A Working Example: Cheap Shop

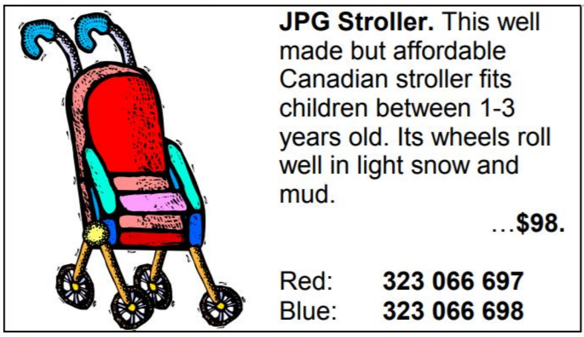
This sections steps through a working example to illustrate how this process can be applied.

The Situation

Cheap Shop is a catalog-based department store known for its low cost merchandise. A customer shops by browsing one of the paper catalogs scattered around the store. As the customer finds each desired item, he or she enters its item code from the catalog onto an order form. The customer then gives this form to a sales clerk at the front counter. After a modest time (about 3-8 minutes), the warehouse clerk delivers the items from the back room to the sale clerk at the front counter. The sales clerk passes it to the customer. The customer checks the items, and pays the sales clerk for the items they want. An example item for the catalog as well as a filled in form is illustrated in Figure 1.

Cheap Shop has contracted you to evaluate an in-store computer system that they have prototyped, where customers would use this prototype system to indicate and buy the items they want. The system would then send this request to the warehouse, after which the items will appear at the front counter for further processing by a clerk.

If the prototype has major problems, you can either suggest how it can be repaired or propose a completely new design.

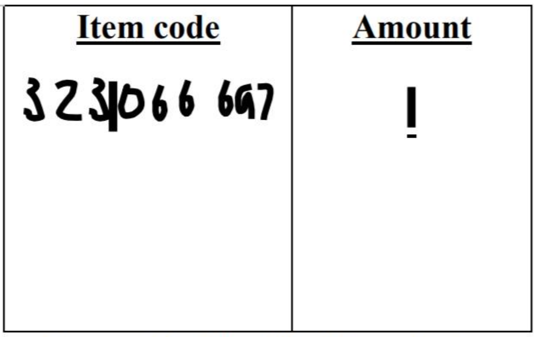


JPG Stroller. This well made but affordable Canadian stroller fits children between 1-3 years old. Its wheels roll well in light snow and mud.

…**$98.**

Red:	**323 066 697**
Blue:	**323 066 698**

Figure 1a: The catalog entry for the

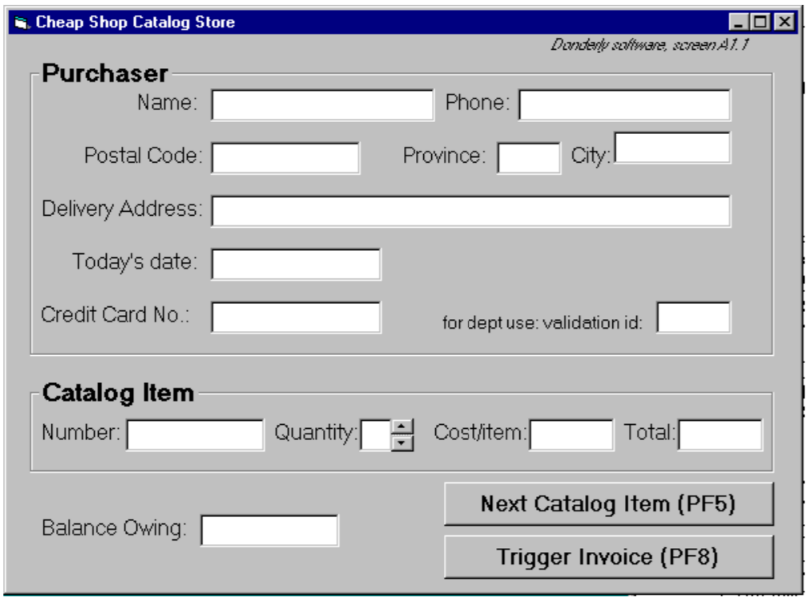


Item code	**Amount**
3231066 697	1

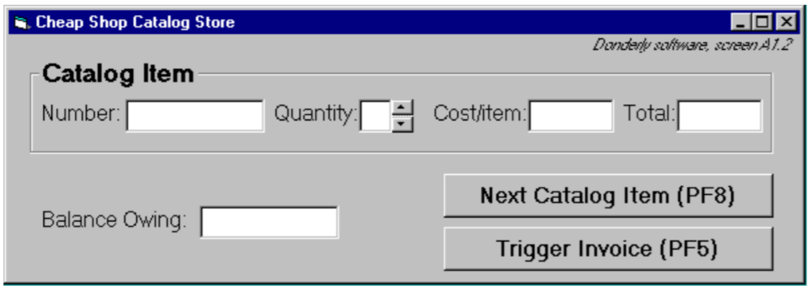
Figure 1b: The filled-in order form

The Cheap Shop Prototype

The prototype illustrated in Figure 2 is intended to be available on all Cheap Shop Department Store computers. Shoppers in the store decide on the item they want by browsing the catalog, and can then purchase items by entering the relevant information into these screens.



Screen 1



Screen 2

Prototype specifications.

To order the first item:
- shoppers follow the sequence on screen 1 to enter their personal information and their first order;
- text is entered via keyboard, and the tab or mouse is used to go between fields.

To order additional items:
- shoppers fill in screen 2 after clicking *Next Catalog Item* (can be repeated).

To complete an order:
- shoppers click ‘Trigger Invoice’;
- the system automatically tells shipping and billing about the order;
- the system returns to a blank screen #1.

To cancel the 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.

Figure 2: The Cheap Shop Prototype

Producing User and Task Descriptions for Cheap Shop

We collected descriptions by monitoring customer activity at the Cheap Shop store. We validated each description by interviewing the customer and by asking them if it reflected what they did. We also sat behind the counter with the store clerks, where we observed what they did and how customers and clerks talked to each other. Later, we gave the complete set of descriptions to the store clerks (the client representative) and asked them if the descriptions typified what they saw in terms of customer requests. Three of our descriptions are included below. Notice that they follow the criteria for good task descriptions identified earlier.

Task 1. Fred Johnson, who is caring for his demanding toddler son, wants a good quality umbrella stroller (red is preferred, but blue is acceptable). He browsers the catalog and chooses the JPG stroller (cost $98., item code 323 066 697). He pays for it in cash, and uses it immediately. Fred is a first-time customer to this store, has little computer experience, and says he types very slowly with one finger.

Discussion. Fred has many properties of our typical expected user: many customers are first time shoppers, and a good number have no computer experience and are poor typists. Similarly, the task type is routine and important. Many people often purchase only one item, and a good number of those pay by cash. As with Fred, people often have a general sense of what they want to buy, but decide on the actual product only after seeing what is available.

Task 2. Mary Vornushia, 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 all made by Furnons Company. She takes the description and total cost away with her to check against other stores. Three hours later, she returns and decides to buy everything but the chair. She pays by credit card and asks for the items to be delivered to her daughter's home at 31247 Lucinda Drive, in the basement suite at the back of the house.

Discussion. Like Mary, a reasonable number of store customers are elderly, with infirmities that inhibit their physical abilities. A modest number of them also enjoy comparison shopping, perhaps because they have more time on their hands or because they are on low income. Although this would be considered a 'major' purchase in terms of the total cost, the number of items purchased is not unusual. Delivery of large items is the norm, and many customers pay by credit card for larger orders.

Task 3. John Forham, the sole sales clerk 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 tells John he will take all but the silverware, and decides to add 1 blue ball to the list. The customer starts paying John by credit card, but then changes his mind and decides to pay cash. The customer tells John he wants the items delivered to his home the day after tomorrow. While this is occurring, 6 other customers are waiting for John. John has been on staff for 1 week, and is only partway through his training program.

Discussion. This task introduces the clerk as a system user. While every store will have a few clerks, they are vastly outnumbered by the number of customers using the system. Because the store has a high turnover in its staff, new employees such as John are also common. Thus John reflects a ‘rare’ but important group of users. The task that John is asked to do by the customer, while complex, is fairly typical i.e., people making large numbers of purchases often ask the clerk to help them. Similarly, clerks mention that customers often change their mind partway through a transaction i.e., by changing what they want to buy and/or by changing how they want to pay for it. Customers, however, rarely give specific delivery dates, with most wanting delivery as soon as possible. Lineups for clerks do happen during busy times.

The Task Walkthrough.

Because we already have an interface in hand, we begin by performing a task centered walkthrough of it. The example below shows our walkthrough analysis performed with a scenario that combines our first task description with this interface.

For reporting purposes, each walkthrough report is preceded by a description of the scenario i.e., which task description and which interface is being analyzed. The table itself tells the story, where it records the step-by-step results of the task walkthrough algorithm. The 1st column describes each task step in sequence. The 2nd column asks if that person has the knowledge or training to do this step, and if it is believable that the person would be motivated to do what is asked of them. The final column records problem details, comments, and (optional) solutions to any problems.

Interface: Cheap Shop prototype #1.
Description #1. Fred Johnson, who is caring for his demanding toddler son, wants a good quality umbrella stroller (red is preferred, but blue is acceptable). He browsers the catalog and chooses the JPG stroller (cost $98., item code 323 066 697). He pays for it in cash, and uses it immediately. Fred is a first-time customer to this store, has little computer experience, and says he types very slowly with one finger.

The sequence in Table 1 illustrates the value of starting the walkthrough at the very beginning of the task. Notably, we see that we are missing information about whether paper or electronic catalogs will be used, how the computerized system is situated in the store environment, and whether signage or other instructional material will tell customers what to do. While doing task centered design, you should examine what information is missing, and list what assumptions you are making. You should validate these assumptions, as incorrect ones can profoundly affect how the interface will perform in the actual setting. Even when assumptions reveal issues 'outside' the actual interface being designed, they can be critical to its success.

Task step	Knowledge? Believable? Motivated?	Comments/solutions.
a Enters store	ok	
b Looks for catalog	Ok if paper catalog is used, but what if the catalog is on-line?	Finding paper catalogs is not a problem in the current store. However, we were not told if the paper catalog would still be used, or if the catalog would be made available on line. ***Note:*** ask Cheap Shop about this. If they are developing an electronic catalog, we will have to consider how our interface will work with it. For now, we assume only a paper catalog is used.
c Finds red JPG stroller in catalog	Ok	The current paper catalog has proven itself repeatedly as an effective way for customers to browse Cheap Shop merchandise and to locate products.
d Looks for computer	Modest problem	As a first time customer, Fred does not know that he needs to order through the computer. Unfortunately, we do not know how the store plans to tell customers that they should use the computer. Is there a computer next to every catalog (so its association can be inferred), or are there limited number of computers on separate counters? Are there signs telling Fred what to do? ***Note:*** ask Cheap Shop about the store layout and possible signage. ***Possible solution:*** Instead of screen 1, a startup screen can clearly indicate what the computer is for e.g., "Order your items here" in large letters.

Table 1: From entering the store to finding the computer

The next sequence in Table 2 illustrates fundamental problems that arise as one walks through task step details, such as how Fred selects and moves between fields, and how Fred enters input. It also illustrates how the inspector can do a reality check on walkthrough steps in the large i.e., whether the expected sequence of activities matches Fred's goals, and whether Fred is willing to enter the expected information. In this case, we see several serious problems that must be repaired.

e Enters name	No motivation to do this!	Fred's task is to buy the stroller, but the scenario shows that the system is asking him for his name. Fred may be reluctant to do so if (say) he believes that he will be added to a mailing list without his permission. ***Note.*** Ask Cheap Shop why they are asking for the customer's name and other contact information.
f Selects the name field	Knowledge lacking. Fred does not know how to select a field.	To enter his name, Fred is expected to click and type into the first text field on this form. Yet Fred has little computer experience, and thus he may not know what to do. He may also be reluctant to experiment with the system. ***Possible solutions:*** a) have the first field pre-selected, with the cursor in it. b)have a poster next to the computer describing these basic acts.

g Types his name	Knowledge lacking: Fred types poorly, does not know name format.	Because Fred types poorly, text entry will be slow and tedious. This further dampens Fred's motivation, as he is entering information that is unimportant to the task. Fred is uncertain about formats: does he type his name as 'Fred Johnson' or 'Johnson, Fred'?
h Moves to phone field	Knowledge lacking	Fred may not know how to tab or mouse over to the next field because of his unfamiliarity with computers.
i Fill in phone, postal code, province, and city.	Poor motivation. Poor format knowledge	If Fred can complete steps e-h, he will be able to continue with the following fields. However, motivation will decrease even further as Fred painfully types unnecessary information into the system. Fred continues to have formatting concerns about how he should enter information. Should the phone number include the area code, spaces and/or dashes? Should he spell out the province or use the abbreviation? Should he leave a space in the postal code?
j Enter delivery address	Violates the task	Fred will use the stroller immediately, but the system asks for his delivery address. Fred may incorrectly assume that he is filling in the wrong form, and may give up. We also noticed that the order of the contact information does not follow the typical flow i.e., one would expect 'Name, Address, City, Province, Postal Code', 'Phone' rather than the odd order shown in the form.
k Enters today's date	No motivation	This is an odd field… why should Fred enter the date when the system already knows it? Can he skip it? If he does fill it in, he would be quite lucky to enter a recognizable date format.
l Enters credit card information	Violates the task	Fred is paying by cash, and thus he is unwilling enter his credit card number. He is also concerned that others may see his credit card information as he types it onto the screen. Finally, this seems an odd place to ask for payment information. Most stores ask for it at the end of transaction, not at the beginning.
m Ignore validation id	Ok	While Fred will likely do the right thing, this field should not be here. It has nothing to do with Fred's task. ***Possible solution.*** Remove it.
Steps e-m	Not needed for task.	***Possible solution.*** This entire part of the interface is not needed or is, at best, optional (e.g., if it is for getting onto the mailing list). Delete it entirely or move it into a very secondary area that can be filled in after the transaction is completed.

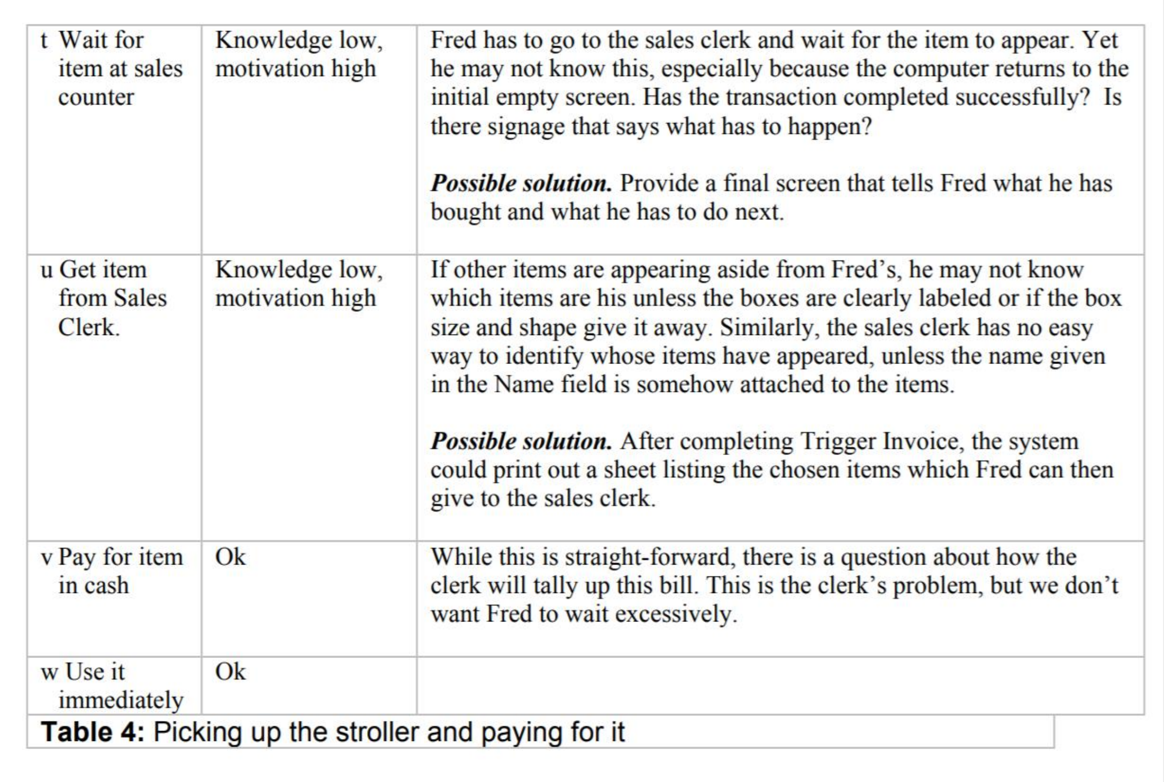
Table 2: Entering personal information

The task steps in Table 3 illustrate how poorly the interface handles the most critical part of the task, where Fred specifies the item he wants to buy, and tries to complete the transaction. We also see that the interface is full of jargon and unneeded or poorly designed interface components.

n Enter item number for the JPG red stroller	Is motivated, but has problems. Error-prone.	The item number for Fred's stroller is written in the paper catalog as 323 066 397. Because catalogs are common, he may be able to figure out what he has to do. However, the format is a bit mysterious – should he include spaces or not? If the paper catalog is in an awkward place, Fred will have to rely on his memory to enter the number or he will constantly be running back and forth between the catalog and the computer. Because Fred is a poor typist, he may have difficulty typing the number correctly.
o Enter quantity	Knowledge low, motivation high	Fred wants one stroller only. However, this 'spinner widget' is somewhat mysterious to Fred. Because he does not know computers, he will likely not know that he can type the amount directly or just click the arrows to select the quantity. ***Partial solution.*** Have the spinner show a 1 by default.
p Enter cost/item	Motivation low	Why should Fred enter the cost? Surely the system knows this. If this field is actually used to display the cost, then it has the wrong visual affordance as it looks like a text box. Perhaps Fred would be willing to enter a deeply discounted cost, but this will probably be treated as a system error.
q Enter total	Motivation low	See above point
r Enter balance owing	Motivation low, knowledge low	See above point. Fred will also be uncertain about how this field differs from the 'Total' field.
s Click Trigger invoice or press PF5	Knowledge low	Being inexperienced with computers, Fred may not recognize or know how to use a clickable button. The 'PF5' label is also mysterious, as Fred does not recognize it as a keyboard shortcut. Fred will find the meaning of 'Trigger Invoice' cryptic, as it is in the language of the database system rather than in his language. This may leave him at a loss of what to do. *Possible solutions.* Remove the PF5 label. Change 'Trigger Invoice' to something more meaningful.

Table 3: Buying the stroller

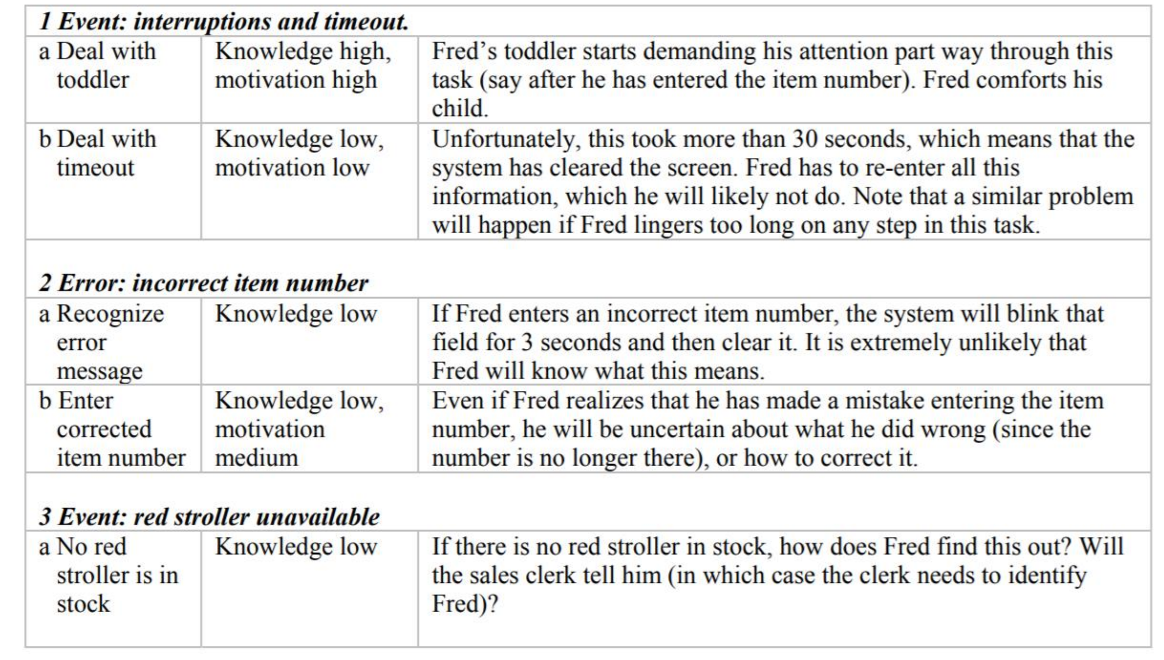
As with the opening task sequence in Table 1, the sequence in Table 4 illustrates how the closing of the task must recognize factors that go beyond the interface. In this case, we see a serious problem of how the electronic part of the task flows through to the physical completion of the task.



t Wait for item at sales counter	Knowledge low, motivation high	Fred has to go to the sales clerk and wait for the item to appear. Yet he may not know this, especially because the computer returns to the initial empty screen. Has the transaction completed successfully? Is there signage that says what has to happen? ***Possible solution.*** Provide a final screen that tells Fred what he has bought and what he has to do next.
u Get item from Sales Clerk.	Knowledge low, motivation high	If other items are appearing aside from Fred's, he may not know which items are his unless the boxes are clearly labeled or if the box size and shape give it away. Similarly, the sales clerk has no easy way to identify whose items have appeared, unless the name given in the Name field is somehow attached to the items. ***Possible solution.*** After completing Trigger Invoice, the system could print out a sheet listing the chosen items which Fred can then give to the sales clerk.
v Pay for item in cash	Ok	While this is straight-forward, there is a question about how the clerk will tally up this bill. This is the clerk's problem, but we don't want Fred to wait excessively.
w Use it immediately	Ok	

Table 4: Picking up the stroller and paying for it

The above sequences in Tables 1-4 walk through the correct task sequence. What happens when errors or other events occur that disrupt this sequence? Table 5 illustrates a set a sequences dealing with problems. It shows that walkthroughs are also effective for discovering how the system design deals with problems, and with events arising from the end-user's real-world context.



1 Event: interruptions and timeout.		
a Deal with toddler	Knowledge high, motivation high	Fred's toddler starts demanding his attention part way through this task (say after he has entered the item number). Fred comforts his child.
b Deal with timeout	Knowledge low, motivation low	Unfortunately, this took more than 30 seconds, which means that the system has cleared the screen. Fred has to re-enter all this information, which he will likely not do. Note that a similar problem will happen if Fred lingers too long on any step in this task.
2 Error: incorrect item number		
a Recognize error message	Knowledge low	If Fred enters an incorrect item number, the system will blink that field for 3 seconds and then clear it. It is extremely unlikely that Fred will know what this means.
b Enter corrected item number	Knowledge low, motivation medium	Even if Fred realizes that he has made a mistake entering the item number, he will be uncertain about what he did wrong (since the number is no longer there), or how to correct it.
3 Event: red stroller unavailable		
a No red stroller is in stock	Knowledge low	If there is no red stroller in stock, how does Fred find this out? Will the sales clerk tell him (in which case the clerk needs to identify Fred)?

b Reenter all information for blue stroller	Motivation low	We cannot imagine that Fred would be willing to go through this whole process again, especially because his demanding toddler is likely loosing patience. ***Possible solution:*** As the customer selects and item, the interface should clearly indicate if it is in stock.

Table 5: Interruptions, errors, and exceptions

While Tables 1-5 appear fairly detailed, they do not cover all task steps. Some steps are intentionally left out because they are identical to previously seen task steps. For example, after identifying problems with selecting and moving between fields as well as typing (steps f-h in Table 2), these details are skipped in other task steps. Other steps are left out because of oversights i.e., they simply were not thought about. For example, no mention is made in Table 5 about how the customer or clerk unpacked the box the stroller came in, or what was done with it afterwards. While oversights will happen, they can be reduced by adding more people to the team and by grounding the task by real observations. For example, videotaping a person doing a real task forces one to account for all visible steps.

Walkthroughs of the interface using the other two descriptions yield other problems. For the second description, we identify a user (Mary) who cannot use the mouse or type because she is arthritic. What should be done with her? We also see serious problems when multiple items are ordered; the system gives no feedback of what was entered, and there is no way to correct errors without re-entering everything from scratch. There is also no easy way for Mary to price-compare (since no printout is provided), nor is there any way for her to recall information that she previously entered. Details are also a problem: there is no easy way for her to tell the system about the unusual delivery address (i.e., that it is the basement in the back). For the third description, we see that the system is completely unsuitable for use by the store clerk. Item entry is too slow, and the clerk will not be able to keep up with the changes requested by the customer. This slowness also affects other customers waiting in line. We also see that the system cannot accommodate delayed delivery.

The bottom line is that this design is a disaster, and it should be completely revamped.

Greenberg, S. (2004) Working through Task-Centered System Design. in Diaper, D. and Stanton, N. (Eds) The Handbook of Task Analysis for Human-Computer Interaction. Lawrence Erlbaum Associates. p49-66.