

① Evolusi Perangkat Lunak tidak bisa dihindari karena

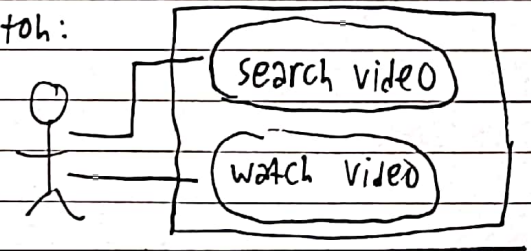
- kebutuhan baru akan muncul saat Perangkat Lunak digunakan.
- Lingkungan Bisnis selalu berubah
- Error-Error harus diperbaiki
- komputer dan peralatan baru selalu ditambahkan ke sistem
- kinerja atau keandalan sistem mungkin harus ditingkatkan

Contoh : Masalah dalam sebuah company adalah sulitnya mengelola dan menerapkan perubahan pada sistem yang membutuhkan perangkat lunak yang lebih mutakhir.

② Pengujian perangkat lunak tidak cukup dilakukan oleh tim developer saja karena menurut Sommerville pengujian software komersil memiliki 3 tahap, yaitu Development Testing, Release Testing, dan user testing dimana pihak pengguna juga melakukan pengujian terhadap perangkat lunak. Selain pengguna dan developer, product owner juga menguji kesesuaian terhadap awal dan Tester menguji aspek spesifik sesuai dengan tugas tester.

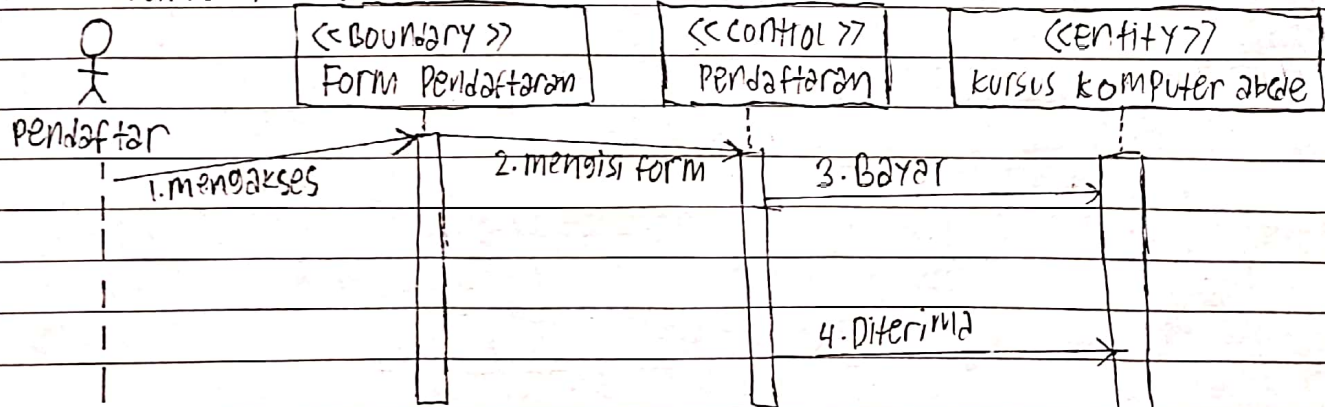
③ Use Case Diagram merupakan gambaran efek fungsionalitas yang diharapkan oleh sistem. komponen ini dapat membantu ketika sedang menyusun requirement pada sebuah sistem, kemudian mengkomunikasikan rancangan aplikasi tersebut pada konsumen. juga merancang test case untuk berbagai fitur yang ada. Use Case ini dapat diaplikasikan ke use case lainnya, sehingga duplikasi fungsionalitas dapat dihindari.

Contoh:



- ④ a. - Entity : kursus komputer abcde
- Boundary : form pendaftaran
- Control : pendaftaran

b. sequence Diagram



C.	<<Boundary>>		
	Form Pendaftaran		kursus komputer abcde
			+ Nama
	+ tulis(data)		- No HP
			- No KTP
			+ Set Nama
	<<control>>		+ Set No KTP
	Pendaftaran		+ Set No HP
	+ SimpanData()		