

Implementacijski dijagrami su:

- **Dijagram komponenti (engl. Component diagram) i**
- **Dijagram rasporeda (engl. Deployment diagram).**

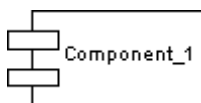
Dijagram komponenti (Component Diagram)

Dijagram komponenti je jedan od UML statičkih dijagrama objektno orijentiranog modela i predstavlja fizičku strukturu sistema ili softvera.

Dijagram komponenti ima strukturu grafa čija čvorišta su komponente.

Komponente su osnovni gradivni blokovi sistema ili softverskog proizvoda koji mogu biti ponovo korišteni (engl. reusable) i zamijenjeni (engl. replaceable). Dakle, komponente je potrebno dizajnirati na način da se mogu ponovo koristiti i u drugim aplikacijama, te je potrebno specifikaciju, dizajn i implementaciju komponenti pažljivo osmisлити.

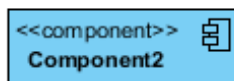
Simbol kojim se obično predstavlja komponenta softvera kao čvorište grafa predstavljen je na slici 1.



Slika 1: Instanca komponente software-a

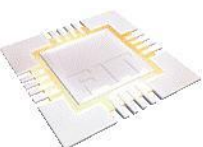
Unutar simbola se upisuje naziv komponente koji treba da bude deskriptivan. Nazivi komponenti se kreiraju po sličnim pravilima kao i nazivi klasa.

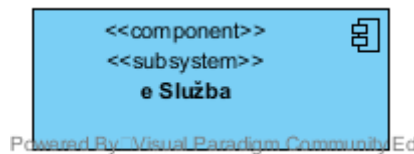
Simbol za instancu komponente u CASE alatu Visual Paradigm 14.0 je predstavljen na slici 2.



Slika 2: Simbol za instancu komponente software-a u CASE alatu

Stereotip komponente omogućava klasificirati komponente. Stereotip komponente može npr. biti podsistem i u tom slučaju se označava kao <<subsystem>> (slika 3). UML je dosta fleksibilan u pogledu ponašanja podsistema u okviru dijagrama komponenti.





Slika 3: Stereotipi komponente

Fizička komponenta (<<artifacts>>) je sve ono što se može pohraniti na tvrdi disk, npr. jedna datoteka koja sadrži kôd klase ili jedna tabela baze podataka.

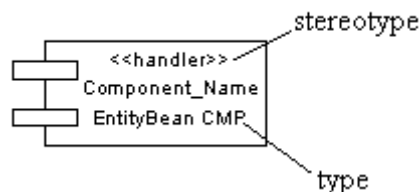
Tip fizičke komponente određuje se na osnovu njenih osobina i može biti:



- Datoteka - pohranjuje npr. izvorni kod jedne klase;
- Dokument je stereotip komponente kreiran u tekst procesoru npr. specifikacija zahtijeva za softver;
- Biblioteka (engl. library) je npr. file.dll ili stdio.h;
- Tabela u bazi podataka;
- Izvršni kod;
- Web stranica ili Web forma.

Standardna komponenta se koristi ukoliko komponenta se ne želi označiti kao posebna dok se specifična komponenta koristi npr. za EJB, JSP, Servlet ili ASP.NET.

Naziv tipa komponente može biti prikaza u samom simbolu komponente(slika 4).



Slika 4: Prikaz naziva tipa komponente software-a

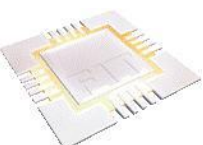
Komponente mogu sadržavati druge komponente ili klase.

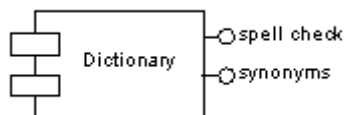
Dijagram komponenti se koristi za definiranje zavisnosti i relacija između objekata na višem nivou od dijagrama klasa, odnosno za modeliranje strukture softvera i prikazivanje ovisnosti između izvornog, binarnog i izvršnog koda.

Komponente mogu biti neovisne ili međusobno povezane vezama ovisnosti i vezama generalizacije ukoliko jedna komponenta nasljeđuje osobine druge komponente.

Sučelje (engl. Interface) komponente ukazuje na to da li komponenta sadrži implementaciju *usluge* koju pruža putem određenog interface-a ili zahtijeva uslugu od druge komponente. Komponente mogu koristiti jedan ili više sučelja (interface-a)(slika 5). Sučelje može biti zahtijevano ili ponuđeno.

Ponuđeno sučelje su usluge koje implementiraju klase koje čine komponentu, dok zahtijevano sučelje traži izvršenje usluge od druge komponente.

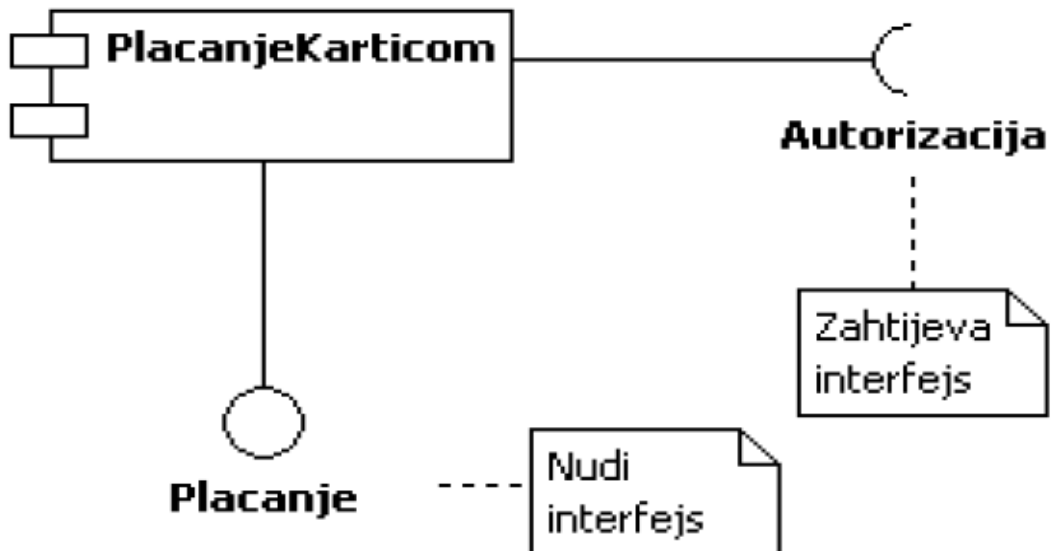




Slika 5: Interface-i komponenti

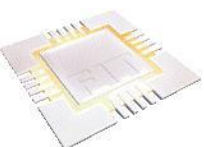
Sučelja komponenti (engl. Component interface) omogućavaju interakciju između komponenti. Zahtijevano sučelje komponente predstavlja se sa simbolom polukruga, a ponuđeno sučelje simbolom kruga (slika 5). U blizini simbola se piše naziv sučelja.

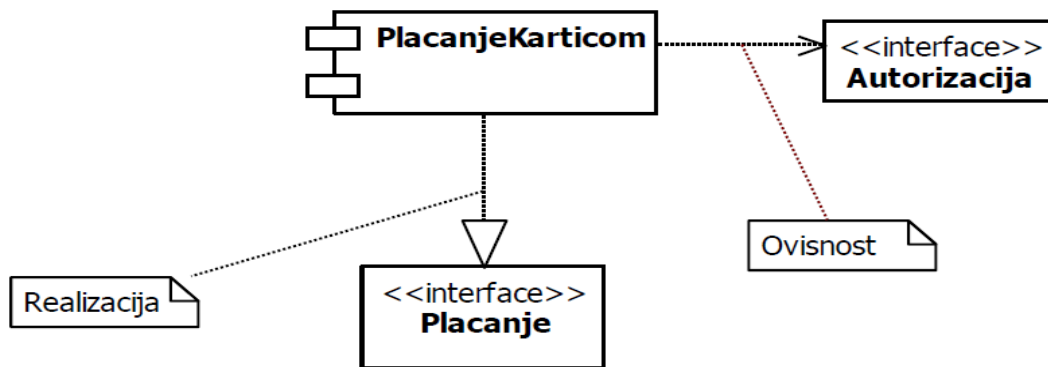
Sučelja komponente se također mogu prikazati kao klase stereotipa <<interface>>. Ukoliko sučelje je ponuđeno između komponente i sučelja se uspostavlja veza realizacije,



Slika 6: Ponuđeno i zahtijevano sučelje

a ukoliko se radi o zahtijevanom sučelju uspostavlja se veza ovisnosti.

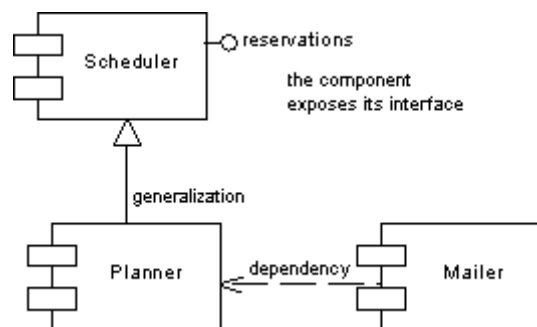




Slika 7: Klase stereotipa <<interface>>

Notacija prikazana na slici 7 je korisna ukoliko se žele prikazati i operacije sučelja.

Primjer dijagrama komponenti prikazan je na slici 8.



Slika 8: Dijagram komponenti software-a

Dijagram komponenti je koristan u fazi analize i dizajna, jer omogućava analitičarima i projektnim vođama definirati komponente software-a koje su im potrebne prije same faze implementacije i također pojednostavljuje dizajn, razvoj i održavanje komponenti.

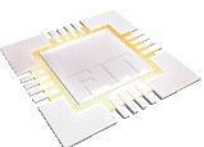
Komponente potvrđuju i osiguravaju realizaciju skupa sučelja (interface-a).

Sučelje komponente odvaja ponašanje od implementacije komponente, te smanjuje mogućnost da promjena jedne komponente prouzrokuje npr. pad sistema.

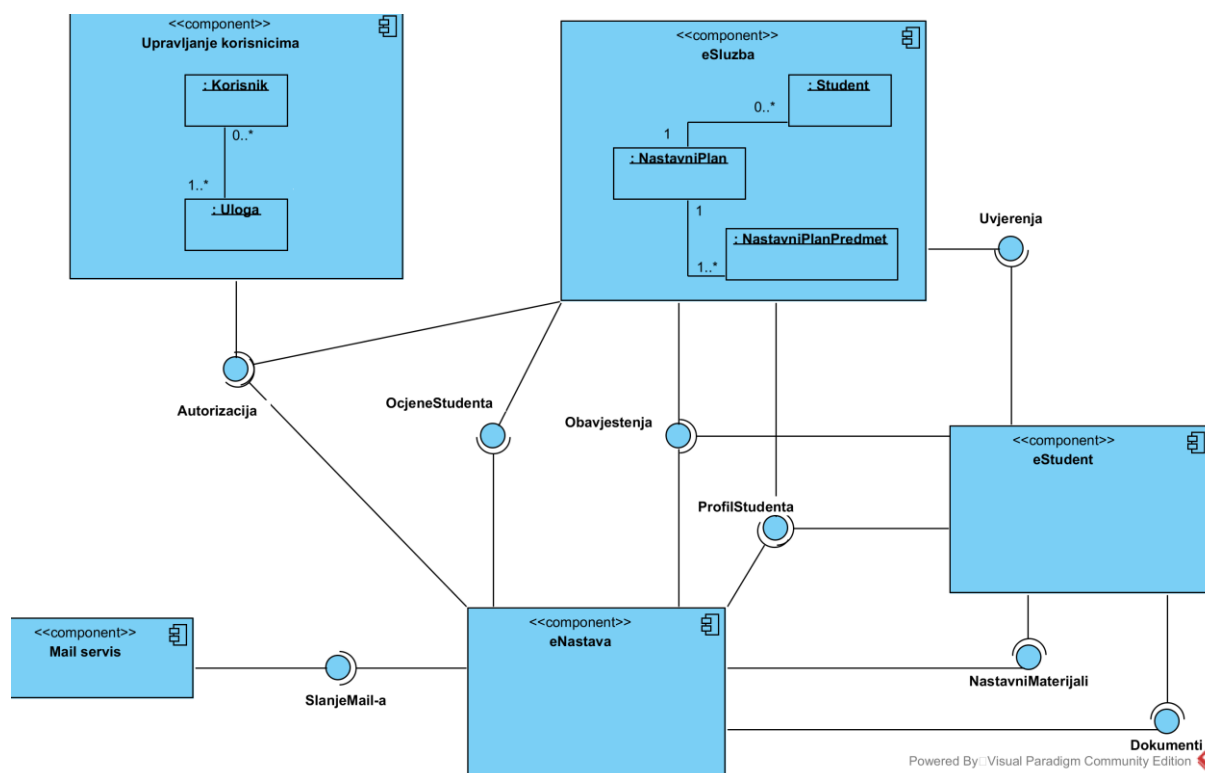
Dijagram komponenti za jedan projekt može se napraviti u cijelosti. Također mogu se koristiti dijelovi postojećih dijagrama komponenti ili u cijelosti dijagrame komponenti napravljene za neki drugi projekt.

Dijagrami komponenti u okviru procesa modeliranja se koristi za:

- modeliranje planiranih softverskih komponenti i interfejsa između njih na početku projekta čime se omogućava da se razmatranje detaljnog dizajna ostavi za bolje vrijeme.
- skrivanje specifikacije detalja komponenti, a fokusiraju se na međusobne relacije komponenti koje se specificiraju kao sučelja između komponenti.
 - da pokažu kako se prethodno dokazane komponente integriraju u tekući dizajn sistema.



Primjer dijagrama komponenti za eUniverzitet je prikazan na slici 9.



Slika 9: Dijagram komponentni - eUniverzitet