Lab 2

Designing Input screens, output reports, and GUI

User interface Design:

Consists of all the hardware, software, screens, menus, functions and features, that affect two-way communications between the user and the system

Input design and prototyping:

To input business data into a system, the system analyst may have to design source documents, input screens, methods and procedures for getting the data into the system (from customer to form to data entry clerk to computer).

We have important concepts to understand before we can start input process:

- **Data capture**: the identification and acquisition of new data.
- **Data entry**: the process of translating data into a computer-readable format.
- **Source document**: a form used to record data about a transaction.
- **Batch processing**: a data processing method whereby data about many transactions is collected as a single file which is then processed.

Input methods and implementation:

In particular, we are interested in how the choice of a method affects data capture, entry and processing.

- *Keyboard*
- *Mouse*
- *Touch screen*
- *Smart cards*
- *Sound and speech*
- *Point of sale*: these terminals capture data at the point of sale and provide time-saving ways to enter data, perform transactional calculations, and produce some output. e.g. Automatic teller machine (ATMs).
- *Optical mark*: like optical character recognition (OCR)
- *Biometric*: based on unique human characteristics. For example, finger prints or voice patterns.

System user issues for input design:

Because inputs originate with system users, human factors play a significant role in input design. Inputs should be as simple as possible and be designed to reduce the possibility of incorrect data being entered. The volume of data to be input should be minimized. The more data that is input, the greater the potential number of input errors and the longer it takes to input that data. Thus, numerous considerations should be given to the data that is captured for input. These general principles should be followed for input design:

- *Capture only variable data*: don't enter constant data because it is probably stored in database table
- *Don't capture data that can be calculated or stored in computer programs.*
- *Use codes for appropriate attributes*: codes can be translated in computer programs by using tables.

If source documents are used to capture data, they should be easy for system users to complete and enter into the system. The following suggestions may help:

- *Include instructions for completing the form*: remember that people don't like to have to read instructions printed on the back side of a form
- *Minimize the amount of handwriting*: many people suffer from poor penmanship. The data entry clerk may misread the data and input incorrect data.
- *Data to be entered should be sequenced so that it can be read from top to bottom and left to right. The following two figures depict a good flow and a bad flow*
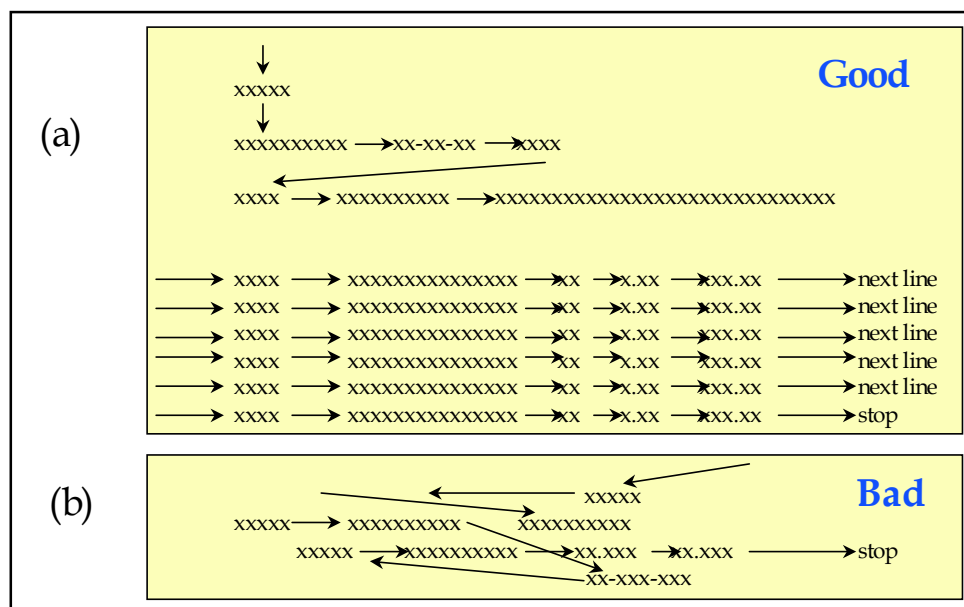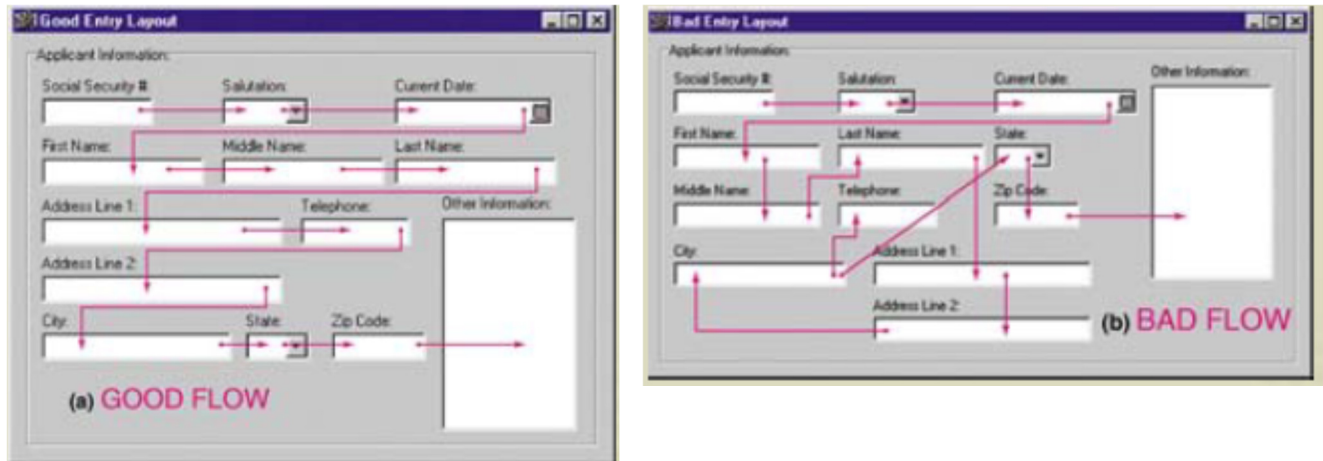
Internal controls- data editing for inputs:

Internal input controls ensure that the data to the computer is accurate and that the system protected against accidental and intentional errors and abuse. The following internal control guidelines are offered:

- **Existence checks**: determine whether all required fields on the input have actually been entered.
- **Data-type checks**: ensure that the correct type of data is input. For example, alphabetic data should not be allowed in a numeric field.
- **Domain checks**: determine whether the input data for each field falls within the range of values defined for that field
- **Combination checks**: determine whether a known relationship between two fields is valid.
- **Format checks**: compare data entered against the known formatting requirements.

Common GUI controls for inputs:

1- **Text Box**: this control requires that the user type the data inside the box. It is most appropriately used when the input data values are unlimited in scope and the analyst is unable to provide the users with a meaningful list of values from which they can select.
2- **Radio button**: this control provides the user with an easy way to quickly identify and select a particular value from a value set. It is most appropriate when a user may be expected to input data that has a limited predefined set of mutually exclusive values.
3- **Check Box**: this control provides the user with the flexibility of selecting the value via the keyboard or mouse. If a user needs to input a data field whose value set consists of a

simple yes or no value, a check box control could be used. It also offers a visual means for the user ti input such data.

4- **List Box**: this control requires that the user select a data item's value from a list of possible choices. It is appropriate for use where there is limited screen space available and the input data item has a large number of predefined, mutually exclusive values from which to choose.it allow user to choose more than 1 item from list.

5- **Drop-Down List**: this control requires the user to select a data item's value from a list of possible choices. It should be used in cases where the data item has a large number of predefined values and screen space availability prohibits the use of a list box. if user should choose 1 item only.

6- **Combination Box**: often called a combo box, that is combines the capabilities of a text box and list box. It is most appropriate used where screen space is limited and it is desirable to provide the user with the option of selecting a value from a list or typing a value that may or may not appear as an option in the list.

7- **Spin Box**: this control allows the user to enter data directly into the associated text box or to select a value by clicking on the buttons to scroll or spin through a list of values. It is most appropriately used to allow the user to make an input selection by using the buttons to navigate through a small set of meaningful choices or by typing the data value into the text box.

8- **Buttons**: they allow the user to commit all of the data to be processed or cancel transaction, or get help.

Cairo University
Faculty of computers and information
System Analysis and Design 2
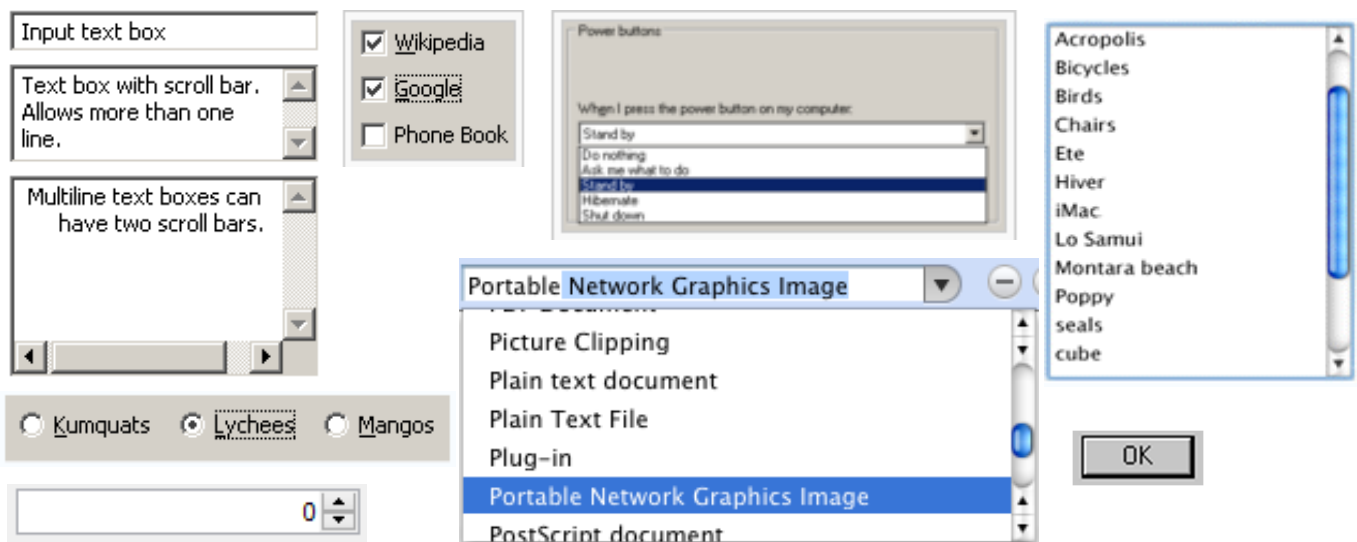
Advanced input controls:

The following two figures illustrate additional controls for data input

- *Drop-down calendar*
- *Slider edit control*
- *Masked edit control*
- *Ellipses control: clicking on the three dots causes a pop-up dialogue to appear for data entry.*
- *Alternative numeric spinner*
- *Internet hyperlink*
- *Check list box*
- *Check tree list box*
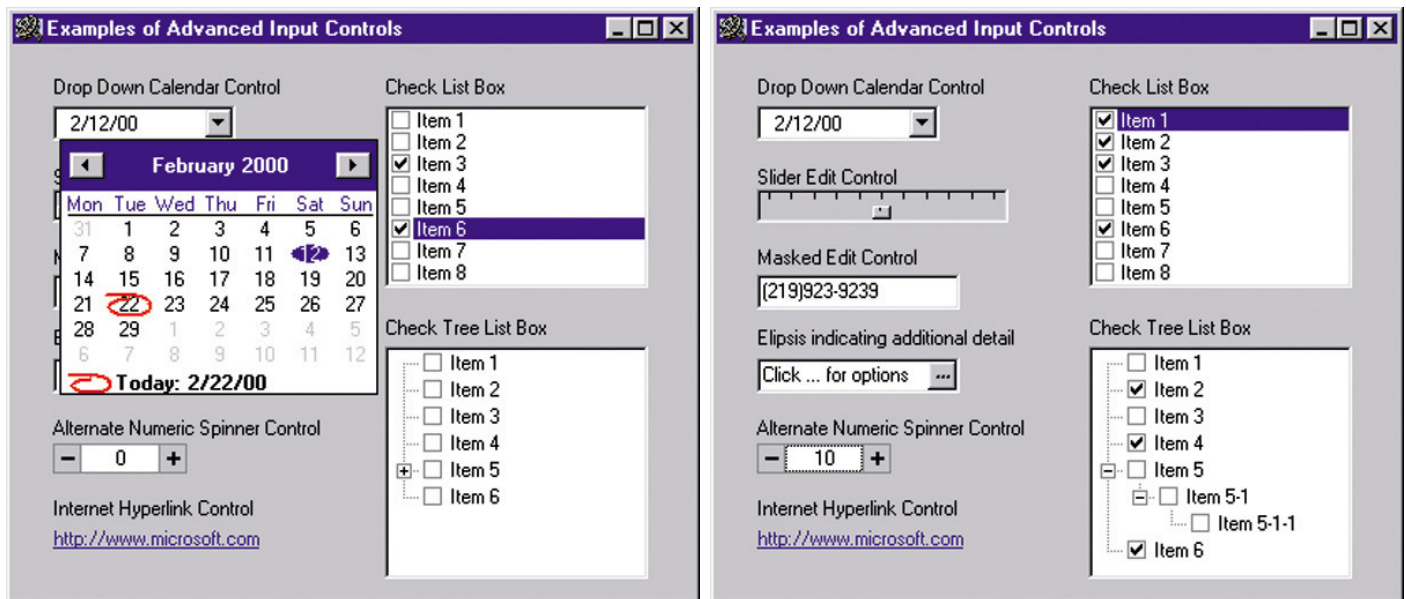
The input design process:

Input design is not a complicated process. Some steps are essential, and others are dictated by circumstances. The steps are:

1- *Identify system inputs and review logical requirements.*
2- *Select appropriate GUI controls*
3- *Design, validate and test inputs using some combination of:*
   a. *Layout tools (e.g., hand sketches, printer/display layout charts)*
   b. *Prototyping tools*
4- *If necessary, design the source document.*

Output design and prototyping:

Outputs present information to system users. Outputs are the most visible component of a working information system. As such, they are often the basis for the users' and management's final assessment of the system's value.

Distribution and audience of outputs:

One way to classify outputs is according to their distribution inside or outside the organization and the people who read and use them. There are three basic types of internal outputs:

- **Detailed reports**: present information with little or no filtering or restrictions.
- **Summary reports**: categorize information for managers who don't want to wade through details.
- **Exception reports**: include only exceptions to some conditions or standard.

Implementation methods for outputs:

- **Printed output**: the most common medium for computer outputs is paper printed outputs.
- **Screen output**: the fastest growing medium for computer outputs is the online display of the information on a visual display device
- **Point-of-sale terminals**: they are both input and output devices. e.g., ATMs
- **E-mail**
- **Hyperlinks**: many outputs are now web-enabled.

Before designing output task, ask yourself several questions:

- What is the purpose of the output?

- Who wants the information, why it is needed, and how will it be used?

- What specific information will be included?

- Will the output be printed, viewed on-screen, or both? What type of device will the output go to?

Output design guidelines:

The following general principles are important for output design:

1- *Computer outputs should be simple to read and interpret.*
   a. *Every output should have a title*
   b. *Every output should be dated and time-stamped*
   c. *Reports and screens should include sections and headings to segment information*
   d. *In form-based outputs, all fields should be clearly labeled*
   e. *In tabular-based outputs, columns should be clearly labeled*
   f. *Only required information should be printed or displayed*
   g. *Information should be balanced on the report or display*
   h. *Users must be able to easily find the output, move forward and backward, and exit the report*
2- *The timing of computer outputs is important*
3- *The distribution of (or access to) computer outputs must be sufficient to assist all relevant system users.*
4- *The computer outputs must be acceptable to the system users who will receive them*

The output design process:

Output design is not a complicated process. Some steps are essential, and others are dictated by circumstances. The steps are:

1- *Identify system outputs and review logical requirements.*
2- *Specify physical output requirements*
3- *As necessary, design any preprinted external forms*
4- *Design, validate and test outputs using some combination of:*
   a. *Layout tools (e.g., hand sketches, printer/display layout charts)*
   b. *Prototyping tools*
   c. *Code generation tools*

User interface design

We will integrate input and output design into an overall user interface that establishes the dialogue users and computer. The dialogue determines everything, from starting the system or logging into the system to setting options and preferences, to getting help.

Type of computer users:

- *Expert user*
- *Novice user*

Human factors:

Before designing user interfaces, you may find it useful to understand the elements that frequently cause people to have difficulty with computer systems. To solve the problems results from these elements you can follow the following rules:

- *Understand your user and their tasks*
- *Involve the user in interface design*
- *Test the system on actual user*
- *Practice iterative design*

Human engineering guidelines:

- *The system user should always be aware of what to do next*
    - *Tell the user what the system expects right now*
    - *Tell the user that data has been entered correctly*
    - *Tell the user that data has not been entered correctly*
    - *Explain to the user the reason for a delay in processing*
    - *Tell the user that a task was completed or was not completed*
- *The screen should be formatted so that the various types of information, instructions and messages always appear in the same general display area*
- *Messages, instructions or information should be displayed long enough to allow the system user to read them.*
- *Default values for fields and answers to be entered by users should be specified*
- *Anticipate the errors user might make*

Dialogue tone and terminology:

The overall flow of screens and messages is called a dialogue. With respect to the tone of the dialogue, the following guidelines are offered:

- *Use simple, grammatically correct sentences*
- *Don't be funny or cute*
- *Don't be condescending*

With respect to terminology, the following suggestions may prove helpful:

- *Don't use computer terminology*
- *Avoid most abbreviations*
- *Use simple terms*
- *Be consistent in your use of terminology: don't use edit and modify to mean the same action*
- *Carefully phrase instructions*
  - o *Use select or choose instead of pick*
  - o *Use type not enter*
  - o *use press not hit*

The user interface design process:

1- *chart the user interface dialogue*
2- *prototype the dialogue and user interface*
3- *obtain user feedback*
4- *if necessary, return to step 1 or 2*

Mock-up and GUI tools

Most development environments, such as Microsoft's visual studio, eclipse and Netbeans used for GUI prototyping applications, and there is other tools used to make mockups like pencil http://c2.com/cgi/wiki?GuiPrototypingTools that can be easily used to construct nonfunctional prototypes of user interface screens. The key term here is nonfunctional. The forms will look like real, but there will be no code to implement any of the buttons or fields [no business logic implemented in prototype even interactive GUI].
You can design your screen either by layout tools (e.g., hand sketches, printer/display layout charts) or you can use any GUI tools to design and automate this process

Cairo University
Faculty of computers and information
System Analysis and Design 2

Example:

We need to design a CV generator system. User will enter his data in a screen and system will generate his CV. Design your screen in any GUI Mockup tool you prefer

We can use mockup screen tool
http://www.mockupscreens.com/index.php?page=Download_trial
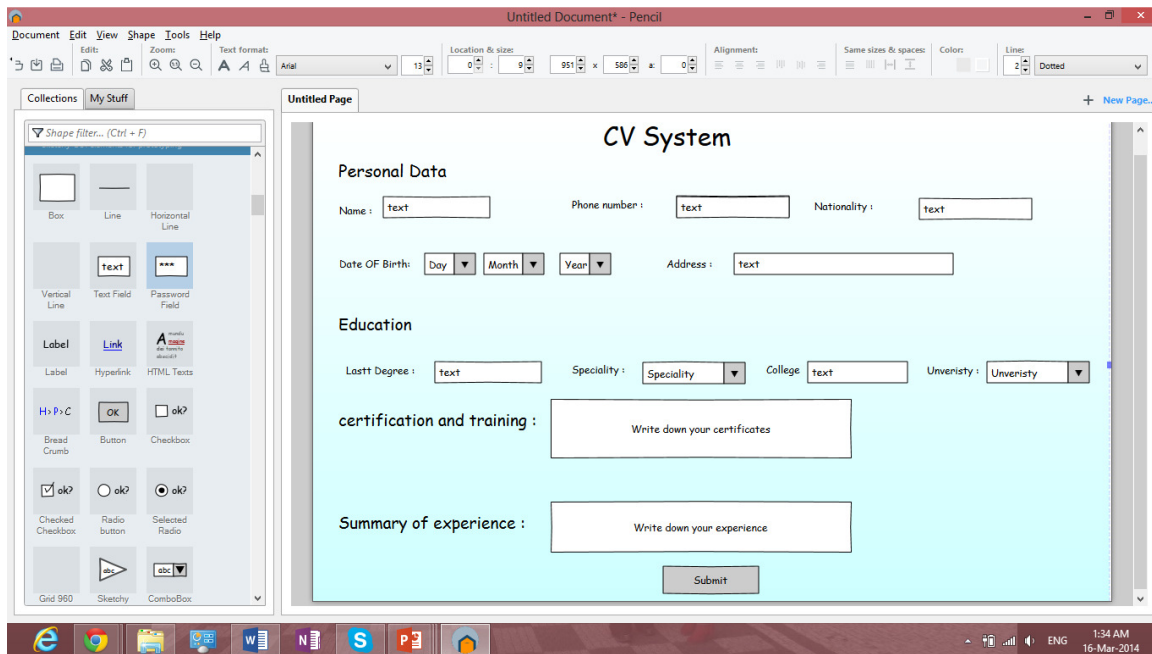or 10 screens (online tool)
http://www.10screens.com/app.html#

10 screens:

Pencil screen

Mock up screen tool