Name: Shivam Tiwari **RollNo:5117060**

EXPERIMENT 05

Aim: Demonstrate an interaction cycle

Theory:

Introduction:

The concept of Interaction Styles refers to all the ways the user can communicate or otherwise interact with the computer system. The concept belongs in the realm of HCI or at least has its roots in the computer medium, usually in the form of a workstation or a desktop computer. These concepts do however retain some of their descriptive powers outside the computer medium. For example, you can talk about menu selection (defined below) in mobile phones. Although WIMP interfaces (windows, icons., menus, pointers,), which are a combination of Menus, dialog boxes, and 'point and click' currently dominates interface designs there are other forms of interfaces:

1. Command line interface
2. Menus
3. Question-answer/Dialog boxes
4. Forms
5. Spreadsheets
6. Point and click
7. Natural language
8. Command Gesturing
9. General Gesturing
10. Direct manipulation
11. Tangible interaction

We will go through the list and identify the technique in terms of how fast it is for the user, how flexible it is for users or expressive it is for the user, how long it takes for the user to learn, and how hard it is for the programmer to implement.

Command line interfaces are the original interfaces for the computer and are still used. Examples of command line interfaces are UNIX operating system commands and the VI editor. They are loved by system administrators for their accuracy, speed and flexibility. If the user knows the commands then typing is faster than searching for it in menus. Consequently, some applications try to offer both; for example auto cad or Windows Excel. Users can directly create script files and verbally specify the command sequence. Some commands can be hard to visualize and searching for commands or files can be frustrating and slow. The interface is easy for programmers to implement.

Menus are a basis of WIMP interfaces and a favorite with inexperienced users. Novice users can easily find commands. Searching the menus, the user builds up a metaphor for the application. Using menus is slow but fun. Using toolkits, menus are not too bad to program. Dialogs or “question and answer” boxes are another old interface style. They are windows that pop up asking for information, fields to be filled or buttons to be pressed. Dialogue windows have been around before WIMP interfaces, they first appeared in database entry application programs. Other examples are wizards and some help agents. They ask a sequence of questions using dialog boxes and in order to determine what the user needs.

They are a rather inflexible form of interface. Dialog boxes are easy to understand but not very flexible or fast. They are easy to program. Forms are much like dialog boxes. They can be more sophisticated, and like dialog boxes not very flexible for the user. Spreadsheets are a flexible and powerful form of interface, especially if the user can specify cell types. Data entry in spreadsheets is typically slow. They are more difficult to program. Point and click interfaces were made popular by the web. They represent I and P in WIMP. They were very suitable for the initial web browsers (gopher) when web pages were all text. Users knew to interpret the underscore as a link to another web page. Now, links are hidden, for examples in images. Icons on the desktop is another example of point and click style interface. The notion of point and click is a short interaction that results in a very specific result. Because the user must move the mouse, this interface style tends to be slow for the user. It is flexible because many different kinds of UI objects can be pointed at. Short key interaction is a “point and click” interaction style without the point. Both are generally easy to implement.

One of the most common examples of natural language interfaces are the interfaces for search engines. Users type words into the search box that the software interprets and return the result for the user. They are hard to implement, but can be very flexible for the user. Command gesturing interface style selects an object and uses a gesture to issue commands. In essence, it is a generalization of “point and click” interfaces. Examples are swiping on smart phones and windows OS. Some games, Brother in Arms, use command gesturing. Keith Ruthowski has demonstrated that a pie menu can become a form of command gesturing, and once the gestures are learned nearly as accurately and fast as text entry. Because algorithms for interpreting gestures are in its infancy, the flexibility of command gesturing is not known. They are difficult to implement. Learning a large set of gestures can take a long time. General gesturing is a more general interface style than command gesturing. There does not have to be an object and the gesture does not have to represent a command. Examples of general gesturing are drawing applications and text entry by writing, like in a notebook. The Wii is advancing general gesturing in games. Because this is a very new interaction style, it is unknown how easy it is to learn, but it should be more flexible. It is more difficult to implement. Direct manipulation is closely related to command gesturing. An example of direct manipulation is drag and drop files into folders or trash. Drawing applications use direct manipulations. They can be slow to use but are fast to learn. They can be difficult to implement.

Tangible interactions refer to manipulating physical objects other than the mouse and keyboard. There are few current popular examples, but RFID and NF technology does make some tangible interactions possible. Low tech examples of tangible interactions are real buttons, switches and sliders. They can be fast or slow to use, but should be easy to learn, and can be hard to implement.

Interaction Aids

Interaction aids are techniques to make user interaction with your app smoother and faster. For example, if the application usage is to fill out a form multiple times then the app can remember the prior value entered by the user and offer it as a default value in the field. There are many interaction aids that your app can use. Let us list some of them.

Form filling interaction aids:

• Sensible defaults regardless of usage history
• Defaults based on usage history
• Defaults based on location
• Displaying units for numbers
• Automatic formatting entries, for example spaces in credit card number input or dashes in phone number
• Tab-ing to the next field
• Removing or adding “enter” quick key submission
• Warning notice when leaving a form

Other interaction aids:

• Short key interaction
• Screen illumination based on ambient light
• View orientation or lack of reorientation based on app context, eg rotating the phone
• Widget sizing based on screen size and task
• Adding or removing potential actions based on prior interaction, eg graying menu items or adding buttons
• Interactions based on user preferences, for example short keys from a profile page
What other interaction aids can you think of?

Conceptual Interaction Models

Preece, Rogers and Sharp in Interaction Design propose that designers should understand users‟ conceptual models for interaction. The understanding can guide designers to the proper

interaction techniques for their system. The most important thing to design is the user's conceptual model. Everything else should be subordinated to making that model clear, obvious, and substantial. That is almost exactly the opposite of how most software is designed. (David Liddle, 1996, Design of the conceptual model, In Bringing Design to Software, Addison-Wesely, 17-31)
The HCI designers‟ goal is to understand the interaction in terms of the users' understanding of them. Preece, Rogers and Sharp propose four conceptual models for interaction concepts, based on the type of activities users perform during the interaction.

• **Instructing** – issuing commands to the system
• **Conversing** – user ask the system questions
• **Manipulating and Navigation** – users interact with virtual objects or environment
• **Exploring and Browser** – system provides structured information

I propose additional conceptual interactions that are more passive:
• **Passive Instrumental** – the system provides passive information to the user, either from the environment or user‟s action.

Users may interact with a system using more than one conceptual interaction model.

Instructional Interactions

Issuing commands is an example of instructional interactions. Instructional interactions are probably the most common form of conceptual interactions. It allows the user the most control over the system. Specific examples vary from using a VCR to programming. In most cases, issuing commands to the operating system (example of "command line" interaction style) are instructional interactions. Icons, menus and control keys are examples of improving the usability command line like instructional interaction. Instructional interactions tend to be quick and efficient.

Conversational Interactions

The conversational Interactions are interactions like user-system dialog. Examples of systems that are primarily conversational are help-systems. Agents (such as the paper clip) use conversational interaction. Implementing conversational models may require voice recognition and text parsing or could use forms. The advantage of conversational models is that it can be more natural, but it can also be a slower interaction. For example, using automated phone based systems is a slow conversational interaction interface. Another disadvantage of conversational interaction is that the user may believe that the system is smarter then it really is, especially if the system uses an animated agent.

Manipulating and Navigational Interactions

This model describes the interaction of manipulating virtual objects or navigating virtual worlds. Navigational interactions are popular in computer games. Manipulating interactions occur in drawing software. Navigational interactions occur even in word processors, for example zooming and using the scroll bar. Direct manipulations are manipulating interactions. Ben Shneiderman (1983) coined the phase and posed three properties.

- continuous representations of objects
- rapid reversible incremental actions with immediate feedback
- physical actions

Apple was the first computer company to design an operating system using direct manipulation on the desktop. Direct manipulation and navigational interactions have a lot of benefits. They are easy to learn, easy to recall, tend to have less error, give immediate feedback, and produce less user anxiety. But they have several disadvantages: the interactions are slower and the user may believe that the interaction is more than it really is. Poor metaphors such as moving the icon of a floppy to eject the floppy can confuse the user.

Exploring and Browsing Interactions

Exploring and browsing interactions refer to searching structured information. Examples of systems using exploring interactions are Music CDs, Movie DVDs, Web, portals. Also searching for files using window explorer is an example of Exploring interactions. Not much progress has been made in this conceptual model for exploring and browsing interactions,probably because structuring information is a non-trivial task and is hard to model.

Passive Informative Interactions

Passive Informative interactions are similar to instruments, for example the speedometer in an auto dashboard. They can provide feedback to users" actions or movements, such as a GPS interface. They can also provide information to changes in the environment such as a light meter or an image in a viewfinder. Smart phones frequently used as instruments and make use of passive instrumental interactions.
Another example of passive informative interaction is using the smart phones to read books. The interaction is very passive and one way. The system is providing information to the users. The user primarily gestures to progress through the book. Viewing images is another passive informative interaction with only interactions for zooming and panning. Passive informative interaction may be a simplified Manipulating and Navigational Interactions.
We can make a table summarize interaction styles and conceptual interaction models. The related conceptual interaction model is the most common model that is supported by the interaction style. Implementation is how hard for the designer to implement. Because I made the table we should go through it and correct it.

Interface	User speed	Flexibility	Learning	Implementation	Conceptual Model
CLI	Fast	High	Slow	Easy	Instructing
Menus	Slow	Medium	Medium	Medium	Instructing
Dialog	Slow	None	Fast	Easy	Conversing
Forms	Medium	Not much	Fast	Easy	Instructing, Conversing, browsing
Spreadsheet	Slow	High	Slow	Hard	Instructing, Conversing
Point & click	slow-fast	none	fast	easy	Instructing, manipulating, browsing
Natural lang	Fast	Very	Slow-fast	Hard	Conversing
Command gesturing	Fast - medium	medium	hard	hard	Instructing
General gesturing	Fast	high	Slow - fast	hard	Manipulating
Tangible Interaction	Slow - fast	low	fast	Medium to hard	Manipulating, browsing

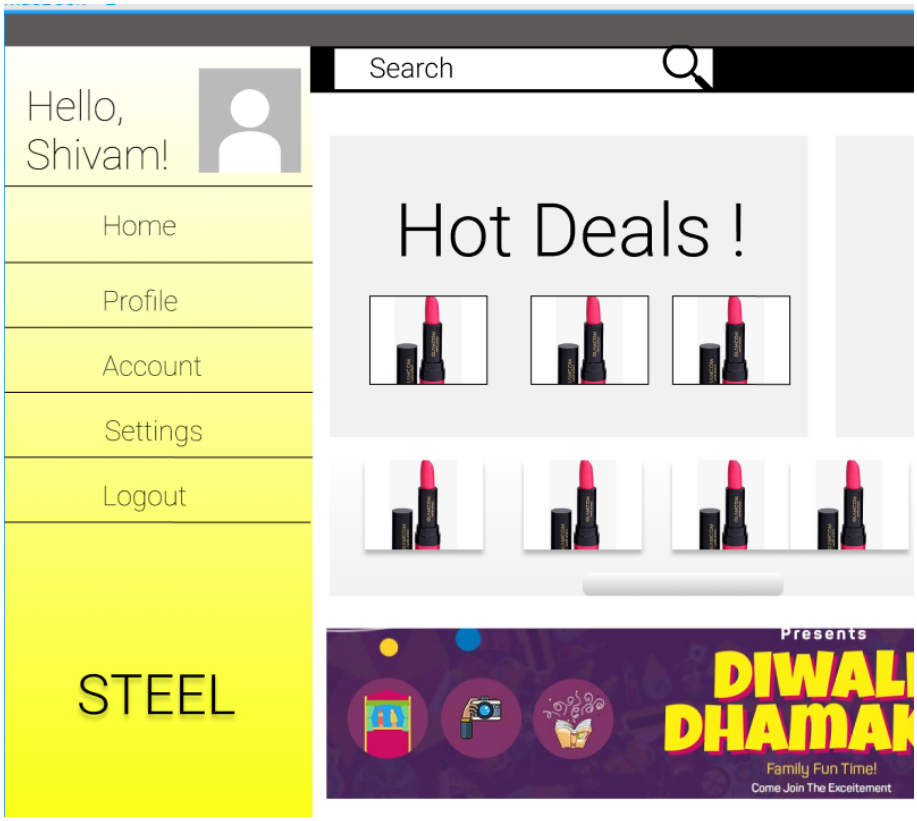
Tools to be used: HTML5 and CSS

Conclusion:
Interaction styles allow the users to communicate or interact with the applications. These include forms, menus, dialog, CLI, gesturing interfaces, NLP, etc. We studied different types of interfaces,

their user speeds, flexibility, learning and the type of conceptual model to which they belong. Interaction aids makes the user's interaction with the application more efficient, smooth and fast. I have designed a form as an interaction style for my application.

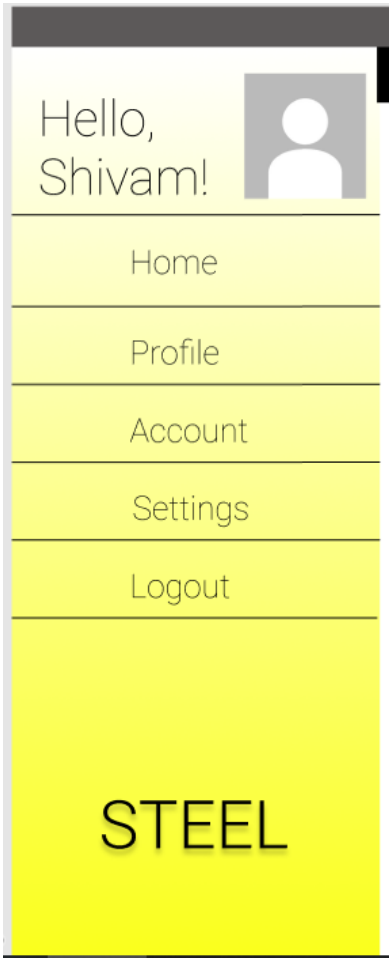
Output:
Interaction styles used

1. Clicks: The click interaction style is used to navigate from one page to another. As you click on a particular item it will navigate you to another page. Clicks are used at many places, to submit forms, to select a particular choice.



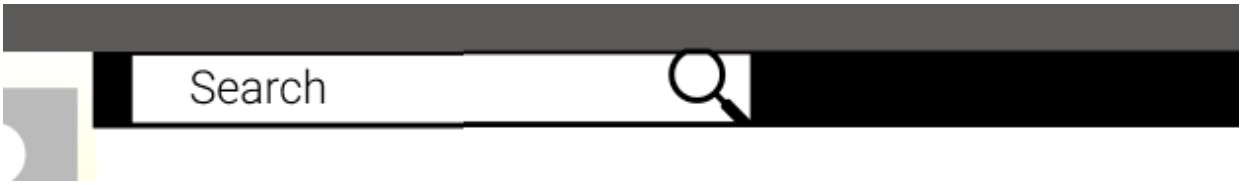
Click can be defined as, The notion of point and click is a short interaction that results in a very specific result. Because the user must move the mouse, this interface style is slow. It is flexible because many different kinds of UI objects can be pointed at. Short key interaction is a point and click interaction style without the point.

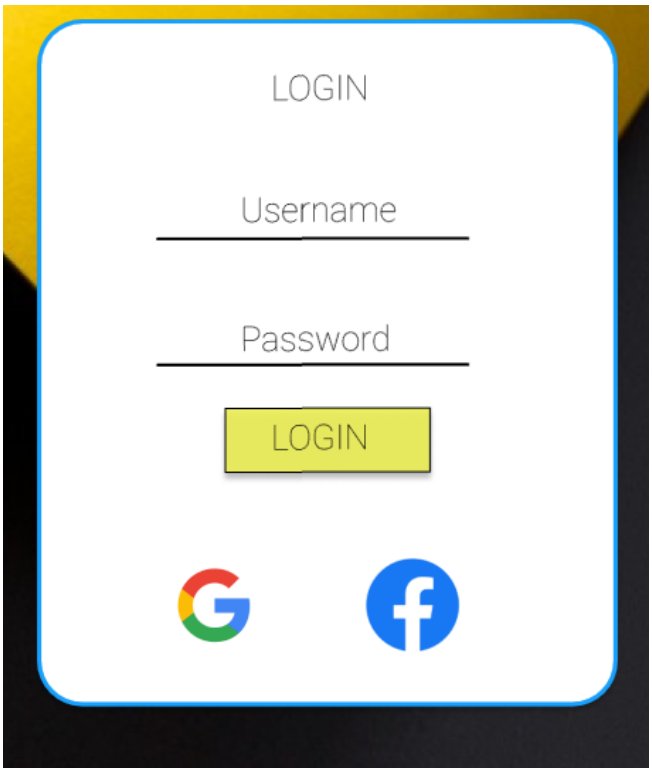
2. Menu: A menu bar is added to the prototype, it helps the user to jump from one section to another. The T&T website has a menu bar comprising the following elements. Home, Reviews, Booking, AboutUs, Members. Each element in the menu serves a different task.

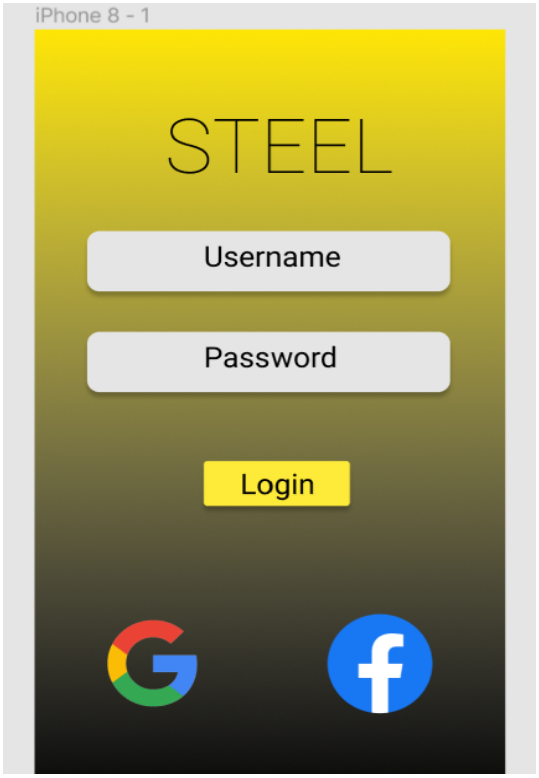


Menu can be defined as a basis of WIMP interfaces and a favorite with inexperienced users. Novice users can easily find commands. Searching the menus, the user builds up a metaphor for the application. Using menus is slow but fun. Using toolkits, menus are not too bad to program.

3. Forms : Forms are used to take user input values. Forms are used to take Login entries, it is also used in bookings where the user has to enter the dates. Search bar is also a type of form, which takes manual entry and on clicking submit it shows the desired outcome.







Forms are sophisticated dialog boxes and not much more flexible. Spreadsheets are a flexible and powerful form of interface, especially if the user can specify cell types. Data entry in spreadsheets is typically slow. They are more difficult to program.

4.Tangible Interaction: The tangible interaction used in this prototype is buttons. These buttons are used to select a particular category e.g., Mens Section, Female Section, Sale etc.

Tangible Interaction is a very interdisciplinary area. It spans a variety of perspectives, such as HCI and Interaction Design, but specializes on interfaces or systems that are in some way physically embodied - be it in physical artefacts or in environments. Tangible interactions refer to manipulating physical objects other than the mouse and keyboard. There are few current popular examples, but RFID and NF technology does make some tangible interactions possible. Low tech examples of tangible interactions are real buttons, switches and sliders. They can be fast or slow to use, but should be easy to learn, and can be hard to implement.

5. Dialog : Dialog boxes give users a confirmation layer before proceeding for any further important task. It may also be used for displaying certain specific forms of messages.