Karakteristike lošeg i dobrog softvera

- Dobar softver ne poseduje ove karakteristike, što znači da je:
  - *fleksibilan* (*flexible*): lako ga je održavati (menjati i proširivati)
  - *robustan* (*robust*): otporan na promene, tj. promene ne utiču na njegovu ispravnost
  - *prenosiv* (*portable*) i *ponovno upotrebljiv* (*reusable*): delovi se mogu iskoristiti u drugim aplikacijama bez mnogo ulaganja
- Naravno, ne postoji idealan softver i nijedan softver ne može biti neograničeno fleksibilan, robustan i prenosiv, ali je cilj da bude takav u opsegu *predvidivih* promena uslova i načina korišćenja u određenom domenu primene. Dobar i iskusan arhitekta softvera i poznavalac njegovog domena mogu ovo da predvide i procene
- Osnovni preduslovi za kvalitetan softver sa ovakvim karakteristikama:
  - jasna, jednostavna, pravilna, razumljiva, fleksibilna *arhitektura*
  - prave apstrakcije, dobra raspodela odgovornosti
  - modularnost, enkapsulacija
  - slabe zavisnosti između delova (apstrakcija, modula), jednostavni interfejsi
- Sve ovo ima svoju cenu, pa je u dobru arhitekturu i dizajn softvera potrebno uložiti više napora i vremena u početku, da bi se kasnije efekti videli sve većom brzinom; kod lošeg softvera je obrnuto
- Karaketeristika nepreporučljivog ponašanja: *žurba ka kodovanju* (*rush to code syndrom*)

Raspodela odgovornosti

- Preduslov za postizanje opisanih vrlina jeste *dobro razdvajanje brige* (*separation of concerns*) i *raspodela odgovornosti* (*distribution of responsiblity*) po apstrakcijama, klasama, modulima
- Elemente softvera (podatke, funkcionalnosti) treba grupisati u klase ili module po principu *jake kohezije i slabe sprege* (*strong cohesion/loose coupling*):
  - elementi unutar iste klase ili modula treba da budu *jako i tesno međusobno povezani* (*kohezija, cohesion*)
  - elementi iz različitih klasa i modula treba da budu *slabo spregnuti* (*loose coupling*)
- Ponekad se ovo izražava i kao *princip jedinstvene odgovornosti klase* (*single responsibility principle*): klasa treba da ima ograničenu odgovornost, tako da postoji samo jedan razlog za njenu izmenu
- Ako se različite odgovornosti mogu nezavisno menjati, postoji ozbiljan razlog za razdvajanje tih odgovornosti u različite klase
- Klasa svakako ne treba da ima previše odgovornosti, posebno ako su one slabo povezane ili nepovezane - raspodela odgovornosti je jedan element *objektne dekompozicije*