Module -1
Introduction to HCI and Usability

SYLLABUS
Introduction- - Components of Interaction – Ergonomics Designing Interactive systems – Understanding Users cognition and cognitive frameworks, User Centered approaches, Usability goals and measures, Universal Usability-Diverse Cognitive and Perceptual abilities, Personality differences, Cultural and International diversity, Users with disabilities- Older Adult users and Children. Guidelines, Principles and Theories.

INTRODUCTION

- Human–computer interaction (HCI), alternatively **man–machine interaction** (MMI) or **computer–human interaction** (CHI) is the **study of interaction between people (users) and computers.**
- Human-Computer Interaction is a multidisciplinary field that focuses on **designing** and **evaluating** computer systems and technologies that people interact with. It is concerned with understanding and improving the interaction between humans and computers to make technology more user-friendly, efficient, and enjoyable.
- HCI specialists consider how to **develop and deploy** computer systems that satisfy human users. The majority of this research focuses on enhancing human-computer interaction by enhancing how people utilize and comprehend an interface
- Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.

GOALS

• A **basic goal** of HCI is – to improve the interactions between users and computers by making computers more usable and receptive to the user's needs.
• A **long term goal** of HCI is – to design systems that minimize the barrier between the human's cognitive model of what they want to accomplish and the computer's understanding of the user's task.

IMPORTANCE OF HCI

1. **Enhancing User Experience**: HCI places a strong emphasis on creating technology that is user-friendly and intuitive. When users have positive experiences with technology, they are more likely to adopt it, use it more effectively, and express higher levels of satisfaction.
2. **Increasing Productivity and Efficiency**: HCI principles are instrumental in designing systems and interfaces that streamline tasks and workflows. By reducing user effort and cognitive load, HCI contributes to increased productivity in various domains, from business applications to healthcare.
3. **Reducing Errors and Frustration**: Through careful design and usability testing, HCI helps identify and mitigate potential sources of errors and user frustration. This can be especially critical in fields like aviation, healthcare, and finance, where errors can have serious consequences.

4. **Enhancing Accessibility and Inclusivity**: HCI ensures that technology is accessible to individuals with disabilities. By considering diverse user needs, HCI promotes inclusivity and equal access to information and services.
5. **Improving Decision-Making**: Effective data visualization and information presentation, key aspects of HCI, help users make informed decisions. In fields like data analytics and business intelligence, HCI plays a vital role in transforming complex data into actionable insights.
6. **Enhancing Safety**: In critical systems such as self-driving cars, medical devices, and industrial equipment, HCI is crucial for designing interfaces that prioritize safety and usability, reducing the likelihood of accidents.
7. **Driving Innovation**: HCI encourages creative thinking in technology design. As new interaction methods and technologies emerge (e.g., touch screens, voice assistants, augmented reality), HCI researchers and designers explore innovative ways to make them user-friendly and functional.
8. **Adapting to Changing Technologies**: HCI researchers and practitioners continuously adapt to evolving technology trends. As technology advances, HCI professionals help users navigate and harness the potential of new tools and interfaces.
9. **Ethical Considerations**: HCI is increasingly addressing ethical concerns related to technology design and use. This includes issues like data privacy, algorithmic bias, and the social impact of technology. HCI professionals work to ensure that technology respects ethical principles and societal values.
10. **User Satisfaction and Loyalty**: A positive user experience fosters user satisfaction and loyalty. Satisfied users are more likely to recommend products or services to others, contributing to brand reputation and growth.
11. **Cost Savings**: Designing technology with HCI principles in mind can lead to cost savings in terms of reduced customer support needs, fewer errors and rework, and increased user efficiency.

Components of Human-Computer Interaction

HCI is primarily composed of four essential elements:
1. **The User:** An individual or a group of individuals who work together on a project is referred to as the user component. HCI researches the needs, objectives, and interaction styles of users.
2. **The Goal-Oriented Task:** When using a computer, a user always has a purpose or aim in mind. To achieve this, the computer presents a digital representation of things.
3. **The Interface:** An essential HCI element that can improve the quality of user interaction is the interface. Many interface-related factors need to be taken into account, including the type of interaction, screen resolution, display size, and even color contrast.
4. **The Context:** HCI is not only about providing better communication between users and computers but also about factoring in the context and environment in which the system is accessed.

Examples of HCI

- IoT: pre-touch sensing and paper ID
- Eye tracking: gaze detection
- Speed recognition
- Cloud Computing

COMPONENTS OF INTERACTION

Human-Computer Interaction (HCI) involves the study of the interaction between people (users) and computers. This interaction is composed of several essential elements. These components are central to understanding and designing effective interactive systems.

Here are the components of interaction:

- **The User**
  - The User is defined as the **individual or group of individuals who interact with the computer system.** HCI research investigates the **needs, objectives,** and **interaction styles** of users.
  - Users are the **people who use the technology** to accomplish work. They **initiate actions, provide input, interpret output, and make decisions** based on the system's responses.
  - Designing for users requires understanding their capabilities and limitations. Users vary widely in age, expertise, cultural background, and physical abilities. These individual differences, such as cognitive abilities, preferences, and prior experiences, influence how users approach and perform interactions.
  - Background experience and knowledge in the task domain and the interface domain play key roles in learning and performance. User characteristics capture attributes like abilities, skills, educational background, preferences, and physical or mental disabilities. **Users can be novices, experts, casual, or frequent users**, which affects design choices.
  - Psychological and physiological attributes of the user provide a basic overview of capabilities and limitations. Interaction design must understand people from psychological and social aspects, including human error.
  - Users may interact individually or work together in a group. Social interactions, such as conversations with other people, are a part of task performance that the interface must accommodate.
- **The Computer (and Interface)**
  - The Computer component represents the **combination of hardware and software that forms the interactive system**. This includes devices like desktop computers, laptops, tablets, and the applications running on them.
  - The computer **executes processes, receives user input, processes it, and produces output**. The capabilities and limitations of the computer system (processing speed, memory, storage) impact performance and user experience. Understanding computers – their limitations, capacities, tools, and platforms – is essential for interaction design.
  - The **Interface** is an essential HCI **element that can improve the quality of user interaction**. It is the part of the computer system that people can see, hear, touch, talk to, or otherwise understand or direct. **The interface effectively translates between the user and the system to ensure successful interaction**.
  - The interface has two components: **Input** and **Output**.
    - **Input** is **how** a **person communicates needs or desires to the computer**. Input devices include keyboards, mouse, trackballs, touchscreens, styluses, gestural interfaces, eye tracking, and speech recognition. These devices vary

        - in how they support tasks like pointing, selecting, dragging, drawing, or data entry.
        - **Output** is **how** the **computer conveys the results or requirements to the user**. Output devices include display screens (various types, sizes, and resolutions), printers, speakers, and other auditory mechanisms. Output displays information in a format and at a pace adapted to the user.
    - The interface is composed of elements like windows, controls (text boxes, lists, buttons, tabs), menus, and display layouts.
    - **The interface mediates between the user's language (task language) and the system's language (core language)**. Different **interaction styles** significantly affect the dialog between user and computer. Common styles include command line, menus, natural language, question/answer, form-fills, spreadsheets, WIMP (Windows, Icons, Menus, Pointer), point and click, and direct manipulation. **Interaction can also be categorised by activity: instructing, conversing, manipulating and navigating, and exploring and browsing.** Direct manipulation, where users interact with visual representations of objects and actions, is particularly effective when the computer provides a direct-manipulation representation of the world of action, allowing the computer to "vanish" as users focus on their task.
- **The Task**
    - A **Task** represents the **goal or objective that the user aims to achieve through interaction with the computer system**. Tasks can range from simple actions to complex operations.
    - Understanding the nature of tasks is crucial for designing interfaces that support users in accomplishing their goals efficiently. **Task analysis** is a key technique used to gain an understanding of what the system must do by analysing the user's goals, intentions, and actions.
    - Tasks can be categorised as **cognitive** (thinking-based), **perceptual** (sensory-based), or **motor** (action-based).
    - The activities performed in completing a task, the methods used, and interactions with other people or systems are part of task analysis.
    - The principle of task compatibility states that the organisation of a system should match the tasks a person must do to perform their job. Users often think in terms of tasks, not applications. The human action cycle describes users' goal-oriented behaviour when interacting with computers.
- **The Environment**
    - The **Environment** encompasses the broader **context or setting in which the interaction takes place**. This interaction does not occur in a vacuum.
    - Contextual factors include the physical environment (e.g., workplace, home, lighting, noise). The **physical environment** in which the interaction takes place is a primary focus of ergonomics. The physical environment is a factor in system design.
    - The **social context** (e.g., collaborative or individual tasks, interaction with colleagues) and **organizational context** (e.g., user support, communication infrastructure, management structure) also influence the interaction. Computers and systems are placed within groups and social situations.
    - Context-of-use theories highlight the critical role of users' complex interactions with other people, other electronic devices, and paper resources. The arguments of situated action and distributed cognition suggest that users' actions are situated in time and

place, influenced by other people and environmental contingencies, and that knowledge is distributed in the environment (paper, electronic files, colleagues).
    - Understanding the context of use, types of activity, user experience goals, accessibility, cultural differences, and user groups is required for optimising interaction. Ubiquitous computing shifts the focus from the desktop to the surrounding environment, where computing devices are integrated with the physical world, and interaction incorporates the environment. The physical and social environments are inextricably intertwined with the use of technology.

In essence, the interaction is a **dynamic process** or dialog between the User and the Computer System, mediated by the Interface, undertaken to achieve specific Tasks, and influenced by the broader physical, social, and organizational Environment.

ERGONOMICS DESIGNING INTERACTIVE SYSTEMS

- Ergonomics plays a crucial role in the design of interactive systems, ensuring that they are user-friendly, efficient, and comfortable for users.
- Ergonomics, also known as **human factors engineering**, is the **scientific disciplin**e concerned with the **understanding of interactions among humans and other elements of a system**. In the context of designing interactive systems, **it focuses on optimizing the system to match the capabilities and limitations of the human users.**

Key Principles of Ergonomics in Interactive Systems Design:

1. **User-Centered Design (UCD):** User-Centered Design places **users at the core of the design process**. It involves methods such as user research, personas, and usability testing. By incorporating **user feedback** iteratively, designers can align the system's features, layout, and functionality with user expectations, resulting in interfaces that are intuitive and user-friendly.
    - UCD is a fundamental principle in ergonomics, emphasizing the importance of involving users throughout the design process.
    - Design decisions should be based on an understanding of the users' needs, preferences, and abilities.
2. **Task Analysis:** Task analysis involves breaking down user activities into discrete tasks. This process aids in understanding the steps users take to achieve their goals. Designers can use this information to streamline workflows, eliminate unnecessary

steps, and create interfaces that match user mental models, contributing to efficiency and ease of use.

- Conducting a thorough task analysis helps designers understand the specific tasks users need to perform with the interactive system.
- Identifying the sequence of actions and decision points aids in creating an efficient and user-friendly interface.

3. **Anthropometry:** Anthropometric data is essential for accommodating the physical diversity of users. Designers must consider factors such as the range of user heights, arm lengths, and hand sizes. This ensures that interactive elements are positioned and sized to be comfortably accessible for a broad spectrum of users.
   - Anthropometric data is essential for designing interfaces that accommodate a diverse range of user sizes and shapes.
   - Designers should consider factors such as reach, vision angles, and hand sizes to ensure inclusivity.
4. **Feedback and Feedforward:** Feedback informs users about the outcomes of their actions, providing a sense of control. Feedforward, on the other hand, offers proactive guidance, helping users anticipate system responses. Both mechanisms contribute to a smooth user experience, reducing uncertainty and minimizing the risk of errors.
   - Providing users with feedback on their actions and the system's status enhances user understanding and control.
   - Feedforward, or proactive information, helps users anticipate the consequences of their actions, reducing errors.
5. **Minimizing Cognitive Load:** Cognitive load refers to the **mental effort required to complete a task**. Designers aim to minimize cognitive load by presenting information in a **clear and organized manner**. Logical information architecture, consistent design patterns, and concise labeling contribute to a seamless user experience, allowing users to focus on their tasks.
   - Cognitive load refers to the mental effort required to perform a task. Design should aim to minimize unnecessary cognitive load, ensuring users can focus on the task at hand.
   - Clear and concise information presentation, logical navigation, and consistent design contribute to lower cognitive load.
6. **Physical Ergonomics:** Considering physical ergonomics involves **designing interfaces and input devices that align with human anatomy and movement**. Key

considerations include the layout of controls, the design of keyboards and mouse, and the placement of touchscreens to **reduce physical strain** during interaction, promoting **comfort and usability**.

- Consider the physical aspects of user interaction, including the design of input devices (e.g., keyboards, mouse, touchscreens) and the layout of controls.
- Ensure that users can interact with the system comfortably and with minimal physical strain.

7. **Accessibility:** Designing for accessibility ensures that interactive systems are **usable by individuals with diverse abilities**. This includes considerations for users with visual, auditory, motor, or cognitive impairments. Adhering to accessibility standards and incorporating features like screen readers and alternative input methods fosters inclusivity.
    - Ergonomics also involves making interactive systems accessible to users with diverse abilities and disabilities.
    - Design for screen readers, alternative input methods, and other assistive technologies to ensure inclusivity.
8. **Error Prevention and Recovery:** Designing interfaces to prevent errors involves clear communication of system status and providing users with cues to avoid mistakes. However, recognizing that errors may occur, designers must also implement effective error recovery mechanisms, enabling users to correct or undo actions easily.
    - Design with the goal of preventing errors, but also provide mechanisms for users to recover from mistakes.
    - Clear error messages, undo functionality, and user-friendly error recovery paths contribute to a positive user experience.
9. **Aesthetics and Enjoyability:** While functionality is crucial, aesthetics and enjoyability contribute to user satisfaction. Well-designed interfaces with visually pleasing elements and a positive overall aesthetic can enhance user engagement and make the interaction more enjoyable.
    - While functionality is crucial, aesthetics and enjoyability also play a role in user satisfaction.
    - A visually pleasing and engaging design can contribute to a positive user experience.
10. **Iterative Design:**

- Ergonomics in interactive systems is an iterative process. Regular testing and refinement based on user feedback contribute to continuous improvement.

UNDERSTANDING USER COGNITION AND COGNITIVE FRAMEWORKS

Understanding **user cognition and cognitive frameworks** is a vital foundation for designing interactive systems. It is essential for optimising the system to match the capabilities and limitations of human users.

What is Cognition and Why is it Important?

Cognition is the core aspect of human intelligence and consciousness, allowing individuals to interact with their environment, make sense of stimuli, and engage in complex mental tasks.

Cognition refers to the **mental processes and activities** involved in **acquiring, processing, storing, and using information**. It encompasses a wide range of mental functions, including:

- **Perception: Interpreting sensory input** rapidly, including visual, auditory, and tactile inputs. How users perceive a device is important. The way an interface is designed greatly affects how well people can perceive. Knowledge of naive psychology can help predict what stimuli are salient and where attention will be focused.
- **Attention:** The ability to **focus on certain stimuli**. Users rapidly recognise slight changes on displays and initiate commands. Users constantly switch attention among multiple tasks, and understanding this impact is important. The way an interface is designed greatly affects how well people can attend. Design implications for attention include clear goals and information presentation.
- **Memory:** Involves **storing and processing information**. Designers must consider how users retrieve stored knowledge. Memory is not the most stable attribute, consisting of long-term and short-term (working) memory. The way an interface is designed greatly affects how well people can remember how to carry out tasks. A system designer can minimize memory load.
- **Learning:** How users learn to interact and become experienced with computer systems. Learning involves complex human processes required in user interfaces. The way an interface is designed greatly affects **how well people can learn how to carry out tasks**. Interfaces that can be learned by exploratory browsing are considered, and cognitive walkthroughs were developed for these.
- **Problem-Solving, Reasoning, and Decision-Making:** Users engage in these mental processes when interacting with systems. Complex human problem-solving is required in user interfaces. Design implications include providing easy-to-access information and help for understanding how to carry out activities more effectively, and using simple, memorable functions to support rapid decision-making and planning.
- **Language Processing (Reading, Speaking, Listening):** Mental processes involved in understanding and using language. The way an interface is designed greatly affects how well people can read, speak, and listen in interaction.

Understanding these cognitive aspects is crucial because it allows designers to create technologies that **both extend human capabilities and compensate for human weaknesses**. It helps designers understand the demands an abstract design would place on the user's memory and interaction timing. Knowledge of human capabilities and limitations helps in designing systems that support the former and compensate for the latter.

Modes of Cognition

Norman (1993) distinguishes between two general modes:

1. Experiential cognition
2. Reflective cognition

Experiential cognition

It is the state of mind in which we perceive, act, and react to events around us effectively and effortlessly. It requires reaching a certain level of expertise and engagement. Examples include driving a car, reading a book, having a conversation, and playing a video game.

Reflective cognition

Reflective cognition involves thinking, comparing, and decision-making. This kind of cognition is what leads to new ideas and creativity. Examples include designing, learning, and writing a book.

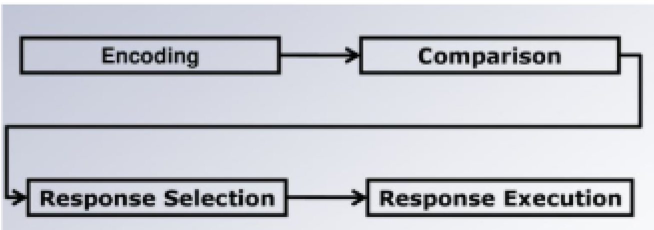
Understanding **human information processing** and models like **GOMS** is fundamental in human-computer interaction (HCI) for designing effective systems.

Human Information Processing

Human information processing models provide a **framework for understanding how users interact with systems by conceptualising the mind as a processing system**. Information enters, is stored, processed, and then output. This perspective allows for predictions about human performance.

The model encompasses various mental processes:

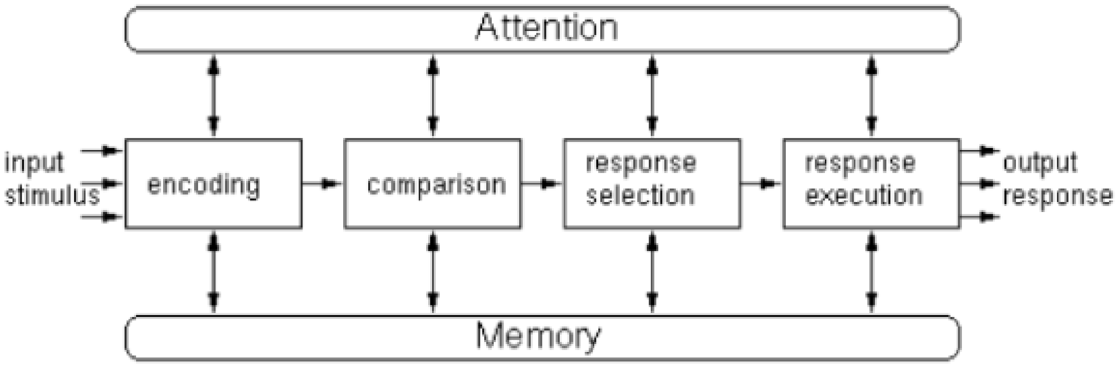
- Information is received through the senses, particularly sight, hearing, and touch in the context of computer use. How information is **perceived** by perceptual processors is key.
- It involves how that information is **attended** to.
- Information is processed and stored in **memory**, including temporary storage in sensory or working memory, and permanently in long-term memory. The capacity and limitations of human memory are relevant.
- Information is used in **reasoning** and **problem-solving**. These are complex human processes required in user interfaces.
- **Learning** how to interact and become experienced with systems is also part of this. Skill acquisition occurs as information structures become better defined through recurrent situations.



1. **Encoding:** Information from environment is encoded in some kind of internal representation
2. **Comparison:** Internal representation is compared with memorized representations
3. **Response Selection:** Concerned with deciding on a response to the encoded stimulus
4. **Response Execution:** Organizing response and necessary action

The **Human Processor Model** is a well-known approach within this framework. It models the cognitive processes of a user interacting with a computer, conceptualising cognition as a series of processing stages (perceptual, cognitive, and motor processors) organised in relation to one another.

An extended model includes the processes of attention and memory. This model predicts which cognitive processes are involved and can be used to calculate how long a user will take to carry out tasks.



In the extended model, cognition is viewed in terms of:

- How information is *perceived* by the perceptual processors
- How that information is *attended*
- How that information is processed and stored in *memory*

Human Processor Model - Perceptual System

- It helps us to conceptualize human behavior and to model human processors.
- It is composed of three different systems, each of which having two components i.e. a processor component and a memory
- Component: -

1. Perceptual System
2. Motor System
3. Cognitive System

- This measures the qualitative aspects of human performance

Understanding these processes helps designers comprehend the demands an abstract design places on the user's memory and interaction timing. Knowledge of human capabilities and limitations supports designing systems that leverage strengths and compensate for weaknesses.

GOMS Model

The GOMS model is a key **predictive modelling technique** in HCI that emerged from psychological theory and models of human information processing, specifically as a simplification of the Human Processor Model. It makes use of a model of mental processing where the user achieves **goals** by solving subgoals in a **divide-and-conquer** fashion.

GOMS is an acronym standing for:

- **G**oals: The user's intention, what they want to achieve (e.g., save a file). Goals are broken down into subgoals, forming a hierarchy.
- **O**perators: Basic actions the user performs (e.g., pressing a key, moving the mouse).
- **M**ethods: Learned sequences of operators and subgoals used to achieve a goal.
- **S**election rules: Rules used to choose among alternative methods for achieving a goal.

GOMS is a generic term referring to a family of models that vary in their granularity. These models describe how a user performs a computer-based task. They are used mainly to predict user performance when comparing different applications and devices.

One of the main attractions of the GOMS approach is that it allows **quantitative predictions** to be made. It can predict the time it takes to perform tasks and identify effective strategies.

Notable applications of GOMS analysis include:

- **Project Ernestine**: A study that used GOMS analysis to predict and explain the performance of telephone call operators using a proposed new workstation. Empirical data from the existing system was compared with hypothetical data derived from the GOMS analysis of the new system.
- **NYNEX Case Study**: A detailed GOMS analysis was performed for a new computer system intended for telephone operators. By taking into account the cognitive and physical processes, the analysis (using a PERT-style GOMS description for parallel activities) predicted that the new system would actually take *longer* to process each call than the existing one. This analysis led to the new system being abandoned before installation, saving significant costs.

GOMS has also been transformed into other methods, such as the **Keystroke Level Model**, which provides a formula for determining the time taken by methods and operations. It shares features with other models like Cognitive Complexity Theory (CCT) and Task-Action Grammar (TAG) which also use goal and task hierarchies.

One of the problems of abstracting a quantitative model from a qualitative description of user performance is ensuring that two are connected. In particular, it has been noted that the form and contents of the GOMS family of models are relatively unrelated to the form and content of the model human processor and it also oversimplified human behavior. More recently, attention has focused on explaining:

1. **Knowledge Representation Models**: How knowledge is represented
2. **Mental Models:** How mental models (these refer to representation people construct in their mind of themselves, others, objects and the environment to help them know what to do in current and future situations) develop and are used in HCI
3. **User Interaction Learning Models:** How users learn to interact and become experienced in using computer systems. With respect to applying this knowledge to HCI design, there has been considerable research in developing.
4. **Conceptual Models:** Conceptual models are (these are the various ways in which systems are understood by different people) to help designers develop appropriate interfaces.
5. **Interface Metaphor:** Interface metaphors are (these are GUIs that consist of electronic counterparts to physical objects in the real world) to match the knowledge requirements of users.

USER CENTERED APPROACHES

A **user-centered approach**, often referred to as **User-Centered Design (UCD)**, is a philosophy in which the development of a product or system is driven primarily by the needs and goals of the intended users. It aims to create interfaces that are directly relevant to tasks and support users effectively, addressing the issues caused by poorly designed interfaces where users are forced to adapt..

Key elements and methods within a user-centered approach include:

- **User-Centered Design (UCD):**
  - **Understanding Users:** Begin by researching and understanding the target audience. This involves conducting user interviews, surveys, and observations to gather insights into their needs, preferences, and behaviors.
  - **Prototyping and Testing:** Create prototypes early in the design process and test them with real users. This iterative feedback loop ensures that the final product aligns with user expectations and needs.

    - **Iterative Refinement:** Continuously refine designs based on user feedback. This iterative process helps to uncover and address usability issues as the design evolves.
- **User Research**: This involves gathering insights into user needs, preferences, and behaviours. Techniques include interviews, surveys, observation, focus groups, task analysis, behavioural analysis, and empathy building. This research helps understand the target audience from the outset.
    - **Methods:** Use a variety of research methods, including interviews, surveys, focus groups, and usability testing.
    - **Behavioral Analysis:** Observe and analyze user behavior to gain insights into how they interact with the product or system in real-world scenarios.
    - **Empathy Building:** Develop empathy for users by understanding their goals, challenges, and context, which helps in designing solutions that truly address their needs.
- **Persona Development**: A method where detailed profiles representing different user types are created based on user research. Personas capture demographics, goals, characteristics, and more, serving as a reference throughout design to ensure decisions align with user needs. They are useful when direct user involvement is limited.
    - **User Profiles:** Create detailed personas representing different user types, including their goals, characteristics, and pain points.
    - **Guiding Design Decisions:** Use personas as a reference point throughout the design process to ensure that decisions align with the needs of specific user groups.
- **Iterative Design**: This is a fundamental principle where designs are developed, evaluated with users, and refined through multiple cycles. Iteration is crucial because initial solutions are rarely perfect, and it ensures continuous improvement based on user feedback. Prototyping is a key part of this, allowing designs to be visualised and tested with users at different stages.
    - **Prototyping:** Develop prototypes of the product or system at various stages to visualize and test design concepts before final implementation.
    - **User Feedback:** Collect feedback from users at each iteration. This feedback is used to make informed design decisions and improvements.
    - **Continuous Improvement:** Iterate on the design based on user input, gradually refining the product to meet user needs effectively.

USABILITY GOALS AND MEASURES

Usability Goals

Usability goals represent the **desired attributes of a system from the user's perspective**, focusing on how the product supports users in carrying out their activities. Setting **explicit goals** is crucial for designers and managers, moving beyond vague terms like "user-friendly" to focus on specific, measurable objectives. These goals are widely accepted as the basis for a user-centered approach, along with an early focus on users and tasks and empirical measurement. Identifying usability and user experience goals is considered essential for making a product successful.

Key usability goals:

- **Effectiveness**: Refers to **how good a product is at doing what it is supposed to do**. It concerns the accuracy and completeness with which specified users can achieve specified goals in particular environments. A key question is whether the system is capable of allowing people to achieve their tasks, such as learning, working efficiently, accessing information, or buying goods. Effectiveness can be difficult to measure directly as it is a **combination of other goals**. Measures can include the **percentage of goals achieved** or the **number of power features used**.
- **Efficiency**: Focuses on **how a product supports users** in carrying out their tasks, particularly with **speed** and **accuracy**. It relates to the resources expended in relation to the accuracy and completeness of goals achieved. An efficient system allows users to complete common tasks with a minimal number of steps. The question for designers is whether users can quickly perform tasks once they have learned the design. Measures include time to complete a task or relative efficiency compared with an expert user.
- **Learnability**: Assesses **how easy it is for users to accomplish basic tasks the first time they encounter the design**. An intuitive design with clear navigation contributes to learnability. A key question is how long it takes for users to learn how to use the actions relevant to a set of tasks. Measures include **time to learn criterion** or **time to learn specific functions**.
- **Memorability** (Retention over time): How easily users can re-establish proficiency when they return to the design after a period of not using it. This is often closely linked to learnability, and frequency of use plays a role. Measures include assessing knowledge retention after various periods.
- **Safety**: Addresses **how well the design prevents errors and helps with recovery** from those that do occur. Error-resistant design through logical layouts and well-defined actions minimizes the likelihood of errors.
- **Utility**: Refers to the **system or product's functionality** – does it do what people want it to do and have good utility. Usability cannot be looked at independently of utility.
- **Satisfaction**: Concerns the **comfort**, **acceptability**, and **pleasantness** of the work system to its users. This can be ascertained through interviews or written surveys. Measures include **rating scales for satisfaction, ease of learning, or error handling**. For products whose use is discretionary (like websites), a difficult interface can cause users to stop using it, highlighting the importance of satisfaction. The ISO 9241 standard focuses on effectiveness, efficiency, and satisfaction as admirable goals.

These goals are often operationalized as questions or turned into **usability criteria**, which are specific objectives that allow the usability of a product to be assessed, often in terms of improving user performance.

Usability Measures

Usability measures (or metrics) are used to **quantify the level of usability achieved**. These measures are essential for empirically evaluating designs with actual users, setting explicit acceptance criteria, tracking progress, and comparing design alternatives.

Common usability measures include:

- **Time to learn**: How long it takes for users to learn specific functions or tasks.
- **Time to complete a task** (Speed of task performance): The speed and accuracy with which users can complete their tasks.
- **Rate of errors by users**: How many errors users make, how severe they are, and how easily they can recover. Measures can include percentage of errors or time spent on errors. Error tolerance is a dimension of usability.
- **User retention of commands over time** (Retention or Recall): How well users maintain their knowledge after a period.
- **Subjective user satisfaction**: How much users liked using the interface. This can be measured through surveys, questionnaires, interviews, or rating scales. Surveys are a familiar, inexpensive, and acceptable companion for usability tests, providing subjective feedback.
- **Task Success Rate**: The percentage of tasks completed successfully in usability tests or real-world usage. Benchmarking task success rates helps track improvements across design iterations.
- **Frequency of help or documentation use**: How often users resort to help systems or manuals.
- **Percentage of favorable to unfavorable user comments**: Assessing user feedback polarity.
- **Percentage of tasks completed per unit time** (speed metric).
- **Ratio of successes to failures**.
- **Number of commands used**.
- **Time spent using help or documentation**.

These measures can be specified in absolute terms (e.g., Task A performed in 12 minutes with no errors by a first-time user after 30 minutes training) or relative terms (e.g., Task B performed 50 percent faster than with the previous system). Usability specifications, which include these quantifiable measures, are a key characteristic of **usability engineering**.

UNIVERSAL USABILITY

Universal usability is a **design philosophy** and process focused on creating products, services, or technologies that are accessible and usable by the widest possible range of people, regardless of their abilities, disabilities, age, gender, or cultural background. It represents a shift towards focusing on user needs rather than just technology capabilities. Achieving universal usability is considered an ultimate goal and a key quality feature.

The primary goal is to create inclusive and user-friendly experiences that cater to a diverse range of users. This approach aims to expand market share, support required government services, and enable creative participation by the broadest possible set of users. Progress towards universal usability can be measured by the steadily increasing percentage of the population that has convenient, low-cost access to computer and Internet services.

Key aspects and challenges of universal usability include designing for user diversity:

- **Diverse Abilities**: This encompasses physical, cognitive, and perceptual differences. Designers must consider users with a wide range of abilities, including those with visual, auditory, motor, or cognitive impairments. This includes designing for users with disabilities.
- **Age Diversity**: Designing for different age groups, such as children and older adults, is essential.
- **Cultural and International Diversity**: User interfaces must accommodate different languages, cultures, and conventions, including text direction, date/time formats, numeric/currency formats, names, addresses, capitalisation, punctuation, sorting sequences, icons, buttons, colours, and grammar. Internationalisation, which allows for the customisation of local versions of interfaces, offers a competitive advantage.
- **Usage Profiles and Skill Levels**: Designs should cater to different user skill levels, such as novice, intermittent, and expert users. Multi-layer designs, which allow users to learn and use functions progressively based on their tasks and knowledge, are seen as promising for promoting universal usability.
- **Hardware and Software Diversity**: Designing for universal usability also involves accommodating a wide range of hardware (e.g., different display sizes, resolutions, input devices) and software platforms. This includes supporting older devices, newer mobile devices with small screens or low bandwidth, and different operating systems or browsers.

Accommodating diversity often results in a better product for all users. Measures taken to address the special needs of one group can benefit many others. For example, curb cuts in sidewalks, designed for wheelchair users, also help parents with strollers, travelers with wheeled luggage, and delivery people. Designing a system usable by someone deaf or hard of hearing benefits those in noisy environments. Rethinking interfaces for different situations often leads to better design overall.

Key principles supporting universal usability:

- **Inclusivity**: Designs should consider the needs and preferences of all users, including those with disabilities. This may involve providing alternative interaction methods, such as text alternatives for images, captions for videos, and keyboard navigation.
- **Flexibility in Use**: The design should allow for a range of ability and preference through choice of methods and adaptability to the user's pace, precision, and custom. Flexibility allows users to interact in terms of their own needs and preferences. Providing multiple ways to access functions and perform tasks contributes to flexibility.
- **Simple and Intuitive Use**: The system should be easy to use regardless of the user's knowledge, experience, language, or concentration level. This involves supporting user expectations, accommodating different language/literacy skills, avoiding unnecessary complexity, and providing clear prompting and feedback.
- **Understandability**: Clear communication, documentation, instructions, tooltips, and help documentation should be easy to comprehend.
- **Compatibility**: Ensure compatibility with a variety of devices and assistive technologies. This includes considering different screen sizes, input methods, and technologies users might employ. Assistive technologies are crucial for users with disabilities.
- **Equitable Use**: The design should be useful to people with diverse abilities and appealing to all, ideally providing the same access for everyone, or equivalent use where identical access is not possible. No user should be excluded or stigmatised.

- **Tolerance for Error**: The design should minimise errors and help users recover from those that occur. This aligns with the usability goal of safety.

These universal design principles are closely related to general user-centered design rules and principles. **Accessibility**, specifically designing systems usable by people of diverse abilities without special modification, is a key component of universal design. This includes making designs perceptible (regardless of sensory abilities), operable (regardless of physical abilities), simple (regardless of experience or literacy), and forgiving (minimising errors and aiding recovery).

Universal usability is linked to overall **usability**, which involves effectiveness, efficiency, safety, utility, learnability, and memorability. While universal usability aims for the broadest reach, usability focuses on how easy and effective a system is for its *specified* users to achieve *specific* goals. Both usability and utility (functionality) are equally important for a product's success. Designing for universal usability contributes to achieving high-quality user experiences.

DIVERSE COGNITIVE AND PERCEPTUAL ABILITIES

Diverse cognitive and perceptual abilities refer to the wide range of **ways individuals process information, perceive the world, and interact with their surroundings**. This diversity in human abilities, backgrounds, motivations, personalities, cultures, and work styles poses a significant challenge to interface designers. Understanding these differences is considered vital for expanding market share, supporting required government services, and enabling creative participation by the broadest possible set of users. Accommodating this diversity often results in a better product for all users.

Understanding the cognitive and perceptual abilities of users is a vital foundation for interactive-system designers. Humans possess remarkable perceptual abilities to interpret sensory input rapidly and initiate complex actions, which makes modern computer systems possible. Users can recognise slight changes on their displays in milliseconds and begin issuing commands. They also have remarkable perceptual abilities for visual information, able to scan, recognise, and recall images rapidly, and detect subtle changes in size, colour, shape, movement, or texture. However, human capacity to process information is limited.

Key areas of human cognitive and perceptual abilities relevant to design:

- **Cognitive Processes:** *Ergonomics Abstracts* offers a classification of human cognitive processes, including:
    - **Short-term and working memory**. This involves the **temporary storage** and manipulation of information. It is central to problem solving and plays a key role in understanding productivity. Memory is not the most stable of human attributes. The classic memory study indicated a limit of 7 ± 2 "chunks" of information in short-term memory. Working memory is positively related to increased reading comprehension, drawing inferences, reasoning skill, and learning technical information. Design should aim to minimise the memory load on the user. Providing external representations and tools can greatly extend and support people's ability to carry out

cognitive activities by reducing memory load. Recognition memory is considered more powerful than recall, and continuous visibility of objects encourages its use.

- **Long-term and semantic memory**. This pertains to the storage of information for the long term.
- **Problem solving and reasoning**. Humans are able to use information to reason and solve problems, even when information is partial or unavailable. Problem solving is the process of finding a solution to an unfamiliar task using existing knowledge, often involving adaptability and creativity. Designs should aim to free users' cognitive resources to work on task-domain actions rather than making them concentrate on interface-domain actions. Reasoning involves working through different scenarios and deciding on the best option or solution.
- **Decision making and risk assessment.** This involves making choices and evaluating potential risks. Simple and memorable functions can support rapid decision-making.
- **Language communication and comprehension**. This encompasses the use and understanding of language. Reading, speaking, and listening are three forms of language processing with both similar (meaning) and different properties (ease, mode).
- **Search, imagery, and sensory memory**. These include processes related to searching for information, mental imagery, and sensory memory.
- **Learning, skill development, knowledge acquisition, and concept attainment**. Users come with different usage profiles and skill levels, such as novice, intermittent, and expert. Background experience and knowledge in the task domain and the interface domain play key roles in learning and performance. Predicting performance on complex cognitive tasks is difficult because of the many strategies that might be employed, and the ratio for times to perform complex tasks between novices and experts can be as high as 100 to 1. Designs can accommodate learning by providing features for novices (explanations) and experts (shortcuts, faster pacing).
- **Executive Functions**: Higher-order cognitive processes responsible for planning, organizing, initiating, and regulating goal-directed behaviour.

- **Perceptual Abilities:** These relate to how individuals interpret and make sense of sensory information from their environment. Vision is the most dominant sense for sighted individuals, followed by hearing and touch. Key aspects include:
  - **Vision:** This is especially important and has been thoroughly studied. Designers need to be aware of the ranges of human perceptual abilities with regard to vision. This involves considering response time to visual stimuli, adaptation to light, identifying objects in context, determining velocity or direction of movement. The visual system responds differently to various colours, and some people are colour-deficient. People's spectral range and sensitivity vary, and peripheral vision is quite different from perception in the fovea. Flicker, contrast, motion sensitivity, depth perception, glare, and visual fatigue must also be considered. Perception is influenced by experience, expectancies, context, and the ability to distinguish signals from noise. **Designs need to present information in a way that can be readily perceived**. This involves making icons and graphical representations distinguishable, using visual grouping methods, ensuring audio sounds are distinguishable, and researching proper colour contrast techniques. Visual distinctiveness of a screen is a strong contributor to rapid lower-level information processing.

    - **Hearing and Touch:** Information is also received through auditory and haptic channels. Considering these senses for richer interaction and redundancy can be beneficial.
    - **Other Senses:** People also acquire information via taste, smell, and kinesthesia (awareness of body position/movement).
- **Modes of Cognition:** Norman distinguishes two general modes:
    - **Experiential cognition:** A state of mind where people perceive, act, and react effectively and effortlessly, requiring expertise and engagement (e.g., driving, reading a book).
    - **Reflective cognition:** Involves thinking, comparing, and decision-making, leading to new ideas and creativity (e.g., designing, learning, writing).
- **Cognitive Style:** People differ in how they think and solve problems, such as being better at verbal vs. spatial thinking, analytic vs. intuitive, or concrete vs. abstract. This might influence their preference for different interface styles.

Factors Affecting Perceptual and Motor Performance also contribute to this diversity:

- Arousal and vigilance
- Fatigue and sleep deprivation
- Perceptual (mental) load
- Knowledge of results and feedback
- Monotony and boredom
- Sensory deprivation
- Nutrition and diet
- Fear, anxiety, mood, and emotion
- Drugs, smoking, and alcohol
- Physiological rhythms Environmental factors such as noise, lighting, movement, and vibration also impact performance and user experience.

Understanding these diverse abilities and limitations is central to designing effective interactive systems. Various **cognitive frameworks and theories** have been developed to explain and predict user behaviour based on theories of cognition:

- **Human Information Processing** models view the mind as a processor with ordered stages and processes acting upon mental representations.
- **Distributed Cognition** suggests that thinking is not just within the user's head but distributed in their environment, involving external things (paper documents, computers, artifacts) and other people.
- **External Cognition** is concerned with the cognitive processes involved when people interact with or create external representations (maps, diagrams, notes) and use tools to aid cognition, extending and supporting their abilities.
- **Mental Models** provide a way of conceptualising the user's understanding of the system.
- Other models like the **Model Human Processor (MHP)**, **GOMS**, **CCT**, and **KLM** provide frameworks for understanding human capabilities and predicting performance, while **Interacting Cognitive Subsystems (ICS)** models perception, cognition, and action.

Designing for diverse cognitive and perceptual abilities involves:

- Accommodating variations in physical abilities and workplaces.

- Addressing users with disabilities, which can include cognitive or sensory impairments. Multi-modal systems can provide access through different channels, benefiting users with sensory, physical, or cognitive impairments. Providing alternative methods, like auditory navigation or tactile diagrams, supports blind users.
- Designing for older adult users and children, who may have different cognitive and perceptual characteristics.
- Catering to personality differences.
- Considering cultural and international diversity, as this impacts scanning patterns and preferences.
- Accommodating hardware and software diversity, which affects how information is presented and interacted with.

Designing for universal usability, which includes accommodating these diverse abilities, helps bring the benefits of technology to the widest possible set of users. It is suggested that designs facilitating multiple natural-language versions or accommodating disabled users often result in interfaces that are better for all users.

PERSONALITY DIFFERENCES

- Users vary in their attitudes towards computers; some dislike or are anxious about them, while others are attracted to them
- Even users who enjoy computers have different preferences for interaction styles, such as pace, graphics versus tabular data, dense versus sparse presentation, and step-by-step versus all-at-once work. These differences are significant.
- A clear understanding of personality and cognitive styles can be helpful for designing interfaces for a specific user community.
- However, there is no simple taxonomy of user personality types.
- One technique, though controversial, is the **Myers-Briggs Type Indicator** (MBTI), based on **Carl Jung**'s theories. Jung proposed four dichotomies: **Extroversion versus Introversion, Sensing versus Intuition, Perceptive versus Judging, and Feeling versus Thinking**. The MBTI relates professions and personality types and has guided designers, but the linkage between personality types and specific interface features is weak.
- Successors to the MBTI include the Big Five Test, based on the OCEAN model: **Openness** to Experience/Intellect, **Conscientiousness, Extraversion, Agreeableness**, and **Neuroticism**.
- Hundreds of other psychological scales exist, such as risk taking versus risk avoidance, internal versus external locus of control, reflective versus impulsive behaviour, convergent versus divergent thinking, and tolerance for ambiguity.
- Another way to assess personality is by studying user behaviour, such as how users organize emails (e.g., in folders versus using search). These distinct approaches may relate to personality variables, indicating that designs must satisfy multiple requirements.
- Designers creating applications for areas like home, education, art, music, and entertainment may benefit from paying more attention to personality types. Consumer-oriented researchers are particularly aware of personality distinctions across market segments.
- Individual personality differences are listed as a primary factor influencing user expectations and attitudes regarding system response time. They are also large, and users are adaptive, changing strategies based on response times. It may be useful to let users set their own interaction pace.

CULTURAL AND INTERNATIONAL DIVERSITY

- Differences in cultural, ethnic, racial, or linguistic backgrounds affect how users scan screens (e.g., Japanese/Chinese versus English/French).
- Users from reflective/traditional cultures may prefer stable displays with single item selection, while those from action-oriented/novelty-based cultures might prefer animated screens and multiple clicks.
- Preferred web page content varies culturally; some emphasize buildings/professors, others students/social life. Mobile device preferences also differ across cultures.
- The growth of a worldwide market necessitates designing for internationalization. Software architectures should facilitate customization of local versions by storing text externally for easy translation.
- Hardware concerns include character sets, keyboards, and special input devices.
- User interface design considerations for internationalization are extensive, including characters, numerals, dates, times, numeric and currency formats, weights and measures, telephone numbers, addresses, names, titles, capitalization, punctuation, sorting, icons, buttons, colors, pluralization, grammar, spelling, etiquette, policies, tone, formality, and metaphors. The list is long and incomplete.
- Effective localization provides a strong advantage in a competitive atmosphere. Usability studies with users from different countries, cultures, and language communities are important for effective designs.
- The role of IT in international development is growing, but accommodating diverse needs in language skills and technology access requires more effort.
- Cultural differences can include expectations about conversation pauses, who speaks when, holiday schedules, and work hours.
- Designing for a particular culture or subculture is important, especially for websites. Cultural difference is not just national difference, but can include age, gender, race, sexuality, class, religion, and political persuasion.
- Gesture use varies culturally; for example, shaking one's head can mean agreement in India, opposite to European interpretation. This is significant for interactions incorporating gestures.
- Personal space differs across cultures, which can cause problems in cross-cultural meetings.
- Localization involves infusing a specific cultural context into an internationalized product. Cultural differences are immense.
- Hofstede's dimensions (power distance, individualism, masculinity, uncertainty avoidance, long-term orientation) suggest implications for design variations across cultures. For instance, in high power distance countries, navigation might be more direct and information more structured. Collectivists may prefer national or global brands. Masculine cultures may respond better to competitive interactions.
- Guidelines for writing text internationally include using simple English, avoiding jargon, slang, clichés, and culturally specific examples. Using images and symbols that are internationally understood is also crucial.
- Colors have different meanings across cultures and disciplines.
- Testing a product translated for a new culture is essential, as it is effectively a new product. International requirements should be established early.

USERS WITH DISABILITY

- Accommodating diverse human abilities is a challenge for designers. Understanding physical, intellectual, and personality differences is vital for expanding market share and supporting government services.
- When digital content is flexibly presented in different formats, all users benefit, but this is most appreciated by users with disabilities.
- Users with disabilities can access content using diverse input and output devices. Examples include screen readers and refreshable braille displays for blind users, magnification for low-vision users, captioning/transcripts for hearing impairments, and speech recognition/alternative input for limited dexterity. Some alternate input/output methods are integrated into technology.
- Applications for users with disabilities have enabled paralyzed, bedridden, or injured people to broaden their horizons and control equipment or computers.
- Disability can result from poor interaction design that necessitates interaction types impossible for someone with an impairment. Accessibility makes experiences usable by all. Technologies now mainstream often originated as accessibility solutions (e.g., SMS for the hearing-impaired). Designing for accessibility inherently results in inclusive design for all.
- Impairments can be classified by type: Sensory (vision, hearing), Physical (motor function loss), and Cognitive (learning, memory loss). Within each type, there is a complex mix of capabilities requiring different design approaches (e.g., colour blindness vs. complete blindness).
- Impairments can also be Permanent, Temporary, or Situational (e.g., noise affecting hearing).
- The number of people with permanent disabilities increases with age.
- Design objectives for accessibility include providing easy access for people with disabilities.
- Types of disabilities include visual, hearing, physical movement, speech or language impairments (like dyslexia), cognitive disorders, and seizure disorders. Approximately 8% of Web users have a disability making traditional use difficult.
- Visual disabilities range from reduced acuity to total blindness; hearing from inability to detect sounds to total deafness. Physical movement disabilities include difficulties with mouse or keyboard use. Cognitive disabilities include memory and perceptual problems. Seizure disorders involve sensitivity to visual flash rates.
- Users relying on accessibility utilities (like screen magnifiers) should be considered in color choices, as these utilities can alter displayed colors.
- Users with disabilities should be included in usability testing.

OLDER ADULT USERS AND CHILDREN

Older Adult Users

- Applications for older adults might increase independence, restore lost skills, and regain confidence.
- The proportion of older people in the population is growing, representing a major and growing market.
- There is no evidence older people are averse to new technologies, though some may lack familiarity and fear learning. They may find terminology alien.
- The proportion of disabilities increases with age; more than half of people over 65 have some disability.

- Technology can support age-related failing vision, hearing, speech, and mobility. Communication tools like email and instant messaging can provide social interaction. Mobile technology can aid memory loss
- Designers can do much to accommodate older adult users, improving user experiences and bringing societal advantages (e.g., e-mail access to grandparents).
- Basic universal design principles are important for older adults, including redundancy, supporting access technologies, clear/simple designs, and being forgiving of errors. Sympathetic and relevant training is also needed.
- Older adults may appreciate being able to slow the pace of interaction.
- Compared to younger adults, older adults are often more educated and have higher vocabulary scores, but may have more difficulty recalling information, problems with multi-step tasks, and prefer scrolling page by page.
- Age classifications for usability research have been inconsistent, hindering guideline development.
- Older adults should be included in usability testing.

Children

- Children have different needs and expectations from adults; interactive quizzes and cartoon characters may be motivating for them but annoying for adults. Adults may prefer talking-heads discussions which children find boring.
- Just as physical objects are designed differently for various age groups, interactive products must match the needs of children.
- Ideas about participatory design are being refined with diverse users, including children. Levels of participation for children in interface development include testers, informants, and partners. Design partners are active team members, potentially forming intergenerational teams.
- Questionnaires for children must use age-appropriate language.
- Smileyometer gauges are an example of adapting data collection techniques for early readers (children).
- Children should be included in usability testing

GUIDELINES

What they are: Low-level focused advice about good practices and cautions against dangers. They prescribe cures, caution against dangers, and provide helpful reminders based on accumulated wisdom. They are narrowly focused and often more technology-oriented than principles. Guidelines are sometimes referred to as **rules**, especially when they are more specific and should be followed.

Why they are used: To record insights and guide the efforts of future designers. They help in developing a shared language and promoting consistency among multiple designers in terminology, appearance, and action sequences. They record best practices derived from practical experience or empirical studies. Guidelines documents engage the design community in discussions about interface issues. Applying guidelines can help clean up cluttered displays, inconsistent layouts, and unnecessary text. They provide designers with the benefit of experience.

Examples: Guidelines exist for navigating interfaces, organizing displays, getting user attention, and facilitating data entry. Specific examples include standardizing task sequences, ensuring descriptive embedded links, using unique headings, using check boxes for binary choices, developing pages that print properly, and using thumbnail images for previews. Guidelines also cover aspects like using color, writing text and messages, designing online help, organizing window and page layouts, and using controls. There are specific guidelines for web pages and weblogs.

Development & Process: Guidelines are created by designers. Early guidelines from Apple and Microsoft were influential. The creation of guidelines documents is often a social process within an organization to build support. Effective guidelines documents should be living texts, adapted to changing needs and refined through experience. Procedures are needed for distribution, enforcement, exemption, and enhancement. This is sometimes referred to as the "four Es": **Education, Enforcement, Exemption, and Enhancement**. Guidelines can exist at multiple levels, such as rigid standards, accepted practices, and flexible guidelines.

Limitations: Controversies over guidelines are lively. Guidelines documents require management processes for education, enforcement, exemption, and enhancement. They tend to be lower in authority and more general than standards. It's possible that applying guidelines without understanding their underlying assumptions could prevent the best design choice.

PRINCIPLES AND THEORIES

Principles

What they are: Middle-level strategies or rules that help in analyzing and comparing design alternatives. They are more fundamental, widely applicable, and enduring than guidelines. Principles are abstract design rules with high generality but lower authority compared to standards. They are derived from psychological, computational, and sociological knowledge and are largely independent of technology. They are less abstract than theories but more so than guidelines. When used in practice, especially for evaluation, they are commonly referred to as **heuristics**. Usability principles tend to be more prescriptive than design principles and are often used for evaluating designs.

Why they are used: They provide a good starting point for designers of various interactive systems. Principles like recognizing user diversity make sense to every designer, although they must be thoughtfully interpreted for different users and situations. Principles help designers cope with varying user skill levels, task profiles, and interaction styles. They are crucial for designing systems that prevent errors. Principles can increase user productivity by promoting simplified data-entry procedures, comprehensible displays, and rapid informative feedback. They are important for universal usability strategies. They provide a framework for heuristic evaluation. Principles capture the essential common properties of good design.

Examples: The Eight Golden Rules of Interface Design are a well-known set of principles, which include striving for consistency, enabling shortcuts for frequent users, offering informative feedback, designing for closure, preventing errors, permitting easy reversal, keeping users in control, and reducing short-term memory load. Other examples include principles of direct manipulation (continuous representation, rapid feedback, physical actions replacing syntax), Norman's principles of good design (visibility, conceptual model, mappings, feedback), universal design principles for older adults (redundancy, supporting access technologies, simplicity, forgiveness of errors). Principles of graphic design (elegance/simplicity, scale/contrast/proportion, organisation/visual structure). Data ethics principles like Fairness, Accountability, Transparency, and Explainability (FATE).

Limitations: Principles often require clarification and must be interpreted, refined, and extended for each specific environment. Abstract principles are less useful for specific design advice compared to guidelines.

Theories

What they are: High-level, widely applicable frameworks for human-computer interaction. They are intended to go beyond the specifics of guidelines and the breadth of principles to develop tested, reliable, and broadly useful explanations or predictions. Theories are an abstraction of reality and thus must be incomplete. They can be descriptive, explanatory, prescriptive, or predictive.

Why they are used: Designers draw upon theories during design and evaluation. Theories are helpful in developing consistent terminology and taxonomies for objects and actions, which supports communication and teaching. Explanatory theories describe sequences of events and cause-and-effect relationships. Predictive theories enable designers to compare proposed designs based on quantitative measures like execution time, error rates, conversion rates, or trust levels. They guide researchers in understanding relationships between concepts and generalizing results, and help practitioners make design trade-offs. Theories underpin predictive evaluation. They can explain user interaction, inform design, and predict user performance.

Types & Examples: Theories can be grouped by the types of skills involved, such as motor, perceptual, or cognitive. Predictive theories include models for motor-task performance like Fitts's Law (for pointing times), keystroke-level models, and information-foraging theory (for predicting user success on websites). Explanatory theories include stage-of-action models like Norman's seven stages and models describing how cognition is socially distributed (distributed cognition). Other theories mentioned include the Goals, Operators, Methods, and Selection rules (GOMS) model, object-action interface models, information-architecture theories, theories of small-group behaviour and sociology, activity theory, external cognition, and embodied interaction. Theories can also be classified as Micro-HCI (focus on measurable lab performance) or Macro-HCI (focus on contextual factors like emotion and social use).

Development: Developing tested, reliable, and broadly useful theories is a goal for the discipline of human-computer interaction. Theories emerge from accumulated experience and research. Building theories can involve alternating data collection and analysis (grounded theory).

Limitations: Many theories are needed for a large topic like user-interface design. It can be difficult to come up with effective theories. Critics argue that theories should be more central to research and practice, noting that it's harder to demonstrate the practical impact of explanatory theories compared to focused predictive ones.