

DIZAJN SISTEMA

Dogovor za izlazak

Tim 3

Sarajevo, 2017.

Sadržaj

Historijat revizije dokumenta	2
Uvod	3
2.1. Svrha dokumenta	3
2.2. Opseg dokumenta	3
2.3. Standardi dokumentovanja	3
2.4. Vanjske reference	4
Use Case dijagrami	5
3.1. Kreiranje korisničkog računa	5
3.2. Prijava na sistem	6
3.3. Kreiranje eventa	7
3.4. Modifikacija eventa	8
3.5. Upravljanje prijavljenim korisnicima i eventima	9
3.6. Generalni Use Case sistema	10
Activity dijagrami	11
4.1 Prijava na sistem	11
4.2 Odjava sa sistema	12
4.3. Kreiranje korisničkog računa	13
4.4. Kreiranje eventa	14
4.5. Pregled korisnika koji dolaze na pojedini event	15
4.6. Slanje pozivnica za event	16
4.7. Postavljanje vremenskog roka za glasanje detalja svakog eventa	17
4.8. Dodavanje detalja na event	18
4.9. Dodavanje opcije na detalj eventa	19
4.10. Glasanje za opcije	20
4.11. Pretraga korisnika po imenu i/ili prezimenu	21
4.12. Grupni chat	22
4.13. Promjena šifre	23
4.14. Brisanje eventa	24
ER dijagram	25
Arhitektura sistema	26

1. Historijat revizije dokumenta

Datum	Verzija	Autor	Komentar
11.4.2017.	v1.0	Trusted Partners	Inicijalna verzija dokumenta.

2. Uvod

Ovaj dokument služi da bi se sistem predstavio na standardiziran i organiziran način. Pruža detaljan, višeslojni pogled na arhitekturu i funkcionalnosti sistema.

Za opis sistema korišteni su sljedeći prikazi:

- dijagrami slučajeva upotrebe (use case dijagrami)
- dijagrami aktivnosti (activity dijagrami)
- ER dijagrami (entity relationship dijagrami)

2.1. Svrha dokumenta

Dokument služi za bolje razumijevanje sistema kao cjeline, njegovih komponenti, funkcionalnosti i međuzavisnosti između njih. Kao vodič za izradu dizajna sistema je SRS (System Requirements Specification), dokument izrađen na osnovu zahtjeva klijenta. Dokument je rezultat rada u fazi dizajna i kao takav služi kao polazna tačka za fazu implementacije.

2.2. Opseg dokumenta

Dijagrami slučajeva upotrebe - reprezentacija korisnikove interakcije sa sistemom. Prikazuje vezu između korisnika i različitih slučajeva upotrebe u kojima korisnik učestvuje.

Dijagrami aktivnosti - namijenjeni su modeliranju dinamičkih aspekata (ponašanja) sistema. Koristi se za prikaz tokova u sistemu.

ER dijagrami - nude detaljan logički prikaz entiteta veza i prikaz entiteta, veza i podataka

2.3. Standardi dokumentovanja

Prilikom izrade ovog dokumenta korišteni su sljedeći softverski alati:

- Libre Office Writer
- Google Docs
- Visual Paradigm for UML

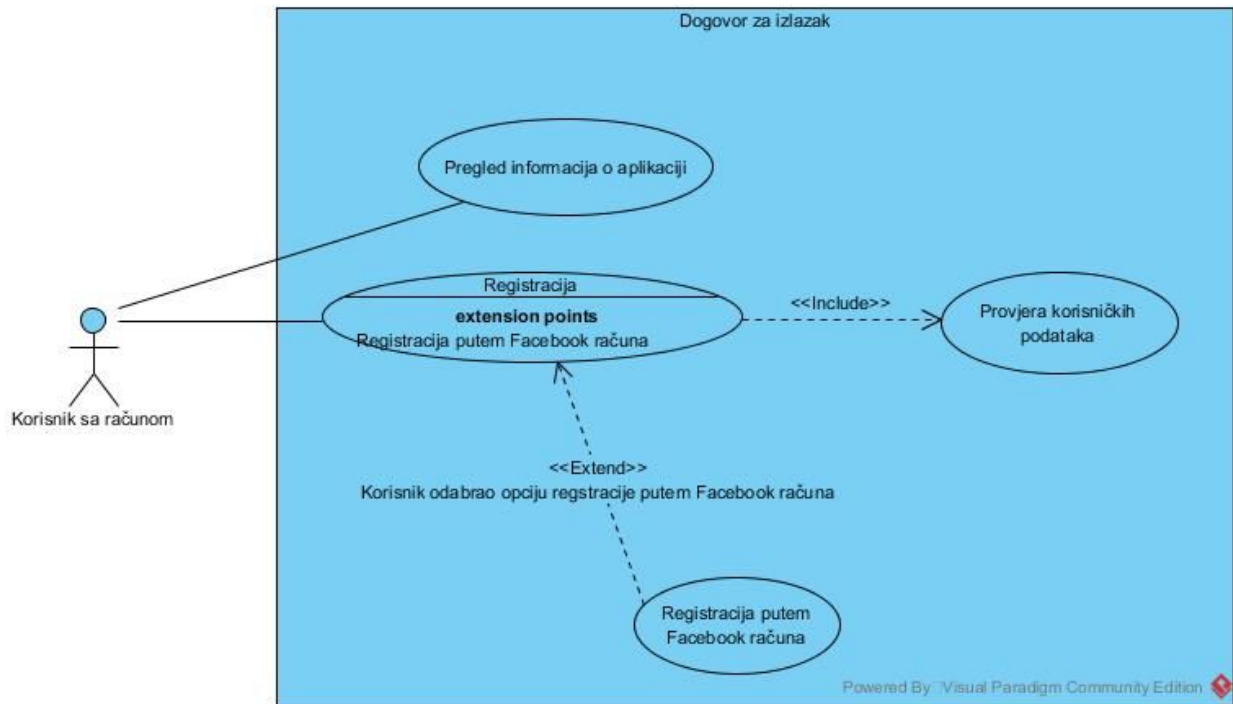
Korišten je font Arial 11.

2.4 Vanjske reference

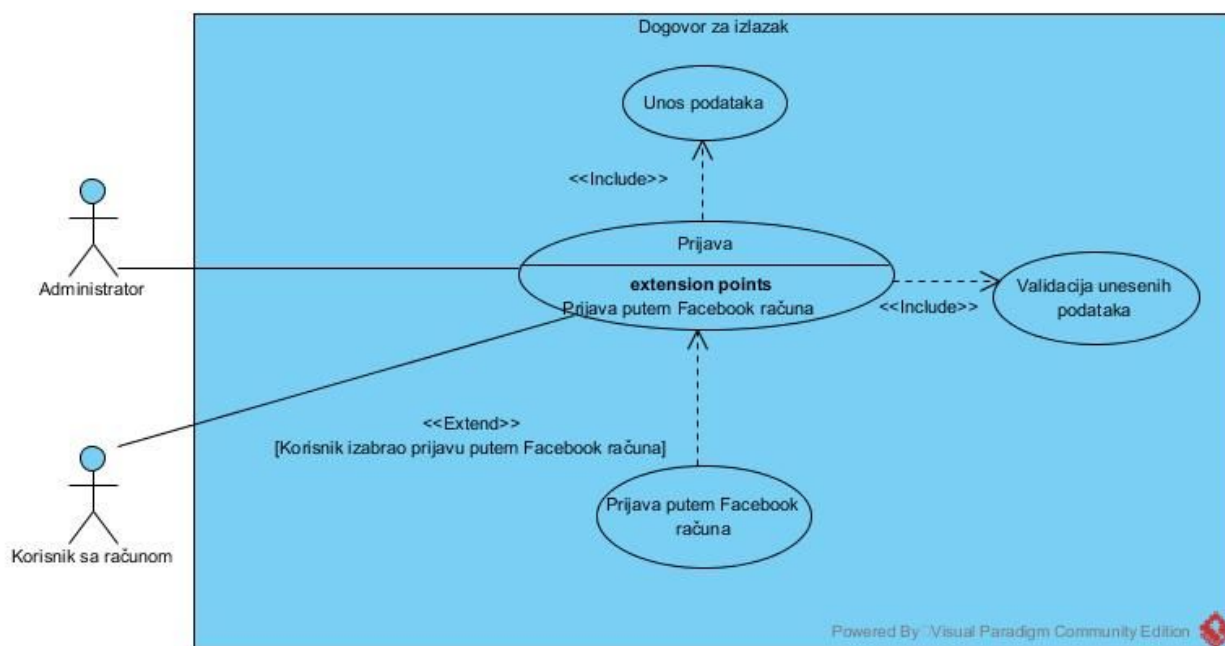
- [1] Software Requirement Specification - SRS
- [2] Predavanja iz predmeta Softverski Inženjering

3. Use Case dijagrami

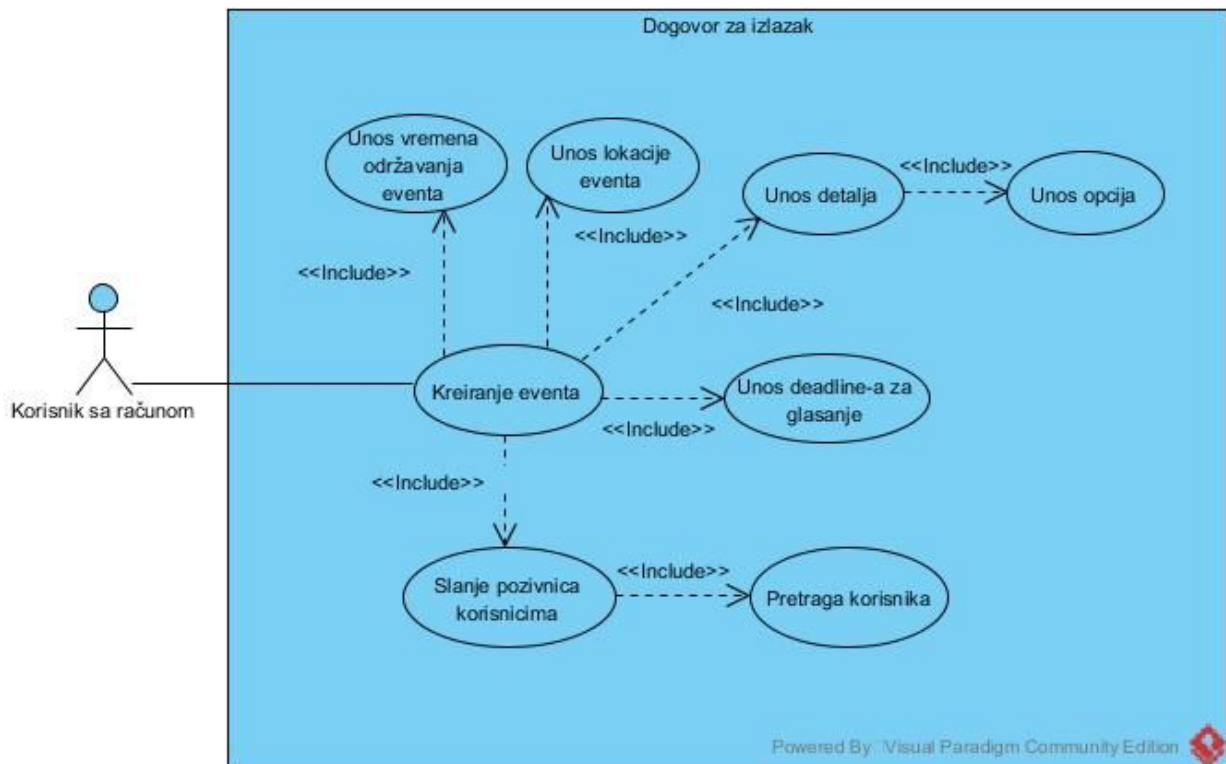
3.1. Kreiranje korisničkog računa



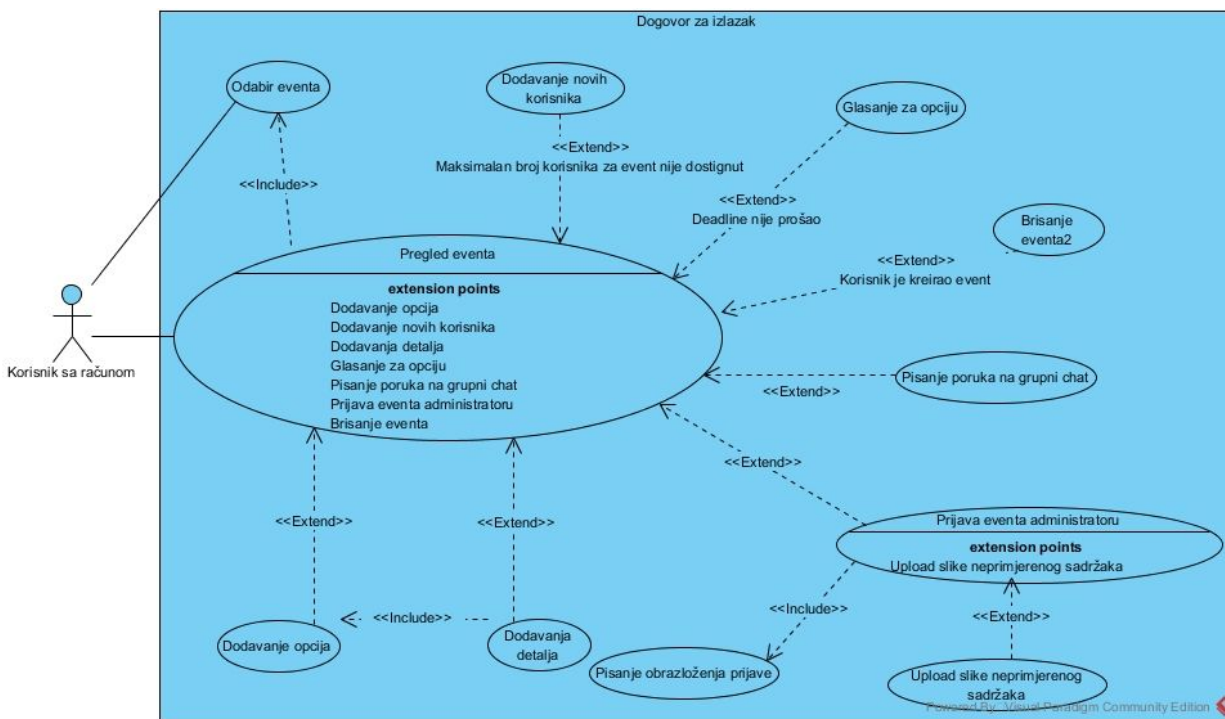
3.2. Prijava na sistem



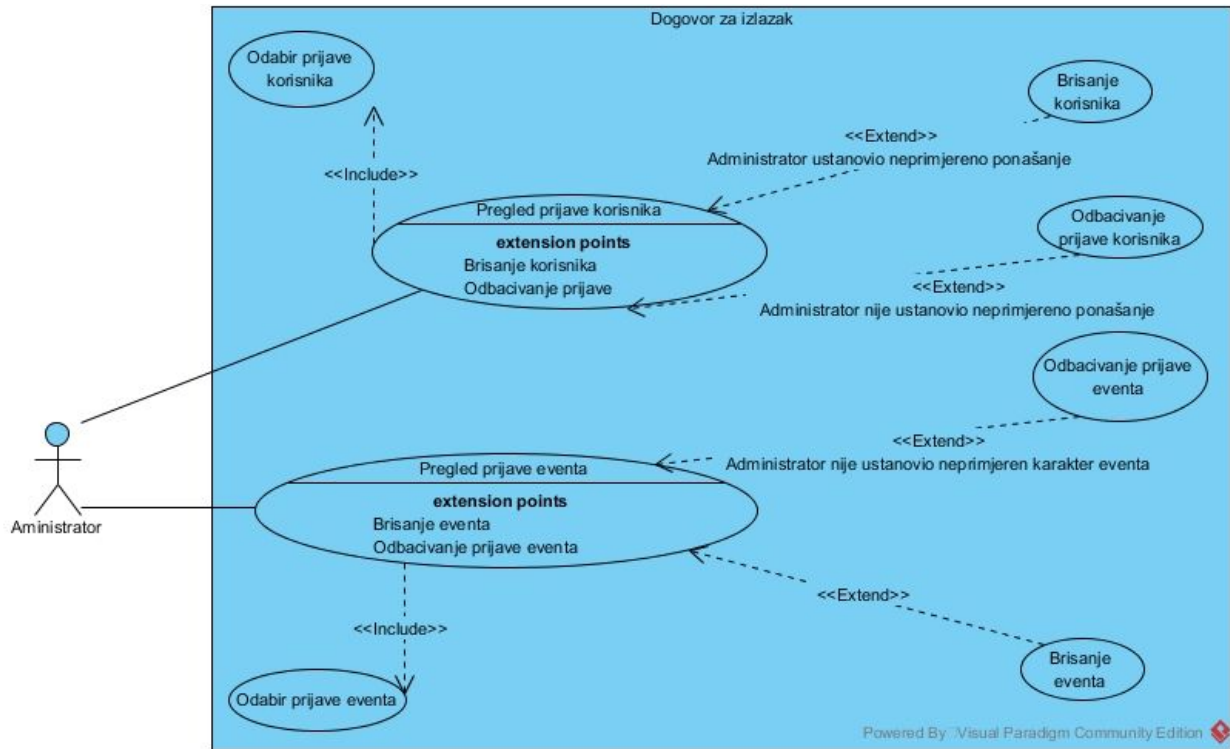
3.3. Kreiranje eventa



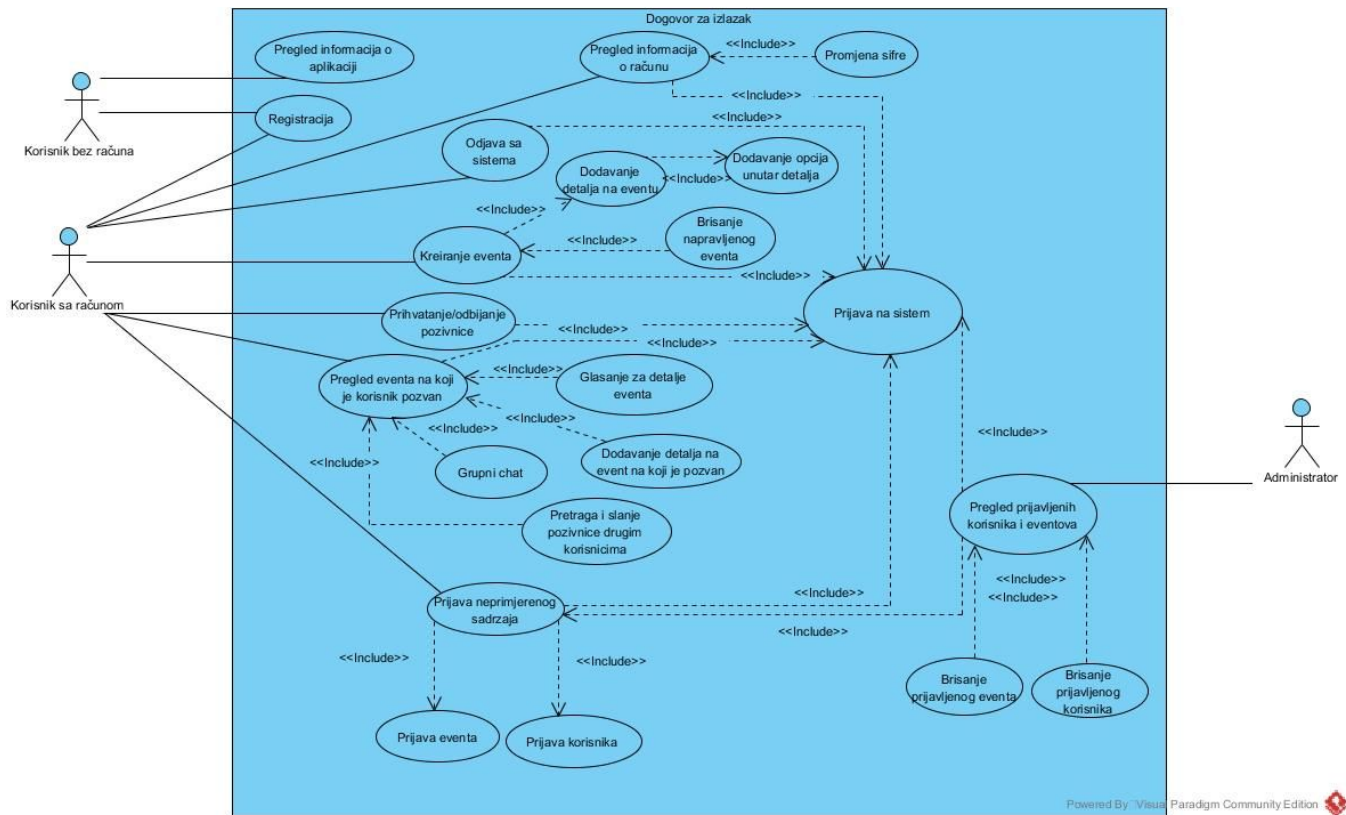
3.4. Modifikacija eventa



3.5. Upravljanje prijavljenim korisnicima i eventima

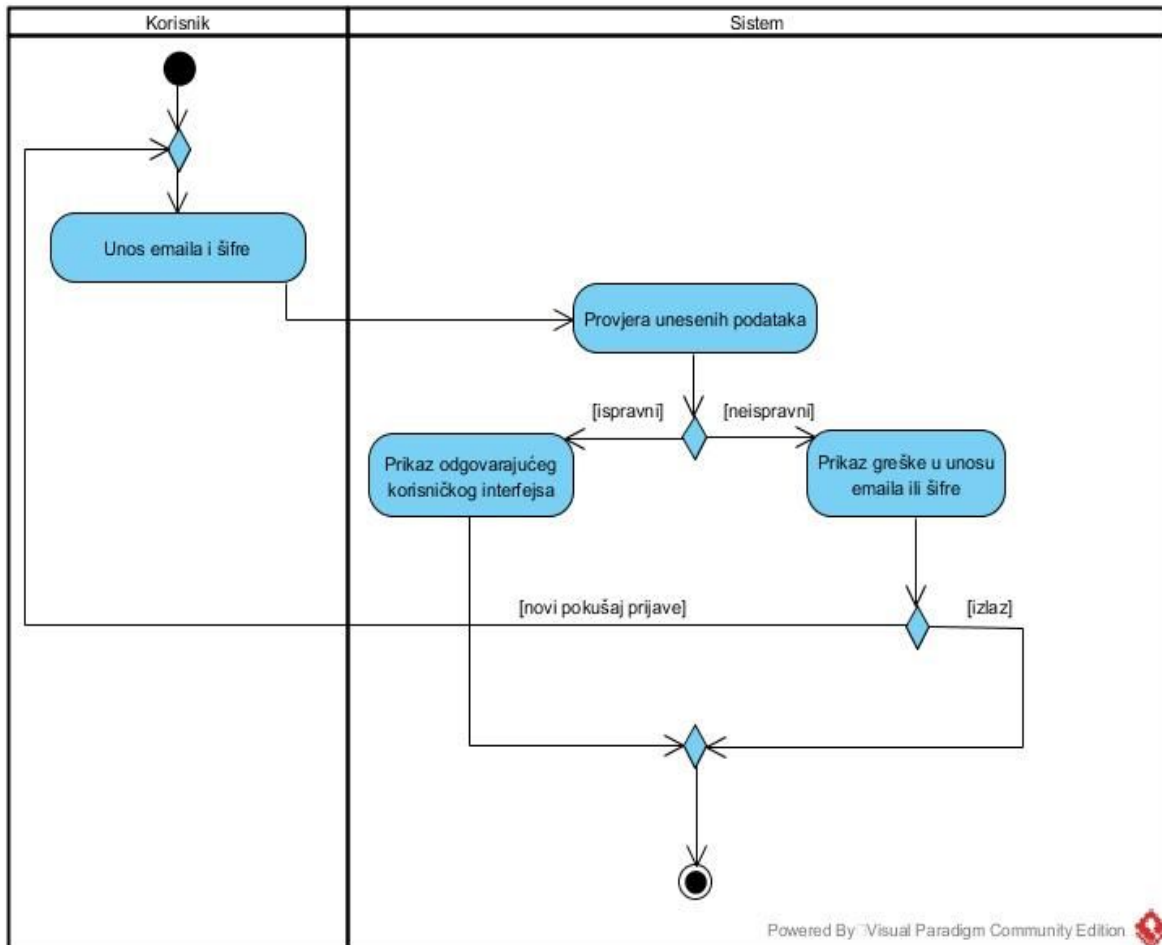


3.6. Generalni Use Case sistema

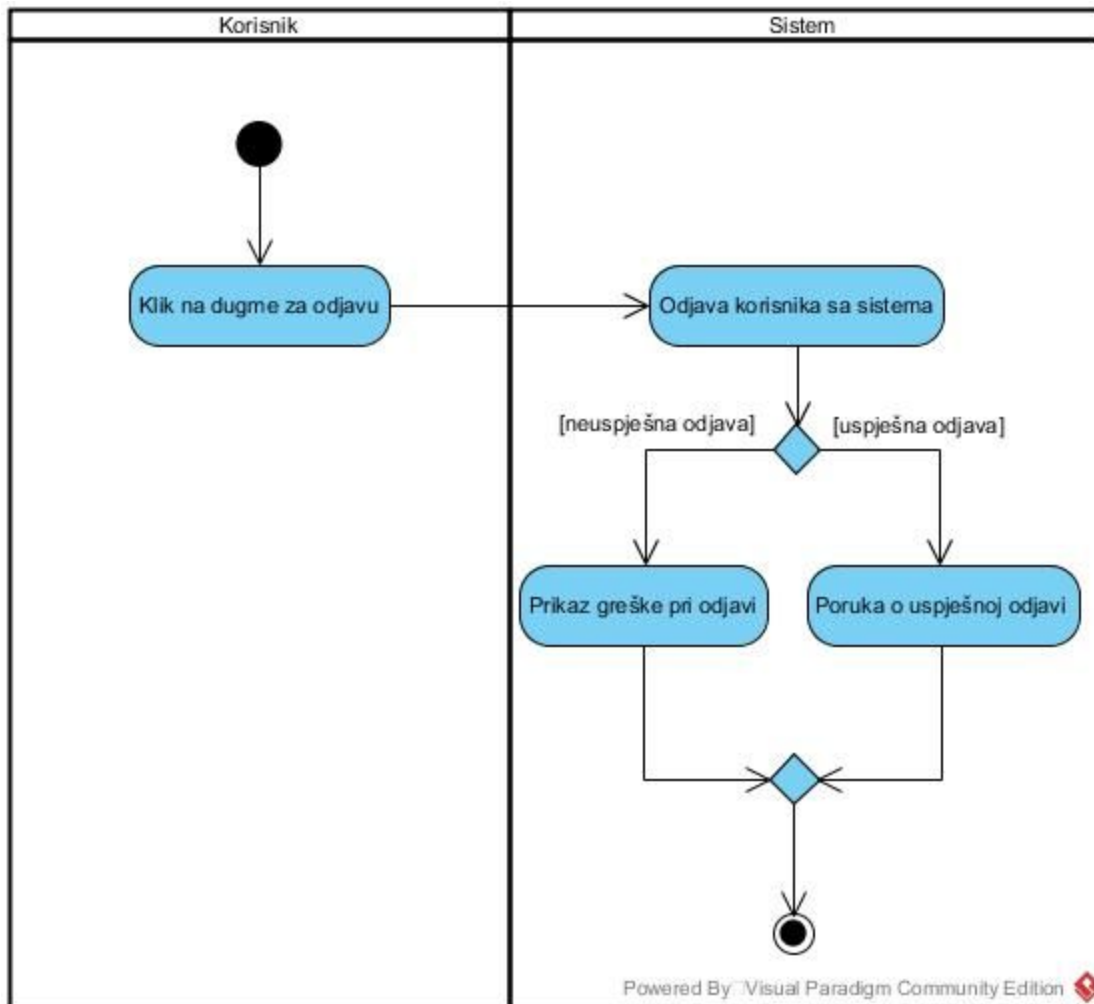


4. Activity dijagrami

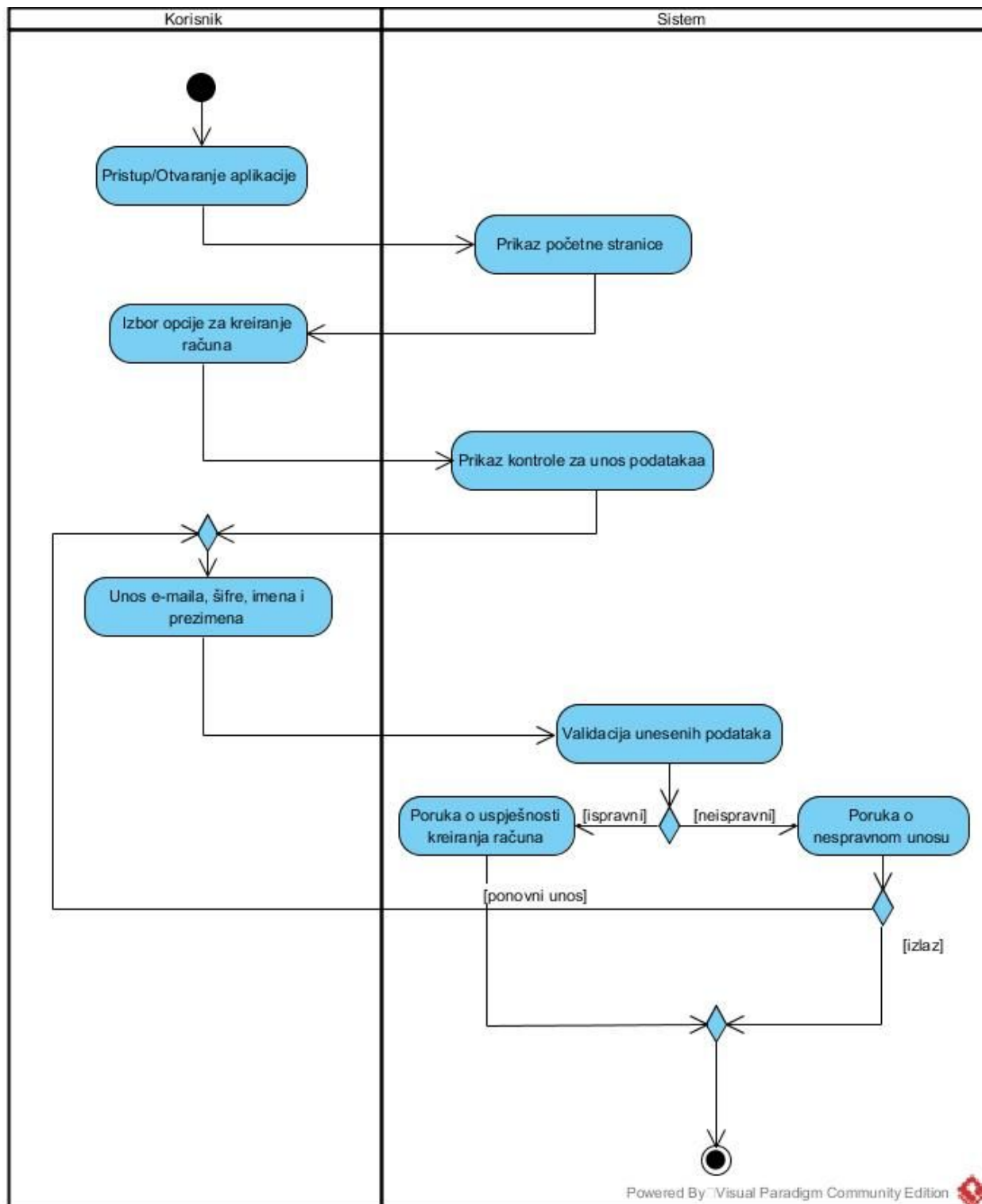
4.1 Prijava na sistem



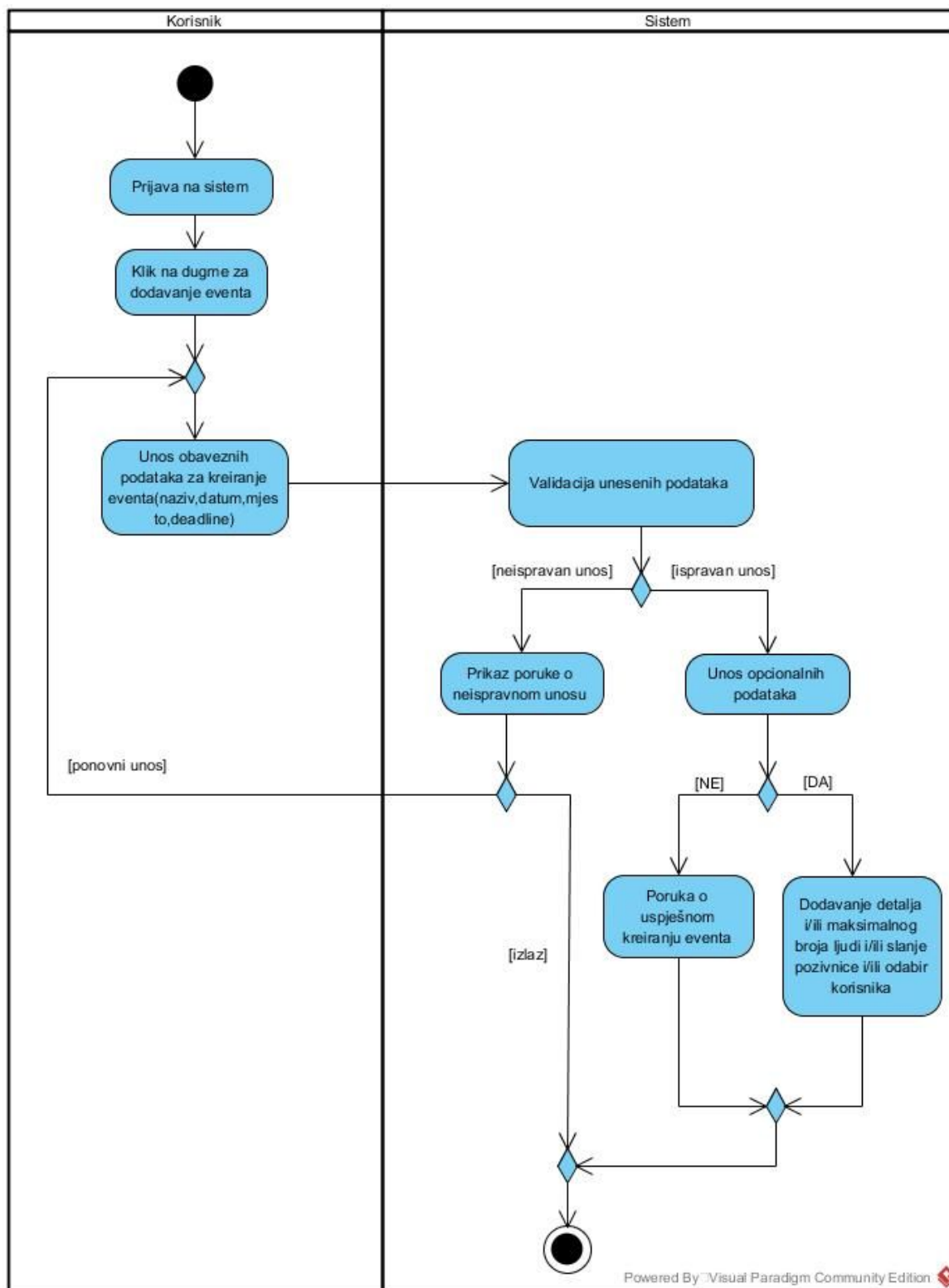
4.2 Odjava sa sistema



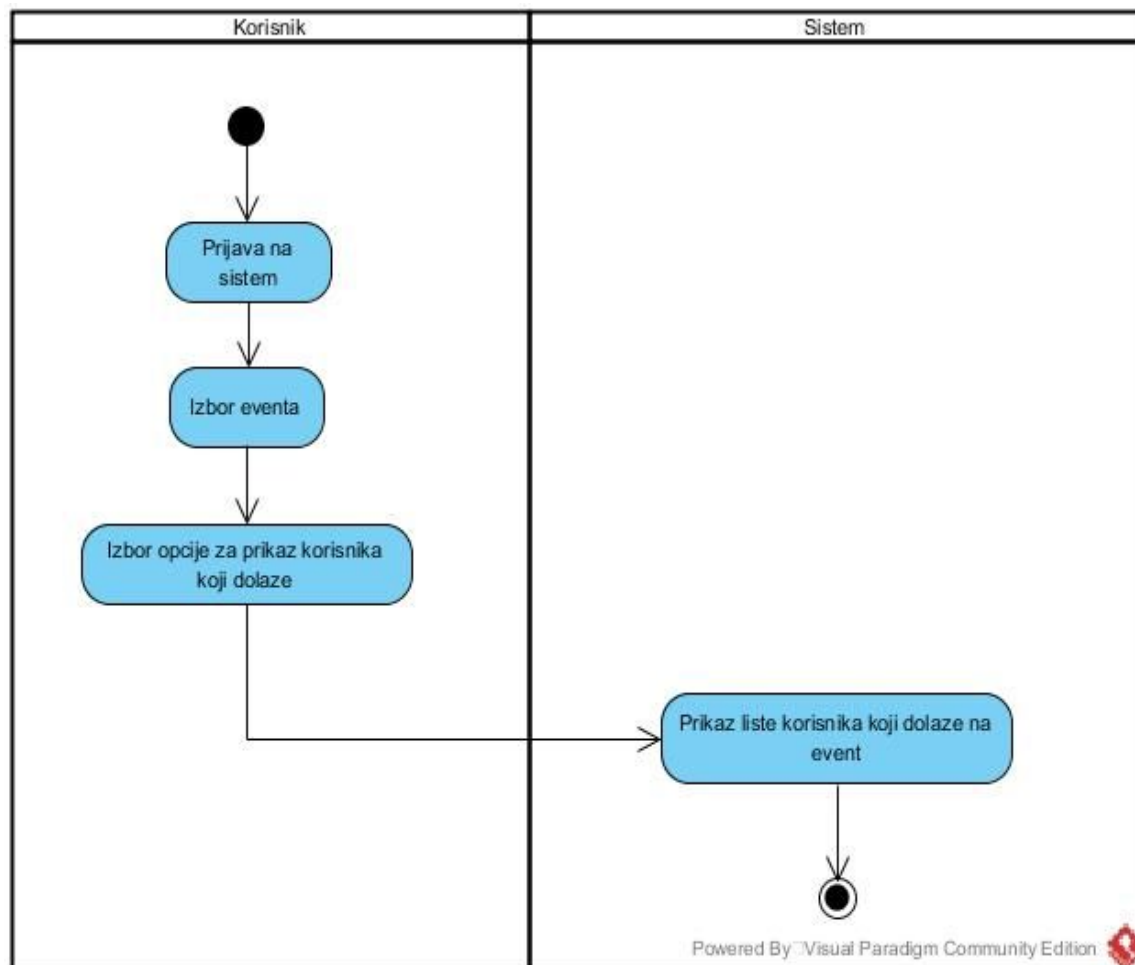
4.3. Kreiranje korisničkog računa



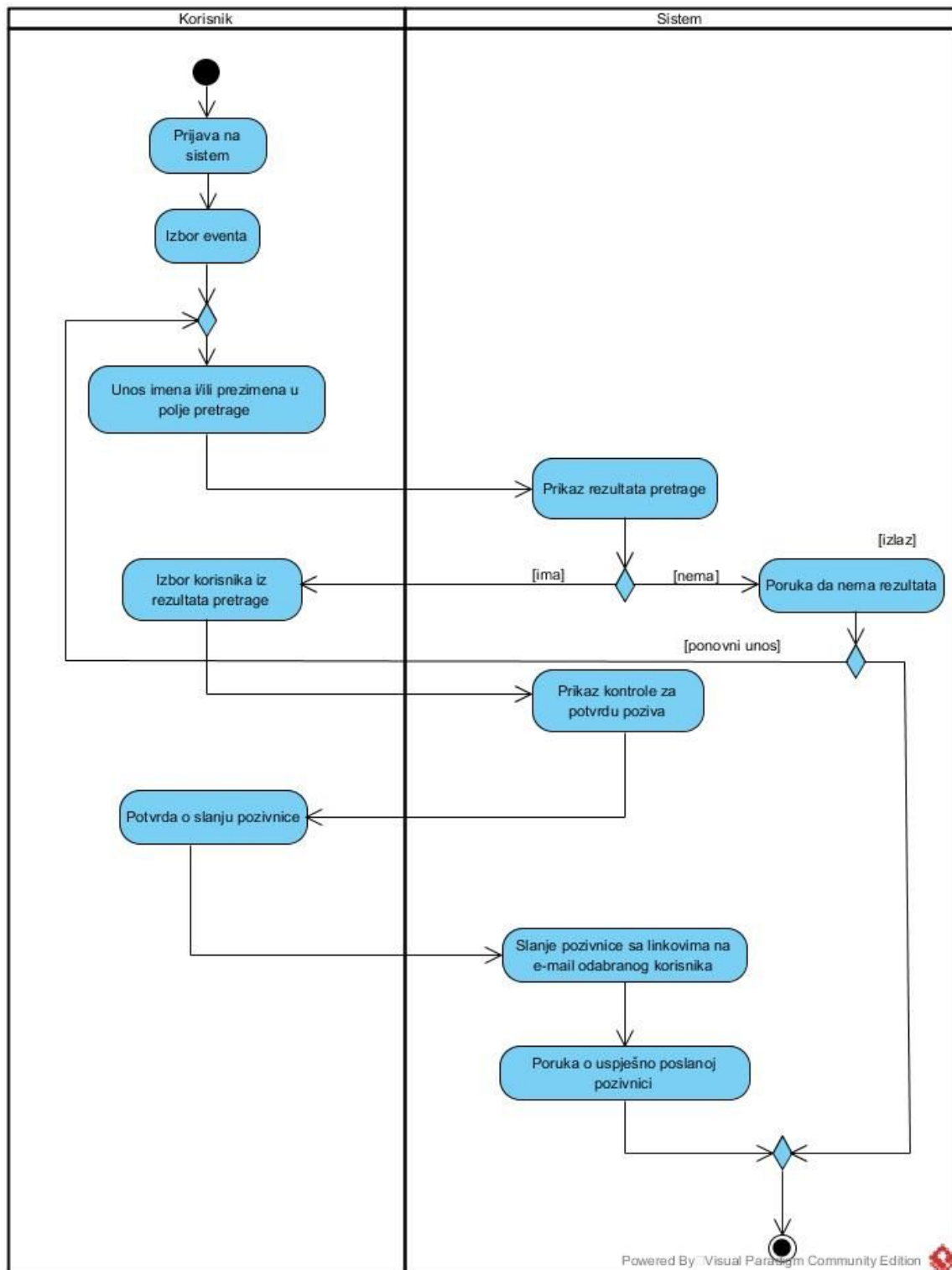
4.4. Kreiranje eventa



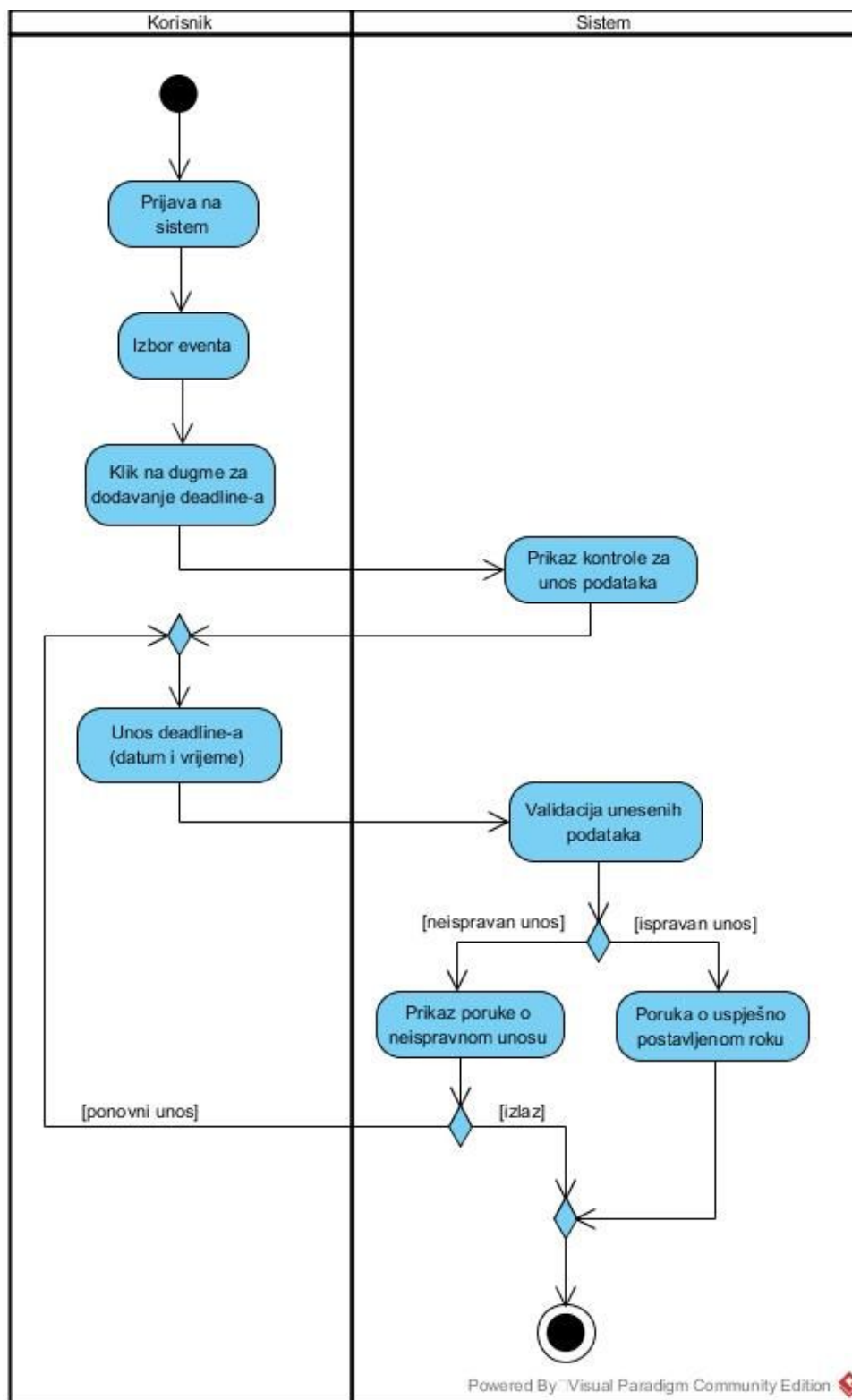
4.5. Pregled korisnika koji dolaze na pojedini event



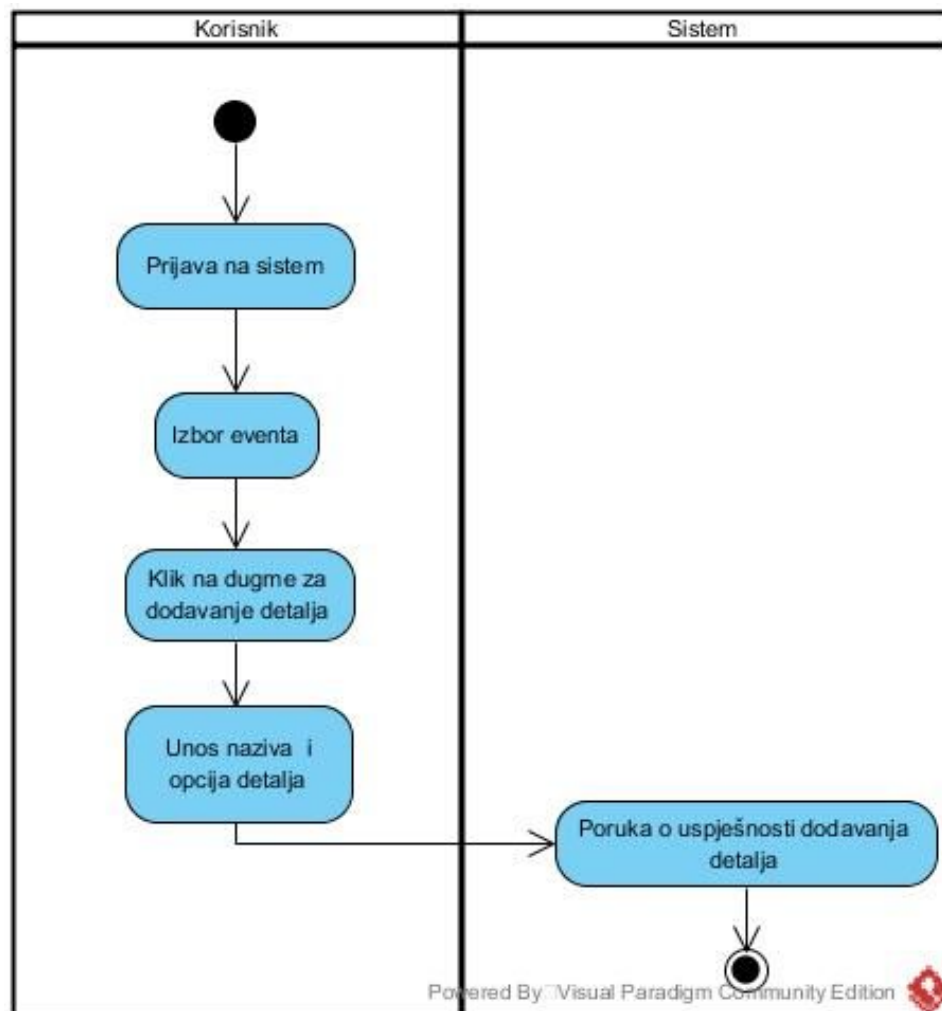
4.6. Slanje pozivnica za event



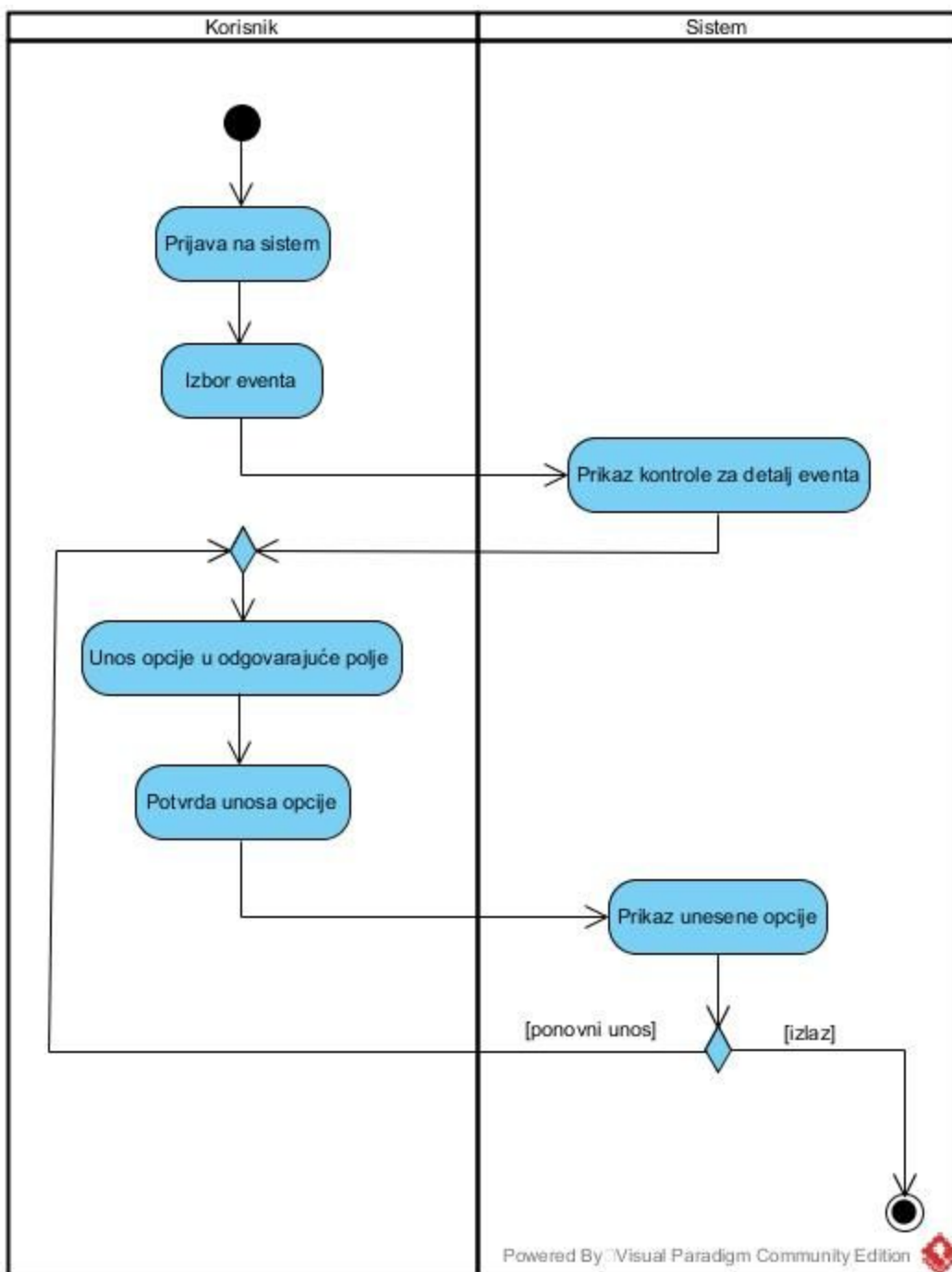
4.7. Postavljanje vremenskog roka za glasanje detalja svakog eventa



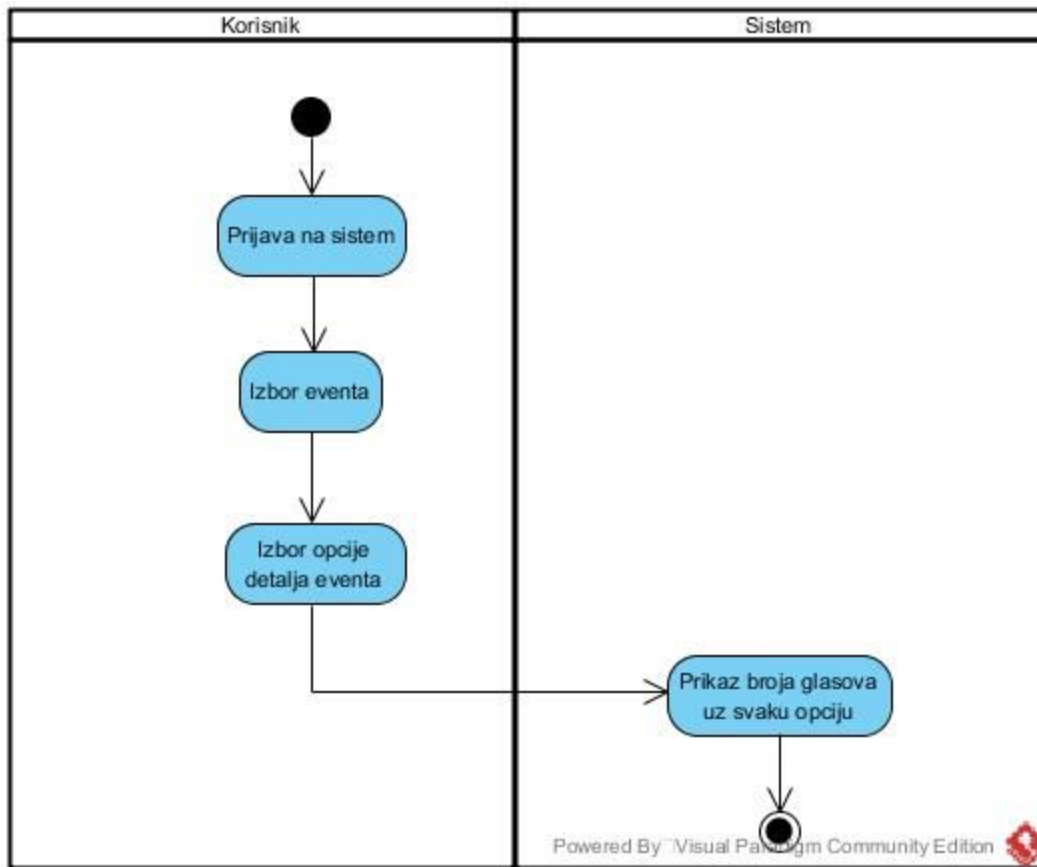
4.8. Dodavanje detalja na event



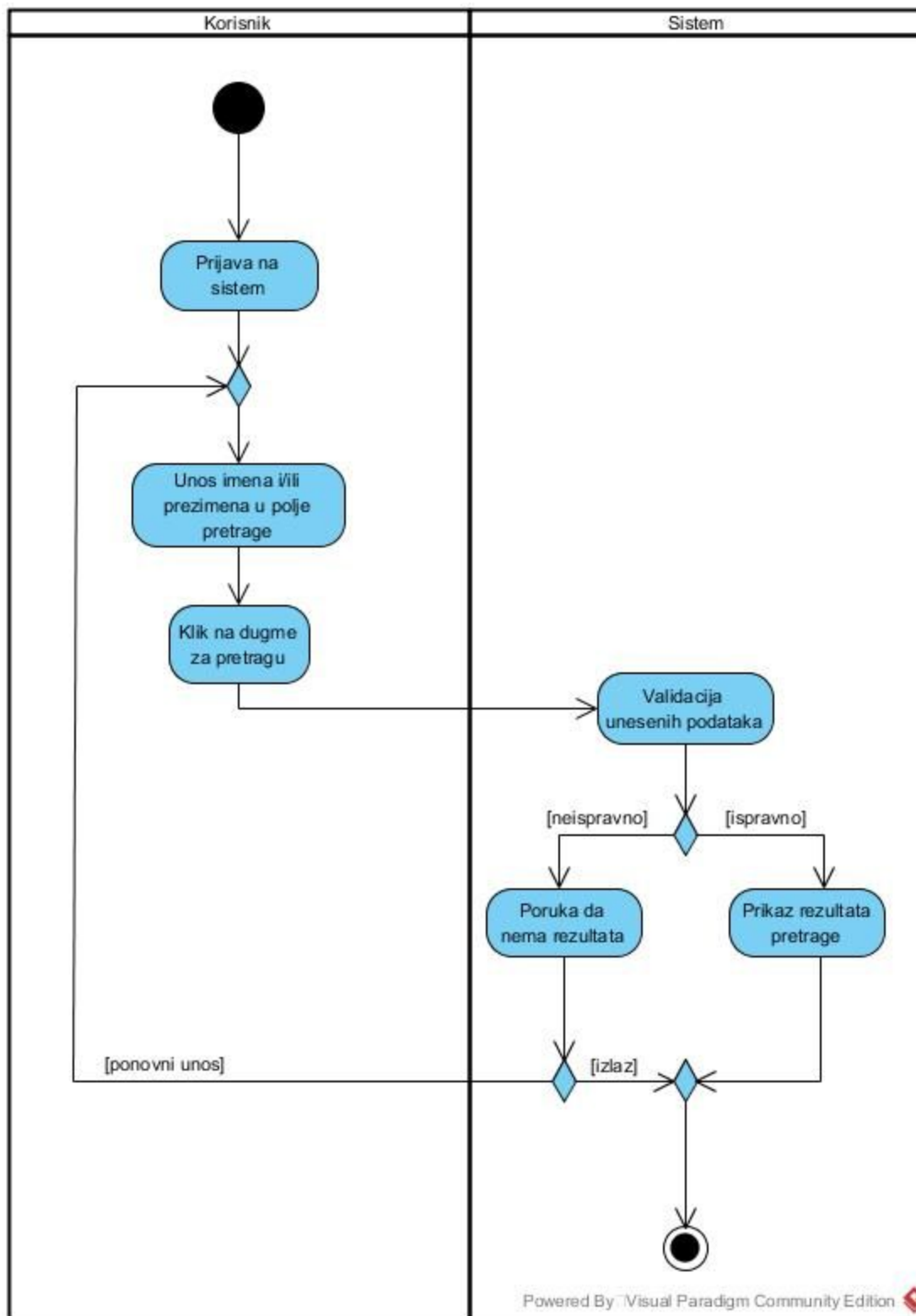
4.9. Dodavanje opcije na detalj eventa



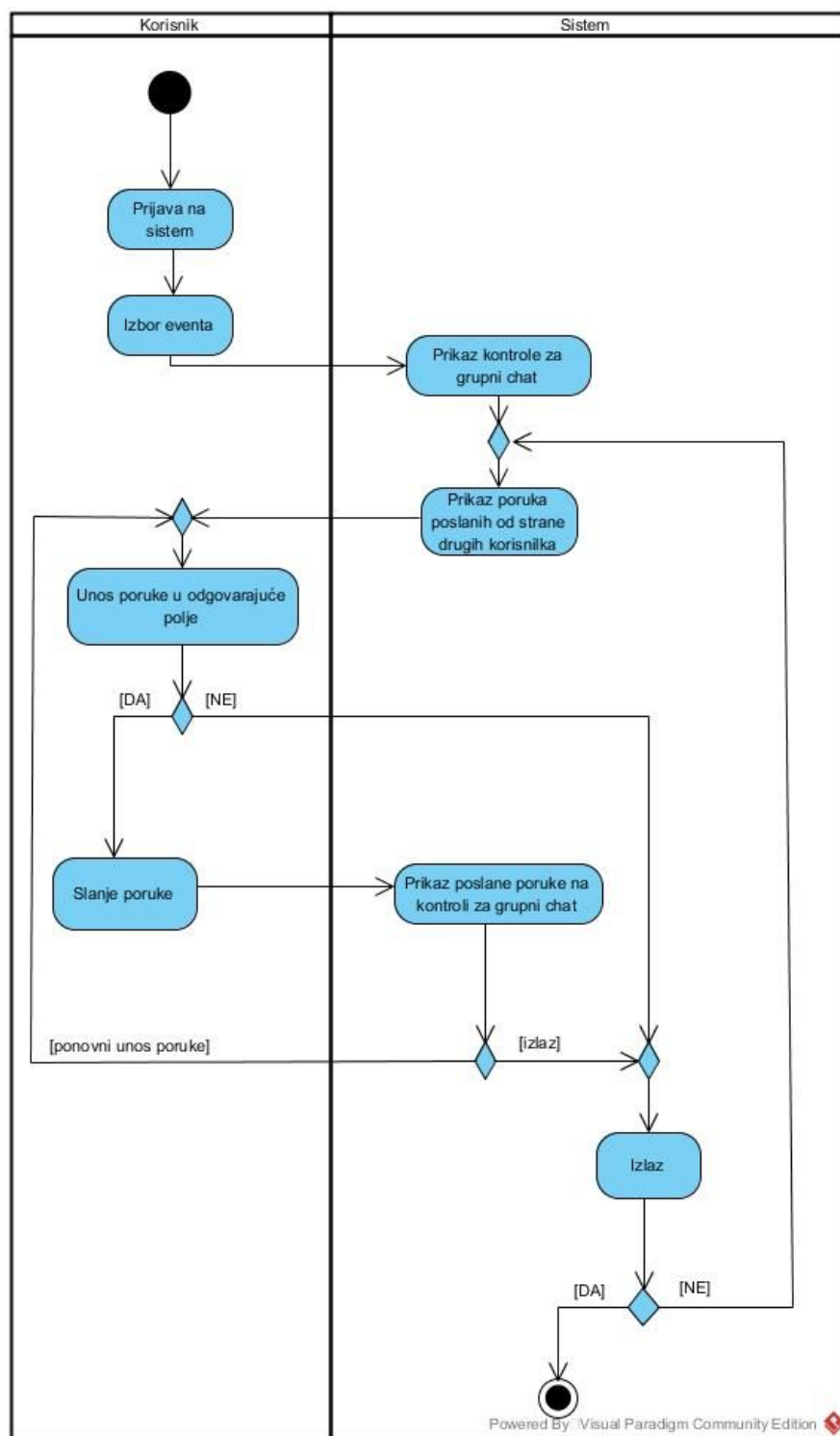
4.10. Glasanje za opcije



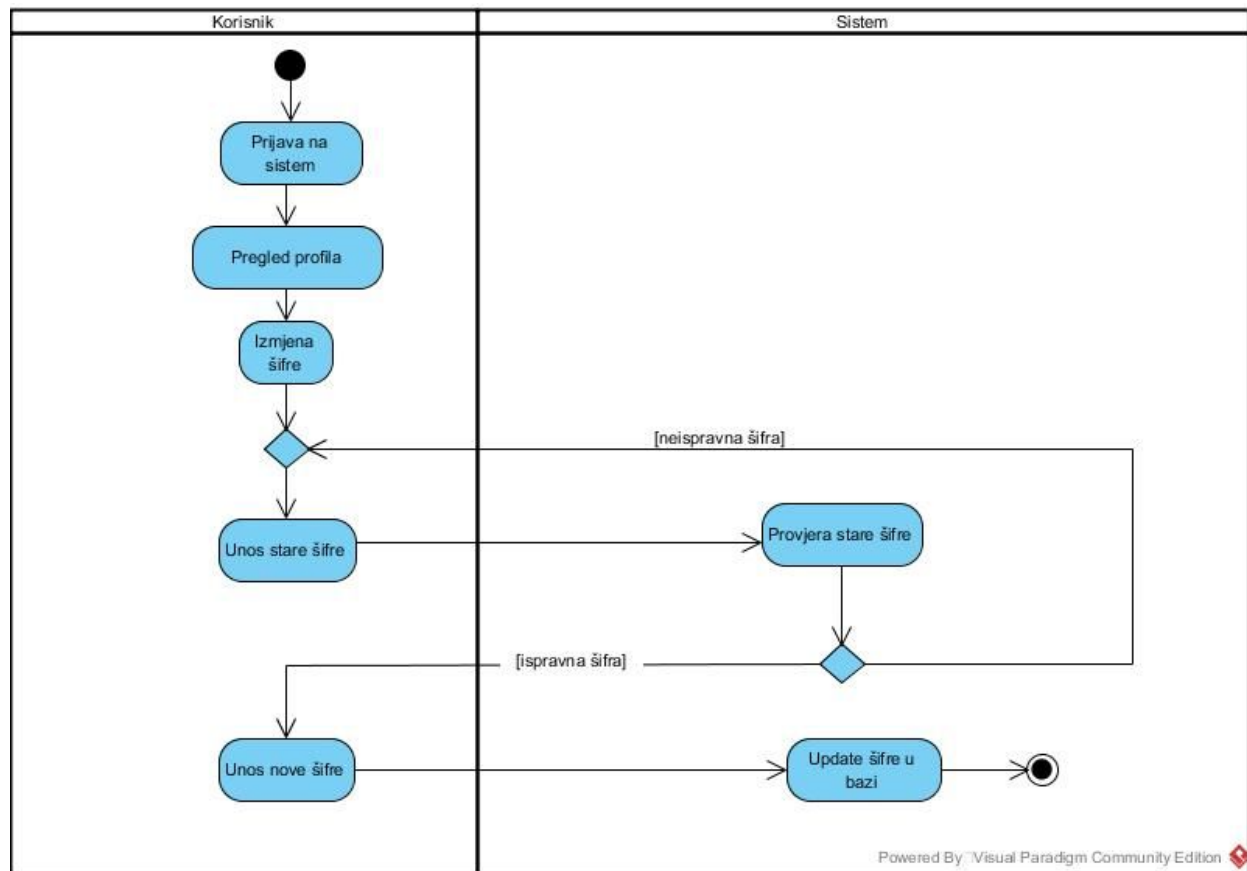
4.11. Pretraga korisnika po imenu i/ili prezimenu



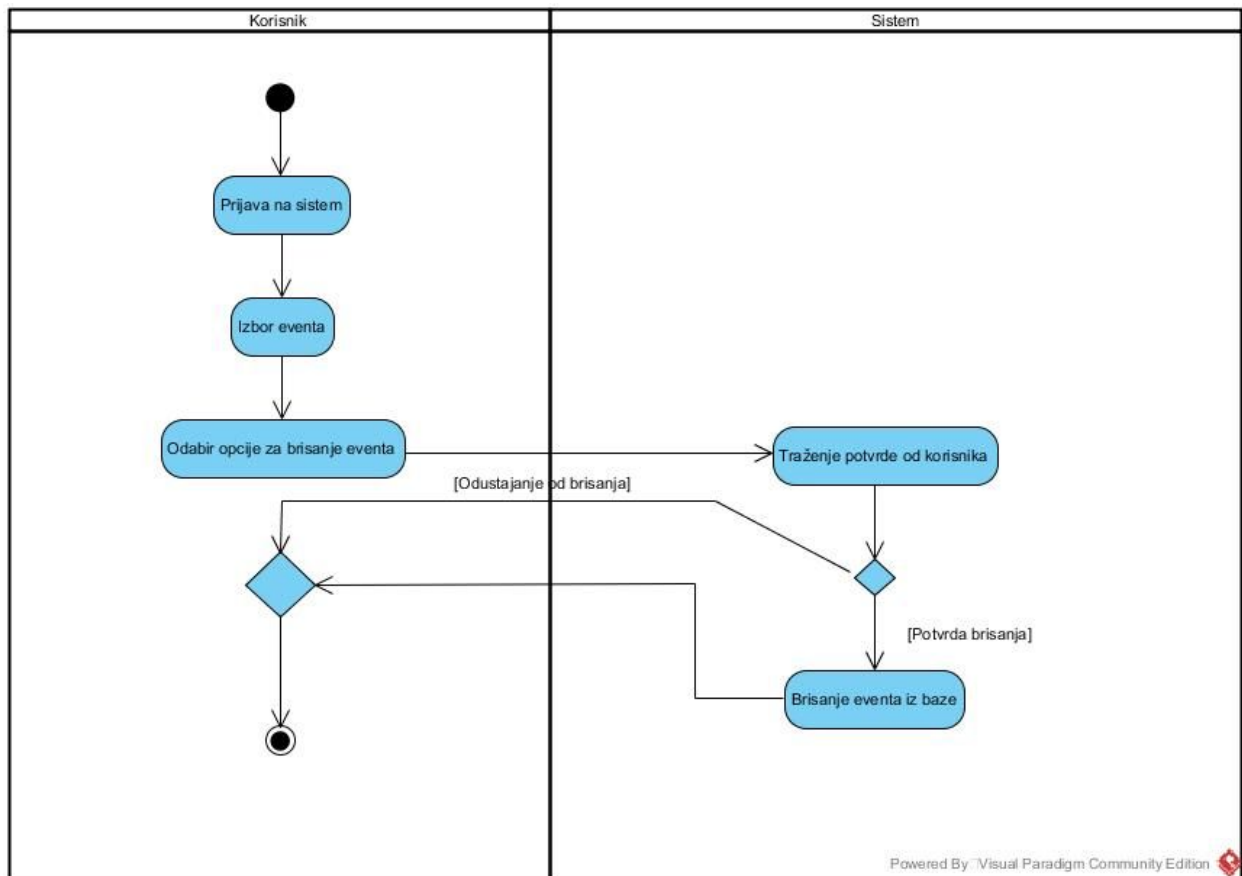
4.12. Grupni chat



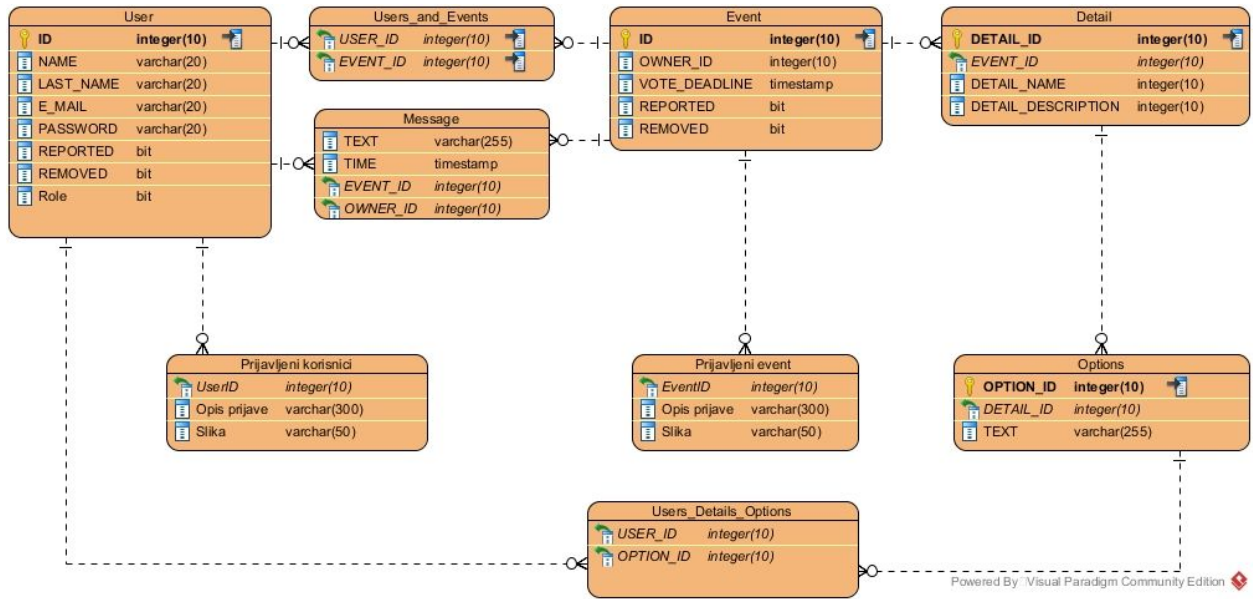
4.13. Promjena šifre



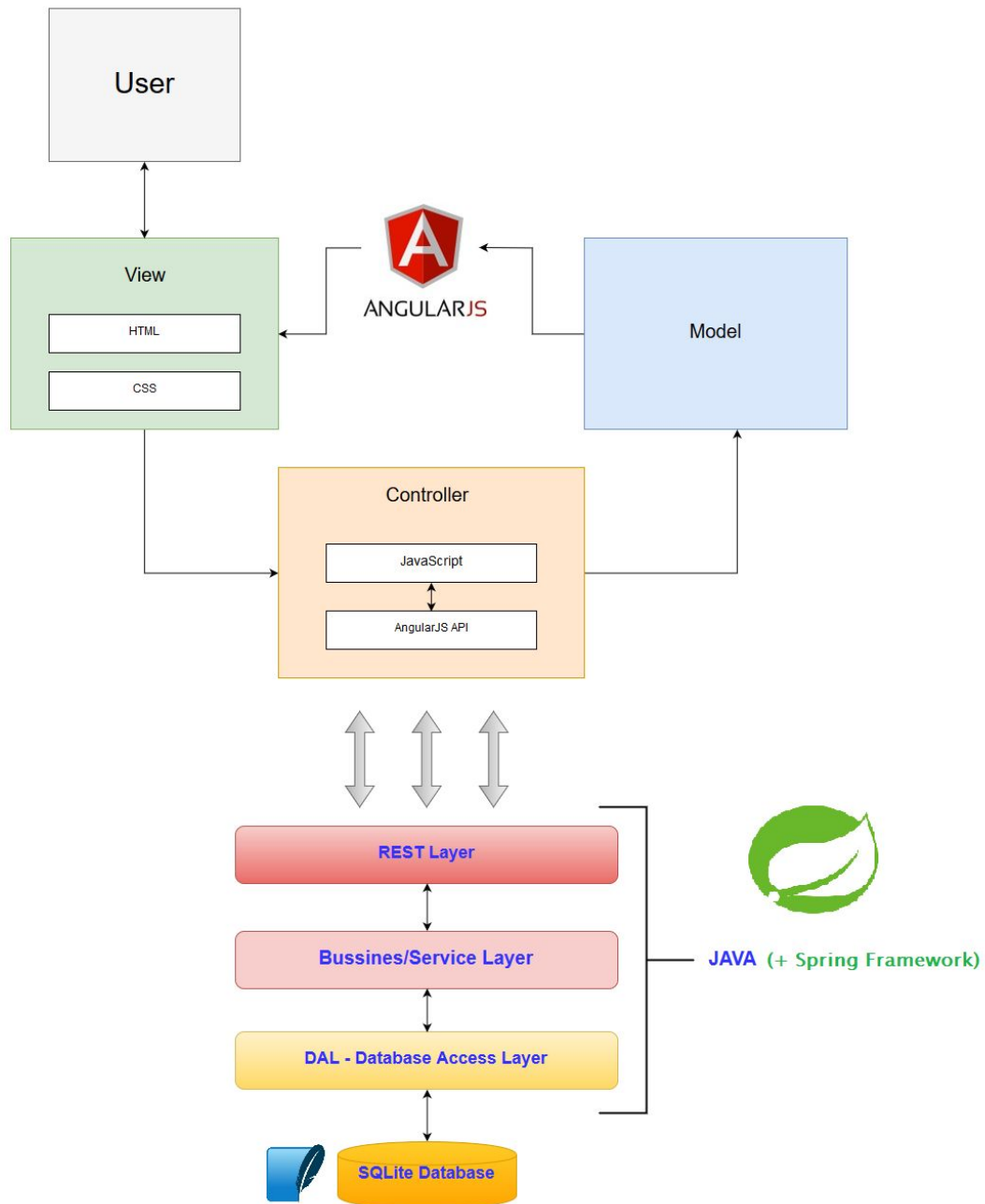
4.14. Brisanje eventa



5.ER dijagram



6.Arhitektura sistema



Za pohranjivanje podataka odabrana je SQLite baza. SQLite baza odlikuje se velikom brzinom odziva i efikasnošću, što je potrebno za dobro korisničko iskustvo prilikom rada unutar aplikacije.

Na serverskoj strani odabran je programski jezik JAVA u kombinaciji sa Spring frameworkom. Spring framework omogućit će jednostavno izlaganje endpointa čija je uloga odgovaranje na pozive servisa koji dolaze sa klijentske strane prilikom same upotrebe aplikacije. Nakon što zahtjev dođe do REST sloja, isti će biti proslijeđen sloju biznis logike koja će dalje rukovati dobavljanjem, obrađivanjem i vraćanjem potrebnih podataka klijentu.

Na klijentskoj strani koristi se AngularJS - JavaScript MVC Framework koji će rukovati prikazivanjem sadržaja korisniku. AngularJS će se pobrinuti za asinhrono učitavanje podataka (bez blokiranja UI-a) i prikazivanje sadržaja pomoću ugrađene funkcionalnosti data-bindinga. Također će vršiti rutiranje unutar aplikacije prilikom navigiranja kroz podstranice, validaciju formi unutar korisničkog interfejsa, te omogućiti ponovno iskorištavanje UI komponenti na više mjesta u aplikaciji.