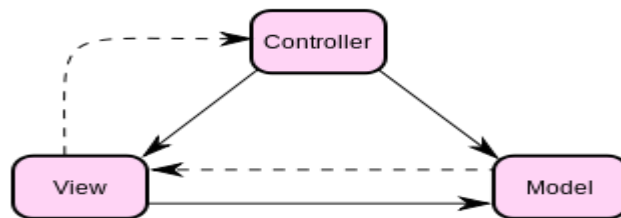


## Model-view-controller(MVC)

Este un model arhitectural utilizat în ingineria software. Succesul modelului se datorează izolării logicii de business față de considerentele interfeței cu utilizatorul, rezultând o aplicație unde aspectul vizual sau/și nivelele inferioare ale regulilor de business sunt mai ușor de modificat, fără a afecta alte nivele.



### Arhitectură

- Model - această parte a controlorului manipulează operațiunile logice și de utilizare de informație (trimisă dinainte de către rangul său superior) pentru a rezulta de o formă ușor de înțeles.
- Vizualizare - acestui membru al familiei îi corespunde reprezentarea grafică, sau mai bine zis, exprimarea ultimei forme a datelor: interfața grafică ce interacționează cu utilizatorul final. Rolul său este de a evidenția informația obținută până ce ea ajunge la controlor.
- Controlor - Cu acest element putem controla accesul la aplicația noastră. Pot fi fișiere, scripturi (eng. scripts) sau programe, în general orice tip de informație permisă de interfață. În acest fel putem diversifica conținutul nostru de o formă dinamică și statică, în același timp.

## Three-tire

O arhitectură pe trei niveluri este o arhitectură client-server în care logica procesului funcțional, accesul la date, stocarea datelor și interfața cu utilizatorul sunt dezvoltate și menținute ca module independente pe platforme separate. Arhitectura pe trei niveluri este un model de design software și o arhitectură software bine stabilită.

Cele trei niveluri dintr-o arhitectură pe trei niveluri sunt:

- Nivelul de prezentare - ocupă nivelul de vârf și afișează informații referitoare la serviciile disponibile pe un site Web. Acest nivel comunică cu alte niveluri prin trimiterea de rezultate către browser și alte niveluri din rețea.
- Nivelul aplicației - se numește de asemenea nivelul intermediar, logica sau logica de afaceri, acest nivel fiind tras de la nivelul de prezentare. Controlează funcționalitatea aplicațiilor prin efectuarea procesării detaliate.
- Data Tier - servere de baze de date unde informația este stocată și preluată. Datele din acest nivel sunt păstrate independent de serverele de aplicații sau de logica de afaceri.

## Database-centric architecture

Această arhitectură are mai multe semnificații distincte, în general legate de arhitecturile software în care bazele de date joacă un rol crucial. Adesea, această descriere are rolul de a contrasta designul cu o abordare alternativă. De exemplu, caracterizarea unei arhitecturi drept "bazată pe bază de date" poate însemna orice combinație a următoarelor:

- utilizând un sistem standard de gestionare a bazelor de date relaționale, în general , spre deosebire de structurile de date și metode de acces personalizate în memorie sau bazate pe fișiere .
- folosind o logică dinamică, de tip table -driven, spre deosebire de logica încorporată în programele compilate anterior . Folosirea logicii conduse de tabel, adică comportament puternic dictat de conținutul unei baze de date, permite ca programele să fie mai simple și mai flexibile.
- folosind proceduri stocate care rulează pe servere de baze de date , spre deosebire de dependența mai mare de logica care rulează în servere de aplicații de nivel mediu într- o arhitectură multi-nivel.
- o arhitectură generală a întreprinderii care favorizează modelele de date partajate permițând fiecărei aplicații să aibă propriul model de date idiosincronic.