Why the User Interface Matters

- Human-computer interaction (HCI) is the study of how humans interact with computer systems.
- Many disciplines contribute to HCI, including CS, psychology, ergonomics and human factors, AI, engineering, as well as philosophy, sociology and anthropology.
- When users interact with a computer system, they do so via a UI.
- The textbook explores how to design good UIs, that are:
    - Easy to use & easy to understand
    - Meet the needs of the intended users and support them in their tasks

What is a Good User Interface Design?

- A good UI design:
    - Encourage easy, natural, and engaging interaction between a user and a system
    - Focus on **usability** ➔ whether a UI is good or bad in relation to its usability
- UI/UX impacts whether users will (correctly) use the application
    - Applications must be “**usable**” and “**useful**”
- Useful
    - Usability: Easy to use and understand (usable)
    - Utility: Satisfy User needs (user satisfaction)

Usability (ISO 9241) - The extent to which a product can be used by **specified** users to achieve specified goals with:

- **Effectiveness:** “the accuracy and completeness with which users can achieve goals in particular environments”
- **Efficiency:** “the resources expended in relation to the accuracy and completeness of the goals achieved”
- **Satisfaction:** “the comfort and acceptability of the work system to its users and other people affected by its use”

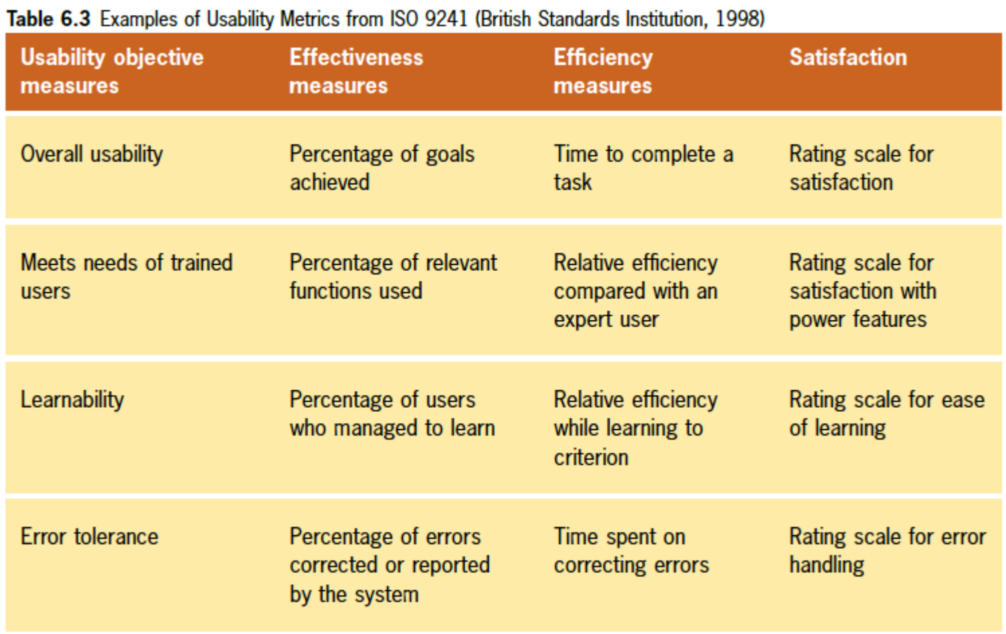


Table 6.3 Examples of Usability Metrics from ISO 9241 (British Standards Institution, 1998)

Usability objective measures	Effectiveness measures	Efficiency measures	Satisfaction
Overall usability	Percentage of goals achieved	Time to complete a task	Rating scale for satisfaction
Meets needs of trained users	Percentage of relevant functions used	Relative efficiency compared with an expert user	Rating scale for satisfaction with power features
Learnability	Percentage of users who managed to learn	Relative efficiency while learning to criterion	Rating scale for ease of learning
Error tolerance	Percentage of errors corrected or reported by the system	Time spent on correcting errors	Rating scale for error handling

Utility ≫ User Centered Design (UCD) - is an approach used to design UI and involve user

- Understanding the users of a computer system, of the tasks, the environment
- **Principles**
    - User **involvement** in development stages
    - Design **iteration**

  - **Multi-disciplinary** design teams
  - Appropriate **allocation** of function between user & system
- **Activities**
  - Understand & specify **context of use**
  - Specify user & organization **requirements**
  - Produce **prototypes**: design solutions
  - **Evaluate** designs with users against requirements

Part 2 – Requirement Gathering Techniques

Observing Your Users

- **Direct Observation (Cheapest)**
    - Field – Work environment, home
    - Controlled – Usability Lab
    - Chance to lose info, no record to look back on
    - Provide you with an insight into the users, their tasks, and the environment.
- **Indirect Observation (Costly – Need to buy equipment)**
    - Video recording
    - Time Consuming
    - Permanent Record

Interviewing Your Users –Talking or Questioning

- **Structured interview**
    - Specific set of topics/questions asked to each interviewee
- **Unstructured (Flexible) interview**
    - No list of questions, Free form
    - Better if you want to measure user opinion about something
- Combination of both is preferable. Allows for good view of user opinion, while allowing to focus on some needs/ specific aspects
- Avoid leading questions – questions that push interviewee in favor of one direction or another (Skewed)

Questionnaires & Surveys (clear and unambiguous questions, avoid complicated questions)

- **Closed Questions**
    - Select from a list of answer choices
    - Yes/no, T/F, or a-d
    - Likert Scale (5-point scale: strongly disagree ➔ strongly agree)
    - Some form of a **rating scale** associated
- **Open Questions**
    - Usually used where there are no predetermined answers
    - “What do you . . .,” , “How do you . . .,” , and “What ways . . ..”
    - More time consuming to analyze

Users and Domain

Users Characteristics:

- Age
- Sex
- Culture
- Psychological: Motivation & Attitude
- Educational background
- Physical Limitations:
  - Vision
  - Movement
  - Thinking
  - Remembering
  - Learning
- Computer/IT experience
- Physical abilities and disabilities:
  - Color blindness
  - Visually impaired
  - Communicating
  - Hearing
  - Mental Health
  - Social Relationship

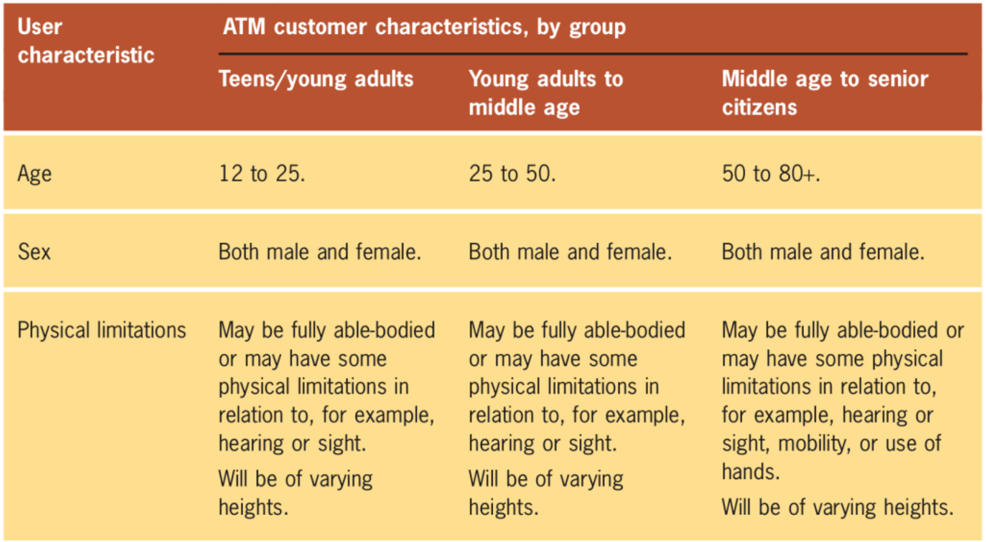
User Profiling

- Profile
  - Step 1: Example of **Initial** User Profile:

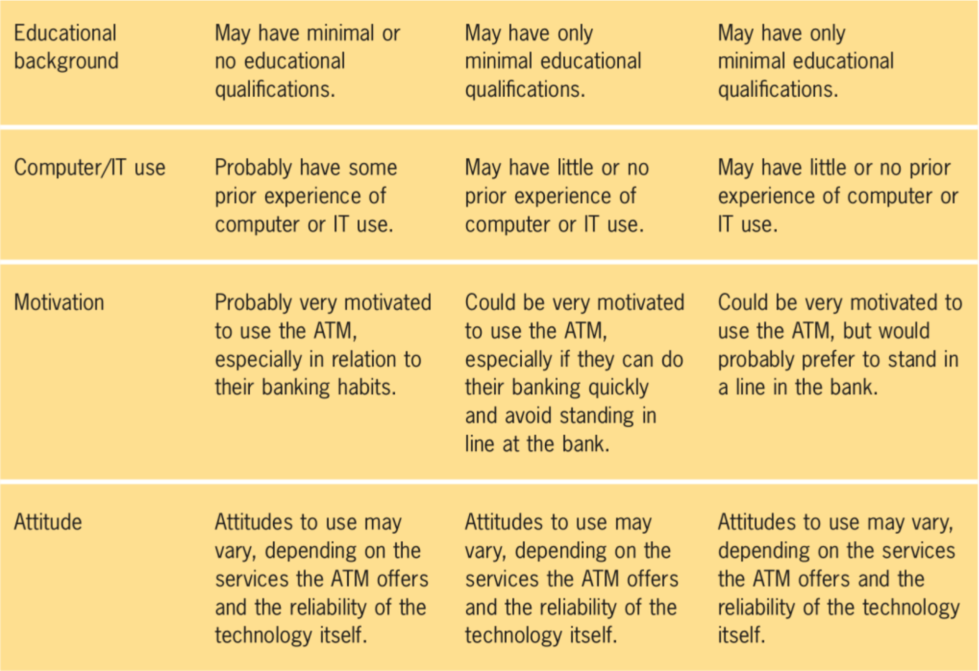
User characteristics	*Customer characteristics*
Age	Will range in age from about 12 to 80+
Sex	Both male and female
Physical limitations	May be fully able-bodied or may have some physical limitations
Educational background	May have only minimal education qualifications and posses limited literacy and numeracy skills
Computer/IT use	May have little or no prior experience of computer or IT use
Motivation	May be very motivated to use the ATM (quick, no waiting)
Attitude	Attitudes to use may vary, depending on the services the ATM offers, the reliability of technology,…

  - Step 2: Example of **detailed** User Profile

Table 3.4 ATM User Groups (adapted from Stone, 2001)



User characteristic	ATM customer characteristics, by group		
	Teens/young adults	Young adults to middle age	Middle age to senior citizens
Age	12 to 25.	25 to 50.	50 to 80+.
Sex	Both male and female.	Both male and female.	Both male and female.
Physical limitations	May be fully able-bodied or may have some physical limitations in relation to, for example, hearing or sight. Will be of varying heights.	May be fully able-bodied or may have some physical limitations in relation to, for example, hearing or sight. Will be of varying heights.	May be fully able-bodied or may have some physical limitations in relation to, for example, hearing or sight, mobility, or use of hands. Will be of varying heights.



Educational background	May have minimal or no educational qualifications.	May have only minimal educational qualifications.	May have only minimal educational qualifications.
Computer/IT use	Probably have some prior experience of computer or IT use.	May have little or no prior experience of computer or IT use.	May have little or no prior experience of computer or IT use.
Motivation	Probably very motivated to use the ATM, especially in relation to their banking habits.	Could be very motivated to use the ATM, especially if they can do their banking quickly and avoid standing in line at the bank.	Could be very motivated to use the ATM, but would probably prefer to stand in a line in the bank.
Attitude	Attitudes to use may vary, depending on the services the ATM offers and the reliability of the technology itself.	Attitudes to use may vary, depending on the services the ATM offers and the reliability of the technology itself.	Attitudes to use may vary, depending on the services the ATM offers and the reliability of the technology itself.

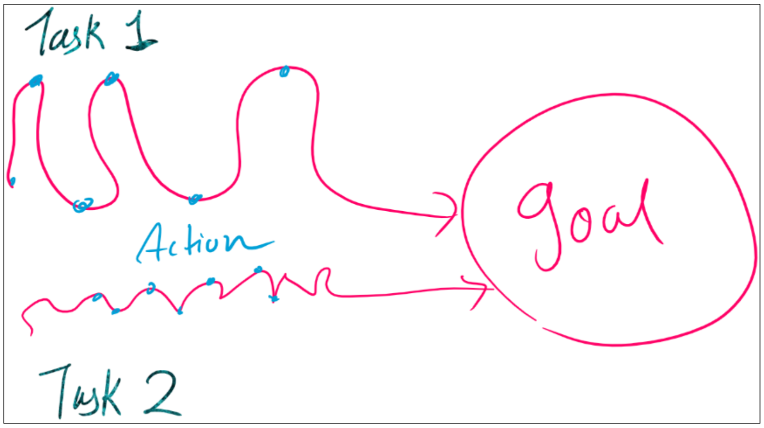
- Persona
  - Personas are based on real people, but aren’t about a real person
  - For Each Persona Includes:
    - Name (picture)
    - Age
    - Occupation
    - Background
    - Key Goals
    - Day in the life
    - Technology proficiency
    - Hobbies
    - Motivators and pain points

User’s Needs:

- **Felt needs:** I’ll Know It When I See It (IKIWISI)
- **Expressed needs:** Said
- **Normative needs:** Identified by experts

Tasks and Work

Goal, Task, and Action



Goal: the end result to be achieved

- High-level abstraction

Task: structured set of related activities that are undertaken in some sequence

- What has to do to accomplish a goal

Action: an individual operation or step that needs to be undertaken as part of the task

Task Characteristics:

- Variety of tasks
- Frequency of tasks
- Knowledge and skill required
- Environmental factors
- Time critical
- Safety hazards
- Will user work alone or with others
- User normally switches between several tasks

Task Analysis:

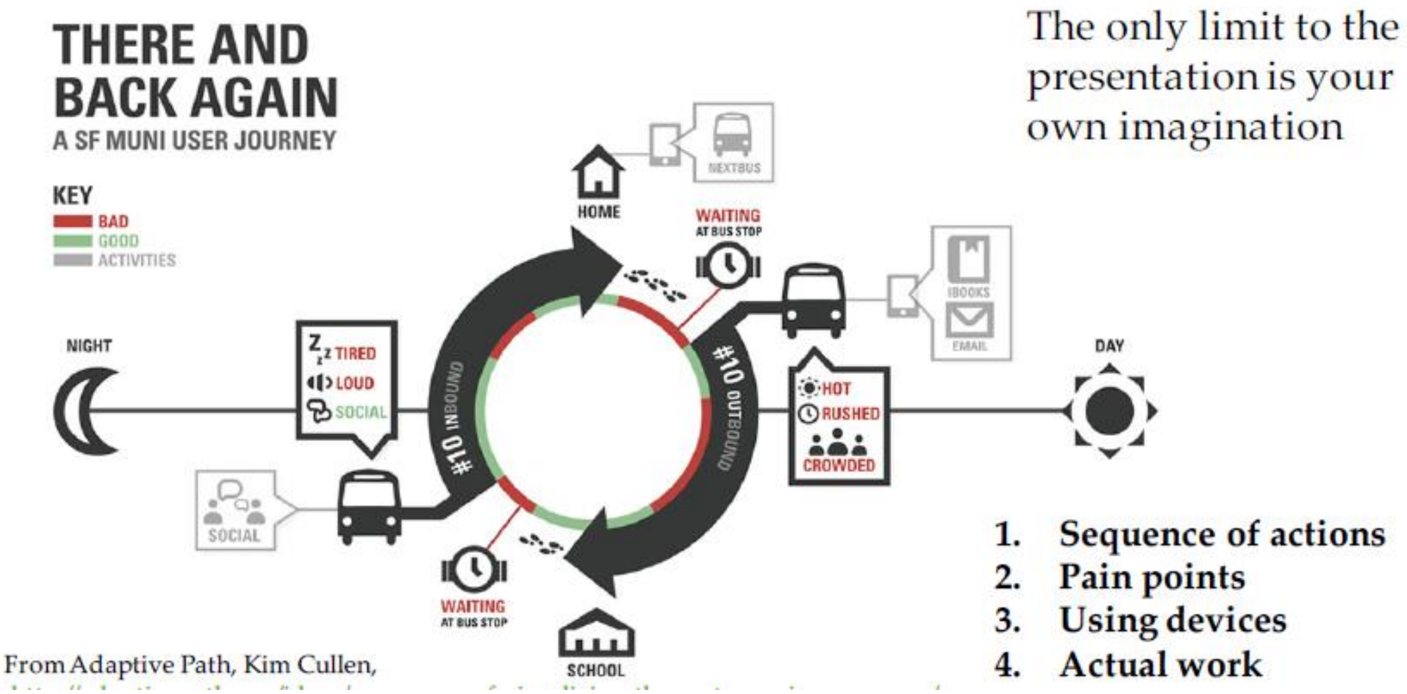
- Clues for Improving Design:
  - Cheat sheets
  - Sticky notes
  - Reference manuals
  - User generated manuals
  - Annotated pages
  - Forms
- Describing How to Do It:
  - Task scenario (Requirements gathering Phase)
    - Narrative & detailed description of a task,
    - Describes the current use of a computer system
    - Usually personalized and describe a specific situation of use
  - Concrete use case (Requirements gathering Phase & Design Phase)
    - Detailed description of a task like task scenario
    - However, is not personalized and so describe the use of a system at a slightly more generic level
  - Essential use case (Design Phase)
    - Describes a task at a high level of abstraction
  - Use scenario (Design Phase)
    - Narrative description of a task, again at a very detailed level
    - It differs from a task scenario in that it describes the anticipated use of the computer system - desired future interaction
- Cognitive Task Analysis:
  - Physical actions: like pressing buttons, moving pointers
  - Mental actions: like deciding which button to press, where to place a pointer, recalling previously stored knowledge from memory, or comparing two objects.

    - Cognitive Walkthrough: evaluates the steps required to perform a task and attempts to uncover mismatches between how the users think about a task and how the UI designer thinks about the task
- Mental Model:
    - Why:
        - Predicting what will happen when the user performs some action for the first time
        - Understanding when the system shows some unexpected behavior
    - How:
        - Experience
        - Training
        - Instruction

User Journey

User Journey: a series of actions that are followed in order to complete some task

- The **task** is undertaken to achieve some **goal**
- An **action** is a step within the journey
- Types of Journey
  - **As-is Journey**: current interaction or usability or currently used by a user
  - **To/Could-be Journey:** Demonstrating the way users could interact with product
- **Why to Create a User Journey?**
  - Demonstrate the **vision** for the project
  - Understand use **behavior**
  - Identify possible **functionality** at a high level
  - Help define **taxonomy** and interface
- **What to Include?**
  - **Context** – where is the user when they are doing this task?
  - **Progression** – what are the actions performed while completing this task?
  - **Devices** – what devices are in use when performing the task?
  - **Functionality/Outcomes** – what is the outcome of the actions and the task?
  - **Emotion/Pain Points** – what is the user’s emotional state in each step? are they engaged, bored, annoyed?
  - **Pain Points** – problems and difficulties with the “current” UI
  - **Touch Points** – user interacts with the system
    - **Before:** ratings & reviews, advertising, etc.…
    - **During:** website, staff, point of sale, etc...
    - **After:** Billing, support teams, thank you cards, etc...



The only limit to the presentation is your own imagination

1. Sequence of actions
2. Pain points
3. Using devices
4. Actual work

From Adaptive Path, Kim Cullen,

Psychological Principles & UX Principles for UI Design

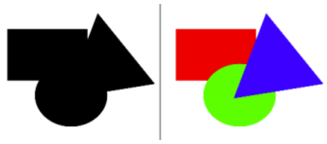
Four Psychological Principles

1. Users **see** what they expect to see
    - Cognitive work
        - Comprehension of product behaviors
    - Prior Knowledge
2. Hard for user to **focus** on more than one activity at a time
    - Visual work
        - Where the eye should start on a screen, finding object, decoding layouts
    - Importance of task/message
3. Easier to perceive a **structured** layout
    - Physical work
        - Keystrokes, mouse movement, gesture (click, drag, double-click), switching between input modes, # of clicks required
4. Easier to **recognize** rather than recall
    - Memory work
        - Recall of product behaviors, commands, names and locations of objects and controls and other relationships between objects

Gestalt Principles

1. Simplicity – People will perceive and interpret ambiguous or complex images as the simplest form possible.
2. Closure – When seeing a complex arrangement of elements, we tend to look for a single recognizable pattern
3. Symmetry and Order – People tend to perceive objects as symmetrical shapes that form around their center
4. Figure / Ground – Elements are perceived as either figure (element in focus) or ground (the background)
5. Uniform Connectedness – Elements that are visually connected are perceived as more related than elements with no connection
6. Common Regions – Elements are perceived as part of a group if they are located within the same closed region
7. Proximity – Objects that are closer together are perceived as more related than objects that are further apart
8. Continuation – Elements arranged on a line are perceived as more related than elements not on the line or curve
9. Common Fate (Synchrony) – Elements that move in the same direction are perceived as more related than elements that are stationary or that move in different directions
10. Parallelism – Elements that are parallel to each other are seen as more related than elements not parallel to each other.
11. Similarity – Elements that share similar characteristics are perceived as more related than elements that don’t share those characteristics.
12. Focus Points – Elements with a point of interest, emphasis, or difference will capture and hold the viewer’s attention

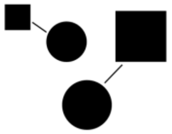
13. Past Experiences – Elements tend to be perceived according to an observer’s past experience.

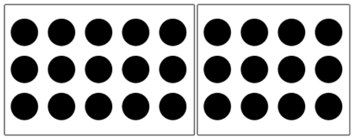
1. 

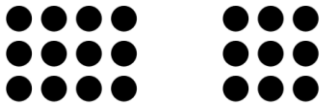
2.

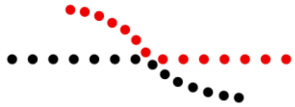
3.

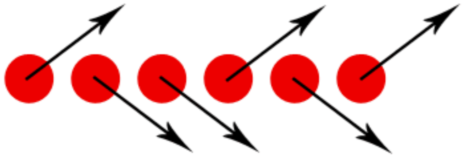
4.

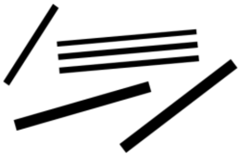
5. 

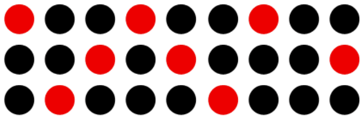
6. 

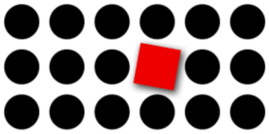
7. 

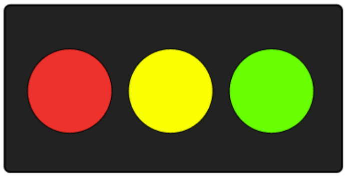
8. 

9. 

10. 

11. 

12. 

13. 

Three Principles from Experience – Norman 1988

1. Visibility
   - It should be obvious what a control is used for
2. Affordance
   - It should be obvious how a control is used
3. Feedback
   - It should be obvious when a control has been used

Usability Requirements

Qualitative: Desired usability goals for a computer system, such as:

- “the system should be easy to use”
- “it should be easy to learn how to order new books”
- “there should be user satisfaction”

  * Typically subjective and difficult to measure and quantify

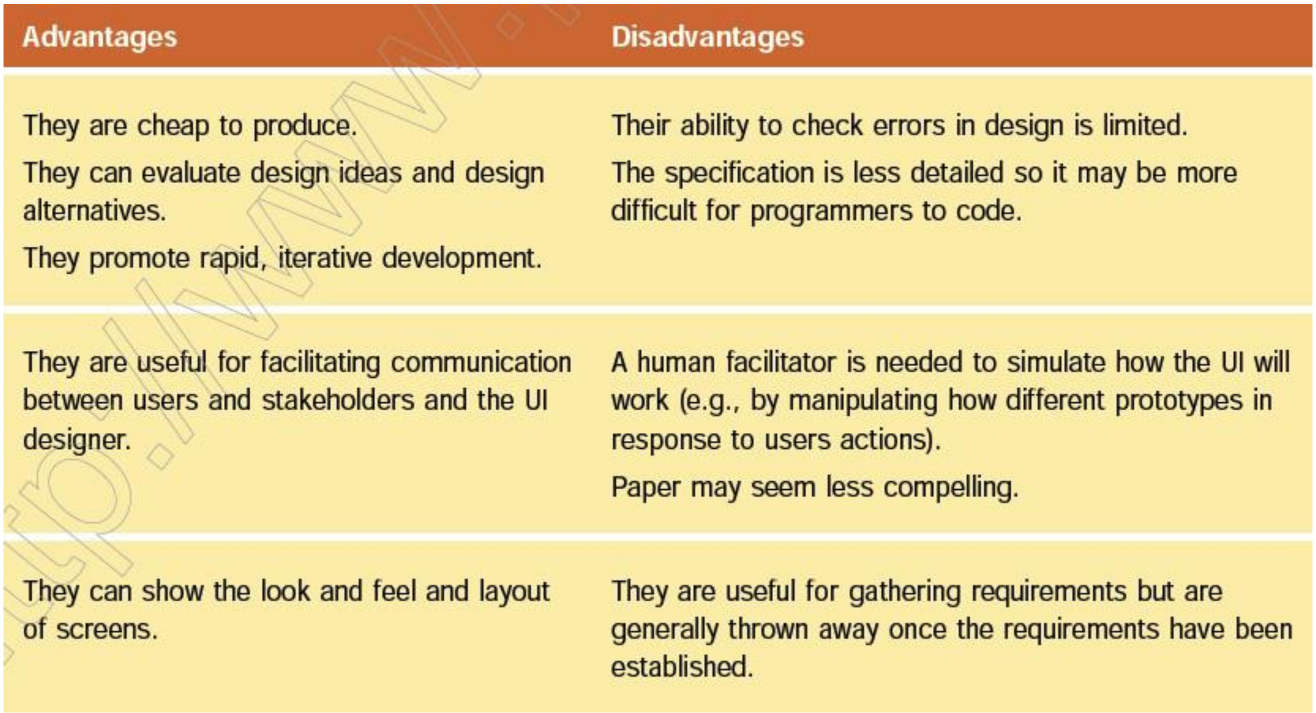
Quantitative: Measure Something (Usability Matrix: Time, numOfErrors, numOfTasks)

- “90% of the users in the 18-25 age group should be able to order a book within 2 minutes”

  * You can use quantitative requirements for testing!

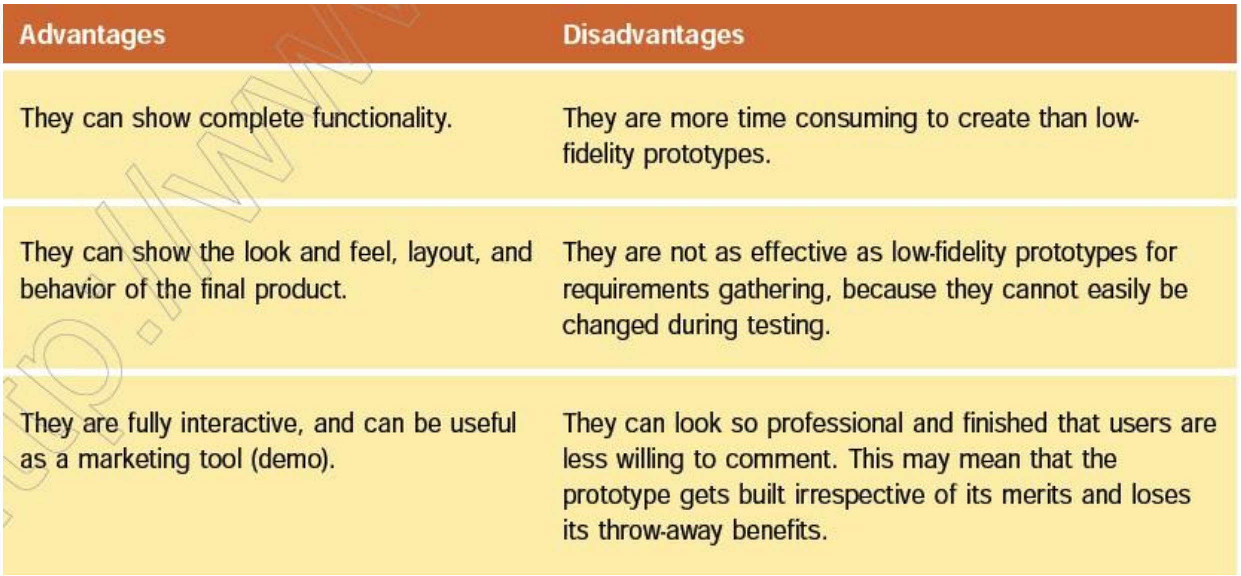
Prototyping

- Low-Fidelity Prototypes
  - Very coarse-grained
  - Fuzzy layouts of general system requirements
  - Paper-based and digital (drawing package like Paint orPowerPoint)
    - Sketching and Screen mockups
      - Quick to build
      - Easy to run
      - Sticky notes & different colors pens on a flipchart/whiteboard
    - Storyboards
      - Sequence of screens (sketches) focusing on a user action
      - Don’t capture every detail, just systems' major functionality
  - Used to gather feedback on the basic functionality or visual layout
  - Users love paper prototypes
    - Opportunity to contribute to the new design



Advantages	Disadvantages
They are cheap to produce. They can evaluate design ideas and design alternatives. They promote rapid, iterative development.	Their ability to check errors in design is limited. The specification is less detailed so it may be more difficult for programmers to code.
They are useful for facilitating communication between users and stakeholders and the UI designer.	A human facilitator is needed to simulate how the UI will work (e.g., by manipulating how different prototypes in response to users actions). Paper may seem less compelling.
They can show the look and feel and layout of screens.	They are useful for gathering requirements but are generally thrown away once the requirements have been established.

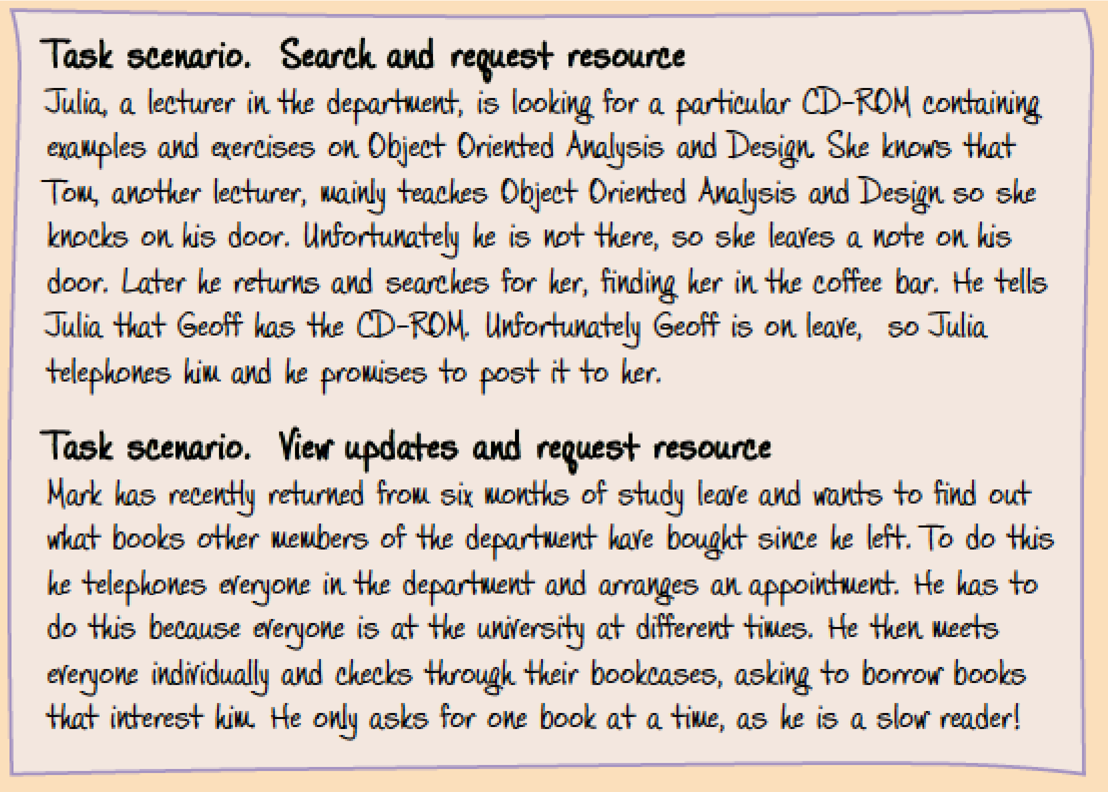
- High-fidelity Prototypes
  - Fine-grained
  - Provide a functional version of the system that users can interact with
  - Highly elaborate and polished digital versions of the system
  - Used to gather detailed information on the processes involved in traversing several parts of the system, or a subset of tasks
  - Using software tools to try out your ideas
  - Usability testing can be undertaken



Advantages	Disadvantages
They can show complete functionality.	They are more time consuming to create than low-fidelity prototypes.
They can show the look and feel, layout, and behavior of the final product.	They are not as effective as low-fidelity prototypes for requirements gathering, because they cannot easily be changed during testing.
They are fully interactive, and can be useful as a marketing tool (demo).	They can look so professional and finished that users are less willing to comment. This may mean that the prototype gets built irrespective of its merits and loses its throw-away benefits.

Work Reengineering and Conceptual Design

⇢ Step 1: Task Scenarios



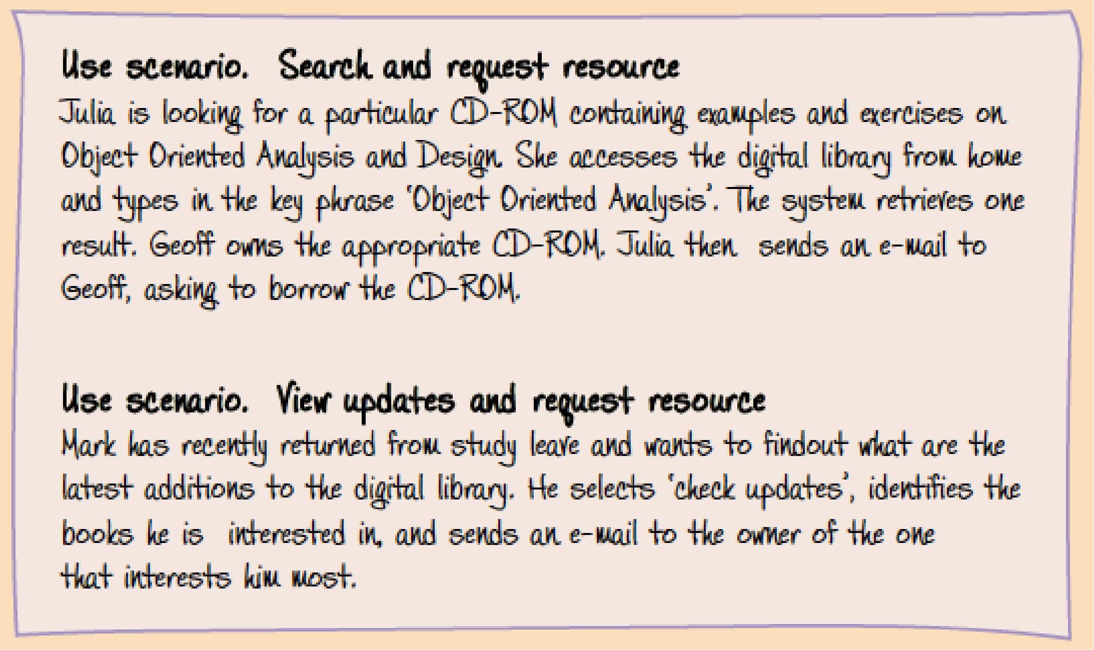
Task scenario. Search and request resource

Julia, a lecturer in the department, is looking for a particular CD-ROM containing examples and exercises on Object Oriented Analysis and Design. She knows that Tom, another lecturer, mainly teaches Object Oriented Analysis and Design so she knocks on his door. Unfortunately he is not there, so she leaves a note on his door. Later he returns and searches for her, finding her in the coffee bar. He tells Julia that Geoff has the CD-ROM. Unfortunately Geoff is on leave, so Julia telephones him and he promises to post it to her.

Task scenario. View updates and request resource

Mark has recently returned from six months of study leave and wants to find out what books other members of the department have bought since he left. To do this he telephones everyone in the department and arranges an appointment. He has to do this because everyone is at the university at different times. He then meets everyone individually and checks through their bookcases, asking to borrow books that interest him. He only asks for one book at a time, as he is a slow reader!

⇢ Step 2: Use Scenarios for the Digital Library



Use scenario. Search and request resource

Julia is looking for a particular CD-ROM containing examples and exercises on Object Oriented Analysis and Design. She accesses the digital library from home and types in the key phrase 'Object Oriented Analysis'. The system retrieves one result. Geoff owns the appropriate CD-ROM. Julia then sends an e-mail to Geoff, asking to borrow the CD-ROM.

Use scenario. View updates and request resource

Mark has recently returned from study leave and wants to findout what are the latest additions to the digital library. He selects 'check updates', identifies the books he is interested in, and sends an e-mail to the owner of the one that interests him most.

⇢Notice that:

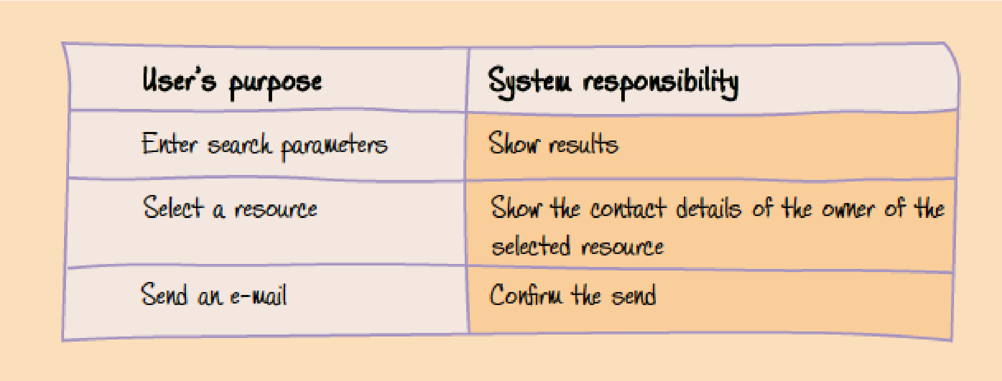
- There are no details about the technology
- The focus is on the users and how they will carry out their tasks
- How the digital library might operate

⇢ Step 3: Essential Use Case ➔ Task Allocation

- We can specify the task allocation using essential use cases
- Sharing the different "essential" tasks between the user and the computer.

Essential use case:

“Search and request resource”

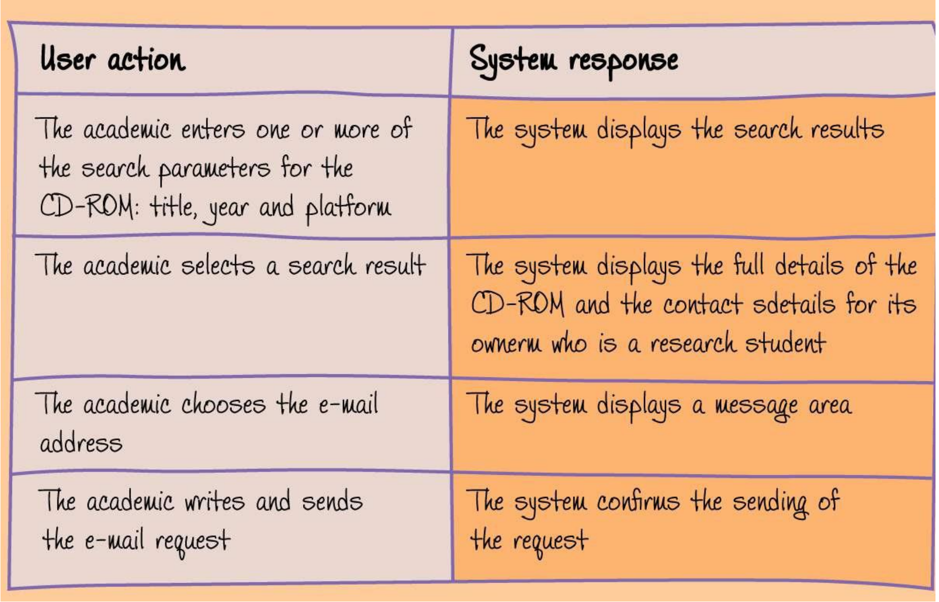


User's purpose	System responsibility
Enter search parameters	Show results
Select a resource	Show the contact details of the owner of the selected resource
Send an e-mail	Confirm the send

Figure 8.3 “Search and request resource” essential use case.

Concrete use case:

“Search and request CD-ROM”



User action	System response
The academic enters one or more of the search parameters for the CD-ROM: title, year and platform	The system displays the search results
The academic selects a search result	The system displays the full details of the CD-ROM and the contact sdetails for its ownerm who is a research student
The academic chooses the e-mail address	The system displays a message area
The academic writes and sends the e-mail request	The system confirms the sending of the request