

# Task-Centered System Design

How to develop task examples

How to evaluate designs through a task-centered walk-through

Exercise: The Cheap Shop interface

Saul Greenberg

## Cheap Shop

Screen 1

**Purchaser**  
Name:  Phone:   
Postal Code:  Province:  City:   
Delivery Address:   
Today's date:   
Credit Card No.:  for dept use: validation id:

**Catalog Item**  
Number:  Quantity:  Cost/item:  Total:   
Balance Owing:

Next Catalog Item (PF5)  
Trigger Invoice (PF8)

Screen 2

**Catalog Item**  
Number:  Quantity:  Cost/item:  Total:   
Balance Owing:

Next Catalog Item (PF8)  
Trigger Invoice (PF5)

Saul Greenberg

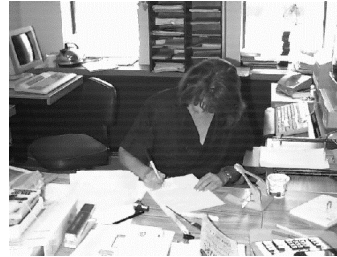
## Task-Centered System Design

### Requirements analysis in HCI

- exactly who would use the system to do exactly what?



vs.



### **The User**

*a pretend person  
who will mould  
themselves to fit  
your system*

### **Mary**

*a real person with  
real constraints  
trying to get her job  
done*

Saul Greenberg

## The Task-Centered Process

### **Phase 1: Identification**

- identify specific users
- articulate example realistic tasks

### **Phase 2: Requirements**

- decide which of these tasks and users the design will support

### **Phase 3: Design**

- base design representation and dialog sequences on these tasks

### **Phase 4: Walkthrough Evaluations**

- using your design, walk through these tasks to test the proposed interface

Saul Greenberg

## **Phase 1: Identification**

### **Get in touch with real people who will be potential users of your system**

- identify specific end users
  - prototypical categories & extremes

### **Spend time with them discussing how the system might fit in**

- who would be willing to talk to you about this?
- if you can't get them interested, who will actually buy/use your system?

### **Learn about the user's tasks**

- articulate concrete, detailed examples of tasks they perform or want to perform that your system should support
  - routine
  - infrequent but important
  - infrequent and incidental

Saul Greenberg

## **Phase 1: Identification**

### **If there are no real users or tasks...**

- think again, there probably are!

Jeff Hawkins, the inventor of the Palm Pilot, was said to have carried a small block of wood around in his shirt pocket ... As various everyday situations arose, he would take out the block of wood and imagine how he would use the device.<sup>1</sup>

<sup>1</sup>see Sato and Salvador, *interactions* 6(5)

The same technique can be used to evoke a response from expected end-users

### **If all else fails...**

- describe your expected set of users, and expected set of tasks
- these will become your 'assumed users and tasks' that can be verified or modified later

Saul Greenberg

## **Phase 1: Identification**

### **Developing good task examples**

#### **1. Says what the user wants to do but does not say how they would do it**

- no assumptions made about the interface
- can be used to compare different design alternatives in a fair way

#### **2. Are very specific**

- says exactly what the user wants to do
- specifies actual items the user would eventually want to input (somehow)

Saul Greenberg

## **Phase 1: Identification**

#### **3. Describes a complete job**

- not just a list of simple things the system should do!
- does more than present a sub-goal independent of other sub-goals
- forces designer to consider how interface features will work together
- contrasts how information input and output is carried through the dialog
  - where does information come from?
  - where does it go?
  - what has to happen next?

#### **4. Says who the users are**

- design success strongly influenced by what users know
- name names, if possible
- reflect real interests of real users
- helps find tasks that illustrate functionality in a person's real work context

Saul Greenberg

## **Phase 1: Identification**

### **5. Are evaluated**

- Circulate descriptions to users, and rewrite if needed
  - ask users for
    - omissions
    - corrections
    - clarifications
    - suggestions

### **6. As a set, identifies a broad coverage of users and task types**

- the typical 'expected' user,                      typical routine tasks
- the occasional but important user,              infrequent but important tasks
- the unusual user                                      unexpected or odd tasks

Saul Greenberg

## **Phase 2: Requirements**

### **Which user types will be addressed by the interface?**

- designs can rarely handle everyone!
- includes why are particular users included / excluded?

### **Which (sub-) tasks will be addressed by the interface?**

- designs can rarely handle all tasks
- requirements listed in terms of how they address tasks
  - Absolutely must include:
    - ...
  - Should include:
  - Could include:
    - ...
  - Exclude:
    - ...
- Discussion includes why items are in those categories

Saul Greenberg

## **Phase 3: Design as Scenarios**

**Develop designs around how well they fit users and specific tasks**

**Use tasks to**

- get specific about possible designs
- consider how design features work together to help a person accomplish real work
- consider the real world contexts of real users

**Reconsider how a design scenario handles each task**

- what the user would do and see step-by-step when performing the task

Saul Greenberg

## **Phase 4: Walk-through Evaluation**

**Good for developing an interface**

- debugging

**Process:**

**1 Select one of the task scenarios**

**2 For each user's step/action in the task:**

- can you build a believable story that motivates the user's actions?
- can you rely on user's expected knowledge and training about system?
- if you cannot:
  - then you've located a problem in the interface!
  - once a problem is identified, assume it has been repaired
- go to the next step in the task

Saul Greenberg

## **Example: The Cheap Shop Catalog Store**

In Cheap Shop, people shop by browsing paper catalogs scattered around the store.

When people see an item they want, they enter its item code from the catalog onto a form.

People give this form to a clerk, who brings the item(s) from the back room to the front counter.

People then pay for the items they want.



Item code	Amount
323066 697	1

Saul Greenberg

## **Developing task examples: Cheap Shop**

*At Cheap Shop, people browse a catalog and then order goods from a clerk.*

### **Task example 1:**

- Fred, who is caring for his demanding toddler son, buys an umbrella stroller (red is preferred, but blue is acceptable), pays for it in cash, and uses it immediately.

Fred is a first-time customer to this store and has little computer experience

Saul Greenberg

## **Developing task examples: Cheap Shop**

*At Cheap Shop, people browse a catalog and then order goods from a clerk.*

### **Task example 2:**

- An elderly arthritic woman is price-comparing the costs of a child's bedroom set, consisting of a wooden desk, a chair, a single bed, a mattress, a bedspread, and a pillow.

She takes the description and total cost away with her, to check against other stores.

Two hours later, she returns and decides to buy everything but the chair.

Saul Greenberg

## **Developing task examples: Cheap Shop**

*At Cheap Shop, people browse a catalog and then order goods from a clerk.*

### **Task example 3:**

- A "Cheap Shop" clerk, who is the sole salesperson in the store, is given a list of 10 items by a customer who does not want to use the computer.

The items are:

- 4 pine chairs, 1 pine table, 6 blue place mats, 6 "lor" forks, 6 "lor" table spoons, 6 "lor" teaspoons, 6 "lor" knives, 1 "tot" tricycle, 1 red ball, 1 "silva" croquet set

After seeing the total, the customer decides to take all but the silverware, and then adds 1 blue ball to the list.

The customer then changes their mind about paying by credit card, and decides to pay cash. The customer wants the items delivered to his home the day after tomorrow.

While this is occurring, 6 other customers are waiting for the salesperson.

Saul Greenberg



## Cheap Shop

Screen 1

**Purchaser**  
Name:  Phone:   
Postal Code:  Province:  City:   
Delivery Address:   
Today's date:   
Credit Card No.:  for dept use: validation id:

**Catalog Item**  
Number:  Quantity:  Cost/item:  Total:

Balance Owing:

Next Catalog Item (PF5)  
Trigger Invoice (PF8)

Screen 2

**Catalog Item**  
Number:  Quantity:  Cost/item:  Total:

Balance Owing:

Next Catalog Item (PF8)  
Trigger Invoice (PF5)

Saul Greenberg

## Specifications

### **To create an order**

- On screen 1, shoppers enter their personal information and their first order
- text is entered via keyboard
- the tab or mouse is used to go between fields.

### **Further orders**

- shoppers go to the 2nd screen by pressing the Next Catalog Item button

### **Order completion**

- shoppers select 'Trigger Invoice'.
- the system automatically tells shipping and billing about the order
- the system returns to a blank screen #1

### **To cancel order**

- Shoppers do not enter input for 30 seconds (as if they walk away)
- The system will then clear all screens and return to the main screen

### **Input checking**

- all input fields checked when either button is pressed.
- erroneous fields will blink for 3 seconds, and will then be cleared.
- the shopper can then re-enter the correct values in those fields.

Saul Greenberg

## Walkthrough template

Task number: \_\_\_\_

Description of Step	Does the user have the knowledge/training to do this?	Is it believable that they would do it? Are they motivated?	Comment / solution

Saul Greenberg

## Limitations

### Tasks almost always embody a process

- may be hard to produce a pure task that is 'system' or 'process' independent
- may encourage designs that do not look at alternative ways to do tasks
- may be impossible to find someone who actually does the task

Saul Greenberg

## **Goal-centered system design**

### **Articulates user goals rather than how they want to do them**

- Goal:
  - a desired end condition
  - tend to be stable
- Task:
  - an intermediate process needed to achieve the goal
  - may change as technology / work patterns change

### **Designer analyzes goals, looking for solutions and how to satisfy them**

- may result in different task / task sequence which could be better

### **Approach:**

- Develop a persona
  - precise, specific description of the user and what they wish to accomplish (Goal)
  - a pretend user that are hypothetical archetypes of actual users
  - discovered as a by-product of investigating the problem domain
- Develop a cast of characters
  - 3 – 12 unique personas
  - one will be the primary persona – the main focus of the design

*See Fred Cooper 'The inmates are running the asylum'*

Saul Greenberg

## **You now know**

**How to develop concrete task examples**

**How to use task examples to motivate your designs**

**How to evaluate designs through task-centered walkthroughs**

Saul Greenberg

