Human Computer Interface
designing user-friendly interfaces

tutorialspoint
SIMPLY EASY LEARNING

www.tutorialspoint.com

About the Tutorial

This tutorial provides the basic knowledge on human computer interface and designing. It also throws a light on the current tools and practices and the future aspects of HCI designing.

Audience

This tutorial has been intended for the users willing to take the human computer interactions as a next level of study in their career.

Prerequisites

The basic knowledge of Graphical User Interface, Object Oriented Programming and software tools and menus are the only prerequisite for this tutorial.

Copyright and Disclaimer

Table of Contents

iii

Human Computer Interface (HCI) was previously known as the man-machine studies or man-machine interaction. It deals with the design, execution and assessment of computer systems and related phenomenon that are for human use.

HCI can be used in all disciplines wherever there is a possibility of computer installation. Some of the areas where HCI can be implemented with distinctive importance are mentioned below:

- Computer Science: For application design and engineering.

- Psychology: For application of theories and analytical purpose.

- Sociology: For interaction between technology and organization.

- Industrial Design: For interactive products like mobile phones, microwave oven, etc.

The world's leading organization in HCI is ACM – SIGCHI, which stands for *Association for Computer Machinery - Special Interest Group on Computer–Human Interaction*. SIGCHI defines Computer Science to be the core discipline of HCI. In India, it emerged as an interaction proposal, mostly based in the field of Design.

Objective

The intention of this subject is to learn the ways of designing user-friendly interfaces or interactions. Considering which, we will learn the following:

- Ways to design and assess interactive systems.

- Ways to reduce design time through cognitive system and task models.

- Procedures and *heuristics* for interactive system design.

Historical Evolution

From the initial computers performing batch processing to the user-centric design, there were several milestones which are mentioned below:

- Early computer (e.g. ENIAC, 1946): Improvement in the H/W technology brought massive increase in computing power. People started thinking on innovative ideas.

- Visual Display Unit (1950s): SAGE (semi-automatic ground environment), an air defense system of the USA used the earliest version of VDU.

- Development of the Sketchpad (1962): Ivan Sutherland developed Sketchpad and proved that computer can be used for more than data processing.

- Douglas Engelbart introduced the idea of programming toolkits (1963): Smaller systems created larger systems and components.

- Introduction of Word Processor, Mouse (1968): Design of NLS (oNLine System).

- Introduction of personal computer Dynabook (1970s): Developed *smalltalk* at Xerox PARC.

- Windows and WIMP interfaces: Simultaneous jobs at one desktop, switching between work and screens, sequential interaction.

- The idea of metaphor: Xerox star and alto were the first systems to use the concept of metaphors, which led to spontaneity of the interface.

- Direct Manipulation introduced by Ben Shneiderman (1982): First used in Apple Mac PC (1984) that reduced the chances for syntactic errors.

- Vannevar Bush introduced Hypertext (1945): To denote the non-linear structure of text.

- Multimodality (late 1980s).

- Computer Supported Cooperative Work (1990's): Computer mediated communication.

- WWW (1989): The first graphical browser (Mosaic) came in 1993.

- Ubiquitous Computing: Currently the most active research area in HCI. Sensor based/context aware computing also known as pervasive computing.

Roots of HCI in India

Some ground-breaking *Creation and Graphic Communication* designers started showing interest in the field of HCI from the late 80s. Others crossed the threshold by designing program for CD ROM titles. Some of them entered the field by designing for the web and by providing computer trainings.

Even though India is running behind in offering an established course in HCI, there are designers in India who in addition to *creativity* and *artistic expression*, consider design to be a *problem-solving activity* and prefer to work in an area where the demand has not been met.

This urge for designing has often led them to get into innovative fields and get the knowledge through self-study. Later, when HCI prospects arrived in India, designers adopted techniques from usability assessment, user studies, software prototyping, etc.

2. GUIDELINES IN HCI

Shneiderman's Eight Golden Rules

Ben Shneiderman, an American computer scientist consolidated some implicit facts about designing and came up with the following eight general guidelines:

1. *Strive for Consistency.*

2. *Cater to Universal Usability.*

3. *Offer Informative feedback.*

4. *Design Dialogs to yield closure.*

5. *Prevent Errors.*

6. *Permit easy reversal of actions.*

7. *Support internal locus of control.*

8. *Reduce short term memory load.*

These guidelines are beneficial for normal designers as well as interface designers. Using these eight guidelines, it is possible to differentiate a good interface design from a bad one. These are beneficial in experimental assessment of identifying better GUIs.

Norman's Seven Principles

To assess the interaction between human and computers, Donald Norman in 1988 proposed seven principles. He proposed the seven stages that can be used to transform difficult tasks. Following are the seven principles of Norman:

1. *Use both knowledge in world & knowledge in the head.*

2. *Simplify task structures.*

3. *Make things visible.*

4. *Get the mapping right (User mental model = Conceptual model = Designed model).*

5. *Convert constrains into advantages (Physical constraints, Cultural constraints, Technological constraints).*

6. *Design for Error.*

7. *When all else fails – Standardize.*

Heuristic Evaluation

Heuristics evaluation is a methodical procedure to check user interface for usability problems. Once a usability problem is detected in design, they are attended as an integral part of constant design processes. Heuristic evaluation method includes some usability principles such as Nielsen's ten Usability principles.

Nielsen's Ten Heuristic Principles

1. *Visibility of system status.*

2. *Match between system and real world.*

3. *User control and freedom.*

4. *Consistency and standards.*

5. *Error prevention.*

6. *Recognition rather than Recall.*

7. *Flexibility and efficiency of use.*

8. *Aesthetic and minimalist design.*

9. *Help, diagnosis and recovery from errors.*

10. *Documentation and Help.*

The above mentioned ten principles of Nielsen serve as a checklist in evaluating and explaining problems for the heuristic evaluator while auditing an interface or a product.

Interface Design Guidelines

Some more important HCI design guidelines are presented in this section. General interaction, information display, and data entry are three categories of HCI design guidelines that are explained below.

General Interaction

Guidelines for general interaction are comprehensive advices that focus on general instructions such as:

- *Be consistent.*

- *Offer significant feedback.*

- *Ask for authentication of any non-trivial critical action.*

- *Authorize easy reversal of most actions.*

- *Lessen the amount of information that must be remembered in between actions.*

- *Seek competence in dialogue, motion and thought.*

- *Excuse mistakes.*

- *Classify activities by function and establish screen geography accordingly.*

- *Deliver help services that are context sensitive.*

- *Use simple action verbs or short verb phrases to name commands.*

Information Display

Information provided by the HCI should not be incomplete or unclear or else the application will not meet the requirements of the user. To provide better display, the following guidelines are prepared:

- *Exhibit only that information that is applicable to the present context.*

- *Don't burden the user with data, use a presentation layout that allows rapid integration of information.*

- *Use standard labels, standard abbreviations and probable colors.*

- *Permit the user to maintain visual context.*

- *Generate meaningful error messages.*

- *Use upper and lower case, indentation and text grouping to aid in understanding.*

- *Use windows (if available) to classify different types of information.*

- *Use analog displays to characterize information that is more easily integrated with this form of representation.*

- *Consider the available geography of the display screen and use it efficiently.*

Data Entry

The following guidelines focus on data entry that is another important aspect of HCI:

- *Reduce the number of input actions required of the user.*

- *Uphold steadiness between information display and data input.*

- *Let the user customize the input.*

- *Interaction should be flexible but also tuned to the user's favored mode of input.*

- *Disable commands that are unsuitable in the context of current actions.*

- *Allow the user to control the interactive flow.*

- *Offer help to assist with all input actions.*

- *Remove "mickey mouse" input.*

3. INTERACTIVE SYSTEM DESIGN

The objective of this chapter is to learn all the aspects of design and development of interactive systems, which are now an important part of our lives. The design and usability of these systems leaves an effect on the quality of people's relationship to technology. Web applications, games, embedded devices, etc., are all a part of this system, which has become an integral part of our lives. Let us now discuss on some major components of this system.

Concept of Usability Engineering

Usability Engineering is a method in the progress of software and systems, which includes user contribution from the inception of the process and assures the effectiveness of the product through the use of a usability requirement and metrics.

It thus refers to the *Usability Function* features of the entire process of abstracting, implementing & testing hardware and software products. Requirements gathering stage to installation, marketing and testing of products, all fall in this process.

Goals of Usability Engineering

- Effective to use - Functional

- Efficient to use - Efficient

- Error free in use - Safe

- Easy to use - Friendly

- Enjoyable in use - Delightful Experience

Usability

Usability has three components - effectiveness, efficiency and satisfaction, using which, users accomplish their goals in particular environments. Let us look in brief about these components.

- Effectiveness: The completeness with which users achieve their goals.

- Efficiency: The competence used in using the resources to effectively achieve the goals.

- Satisfaction: The ease of the work system to its users.

Usability Study

The methodical study on the interaction between people, products, and environment based on experimental assessment. Example: Psychology, Behavioral Science, etc.

Usability Testing

The scientific evaluation of the stated usability parameters as per the user's requirements, competences, prospects, safety and satisfaction is known as usability testing.

Acceptance Testing

Acceptance testing also known as User Acceptance Testing (UAT), is a testing procedure that is performed by the users as a final checkpoint before signing off from a vendor. Let us take an example of the handheld barcode scanner.

```
Let us assume that a supermarket has bought barcode scanners from a
vendor. The supermarket gathers a team of counter employees and make
them test the device in a mock store setting. By this procedure, the
users would determine if the product is acceptable for their needs. It
is required that the user acceptance testing "pass" before they receive
the final product from the vendor.
```

Software Tools

A software tool is a programmatic software used to create, maintain, or otherwise support other programs and applications. Some of the commonly used software tools in HCI are as follows:

Specification Methods: The methods used to specify the GUI. Even though these are lengthy and ambiguous methods, they are easy to understand.

Grammars: Written Instructions or Expressions that a program would understand. They provide confirmations for completeness and correctness.

Transition Diagram: Set of nodes and links that can be displayed in text, link frequency, state diagram, etc. They are difficult in evaluating usability, visibility, modularity and synchronization.

Statecharts: Chart methods developed for simultaneous user activities and external actions. They provide link-specification with interface building tools.

Interface Building Tools: Design methods that help in designing command languages, data-entry structures, and widgets.

Interface Mockup Tools: Tools to develop a quick sketch of GUI. E.g., Microsoft Visio, Visual Studio .Net, etc.

Software Engineering Tools: Extensive programming tools to provide user interface management system.

Evaluation Tools: Tools to evaluate the correctness and completeness of programs.

HCI and Software Engineering

Software engineering is the study of designing, development and preservation of software. It comes in contact with HCI to make the man and machine interaction more vibrant and interactive.

Let us see the following model in software engineering for interactive designing.

The Waterfall Method
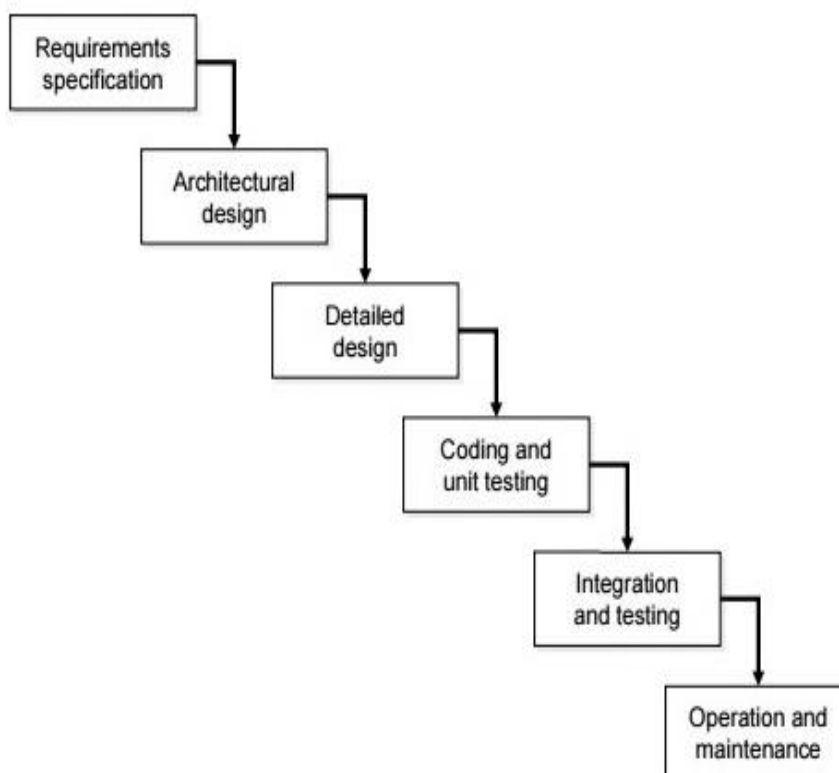
Interactive System Design
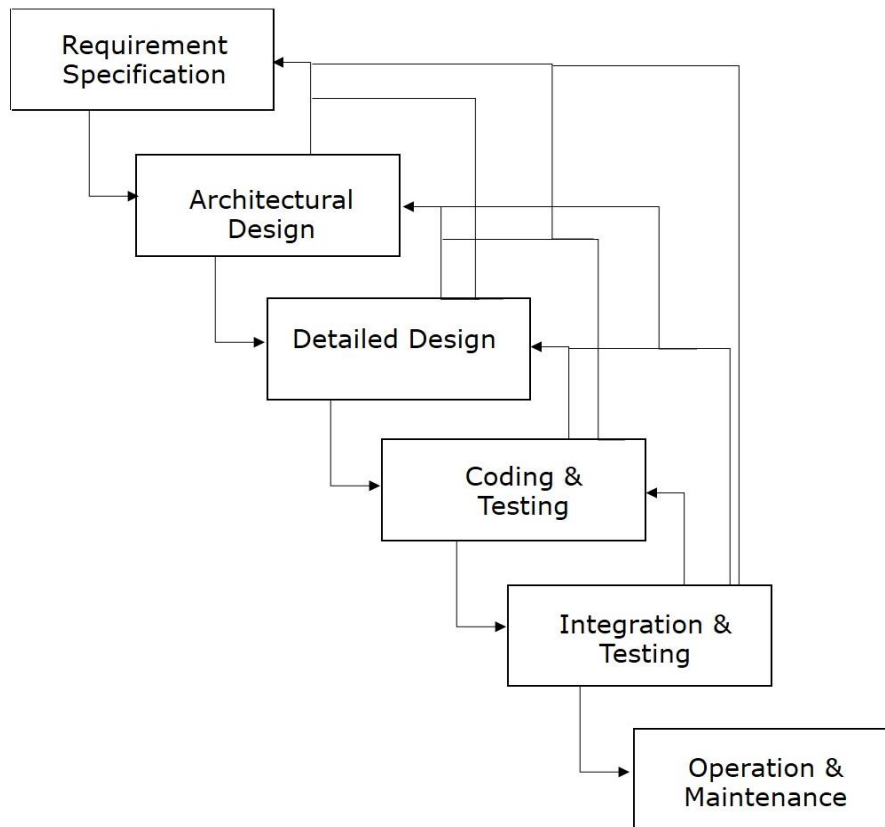
The uni-directional movement of the waterfall model of Software Engineering shows that every phase depends on the preceding phase and not vice-versa. However, this model is not suitable for the interactive system design.

The interactive system design shows that every phase depends on each other to serve the purpose of designing and product creation. It is a continuous process as there is so much to know and users keep changing all the time. An interactive system designer should recognize this diversity.

Prototyping

Prototyping is another type of software engineering models that can have a complete range of functionalities of the projected system.

In HCI, prototyping is a trial and partial design that helps users in testing design ideas without executing a complete system.

Example of a prototype can be **Sketches**. Sketches of interactive design can later be produced into graphical interface. See the following diagram.
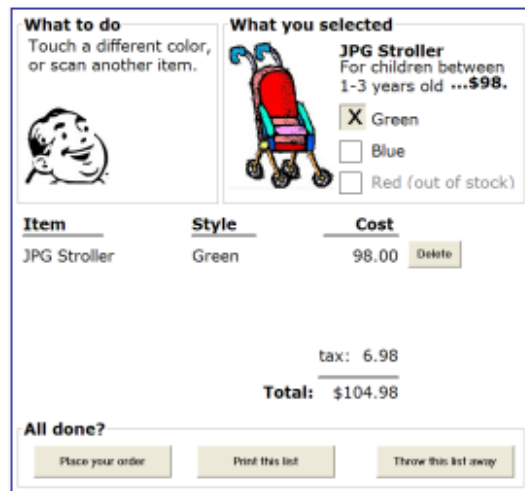
Interface of a proposed system

A sketch of the interface

The above diagram can be considered as a **Low Fidelity Prototype** as it uses manual procedures like sketching in a paper.

A **Medium Fidelity Prototype** involves some but not all procedures of the system. E.g., first screen of a GUI.

Finally, a **Hi Fidelity Prototype** simulates all the functionalities of the system in a design. This prototype requires, time, money and work force.

User Centered Design (UCD)

The process of collecting feedback from users to improve the design is known as *user centered design* or UCD.

UCD Drawbacks

- Passive user involvement.

- User's perception about the new interface may be inappropriate.

- Designers may ask incorrect questions to users.

Interactive System Design Life Cycle (ISLC)

The stages in the following diagram are repeated until the solution is reached.

Diagram
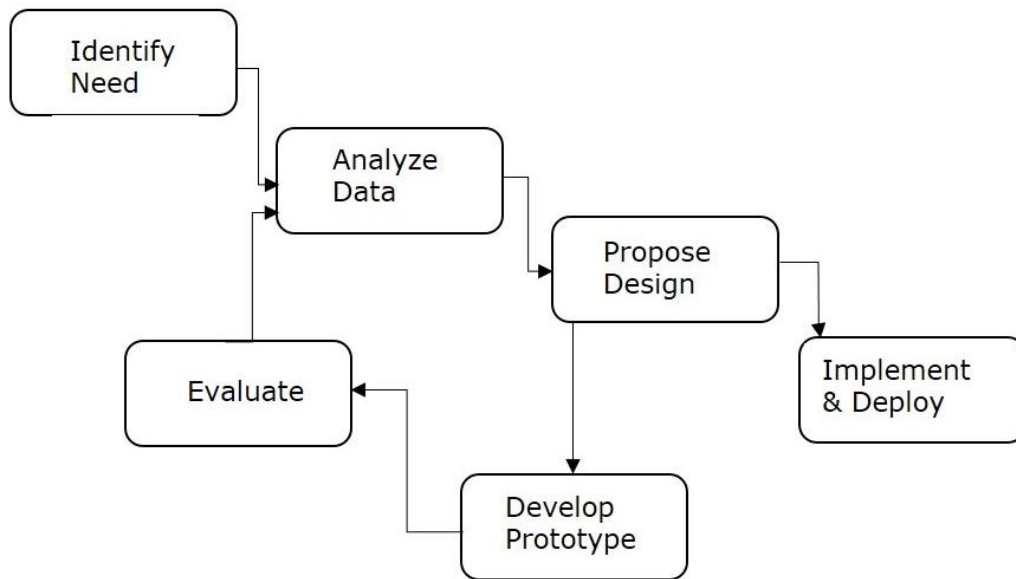
GUI Design & Aesthetics

Graphic User Interface (GUI) is the interface from where a user can operate programs, applications or devices in a computer system. This is where the icons, menus, widgets, labels exist for the users to access.

It is significant that everything in the GUI is arranged in a way that is recognizable and pleasing to the eye, which shows the aesthetic sense of the GUI designer. GUI aesthetics provides a character and identity to any product.

HCI in Indian Industries

For the past couple of years, majority IT companies in India are hiring designers for HCI related activities. Even multi-national companies started hiring for HCI from India as Indian designers have proven their capabilities in architectural, visual and interaction designs. Thus, Indian HCI designers are not only making a mark in the country, but also abroad.

The profession has boomed in the last decade even when the usability has been there forever. And since new products are developed frequently, the durability prognosis also looks great.

As per an estimation made on usability specialists, there are mere 1,000 experts in India. The overall requirement is around 60,000. Out of all the designers working in the country, HCI designers count for approximately 2.77%.

HCI Analogy

Let us take a known analogy that can be understood by everyone. A film director is a person who with his/her experience can work on script writing, acting, editing, and cinematography. He/She can be considered as the only person accountable for all the creative phases of the film.

Similarly, HCI can be considered as the film director whose job is part creative and part technical. An HCI designer have substantial understanding of all areas of designing. The following diagram depicts the analogy:
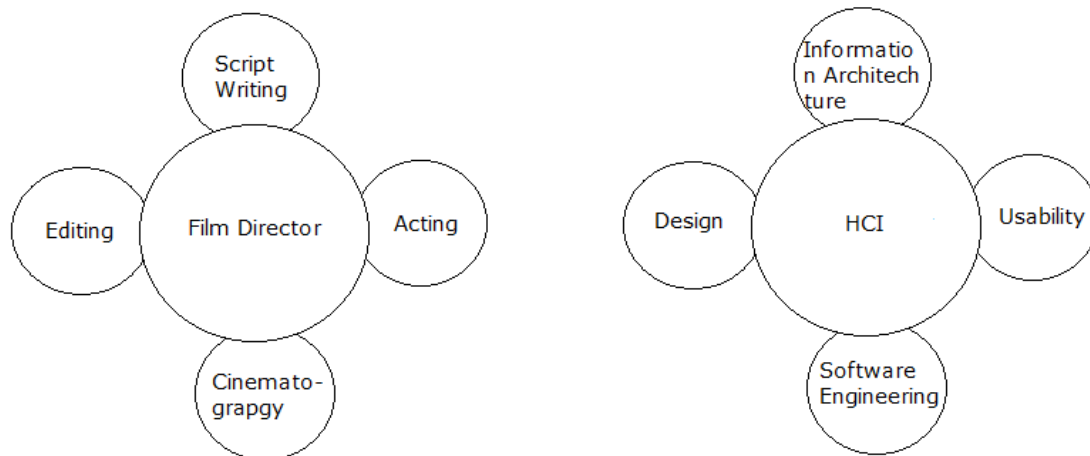
4. INTERACTIVE DEVICES

Several interactive devices are used for the human computer interaction. Some of them are known tools and some are recently developed or are a concept to be developed in the future. In this chapter, we will discuss on some new and old interactive devices.

Touch Screen

The touch screen concept was prophesized decades ago, however the platform was acquired recently. Today there are many devices that use touch screen. After vigilant selection of these devices, developers customize their touch screen experiences.

The cheapest and relatively easy way of manufacturing touch screens are the ones using electrodes and a voltage association. Other than the hardware differences, software alone can bring major differences from one touch device to another, even when the same hardware is used.

Along with the innovative designs and new hardware and software, touch screens are likely to grow in a big way in the future. A further development can be made by making a sync between the touch and other devices.

In HCI, touch screen can be considered as a new interactive device.

Gesture Recognition

Gesture recognition is a subject in language technology that has the objective of understanding human movement via mathematical procedures. Hand gesture recognition is currently the field of focus. This technology is future based.

This new technology magnitudes an advanced association between human and computer where no mechanical devices are used. This new interactive device might terminate the old devices like keyboards and is also heavy on new devices like touch screens.

Speech Recognition

The technology of transcribing spoken phrases into written text is Speech Recognition. Such technologies can be used in advanced control of many devices such as switching on and off the electrical appliances. Only certain commands are required to be recognized for a complete transcription. However, this cannot be beneficial for big vocabularies.

This HCI device help the user in hands free movement and keep the instruction based technology up to date with the users.

Keyboard

A keyboard can be considered as a primitive device known to all of us today. Keyboard uses an organization of keys/buttons that serves as a mechanical device for a computer. Each key in a keyboard corresponds to a single written symbol or character.

This is the most effective and ancient interactive device between man and machine that has given ideas to develop many more interactive devices as well as has made advancements in itself such as soft screen keyboards for computers and mobile phones.

Response Time

Response time is the time taken by a device to respond to a request. The request can be anything from a database query to loading a web page. The response time is the sum of the service time and wait time. Transmission time becomes a part of the response time when the response has to travel over a network.

In modern HCI devices, there are several applications installed and most of them function simultaneously or as per the user's usage. This makes a busier response time. All of that increase in the response time is caused by increase in the wait time. The wait time is due to the running of the requests and the queue of requests following it.

So, it is significant that the response time of a device is faster for which advanced processors are used in modern devices.

5. DESIGN PROCESS AND TASK ANALYSIS

HCI Design

HCI design is considered as a problem solving process that has components like planned usage, target area, resources, cost, and viability. It decides on the requirement of product similarities to balance trade-offs.

The following points are the four basic activities of interaction design:

1. *Identifying requirements*

2. *Building alternative designs*

3. *Developing interactive versions of the designs*

4. *Evaluating designs*

Three principles for user-centered approach are:

1. *Early focus on users and tasks*

2. *Empirical Measurement*

3. *Iterative Design*

Design Methodologies

Various methodologies have materialized since the inception that outline the techniques for human–computer interaction. Following are few design methodologies:

- **Activity Theory**: This is an HCI method that describes the framework where the human-computer interactions take place. Activity theory provides reasoning, analytical tools and interaction designs.

- **User-Centered Design**: It provides users the center-stage in designing where they get the opportunity to work with designers and technical practitioners.

- **Principles of User Interface Design**: *Tolerance, simplicity, visibility, affordance, consistency, structure* and *feedback* are the seven principles used in interface designing.

- **Value Sensitive Design**: This method is used for developing technology and includes three types of studies: *conceptual, empirical* and *technical*.

  o *Conceptual* investigations works towards understanding the values of the investors who use technology.

tutorialspoint
SIMPLYEASYLEARNING

- o *Empirical* investigations are qualitative or quantitative design research studies that shows the designer's understanding of the users' values.

- o *Technical* investigations contain the use of technologies and designs in the conceptual and empirical investigations.

Participatory Design

Participatory design process involves all stakeholders in the design process, so that the end result meets the needs they are desiring. This design is used in various areas such as software design, architecture, landscape architecture, product design, sustainability, graphic design, planning, urban design, and even medicine.

Participatory design is not a style, but focus on processes and procedures of designing. It is seen as a way of removing design accountability and origination by designers.

Task Analysis

Task Analysis plays an important part in User Requirements Analysis.
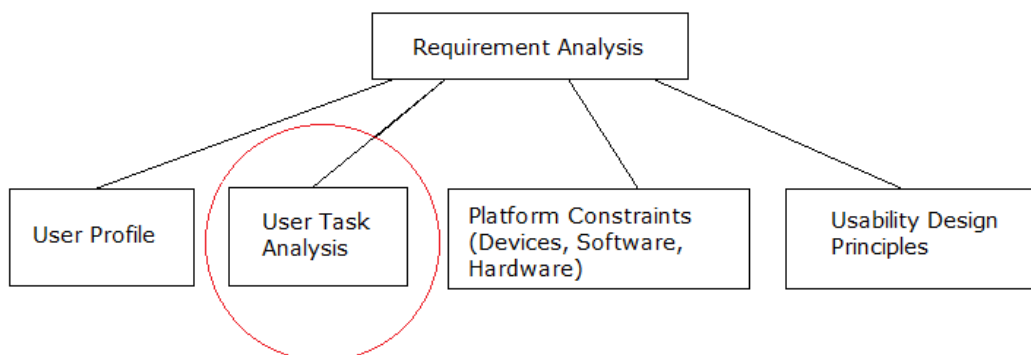
Task analysis is the procedure to learn the users and abstract frameworks, the patterns used in workflows, and the chronological implementation of interaction with the GUI. It analyzes the ways in which the user partitions the tasks and sequence them.

What is a TASK?

Human actions that contributes to a useful objective, aiming at the system, is a task. Task analysis defines performance of users, not computers.

Hierarchical Task Analysis

Hierarchical Task Analysis is the procedure of disintegrating tasks into subtasks that could be analyzed using the logical sequence for execution. This would help in achieving the goal in the best possible way.

> "A hierarchy is an organization of elements that, according to prerequisite relationships, describes the path of experiences a learner must take to achieve any single behavior that appears higher in the hierarchy. (Seels & Glasgow, 1990, p. 94)".

Techniques for Analysis

- Task decomposition: Splitting tasks into sub-tasks and in sequence.

- Knowledge-based techniques: Any instructions that users need to know.

'User' is always the beginning point for a task.

- Ethnography: Observation of users' behavior in the use context.

- Protocol analysis: Observation and documentation of actions of the user. This is achieved by authenticating the user's thinking. The user is made to think aloud so that the user's mental logic can be understood.

Engineering Task Models

Unlike Hierarchical Task Analysis, Engineering Task Models can be specified formally and are more useful.

Characteristics of Engineering Task Models

- Engineering task models have flexible notations, which describes the possible activities clearly.

- They have organized approaches to support the requirement, analysis, and use of task models in the design.

- They support the recycle of in-condition design solutions to problems that happen throughout applications.

- Finally, they let the automatic tools accessible to support the different phases of the design cycle.

ConcurTaskTree (CTT)

CTT is an engineering methodology used for modeling a task and consists of tasks and operators. Operators in CTT are used to portray chronological associations between tasks. Following are the key features of a CTT:

- Focus on actions that users wish to accomplish.

- Hierarchical structure.

- Graphical syntax.

- Rich set of sequential operators.

Dialog

A dialog is the construction of interaction between two or more beings or systems. In HCI, a dialog is studied at three levels:

- *Lexical*: Shape of icons, actual keys pressed, etc., are dealt at this level.

- *Syntactic*: The order of inputs and outputs in an interaction are described at this level.

- *Semantic*: At this level, the effect of dialog on the internal application/data is taken care of.

Dialog Representation

To represent dialogs, we need formal techniques that serves two purposes:

- It helps in understanding the proposed design in a better way.

- It helps in analyzing dialogs to identify usability issues. E.g., Questions such as "does the design actually support undo?" can be answered.

Introduction to Formalism

There are many formalism techniques that we can use to signify dialogs. In this chapter, we will discuss on three of these formalism techniques, which are:

- The state transition networks (STN)

- The state charts

- The classical Petri nets

State Transition Network (STN)

STNs are the most spontaneous, which knows that a dialog fundamentally denotes to a progression from one state of the system to the next.

The syntax of an STN consists of the following two entities:

- Circles: A circle refers to a state of the system, which is branded by giving a name to the state.

- Arcs: The circles are connected with arcs that refers to the action/event resulting in the transition from the state where the arc initiates, to the state where it ends.

STN Diagram
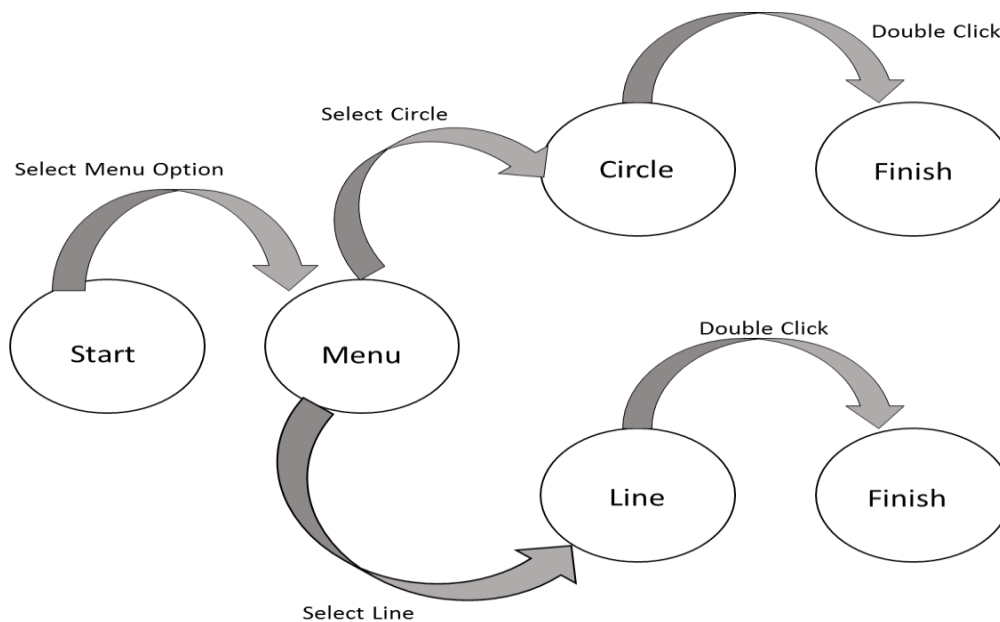
StateCharts

StateCharts represent complex reactive systems that extends Finite State Machines (FSM), handle concurrency, and adds memory to FSM. It also simplifies complex system representations. StateCharts has the following states:

Active state: The present state of the underlying FSM.

Basic states: These are individual states and are not composed of other states.

Super states: These states are composed of other states.

Illustration

For each basic state b, the super state containing b is called the ancestor state. A super state is called OR super state if exactly one of its sub states is active, whenever it is active.
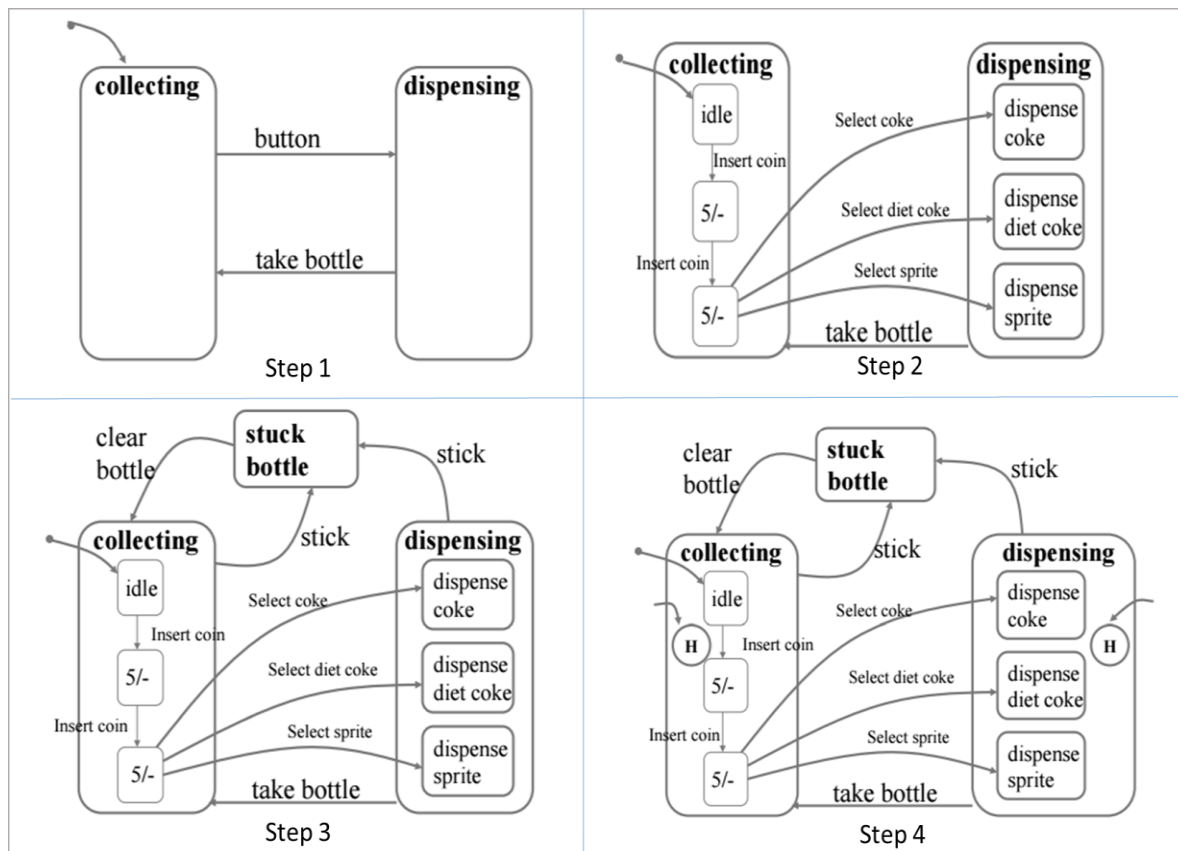
Let us see the StateChart Construction of a machine that dispense bottles on inserting coins.

The above diagram explains the entire procedure of a bottle dispensing machine. On pressing the button after inserting coin, the machine will toggle between bottle filling and dispensing modes. When a required request bottle is available, it dispense the bottle. In the background, another procedure runs where any stuck bottle will be cleared. The 'H' symbol in Step 4, indicates that a procedure is added to History for future access.

Petri Nets

Petri Net is a simple model of active behavior, which has four behavior elements such as - places, transitions, arcs and tokens. Petri Nets provide a graphical explanation for easy understanding.

- *Place*: This element is used to symbolize passive elements of the reactive system. A place is represented by a circle.

- *Transition*: This element is used to symbolize active elements of the reactive system. Transitions are represented by squares/rectangles.

- *Arc*: This element is used to represent causal relations. Arc is represented by arrows.

- *Token*: This element is subject to change. Tokens are represented by small filled circles.

Visual Thinking

Visual materials has assisted in the communication process since ages in form of paintings, sketches, maps, diagrams, photographs, etc. In today's world, with the invention of technology and its further growth, new potentials are offered for visual information such as thinking and reasoning. As per studies, the command of visual thinking in human-computer interaction (HCI) design is still not discovered completely. So, let us learn the theories that support visual thinking in sense-making activities in HCI design.

An initial terminology for talking about visual thinking was discovered that included concepts such as visual immediacy, visual impetus, visual impedance, and visual metaphors, analogies and associations, in the context of information design for the web.

As such, this design process became well suited as a logical and collaborative method during the design process. Let us discuss in brief the concepts individually.

Visual Immediacy

It is a reasoning process that helps in understanding of information in the visual representation. The term is chosen to highlight its time related quality, which also serves as an indicator of how well the reasoning has been facilitated by the design.

Visual Impetus

Visual impetus is defined as a stimulus that aims at the increase in engagement in the contextual aspects of the representation.

Visual Impedance

It is perceived as the opposite of visual immediacy as it is a hindrance in the design of the representation. In relation to reasoning, impedance can be expressed as a slower cognition.

Visual Metaphors, Association, Analogy, Abduction and Blending

- When a visual demonstration is used to understand an idea in terms of another familiar idea it is called a visual metaphor.

- Visual analogy and conceptual blending are similar to metaphors. Analogy can be defined as an implication from one particular to another. Conceptual

blending can be defined as combination of elements and vital relations from varied situations.

The HCI design can be highly benefited with the use of above mentioned concepts. The concepts are pragmatic in supporting the use of visual procedures in HCI, as well as in the design processes.

Direct Manipulation Programming

Direct manipulation has been acclaimed as a good form of interface design, and are well received by users. Such processes use many source to get the input and finally convert them into an output as desired by the user using inbuilt tools and programs.

"Directness" has been considered as a phenomena that contributes majorly to the manipulation programming. It has the following two aspects.

- Distance

- Direct Engagement

Distance

Distance is an interface that decides the gulfs between a user's goal and the level of explanation delivered by the systems, with which the user deals. These are referred to as the *Gulf of Execution* and the *Gulf of Evaluation*.

The Gulf of Execution

The Gulf of Execution defines the gap/gulf between a user's goal and the device to implement that goal. One of the principal objective of Usability is to diminish this gap by removing barriers and follow steps to minimize the user's distraction from the intended task that would prevent the flow of the work.

The Gulf of Evaluation

The Gulf of Evaluation is the representation of expectations that the user has interpreted from the system in a design. As per Donald Norman, *The gulf is small when the system provides information about its state in a form that is easy to get, is easy to interpret, and matches the way the person thinks of the system*.

Direct Engagement

It is described as a programming where the design directly takes care of the controls of the objects presented by the user and makes a system less difficult to use.

The scrutiny of the execution and evaluation process illuminates the efforts in using a system. It also gives the ways to minimize the mental effort required to use a system.

Problems with Direct Manipulation

- Even though the immediacy of response and the conversion of objectives to actions has made some tasks easy, all tasks should not be done easily. For example, a repetitive operation is probably best done via a script and not through immediacy.

- Direct manipulation interfaces finds it hard to manage variables, or illustration of discrete elements from a class of elements.

- Direct manipulation interfaces may not be accurate as the dependency is on the user rather than on the system.

- An important problem with direct manipulation interfaces is that it directly supports the techniques, the user thinks.

Item Presentation Sequence

In HCI, the presentation sequence can be planned according to the task or application requirements. The natural sequence of items in the menu should be taken care of. Main factors in presentation sequence are:

- Time

- Numeric ordering

- Physical properties

A designer must select one of the following prospects when there are no task-related arrangements:

- Alphabetic sequence of terms

- Grouping of related items

- Most frequently used items first

- Most important items first

Menu Layout

- Menus should be organized using task semantics.

- Broad-shallow should be preferred to narrow-deep.

- Positions should be shown by graphics, numbers or titles.

- Subtrees should use items as titles.

- Items should be grouped meaningfully.

- Items should be sequenced meaningfully.

- Brief items should be used.

- Consistent grammar, layout and technology should be used.

- Type ahead, jump ahead, or other shortcuts should be allowed.

- Jumps to previous and main menu should be allowed.

- Online help should be considered.

Guidelines for consistency should be defined for the following components:

- Titles

- Item placement

- Instructions

- Error messages

- Status reports

Form Fill-in Dialog Boxes

Appropriate for multiple entry of data fields:

- Complete information should be visible to the user.

- The display should resemble familiar paper forms.

- Some instructions should be given for different types of entries.

Users must be familiar with:

- Keyboards

- Use of TAB key or mouse to move the cursor

- Error correction methods

- Field-label meanings

- Permissible field contents

- Use of the ENTER and/or RETURN key.

Form Fill-in Design Guidelines:

- Title should be meaningful.

- Instructions should be comprehensible.

- Fields should be logically grouped and sequenced.

- The form should be visually appealing.

- Familiar field labels should be provided.

- Consistent terminology and abbreviations should be used.

- Convenient cursor movement should be available.

- Error correction for individual characters and entire field's facility should be present.

- Error prevention.

- Error messages for unacceptable values should be populated.

- Optional fields should be clearly marked.

- Explanatory messages for fields should be available.

- Completion signal should populate.

7. INFORMATION SEARCH AND VISUALIZATION

Database Query

A database query is the principal mechanism to retrieve information from a database. It consists of predefined format of database questions. Many database management systems use the Structured Query Language (SQL) standard query format.

Example

```
SELECT DOCUMENT#

FROM JOURNAL-DB

WHERE (DATE >= 2004 AND DATE <= 2008)

AND (LANGUAGE = ENGLISH OR FRENCH)

AND (PUBLISHER = ASIST OR HFES OR ACM)
```

Users perform better and have better contentment when they can view and control the search. The database query has thus provided substantial amount of help in the human computer interface.

The following points are the five-phase frameworks that clarifies user interfaces for textual search:

1. *Formulation: expressing the search*

2. *Initiation of action: launching the search*

3. *Review of results: reading messages and outcomes*

4. *Refinement: formulating the next step*

5. *Use: compiling or disseminating insight*

Multimedia Document Searches

Following are the major multimedia document search categories.

Image Search

Preforming an image search in common search engines is not an easy thing to do. However there are sites where image search can be done by entering the image of your choice. Mostly, simple drawing tools are used to build templates to search with. For complex searches such as fingerprint matching, special

softwares are developed where the user can search the machine for the predefined data of distinct features.

Map Search

Map search is another form of multimedia search where the online maps are retrieved through mobile devices and search engines. Though a structured database solution is required for complex searches such as searches with longitude/latitude. With the advanced database options, we can retrieve maps for every possible aspect such as cities, states, countries, world maps, weather sheets, directions, etc.

Design/Diagram Searches

Some design packages support the search of designs or diagrams as well. E.g., diagrams, blueprints, newspapers, etc.

Sound Search

Sound search can also be done easily through audio search of the database. Though user should clearly speak the words or phrases for search.

Video Search

New projects such as Infomedia helps in retrieving video searches. They provide an overview of the videos or segmentations of frames from the video.

Animation Search

The frequency of animation search has increased with the popularity of Flash. Now it is possible to search for specific animations such as a moving boat.

Information Visualization

Information visualization is the interactive visual illustrations of conceptual data that strengthen human understanding. It has emerged from the research in human-computer interaction and is applied as a critical component in varied fields. It allows users to see, discover, and understand huge amounts of information at once.

Information visualization is also an assumption structure, which is typically followed by formal examination such as statistical hypothesis testing.

Advanced Filtering

Following are the advanced filtering procedures:

- *Filtering with complex Boolean queries*

- *Automatic filtering*

- *Dynamic queries*

- *Faceted metadata search*

- *Query by example*

- *Implicit search*

- *Collaborative filtering*

- *Multilingual searches*

- *Visual field specification*

Hypertext and Hypermedia

Hypertext can be defined as the text that has references to hyperlinks with immediate access. Any text that provides a reference to another text can be understood as two nodes of information with the reference forming the link. In hypertext, all the links are active and when clicked, opens something new.

Hypermedia on the other hand, is an information medium that holds different types of media, such as, video, CD, and so forth, as well as hyperlinks.

Hence, both hypertext and hypermedia refers to a system of linked information. A text may refer to links, which may also have visuals or media. So hypertext can be used as a generic term to denote a document, which may in fact be distributed across several media.

Object Action Interface Model for Website Design

Object Action Interface (OAI), can be considered as the next step of the Graphical User Interface (GUI). This model focusses on the priority of the object over the actions.

OAI Model

The OAI model allows the user to perform action on the object. First the object is selected and then the action is performed on the object. Finally, the outcome is shown to the user. In this model, the user does not have to worry about the complexity of any syntactical actions.

The object–action model provides an advantage to the user as they gain a sense of control due to the direct involvement in the design process. The computer serves as a medium to signify different tools.

8. OBJECT ORIENTED PROGRAMMING

Object Oriented Programming Paradigm (OOPP)

The Object oriented programming paradigm plays an important role in human computer interface. It has different components that takes real world objects and performs actions on them, making live interactions between man and the machine. Following are the components of OOPP:

- This paradigm describes a real-life system where interactions are among real objects.

- It models applications as a group of related objects that interact with each other.

- The programming entity is modeled as a class that signifies the collection of related real world objects.

- Programming starts with the concept of real world objects and classes.

- Application is divided into numerous packages.

- A package is a collection of classes.

- A class is an encapsulated group of similar real world objects.

Objects

Real-world objects share two characteristics: They all have state and behavior. Let us see the following pictorial example to understand Objects.

State	Name Color Breed Hungry
Behavior	Barking Fetching Wagging Tail

In the above diagram, the object 'Dog' has both state and behavior.

An object stores its information in attributes and discloses its behavior through methods. Let us now discuss in brief the different components of object oriented programming.

Data Encapsulation

Hiding the implementation details of the class from the user through an object's methods is known as data encapsulation. In object oriented programming, it binds the code and the data together and keeps them safe from outside interference.

Public Interface

The point where the software entities interact with each other either in a single computer or in a network is known as pubic interface. This help in data security. Other objects can change the state of an object in an interaction by using only those methods that are exposed to the outer world through a public interface.

Class

A class is a group of objects that has mutual methods. It can be considered as the blueprint using which objects are created.

Classes being passive do not communicate with each other but are used to instantiate objects that interact with each other.
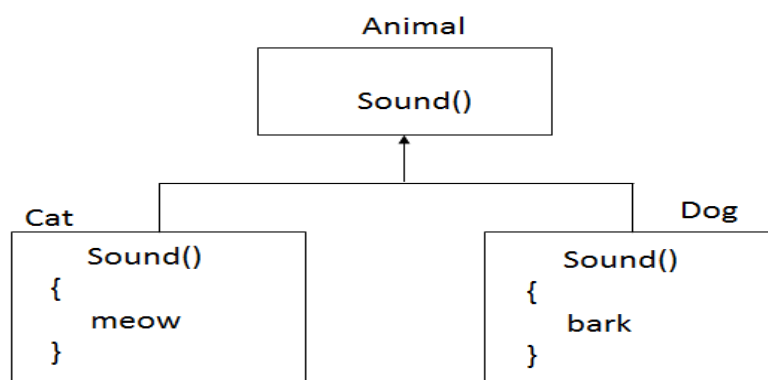
Inheritance

Inheritance as in general terms is the process of acquiring properties. In OOP one object inherit the properties of another object.

Polymorphism

Polymorphism is the process of using same method name by multiple classes and redefines methods for the derived classes.
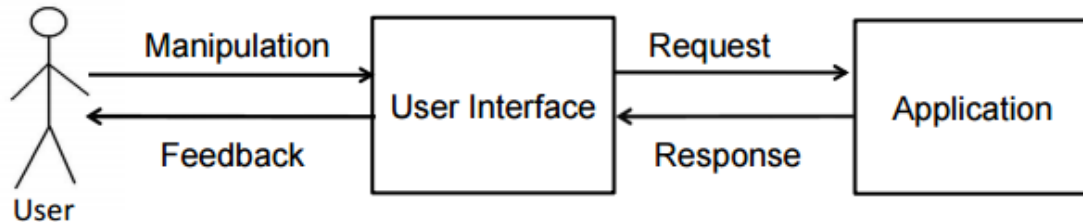
Example

Object Oriented Modeling of User Interface Design

Object oriented interface unites users with the real world manipulating software objects for designing purpose. Let us see the diagram.

Interface design strive to make successful accomplishment of user's goals with the help of interaction tasks and manipulation.

While creating the OOM for interface design, first of all analysis of user requirements is done. The design specifies the structure and components required for each dialogue. After that, interfaces are developed and tested against the Use Case. Example - Personal banking application.

The sequence of processes documented for every Use Case are then analyzed for key objects. This results into an object model. Key objects are called analysis objects and any diagram showing relationships between these objects is called object diagram.

9. SUMMARY

We have now learnt the basic aspects of human computer interface in this tutorial. From here onwards, we can refer complete reference books and guides that will give in-depth knowledge on programming aspects of this subject. We hope that this tutorial has helped you in understanding the topic and you have gained interest in this subject.

We hope to see the birth of new professions in HCI designing in the future that would take help from the current designing practices. The HCI designer of tomorrow would definitely adopt many skills that are the domain of specialists today. And for the current practice of specialists, we wish them to evolve, as others have done in the past.

In the future, we hope to reinvent the software development tools, making programming useful to people's work and hobbies. We also hope to understand the software development as a collaborative work and study the impact of software on society.