

Dijagrami aktivnosti i sekvencijski dijagrami

Programsko inženjerstvo

2025/2026

Sadržaj predavanja

01

Dijagram aktivnosti
Activity Diagram

02

Sekvencijski dijagram
Sequence Diagram

03

Primjeri iz prakse
Real-world examples



IEEE istraživanje: Timovi koji koriste UML dijagrame imaju 40% manje grešaka u implementaciji

Koriste: 78% Fortune 500 | 85% tech tvrtke | 92% banke

Pregled UML dijagrama

Strukturni dijagrami

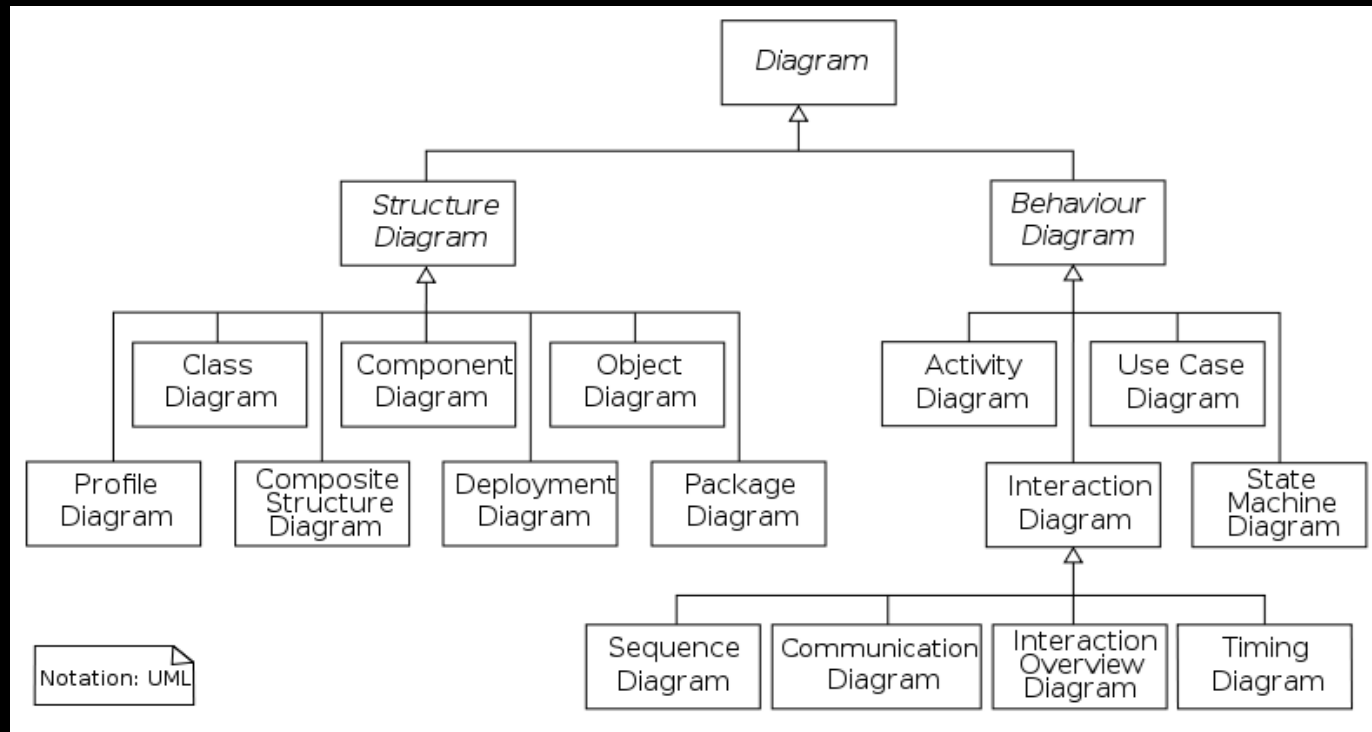
- Class Diagram
- Object Diagram
- Component Diagram
- Deployment Diagram
- Package Diagram

Ponašajni dijagrami

- Use Case Diagram
- Activity Diagram 🔥 DANAS!
- State Machine Diagram
- Sequence Diagram 🔥 DANAS!
- Communication Diagram

💡 Activity i Sequence su najčešće korišteni za dokumentaciju API-ja i mikroservisa

Pregled UML dijagrama



Zašto koristiti UML dijagrame?

Dokumentacija

Vizualna dokumentacija razumljiva svima

Otkrivanje grešaka

Ranije otkrivanje dizajnerskih grešaka

Planiranje

Bolji pregled arhitekture sustava

Onboarding

Brže uključivanje novih članova tima

 40% manje grešaka | 3x brži onboarding | 60% manje redesigna

Dio 1

Dijagram aktivnosti

Activity Diagram

Modeliranje poslovnih procesa i algoritama

Što je dijagram aktivnosti?

Ponašajni UML dijagram koji prikazuje tijek izvođenja aktivnosti poslovnog procesa ili tok objekata (podataka).

Naglasak je na:

- Slijed i redoslijed akcija
- Uvjeti koji upravljaju tijekom izvođenja
- Paralelno izvođenje

 PRIMJENA: Poslovni procesi, use case tijekovi, algoritmi

Real-world primjer

Uber narudžba vožnje

1. Korisnik otvara app

2. Unos odredišta

3. Izbor tipa vozila

4. Potvrda i plaćanje

Komponente dijagrama aktivnosti

● Čvorovi

Akcije - značenje u domeni
Upravljanja - koordinacija tijeka

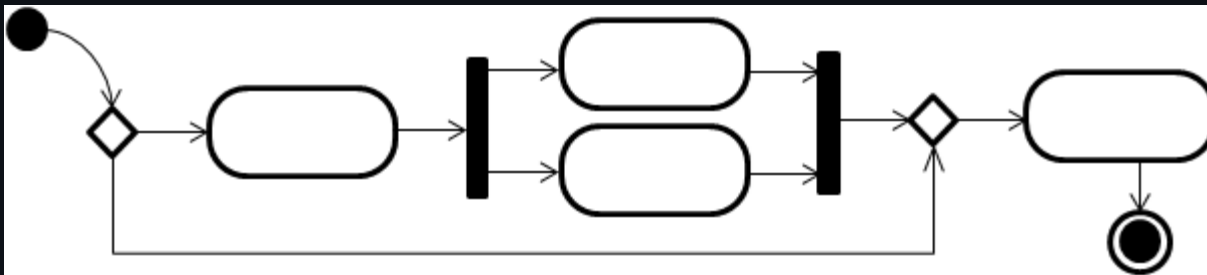
→ Veze

Tijek kontrole - strelice
Tok podataka - prenos objekata

▮ Particije

Swim lanes - grupiranje
Podjela po aktorima

Rezultat: Aktivnost prikazana kao koordinirani tijek akcija



Čvor akcije (Action Node)

Osnovni element dijagrama

Prikaz

Ime akcije

Zaobljeni pravokutnik

Imenovanje

Glagol ili imenica + opis:
"ispuni narudžbu", "provjeri podatke"

Real-world: Netflix flow

Odaberi sadržaj

Provjeri pretplatu

Pokreni streaming

 Imenujte akcije iz perspektive korisnika!

Čvorovi upravljanja tijekom izvođenja



Početni čvor

Početak aktivnosti



Završni čvor

Kraj aktivnosti



Čvor izbora

Decision - jedna grana



Spajanje

Merge - zajednička točka



Paralelizam

Fork - sve grane



Sinkronizacija

Join - čeka sve

Odluke vs Paralelizam

◇ Čvor izbora (Decision)

Tok ide u JEDNU granu

IF-THEN-ELSE logika

Primjer: Instagram login

- [Ispravna] → Feed
- [Neispravna] → Greška

▮ Fork & Join (Paralelizam)

Tok idu u SVE grane istovremeno

Fork: dijeli na paralelne grane

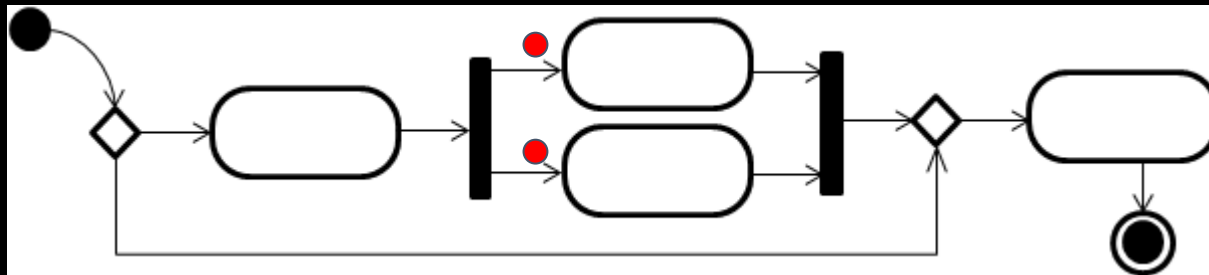
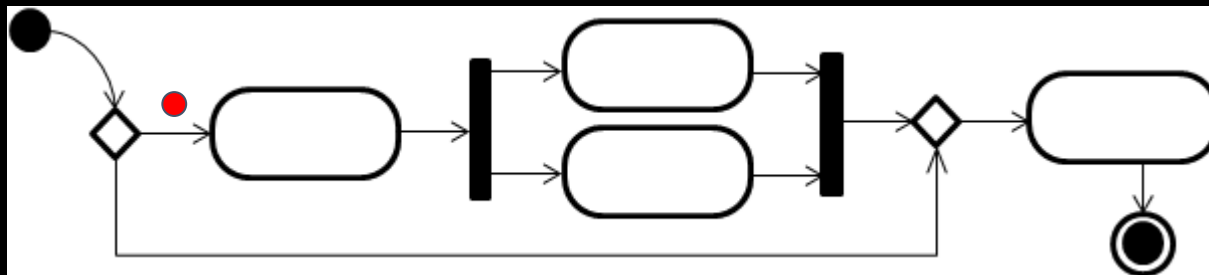
Join: čeka završetak svih

Primjer: Spotify download

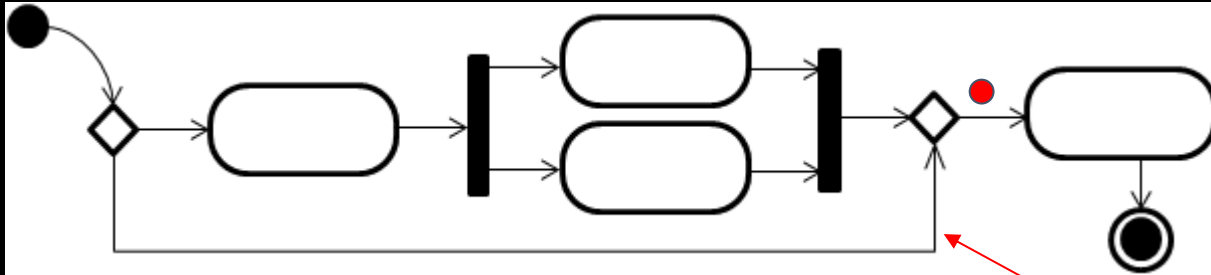
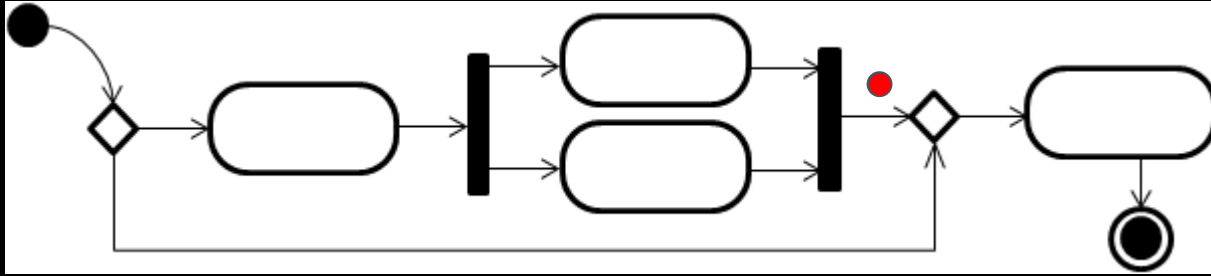
- Song 1 ↓ Song 2 ↓ Song 3

⚠ Česta greška: Brkanje izbora (XOR - jedna grana) i paralelizma (AND - sve grane)

Odluke vs Paralelizam - grananje



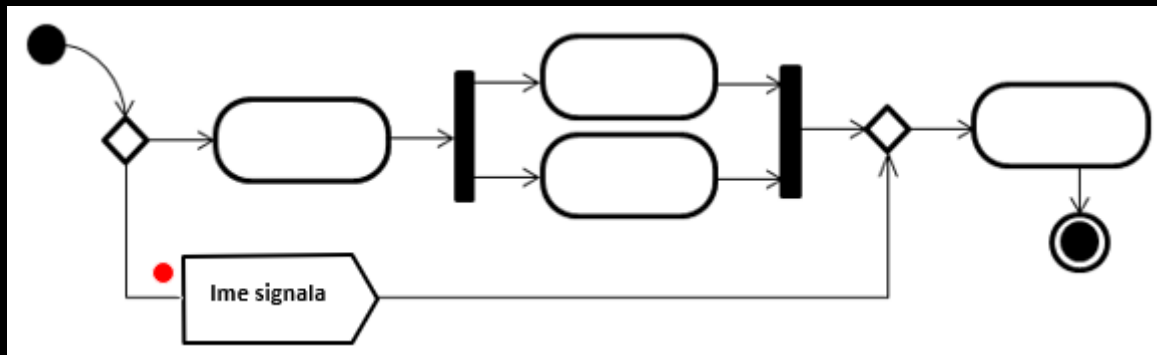
Odluke vs Paralelizam – sinkronizacija i spajanje



Signali

Slanje signala

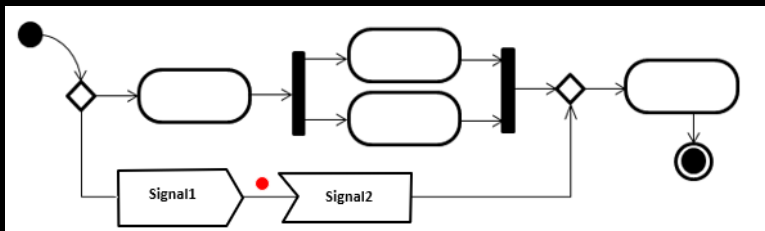
- Kad dođe značka, šalje se signal
- Ne čeka se odgovor



Signali

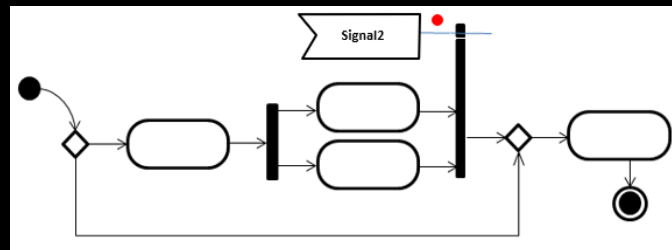
Primanje signala

- Kad dođe značka, čeka se pojava signala
- Signal je uvjet za nastavak slanja značke (kao zahtjev i odgovor)



Primanje signala

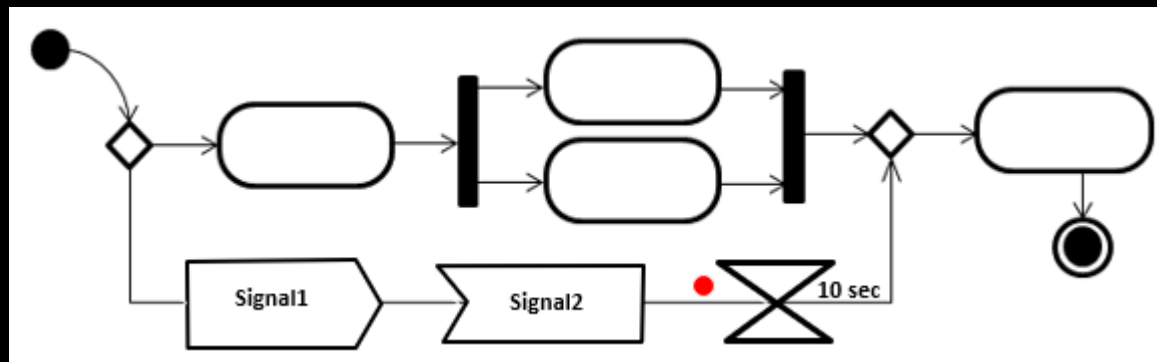
- Kad nema dolaznu granu, uvijek se očekuje signal
- Pojava signala pokreće tijek izvođenja



Vremenski signali

Čekanje vremena

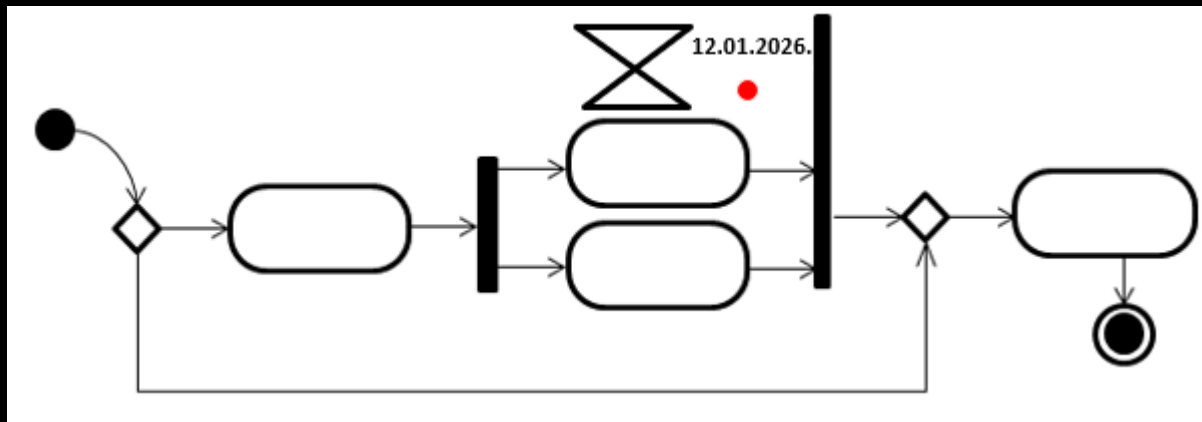
- Nakon definiranog intervala čekanja značka se šalje na izlaznu granu



Vremenski signali

Čekanje vremena

- Kad nema dolaznu granu, onda nije čekanje
- Može biti periodički signal ili jednokratni vremenski signal



Particije (Swim Lanes)

Zašto koristiti?

- Jasna podjela odgovornosti
- Lakša komunikacija
- Identifikacija bottleneck-ova

💡 Granice NE smetaju tijeku!

Primjer: Online kupovina



Kupac



Shop



Banka

Odaberi



Unesi

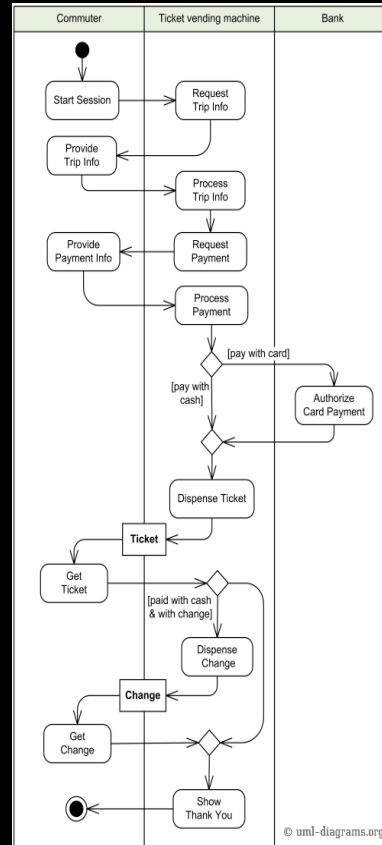


Provjeri



Autoriziraj

Particije (Swim Lanes)



Dijagram aktivnosti: Mogućnosti i ograničenja

✓ Mogućnosti

- ✓ Prikaz paralelnosti
- ✓ Control flow - proceduralni tijek
- ✓ Data flow - tok podataka
- ✓ Signali i vremenski eventi

⚠ Ograničenja

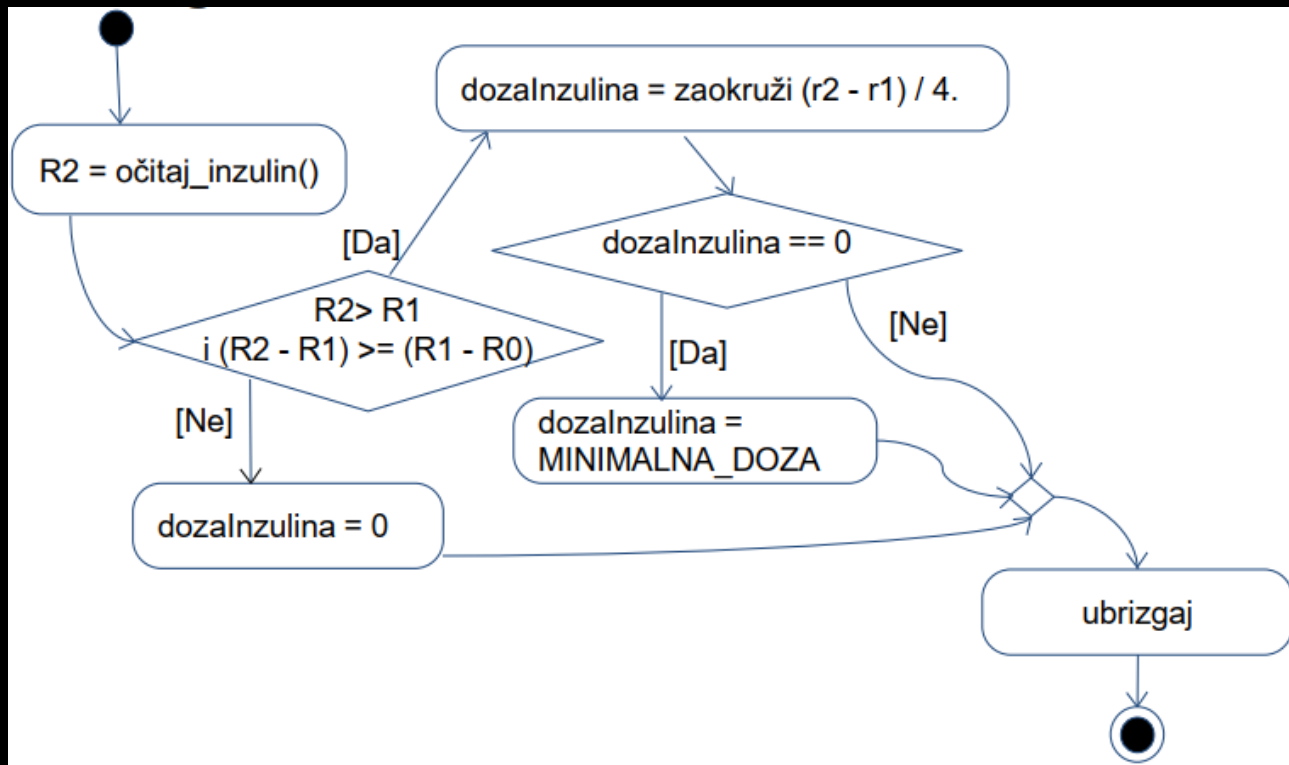
- X Nedeterminizam - proizvoljni redoslijed
- X Event-driven - ograničeno modeliranje
- 🤖 Za state-based ponašanje koristite State Machine

 67% arhitekata koristi activity dijagrame za dokumentiranje procesa (IEEE 2024)

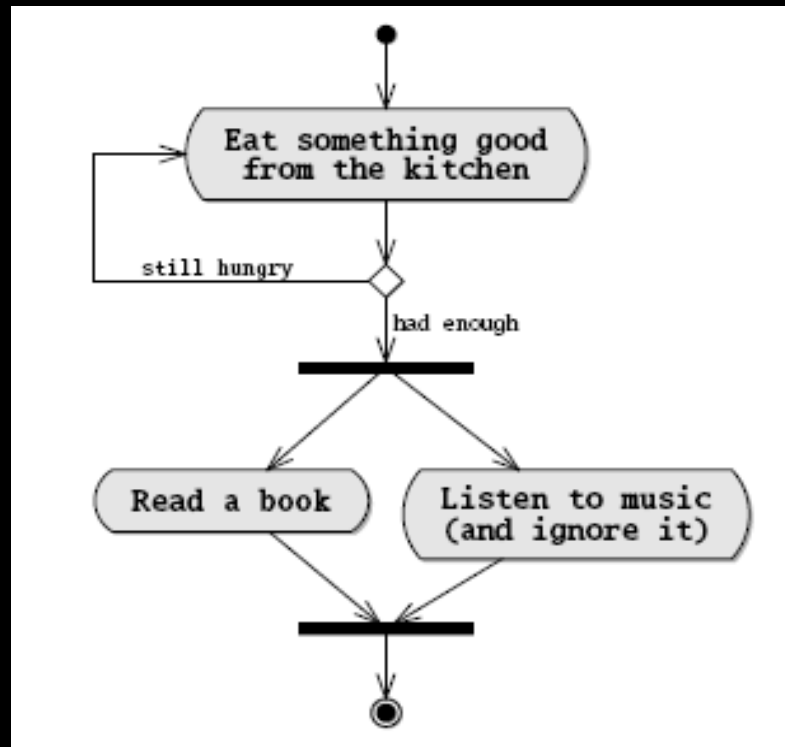
Dijagram aktivnosti - primjer

UVJET	AKCIJA
Razina šećera opada ($r_2 < r_1$)	doza Inzulina = 0
Razina šećera stabilna ($r_2 = r_1$)	doza Inzulina = 0
Razina šećera raste ($r_2 > r_1$), rast šećera je u opadanju $r_2 - r_1 < r_1 - r_0$	doza Inzulina = 0
Razina šećera raste ($r_2 > r_1$), rast šećera je je stabilan ili u porastu $r_2 - r_1 \geq r_1 - r_0$	dozaInzulina = zaokruži $(r_2 - r_1) / 4$. Ako je zaokružena vrijednost = 0, onda dozaInzulina = MINIMALNA_DOZA

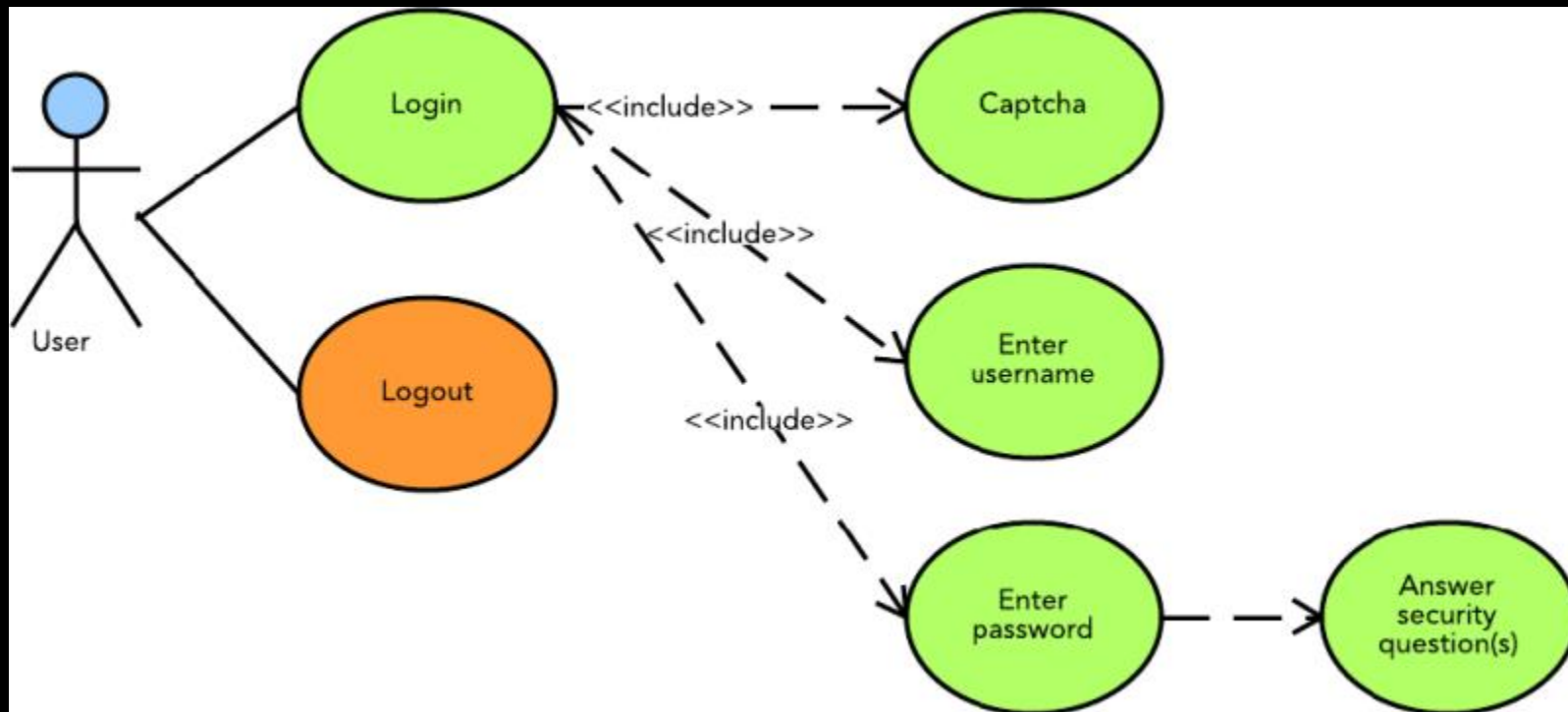
Dijagram aktivnosti - primjer



Dijagram aktivnosti - primjer



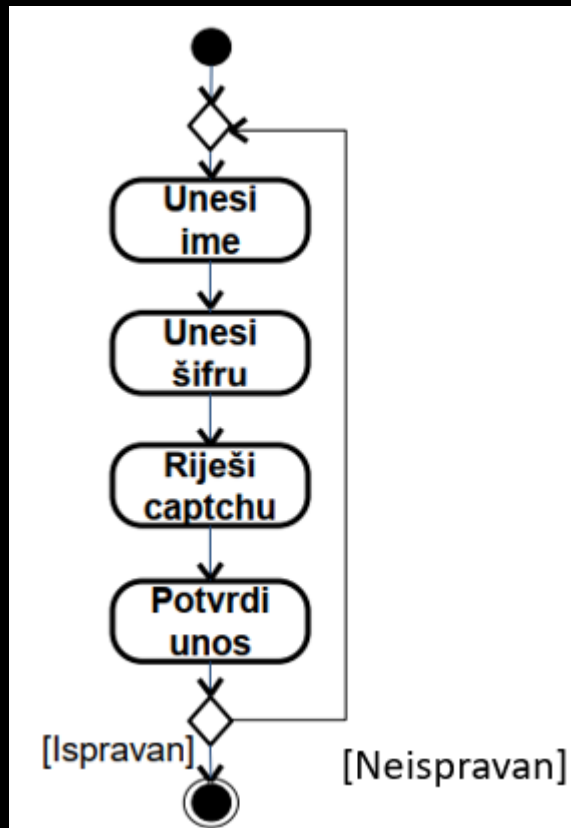
Dijagram aktivnosti - primjer



Dijagram aktivnosti - determinizam

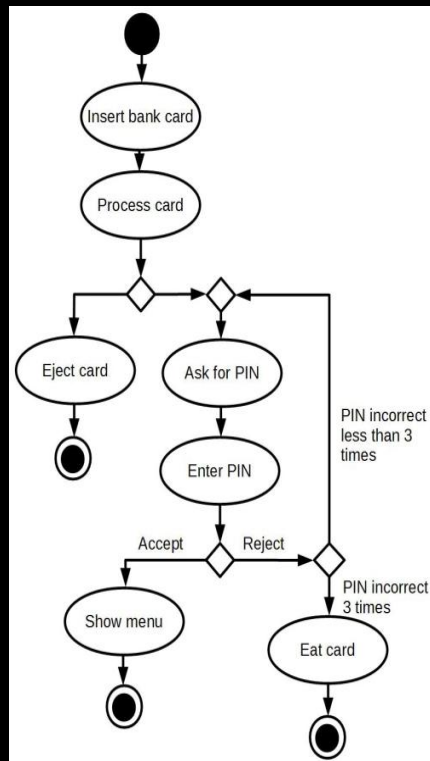
Zašto koristiti?

- Definiranje konkretnog redoslijeda
- Mogući i drugi redoslijedi izvođenja
- Ne može se izraziti nedeterminizam

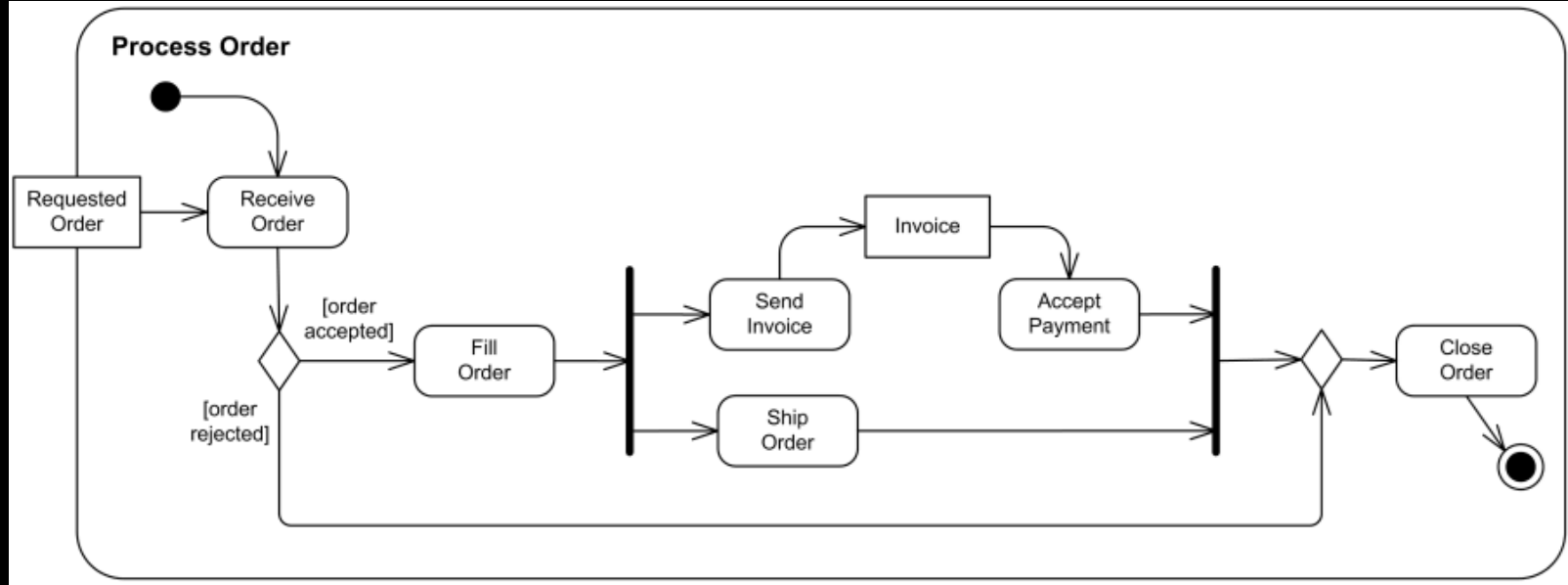


Dijagram aktivnosti – primjer s pogreškom

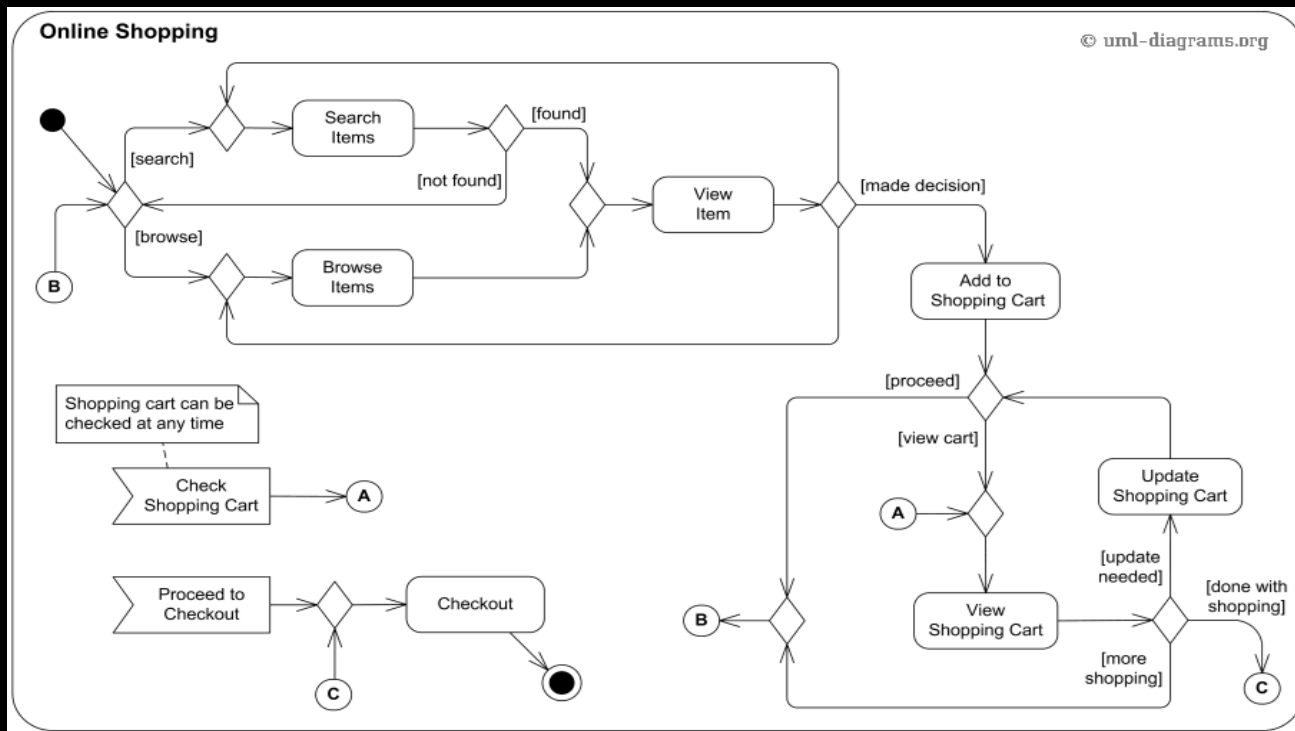
⚠ Pogreška – nedostaju uvjeti kod grananja



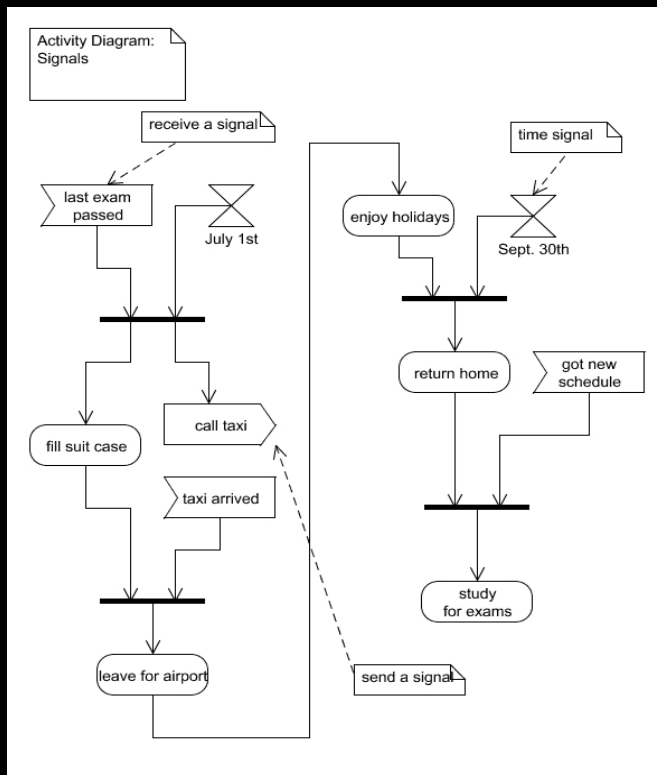
Dijagram aktivnosti – primjer modeliranja procesa kupovine



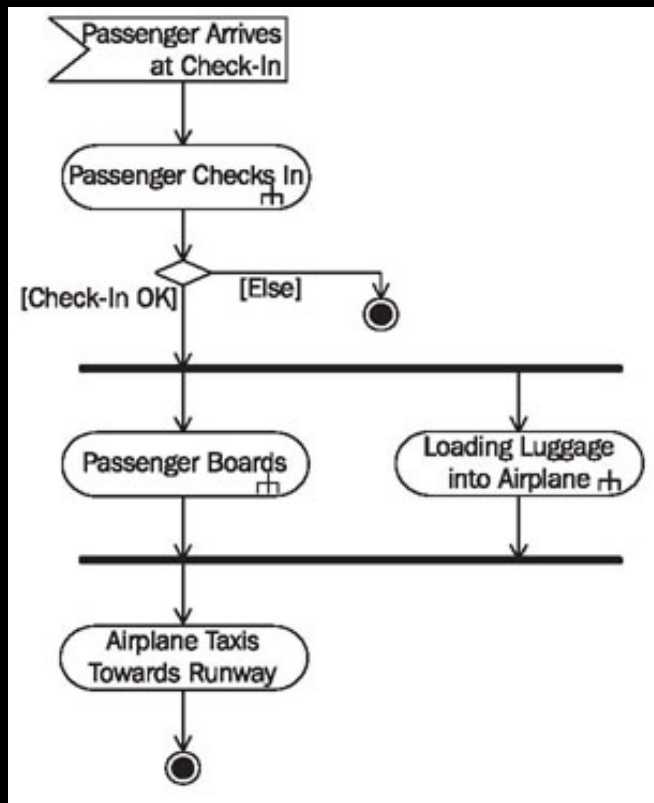
Dijagram aktivnosti – primjer modeliranja procesa kupovine



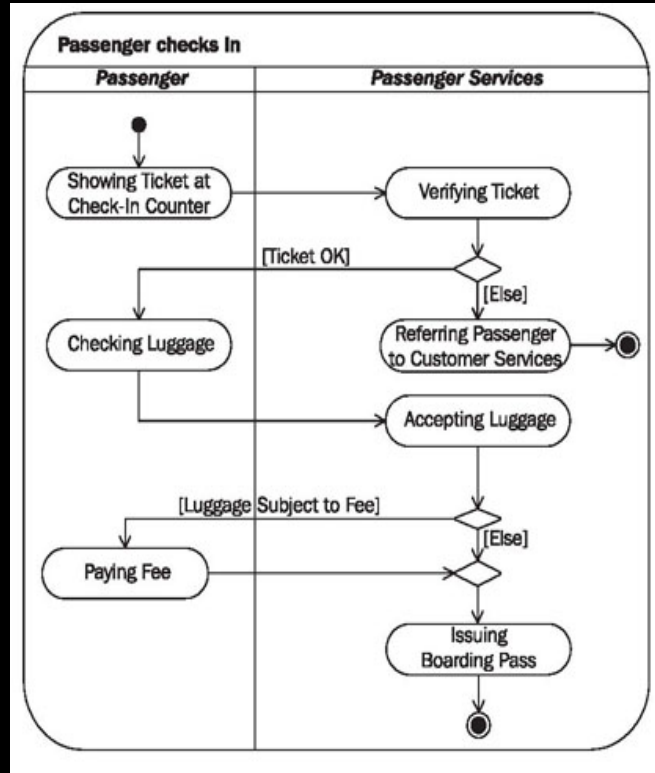
Dijagram aktivnosti sa signalima



Dijagram aktivnosti – prijava putnika u zračnoj luci



Dijagram aktivnosti – prijava putnika u zračnoj luci



Dio 2

Sekvencijski dijagram

Sequence Diagram

Interakcija između dijelova sustava kroz vrijeme

Što je sekvencijski dijagram?

Prikazuje interakciju između dijelova sustava - objekata, komponenti ili aktera - kroz vremensku dimenziju.

Koristi se za:

- Interakcije između objekata u use case-u
- Ostvarenje funkcionalnosti kroz interakciju
- Dokumentiranje redoslijeda poruka

 Najpopularniji dijagram za interakcije!

OAuth 2.0 Login

User → App: login()
App → OAuth: redirect
OAuth → App: token
App → User: session

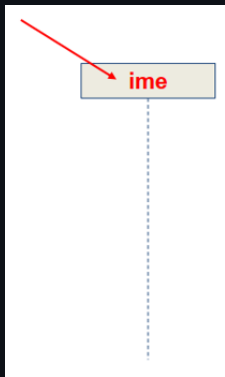


89% API dokumentacija koristi sekvencijske dijagrame za OAuth tokove


Označavanje sekvencijskog dijagrama

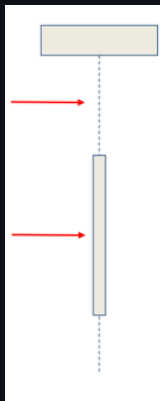
Objekt

Sudionik u komunikaciji
Izmjenjuje poruke s ostalim objektima
instanca:Razred
mojRepo:Repository



Životna linija

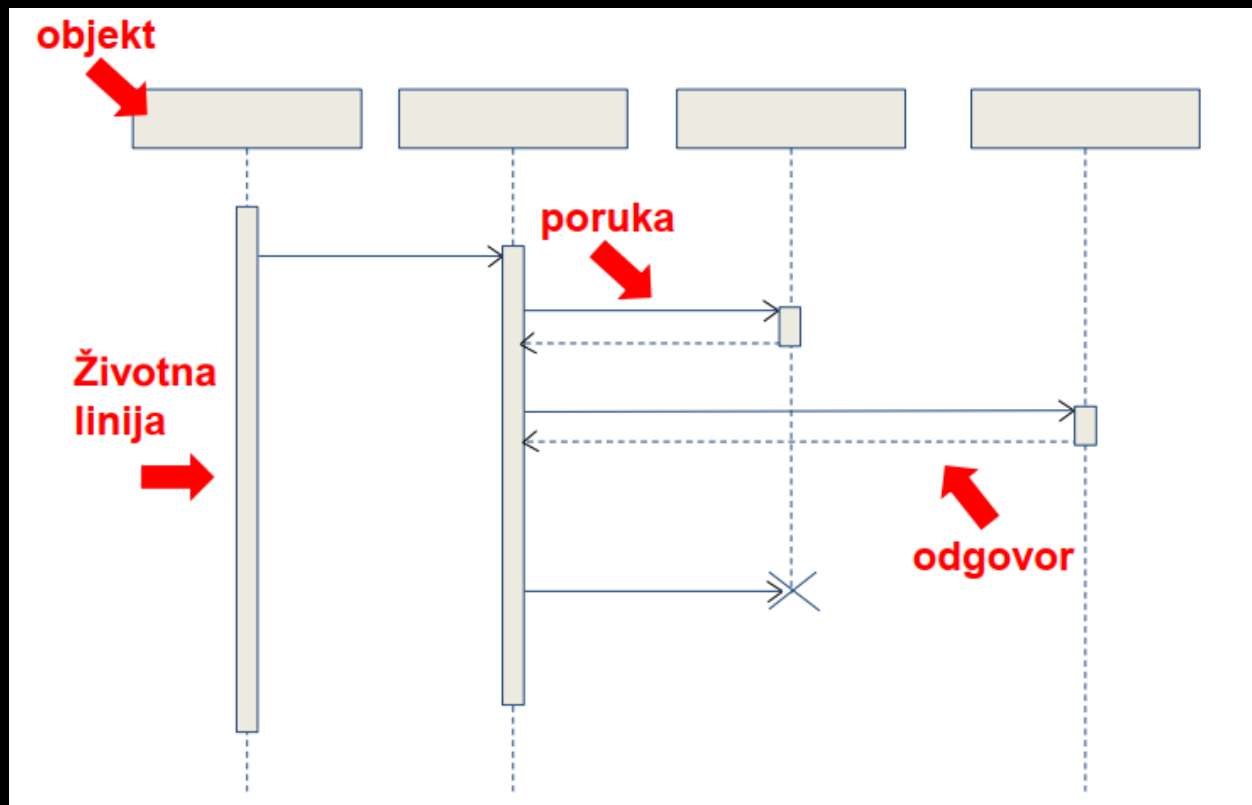
Lifeline - period postojanja
 Vrijeme teče dolje



Aktivacija

Activation bar
Objekt obrađuje poruku

Označavanje sekvencijskog dijagrama



Tipovi poruka (Message Arrows)

—▶ Sinkrona

Čeka odgovor
Primjer: REST API, DB upit

—→ Asinkrona

Ne čeka
Primjer: Message queue

--> Povratna

Return
Primjer: response

↪ Refleksivna

Self-call
Primjer: validate()

Stvaranje

Create
Primjer: new Instance()

✕ Uništavanje

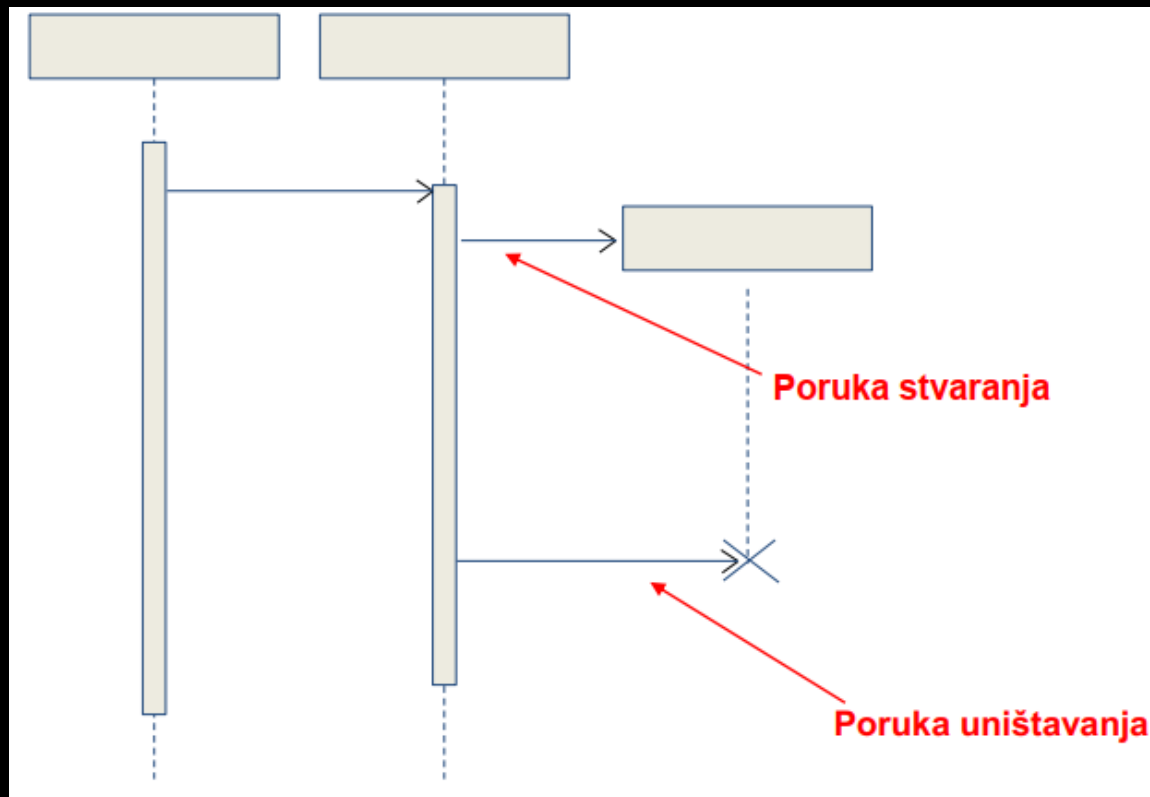
Destroy
Primjer: close()

Potpis: message_name(arguments):return_type — sve osim imena je opcionalno

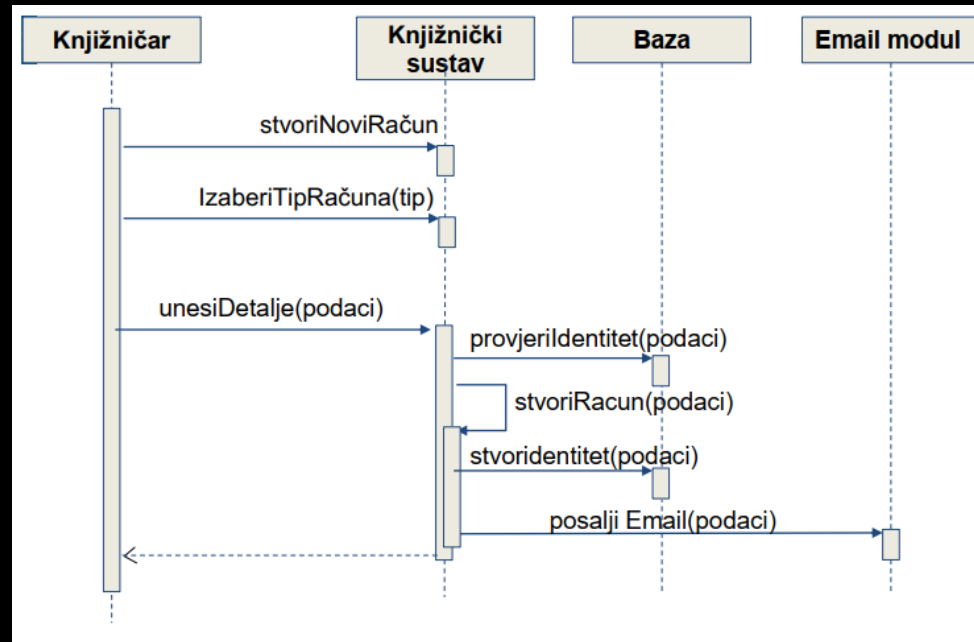
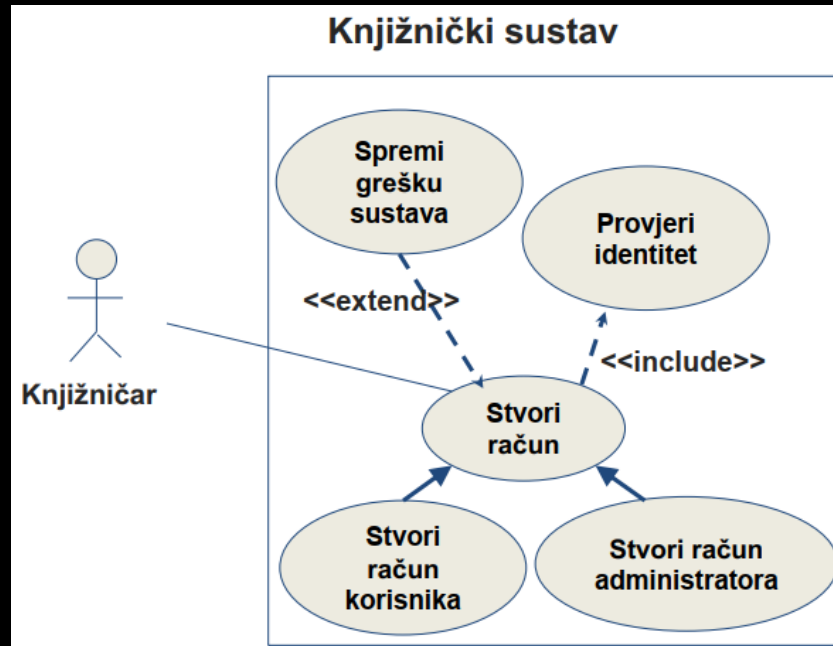
Tipovi poruka (Message Arrows)



Tipovi poruka (Message Arrows)



Sekvencijski dijagram – primjer implementacije



Fragmenti (Combined Fragments)

alt - Alternativa

IF-THEN-ELSE
Izbor između sekvenci

✓ Login success/failed

opt - Opcionalnost

IF-THEN (bez else)
Ako je uvjet ispunjen

✓ Premium features

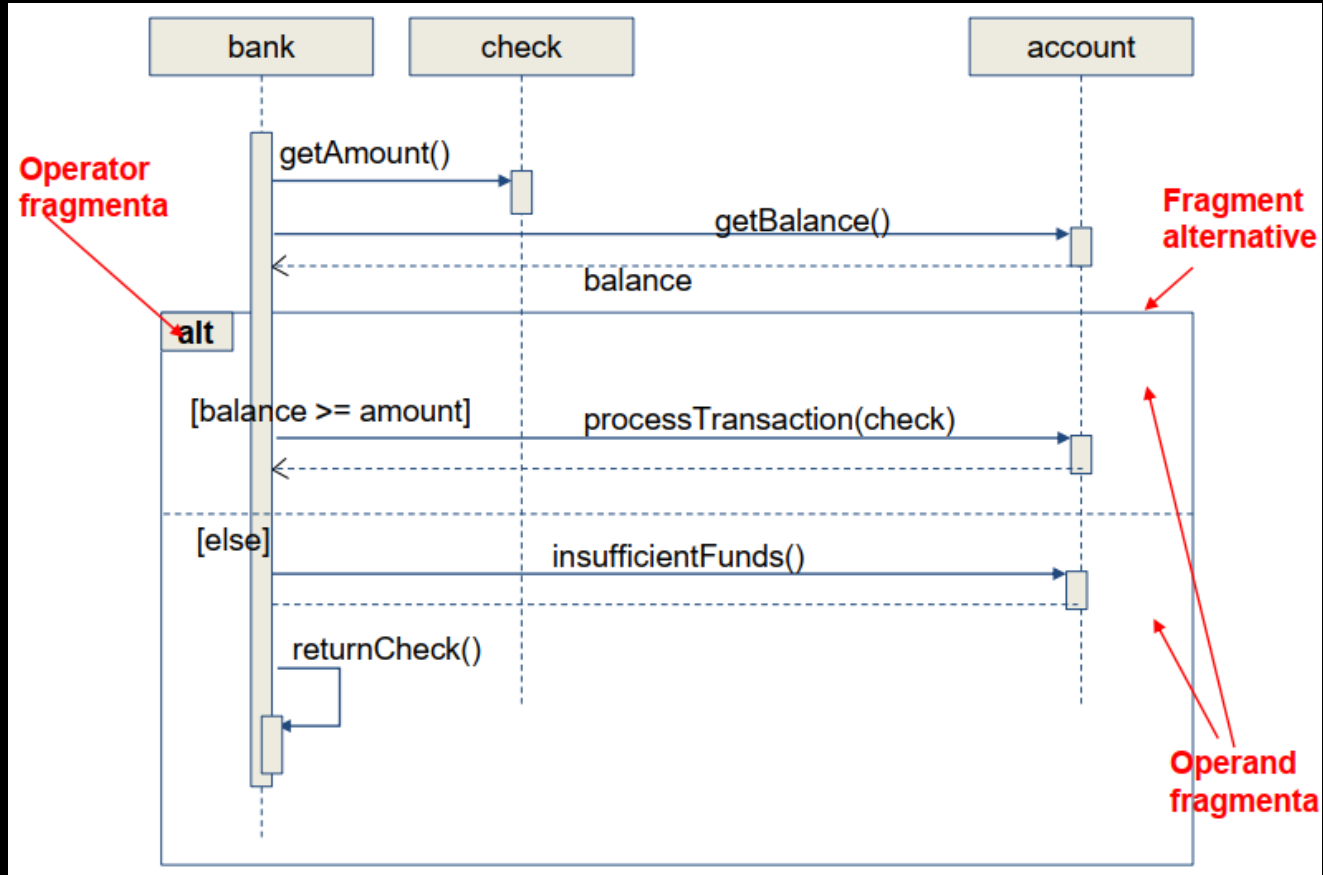
loop - Petlja

WHILE / FOR
Ponavlja dok vrijedi uvjet

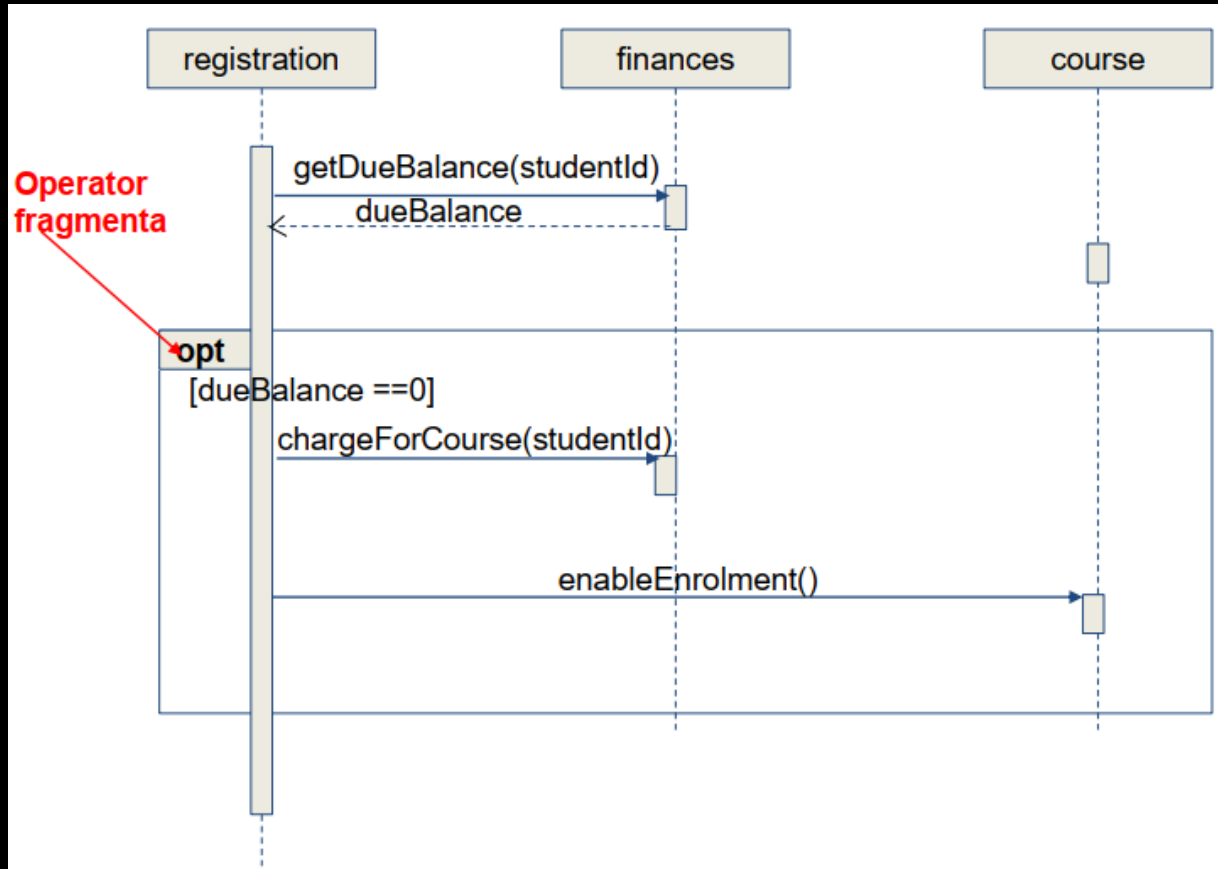
✓ Pagination

Operatori: alt, opt, loop, par (paralelno), break, critical, neg, ref

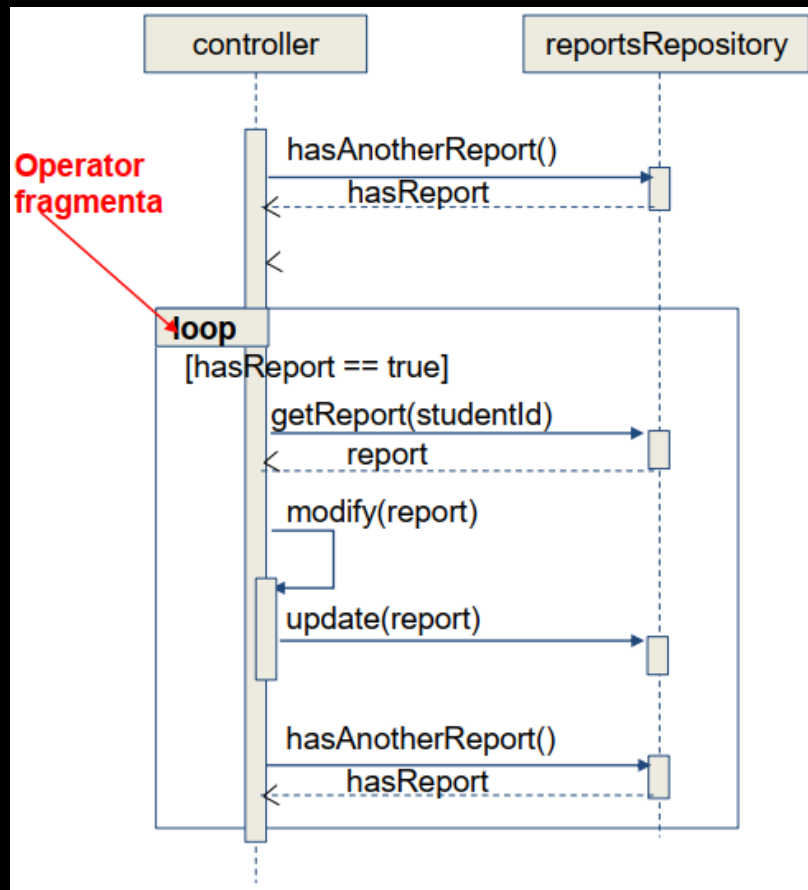
Fragment alternative



Fragment opcionalnosti



Fragment petlje



Primjer: Stvaranje korisničkog računa

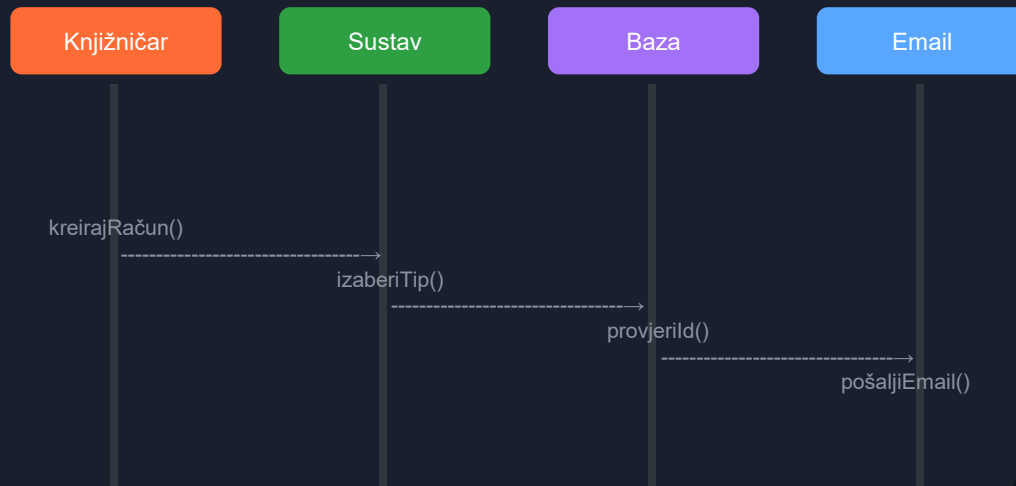
1 Sudionici

- Knjižničar (aktor)
- Knjižnički sustav
- Baza podataka
- Email modul

2 Tijek

1. Zahtjev za novi račun
2. Izbor tipa računa
3. Unos podataka
4. Provjera identiteta
5. Stvaranje u BP
6. Email potvrda

Sekvencijski dijagram



Activity vs Sequence diagram 🤔

Dijagram aktivnosti

Fokus: ŠTO se događa

- Poslovni procesi i workflow
- Algoritmi i logika
- Paralelno izvođenje
- Swim lanes (odgovornosti)

 Primjer: Online narudžba od košarice do dostave

Sekvencijski dijagram

Fokus: TKO s KIM komunicira

- Interakcija između objekata
- API pozivi i responses
- Vremenski slijed poruka
- Životni ciklus objekata

 Primjer: OAuth 2.0 autentifikacija

💡 Pro tip: Koristi OBOJE! Activity za high-level, Sequence za detalje.

Hvala na pažnji!



Activity Diagram
Procesi, algoritmi, paralelizam



Sequence Diagram
Interakcije, API, vrijeme



Resursi: uml-diagrams.org • PlantUML

Pitanja?