SASTRA
ENGINEERING · MANAGEMENT · LAW · SCIENCES · HUMANITIES · EDUCATION
DEEMED TO BE UNIVERSITY
(U/S 3 of the UGC Act, 1956)
THINK MERIT | THINK TRANSPARENCY | THINK SASTRA

Chapter 7

Design Prototyping and Construction

B.Tech CSBS
VII Semester

Handled by
Smt. T.M. Latha

Outline

1. Prototyping and Construction
2. Conceptual Design
3. Physical Design
4. Tool Support

Prototyping and Construction

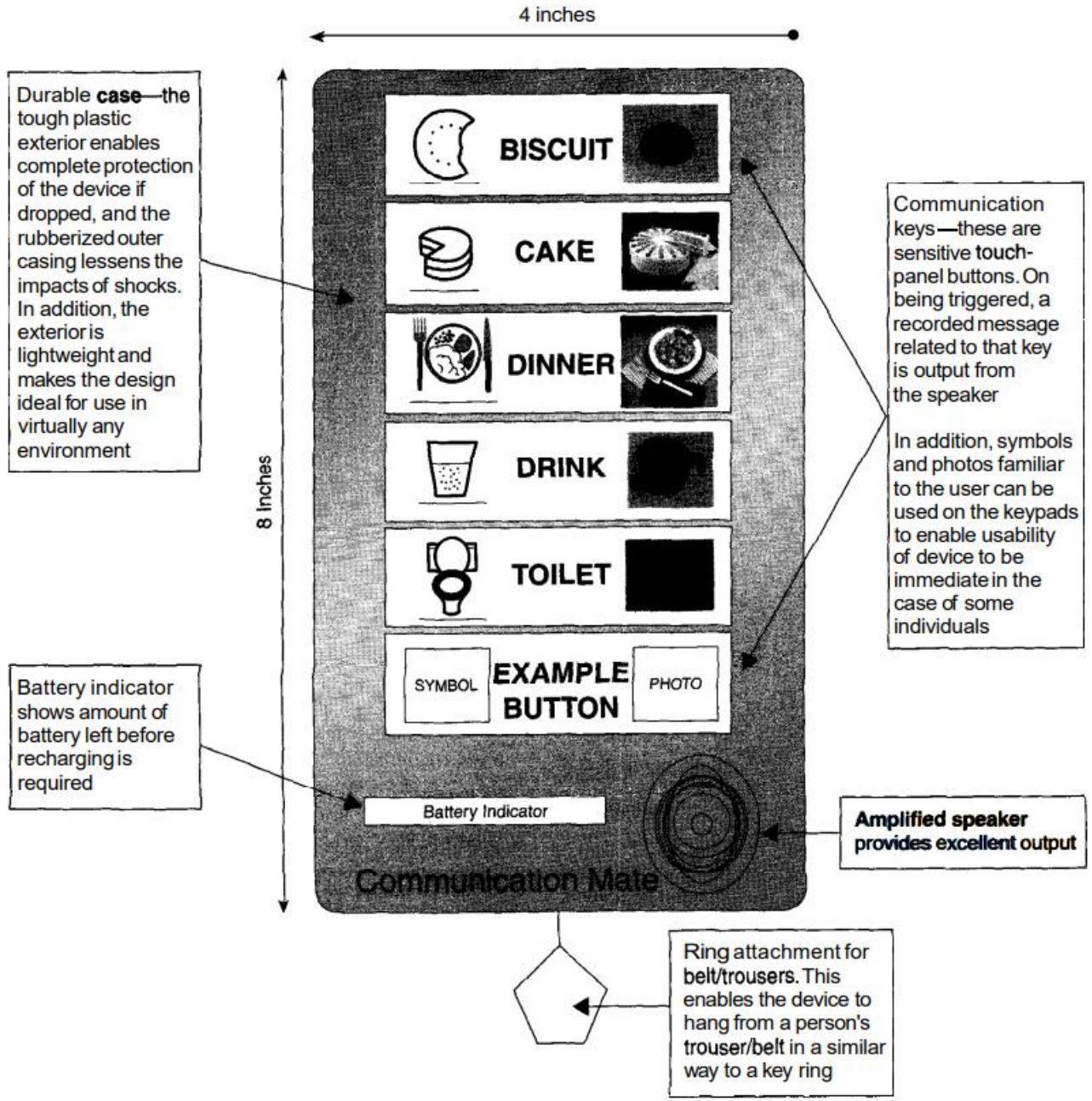
- Prototype
  - Anything
    - from a paper-based storyboard to complex piece of software
    - from cardboard mockup to a molded or pressed piece of material

  - Allows stakeholders
    - to interact with an envisioned product
    - To gain some experience of using it in a realistic setting
    - To explore imagined uses

  - Eg.
    - PalmPilot
      - Hawkin
        - Carved up a piece of wood about the size and shape of the device
        - Used to carry this wood around with him
        - Pretended to enter information into it

- Prototype benefits
  - Useful aid to discuss ideas with stakeholders
  - Communication device among team members
  - Reflects in design
  - Answers questions
  - Supports designers in choosing b/w alternatives

- Prototype
  - Serve a variety of purposes
    - To test out the technical feasibility of an idea
    - Clarify vague requirements
    - Do some user testing and evaluation
    - Check certain design direction is compatible with rest of the system


4 inches
8 Inches
Durable **case**—the tough plastic exterior enables complete protection of the device if dropped, and the rubberized outer casing lessens the impacts of shocks. In addition, the exterior is lightweight and makes the design ideal for use in virtually any environment
BISCUIT
CAKE
DINNER
DRINK
TOILET
SYMBOL
EXAMPLE BUTTON
PHOTO
Communication keys—these are sensitive touch-panel buttons. On being triggered, a recorded message related to that key is output from the speaker
In addition, symbols and photos familiar to the user can be used on the keypads to enable usability of device to be immediate in the case of some individuals
Battery indicator shows amount of battery left before recharging is required
Battery Indicator
Amplified speaker provides excellent output
Communication Mate
Ring attachment for belt/trousers. This enables the device to hang from a person's trouser/belt in a similar way to a key ring

Low-fidelity Prototyping

- does not look very much like the final product

- Eg:
  - Use of different materials from the intended final version

- Useful bcoz they tend to be
  - Simple
  - Cheap
  - Quick to produce

- Explained using
  - Storyboarding
  - Sketching

Storyboarding

- Consists of series of sketches
- Shows how a user might progress through a task

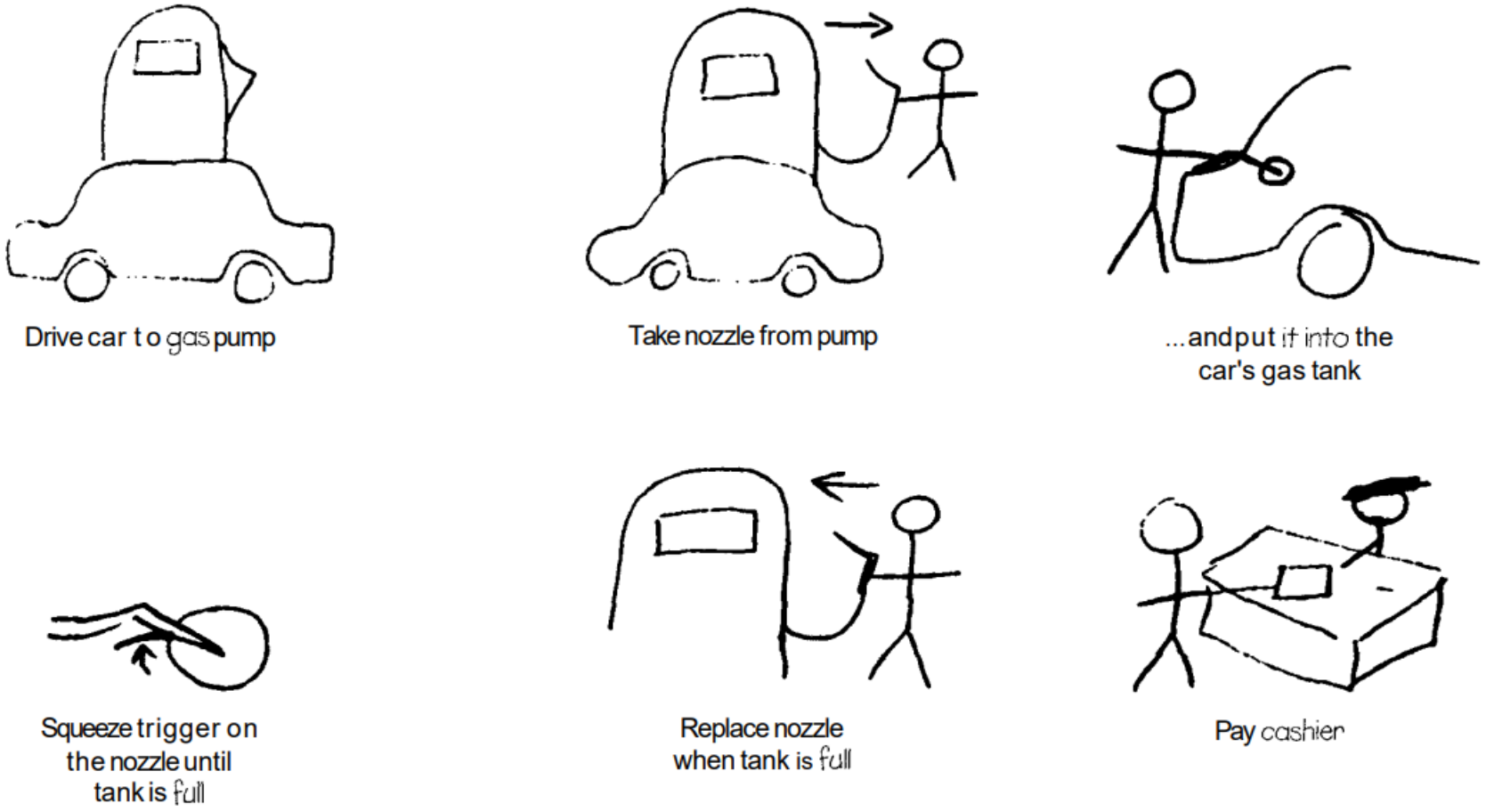


Figure 8.4 A storyboard depicting how to fill a car with gas.

Sketching

- Low-fidelity prototyping relies on sketching
- Symbols and icons for elements to sketch
  - Boxes
  - Stick figures
  - Stars

Storyboarding

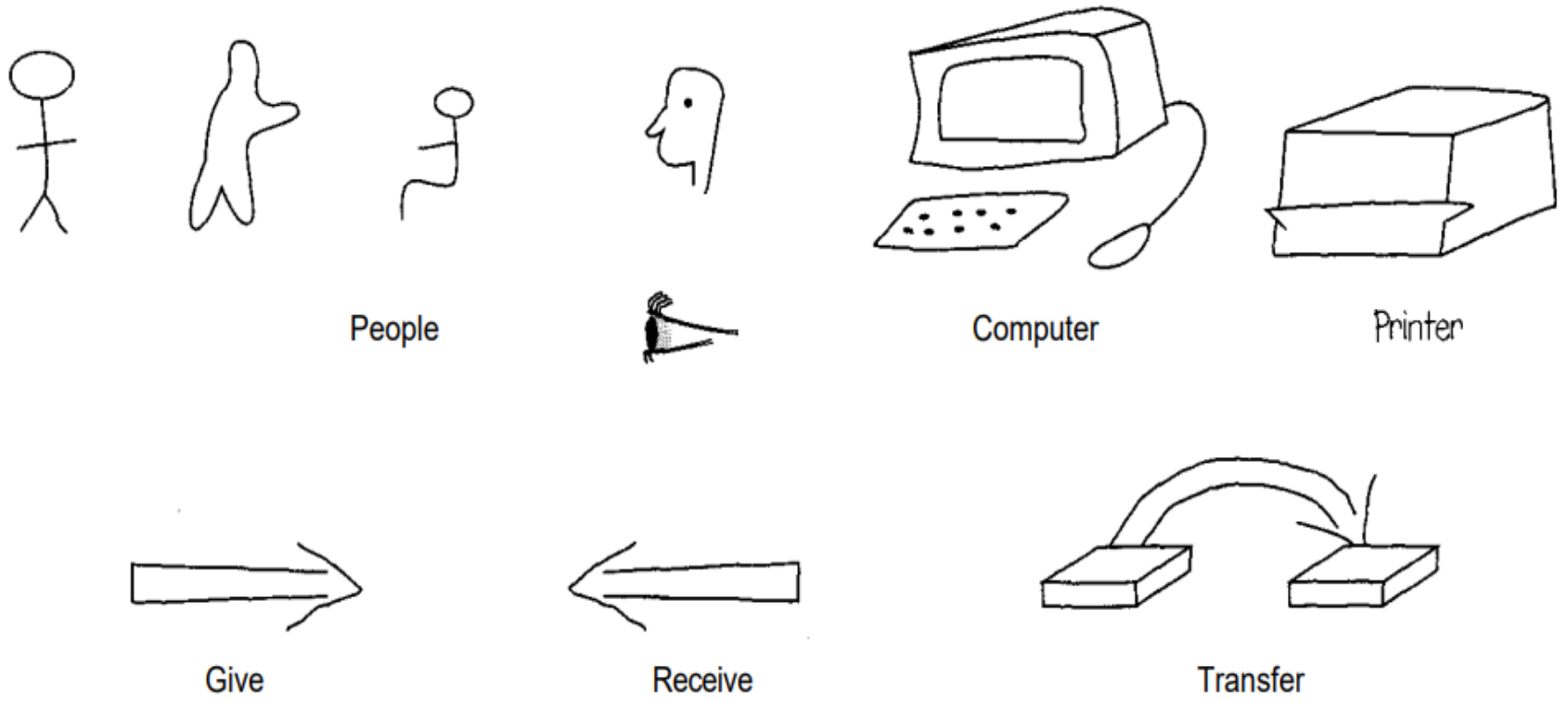


Figure 8.3 Some simple sketches for low-fidelity prototyping.

Wizard of Oz

- Software based low-fidelity prototyping
- User interacts with the software as though interacting with the product
- Computer is connected to other machine
  - Human operator simulates the s/w's response to the user

High-fidelity Prototyping

- Uses materials you would expect to be in the final product
- Produces a prototype that looks like final thing
- Common prototyping tools
  - Macromedia Director
  - Visual Basic
  - Smalltalk

Table 8.1 Relative effectiveness of low- vs. high-fidelity prototypes (Rudd et al., 1996)

Type	Advantages	Disadvantages
Low-fidelity prototype	• Lower development cost. • Evaluate multiple design concepts. • Useful communication device. • Address screen layout issues. • Useful for identifying market requirements. • Proof-of-concept.	• Limited error checking. • Poor detailed specification to code to. • Facilitator-driven. • Limited utility after requirements established. • Limited usefulness for usability tests. • Navigational and flow limitations.
High-fidelity prototype	• Complete functionality. • Fully interactive. • User-driven. • Clearly defines navigational scheme. • Use for exploration and test. • Look and feel of final product. • Serves as a living specification. • Marketing and sales tool.	• More expensive to develop. • Time-consuming to create. • Inefficient for proof-of-concept designs. • Not effective for requirements gathering.

- Problems of high-fidelity prototyping
  - They take too long to build
  - Reviewers and testers tend to comment on superficial aspects rather than content
  - Developers are reluctant to change something they have crafted for hours
  - A software prototype can set expectations too high
  - Just one bug in a high-fidelity prototype can bring the testing to a halt

Conceptual design: moving from requirements to design

- Three perspectives for developing a conceptual model
  - Which interaction mode?
  - Is there a suitable interface metaphor?
  - Which interaction paradigm?

Which interaction mode?

- Instructing
- Conversing
- Manipulating
- Navigating
- Exploring and browsing

  - Eg:
    - Computer game - manipulating and navigating style

    - Drawing package – instructing and conversing

Is there a suitable interface metaphor?

- Combining familiar knowledge with new knowledge
  - Evaluation of interface metaphors
    - How much structure does the metaphor provide?
      - A good metaphor will require structure, and preferably familiar structure
    - How much of the metaphor is relevant to the problem?
    - Is the interface metaphor easy to represent?
    - Will your audience understand the metaphor?
    - How extensible is the metaphor?

Which interaction paradigm?

- Ubiquitous computing
- Pervasive computing
- Wearable computing

Expanding the conceptual model

- With
  - What functions will the product perform?
  - How are functions related to each other?
  - What information needs to be available?

What functions will the product perform?

- How the task will be divided up b/w human and machine
- Eg.
  - Shared calendar
    - System: suggest dates when a set of people are able to meet?
      - but is that as far as it should go?
        - Should it automatically book the dates,
        - should it email the people concerned informing them of the meeting
        - asking if this is acceptable?
    - Human:
      - Responsible for checking this out?

- Scenarios
- Essential use cases
- Use cases
  - Help clarify answers to these questions

- Task allocation
  - Deciding what the system will do
  - What must be left for the user

- Another aspect concerns
  - Device
  - Software
  - user

How are the functions related to each other?

- Functions may be related temporarily
  - Eg.
    - One must be performed before another
    - Two can be performed in parallel

- In OO software, we draw
  - Sequence diagram

What information needs to be available?

- What data is required to perform a task?
- How is the data to be transformed by the system?
- Eg:
  - Task of booking a meeting among a set of people using shared calendar
    - Who is to be at the meeting?
    - How long the meeting is to take?
    - What its location should be?

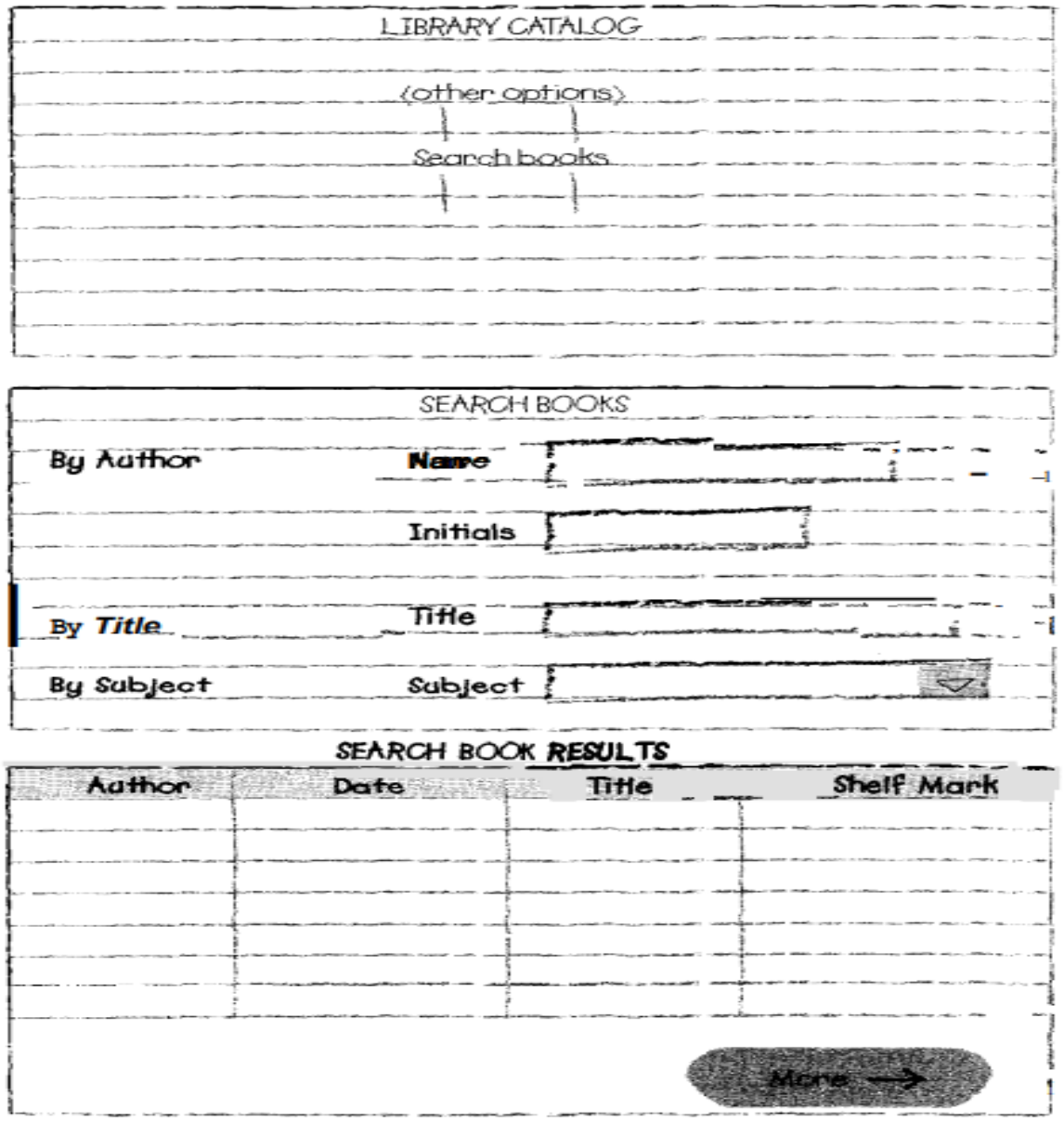


Figure 8.10 A card-based prototype for borrowing a book in the library catalog system.

Physical Design: getting concrete

- Designing the interface
  - Screen/keypad design
  - Icons to use
  - How to structure menus
- There is no rigid border b/w conceptual design and physical design

Guidelines for physical design

- Physical interface must not conflict with user's cognitive process
- Design the physical form with human characteristics such as
  - Attention
  - Perception
  - Memory
  - Eg:
    - To help avoid memory load
      - interface should list options instead of making users remember long list of possibilities

- Golden rules of interface design

  - Strive for consistency
    - ‘File’ menu – top left corner
    - Action that results in data loss – confirmation

  - Enable frequent users to use shortcuts
    - Word processor- short cut /quick keys

  - Offer informative feedback
    - Instead saying ‘Error-404’ make it clear as ‘The URL is unknown’
    - Feedback is influenced by kinds of users
      - What is meaningful to a scientist may not be meaningful to a manager/architect

  - Design dialogs to yield closure
    - Make it clear when an action has completed successfully: ‘printing completed’

- Offer error prevention and simple error handling
  - System should be forgiving about the errors made
  - System should support the user in getting back on track

- Permit easy reversal of actions
  - Provide an ‘undo’ key where possible

- Support internal locus of control
  - Users feel more comfortable, if they feel in control of interaction

- Reduce short-term memory load
  - Offer users options than ask them to remember information

Different kinds of widget

- Interfaces made up of
  - Widgets
  - Elements
    - Dialog boxes
    - Menus
    - Icons
    - Toolbars
      - Should be designed/chosen from predefined set of widgets

- Sometimes decisions are made through style guide
  - Dictates look and feel of the interface
    - Commercially produced (Windows style guide)
    - May be internal to a company

- Style guides
  - Eg.
    - Which menu is always on the right-hand side of the toolbar?
    - What icon is used to represent ‘close’ and ‘print’?
    - Which typeface is used in menus and dialog boxes?

- Three aspects of interface design
  - Menu design
  - Icon design
  - Screen layout
    - Applicable to wide range of interactive products
      - Standard desktop interfaces
      - Mobile communicator functions
      - Microwave ovens

- Menu design
  - Provide users with a choice
    - Choice of commands
    - Choice of options related to a command
  - Provide the means for user to perform actions
  - Designed as
    - drop-down
    - Pop-up
    - Single-dialog menus
  - Grouping can be used to good effect in menu design
    - Menu can be divided into collections of related items
      - Each collection can be separated from others
  - Opposite operations should be clearly separated to avoid accidentally losing work
    - such as ‘quit’ and ‘save’

- Menu names need to be
  - Short
  - Clear
  - Unambiguous
  - Distinguishable
- The space for listing them is restricted
  - “store the file in system’ – ‘save’
- Need to consider logical groupings
  - Query entry
  - Add entry
  - Edit entry
  - Move entry
  - Delete entry

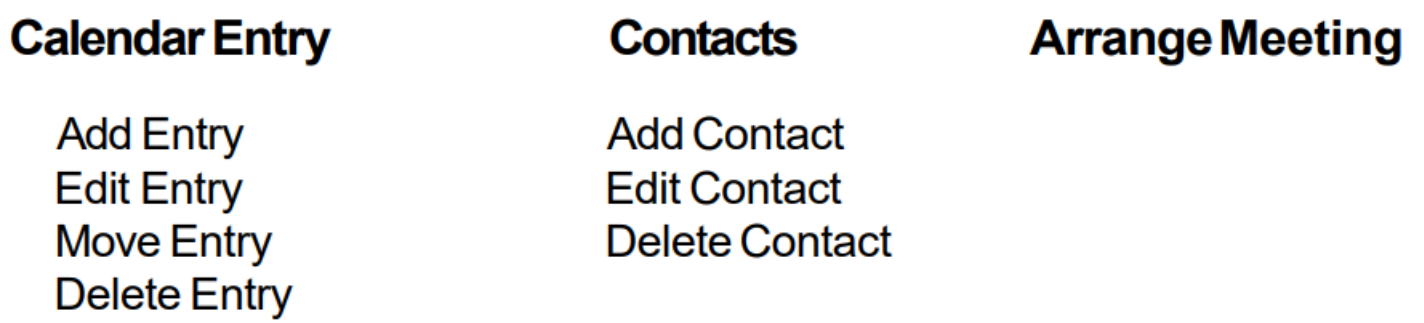
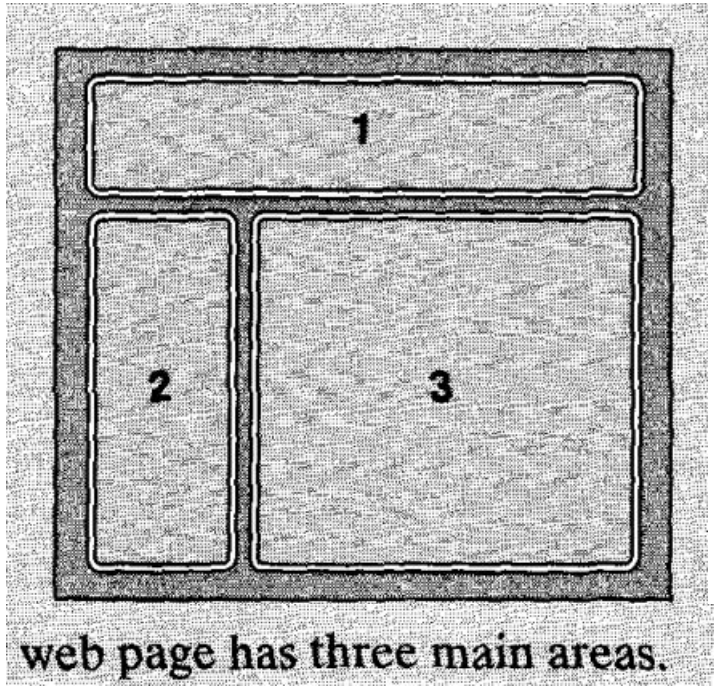


Figure 8.13 Possible menu groupings for the shared calendar system.

- Icon design
  - Icons are
    - Cultural and context-specific
    - Drawn on
      - Existing traditions
      - Standards
  - Designing good icon takes time
  - Concrete objects or things are easier to represent as icons
    - Eg:
      - Pair of scissors to represent ‘cut’
      - Binoculars/magnifying glass for ‘search’
  - users can readily perceive their meaning through icon

- Screen design
  - Two aspects
    - How task is split across number of screens
    - How the individual screens are designed
  - How task is split across number of screens
    - Supported by taking reference to task analysis
      - Tasks
        - Subtasks
          - Plans of action
  - How the individual screens are designed
    - Draw more clearly from visual communication principles
      - Eg:
        - Drawing user attention by
          - Color
          - Motion
          - Boxing and grouping to aid understanding and clarity



web page has three main areas.

Tool Support

- Nine facilities that user interface software tools should provide

- help design the interface given a specification of the end users' tasks
- help implement the interface given a specification of the design
- create easy-to-use interfaces
- allow the designer to rapidly investigate different designs
- allow nonprogrammers to design and implement user interfaces
- automatically evaluate the interface and propose improvements
- allow the end user to customize the interface
- provide portability
- be easy to use

THANK YOU