I. Dix Ch. 5 - Interaction Design Basics

1. What is design?
   i. Achieving goals within constraints.
      a. Goals: What is the purpose of the product?
      b. Constraints: time, materials, standards, etc..
      c. Trade-off: choosing the appropriate goals and constraint, so that one can be sacrificed.

2. Golden rule of design:
   i. Understanding computers its limitations, capacities, tools etc.
   ii. Understanding Humans its social, psychological aspects etc.

3. People make mistakes, but this is not human error, its human nature. A design must take this into consideration, as people are adaptable.

4. Put the user first, keep the user in the center and remember the user at the end.

5. Usability interface must always be built first.

6. Process of design
   i. Requirements: What is wanted?, Ethnography
   ii. Analysis: The above-mentioned results are ordered in some way.
   iii. Design: Take record of design at stages with notations. Describe interactions
   iv. Iteration and prototyping
   v. Implementation and deployment: Hardware, code, manual and documentation, if we are happy with design.

7. User focus
   i. Who are they?- young/old, experienced/novice, etc.
   ii. not like us
   iii. Talking to them- participatory design, understand their needs, work context and trust.
   iv. Watch: observe them and take notes and discuss them with users
   v. Imagination: should not make assumptions.

8. Scenarios
   i. stories of interaction - save/quit button
   ii. Communication with designers, clients and users.
   iii. used for validating models
   iv. screen-shots and pictures gives sense
   v. linear process can be understood
   vi. no alternatives.

9. Navigation Design
   i. Levels
      a. widgets
      b. windows/screens
      c. navigation within application
      d. environment
   ii. goal seeking behavior- user
iii. after each step - users must realize that they are getting closer to their goal
   a. where you are?
   b. what you can do?
   c. where you are going? - what will happen?
   d. where you have been? - what you have done?

iv. Hierarchical organization of functional tasks
   a. can be used for design
   b. must be deep enough to store in working memory

10. must be grouped in some logic.
   
i. Dialog
      a. system can have multiple paths and loops
      b. network diagram showing tasks and states

ii. Screen Design and Layout - Ask, think and design
   a. Tools
      a. Grouping - tasks that are logically same must be grouped together
      b. order - match the natural order of groups with the order of design
      c. decoration - line division, font style
      d. alignment - numbers, lists
      e. white space
   b. User action and control
      a. entering info in form based layout - logical layout and alignment
      b. active vs passive screens
      c. affordances - must mimic real world objects directly or must emulate them
   c. Appearance
      a. Presenting information - maps, tables
      b. Aesthetics vs utility

11. Iteration and prototyping
   i. start with storyboards and paper design
   ii. working prototype
   iii. evaluate the above
   iv. list the faults and changes
   v. Iterate the process with the changes
II. Dix Ch. 7 Design Rules

1. Principles
   i. Learnability - ease of use and learn
   ii. Flexibility - multiplicity of exchange of info
   iii. Robustness - level of support for the user

2. Learnability - learn and maximize the performance for novice users
   i. Predictability
      a. No surprises
      b. tasks must be related to info presented, so users do not have to think
      c. knowledge of which actions can be performed - operational visibility
   ii. Synthesizability
      a. assumes users have some mental model of the system
      b. building interactions based on the above.
      c. notifying the users of the change in internal state
      d. eventual honesty
   iii. Familiarity
      a. Correlation between users existing knowledge and knowledge for interaction with system
      b. word processor - type writer - metaphor
   iv. Generalizability
      a. extending knowledge of similar un-encountered situations.
   v. Consistency
      a. similar behavior for similar tasks

3. Flexibility
   i. Dialog initiative
      a. system pre-emptive and user pre-emptive dialogs
      b. the latter not desirable, as user can get lost.
   ii. Multi-threading
      a. concurrent and interleaved multi threading.
      b. windows representing tasks.
   iii. Task migratability
      a. transfer of control between system and users
      b. spell-checking
   iv. substitutivity
      a. input variations
      b. output variations
      c. equal opportunity - if you can see it, you can use it!
   v. customizability
      a. not in the system side
      b. to increase adaptability

4. Robustness
   i. Observability
a. observing internal state by user
b. browsability - allowing users to explore the internal state
c. defaults - passive recall to prevent error
d. reachability
e. persistence - duration of communication act and ability of user to make use of that effect.
f. recoverability - forward and backward error recovery. can be initiated by system or user. should reflect the work
   ii. responsiveness - rate of comm. between user and system- short is desirable
   iii. task confidence
5. Standards
   i. BSI(British standard institution) and ISO
6. Guideline
   i. Smith and mosier guielines are
   ii. data entry
   iii. data display
   iv. sequence control
   v. user guidance
   vi. data transmission
   vii. data protection
III. Dix Ch. 8 - Implementation Support

1. Elements
   i. WIMP
   ii. Abstract terminal - imaging model
   iii. each abstract terminal is a process

2. Architecture
   i. final proposal - client-serve architecture
   ii. Ex. X window system with X protocol, making X device independent
   iii. X server - allows(denies) access to display
   iv. interprets requests
   v. Policies:
      a. rules for transferring data b/w clients
      b. method of selecting active client
      c. layout schemes for overlapping windows

3. Programming the application
   i. Read-evaluation loop (MAC) - server is concerned with sending the appropriate events to clients, clients are ready for any type of events
      cumbersome as programmer must execute this control for every task.
   ii. Notification based- central notifier receives events, apps does a callback for these.
   iii. not as cumbersome as the control is at notifier.

4. UI MGMT. SYS.
   i. conceptual architecture
      a. Portability same app in different systems.
      b. Reusability - reduce dev. costs
      c. Multiple interfaces - increase flexibility
      d. customization - increase effectiveness
   ii. techniques for implementing whilst in the constraint
   iii. support techniques for managing, implementing and evaluating at run-time.
   iv. Logical components
      a. presentation
      b. dialog control
      c. app interface
   v. vs MVC

5. Implementation considerations
   i. Menu networks - comm. b/w apps and presentation in terms of menus and sub-menus
   ii. Grammar notations - dialog and feedback b/w apps and presentations
   iii. state transition diagram
   iv. event languages
v. declarative languages - shared database of relation b/w apps and presentations
vi. constraints
vii. graphical specification