

# UML diagrammit (Visiolla)

## Vihjeitä ja lähteitä:

[Hyvä materiaali UML:stä, Pasi Kellokoski](#)

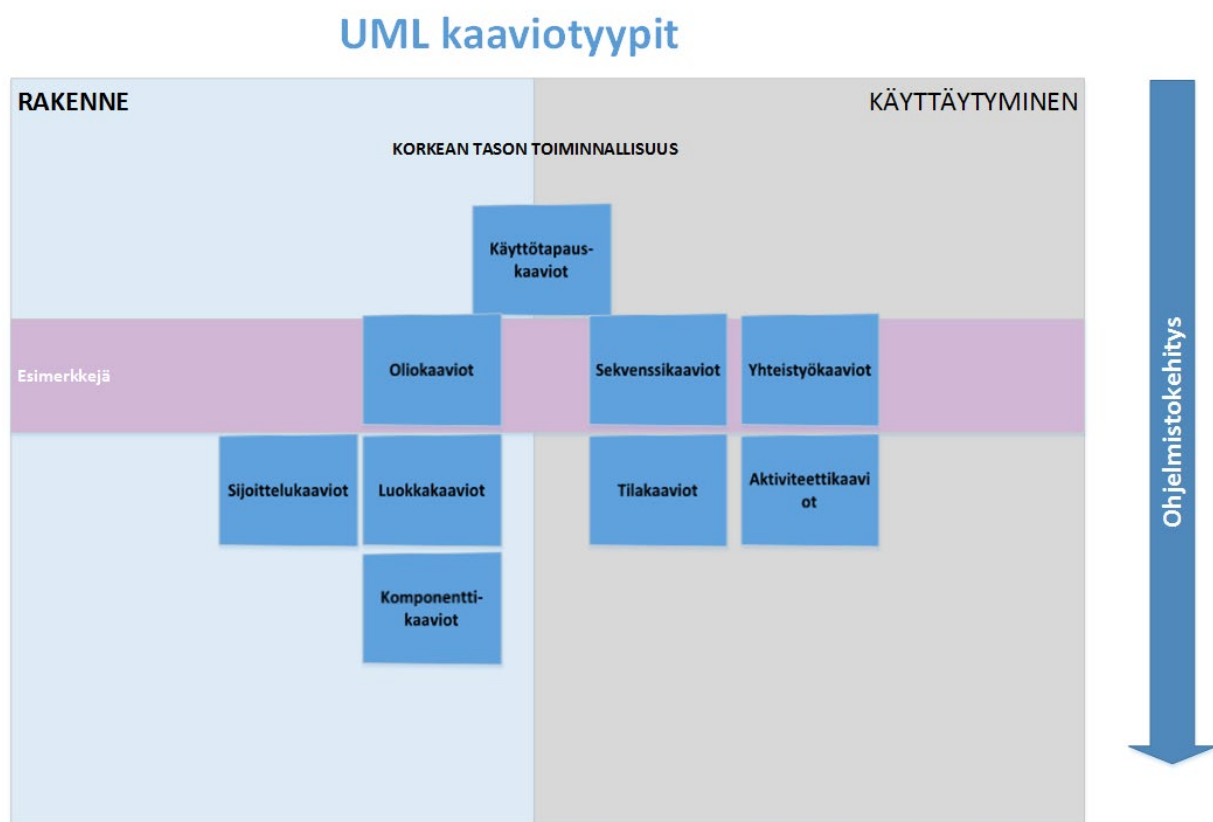
[UML ja Visio \(vanhempi versio, periaate toimii edelleen\), Jouni Huotari](#)

[Loistava pikaopas, Samuel Lahtinen](#)

[draw.io, online kaavioeditori](#)

UML on laaja mallien kuvaustapa, graafista suunnittelua.

## UML kaaviotyypit



## Kaavioiden peruskäsitteet

- Kaaviot koostuvat peruselementeistä ja niiden välisistä suhteista
- Elementit ovat geometrisia kuvioita, niillä on usein myös sisäinen rakenne.
- Suhteet ovat elementtejä yhdistäviä viivoja. Niihin voidaan liittää tarkempaa informaatiota.
- Graafista ulkoasua ei ole tarkasti määritetty.

# Symboleita

## ELEMENTTEJÄ

Luokka

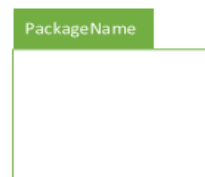
Näkyvyys:  
(+) public  
(-) private  
(#) protected



Tila



Pakkaus



Kommentti



Toimija  
(actor)



## SUHTEITA

Riippuvuus  
(dependency)  
Heikoin suhde  
Esim. A käyttää B:tä  
jossain  
jäsenfunktiossaan



Assosiaatio  
(association)  
Vahvempi kuin  
Edellinen  
(määrittelemätön)



Assosiaatio  
A tarvitsee B:tä, B:n ei  
tarvitse tietää mitään  
A:sta  
(yksisuuntainen)



Assosiaatio  
A tarvitsee B:tä,  
B tarvitsee A:ta  
(kaksisuuntainen)



Kooste  
(aggregation)  
A:lla voi olla monta  
B:tä esim.  
joukkueessa on  
monta pelaajaa



Muodoste  
(composition)  
A:lla voi koostua  
monesta B:stä (esim.  
shakkilauta koostuu  
ruuduista)

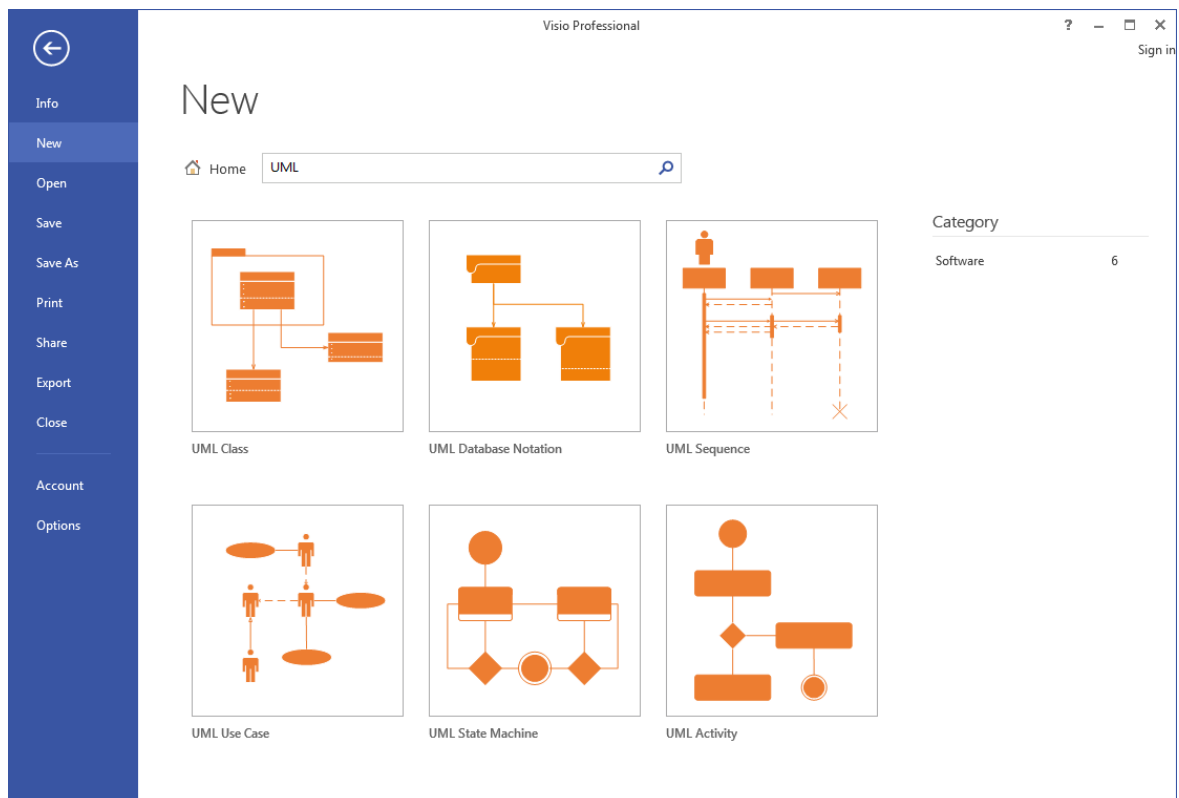


Yleistys  
(periytyminen,  
inheritance)



## Visio

Kun aloitat tekemään UML-kaavioita, valitse aina mallipohjaksi jokin UML-pohja. Hakusanalla UML löytyy ainakin seuraavia mallipohjia:

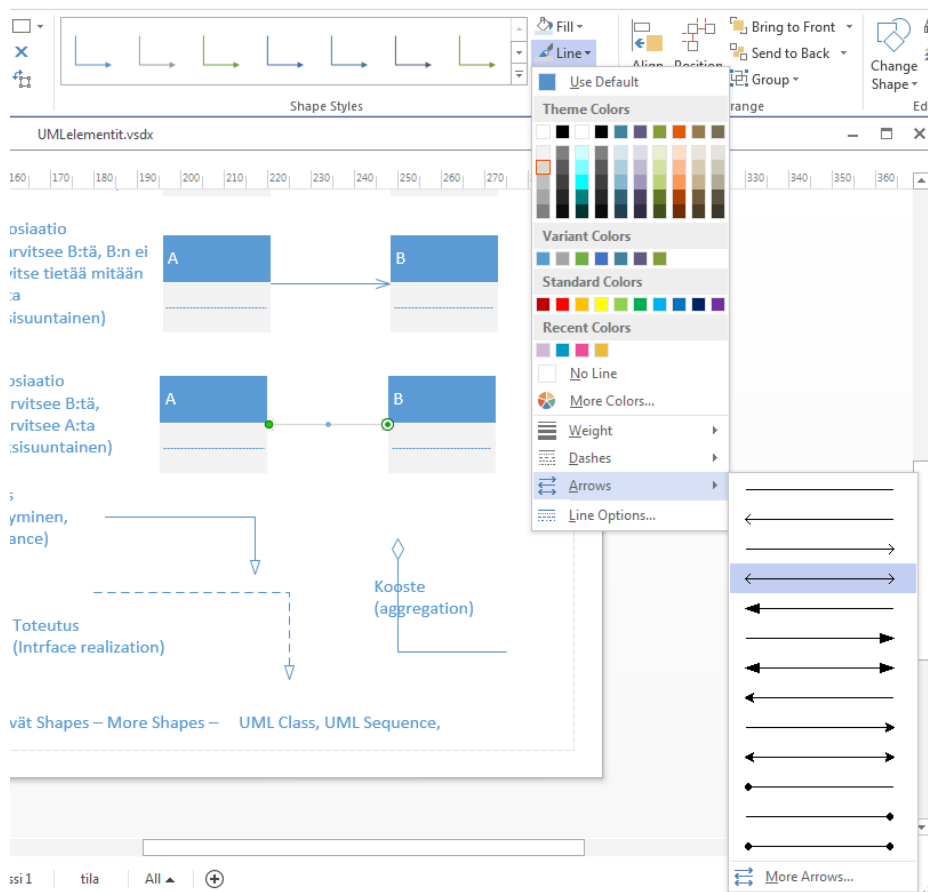


Erlaisia muotoja löytyvät: Shapes – More Shapes –

- UML Class
- UML Sequence
- UML State Machine
- UML Use Case
- UML Activity

Vihje: osa suhteista (yhdysviivoista) on sidottu Visiossa kaaviotyyppiin, esim. UML -kaavioista ei löydy sopivaa viivaa kaksisuuntaiselle assosiaatiolle:

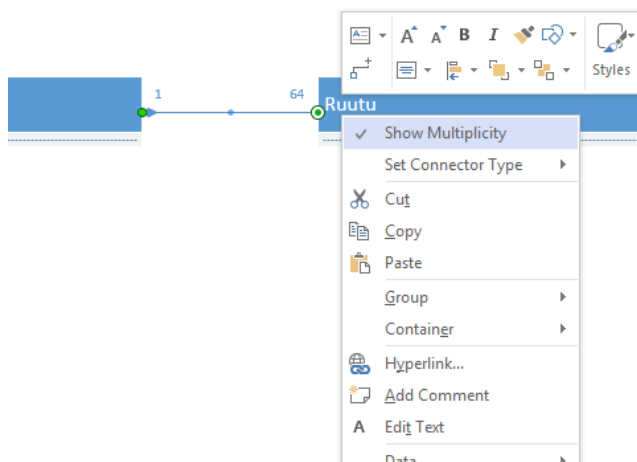
- kaksisuuntaisen assosiaationuolen saa tehtyä valitsemalla More Shapes - Business - Panel - Organization Chart Shapes ja muodoista Dynamic Connector. Sen muotoa voi vaihtaa vaihtamalla



## Lukumäärät

Assosiaatioilla, koosteilla ja muodosteilla voi olla lukumääräsuhteita.

Lukumääriä voidaan sijoittaa suhteisiin Visiossa klikkaamalla hiiren kakkospainikkeella muotoa ja valisemalla ponnahdusvalikosta Show Multiplicity.



## Käyttötapaus, Use Case

### Käyttötapaus

Käyttötapaukset (use cases) ilmaisevat järjestelmän palvelut. Ne kuvaavat järjestelmän toiminnallisuutta ja osoittavat ulkopuoliset vuorovaikutukseen osallistujat eli toimijat (actors).

Käyttötapauksen nimen tulee ilmaista mitä järjestelmä tekee, se kuvaa toimintaa. Yksi tapahtumasarja on käyttötapauksen käyttötilannetta.

Käyttötapauksen nimi kirjoitetaan isolla alkukirjaimella (UML).

### Toimija (Actor)

Toimija kommunikoi systeemin kanssa käyttötapausten kautta ja osallistuu käyttötapauksen suorittamaan palveluun.

Kaikki käyttötapaukset kommunikoivat jonkin toimijan kanssa, toimija voi olla ihminen tai toinen järjestelmä.

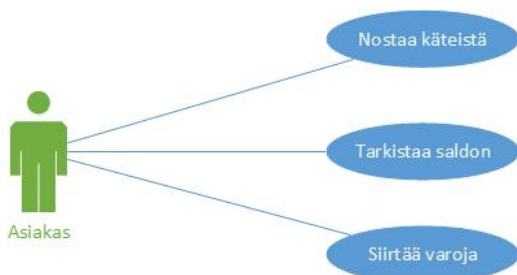
### Käyttötapauskaaviot tarkentuvat suunnittelun edetessä

Käyttötapausmalli muodostuu asteittain tarkentuvista käyttötapauskaavioista.

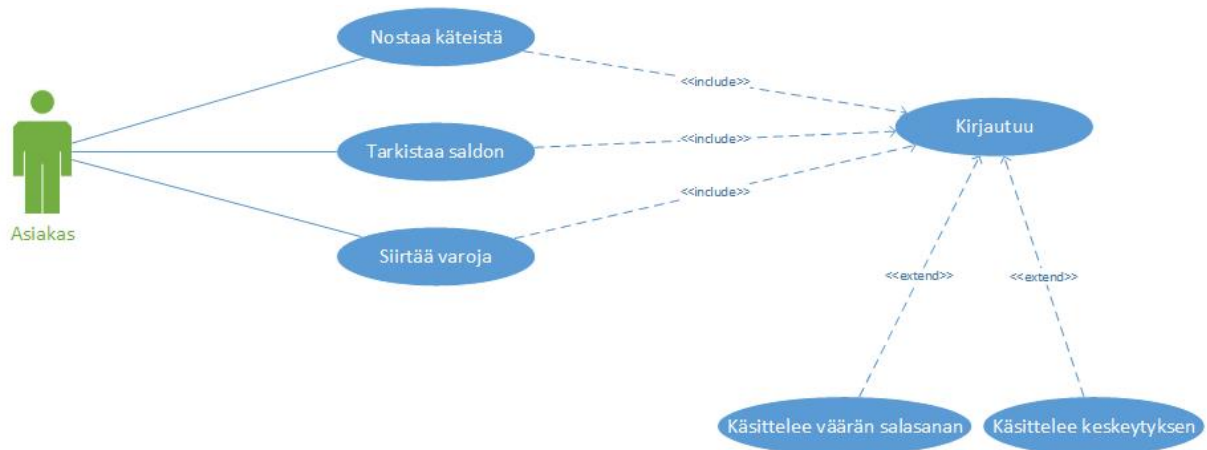
1. Käyttötapauskartta kuvaa tietojärjestelmän tarjoamat palvelut ja niihin osallistuvat toimijat
2. Riippuvuuskaaviot tarkentavat käyttötapaukset osakäyttötapauksiksi ja kuvaavat näiden välisiä suhteita
3. Käyttötapausten sanalliset kuvaukset sisältävät käyttötapauksen käyttötilanteet ja niiden kuvaukset eli tapahtumien kulku, muunnelmät ja poikkeustilanteet sekä näytön rakennekaaviot ja raporttimallit

Sovellamme projektissamme jakoa siten, että kohta 1 toteutetaan ehdotusvaiheessa, kohta 2 toisen prototyypin vaiheessa ja kohta kolme ohjelmallisia toimintoja sisältävän prototyypin yhteydessä.

### Esimerkki yksinkertaisesta käyttötapauskaaviosta



## Esimerkki riippuvuuskaaviosta



Jos käyttötapaus sisältyy toiseen, laajemmasta piirretään katkonuoli sisältävään käyttötapaukseen päin (kirjautuminen).

Jos käyttötapaus muuttuu toiseksi, nuoli piirretään laajenetusta käsitteestä kohti alkuperäistä tapausta.

Esimerkki käyttötapauksen kuvaamisesta

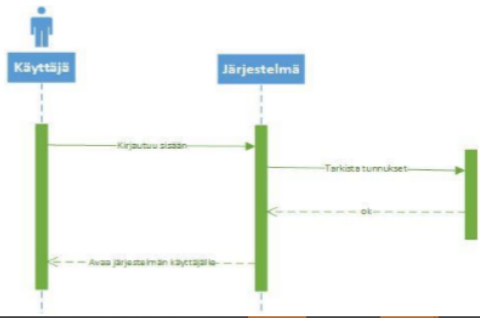
Tunniste	Kirjautuminen - sisäänkirjautuminen
Kuvaus	Käyttäjä kirjautuu onnistuneesti sisään
Alkuehto	Käyttäjän tiedot löytyvät tietokannasta, hänelle on valmis tunnus ja salasana
Normaali tapahtumien kulkuk	Käyttäjä kirjoittaa oman käyttäjätunnuksensa Login name -kenttään ja salasanan Password-kenttään. Painamalla Login -painiketta käyttäjä kirjautuu sisään järjestelmään.
Vaihtoehtoinen tapahtumien kulkuloppuehto	Jos käyttäjätunnus tai salasana tai molemmat ovat väärä, järjestelmään ei pääse kirjautumaan sisään. Ikkunat tyhjäntyvät. Käyttäjä on onnistuneesti kirjautunut sisään järjestelmään.
rikoisvaatimukset	Käyttäjätunnus ja salasana ovat oikeat.
Käyttäjät	Pääkäyttäjä ja käyttäjät
Järjestelmäversio	2.0
Näyttömalli	

Login

Password

Login

Sekvenssikaavio



Tekniikkaa

Visiolla luodaan mallipohja UML Use Case, kaikki tarvittavat symbolit ovat automaattisesti käytössä.