Task centered design: Background

The Situation

- A small library has contracted you to build a computer system that will let librarians and their assistants deal with routine requests by the library clients. The computer(s) will be situated on the check-in / checkout counter.

Saul Greenberg

The following example will be used in labs to walk students through an example that is similar in scope to the example they will choose for assignment 1. The following slides should indicate what their report should include, as well as what processes they should follow.

Introduce the situation to the students. is too sparse to actually do much work on.

The background (coming next) will set the real scene for the problem.

Section 1: Tasks and Requirements

Introduction: Background to the system

- This is a small library that serves a town of about 10,000 people
- About 500 people use the library each day, although this varies.
- Its holdings include books, CDs, and cassette tapes
- The library already has a computer system that collects, in a database, all its holdings and a unique bar code number. All holdings already have a bar code label on them.
- The library also has a computer system that allows its clients to peruse its holdings (these computers are set up on booths in the middle of the library). As this system is satisfactory, and will not have to be replaced.
- However, the computer system used by librarians and their assistants is badly out of date and awkward to use. This is the one that will be replaced in this project.

Saul Greenberg

This background sets the scene a bit better. It says what is already in the library (in general terms), what the library is like, and more or less states what systems are already in place.

Section 1: Tasks and Requirements

Introduction: Expected Users

- The users of the system are experienced staff: the librarians and library assistants.
  (library clients will not be allowed to use the system)
- Library staff are all experienced at all routine library operations
- The library expects all its staff to be trained on system use, either formally (through a course if necessary), or by apprenticeship (where staff will learn on the job from other proficient staff members)
- Library staff are currently all experience with PCs and the usual suite of applications that run on Window's 95

Saul Greenberg

The expected users tells us quite a bit.

We now know that only trained and computer-proficient staff will use it, so we do not have to design a walk up and use system.

We also know that the staff know their jobs, so what we have to do is design a system that will let them do their jobs (rather than teach them how to do their work-related tasks).

Section 1: Tasks and Requirements

Introduction: Work Contexts

- Librarians do many chores, such as re-shelving books, tidying up the library, helping clients find books, sorting new holdings, and so on
- One of their chores is to work the counter, which is the emphasis of this project.
- During quiet periods
  - staff do routine chores, and only go to the counter when a client approaches it.
- During busy periods,
  - one staff member is always at the counter, and calls other staff to the counter when the line-up starts growing
  - lineups routinely grow to about 3-5 people, with longer lineups being rare
  - customers rarely have to wait more than 5 minutes before being served

Saul Greenberg

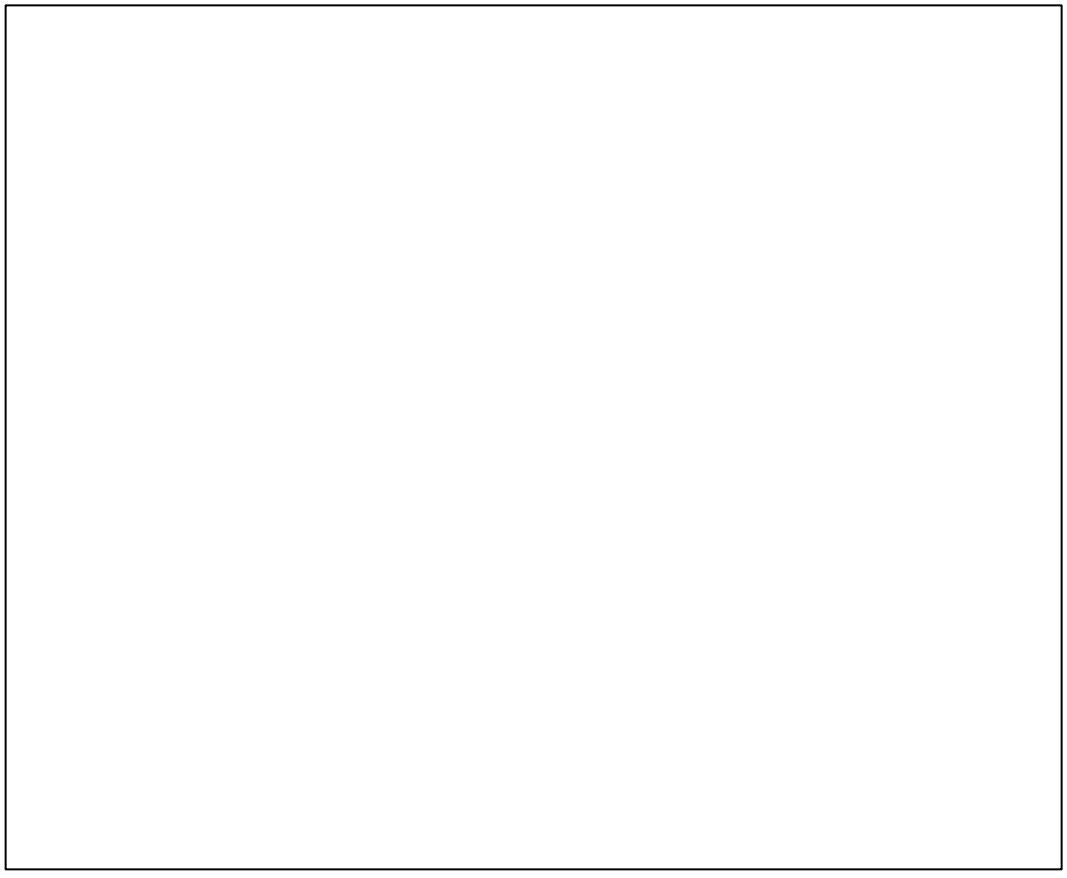
The work context describes in general what the staff does, as well as gives some specifics on routine work that the system will help the staff handle.

The work context continues by stating how staff deal with busy and quiet periods, as well as expected (and acceptable) lineup size and waiting periods

Section 1: Tasks and Requirements

Introduction: Work Contexts

- The library moves from quiet to quite busy periods
- During quiet periods, the staff do their routine chores, and only go to the counter when a client approaches it.
- During busy periods, a staff is always at the counter, and calls other staff to the counter when the line-up starts growing
- During busy periods, lineups routinely grow to about 3-5 people, with longer lineups being rare
- However, people rarely have to wait more than 5 minutes before being served



Section 1: Tasks and Requirements

Introduction: What the envisaged system will be used for

- The system will handle routine counter work, which now includes
  - helping staff answer customer requests (either face to face or by the phone)
  - telling clients their status ie what books they have out, what fines are pending, and so on
  - checking library holdings in and out
  - checking for late fines and informing clients
  - collecting fines
  - providing new library cards
  - checking for expired cards
  - renewing library cards
  - phoning people who have overdue books

Saul Greenberg

The work context continues by stating how staff deal with busy and quiet periods, as well as expected (and acceptable) lineup size and waiting periods

Section 1: Tasks and Requirements

Introduction: System constraints

- The library already has a well maintained computer system that contains all the holdings in a reasonably fast database. The library does not expect to change this system, and ask that your system link into it.
- The library also have several modern PCs running Window's 95, already located on the counter. They expect that your system will be built on that platform
- There is a very modest budget for additional equipment, if needed.

Saul Greenberg

The constraints set some real world limits on what we can do. That is, an immersive 3-d environment won't fly!

We have a reasonable (we hope) back end system already in place, and we have to develop our application to run within a conventional GUI

Section 2: Concrete task examples

What you have to do

- for details on what makes a good task, read the assigned readings, the assignment and Appendix 1 carefully
- go the work site (if possible), and
  - interview staff/end-users
  - observe people doing real tasks

Stress that they *must* read the assignment details and readings. You should read them as well, and use them as appropriate

If you have time, you may even want to walk through sections of the handouts (your choice)

Stress that good task descriptions are the cornerstone for much of their work. If they blow it here, it will seriously affect their project.

Section 2: Concrete task examples

Some examples garnered by talking to the library staff and observing them work

- Joan, a regular and experienced library employee, is working behind the counter. Mary, a regular library customer brings three books to the counterand asks that they be checked out. These are: <3 books should be listed here>. Mary does not have her library card. Joan finds Mary library number, checks out the books for her, and reminds Mary that she has some late fines to pay. Mary says she will pay for them next time. Joan gives Mary the books, and Mary leaves.

Discussion.

- This is a fairly routine task, as validated by Joan. Books are checked out, and the client is reminded of late fees. It also illustrates some working practices. For example, most clients do not have their cards, and expect librarians to look them up. This is acceptable library policy. Similarly, staff can choose to allow customers to defer paying their fees.
- Joan is also a typical system user, while Mary is a typical client

Saul Greenberg

Stress the characteristics of a good task:

Says what the user wants to do but does not say how they would do it

-doesn't make any assumptions about the system interface

Are very specific

-says exactly what the user wants to do

Describes a complete job

- lists all aspects of the task, from beginning to end

Says who the users are

-use particular people, if possible

-reflects real interests of real users

Section 2: Concrete task examples

- Joan's next client is Saul. Saul is a regular library patron, well known to staff, and is somewhat notorious for exceeding book checkout limits, returning books late, never having his library card, and accumulating library fines. He is returning 3 books (2 which are overdue) <list here>, and Joan starts checking them in. One of the books, however, is missing its bar code number. She looks up the title, checks it in, and sets it aside for repair.
- While she is doing this, Saul brings 8 books to the counter and 5 audio tapes <list here>. Joan starts checking out his holdings. , She notices that he has reached the maximum level of $10. books. She tells Saul about the fine, and he pays it. After four books, she notices that Saul is max'ed out on the allowable books and asks Saul if she can check the rest out on his son's account, and he says yes. She continues to do so on his son's account until all books are checked out. Saul then asks if he can renew any books that are overdue. 3 of them are, and Mary does this.

Discussion.

- This is a complex task that contains many situations that, while less routine, are still important. In particular, it contains some elements of the library policy (eg dealing with maximum fines), some workarounds (eg, max'ed out accounts), and some less routine situations (eg, missing bar codes).

Saul Greenberg

Indicate that you will not be showing many more tasks, but that this should give a feel for what a task description should look like.

Also indicate that the tasks may include events that are not part of a "formalized" procedure eg, workarounds, special cases, and so on. In one sense, these tasks include the folklore of how people really work, rather than a rationalization of what they are supposed to do.

Section 3: Tentative list of requirements

Absolutely must include:

- rapid check-in and check-out of holdings
- lookup of customers
- status of customer holdings (eg, books checked out, books overdue, fines)
- ability to clear fines
- ….

Should include:

- …

Could include:

- …

Exclude:

- ...

Discussion

- Why items are in those categories

Saul Greenberg

This is where you start thinking about what the system requirements are.

Note that they are prioritized! In all probability, you will not be able to implement all requirements. Its up to you to decide which of these are critical, and which are not.

What I did not cover:

you should be validating all these tasks with the staff ie, whether they are correct, whether details are missing, the importance of the task elements, and situations that have not been covered by the set of tasks

you should validate the system requirements with the staff. What you think is important may not be important to them and vice versa.