.

Samo funkcionalni zahtevi

Još jedna opcija je da organizujete svoje funkcionalne zahteve u slučaju upotrebe ili po karakteristikama i da uključite i slučajeve upotrebe i funkcionalne zahteve u SRS ili skladište zahteva. Ovo je

pristup koji je koristio CTS tim, a isto smo uradili i na nekoliko projekata za razvoj veb lokacija. Pisali smo većinu naših slučajeva u vrlo sažetom obliku, ne dovršavajući ceo obrazac sa slike 2 u poglavlju 2. Detalji su zatim specificirani kroz skup funkcionalnih zahteva. Ovaj pristup ne rezultira posebnim dokumentom o potrebama korisnika.

Koristite slučajeve upotrebe i testove

Ako napišete detaljne specifikacije slučajeva upotrebe i funkcionalne zahteve, možda ćete primetiti neko dupliranje, posebno oko normalnog protoka. Mala je vrednost pisanja istog zahteva dva puta. Dakle, druga strategija je pisati prilično kompletne specifikacije slučaja upotrebe, ali zatim napisati testove prihvatanja kako biste utvrdili da li sistem pravilno postupa sa osnovnim ponašanjem slučaja upotrebe, alternativnim putevima uspeha i različitim stvarima koje mogu poći po zlu.

IZBEGNITE NEKE ZAMKE PRI RADU SA SLUČAJEVIMA KORIŠĆENJA

Daje se pet preporuka za izbegavanje zamki pri kreiranju slučajeva korišćenja.

- **Koristite isuviše mnogo slučajeva korišćenja:** Ne pišite poseban slučaj korišćenja za svaki scenario. Imaćete mnogo više slučajeva korišćenja nego poslovnih zahteva i svojstava, ali i mnogo više funkcionalnih zahteva nego slučajeva korišćenja.
- **Upotreba vrlo složenih slučajeva korišćenja:** Vrlo složeni slučajevi korišćenja nisu razumljivi. Ako ne možete da pojednostavite poslovne zadatke, možete da pojednostavite njihovo predstavljanje u slučajeva korišćenja. Izaberite jedan uspešan put slučaja korišćenja i programirajte ga normalnim tokom, tj. scenarijom. Koristite alternativne tokove (scenarije) i izuzetke za opis grananja u tokovima. Možete imati puno alternativnih tokova, ali će oni biti kratki i laki za razumevanje. Ako tok ima više od 10 do 15 koraka (akcija), vidite da li on stvarno predstavlja samo jedan scenario. Međutim nemojte ga slobodno preseći, samo zato što je dugačak.
- **Uključivanje projektovanje u slučajeve korišćenja:** Slučajevi korišćenja bi trebalo da se usmere ka opisivanju ŠTA korisnik očekuje od sistema, a ne KAKO sistem to treba da izvrši, ili kako će izgledati prikaz na monitoru. Koristite konceptijske interakcije između aktera i sistema. Možete koristiti skice izgleda prikaza na monitoru, ali samo radi vizualizacije interakcije aktera i sistema, a ne kao specifikaciju projektnog rešenja.
- **Uključivanje definisanje podataka u slučajeve korišćenja:** Neki autori uključuju definisanje značajnih podataka u slučajevima korišćenja. To otežava njihovo nalaženje, jer nije jasno koji slučaj korišćenja koristi koji podatak. Takođe, može voditi i duploj specifikaciji istog podatka. Definicije podataka treba posebno skladištiti u rečniku podataka i u modelu podataka.
- **Slučajevi korišćenja koje korisnici ne razumeju:** Nejasne i efektivne komunikacije. Problem je ako korisnici ne mogu da povežu neki slučaj korišćenja sa njihovim poslovnim procesom i ciljevima. Pišite slučajeve korišćenja iz perspektive korisnika, ne iz perspektive sistema, i pitajte korisnike da ih recenziraju. Koristite jednostavne slučajeve korišćenja, a da mogu da ostvare postavljene ciljeve.

PREDNOSTI KORIŠĆENJA ZAHTEVA KONCENTRISANIH NA KORISNIKA

Slučajevi korisnika pomažu da se razjasne razne nejasnoće i protivurečnosti u ranim fazama razvoja sistema, a i mogu da generišu testove na osnovu slučajeva korišćenja.

Snaga slučajeva korišćenja i priča korisnika je u njihovoj perspektivi klasa korisnika, koncentrisanoj ka korisniku. Korisniku će biti jasnije šta može da očekuje od sistema, nego da je primenjen koncept slučajeva korišćenja koji je usmeren ka svojstvima (feature-based).

Slučajevi korišćenja pomažu analitičarima i inženjerima razvoja da razumeju posao korisnika. Slučajevi korisnika im pomažu da razjasne razne nejasnoće i protivurečnosti u ranim fazama razvoja sistema, a i mogu da generišu testove na osnovu slučajeva korišćenja.

Specificiranje prevelikog broja dostižnih funkcionalnih zahteva unapred, dovodi do primene i neposrednih zahteva. Fokus na korisnika pre kreiranja slučajeva korišćenja, dovodi do sistema koji dozvoljava korisniku da izvrši određene zadatke, koji su verifikovani.

Razvoj zahteva korisnika pomaže u određivanju prioriteta zahteva. Najveći prioritet dobijaju funkcionalni zahtevi koji proizilaze iz korisničkih zahteva koji imaju najviši prioritet, kao što su:

- oni koji opisuju ključne poslovne procese koje sistem treba da podrži,
- oni koje koriste često veći broj korisnika,
- oni koje zahteva glavna klasa korisnika
- oni koji treba da zadovolje propise.

Slučajevi korišćenja donose i tehničku korist. Oni otkrivaju važne objekte domena i njihove međusobne odnose. Inženjeri razvoja objektno-orijentisanih sistema mogu da mapiraju slučajeve korišćenja u modela objekata i sekvencijalne dijagrame.

Kako se poslovni procesi vremenom menjaju, i zadaci koji se vezuju za pojedine zahteve korisnika se vremenom menjaju. Ako povežete funkcionalne zahteve, projektna rešenja, kod, i testove, sa njihovim zahtevima korisnika od kojih su potekli, onda možete onda možete da kaskadno povežete ove promene zahteva korisnika kroz ceo sistem. To znači jači "glas kupca".

VIDEO 11 - WHAT IS A USE CASE - WIEGERS (VIDEO)

Trajanje: 6:56 minuta

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

VIDEO 12 - USE CASES, SCENARIOS, AND STORIES - WIEGERS (VIDEO)

Trajanje: 4:57 minuta

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

VIDEO 13 - USE CASES AND FUNCTIONAL REQUIREMENTS - WIEGERS (VIDEO)

Trajanje: 4:53 minuta

Ova lekcija sadrži video materijal. Ukoliko želite da pogledate ovaj video morate da otvorite LAMS lekciju.

▼ Poglavlje 4

Vežba

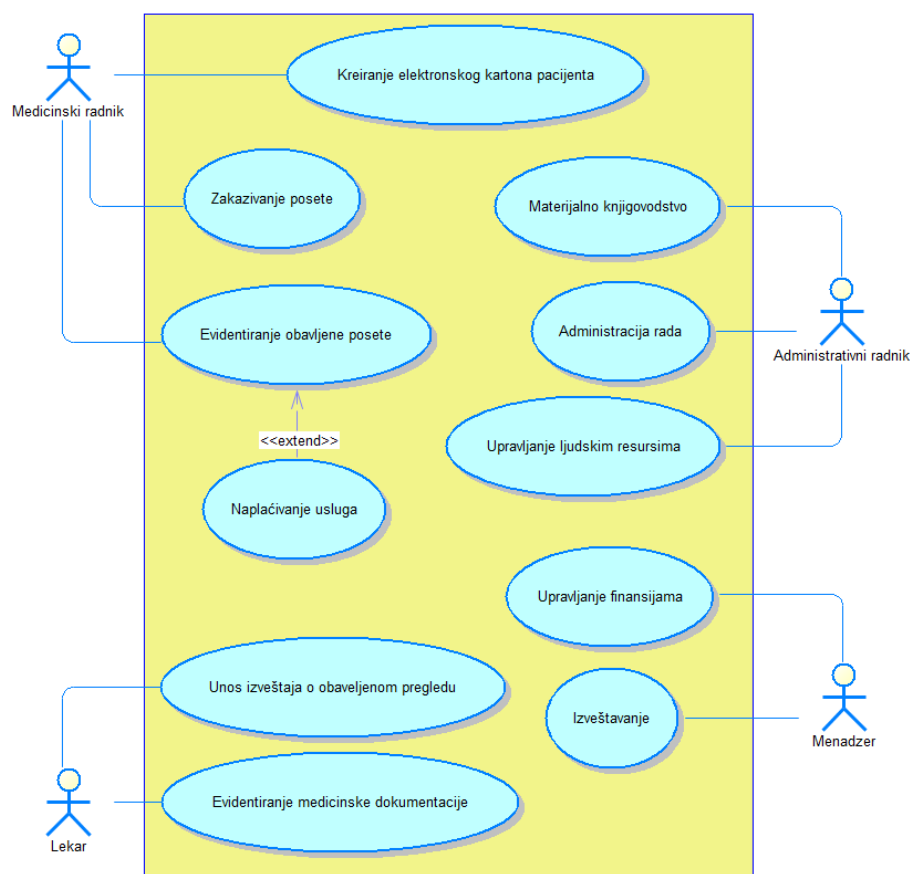
DIJAGRAM SLUČAJEVA KORIŠĆENJA

Dijagram slučajeve korišćenja za poslovni sistem privatne klinike

U ovoj vežbi je izrađen dijagram slučajeve korišćenja (slika 1), koji predstavlja osnov za razumevanje potreba korisnika i za utvrđivanje njegovih zahteva. Skiciranje slučajeve upotrebe na ovaj način biće ispraćeno i definisanjem scenarija, tj. tokova akcija, koje se moraju izvršiti u svakom pojedinačnom slučaju korišćenja, radi utvrđivanja funkcionalnih zahteva u vežbama koje slede.

Slučaj korišćenja prikuplja sve scenarije, pokazujući sve načine na koje se cilj može uspešno postići ili neuspešno završiti. Ukoliko je neki korak u slučaju korišćenja složen i može uspeti ili ne uspeti, treba ga predstaviti kao pod slučaj korišćenja. Da biste slučajeve korišćenja pravilno identifikovali, o njima treba razmišljati kao o priči ili igri koja se neprekidno odvija. Ako se ta priča ilustruje, a zatim prateći ilustraciju, napiše u tačnim koracima, umnogome će pomoći svim učesnicima na razvoju bilo kog softverskog proizvoda.

Opisi slučajeve korišćenja za aktera Medicinski radnik, uslediće na slikama 2-9.



Slika 4.1 Dijagram slučajeva korišćenja za poslovni sistem privatne klinike

OPISI SLUČAJA KORIŠĆENJA UC-1 ZA POSLOVNI SISTEM PRIVATNE KLINIKE

UC-1: Kreiranje elektronskog kartona pacijenta

ID i naziv:	UC-1 Kreiranje elektronskog kartona pacijenta		
Kreator:	Marina Damjanović	Datum kreiranja:	08/09/19
Primarni akter:	Medicinski radnik	Sekundarni akteri:	Lekar
Pokretač:	Medicinski radnik treba da kreira elektronski karton za pacijenta koji nikada nije lečen u klinici		
Opis:	Medicinskom radniku treba omogućiti unos podataka koji se traže za formiranje elektronskog kartona pacijenta i čuvanje kartona za dalju upotrebu.		
Preduslovi:	PRE-1. Postoji konekcija na internet. PRE-2. Baza podataka je onlajn. PRE-3. Medicinski radnik je autentifikovan pomoću svojih podataka za pristup sistemu.		
Postuslovi:	POST-1. Elektronski karton novog pacijenta je uspešno sačuvan. POST-2. Pacijentu se može zakazati pregled.		
Normalni tok:	1.0 Kreiranje elektronskog kartona novog pacijenta <ol style="list-style-type: none"> 1. Medicinski radnik zahteva kreiranje novog pacijenta 2. Sistem daje formu za evidentiranje ličnih podataka, prebivališta, kontakt podataka, podataka o staratelju i dodavanje komentara. 3. Sistem daje mogućnost unosa svih podataka ručno ili preko čitača lične karte (pogledati 1.1) 4. Medicinski radnik unosi potrebne podatke i potvrđuje unos klikom na dugme za čuvanje 5. Medicinski radnik traži napuštanje slučaja korišćenja ili traži unos dodatnih podataka (pogledati 1.2) 6. Sistem čuva elektronski karton novog pacijenta. 		

Slika 4.2 UC-1 Kreiranje elektronskog kartona pacijenta [Izvor: Marina Damjanović]

Alternativni tokovi:	1.1 Unos ličnih podataka preko čitača lične karte <ol style="list-style-type: none"> 1. Medicinski radnik bira učitavanje ličnih podataka direktno iz lične karte. 2. Sistem daje signal da je spreman za očitavanje lične karte. (pogledati 1.1.E1) 3. Medicinski radnik unosi ličnu kartu u čitač i pokreće očitavanje podataka. 4. Sistem obrađuje podatke i daje signal kada su podaci očitani i lična karta se može skloniti sa čitača. 5. Medicinski radnik odstranjuje ličnu kartu iz čitača. 6. Medicinski radnik menja podatke ako je potrebno i potvrđuje unos klikom na dugme za čuvanje. 7. Sistem čuva elektronski karton novog pacijenta. 1.2 Unos dodatnih podataka o pacijentu <ol style="list-style-type: none"> 1. Medicinski radnik traži unos podataka o privatnom zdravstvenom osiguranju. 2. Sistem daje pretragu postojećih osiguravajućih kuća, tipa osiguranja i mogućnost unosa osiguravajućeg koda. 3. Medicinski radnik unosi podatke dobijene od pacijenta i zahteva čuvanje. 4. Sistem ažurira elektronski karton novog pacijenta. 5. Medicinski radnik traži unos podataka o rizicima i alergijama. 6. Sistem daje forme za unos podataka o tipovima rizika ili alergija i vremenskom trajanju. 7. Medicinski radnik unosi podatke dobijene od pacijenta i zahteva čuvanje. 8. Sistem ažurira elektronski karton novog pacijenta.
Izuzeci:	1.1.E1 Aplikacija za očitavanje lične karte nije aktivna <ol style="list-style-type: none"> 1. Sistem prikazuje poruku da aplikacija za očitavanje lične karte nije trenutno aktivna. 2. Sistem vraća korisnika na formu za ručni unos ličnih podataka. 3. Medicinski radnik unosi potrebne podatke i zahteva čuvanje. 4. Sistem čuva elektronski karton novog pacijenta.
Prioritet:	Visok
Frekvencija upotrebe:	Minimum 1 dnevno, maksimum 10 dnevno. Na nedeljnom nivou, minimum 6 puta, maksimum 60 puta.
Poslovna pravila:	...
Druge informacije:	Treba da postoji mogućnost učitavanja ličnih podataka iz čipovane lične karte. Čitač lične karte mora da bude integrisan sa sistemom i treba obezbediti aplikaciju koja omogućava čitanje podataka.
Pretpostavke:	Pretpostavlja se da je lična karta važeća. Pretpostavlja se da je čitač lične karte ispavan.

Slika 4.3 UC-1 Kreiranje elektronskog kartona pacijenta [Izvor: Marina Damjanović]

OPISI SLUČAJA KORIŠĆENJA UC-2 ZA POSLOVNI SISTEM PRIVATNE KLINIKE

UC-2: Zakazivanje posete

ID i naziv:	UC-2 Zakazivanje posete		
Kreator:	Marina Damjanović	Datum kreiranja:	08/09/19
Primarni akter:	Medicinski radnik	Sekundarni akteri:	Lekar
Pokretač:	Medicinski radnik treba da zakaže posetu pacijentu koji ima kreiran elektronski zdravstveni karton u sistemu privatne klinike.		
Opis:	Medicinskom radniku treba omogućiti zakazivanje posete lekaru u terminu u kome je lekar dostupan za rad, u skladu sa važećim rasporedom.		
Preduslovi:	PRE-1. Postoji konekcija na internet. PRE-2. Baza podataka je onlajn. PRE-3. Medicinski radnik je autentifikovan pomoću svojih podataka za pristup sistemu. PRE-4. Pacijent ima kreiran elektronski zdravstveni karton u sistemu privatne klinike.		
Postuslovi:	POST-1. Pacijentu je uspešno zakazana poseta adekvatnom lekaru u slobodnom terminu.		
Normalni tok:	2.0 Zakazivanje posete lekaru <ol style="list-style-type: none"> 1. Medicinski radnik zahteva prikaz rasporeda rada za predstojeći period. 2. Sistem daje prikaz opšteg rasporeda rada u formi kalendara. 3. Medicinski radnik bira dan kada pacijent želi da zakaže posetu 4. Medicinski radnik proverava dostupnost lekara za odabrani dan i slobodne termine. 5. Medicinski radnik zahteva da zakaže posetu za odabranog lekara i odabrani termin. (pogledati 2.0.E1 i 2.0.E2) 6. Sistem traži od medicinskog radnika da unese JMBG pacijenta kome želi da zakaže posetu. 7. Sistem nalazi registrovanog pacijenta ili zahteva od medicinskog radnika da prvo kreira elektronski karton za novog pacijenta. (pogledati 2.1) 8. Sistem uspešno čuva rezervisanu posetu. 		

Slika 4.4 UC-2 Zakazivanje posete [Izvor: Marina Damjanović]

Alternativni tokovi:	2.1 Zakazivanje posete lekaru pacijentu koji nema otvoren karton u sistemu privatne klinike <ol style="list-style-type: none"> 1. Sistem prikazuje poruku da pacijentu nema otvoren karton u sistemu privatne klinike. 2. Sistem daje mogućnost medicinskom radniku da kreira elektronski karton novom pacijentu (3a) ili da prekine proces zakazivanja posete (4b). 3. a. Medicinski radnik bira da kreira elektronski karton novom pacijentu 3. b. Sistem pokreće izvršenje slučaja korišćenja UC-1. 3. c. Medicinski radnik kreira elektronski karton novom pacijentu. 3. d. Sistem preusmerava medicinskog radnika da nastavi sa izvršenjem tekućeg slučaja korišćenja. 4. a. Medicinski radnik bira da prekine proces zakazivanja. 4. b. Sistem prekida izvršenje slučaja korišćenja.
Izuzeci:	2.0.E1 Željeni lekar nije u rasporedu rada za odabrani dan <ol style="list-style-type: none"> 1. Sistem ispisuje poruku da traženi lekar nije u rasporedu za odabrani dan. 2. Medicinski radnik ponovo izvršava normalan tok tekućeg slučaja korišćenja. 3. Medicinski radnik vrši pretragu rasporeda rada prema željenom lekaru kako bi pronašao prvi sledeći slobodan termin kod istog lekara. 2.0.E2 Traženi lekar nema više dostupnih termina za odabrani dan <ol style="list-style-type: none"> 1. Sistem ispisuje poruku da traženi lekar nema više dostupnih termina za odabrani dan. 2. Medicinski radnik ponovo izvršava normalan tok tekućeg slučaja korišćenja. 3. Medicinski radnik vrši pretragu rasporeda rada prema željenom lekaru kako bi pronašao prvi sledeći slobodan termin kod istog lekara.
Prioritet:	Visok
Frekvencija upotrebe:	U proseku 20 puta na dan, a maksimalno 200 puta na nedeljnom nivou.
Poslovna pravila:	...
Druge informacije:	/
Pretpostavke:	<p>Podrazumeva se da su dnevni rasporedi unapred kreirani. Dnevni rasporedi su kreirani na osnovu rasporeda rada angažovanog osoblja, rasporeda radnih/neradnih dana i radnog vremena klinike.</p> <p>Podrazumeva se da su rasporedi usaglašeni sa regularnim godišnjim kalendarima.</p>

Slika 4.5 UC-2 Zakazivanje posete [Izvor: Marina Damnjanović]

OPISI SLUČAJA KORIŠĆENJA UC-3 ZA POSLOVNI SISTEM PRIVATNE KLINIKE

UC-3: Evidentiranje realizovane posete

ID i naziv:	UC-3 Evidentiranje realizovane posete		
Kreator:	Marina Damjanović	Datum kreiranja:	08/09/19
Primarni akter:	Medicinski radnik	Sekundarni akteri:	Lekar, Menadžer
Pokretač:	Medicinski radnik treba da evidentira da je poseta lekaru realizovana.		
Opis:	Medicinskom radniku treba omogućiti unos podataka o realizaciji posete lekaru koja je prethodno bila zakazana.		
Preduslovi:	PRE-1. Postoji konekcija na internet. PRE-2. Baza podataka je onlajn. PRE-3. Medicinski radnik je autentifikovan pomoću svojih podataka za pristup sistemu. PRE-4. Poseta lekaru je prethodno bila zakazana.		
Postuslovi:	POST-1. Medicinski radnik je uneo potvrdu o realizaciji zakazane posete.		
Normalni tok:	3.0 Evidentiranje realizovane posete lekaru <ol style="list-style-type: none"> 1. Medicinski radnik zahteva prikaz rasporeda poseta za dan. 2. Sistem daje dnevni raspored poseta za dan. 3. Medicinski radnik bira posetu za koju treba potvrditi realizaciju. (pogledati 3.0.E1 i 3.0.E2) 4. Medicinski radnik unosi podatke o trajanju posete. 5. Medicinski radnik proverava da li ima uputa za druge lekare nakon obavljenog pregleda i zakazuje nove preglede, ako ih ima. (pogledati 3.1) 6. Medicinski radnik proverava da li ima usluga za naplatu i izvršava naplaćivanje usluga, ako ih ima (pogledati UC-4) 7. Medicinski radnik traži da sačuva evidenciju o obavljenoj poseti klikom na dugme. 8. Sistem ažurira istoriju dnevnih poseta za taj dan. 9. Sistem ažurira istoriju poseta pacijenta u okviru njegovog elektronskog kartona. 		

Slika 4.6 UC-3 Evidentiranje realizovane posete [Izvor: Marina Damjanović]

Alternativni tokovi:	3.1 Zakazivanje dodatnih poseta za pacijenta koji je obavio pregled kod lekara <ol style="list-style-type: none"> 1. Medicinski radnik zahteva izvršenje slučaja korišćenja UC-2. 2. Sistem pokreće normalan tok slučaja korišćenja UC-2. 3. Medicinski radnik izvršava slučaj korišćenja UC-2. 4. Sistem preusmerava medicinskog radnika da nastavi sa izvršenjem tekućeg slučaja korišćenja.
Izuzeci:	3.0.E1 Pacijent je hitan slučaj i nije imao kreiranu rezervaciju posete <ol style="list-style-type: none"> 1. Medicinski radnik zahteva unos hitne posete. 2. Sistem daje formu za evidentiranje hitne posete. 3. Medicinski radnik unosi podatke o trajanju hitne posete i beleži koji je lekar preuzeo pacijenta. 4. Sistem traži od medicinskog radnika da unese JMBG pacijenta koji je imao hitnu posetu. 5. Sistem nalazi registrovanog pacijenta (5.a) ili zahteva od medicinskog radnika da prvo kreira elektronski karton za novog pacijenta (6.a). 5.a. Medicinski radnik proverava ostale podatke o obavljenom pregledu i vrši naplatu usluga. 5.b. Sistem pokreće izvršenje slučaja korišćenja UC-4. 6.a. Sistem pokreće izvršenje slučaja korišćenja UC-1. 6.b. Medicinski radnik kreira elektronski karton novom pacijentu. 6.c. Sistem preusmerava medicinskog radnika da nastavi sa izvršenjem tekućeg slučaja korišćenja. 3.0.E1 Pacijent se nije pojavio ili je prethodno otkazao pregled <ol style="list-style-type: none"> 1. Medicinski radnik označava termin kao ponovo slobodan i čuva promenu. 2. Sistem prikazuje termin prikazuje kao slobodan i dozvoljava novu rezervaciju posete u tom terminu.
Prioritet:	Srednji
Frekvencija upotrebe:	U proseku 30 puta na dan, a maksimalno 200 puta na nedeljnom nivou.
Poslovna pravila:	...
Druge informacije:	/
Pretpostavke:	/

Slika 4.7 UC-3 Evidentiranje realizovane posete [Izvor: Marina Damnjanović]

OPISI SLUČAJA KORIŠĆENJA UC-4 ZA POSLOVNI SISTEM PRIVATNE KLINIKE

UC-4: Naplaćivanje usluga

ID i naziv:	UC-4 Naplaćivanje usluga		
Kreator:	Marina Damnjanović	Datum kreiranja:	08/09/19
Primarni akter:	Medicinski radnik	Sekundarni akteri:	Lekar, Menadžer
Pokretač:	Medicinski radnik treba da naplati usluge koje je evidentirao u sistemu.		
Opis:	Medicinskom radniku treba omogućiti da može pacijentu da naplati korišćene medicinske usluge, prema važećem cenovniku klinike, i da mu izda dokaz o plaćanju u vidu fiskalnog računa.		
Preduslovi:	PRE-1. Postoji konekcija na internet. PRE-2. Baza podataka je onlajn. PRE-3. Fiskalna kasa je u funkciji. PRE-4. Fiskalna kasa je uključena.		
Postuslovi:	POST-1. Naplata usluga je uspešno izvršena i evidentirana u sistemu. POST-2. Pacijent je dobio račun kao potvrdu plaćanja.		
Normalni tok:	4.0 Naplaćivanje pruženih medicinskih usluga pacijentu <ol style="list-style-type: none"> 1. Medicinski radnik zahteva da naplati usluge koje su evidentirane u okviru posete pacijenta. 2. Sistem daje medicinskom radniku pregled stavki koje će biti naplaćene prema važećem cenovniku. 3. Sistem daje medicinskom radniku da odabere način na koji će pacijentu naplatiti usluge: u gotovini ili karticom (videti 4.1) 4. Medicinski radnik bira način plaćanja i traži izdavanje računa. 5. Sistem obrađuje naplatu i komunicira sa fiskalnom kasom, na kojoj se štampa račun. 6. Sistem prikazuje poruku o uspešnom naplaćivanju usluga. 7. Sistem ažurira podatke o plaćanjima. 		

Slika 4.8 UC-4 Naplaćivanje usluga [Izvor: Marina Damnjanović]

Alternativni tokovi:	4.1 Plaćanje se vrši karticom <ol style="list-style-type: none"> 1. Medicinski radnik bira naplaćivanje usluga sa kartice. 2. Sistem traži unos iznosa za naplatu na POS terminalu. 3. POS terminal traži od medicinskog radnika da prisloni platnu karticu na POS terminal radi izvršenja transakcije. (videti 4.1.E1 i 4.1.E2) 4. POS terminal obrađuje transakciju. 5. Transakcija uspešno prolazi, na fiskalnoj kasi se štampa račun. 6. Sistem prikazuje poruku o uspešnom naplaćivanju usluga. 7. Sistem ažurira podatke o plaćanjima.
Izuzeci:	4.1.E1 Transakcija je neuspešna zbog prekida signala na POS terminalu <ol style="list-style-type: none"> 1. Sistem prikazuje poruku da je došlo do greške u komunikaciji sa POS terminalom. 2. Sistem daje mogućnost medicinskom radniku da pokuša ponovo da obavi transakciju (3.a) ili da naplati u gotovini (4.a) 3.a. Medicinski radnik ponovo izvršava alternativni tok 4.1 4.a. Medicinski radnik bira naplatu u gotovini i direktno štampanje računa. 4.1.E1 Transakcija je neuspešna zbog nedovoljno sredstava na računu pacijenta <ol style="list-style-type: none"> 3. Sistem prikazuje poruku da je došlo do greške zbog nedovoljnog stanja na računu pacijenta. 4. Sistem daje mogućnost medicinskom radniku da ga preusmeri na naplatu usluga u gotovini.
Prioritet:	Visok
Frekvencija upotrebe:	U proseku 30 puta na dan, a maksimalno 200 puta na nedeljnom nivou.
Poslovna pravila:	...
Druge informacije:	Mora da postoji integracija sistema sa fiskalnom kasom. Mora da postoji integracija POS terminala i sistema.
Pretpostavke:	Fiskalna kasa je u ispravnom stanju. POS terminal je u ispravnom stanju. Cenovnik usluga je ispravno unet.

Slika 4.9 UC-4 Naplaćivanje usluga [Izvor: Marina Damnjanović]

ZADACI ZA VEŽBU

Tekst zadatka za vežbu

ZADATAK 1.

Pregledajte prethodno opisane slučajeve upotrebe kako biste bili sigurni da su koraci tačni. Proverite da li su razmotrene sve varijacije u normalnom toku i da li su svi izuzeci predviđeni. Dajte kritičko mišljenje o tome li su slučajevi upotrebe opisani na način na koji bi kupci smatrali da je razuman. (5 min)

ZADATAK 2.

Napišite samostalno slučajeve korišćenja gde je primarni akter Lekar. (20 min)

ZADATAK 3.

Pokušajte da napišete prethodno opisane slučajeve upotrebe kao korisničku priču ili skup korisničkih priča da biste procenili razlike između tradicionalnog i agilnog pristupa u opisanju korisničkih zahteva. (10 min)

ZADATAK 4.

Nacrtajte dijagram slučajeva korišćenja za sistem za upravljanje radom taksi udruženja koji je opisan u prvoj vežbi. (15 min)

ZADATAK 5.

Nacrtajte dijagram slučajeva korišćenja za ISUM-ov modul E-student. (15 min)

▼ Poglavlje 5

Domaći zadatak

DOMAĆI ZADATAK 6

Tekst domaćeg zadatka

Nacrtajte dijagram slučajeva korišćenja za sistem koji ste dobili za DZ01. Opišite 1 slučaj korišćenja od navedenih, pomoću šablona datog u predavanju. Obavezno uključite alternativne tokove i izuzetke.

Napomene:

Zadatak se rešava opisno i šalje kao .docx fajl. Dijagram pošaljite kao .oom fajl iz PowerDesigner alata.

Rešenje zadatka pošaljite na mejl adresu predmetnog asistenta. Rok za izradu je definisan Plan i programom predmeta.

▼ Poglavlje 6

Projektni zadatak

ZADATAK ZA RAD NA PROJEKTU

Tekst zadatka za rad na projektu

Najpre, nacrtajte **dijagram slučajeva korišćenja** za vaš sistem.

Zatim, pomoću šablona datog u uzorku dokumenta Slučajevi korišćenja, kreirajte jedan takav dokument i napravite listu slučajeva korišćenja za vaš trenutni projekat. Opišite sve identifikovane slučajeve korišćenja, takođe prema datom šablonu. Kad god je moguće, uključite alternativne tokove i izuzetke. Polje koje se odnosi na poslovne zahteve za sada ostavite prazno. Oformite **dokument Slučajevi korišćenja**.

▼ Poglavlje 7

Zaključak

ZAKLJUČAK

1. Slučajevi korišćenja i priče korišćenja služe da se utvrde šta korisnici žele da sistem rad. Korisne su za poslovne aplikacije, razvoj veb sajtova, i interaktivnih sistema.
2. Slučaj korišćenja opisuje sekvencu interakcija između sistema i spoljnih aktera. Priča korisnika (user story) je kratak opis svojstva ispričan iz perspektive osobe koja želi novu sposobnost.
3. Na osnovu specifikacije slučaja korišćenja, analitičar onda dobija funkcionalne zahteve. Priče korisnika se detaljišu i dele na manje, i vode dobijanju testova prihvatanja.
4. Priče korisnika nude prednost u konciznosti i jednostavnosti, a slučajevi korišćenja daju učesnicima strukturu i kontekst, što pričama korisnika nedostaje.
5. Slučajevi korišćenja obezbeđuju apstraktnu vizualno predstavljanje zahteva korisnika. Scenario je opis jednog primera upotrebe sistema. Jedan slučaj korišćenja ima kolekciju scenarija.
6. Preduslovi definišu šta mora da bude ispunjeno da bi sistem počeo da izvršava slučaj korišćenja. Postuslovi opisuju stanje sistema posle uspešnog izvršenja slučaja korišćenja.
7. Jedan scenario predstavlja jedan normalan tok događaja za slučaj korišćenja. Može imati i alternativne tokove (scenarije). Izuzeci opisuju uslove za prekid tokova zbog grešaka.
8. Veza extend može povremeno da proširi osnovni slučaj korišćenja. Veza include omogućuje da se jedan slučaj korišćenja koristi u više slučajeva korišćenja.
9. Na radionicama se utvrđuju svi elementi slučajeva korišćenja. Ili se koriste lepljive beleške, ili se projektuje uzorak specifikacije slučajeva korišćenja i kolektivno popunjavaju.
10. Iz dobijenih slučajeva korišćenja, biznis analitičar definiše funkcionalne zahteve. Analitičar takođe generiše modele analize, kao što je dijagram stanja
11. Slučajevi korišćenja otkrivaju važne objekte domena i njihove međusobne odnose. Projektanti OO sistema mogu da mapiraju slučajeve korišćenja u modela objekata i sekvencijalne dijagrame

REFERENCE

Nastavi materijal pripremljen za studente se pravi s namerom da im omogući brži i skraćeni uvid u program lekcije, a na bazi jedne ili više referentnih udžbenika i drugih izvora . Nastavni materijal nije zamena za ove udžbenike, koje treba koristiti ako student želi da se detaljnije upozna sa nastavnim materijom. Očekuje se od studenta da poseduje bar jedan od navedenih udžbenika u Planu i programu predmeta.

Ova lekcija je urađena na bazi teksta datom u **poglavlju 8** knjige: **Karl Wiegers, Joy Beaty, Software Requirements, 3rd ed., Microsoft, 2013**. Za detaljnije proučavanje i primere, studentima se preporučuje da pročitaju ovo poglavlje. Manji uticaj na sadržaj lekcije imaju ostale reference navedene u Planu i programu predmeta,