



Desain dan Pemrograman Berorientasi objek

UML dan Framework MVC

Rosa A. S.



Rosa Ariani Sukamto

- Blog: <http://hariiniadalahhadiah.wordpress.com>
- Facebook: <https://www.facebook.com/rosa.ariani.sukamto>
- Email: rosa.ariani@upi.edu
- Website: <https://rosa-as.id>
- Youtube: <https://www.youtube.com/c/RosaArianiSukamto>

Pemodelan

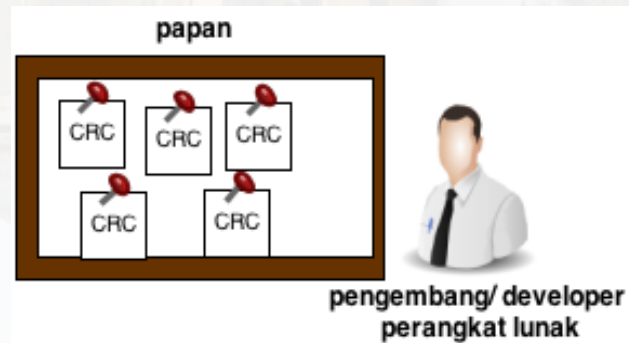
- Pemodelan adalah gambaran dari realita yang simpel dan dituangkan dalam bentuk pemetaan dengan aturan tertentu.
- Desain dengan memodelkan diperlukan agar perangkat lunak yang dibuat tidak seperti ilustrasi berikut:



CRC (*Component, Responsibility, Collaborator*)

- Pada awalnya *developer* menggunakan CRC yang ditempel di papan untuk mendesain kelas-kelas yang dibuat.
- CRC ini awalnya berupa kartu-kartu dari kertas.

Nama Kelas: Manusia	
Kelas Orang Tua:	
Kelas Anak:	
Tanggung Jawab	Kelas Terkait
Jenis kelamin Usia Menghitung usia pada tahun yang diberikan Kemampuan memasak	Masakan, Resep



UML

- Seiring berjalannya waktu maka lahirnya UML (Unified Modeling Language) yang berisi diagram-diagram untuk perancangan pembuatan perangkat lunak dengan paradigma pemrograman berorientasi objek.
- Pembuatan UML diharapkan semua *developer* dapat membaca desain yang dibuat, seperti halnya jika semua orang memahami satu buah bahasa.
- UML memiliki beberapa tingkatan dalam pemodelan, yang dibahas adalah UML dalam level teknis.

Diagram Use Case

- Dalam UML ada diagram use case dan kelas
- Diagram use case digunakan untuk memodelkan proses bisnis yang dalam struktur MVC (Model View Controller) masuk dalam desain awal kelas-kelas di dalam modul *Controller*

Diagram Kelas

- Diagram kelas berisi rancangan dari semua kelas yang dibuat, termasuk kelas *Controller* yang dikembangkan dari diagram use case.



MVC (Model View Controller)

- Kerangka kerja/*framework* ini muncul untuk memudahkan pengerjaan sebuah perangkat lunak dimana dikerjakan oleh tim yang dapat terdiri dari:
 - Programmer di bagian Controller
 - Database administrator programmer di bagian Model
 - Desainer di bagian View
- Sehingga dapat memudahkan proses pengembangan maupun pemeliharaan perangkat lunak
- Jika tampilan diubahpun juga tidak akan mempengaruhi kode program pada *controller* dan *model*

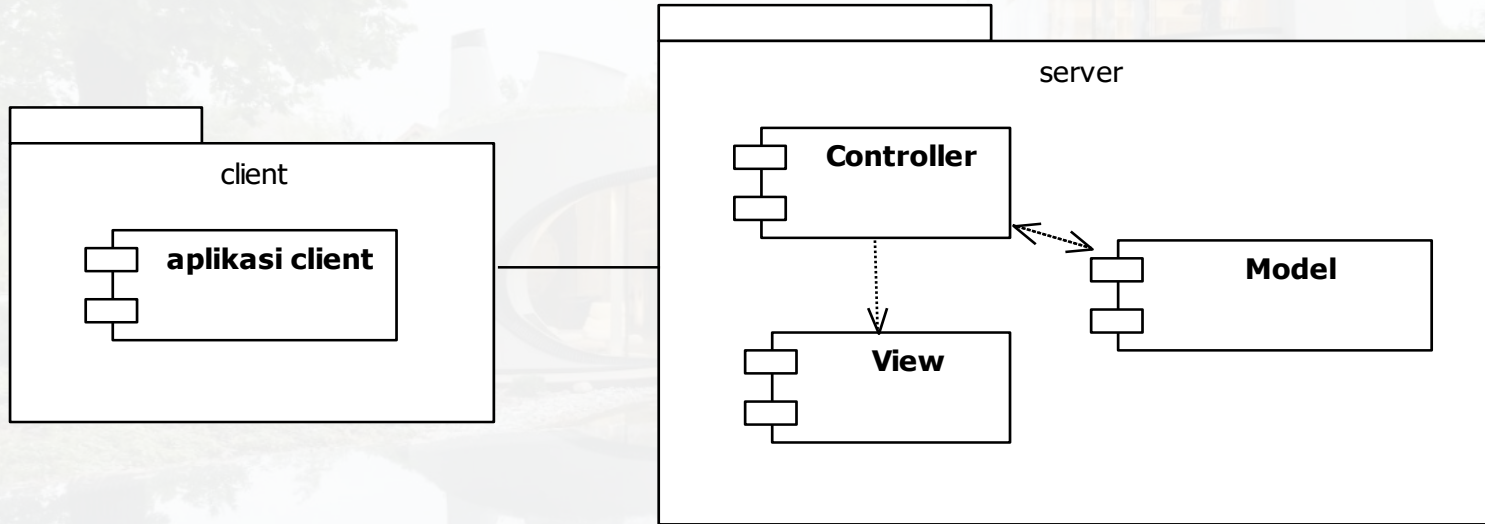
MVC (Model View Controller)

- Ada beberapa model alur dari MVC namun pada prinsipnya adalah bagian model adalah bagian yang tidak langsung diakses dari bagian awal yang diakses oleh *user*



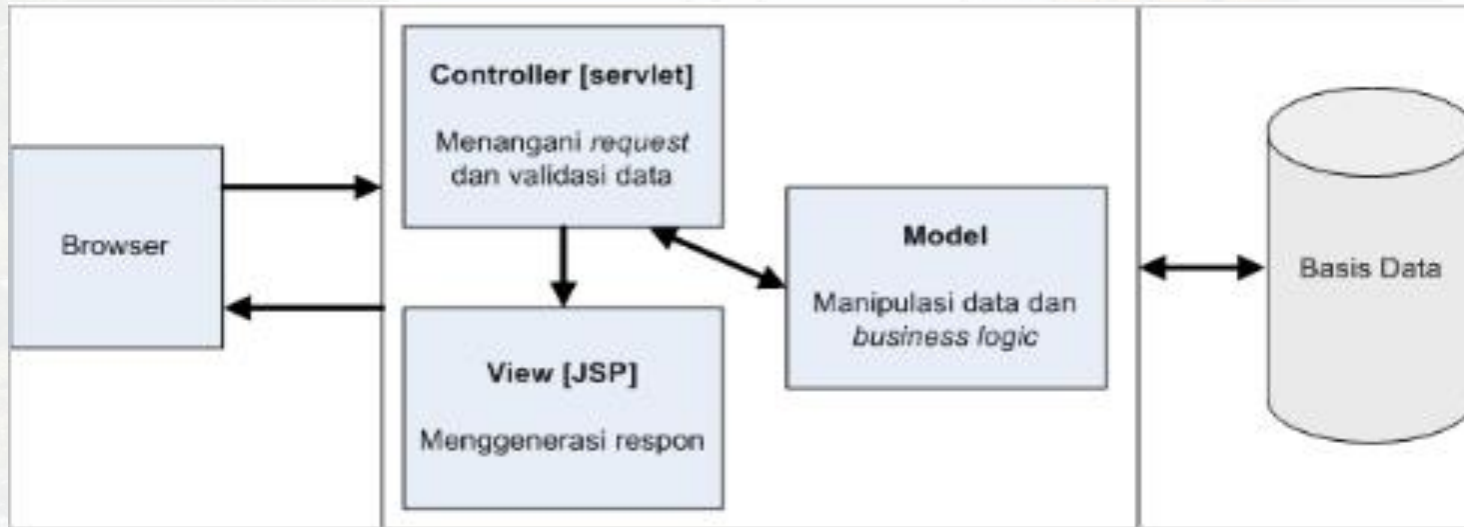
MVC (Model View Controller)

- Ilustrasi MVC



MVC (Model View Controller)

- Ilustrasi MVC pada framework Struts di Java



MVC (Model View Controller)

- Untuk lebih jelasnya silakan dibuat modul MVC sederhana dalam bahasa pemrograman Java dan PHP yang dilampirkan.

