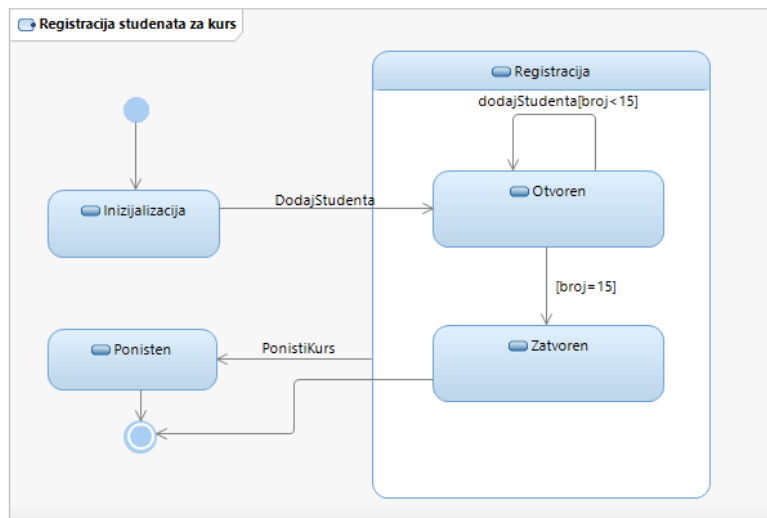


## VEŽBA 6.

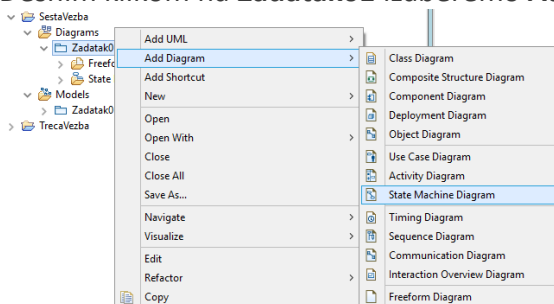
U ovoj vežbi upoznaćemo se sa dijagramima stanja.

### Zadatak 1: Koristedi RSA kreirati dijagram stanja i nacrtati dijagram sa slike

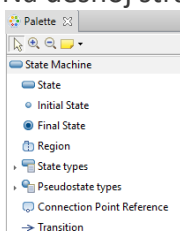


#### Rešenje:

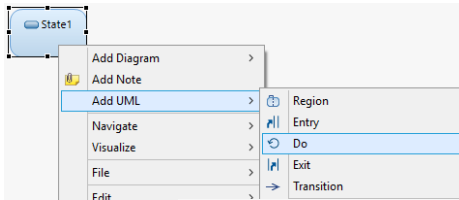
1. **File > New > Model Project** vežbu nazvati *Vežba06*. Klikom na **Next**, polje **File name** nazvati *Zadatak01*. Zatim klik na **Finish**.
2. Desnim klikom na **Zadatak01** izaberemo **Add Diagram > State Machine Diagram**



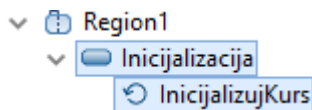
3. Na desnoj strani ekrana nalazi se **Palette**, iz ove palete možemo birati potrebne elemente za naš dijagram.



- Iz palete biramo **Initial State** postavljamo ga prostim prevlačenjem u glavnoj regiji dijagrama stanja. Glavnu regiju dijagrama stanja nazvaćemo **Registracija stuenata za kurs**.
- Iz iste palete biramo **State** i postavljamo je u glavnoj regiji. Prvo stanje nazivamo **Inicijalizacija**. Desnim klikom na stanje biramo **Add UML > Do**.

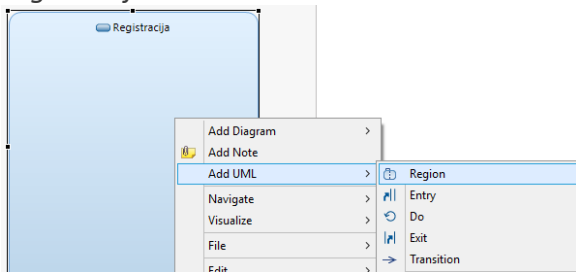


- Zatim biramo **Create Activity** i ovu aktivnost nazivamo **InicijalizujKurs**. Hijerarhijska struktura bi trebalo ovako da izgleda:



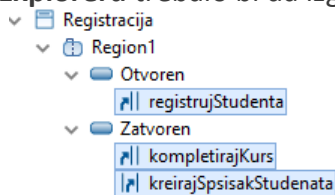
\*Ukoliko se ne otvori prozor da dodeljivanje imena na izabranu aktivnost, desnim klikom na **Do > Refactor > Rename**

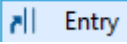
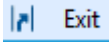
- Sada je potrebno napraviti novo stanje **Registracija** koje će ujedno biti i podregija za dva dodatna stanja **Otvoren** i **Zatvoren**. Prostom prevlačenjem **State** u glavnoj regiji postavljamo novo stanje i nazivamo ga **Registracija**. Desnim klikom na novokreirano stanje biramo **Add UML > Region**.

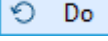


Unutar ove nove regije unosimo dva nova **State** i jednom dodeljujemo ime *Otvoren*, i drugome *Zatvoren*.

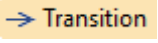
- Sada je potrebno dodeliti aktivnosti novokreiranim stanjima, (*ponoviti dodeljivanje aktivnosti stanjima kao što je opisano u koraku 5.*) Po završetku dodeljivanja aktivnosti stanjima hijerarhijska struktura u **Project Exploreru** trebalo bi da izgleda ovako:



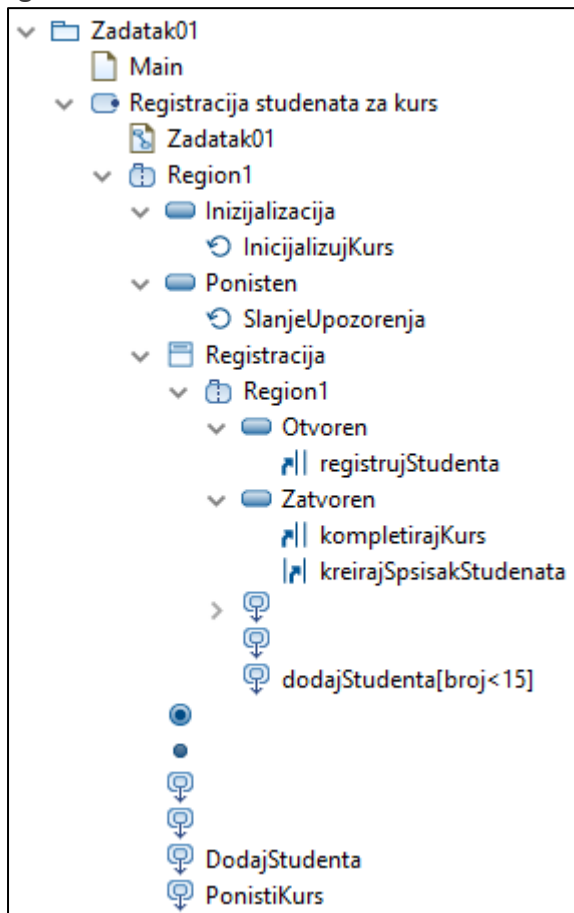
Aktivnost  **Entry** opisuje neke početne (ulazne) aktivnosti nekog stanja, dok  **Exit** opisuje krajnje (izlazne) aktivnosti nekog stanja.

9. Sada pravimo poslednje stanje **Ponisten** i dodeljujemo mu aktivnost  **Do** koju nazivamo **SlanjeUpozorenja**.

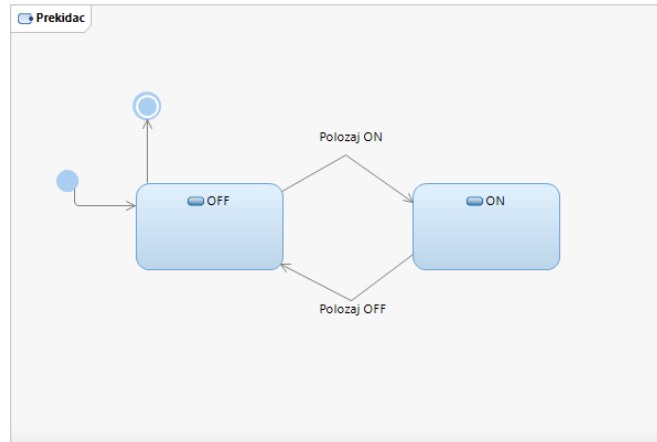
10. Iz palete biramo element  **Final State** i pozicioniramo ga kao što je prikazano na početnom dijagramu.

11. Sada je potrebno upotrebiti  **Transition** i povezati sva stanja. Tranzicija označava prelazak stanja iz jedno u drugo. Obratiti pažnju da se kod stanja **Otvoren** proces ponavlja 15. puta do njegovog zatvaranja!

12. Ukoliko ste pažljivo ispratili korake, hijerarhijska struktura **Zadatak01** u Project Exploreru bi trebala da izgleda ovako:

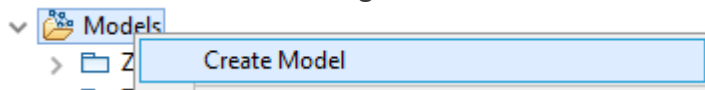


## Zadatak 2: Nacrtati dijagram stanja prekidača za sijalice

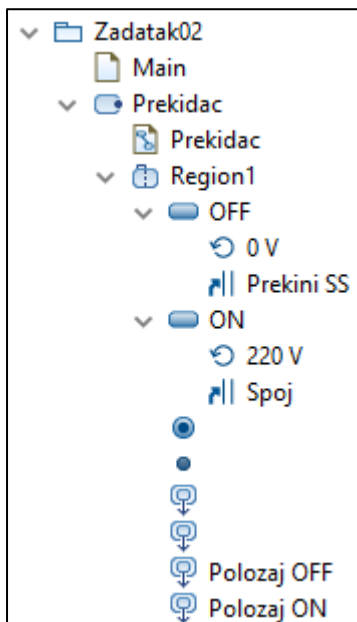


### Rešenje:

1. Kreirati nov model i nazvati ga **Zadatak02**



2. Za kreiranje dijagrama stanja ponoviti isti postupak kao iz predhodnog zadatka (*Korak 2.*)
3. Sa stečenim znanjem iz predhodnog zadatka **Zadatak01**, kreirati dijagram stanja prekidača za sijalice.
4. Po završetku trebalo bi da imate ovakvu hijerarhijsku strukturu:

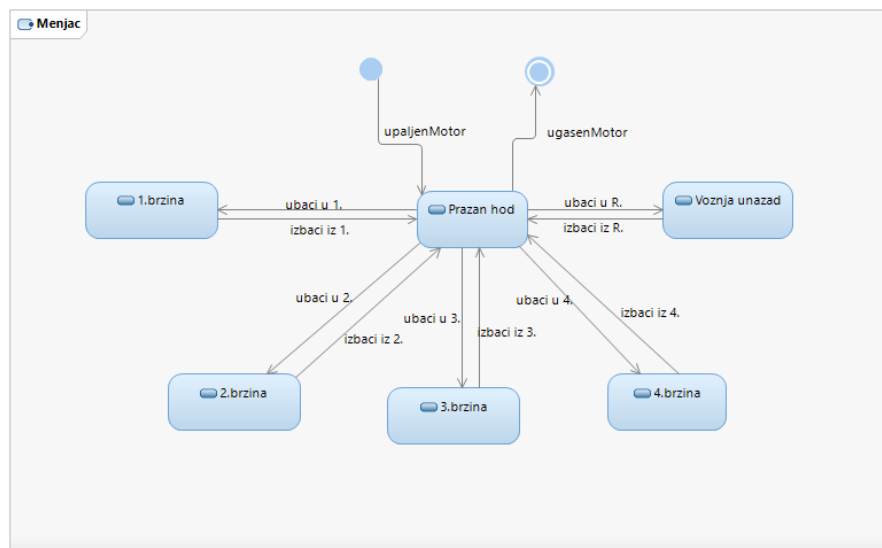


### Zadatak 3: Nacrtati dijagram stanja za upravljanje menjačem u automobilu

Menjač se može nalaziti u sledećim stanjima.

- Prazni hod
- Vožnja unazad
- 1.brzina
- 2.brzina
- 3.brzina
- 4.Brzina

Rešenje:



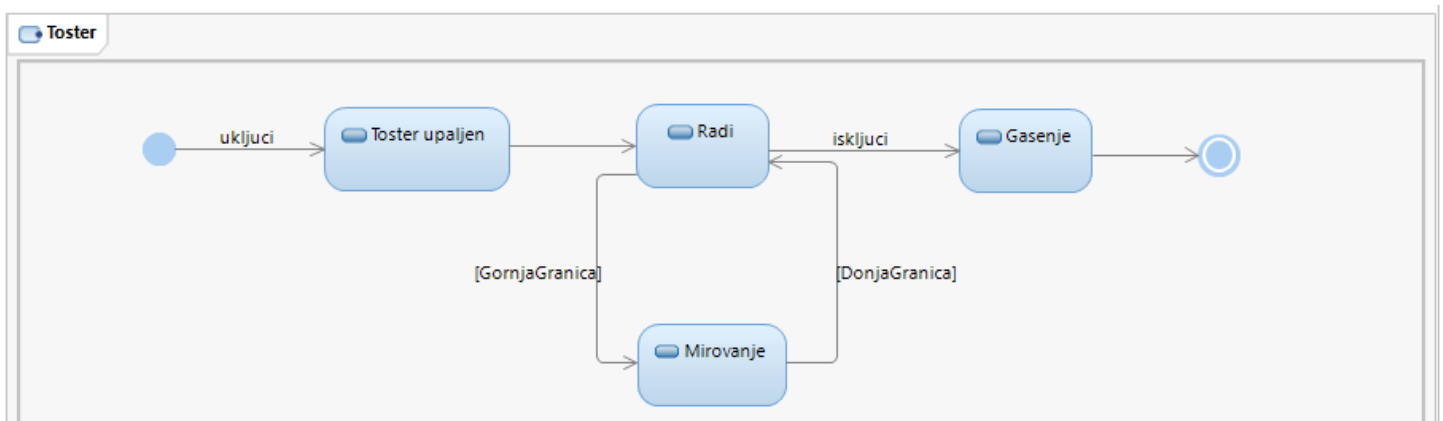
Iz predhodno stečenog iskustva iz *Zadatak01* i *Zadatak02* kreirati dijagram stanja menjača automobila.

#### Zadatak 4: Nacrtati dijagram stanja tostera za hleb

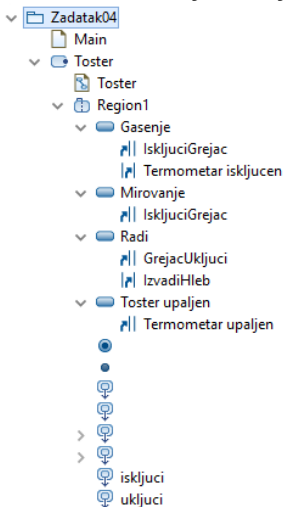
Prvo moramo uključiti toster, staviti hleb i čekati nekoliko minuta da bi se ispekao. Da bi sprečili da nam hleb izgori, grejač tosteru mora davati temperaturu u određenom intervalu.

Termometar mora meriti temperaturu grejača i kada se dostigne gornja granica temperature grejač prelazi u stanje isključenosti. Grejač ostaje u tom stanju dok mu se temperatura ne spusti do donje granice i tada opet započinje sa radom.

#### Rešenje:



Po završetku hijerarhija zadatka izgleda:



\*U zadacima, u hijerarhiji nije bitan redosled elemenata već je bitno da je broj elemenata jednak zbiru kao što je prikazano na slikama.