ARCHITECTURAL DESIGN

**WebApp Architectural Design**

* client-server app : typically multi-layered architectures
  + UI / View layer – designed around characteristics of web browser running on either PC or Mobile devices
  + Controller layer - directs info. flow to/from browser based on biz. rules
  + Content / Model / Data Layer – may also contain biz. rules
* Biz. rules – PHP or Javascript
* allocate features to client/server according to security + usability reqs
* Influenced by content structure + its physical location

**MobileApp Architectural Design**

* also typically multi-layered architectures (UI / Biz / Data Layer)
  + Thin Web-based client – only UI in mobile device & Biz + Data in server
  + Rich Web-based client – all 3 layers in client
* Mobile devices differ from one another (screen size, input device, OS …) 🡪 considerations influencing arch. design of mobile apps :
  + type of web client (thin or rich) to be built
  + device category (smartphone / tablets) supported
  + degree of connectivity (occasional / persistent) + bandwidth required
  + constraint imposed by platform
  + importance of reuse & maintenance
  + device resource constraints (mem size, battery life, processor speed)

COMPONENT-LEVEL DESIGN

**WebApp Component-Level Design**

* Boundary between content and function = blurred!
* WebApp component = either:
* well-defined cohesive function 🡪 manipulates content / provides computational or data processing for an end- user
* cohesive package of content and functionality 🡪 provides the end-user with some required capability

🡺 often incorporates content / functional design

WebApp Component-Level **Content Design**

* Focuses on content objects and how they may be packaged for presentation to the end-user
* Often contents can be manipulated individually! BUT size and complexity increase 🡪 need formal representations for easy reference + manipulation
* For highly dynamic content (e.g. auction) 🡪 establish a clear structural model to incorporate content components

WepApp Component-Level **Functional Design**

* WebApps provide sophisticated processing functions
* perform dynamic processing to create content + navigational capability
* provide biz. domain appropriate computation or data processing
* provide database query and access
* establish interfaces with external corporate systems
* WebApp functionality is delivered as a series of comps dev. in parallel
* @ arch. Design, content + functionality 🡪 create a functional arch.
* functional arch. = represents functional domain of the WebApp and describes key functional comps + how they interact with each other

**Component-Level Design for Mobile Apps**

* UI layer (small display) - be more selective in choosing content to display (tailor content to a specific user groups)
* Biz. & Data layer - impl. by composing web or cloud service components
* Arch. design :
  + thin web-based client – connectivity issues must be considered
  + rich client – connectivity issues are not significant concern
* Porting desktop app 🡪 mobile app
  + review biz. layer comps for satisfying non-func requirements

**Traditional Component-Level Design**

* each block of code has a single entry at the top & exit at the bottom
* only 3 control structures are required: sequence, condition (if-then-else), and repetition (looping)
* enhance readability, testability, maintainability 🡪 reduces complexity