HCI QP Sessional Answers

1.a) Explain in detail about human input and output channels.

Ans.1.a)

A person's interaction with the outside world occurs through information being received and sent: input and output. In an interaction with a computer the user receives information that is output by the computer and responds by providing input to the computer–the user's output becomes the computer's input and vice versa.

There are five major senses: sight, hearing, touch, taste and smell. Of these, the first three are the most important to HCI. Taste and smell do not currently play a significant role in HCI, and it is not clear whether they could be exploited at all in general computer systems, although they could have a role to play in more specialized systems (smells to give warning of malfunction, for example) or in augmented reality systems. vision, hearing and touch are central.

1. Vision

Human vision is a highly complex activity with a range of physical and perceptual limitations, we can roughly divide visual perception into two stages: the physical reception of the stimulus from the outside world, and the processing and interpretation of that stimulus.

2. Hearing

The sense of hearing is often considered secondary to sight, but we tend to underestimate the amount of information that we receive through our ears. hearing begins with vibrations in the air or sound waves. The ear receives these vibrations and transmits them, through various stages, to the auditory nerves. The ear comprises three sections, commonly known as the Outer ear, middle ear and inner ear.

3. Touch

Touch provides us with vital information about our environment. It tells us when we touch something hot or cold and can therefore act as a warning. It also provides us with feedback when we attempt to lift an object, for example. Consider the act of picking up a glass of water. If we could only see the glass and not feel when our hand made contact with it or feel its shape, the speed and accuracy of the action would be reduced. This is the experience of users of certain virtual reality games: they can see the computer-generated objects which they need to manipulate but they have no physical sensation of touching them.

4. Movement

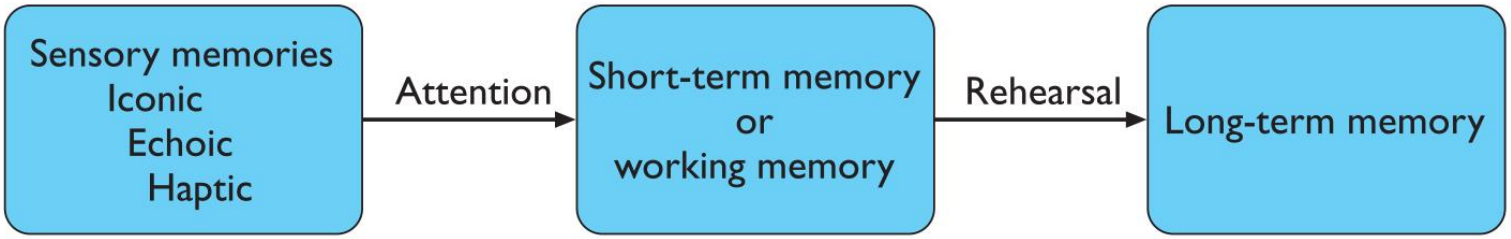
A simple action such as hitting a button in response to a question involves a number of processing stages. The stimulus (of the question) is received through the sensory receptors and transmitted to the brain. The question is processed, and a valid response generated. The brain then tells the appropriate muscles to respond. Each of these stages takes time, which can be roughly divided into reaction time and movement time.

1.b) What are the different types of memory in human brain?

Ans.1.b)

Our memory contains our knowledge of actions or procedures. It allows us to repeat actions, to use language, and to use new information received via our senses. It also gives us our sense of identity, by preserving information from our past experiences.

Memory is the second part of our model of the human as an information-processing system. Memory is associated with each level of processing. Bearing this in mind, we will consider the way in which memory is structured and the activities that take place within the system. It is generally agreed that there are three types of memory or memory function: sensory buffers, short-term memory or working memory, and long-term memory.



A model of the structure of memory

1. Sensory memory

The sensory memories act as buffers for stimuli received through the senses. A sensory memory exists for each sensory channel: iconic memory for visual stimuli, echoic memory for aural stimuli and haptic memory for touch. These memories are constantly overwritten by new information coming in on these channels.

2. Short-term memory

Short-term memory or working memory acts as a "scratchpad" for temporary recall of information. It is used to store information which is only required fleetingly. Short-term memory can be accessed rapidly, in the order of 70 ms. It also decays rapidly, meaning that information can only be held there temporarily, in the order of 200 ms. Short-term memory also has a limited. capacity..

3. Long-term memory

If short-term memory is our working memory or 'scratchpad, long-term memory is our main resource. Here we store factual information, experiential knowledge, procedural rules of behavior – in fact, everything that we _know_. It differs from short-term memory in a number of significant ways. First, it has a huge, if not unlimited, capacity. Secondly, it has a relatively slow access time of approximately a tenth of a second. Thirdly, forgetting occurs more slowly in long-term memory, if at all.

2.a) Define Problem Solving & list the theories involved in problem solving.

Ans.2.a)

Problem solving

Human problem solving is characterized by the ability to adapt the information we have to deal with new situations often solutions seem to be original and creative. There are a number of different views of how people solve problems. The Gestalt view that problem solving involves both reuse of knowledge and insight. This has been largely superseded but the questions it was trying to address remain and its influence can be seen in later research. In the 1970s by Newell and Simon, was the problem space theory, which takes the view that the mind is a limited information processor.

The theories involved in Problem Solving

1. **Gestalt theory**

   Gestalt psychologists were answering the claim, made by behaviorists, that problem solving is a matter of reproducing known responses or trial and error. This explanation was considered by the Gestalt school to be insufficient to account for human problem-solving behavior. Instead, they claimed, problem solving is both productive and reproductive. Reproductive problem solving draws on previous experience as the behaviorists claimed, but productive problem solving involves insight and restructuring of the problem. Indeed, reproductive problem solving could be a hindrance to finding a solution, since a person may fixate on the known aspects of the problem and so be unable to see novel interpretations that might lead to a solution. Gestalt psychologists backed up their claims with experimental evidence.

2. **Problem space theory**

   Newell and Simon proposed that problem solving centers on the problem space. The problem space comprises problem states, and problem solving involves generating these states using legal state transition operators. The problem has an initial state, and a goal state and people use the operators to move from the former to the latter. Such problem spaces maybe huge, and so heuristics are employed to select appropriate operators to reach the goal. One such heuristic is means–ends analysis. In means–ends analysis the initial state is compared with the goal state and an operator chosen to reduce the difference between the two. Newell and Simon's theory, and their General Problem Solver model, which is based on it, have largely been applied to problem solving in well-defined domains, for example solving puzzles. These problems may be unfamiliar but the knowledge that is required to solve the mis present in the statement of the problem and the expected solution is clear. In real-world problems finding the knowledge required to solve the problem may be part of the problem, or specifying the goal may be difficult.

2.b) Explain in detail about models of interaction.

Ans.2.b)

Models of Interaction:

Norman's execution-evaluation cycle and the interaction framework proposed by Abowd and Beale.

1. **Norman's execution-evaluation cycle: Norman's model describes the interaction between the user and the system in terms of a cycle consisting of seven stages:**
2. **Establishing the goal: The user forms a goal, which is their notion of what needs to be done.**
3. **Forming the intention: The user translates the goal into a more specific intention and determines the actions required to achieve it.**
4. **Specifying the action sequence: The user plans and specifies the sequence of actions needed to execute their intention.**
5. **Executing the action: The user performs the planned actions to interact with the system.**
6. **Perceiving the system state: After executing the actions, the user perceives the new state of the system.**
7. **Interpreting the system state: The user interprets the system state based on their expectations and goals.**
8. **Evaluating the system state: The user evaluates whether the system state aligns with their goals and intentions. If not, they may need to formulate a new goal and repeat the cycle.**

This model emphasizes the user's perspective and their interaction with the system to achieve their goals. It highlights the importance of feedback and evaluation in determining the success of the interaction.

1. **The interaction framework by Abowd and Beale: The interaction framework extends Norman's model by explicitly including the system and breaking the interaction into four main components: the System, the User, the Input, and the Output. The interface is formed by the combination of the Input and Output components.**

The framework identifies four steps in the interactive cycle, each corresponding to a translation between the components:

1. **User's goal and task formulation: The user starts the cycle by formulating a goal and a task to achieve that goal.**
2. **Translation from User to Input: The user's goal and task are translated into input language, which is understood by the system.**
3. **Translation from Input to System: The translated input language is used to stimulate the system and initiate the desired actions.**
4. **Translation from System to Output: The system's response is translated into output language, which is perceivable by the user.**

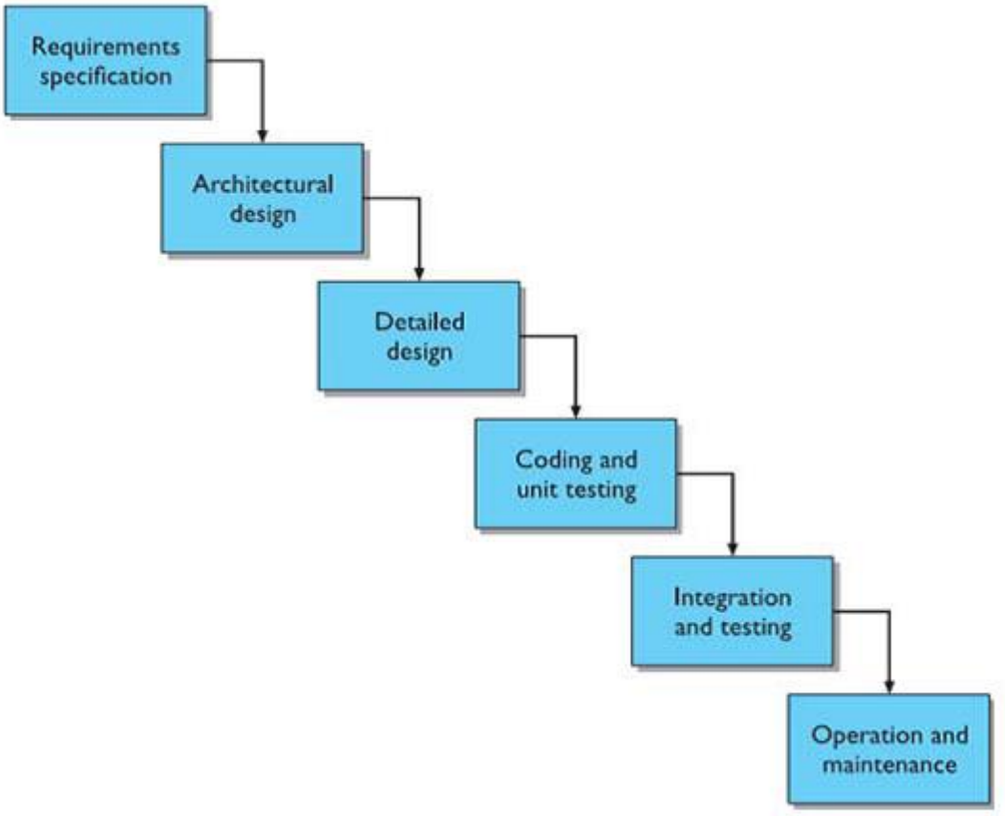
This framework emphasizes the importance of effective translation between the user, system, input, and output components for successful interaction. It recognizes the role of the interface in facilitating communication and understanding between the user and the system.

Overall, these models provide a conceptual framework for understanding and evaluating the interaction between users and systems. They help identify potential issues and improve the design of interactive systems to better support user tasks.

3.a) Explain the software life cycle model in HCI software process.

Ans.3.a)

The software life cycle refers to the stages involved in the development of a software product with a focus on human-computer interaction. The software life cycle process in HCI software development typically includes the following stages:



1. **Requirements Gathering:**

   This stage involves understanding the needs and requirements of the users and stakeholders. It includes activities such as user research, interviews, and observations to gather information about user preferences, goals, and tasks.
2. **Design:**

   In this stage, the design team creates the user interface and interaction design based on the gathered requirements. It includes creating wireframes, prototypes, and visual designs to define the structure, layout, and functionality of the software.
3. **Implementation:**

   The design is then translated into actual code during the implementation stage. The development team writes the software code, integrates different components, and ensures that the software functions as intended.
4. **Integration and Testing:**

   Once the individual components are implemented, they are integrated into a cohesive system. Testing is performed to ensure that the software behaves correctly, meets the requirements, and provides a satisfactory user experience. This stage may also involve acceptance testing with the customers to ensure that the system meets their needs.
5. **Maintenance:**

   After the software is released, it enters the maintenance stage. This stage involves ongoing support, bug fixes, updates, and enhancements based on user feedback and changing requirements. Maintenance continues until a new version of the software is released or the product is phased out.

It is important to note that the software life cycle in HCI software development is iterative and never complete. The design process involves continuous refinement and improvement based on user feedback and evaluation.

3.b) Explain in detail the interaction design process.

Ans.3.b)

The Interaction Design Process.

A system has been designed and built, and only when it proves unusable do they think to ask how to do it right! In other companies usability is seen as equivalent to testing – checking whether people can use it and fixing problems, rather than making sure they can from the beginning. In the best companies, however, usability is designed in from the start.

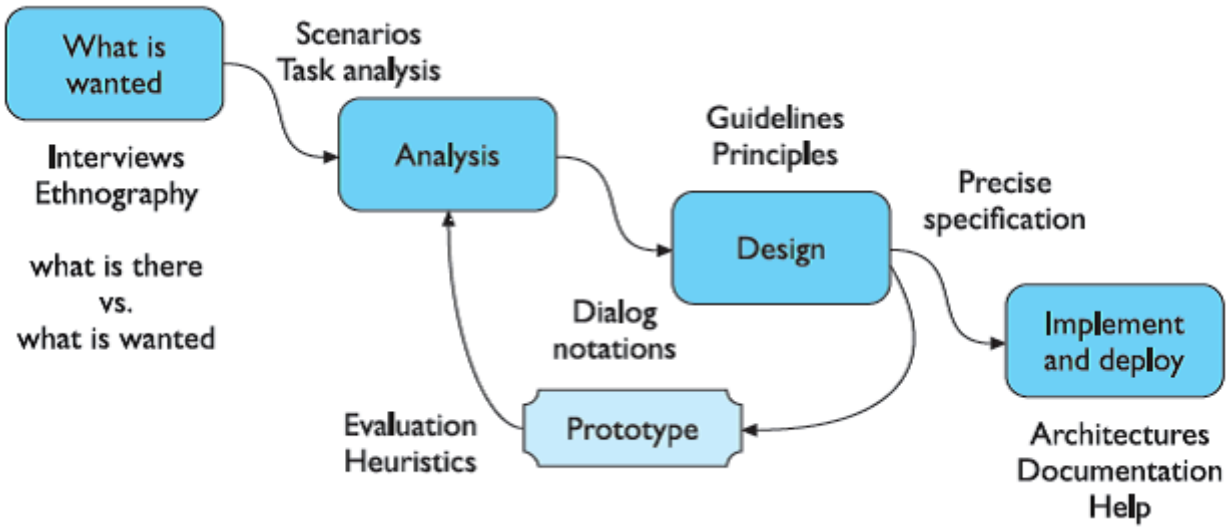


Figure: Interaction Design Process

1. Requirements :

What is wanted, the first stage is establishing what exactly is needed. As a precursor to this it is usually necessary to find out what is currently happening.

2. Analysis:

The results of observation and interview need to be ordered in some way to bring out key issues and communicate with later stages of design.

3. Design:

Well, this is all about design, but there is a central stage when you move from what you want, to how to do it. There are numerous rules, guidelines and design principles that can be used to help.

4. Iteration and prototyping:

Humans are complex and we cannot expect to get designs right first time. We therefore need to evaluate a design to see how well it is working and where there can be improvements.

5. Implementation and deployment:

Finally, when we are happy with our design, we need to create it and deploy it. This will involve writing code, perhaps making hardware, writing documentation and manuals – everything that goes into a real system that can be given to others.

4.a) Enumerate Norman's seven principles for transferring difficult task to simple one in design?
Ans.4.a)

Norman's Seven Principles for Transforming Difficult Tasks into Simple Ones

1. Use both knowledge in the world and knowledge in the head. People work better when the knowledge they need to do a task is available externally – either explicitly or through the constraints imposed by the environment. But experts also need to be able to internalize regular tasks to increase their efficiency. So, systems should provide the necessary knowledge within the environment and their operation should be transparent to support the user in building an appropriate mental model of what is going on.

2. Simplify the structure of tasks. Tasks need to be simple in order to avoid complex problem solving and excessive memory load. There are a number of ways to simplify the structure of tasks. One is to provide mental aids to help the user keep track of stages in a more complex task. Another is to use technology to provide the user with more information about the task and better feedback. A third approach is to automate the task or part of it, as long as this does not detract from the user's experience. The final approach to simplification is to change the nature of the task so that it becomes something simpler. In all of this, it is important not to take control away from the user.

3. Make things visible: Bridge the gulfs of execution and evaluation. The interface should make clear what the system can do and how this is achieved and should enable the user to seem clearly the effect of their actions on the system.

4. Get the mappings right. User intentions should map clearly onto system controls. User actions should map clearly onto system events. So, it should be clear what does what and by how much. Controls, sliders and dials should reflect the task – so a small movement has a small effect and a large movement a large effect.

5. Exploit the power of constraints, both natural and artificial. Constraints are things in the world that make it impossible to do anything but the correct action in the correct way. A simple example is a jigsaw puzzle, where the pieces only fit together in one way. Here the physical constraints of the design guide the user to complete the task.

6. Design for error. To err is human, so anticipate the errors the user could make and design recovery into the system.

7. When all else fails, standardize. If there are no natural mappings then arbitrary mappings should be standardized so that users only have to learn them once. It is this standardization principle that enables drivers to get into a new car and drive it with very little difficulty – key controls are standardized. Occasionally one might switch on the indicator lights instead of the windscreen wipers, but the critical controls (accelerator, brake, clutch, steering) are always the same.

4.b) Explain about the various factors distinguishing evaluation techniques.

Ans.4.b)

The various factors that distinguish evaluation techniques include:

1. **Purpose: Evaluation techniques can be used to assess the functionality and usability of a system, as well as to identify specific problems with the design.**
    a. **Formative Evaluation: The primary purpose is to identify and address usability issues during the design and development process. It helps designers make improvements and refinements to enhance the overall user experience.**
    b. **Summative Evaluation: The main goal is to assess the overall effectiveness and usability of the system after it has been developed. It provides insights into the system's performance and user satisfaction.**
2. **Methodology: Evaluation techniques can be conducted in a laboratory setting, in the field, or in collaboration with users. They can involve both qualitative and quantitative methods.**
    a. **Inspection Methods: These involve experts or evaluators examining the interface without direct user involvement. Examples include Heuristic Evaluation, Cognitive Walkthrough, and expert reviews. These methods are valuable for uncovering potential usability issues based on established principles and guidelines.**
    b. **User Testing: This involves actual users interacting with the system, providing direct feedback on usability, preferences, and overall user experience. Methods such as Usability Testing, A/B testing, and field studies fall into this category.**
3. **Goals: The goals of evaluation include assessing the extent and accessibility of the system's functionality, evaluating users' experience of the interaction, and identifying any specific problems with the system.**
    a. **Objective Metrics: Evaluation techniques with a focus on objective metrics aim to measure specific, quantifiable aspects of user performance. For example, measuring task completion time, error rates, and efficiency provides concrete data for assessing usability objectively.**
    b. **Subjective Feedback: Techniques emphasizing subjective feedback aim to gather users' opinions, preferences, and qualitative insights. This includes understanding users' satisfaction, perceptions, and overall experiences with the interface.**
4. **Focus: Evaluation techniques can assess both the design and implementation of a system, considering factors such as functionality, usability, learnability, satisfaction, and emotional response.**
    a. **Task Level Evaluation: Focuses on specific tasks or interactions within the system. Evaluators assess how well users can perform specific actions and achieve their goals. This type of evaluation is crucial for identifying and addressing usability issues at the task level.**
    b. **Holistic Evaluation: Examines the overall user experience and satisfaction with the entire system. It considers the system as a whole, including navigation, aesthetics, and the overall flow of interactions. Holistic evaluation provides a broader view of the user experience.**
5. **User Involvement: Evaluation techniques can involve user participation through empirical or experimental methods, observational methods, query techniques, or physiological monitoring.**
    a. **Inspection Methods (Expert Analysis): These methods involve usability experts or evaluators who assess the system without direct user involvement. Expert analysis relies on the knowledge and experience of evaluators to identify potential issues based on established heuristics, guidelines, or cognitive models.**
    b. **User Testing: Involves actual users interacting with the system, providing direct feedback and insights. Users' experiences and preferences are directly observed and recorded, offering a more authentic representation of real-world usage.**
6. **Expert Analysis: Evaluation techniques can also involve expert analysis, such as cognitive walkthroughs, heuristic evaluations, the use of models, and the review of previous work.**
    a. **Inspection Methods: Techniques like Heuristic Evaluation and Cognitive Walkthrough rely on the expertise of evaluators who analyze the system based on established principles. This expert analysis can uncover potential usability issues and provide actionable insights for improvement.**
    b. **User Testing: While users provide valuable feedback, expert analysis is still involved in interpreting the data, identifying patterns, and making informed recommendations for design enhancements.**

5.a) Explain in detail about cognitive model and its techniques.

Ans.5.a)

Cognitive complexity refers to the number of mental structures an individual uses, how abstract they are and how they interact to shape his discernment, or an individual difference variable linked with a wide range of communication skills and associated abilities.

Individuals with high cognitive complexity have the capacity to analyze a situation to discern various constituent elements and explore connections and possible relationships among the elements. These individuals think in a multidimensional way.

The assumption of the complexity theory is that the more an event can be differentiated, and parts considered in novel relationships, the more sophisticated the response and successful the solution. Whereas less complex individuals can be trained to understand a complicated set of detailed differentiations for a specific context, highly complex individuals are highly flexible in creating distinctions in new situations.

Individuals with high cognitive complexity are open to new information, attracted to other individuals of high complexity, highly flexibility, socially influential, problem solvers, strategic planners, highly creative, effective communicators, and generally good leaders.

Some techniques used in cognitive model include:

1. **Hierarchical models:**

These models represent a user's task and goal structure by breaking down complex tasks into smaller subgoals.

2. **Linguistic models:**

These models represent the grammar and language used in the interaction between the user and the system.

3. **Physical and device models:**

These models represent human motor skills and the interaction with physical devices, such as keyboards, mice, and touchscreens.

4. **Cognitive architectures:**

These underlying frameworks provide a structure for cognitive models and represent the cognitive processes involved in human information processing.

5. **Problem space models:**

These models represent a problem-solving process by defining a set of states and operations that can be performed on those states to achieve a desired goal.

5.b) Explain in detail about the various socio-technical models?

Ans.5.b)

The socio-technical systems view came about to counter this technology-centric position, by stressing that work systems were composed of both human and machine elements and that it was the interrelationship between these that should be central.

Socio-technical models for interactive systems are therefore concerned with technical, social, organizational and human aspects of design. They recognize the fact that technology is not developed in isolation but as part of a wider organizational environment. It is important to consider social and technical issues side by side so that human issues are not overruled by technical considerations.

The key focus of the socio-technical approach is to describe and document the impact of the introduction of a specific technology into an organization. Methods vary but most attempt to capture certain common elements:

- **The problem being addressed: there is a need to understand why the technology is being proposed and what problem it is intended to solve.**
- **The stakeholders affected, including primary, secondary, tertiary and facilitating, together with their objectives, goals and tasks.**
- **The workgroups within the organization, both formal and informal.**
- **The changes or transformations that will be supported.**
- **The proposed technology and how it will work within the organization.**
- **External constraints and influences and performance measures.**

Various Socio-Technical Models are:

1. **CUSTOM methodology**
   CUSTOM is a socio-technical methodology designed to be practical to use in small organizations. It is based on the User Skills and Task Match (USTM) approach, developed to allow design teams to understand and fully document user requirements. CUSTOM focusses on establishing stakeholder requirements: all stakeholders are considered, not just the end-users.

2. **Open System Task Analysis (OSTA)**
   OSTA attempts to describe what happens when a technical system is introduced into an organizational work environment. Like CUSTOM, OSTA specifies both social and technical aspects of the system. However, whereas in CUSTOM these aspects are framed in terms of stakeholder perspectives, in OSTA they are captured through a focus on tasks.

3. **Soft Systems Methodology**
   Soft Systems Methodology (SSM) arises from the same tradition but takes a view of the organization as a system of which technology and people are components. There is no assumption of a particular solution: the emphasis is rather on understanding the situation fully.

4. **Participatory design**
   Participatory design is a philosophy that encompasses the whole design cycle. It is design in the workplace, where the user is involved not only as an experimental subject or as someone to be consulted when necessary but as a member of the design team. Users are therefore active collaborators in the design process, rather than passive participants whose involvement is entirely governed by the designer.

5. **Effective Technical and Human Implementation of Computer-based Systems (ETHICS)**
   ETHICS methodology, stakeholders are included as participants in the decision making process. ETHICS considers the process of system development as one of managing change: conflicts will occur and must be negotiated to ensure acceptance and satisfaction with the system. If any party is excluded from the decision-making process then their knowledge and contribution is not utilized, and they are more likely to be dissatisfied. However, participation is not always complete.

6. **Ethnographic methods**
   Ethnography is based on very detailed recording of the interactions between people and between people and their environment. It has a special focus on social relationships and how they affect the nature of work. The ethnographer does not enter actively into the situation and does not see things from a particular person's viewpoint.

7. **Contextual inquiry**
   Contextual inquiry has much in common with the ethnographic tradition: it studies the user in context, trying to capture the reality of his work culture and practice. However, it is also an approach rooted in practice and it differs in a number of significant ways from pure ethnographic study: the intention is to understand and to interpret the data gathered, and rather than attempting to take an open-ended view, the investigator acknowledges and challenges her particular focus. In addition, the explicit aim is to design a new system, whereas in a pure ethnographic study, it would be open ended.

6.a) Explain in detail about communication and collaboration models.

Ans.6.a)

Communication and Collaboration Models

All computer systems, single-user or multi-user, interact with the workgroups and organizations in which they are used.

We need to understand normal human–human communication:

- **face-to-face communication involves eyes, face and body.**
- **conversation can be analyzed to establish its detailed structure.**

This can then be applied to text-based conversation, which has:

- **reduced feedback for confirmation.**
- **less context to disambiguate utterances.**
- **slower pace of interaction but is more easily reviewed.**

Group working is more complex than that of a single person:

- **it is influenced by the physical environment.**
- **experiments are more difficult to control and record.**
- **field studies must take into account the social situation.**

Types of Communication and Collaboration Models

1. **Face-To-Face Communication**
    a. **Transfer effects and personal space**
    b. **Eye contact and gaze**
    c. **Gestures and body language**
    d. **Back channels, confirmation and interruption**
    e. **Turn-taking.**
2. **Conversation**
    a. **Basic conversational structure**
    b. **Context**
    c. **Topics, focus and forms of utterance.**
    d. **Breakdown and repair**
    e. **Speech act theory**
3. **Text Based Communication**
    a. **Back channels and affective state**
    b. **Grounding constraints**
    c. **Turn-taking.**
    d. **Context and deixis**
    e. **Pace and granularity**
4. **Group Working**
    a. **Group dynamics**
    b. **Physical layout**
    c. **Distributed cognition**

6.b) Discuss about Linguistic Models.

Ans.6.b)

Linguistic Models

The user's interaction with a computer is often viewed in terms of a language, so it is not surprising that several modeling formalisms have developed centered around this concept. BNF grammars are frequently used to specify dialogs.

The models here, although similar in form to dialog design notations, have been proposed with the intention of understanding the user's behavior and analyzing the cognitive difficulty of the interface.

Linguistic models play a crucial role in understanding cognitive processes in interactive systems. These models represent the user-system grammar, which helps in analyzing and predicting how users interact with the system through language-based communication.

By studying linguistic models, we can gain insights into how users understand and interpret system prompts, commands, and feedback. These models help in understanding the structure and organization of user-system dialogue, including the syntax, semantics, and pragmatics of the language used.

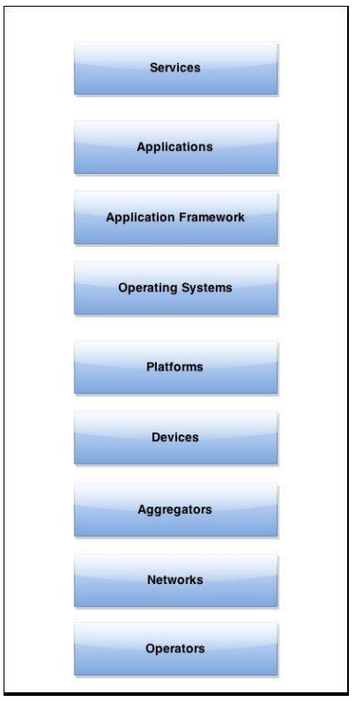
Linguistic models also aid in identifying potential language-related challenges and barriers that users may face while interacting with the system. They help in designing user interfaces and system responses that are clear, concise, and aligned with users' mental models and expectations.

Overall, linguistic models provide a framework for analyzing and improving the effectiveness of language-based communication in interactive systems, leading to enhanced user experience and system usability.

7.a) With neat diagram of mobile ecosystem, discuss its platforms and application frameworks.
Ans.7.a)

The Mobile Ecosystem

Mobile is an entirely unique ecosystem and like the Internet, it is made up of many different parts that must all work seamlessly together. With mobile technology, the parts are different, and because you can use mobile devices to access the Internet, which means that not only do you need to understand the facets of the Internet, but you also need to understand the mobile ecosystem.



- **Platforms**

A mobile platform's primary duty is to provide access to the devices. To run software and services on each of these devices, you need a platform, or a core programming language in which all of your software is written. Like all software platforms, these are split into three categories: licensed, proprietary, and open source.

The mobile ecosystem consists of various platforms that cater to different devices and operating systems. These platforms include:

1. iOS: Developed by Apple, iOS is the operating system used on iPhones, iPads, and iPod Touch devices. It offers a seamless user experience and is known for its security and stability.

2. Android: Developed by Google, Android is an open-source operating system used by a wide range of mobile devices. It provides a customizable interface and supports a vast number of applications.

3. Windows Mobile: Developed by Microsoft, Windows Mobile is an operating system designed for smartphones and tablets. It offers integration with other Microsoft products and services.

4. BlackBerry OS: Developed by BlackBerry Limited, BlackBerry OS is a proprietary operating system used on BlackBerry smartphones. It is known for its strong security features and efficient communication capabilities.

5. Symbian: Symbian was once a popular operating system for mobile devices, particularly Nokia phones. However, its usage has declined in recent years.

6. Other platforms: There are also other platforms such as Tizen, Firefox OS, and Ubuntu Touch, although their market share is relatively small compared to iOS and Android.

Each platform has its own set of features, design guidelines, and development tools, which developers need to consider when creating mobile applications.

- **Application Frameworks**

Application frameworks often run on top of operating systems, sharing core services such as communications, messaging, graphics, location, security, authentication, and many others.

i. **Java**
Applications written in the Java ME framework can often be deployed across the majority of Java-based devices but given the diversity of device screen size and process or power, cross-device deployment can be a challenge.

ii. **S60**
The S60 platform, formerly known as Series 60, is the application platform for devices that run the Symbian OS. S60 is often associated with Nokia devices—Nokia owns the platform—but it also runs on several non-Nokia devices. S60 is an open source framework. S60 applications can be created in Java, the Symbian C++ framework, or even Flash Lite.

iii. **BREW Applications written in the BREW application framework can be deployed across the majority of BREW-based devices, with slightly less cross-device adaptation than other frameworks.**

iv. **Flash Lite**
Adobe Flash Lite is an application framework that uses the Flash Lite and Action Script frameworks to create vector-based applications. Flash Lite applications can be run within the Flash Lite Player, which is available on a handful of devices around the world. Flash Lite is a promising and powerful platform, but there has been some difficulty getting it on devices. A distribution service for applications written in Flash Lite is long overdue.

v. **Windows Mobile**
Applications written using the Win32 API can be deployed across the majority of Windows Mobile-based devices. Like Java, Windows Mobile applications can be downloaded and installed over the air or loaded via a cable-connected computer.

vi. **Cocoa Touch**
Cocoa Touch is the API used to create native applications for the iPhone and iPod touch. Cocoa Touch applications must be submitted and certified by Apple before being included in the App Store. Once in the App Store, applications can be purchased, downloaded, and installed over the air or via a cable-connected computer.

vii. **Android SDK**
The Android SDK allows developers to create native applications for any device that runs the Android platform. By using the Android SDK, developers can write applications in C/C++ or use a Java virtual machine included in the OS that allows the creation of applications with Java, which is more common in the mobile ecosystem.

viii. **Web Runtimes (WRTs)**
Nokia, Opera, and Yahoo! provide various Web Runtimes, or WRTs. These are meant to be mini frameworks, based on web standards, to create mobile widgets. Both Opera's and Nokia's WRTs meet the W3C-recommended specifications for mobile widgets.

ix. **WebKit**
WebKit is a browser technology, so applications can be created simply by using web technologies such as HTML, CSS, and JavaScript. WebKit also supports a number of recommended standards not yet implemented in many desktop browsers. Applications can be run and tested in any WebKit browser, desktop, or mobile device.

x. **The Web**
The Web is the only application framework that works across virtually all devices and all platforms. Although innovation and usage of the Web as an application framework in mobile has been lacking for many years, increased demand to offer products and services outside of operator control, together with a desire to support more devices in shorter development cycles, has made the Web one of the most rapidly growing mobile application platforms to date.

7.b) Describe the following:

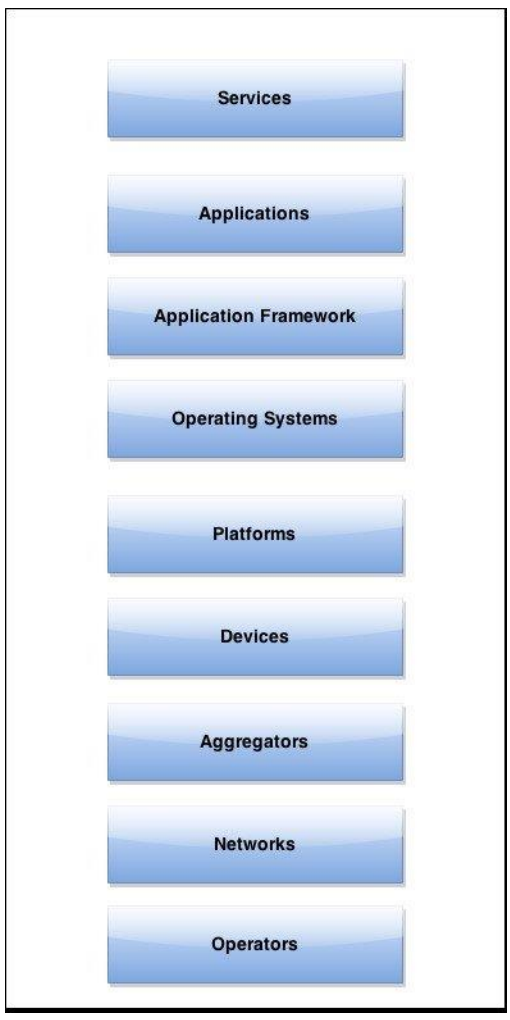
a. Mobile Ecosystem

b. Platforms

Ans.7.b)

a. Mobile Ecosystem

Mobile is an entirely unique ecosystem and like the Internet, it is made up of many different parts that must all work seamlessly together. With mobile technology, the parts are different, and because you can use mobile devices to access the Internet, which means that not only do you need to understand the facets of the Internet, but you also need to understand the mobile ecosystem.



i. **Operators**

The base layer in the mobile ecosystem is the operator. Operators go by many names, depending on what part of the world you happen to be in or to whom you are talking. Operators can be referred to as Mobile Network Operators (MNOs); mobile service providers, wireless carriers, or simply carriers; mobile phone operators; or cellular companies.

ii. **Networks**

Operators operate wireless networks. Remember that cellular technology is just a radio that receives a signal from an antenna. The type of radio and antenna determines the capability of the network and the services you can enable on it.

iii. **Devices**

What you call phones, the mobile industry calls handsets or terminals. These are terms that I think are becoming outdated with the emergence of wireless devices that rely on operator networks, but do not make phone calls.

iv. **Platforms**

A mobile platform's primary duty is to provide access to the devices. To run software and services on each of these devices, you need a platform, or a core programming language in which all of your software is written. Like all software platforms, these are split into three categories: licensed, proprietary, and open source.

v. **Application Frameworks**

Application frameworks often run on top of operating systems, sharing core services such as communications, messaging, graphics, location, security, authentication, and many others.

b. Platforms

Platforms

A mobile platform's primary duty is to provide access to the devices. Torun software and services on each of these devices, you need a platform, or a core programming language in which all of your software is written. Like all software platforms, these are split into three categories: licensed, proprietary, and open source.

In the context of mobile information architecture, platforms refer to the different operating systems or software environments on which mobile applications are developed and run. These platforms include iOS (Apple's operating system), Android (Google's operating system), Windows Mobile, and others. Each platform has its own set of design guidelines, development tools, and user interface elements that need to be considered when designing and organizing information for mobile applications.

8.a) Explain the various mobile information architecture.

Ans.8.a)

The various mobile information architecture.

1. **Site Maps:**

   Site maps are a classic information architecture deliverable. They visually represent the relationship of content to other content and provide a map for how the user will travel through the informational space. Mobile site maps are similar to site maps used on the web. However, there are a few tips specific to mobile that should be considered.

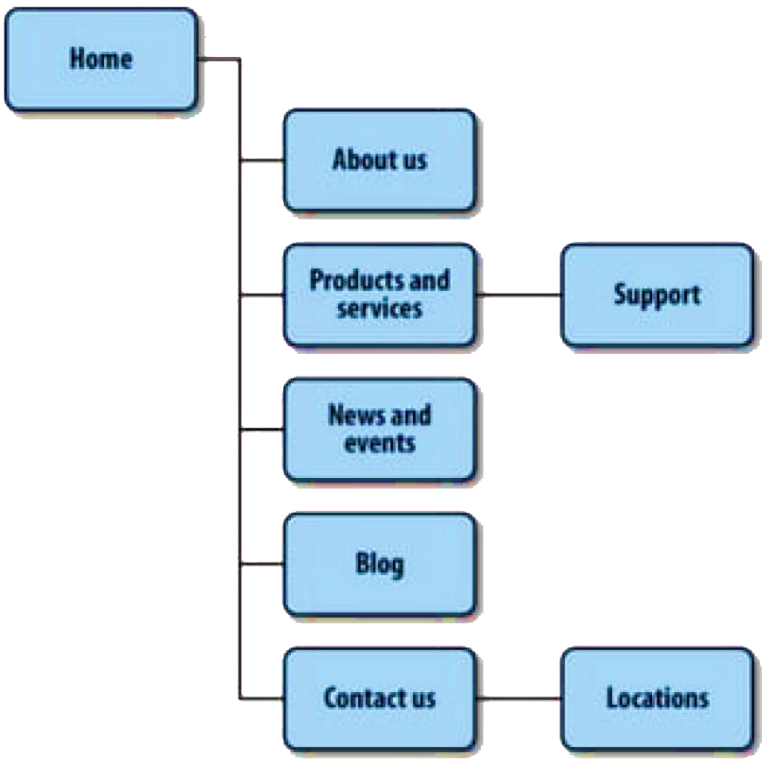


Fig: Example of a sitemap

2. **Clickstreams:**

   Clickstreams are used for showing the behavior on websites and displaying the order in which users travel through a site's information architecture. They are usually based on data gathered from server logs and are typically historical, used to identify flaws in the information architecture. Clickstreams can be visualized using heat-mapping or simple percentages to show where users are going.

3. **Wireframes:**

   Wireframes are a way to lay out information on the page, referred to as information design. They show how the user will directly interact with the content. Wireframes make the information space tangible and useful. They are like the peanut butter to the site map jelly in the information architecture sandwich.

4. **Prototyping**

   Wireframes lack the capability to communicate more complex, often in-place, interactions of mobile experiences. This is where prototypes come in.

   Prototypes might sound like a scary (or costly) step in the process. Some view them as redundant or too time-consuming, preferring to jump in and start coding things. But as with wireframes, I have found that each product we have built out some sort of prototype has saved both time and money.

   Types of Prototypes

   i. **Paper Prototype**
      A paper prototype, where the interaction is nothing more than drawings on note cards.
   ii. **Context Prototype**
      An example of a context prototype, or taking images loaded onto a device and testing them in the mobile context.
   iii. **HTML Prototypes**
      An HTML/XHTML prototype that you can actually interact with on real mobile devices.

These components, along with clear and simple labels, support the defined goals of the mobile application and help create an effective mobile information architecture.

8.b) List and explain the elements of mobile design.

Ans.8.b)

The Elements of Mobile Design

Good design requires three abilities: the first is a natural gift for being able to see visually how something should look that produces a desired emotion with the target audience. The second is the ability to manifest that vision into something for others to see, use, or participate in. The third knows how to utilize the medium to achieve your design goals.

Elements of mobile design that you need to consider, starting with the context and layering in visual elements or laying out content to achieve the design goal. Then, you need to understand how to use specific tools to create mobile design, and finally, you need to understand the specific design considerations of the mobile medium.

i. Context

I will not labor the point except to say that context is core to the mobile experience. As the designer, it is your job to make sure that the user can figure out how to address context using your app.

ii. Message

Message is the overall mental impression you create explicitly through visual design. I like to think of it as the holistic or at times instinctual reaction someone will have to your design. If you take a step back, and look at a design from a distance, what is your impression? Or conversely, look at a design for 30 seconds, and then put it down.

iii. Look and Feel

Look and feel is used to describe appearance, as if I want a clean look and feel, or I want a usable look and feel. The problem is: as a mobile designer, what does it mean? And how is that different than messaging? I think of looking and feel in a literal sense, as something real and tactile that the users can look at, and then feel something they can touch or interact with. Look and feel is used to evoke action how the user will use an interface.

iv. Layout

Layout is an important design element, because it is how the user will visually process the page, but the structural and visual components of layout often get merged together, creating confusion, and making your design more difficult to produce. The first-time layout should rear its head is during information architecture.

v. Color

The fifth design element, color, is hard to talk about in a black-and-white book. It is fitting, because it was not that long ago that mobile screens were available only in black and white well, technically, it was black on a green screen. These days, we have the entire spectrum of colors to choose from for mobile designs.

vi. Typography

Typography involves the choice of fonts, font sizes, and text formatting. Clear and readable typography is crucial for effective communication of information on mobile screens, considering their smaller size.

vii. Graphics

Graphics, including images and icons, contribute to the visual appeal and communication of information. Well-designed graphics enhance the user experience and can be used to represent actions, features, or convey emotions.

9.a) Discuss in detail the purpose of drag and drop.

Ans.9.a)

One of the most useful purposes of drag and drop is to allow the user to directly place objects where they want them on the page. A typical pattern is Drag and Drop Modules on a page. Netvibes provides a good example of this interaction pattern.

The purpose of drag and drop is to provide a user-friendly and intuitive way for users to interact with digital content. It allows users to select an object or piece of information and move it to a different location by dragging it with the cursor and dropping it in the desired location.

In terms of user interface design, drag and drop is an important interaction pattern that can enhance the usability and efficiency of an interface. It can simplify complex tasks by allowing users to directly manipulate objects instead of relying on traditional input methods like menus or buttons. Drag and drop can also improve the overall user experience by providing visual feedback and affordances that indicate the draggable nature of objects and valid drop targets.

Designing drag and drop interactions requires careful consideration of various factors, such as indicating draggability, defining valid drop targets, and choosing the appropriate visual representations for dragged objects. It is important to create a clear and consistent interface that guides users in understanding how to use drag and drop effectively.

Overall, drag and drop is a powerful tool in user interface design that can enhance usability, efficiency, and user satisfaction.

9.b) Explain the steps involved in designing a web interface.

Ans.9.b)

The steps involved in Designing a Web Interface.

1. **User Interface Design Basics**
   User Interface (UI) Design focuses on anticipating what users might need to do and ensuring that the interface has elements that are easy to access, understand, and use to facilitate those actions. UI brings together concepts from interaction design, visual design, and information architecture.
2. **Choosing Interface Elements**
   Users have become familiar with interface elements acting in a certain way, so try to be consistent and predictable in your choices and their layout. Doing so will help with task completion, efficiency, and satisfaction.
3. **Interface elements include but are not limited to:**
   Input Controls: buttons, text fields, checkboxes, radio buttons, dropdown lists, list boxes, toggles, date field
   Navigational Components: breadcrumb, slider, search field, pagination, slider, tags, icons Informational
   Components: tooltips, icons, progress bar, notifications, message boxes, modal windows Containers: accordion There are times when multiple elements might be appropriate for displaying content. When this happens, it is important to consider the trade-offs.
   For example, sometimes elements that can help save you space, put more of a burden on the user mentally by forcing them to guess what is within the dropdown or what the element might be.
4. **Best Practices for Designing an Interface**
   Everything stems from knowing your users, including understanding their goals, skills, preferences, and tendencies. Once you know about your user, make sure to consider the following when designing your interface:
5. **Keep the interface simple.**

- **The best interfaces are almost invisible to the user. They avoid unnecessary elements and are clear in the language they use on labels and in messaging.**
- **Create consistency and use common UI elements. By using common elements in your UI, users feel more comfortable and are able to get things done more quickly.**
- **It is also important to create patterns in language, layout and design throughout the site to help facilitate efficiency. Once a user learn show to do something, they should be able to transfer that skill to other parts of the site.**
- **Be purposeful in page layout. Consider the spatial relationships between items on the page and structure the page based on importance.**
- **Careful placement of items can help draw attention to the most important pieces of information and can aid scanning and readability.**
- **Strategically use color and texture. You can direct attention toward or redirect attention away from items using color, light, contrast, and texture to your advantage.**
- **Use typography to create hierarchy and clarity. Carefully consider how you use typeface. Different sizes, fonts, and arrangement of the text to help increase scalability, legibility and readability.**
- **Make sure that the system communicates what is happening. Always inform your users of location, actions, changes in state, or errors.**
- **The use of various UI elements to communicate status and, if necessary, next steps can reduce frustration for your user. Think about the defaults.**
- **By carefully thinking about and anticipating the goals people bring to your site, you can create defaults that reduce the burden on the user.**
- **This becomes particularly important when it comes to form design where you might have an opportunity to have some fields pre-chosen or filled out.**

10.a) Discuss in detail the various types of selection patterns.

Ans.10.a)

The various types of Selection Patterns

1) **Toggle Selection: Checkbox or control-based selection.**
   The most common form of selection on the Web is Toggle Selection. Checkboxes and toggle buttons are the familiar interface for selecting elements on most webpages. The way to select an individual mail message is through the row's checkbox. Clicking on the row itself does not select the message. We call this pattern of selection Toggle Selection since toggle-style controls are typically used for selecting items. Once items have been check-selected, actions can be performed on them. Usually, these actions are performed on the selection by clicking on a separate button (e.g., the Delete button). Gmail is a good example of actions in concert with Toggle Selection.

2) **Collected Selection: Selection that spans multiple pages.**
   Toggle Selection is great for showing a list of items on a single page. But what happens if you want to collect selected items across multiple pages? Collected Selection is a pattern for keeping track of selection as it spans multiple pages. In Gmail, you can select items as you move from page to page. The selections are remembered for each page. If you select two items on page one, then move to page two and select three items, there are only three items selected. This is because actions only operate on a single page. This makes sense, as users do not normally expect selected items to be remembered across.

3) **Object Selection: Direct object selection.**
   Object Selection is when selection is made directly on objects within the interface. Sometimes using a checkbox does not fit in with the style of interaction desired. Laszlo's WebTop mail allows the user to select messages by clicking anywhere in the row.

4) **Hybrid Selection: Combination of Toggle Selection and Object Selection**
   Mixing Toggle Selection and Object Selection in the same interface can lead to a confusing interface. Referring back to Yahoo! Bookmarks, you will see an odd situation arise during drag and drop. In Yahoo! Bookmarks, one item is selected, but two items can be dragged by dragging on the unselected item

10.b) Explain in detail the various ways to reveal contextual tools.

Ans.10.b)

The various ways to reveal Contextual Tools

Contextual Tools are the Web's version of the desktop's right-click menus. Instead of having to right-click to reveal a menu, we can reveal tools in context with the content. We can do this in a number of ways:

i. **Always-Visible Tools: Place Contextual Tools directly in the content.**
   The simplest version of Contextual Tools is to use Always-Visible Tools. Digg is an example of making Contextual Tools always visible.

ii. **Hover-Reveal Tools: Show Contextual Tools on mouse hover.**
   One way to do this is to reveal the tools when the user pauses the mouse over an object. The Hover-Reveal Tools pattern is most clearly illustrated by 37 Signal's Backpack it. To-do items may be deleted or edited directly in the interface. The tools to accomplish this are revealed on mouse hover.

iii. **Toggle-Reveal Tools: A master switch to toggle on/off Contextual Tools for the page.**
   A variation on the two previous approaches is to not show any Contextual Tools until a special mode is set on the page. A good example of Toggle-Reveal Tools is in Basecamp's category editing.

iv. **Multi-Level Tools: Progressively reveal actions based on user interaction.**
   Contextual Tools can be revealed progressively with Multi-Level Tools. Songza* provides a set of tools that get revealed after a user clicks on a song. Additional tools are revealed when hovering over the newly visible tools.

v. **Secondary Menus: Show a secondary menu (usually by right-clicking on an object).**
   A secondary menu is shown, usually by right-clicking on an object, to reveal additional contextual tools.