Module -4
Cognitive Systems and Evaluation of HCI

SYLLABUS

Cognitive Models- Goal and task hierarchies, GOMS Model. Introducing Evaluation- Types of Evaluation, Other Issues to Consider When Doing Evaluation. Conducting Experiments. Usability testing – Heuristic evaluation and walkthroughs, Analytics and predictive models.

COGNITIVE MODELS

Cognitive models in Human-Computer Interaction (HCI) are part of the high-level theories and models used by successful interactive system designers to go beyond intuitive judgments. They aim to represent aspects of a user's understanding, knowledge, intentions, or processing as they interact with a system. The goal of developing these theories and models is to describe objects and actions with consistent terminology, support communication and teaching, and in some cases, make predictions.

Cognitive models can be categorized in several ways. One grouping is based on the human capacities involved, such as **motor**, **perceptual**, or **cognitive** aspects. Cognitive theories, involving memory (short-term, working, long-term), are central to problem solving. Another approach groups cognitive models by their focus, such as design by levels, stages of action (like Norman's seven stages), consistency, contextual awareness, and evolutionary dynamics. Cognitive architectures underlie these models, providing foundational assumptions about how the user performs cognitive processing.

Predicting performance on complex cognitive tasks is particularly difficult due to the many strategies users might employ and opportunities for error. Traditional psychological theories need to be extended and refined to accommodate the complex human learning, memory, and problem solving required in user interfaces.

GOAL AND TASK HIERARCHIES

Goal and Task Hierarchies represent a key category of cognitive models. They are based on the premise that **users achieve goals by breaking them down into smaller subgoals** in a divide-and-conquer fashion. This decomposition results in a hierarchy of goals and subgoals.

These methods generally operate at a relatively **low level**, focusing on routine learned behavior rather than complex problem solving. The most abstract task that doesn't require problem-solving is sometimes referred to as the unit task.

Issues in goal hierarchy methods include representing the multiple ways a user might achieve a goal and how to treat user errors. Prediction of error behavior is generally poor among these techniques, although some, like Cognitive Complexity Theory (CCT), can represent error behavior. Goal hierarchies face challenges with more complex tasks and display-based systems that encourage exploratory interaction rather than strictly planned sequences.

Goal-oriented cognitive models are closely related to task analysis techniques, such as Hierarchical Task Analysis (HTA). Both involve decomposing tasks into hierarchical structures. However, their purpose and granularity differ: goal-oriented cognitive models aim to understand the user's internal cognitive processes at a finer level, while task analysis focuses more on observable behavior from an external perspective, including actions not involving a computer. Task analysis is typically used earlier in the design lifecycle (requirements capture), whereas cognitive models are often used later (evaluation).

GOMS MODEL

The GOMS model, developed by Card, Moran, and Newell, is a **prominent model within the goal and task hierarchy category**. GOMS is a generic term for a family of models that vary in granularity. It was an attempt to model the knowledge and cognitive processes involved in user interaction.

A GOMS description is composed of four elements:

1. **Goals:** Describe what the user wants to achieve. In GOMS, goals represent a 'memory point' for the user, aiding evaluation and error recovery.
2. **Operators:** The lowest level of analysis, representing basic user actions. These can be perceptual, motor, or cognitive acts necessary to change the user's mental state or affect the task environment. The granularity of operators is flexible.
3. **Methods:** Sequences of operators and subgoals used to achieve a goal. There are typically several ways (methods) a goal can be broken down.
4. **Selection Rules:** The means for choosing between competing methods. These rules typically depend on the user, the system state, and goal details. Selection rules can be tested against user traces.

The goal hierarchies described in a GOMS analysis are often **below the level of the unit task**. A typical GOMS analysis decomposes a high-level goal into a sequence of unit tasks, which are further decomposed down to basic operators. The original method largely side-stepped the complex problem-solving and domain understanding needed for decomposition above unit tasks. While the general notation could describe higher-level subgoal structures (similar to task analysis), the central feature for GOMS is the goal hierarchy itself, which is then analyzed for complexity, learnability, etc..

GOMS analysis can yield measures of performance. The stacking depth of a goal structure can estimate short-term memory requirements. GOMS is useful for describing how experts perform routine tasks. Coupled with physical device models like the **Keystroke-Level Model (KLM)**, a simplified version of GOMS, it can predict user performance in terms of execution times for error-free expert performance. KLM focuses on unit tasks and sums the times for basic operations like keystroking, pointing, homing, drawing, thinking, and waiting for the system.

Despite its utility, GOMS models have faced criticism. They concentrate on **expert users and error-free performance**, with insufficient emphasis on learning, problem solving, error handling, subjective satisfaction, and retention. Additionally, GOMS analysis risks being defined by the computer dialog rather than reflecting a user's natural hierarchy. While implemented in more complex cognitive architectures like Soar and ACT-RPM, these models still largely inherit the limitation of predicting only skilled user execution time for familiar tasks. GOMS tools exist to facilitate modeling.

EVALUATION

Evaluation is a critical part of the design process for interactive systems. It is the process of **assessing designs and testing systems** to ensure they behave as expected and meet user requirements. Evaluation aims to determine the **usability and acceptability** of a product or design, measured against criteria like the number of user errors, how appealing it is, and how well it matches requirements. The goals of evaluation include assessing the extent and accessibility of the system's functionality, assessing users' experience of the interaction, and identifying any specific problems with the system. Evaluation supports the design of usable interactive systems.

Why Evaluation is Important

Extensive testing and evaluation are necessities for designers. Evaluation helps to ensure that designs meet users' needs. Evaluating the design before implementation can help **avoid expensive mistakes**, as errors discovered later are more costly to fix. It allows designers to **assess whether a system adheres to usability principles** and can **support communication and teaching** about user interfaces. Evaluation allows for the **comparison of design alternatives** and can provide predictions, such as speed or error rates. Evaluation also helps to **identify usability problems**.

When to Evaluate

Evaluation should take place **throughout the design process**, not just at the end. It should occur at different stages, including early, middle, and late phases of design. Ideally, the first evaluation should be performed **before any implementation work has started** to evaluate the design itself. Evaluation continues through prototype development and can be done when the system is complete or in active use.

Evaluations done during design to check if the product meets users' needs are known as **formative evaluations**. Evaluations performed at the end to verify if the product is good enough, perhaps to satisfy a sponsoring agency or check adherence to a standard, are called **summative evaluations**. Iterative cycles of design and evaluation are essential for ensuring products meet user needs.

TYPES OF EVALUATION

Evaluation techniques can be broadly categorized based on whether they involve expert analysis or user participation. Another way to categorize evaluation is by paradigms: "quick and dirty," usability testing, field studies, and predictive evaluation.

1. **Evaluation through Expert Analysis (Inspections)**: These methods rely on designers or human factors experts assessing the design's impact on a typical user. They are often relatively cheap as they don't require user involvement.
    - **Cognitive Walkthrough:** Evaluators step through the sequence of actions required for a user to accomplish a task, focusing on how easy the system is to learn through

exploration. This approach attempts to introduce psychological theory into a walkthrough technique.

  - **Heuristic Evaluation:** Several evaluators independently critique a system based on a set of usability principles (heuristics) to identify potential problems. Between three and five evaluators are often sufficient.
  - **Model-Based Evaluation:** Certain cognitive and design models can be used to predict user performance or evaluate design options. Examples include GOMS and the Keystroke-Level Model (KLM). Dialog models like state transition networks can evaluate dialog sequences.
  - **Review Methods:** Experts review software based on guidelines.
  - **Analytic Methods:** These are expert-based approaches.
2. **Evaluation through User Participation**: These techniques involve testing with actual users for whom the system is intended.

  - **Usability Testing:** Involves measuring how well a design supports users' performance on various tasks. It's done in controlled settings, often usability labs, with users carrying out set tasks. Observation of user activity is a technique used.
  - **Controlled Psychologically Oriented Experiments:** Users are taken out of their normal environment for controlled tests. These involve manipulating independent variables (like interface style or number of menu items) and measuring dependent variables (like speed or errors) to test hypotheses. They provide objective measures.
  - **Observational Methods:** Involve observing users in their natural environment (field studies) or controlled settings (laboratory). Evaluators can be onlookers, participant observers, or ethnographers.
  - **Query Techniques:** Involve collecting data by asking users.
    - **Interviews:** Can be structured, semi-structured, or unstructured ("conversation with a purpose"). Used to collect users' opinions or gain detailed feedback.
    - **Questionnaires (Surveys):** A well-established technique for collecting demographic data and users' opinions. Can use closed or open questions. Likert scales and semantic differential scales are examples used.
  - **Field Studies:** Evaluation done in a natural environment like a home or workplace. There is often little or no control over user activities to see how the product is used in the real world. Ethnographic studies are a type of field study.
  - **Laboratory Studies:** Evaluation conducted in a controlled environment.
3. **Other Evaluation Techniques:**
  - **Physiological Monitoring:** Measures physical changes like eye tracking, heart rate, and skin conductance.
  - **Analytics:** Collecting data about user interactions to identify usage patterns, often for existing products like websites.
  - **A/B Testing:** Comparing two or more design alternatives by giving different versions to different user sets and measuring performance against success criteria.
  - **Predictive Models:** Provide an approximation of user behavior, often used as a quick way to assess design potential. Examples include the GOMS model and Fitts' Law.

ISSUES TO CONSIDER WHEN DOING EVALUATION

Choosing an appropriate evaluation method requires careful consideration, as each method has strengths and weaknesses. Several factors distinguish evaluation techniques and influence selection. The DECIDE framework provides a checklist for planning evaluations.

1. **Determine the Overall Goals:** Clarifying the high-level goals is the first step. What does the evaluation need to achieve, and for whom?. Goals drive the evaluation.
2. **Explore Specific Questions:** Break down goals into specific questions to be answered. These can be further decomposed into finer-grained sub-questions. Good questions focus the evaluation.
3. **Choose Evaluation Paradigm and Techniques:** Select the appropriate approach and methods based on the goals and questions. Practical and ethical issues must also be considered, often requiring trade-offs. **Combining techniques (triangulation)** can provide different perspectives and a broader picture of usability.
4. **Identify Practical Issues:** Consider constraints such as:
    - **Users/Participants:** Identifying typical users and sample tasks. How many users are needed?. Recruiting representative participants. Dealing with limited access to users.
    - **Facilities and Equipment:** Where the evaluation will take place (laboratory vs. field). Availability and setup of necessary equipment (cameras, etc.).
    - **Schedules and Budgets:** Time available. Costs of product and finances allocated for testing. Short schedules and low budgets are routine practical issues.
    - **Evaluators' Expertise:** The experience of the design and evaluation team. Some techniques require higher expertise.
    - **Environment:** The environment where the interface is used.
5. **Decide How to Deal with Ethical Issues:** Ensure high ethical standards are maintained. Key considerations include:
    - **Informed Consent:** Participants must be informed about what they will do, how data will be used, and their rights.
    - **Privacy and Confidentiality:** Protecting personal information.
    - **Not Causing Harm:** Avoiding causing duress or disturbing users. Participants should be able to stop at any time.
6. **Evaluate, Interpret, and Present Data:** Decide what data to collect, how to analyze it, and how to present findings.
    - **Qualitative vs. Quantitative Data:** Determine the type of measures needed. Some techniques provide qualitative information (e.g., cognitive walkthrough, interviews), while others provide quantitative data (e.g., experiments, physiological monitoring). Both objective and subjective approaches should ideally be used. Basic qualitative analysis involves identifying themes or categorizing data.
    - **Reliability:** How consistently the technique produces the same results under the same circumstances.
    - **Validity:** Whether the technique measures what it is intended to measure. Ecological validity concerns how the evaluation environment might distort results compared to real-world use.
    - **Bias:** Avoiding biases that could distort results.

- **Scope:** How generalizable the findings are.

CONDUCTING EXPERIMENTS

Conducting experiments in the field of Human-Computer Interaction (HCI) is a rigorous method used to evaluate interface designs and gain a scientific understanding of user performance and basic design principles. It draws heavily on the classic experimental methods of psychology.

Purpose and Goals

The primary goal of an experiment is to answer a question or test a specific hypothesis to discover new knowledge. This involves manipulating variables to show statistically significant differences between conditions. While usability tests are designed primarily to find flaws in user interfaces, controlled experiments aim to validate or reject a hypothesis and can provide data to support design decisions, especially when comparing alternatives. Experiments provide objective measures of performance, such as speed, error rates, and sometimes subjective measures like user satisfaction.

Basic Outline of the Classic Experimental Method

The classic scientific method applied to interface research involves several key steps:

1. **Understanding a Practical Problem and Related Theory:** Start by identifying a real-world problem and considering the theoretical framework surrounding it.
2. **Lucid Statement of a Testable Hypothesis:** Formulate a clear prediction about the outcome of the experiment, framed in terms of independent and dependent variables. The experiment aims to disprove the null hypothesis, which states there is no difference between conditions.
3. **Manipulation of a Small Number of Independent Variables:** Identify and change one or a small number of variables that are believed to cause a difference in user behavior or performance.
4. **Measurement of Specific Dependent Variables:** Carefully choose and measure the variables that are expected to be affected by the manipulation of the independent variables. Common dependent variables include time to complete a task and the number of errors made.
5. **Careful Selection and Assignment of Participants (Subjects):** Judiciously select users who are representative of the target population and carefully or randomly assign them to different experimental groups or conditions.
6. **Control for Bias:** Implement measures to prevent factors unrelated to the independent variables from influencing the results. This includes controlling for bias in participants, procedures, and materials.
7. **Application of Statistical Tests:** Use appropriate statistical methods to analyze the collected data and determine if the observed differences are statistically significant, meaning they are unlikely to have occurred by chance.
8. **Interpretation of Results, Refinement of Theory, and Guidance:** Analyze the data to interpret the findings, resolve the practical problem, refine the theoretical understanding, and provide guidance for designers and future researchers.

Key Considerations and Practical Issues

- **Variables and Conditions:** Experiments typically involve manipulating independent variables across different conditions (e.g., different interface styles) and measuring the effect on dependent variables (e.g., performance time). Experiments require at least two conditions.
- **Participants:** Careful selection and assignment of participants are crucial. While usability tests might use as few as three participants, controlled experiments often require more for statistical significance.
- **Experimental Design Types:** There are different ways to allocate participants to conditions:
    - **Different Participants Design (Between-Subjects):** Different groups of participants are assigned to different conditions.
    - **Same Participants Design (Within-Subjects):** All participants perform in all conditions. This requires counterbalancing the order of conditions to avoid learning effects or order effects.
    - **Matched Pair Participants:** Participants are matched based on relevant characteristics and then assigned to different conditions.
- **Control and Environment:** Experiments are typically conducted in **controlled settings**, often specialized **usability laboratories** or research labs, to minimize extraneous influences and distractions. This high level of control is necessary to confidently attribute measured effects to the manipulated variables. While field studies offer naturalness, laboratory studies allow for controlled experimentation suitable for testing specific interface features and hypotheses. However, it's noted that sometimes manipulating the context in a laboratory is done deliberately to uncover specific problems or compare alternatives in a controlled setting.
- **Bias:** Controlling for biasing factors is essential to ensure the reliability and validity of the results. Biased samples of respondents can produce erroneous results in surveys, and the principle applies equally to experiments.
- **Data Collection and Analysis:** Data collected in experiments is often quantitative (e.g., time, errors). Statistical tests are applied to analyze this data and determine statistical significance. Careful consideration is needed in choosing the appropriate statistical test, as different tests have different assumptions.
- **Reliability and Validity:** Controlled experimentation yields narrow but reliable results. Reliability concerns the consistency of results upon repetition, while validity concerns whether the experiment measures what it intends to measure. Replication with similar conditions enhances both reliability and validity. Ecological validity, or how well the lab results generalize to real-world use, is a potential concern when experiments are conducted in artificial settings.
- **Tasks:** Participants perform carefully prepared sets of tasks. Deciding on representative tasks and defining what constitutes "better performance" (the dependent variables) is critical.
- **Pilot Studies:** It is recommended to test experimental materials and methods through pilot studies before conducting the full experiment.

Advantages and Disadvantages

- **Advantages:** Experiments provide **objective measures of performance** and can include subjective measures of user satisfaction. They allow for **rigor and confidence** in concluding that one interface feature is better than another based on statistical significance.
- **Disadvantages:** Conducting experiments requires a **rigorously controlled environment**. The experiment conductor must have **expertise** in setting up, running, and analyzing the collected data, including applying statistical methods. Creating **multiple prototypes** representing design alternatives may be necessary.

In summary, conducting controlled experiments in HCI is a powerful approach rooted in the scientific method, used to test specific hypotheses, compare design alternatives objectively, and build a reliable understanding of human performance with interactive systems. While demanding in terms of control and expertise, they provide a level of confidence in the results that complements other evaluation methods.

USABILITY TESTING

U**sability testing** is a fundamental and essential process in Human-Computer Interaction (HCI) used to evaluate interactive products and interfaces. It emerged significantly in the early 1980s, indicating a shift in attention toward user needs and user experience

Purpose and Goals

- The primary purpose of usability testing is to determine whether a product is **usable by the intended user population to achieve their tasks**.
- It aims to find **flaws in user interfaces**.
- It establishes a **communication bridge between developers and users**, allowing developers to learn about user goals, perceptions, questions, and problems, and users to see system capabilities early.
- Testing helps **validate design alternatives** or ensure specific requirements were met.
- While traditional usability metrics focused on effectiveness, efficiency, and satisfaction, usability testing today also considers the **entire user experience**, including aspects like enjoyment and fun.
- For discretionary systems like websites, a difficult-to-use interface found through testing can indicate why people might stop using it. For mandatory business applications, testing can reveal issues leading to lowered worker productivity.
- Testing can provide **supportive confirmation of design progress** and offer **specific recommendations for changes**.

Importance and Benefits

- Usability testing is vital for product acceptance, especially when competitive products offer similar functionality.
- It can not only speed up projects but also produce **dramatic cost savings**.
- Investing in user testing is considered the **only validated methodology** that consistently produces successful results and avoids wasting significant resources.
- It provides direct observation of users doing tasks, analyzing the gap between users' mental models and the system's conceptual model.
- It helps uncover and address usability issues as the design evolves in an iterative process.
- Observing user reactions can help developers understand usability issues that are difficult to glean from reports or presentations.

Methods and Process

- Usability testing typically involves measuring typical users' performance on **carefully prepared tasks** that are representative of those for which the system was designed.

- It often uses a **combination of methods** to collect data in a controlled setting, including observation, user satisfaction questionnaires, and interviews. Key stroke logs and video recordings are common data collection methods. Eye-tracking hardware and software can also be used.
- Participants may be asked to **think aloud** while performing tasks to reveal their thoughts and plans.
- Common quantitative performance measures obtained include:
  - Time to complete a task.
  - Number or percentage of errors made.
  - Percentage of tasks completed (success rate).
  - Time spent on errors.
  - Task completion rate per unit time (speed metric).
  - Navigation path or routes taken through the software.
- Process data, such as observation of participant behavior and self-reports, can provide insight into user expectations and task flow, helping identify gaps between the user's mental model and the system's interaction model.
- User satisfaction is often measured using questionnaires or rating scales. Standard questionnaires like SUS (System Usability Scale) and MPUQ (Mobile Phone Usability Questionnaire) exist.
- A **test plan** is developed, defining the scope and purpose, creating a timetable, outlining methodology, developing scenarios, selecting participants, and identifying the facility.
- A **pilot test** with one to three participants is recommended before the main test to refine procedures, tasks, and questionnaires.
- During the test, it's important to ensure all materials are ready, explain the purpose (testing the software, not the user), explain data usage, get consent if needed, and ensure participant questions are answered.
- After the test, a closing interview or debriefing is valuable to ask follow-up questions and allow participants to provide explanations or ask questions.

Where it is Conducted

- Usability testing is commonly done in **controlled laboratory-like conditions**.
- **Usability laboratories** are often specially designed facilities, sometimes with one-way mirrors and recording equipment, to control the environment and allow observation. Having a physical lab demonstrates an organization's commitment to usability.
- Testing can also take place in **informal controlled environments** like an office or conference room.
- While labs are good at revealing usability problems, they can be poor at capturing the context of use.
- Testing can also occur in **natural settings**, referred to as **field tests** or "in-the-wild" studies. Field tests put new interfaces to work in realistic environments. Portable usability laboratories can support thorough field testing. Field tests are better for understanding natural behavior, long-term use, and integration into users' real-world activities and environments.
- Testing can also be conducted **remotely**, via phone or digital communication.

Participants

- Testing involves **typical users** who are representative of the intended user audience.

- Participants should be carefully selected and screened based on characteristics relevant to the product and tasks, such as experience level, age, gender, and cultural diversity.
- While controlled experiments often require many participants for statistical power, usability tests are designed to find flaws and may use **fewer participants**, sometimes as few as three. Some approaches, like "discount usability testing," recommend three to six participants. However, testing complex systems or anticipating a wide range of users (universal usability) may require a broader or highly diverse subject pool. For large-scale web services, testing might involve thousands of users (A/B testing).

Relationship to Other Evaluation Methods

- Usability testing is sometimes used synonymously with usability evaluation or usability studies, which can encompass a broader range of techniques like observation, questionnaires, and interviews.
- **Controlled experiments** differ from usability tests in their primary goal and rigor. Experiments aim to test hypotheses and show statistically significant differences by manipulating variables, requiring rigorous procedures and larger participant numbers for generalizable research findings. Usability tests, while measuring performance, focus on finding and reporting design problems quickly to inform iteration, often with less statistical analysis.
- **Expert reviews** (like heuristic evaluations and walkthroughs) are different from usability testing as they do not directly involve users, relying on experts' knowledge and analysis. Usability testing and expert reviews often reveal *different* usability problems. Expert reviews can complement usability testing and are sometimes performed earlier in the design process.

Types of Usability Testing/Related Evaluations

- **Paper mockups/prototyping:** Early, informal testing using paper screens, where an administrator simulates the computer. Inexpensive, rapid, and productive.
- **Discount usability testing:** A quick approach often using a small number of participants (3-6) to find serious problems quickly. Useful for formative evaluation.
- **Summative evaluation:** More extensive testing, often with more participants, done near the end of the design process to provide evidence for product announcements or training needs.
- **Universal usability testing:** Testing with highly diverse users, hardware, software, and networks to ensure accessibility for a wide range of people.
- **Field tests and portable labs:** Conducting testing in realistic environments, potentially over longer periods (longitudinal). Logging software is useful. Portable lab equipment exists. Beta testing with large numbers of users is a form of field testing.
- **"Can-you-break-this" tests:** Users actively try to find fatal flaws or crash the system. Useful for identifying serious bugs before release.
- **A/B testing:** An online method comparing two or more design alternatives (often web pages or apps) by releasing them to different user groups and collecting data on actual use. Involves large numbers of users and authentic situations. Can be extended to multivariate testing.

Limitations

- Usability testing often emphasizes **first-time usage** and may have **limited coverage of the interface features** within typical test sessions (1-3 hours). It is difficult to ascertain performance after extended use (weeks or months).

- It requires a **rigorously controlled environment**.
- Conducting tests effectively requires a test conductor with **expertise** in user interface evaluation.
- Setting up a usability lab can be expensive.
- Usability tests may be poorly suited for detecting problems with **consistency**.
- Complex systems can be hard to test with simple controlled experiments (a related method).
- Ecological validity (generalizability to the real world) can be a concern when testing is done in artificial lab settings.

In essence, usability testing is a key evaluation method that provides empirical evidence of how well users can interact with a system to achieve their goals, often conducted in controlled settings but increasingly incorporating field methods and remote participation to better capture the nuances of real-world use and inform iterative design improvements.

HEURISTIC EVALUATION AND WALKTHROUGHS

H**euristic Evaluation** and **Walkthroughs** are significant methods within the broader category of **inspection methods** used in usability evaluation. These methods are often performed by experts and generally do not require the direct involvement of typical users during the evaluation session itself, making them relatively quick and inexpensive alternatives or complements to user testing.

Heuristic Evaluation

- Heuristic evaluation is an informal usability inspection technique where experts evaluate user interface elements (like dialog boxes, menus, and navigation) to determine if they conform to a set of usability principles known as heuristics. It's a method for structuring the critique of a system using relatively simple and general heuristics. The key purpose is for experts, guided by heuristics, to apply their knowledge of typical users to predict usability problems that users might encounter.
- **How it's Done:**
    - It typically involves several evaluators independently critiquing a system.
    - The process includes a briefing session where evaluators are informed about the goal.
    - Each expert then independently inspects the product for 1–2 hours, using the heuristics for guidance.
    - Evaluators should make at least two passes through the system.
    - They compare their findings against the list of usability principles and identify other relevant problems.
- **Participants:** Experts, or researchers acting as experts, who are knowledgeable about interaction design, user needs, and typical user behavior. Non Experts who understand the task can also participate. Nielsen suggests that between three and five evaluators are sufficient, with five typically discovering about 75% of the overall usability problems. More experienced usability practitioners tend to find more problems.
- **Heuristics:** The evaluation is guided by a set of usability principles or rules of thumb. These heuristics closely resemble high-level design principles like consistency, reducing memory load, and using terms users understand. The original sets were derived empirically or from

analysis. New sets may be needed for different types of products like web, mobile, or collaborative technologies, based on a combination of usability and user experience goals, research, and market research. Heuristics can be adapted for specific products or decomposed into questions to guide evaluators.

- **When Used:** Heuristic evaluation can be performed on design specifications, prototypes, storyboards, and fully functioning systems. It can be used at any stage of a design project, including relatively early stages with a substantial prototype. It is often recommended to perform heuristic evaluation before user testing to eliminate a number of design problems.
- **Advantages:**
    - **Quick and relatively inexpensive** compared to laboratory and field evaluations that involve users.
    - Users do not need to be present.
    - Flexible in terms of what can be evaluated.
    - Experts frequently suggest solutions to problems.
    - Useful for identifying many usability problems.
    - Particularly good at identifying minor problems like inconsistencies that users might overlook while focusing on tasks.
- **Disadvantages:**
    - Can be subjective and rely on the evaluator's expertise and interpretation.
    - Potential for evaluator bias, although using multiple evaluators helps.
    - Findings are not always as accurate as they might seem.
    - May miss some usability problems that user testing would reveal.
    - Applying high-level heuristics can be tricky, especially for those new to the method.
    - Effectiveness can vary based on the evaluator's experience.

Walkthroughs

- **Definition and Purpose:** Walkthroughs are an alternative approach to heuristic evaluation for predicting users' problems without requiring user testing. They involve walking through a task with the product or design and noting problematic usability features. While most walkthrough methods do not involve users, some do.
- **Types:**
    - **Cognitive Walkthroughs:** Simulate a user's problem-solving process at each step in the human-computer dialogue, checking if the user's goals and memory lead them to the next correct action. They focus on evaluating designs for **ease of learning**, especially learning through exploration.
    - **Pluralistic Walkthroughs:** Involve a team that includes users, developers, and usability specialists.
- **How Cognitive Walkthroughs are Done:**
    - Identify characteristics of typical users and develop sample tasks.
    - Obtain a description, mock-up, or prototype and a clear sequence of actions for the task.
    - A designer and one or more UX researchers (or experts) come together to do the analysis.
    - They walk through the action sequences step-by-step, often simulating a "day in the life".

    - At each step, they answer specific questions focusing on the user's perspective: Will the correct action be evident? Will the user notice it's available? Will they understand the feedback?.
    - They provide a "story" explaining why each step is good or bad for a new user.
    - Can be done privately or in a group meeting.
- **Participants:** Experts simulating users. Evaluators. Designer and UX researchers. Pluralistic walkthroughs include users, developers, and specialists.
- **When Used:** Can be applied to design specifications, prototypes, storyboards, and fully functioning systems (as a general inspection method). Useful for assessing task flow early in the design process before empirical testing is possible. Well-suited for examining small parts of a system in detail. Can be adapted for different interfaces like web, handhelds, VCRs, smartphones, large displays, and public displays. Useful for assessing exploratory learning and first-time use.
- **Advantages:**
    - **Allow clear evaluation of the task flow early**.
    - **Focus on ease of learning** which is motivated by how users learn by exploration.
    - Do not require a functioning prototype.
    - Low cost.
    - Can be performed by developers.
    - More structured than a heuristic evaluation, potentially suffering less from subjectivity due to the task focus.
    - Provide detailed findings by examining each step of a task.
    - Good for evaluating alternate solutions.
- **Disadvantages:**
    - Can be tedious to perform.
    - May miss inconsistencies and general, recurring problems.
    - May take longer than heuristic evaluation to cover the same breadth of the interface because they examine steps in detail. Consequently, you may not evaluate as much of the interface in the same amount of time compared to heuristic evaluation.

Relationship and Comparison

- Both heuristic evaluation and walkthroughs are valuable **inspection methods** that rely on expert knowledge to predict user problems.
- They are generally quicker and less expensive than formal user testing.
- A key distinction lies in their **focus**: Heuristic evaluation uses general principles to review interface elements broadly, while walkthroughs, especially cognitive walkthroughs, simulate specific user tasks step-by-step to assess the ease of learning and task flow.
- Cognitive walkthroughs provide more **detailed, step-by-step findings**, making them suitable for evaluating small parts of a system thoroughly. Heuristic evaluation can cover a broader range of the interface more quickly.
- Both methods often reveal **different usability problems** compared to user testing. They can complement user testing and other evaluation methods. Performing heuristic evaluation before user testing is recommended to address initial design flaws.
- Walkthroughs are described as being more structured than heuristic evaluations.

In summary, heuristic evaluation and walkthroughs are essential expert-based methods for identifying usability issues early and efficiently in the design process, with heuristic evaluation offering a broader, principle-based review and walkthroughs providing a detailed, task-focused analysis centered on learnability.

ANALYTICS AND PREDICTIVE MODELS

Analytics and **Predictive Models** are valuable methods for evaluating user interfaces and predicting user behavior, often used as alternatives or complements to direct user testing.

Analytics

- **Definition and Purpose:** Analytics involves analyzing data gathered computationally, especially from large volumes of data, to reveal patterns, trends, and associations, particularly relating to human behavior and interactions. The purpose is to derive various measures of user performance and usage patterns.
- **Types:**
  - **Web analytics:** Commonly used to measure website traffic by analyzing users' click data. It can show who is using the website, when, for how long, where their IP addresses are located, bounce rates, and more. Examples include Google Analytics, ClickTale, Clicky, Crazy Egg, KISSmetrics, Mox Analytics, and TruSocialMetrics. The NIST Web Metrics page also provides extensive testing tools like WebSAD, WebCAT, and WebVIP to guide web designers. Researchers have also developed statistical models like WebTango to predict website ratings based on layout metrics.
  - **Learning analytics:** Applied to learning systems.
  - **Visual analytics:** Displays and allows manipulation of thousands or millions of data points visually, such as in social network analysis. This research area combines data mining and information visualization, keeping the human in the loop but augmenting human capabilities with computation.
- **How it's Done:** Involves computationally analyzing large datasets to find patterns and trends. For web analytics, this includes analyzing click data or usage logs. Often requires tools or platforms designed for analyzing and making predictions from large volumes of data. Using these tools typically requires data analytic skills and statistical knowledge, although many tools are fairly straightforward to use for basic analysis. Initial analysis often begins by looking over the data for anomalies (data cleansing) and identifying patterns or calculating simple numerical values like ratios, averages, or percentages. More detailed work follows using structured frameworks or theories. Data wrangling, the process of cleaning and preparing data, can be time-consuming.
- **Advantages:** Allows evaluation without users being present. Useful for getting an overview of how users are using a system, like a website. Can reveal subtle trends. Particularly useful for large volumes of data ("Big Data"). Provides quantitative data.
- **Disadvantages:** Requires tools and potentially data analytic skills. Can involve tedious data cleaning. Privacy issues need to be considered.
- **Relationship to other methods:** Analytics is sometimes referred to as a type of analytical evaluation. It is listed alongside inspection methods and predictive models as evaluation techniques that don't necessarily require user presence.

Predictive Models

- **Definition and Purpose:** Predictive models are evaluation techniques that provide various measures of user performance, typically using formal methods or formulas, without requiring actual user testing. Their purpose is to estimate the efficiency of different designs for various kinds of tasks and enable the comparison of design alternatives based on numeric predictions like speed or errors. They are often based on psychological theory.
- **How it's Done:** Predictive modeling involves analyzing the physical and mental operations needed to perform tasks at the interface and operationalizing these as quantitative measures using formulas. Applying these models requires evaluators skilled in their use.
- **Specific Predictive Models:**
  - **GOMS (Goals, Operators, Methods, Selection Rules):** A well-known predictive modeling technique, or rather, a family of models that vary in granularity. GOMS describes how a user performs a computer-based task using goals and the selection of methods and operations from memory. It models procedural knowledge and allows for quantitative predictions. GOMS models have been used to predict user performance when comparing different applications and devices and are particularly effective for predicting expert, error-free performance for certain kinds of tasks. GOMS is a simplification of the human processor model.
  - **Keystroke Level Model (KLM):** A member of the GOMS family, considered a "daughter" of GOMS. It provides a formula for determining the amount of time methods and operations take. KLM offers rough measures of user performance in terms of execution times for basic sequences of actions. It is described as an "engineering"-level technique built upon theoretical principles but not requiring deep theoretical background.
  - **Fitts' Law:** Predicts the time it takes to reach a target using a pointing device. It models the relationship between speed and accuracy when moving towards a target on a display. In interaction design, it's used to describe the time to point at a target based on its size and distance, specifically for devices like a mouse. Fitts' Law is well established and accurate for predicting pointing times. It can help designers decide button placement, size, and spacing.
  - **Human Processor Model:** Based on the information processing model, it models a user's cognitive processes (perceptual, cognitive, and motor processors) when interacting with a computer. It predicts which cognitive processes are involved and how long a user will take to carry out various tasks, which can be useful for comparing different interfaces. This model was simplified into GOMS.
  - **WMS model:** Mentioned as another predictive model based on psychological theory.
- **Participants:** Predictive models require evaluators who are skilled in applying the specific models. They do not require typical users or usability experts in the same way user testing or heuristic evaluations do.
- **When Used:** Predictive models can be used at an early stage of design before building elaborate prototypes. They are successful for systems with limited, clearly defined functionality, such as telephone systems or data entry applications. They are useful for comparing different design alternatives and estimating performance for various tasks.
- **Advantages:** Do not require user testing. Can predict efficiency and potential problems early in the design process. Provide quantitative data and precise predictions. Relatively low cost compared to user testing. Can help optimize design features like button placement.
- **Disadvantages:** Predictive models can primarily make predictions about predictable behavior. It is difficult to use them to evaluate how systems will be used in real-world contexts, where behavior is often unpredictable. Applying them requires specific skills. Some models, like

GOMS/architectures, may inherit limitations, predicting only skilled user execution time for familiar tasks, and may need to be extended to predict learning time, errors, and performance under stress. They may be tedious to perform.

- **Relationship to other methods:** Predictive modeling is a category of analytical evaluation. Like inspection methods, they allow evaluation without users being present. They can complement other evaluation techniques.

In summary, both analytics and predictive models offer valuable, often quantitative, insights into user behavior and system performance without necessarily requiring direct user participation in the evaluation session. Analytics focuses on analyzing collected data from real usage or logs, especially for large datasets, while predictive models use theoretical formulas and expert analysis to forecast user performance based on task structure and interface design.