Architect

s/w design principles

effectively manage to design

reduce errors

1 problem partitioning:divide problem into manageable pieces

Benefit:s/w simple, easy understand, test, modify

2 abstraction : consider component at abstract level not consider internal details of implementation

Types

Functional : detail of algo to accomplish function is not visible

Data: detail of data element are not visible to the users of data

3 modularity : division of s/w into different-2 modules different name and address and integrate later

4 top down and bottom up strategy : strategy of design to organize program modules in such a way such that can develop easily

Functional independence: function that perform one task and do not excessively interact with other module

Benefit: easy to maintain, test,reduce errors,can be reused.

Measured using two criteria

Cohesion: measure func strength

Coupling:measure independence among modules

2 info hiding: data include in module is inaccessible by other modules if not require

Benefit: errorless, modification easily

4 top-down: identify main component and decompose it into detail sub-component

Down-up: begins with lower detail and move towards the hierarchy(in case of existing system it is suitable

Hld: it is overall system design. System architecture,database design,brief description on services,platform it is created before lld

Converts client requirement into high level sol overall architecture ,sol architect

Hld by design team, client, and review team input is srs( software requirement specification)

Hld output is database design and functional design

LLd : Low level design

Detailing of hld

Refers to component level design process

Micro level design

Lld by designer and developers

Website architecture hierarical structure of web page

Home ->about,blog,services

About->team,history

Blog->topic1,2,3

Services-> service 1,2,3

Best practice

1 simple top level menu

2 keep url simple and user friendly