HCI Summary

This document is a comprehensive summary of the key topics from the Human-Computer Interaction (HCI) course in the Informatics and Computing Engineering Bachelor's Degree at FEUP. It was created by Francisco Afonso in collaboration with an AI assistant to provide a structured and accessible overview of the main concepts, principles, and methodologies covered throughout the course. The aim is to serve as a study guide and reference for understanding HCI's theoretical and practical aspects.

1 - Introduction

Introduction to Human-Computer Interaction (HCI)

- **Objective** - Study the interaction between humans and computers, focusing on designing technology that facilitates this interaction.
- **Key Questions**:
  - What do interfaces have in common?
  - Why are there different solutions for similar purposes?
  - Are existing solutions efective?

Key Concepts

- **User Interface (UI)**:
  - The visible part of the system (visual, audio, tactile).
  - Purpose: Enable interaction, task execution, and provide feedback to users.
- **User Experience (UX)**:
  - Encompasses the entire experience with the system, including emotional, aesthetic, and contextual aspects.
  - UX is designed for users, not directly created.

Design Principles in HCI

- **Common Myths**:
  - "Good design means good graphics."
  - "Marketing knows the users."
  - "Good design is common sense."
  - "The interface can be designed at the end."
- **HCI Mantras**:
  - Know your users: their physical, cognitive, and social context.
  - "The user is not like me": Avoid designing based on personal preferences.

User-Centered Design (UCD) Process

- Iterative process focused on users' needs.
- Combines investigative methods (e.g., interviews) with generative methods (e.g., brainstorming).

Iteration and Prototyping

- **Fail Fast Philosophy**: Fail early, quickly, and often to improve designs.
- **Iterative Process**: Ideate -> Prototype -> Test -> Repeat.

2 - User Research

Why User Research?

- Focuses on:
  - Potential users, their characteristics, tasks, and context.
  - Expectations and goals.
- **Purpose**: To design a usable and useful interface by understanding the users.

Types of Participants

- **Users**: Final users of the product, often unaware of their needs and expectations.
- **Stakeholders**: Define high-level goals but may misinterpret user needs.
- **Designers**: Bridge users and stakeholders by designing and iterating on interfaces.

Key Concepts

1. **User Characteristics**:

    - **Definition**: Knowledge of tasks and tools, mental models, and level of expertise.
    - **Differences**: Personal traits, physical/cognitive abilities, cultural diversity.
    - **Expertise Evolution**:
        - Beginner → Advanced Beginner → Competent → Expert.

2. **Task Analysis**:

    - **Formal**: Structured lists or diagrams for existing systems.
    - **Informal**: Exploratory approach for new designs.

User Research Methods

- **Implicit**: Eye tracking, observation studies, reaction measurements.

- **Explicit**: Surveys, self-reports, think-aloud protocols.

- **Qualitative vs. Quantitative**:

    - **Qualitative**: Subjective insights (e.g., questionnaires).
    - **Quantitative**: Numeric data (e.g., task completion time).

Ethnography and Observation

- **Ethnography**: Long-term analysis of user behavior in context.
- **Observation**: Monitoring user behavior to understand actions, tools, and breakdown points.

PACT Analysis

- **People**: Attributes (e.g., physical, cognitive, cultural traits).
- **Activities**: Tasks and their frequency, complexity, and goals.
- **Context**: Physical, social, and organizational environments.
- **Technologies**: Current tools and how new ones might be applied.

2.1 - Personas

What are Personas?

- **Definition**: Archetypes of typical users based on patterns in motivations, needs, and frustrations.

- **Purpose**:

    - Guide product decisions by understanding user behavior and goals.
    - Prevent designing based on designers' personal preferences.

Creating Personas

1. **Process**:

    - Identify patterns during user interviews.
    - Define groups based on:
        - Motivations.
        - Needs.
        - Pain points.

2. **Characteristics**:

    - Rich descriptions (name, photo, personal traits).
    - Specific details ("owns a dog named Billy").
    - Not real people, but realistic and relatable.

Best Practices

- Keep personas simple and goal-focused.
- Prioritize behavior patterns over job/role descriptions.
- Limit the number of personas to a manageable set.
- Avoid correlating market segments directly with personas.

Context/Activity Scenarios

- Complement personas with scenarios describing tasks in context.
- Example:
    - Paulo wants to check employee expenses. *He navigates the system to retrieve the information without UI specifics.*

3 - Ideation

Conceptual Model

- **Definition**:

  - Represents what users can do with the system and the underlying task-related concepts.
  - Includes **Objects** with attributes, and **Actions** in the task domain.
  - Establishes mappings between these elements to organize ideas before designing the user interface through **Relations**.

- **Purpose**:

  - Helps clarify how the system operates and aligns with user tasks.
  - Incorporates familiar metaphors for the target users.

- **It is NOT**:

  - The user interface.
  - The system architecture.
  - Specific interactions like key presses, mouse actions, or UI layouts.
  - The user's mental model.

Functional and Usability Requirements

- **Functional Requirements**:

  - Derived from personas and scenarios, focusing on tasks the system should support (e.g., logging in, managing photos, sharing content).

- **Task Selection**:

  - Tasks should be:
    - Real and representative.
    - Focused on "what" (not "how") the system will achieve the task.
    - A mix of simple/common and complex/rare tasks.

- **Usability Requirements**:

  - Evaluate usability through:

    - **Efficacy**: Achieving goals without errors.
    - **Efficiency**: Resources and time spent.
    - **Satisfaction**: Users' subjective feedback.

Ideation Principles

- **Myths and Realities**:

  - Innovation doesn't start with an idea; instead, ideas are the output of understanding customer needs and challenges.
  - **Quantity vs. Quality**: Generating numerous ideas leads to higher quality solutions due to iterative learning.

- **Best Practices**:

  - Start with individual ideation, then collaborate as a group.
  - Share ideas openly and without judgment.
  - Generate ideas in parallel to encourage diversity.
  - Avoid premature convergence on a single solution.
  - Prototype ideas for validation.

- **Techniques for Ideation**:

  - **Crazy 8**:
    - A Design Sprint technique where each participant sketches 8 ideas in 8 minutes.
    - Objective: Move beyond initial ideas to explore a wide range of solutions.
    - Follow-up: Share ideas and discuss with the team.

4 - Prototyping

What is Prototyping?

- **A partial representation of a system**, aimed at exploring and validating ideas.

- **Goals**:

  - Test concepts early and cost-effectively.
  - Identify bad ideas quickly.
  - Facilitate user feedback.

Prototyping Techniques

1. **Experiential Techniques**:

   - **Storyboarding**:
     - A sequence of drawings to depict settings, context, and events. Focus: Context, task sequence, and user goals (no UI specifics).
   - **Bodystorming**:
     - Physical enactment of scenarios to ideate and evaluate system functionalities.
   - **Wizard of Oz**:
     - Simulate system functionalities with a human (e.g., mimicking complex AI behavior).

2. **Prototype Fidelity**:

   - **Fidelity**: Focuses on the visual appearance.

- **Functionality**: Focuses on the system's operational aspects.
- **Horizontal vs. Vertical**:
  - Horizontal: Broad coverage of features with minimal depth.
  - Vertical: In-depth representation of specific features.

Prototype Types

1. **Low-Fidelity (Lo-Fi) Prototypes**:
   - Characteristics:
     - Quick, easy, and inexpensive to create.
     - Prioritizes conceptual ideas over aesthetics.
     - Examples: Paper sketches, cardboard models.
   - Pros: Encourages iteration, disposable.
   - Cons: Limited scalability, lacks detailed realism.
2. **Mock-ups**:
   - Characteristics:
     - Digital or printed representations of UI.
     - Can range from wireframes to high-resolution designs.
   - Pros: Scalable and supports consistency in design.
   - Cons: Hi-Fi versions may prematurely influence user feedback.
3. **Interactive Mock-ups**:
   - Add interactivity through tools like clickable hotspots.
   - Allows simulation of user flows without actual programming.
4. **Functional Prototypes**:
   - Include partial implementation of system logic.
   - May be developed on the final platform or a prototyping tool.
   - Typically involve higher fidelity and are closer to the final product.

Prototype Tools

1. **Simple Hyperlinking**:
   - **Marvel App**:
     - Upload sketches or images and add tappable areas.
     - Suitable for basic flows and mobile testing.
2. **Advanced Collaboration**:
   - **Figma**:
     - Supports real-time collaboration and interactive flows.
     - Free for students.
3. **Testing-Focused**:
   - **Quant-UX**:
     - Advanced widgets and testing tools.
     - Free and open-source.
4. **Promising Newcomer**:
   - **Penpot**:
     - Open-source, with real-time collaboration features.
5. **Commercial Tools**:
   - Popular options include **Webflow**, **InVision**, and **JustInMind**.

5 - Evaluation

Evaluation in the Design Process

- **Purpose**: Ensure the product meets user needs and achieves usability goals.
- **Approach**:
  - Involves interdisciplinary teams with equal contributions.
  - Incorporates iterative design with brainstorming and prototype testing.
  - Evaluation can occur at any stage: conceptual model, prototypes, or finished product.

The 5W 1H Framework

1. **Why**: To determine usability and user satisfaction.
2. **What**: Evaluate models, prototypes, or final systems.
3. **Where**: Conduct in natural or controlled environments.
4. **When**: During design or post-release for insights on future designs.
5. **Who**: Involve users or experts.
6. **How**: Use formative or summative methods.

Types of Evaluation

1. **Formative**:
   - Conducted during design.
   - Focus on identifying problems and refining designs.
   - Example: Observing user interactions.
2. **Summative**:
   - Post-design evaluation of overall success.
   - Focus on performance metrics like time, errors, and satisfaction.

Users or Not?

1. **User Involvement**:
   - Field Studies: Natural settings to observe real-world use.
   - Usability Testing: Controlled tests of user performance and satisfaction.
2. **Without User Involvement**:
   - **Heuristic Evaluation**: **Expert reviews** based on usability heuristics, good for evaluating initial designs and prototypes, it's fast, sort of cheap and easy to use.
   - Predictive Evaluation: Predicts performance using models (e.g., GOMS, KLM).

Nielsen’s Usability Heuristics

1. Visibility of system status.
2. Match between the system and the real world.
3. User control and freedom.
4. Consistency and standards.
5. Error prevention.
6. Recognition rather than recall.
7. Flexibility and efficiency of use.
8. Aesthetic and minimalist design.
9. Help users recognize, diagnose, and recover from errors.
10. Help and documentation.

Testing with Users

- **Participants**: Should represent the target audience.
- **Number of Users**:
  - **Formative Testing**:
    - 1 user -> 33% of problems.
    - 5 users -> 85%.
    - 15 users -> 99%
    - 3 iterations with 5 users -> better than 1 x 15
  - **Summative Testing**:
    - Larger samples (10-20+) for statistical validity.
- **Methods**:
  - A/B Testing: Compare versions within or between groups.
  - Variables:
    - **Dependent**: Time, errors, satisfaction (linked to prototype purpose).
    - **Independent**: User demographics, design characteristics.

Data Collection

1. **Types of Data**:
   - **Quantitative**: Measurable (e.g., time taken, errors).
   - **Qualitative**: Open-ended insights (e.g., "What did you like?").
2. **Objectivity**:
   - **Objective**: Measurable, bias-free (e.g., time, errors).
   - **Subjective**: User perceptions (e.g., preferences, satisfaction).
3. System Usability Scale (SUS) : A widely-used method to assess user satisfaction and usability.

6 - Experimental Design and Ethics

Experimental Design

- **Methods**:
  1. **Formative Assessment**:

        - Identifies usability issues using:
            - Direct observation.
            - Think-aloud protocols.
            - Wizard-of-Oz.
            - Interviews or questionnaires.

    2. **Summative Assessment**:

        - Measures performance using:
            - Usability tests.
            - A/B tests.

- **Tasks**:

    - Should be real and representative, focusing on what rather than how.
    - Example:
        - **Wrong**: "Select Uber and type 'Aliados.'"
        - **Right**: "Book a ride to Aliados using Uber."
    - Maintain consistency for summative evaluations to ensure fairness.

Usability Measures

- Examples of metrics:
    - Time to complete tasks.
    - Number of errors or help requests.
    - Completed tasks percentage.
    - User satisfaction.

Data Collection

- **Direct Observation**:

    - Observing users in real-time or through video recordings.

- **Indirect Observation**:

    - Diaries (manual tracking).
    - Automated interaction logs.

- **Pilot Tests**:

    - Conducted with 2-3 individuals to validate: Instructions, tasks, questionnaires, duration, etc.

Challenges in Experimental Design

- **Side Effects**:
    - **Reward Limits Creativity**: Users may focus on rewards.
    - **Observer Effect**: Users behave differently when observed.
    - **Novelty Effect**: Initial excitement for a new system may fade.
    - **Survivor Bias**: Ignoring failed solutions can lead to incomplete insights.

Ethical Considerations

- **Assumptions**:

    - Scientists are ethical? Not always (e.g., Tuskegee Syphilis Study, Milgram Experiment).

- **Participant Welfare**:

    - Avoid stress or discomfort.
    - Obtain informed consent.
    - Ensure voluntary participation and anonymity.
    - Emphasize system testing, not user evaluation.

- Special considerations for: Children, individuals with disabilities, minorities, etc.

Principles of Ethics

1. **Respect for People**:

    - Respect autonomy and choices.
    - Use informed consent.
    - Protect vulnerable individuals.

2. **Charity (Doing Good)**:

    - Prevent harm and minimize risks.
    - Act kindly beyond duty.
    - Balance risks and benefits.

3. **Justice**:

    - Ensure fairness in recruitment and distribution of risks/benefits.

- Consider participants’ time, comfort, privacy, and control throughout the evaluation process.

7 - Qualitative Analysis

What is Qualitative Data?

- **Non-numeric data** such as text, images, audio, and video.
- Focuses on:
  - Attitudes, behaviors, motivations, and experiences.
  - Describing and explaining events and opinions.

Techniques for Qualitative Analysis

1. **Word Clouds**:

   - Represent word frequency but lack content analysis.

2. **Thematic Analysis**:

   - A systematic approach involving:
     - **Codes**: Labels for meaning (e.g., “password”).
     - **Themes**: Patterns in data formed by groups of codes (e.g., “security”).

3. **Affinity Diagrams**:

   - Use collaborative tools (e.g., post-its) to group data into themes.
   - Applicable in ideation, synthesizing research, or team consensus.

4. **Other Techniques**:

   - Sentiment Analysis: Evaluating emotional tone.
   - Factorial Analysis: Examining interrelations among variables.

8 - Quantitative Analysis

What is Quantitative Analysis?

- Focuses on statistical methods for **measurable data**.
- Includes:
  - **Descriptive Statistics**: Summarizing sample characteristics.
  - **Statistical Inference**: Drawing conclusions about populations from samples.

Key Concepts in Statistics

- **Variables**:

  - **Dependent**: Measured outcomes (e.g., number of errors).
  - **Independent**: Conditions or groups being tested (e.g., user age).

- **Measures**:

  - **Nominal**: Categories (e.g., color).
  - **Ordinal**: Ranked scales (e.g., satisfaction levels).
  - **Continuous**: Numeric values (e.g., time, errors).

- **Descriptive Statistics**:

  - **Mean**: Average value.
  - **Standard Deviation**: Variability in data.
  - **Median**: Middle value.
  - **Quartiles**: Data distribution segments.
  - **Mode**: Most frequent value.

Statistical Inference

1. **Hypothesis Testing**:

   - Null Hypothesis (H0): Assumes no relationship (e.g., “No performance difference”).
   - Confidence Level (1-α): Probability of not making a false assumption (e.g., 95%).

2. **Common Tests**:

   - **T-Student**: Compare means (e.g., trackpad vs. mouse time).
   - **Chi-Square**: Compare expected vs. observed frequencies (e.g., preferences).
   - **Pearson’s Correlation**: Measure relationship between variables (e.g., height vs. shoe size).

Using Statistical Methods

- **T-Student**:

  - Assumes normal distribution.
  - Tests if two means are significantly different.

- **Confidence Intervals**:

    - Define a range within which the true population mean likely lies.

- **Chi-Square**:

    - Tests if observed distributions differ from expected ones.

- **Pearson's Correlation**:

    - Measures the strength and direction of a relationship between two variables.

9 - Help and Documentation

User Strategies for Seeking Information

1. **Browsing**:
    - Casual exploration of information.
    - Requires clear structure (e.g., breadcrumbs, anchors to familiar locations).
2. **Search**:
    - Users know their target and seek help finding it (e.g., using a search bar).
3. **Query**:
    - Users have a general idea but lack specifics.
    - Good systems offer flexible criteria, examples, and effective result displays.
4. **Structured Search**:
    - Uses categories/subcategories to refine focus dynamically.
5. **Guided Search**:
    - Step-by-step process for limited, sequenced options (e.g., purchase wizards).

Documentation Types

1. **Manuals**:

    - **User Manual**: Step-by-step guidance for task-oriented learning.
    - **Quick Start Guide**: Basic instructions for initial use.
    - **Reference Manual**: Exhaustive functional details for expert users.
    - **Quick Reference Guide**: Compact cheat sheets for frequent tasks.

2. **Interactive Help**:

    - Directly integrated with the system, often context-sensitive.
    - Features:
        - Availability throughout the system.
        - Consistency and flexibility.
        - Robustness in failure scenarios.

Key Features of Good Documentation

- **Learning Facilitation**:

    - Build user mental models using relatable examples and visual cues.
    - Present information incrementally, from general to specific.
    - Avoid technical jargon; use user-friendly language.

- **Ease of Navigation**:

    - Organized by user goals and tasks.
    - Visual aids for motivation, concepts, and procedures.
    - Simplified table of contents and task-oriented indexing.

Interactive Help Guidelines

1. Accessible in all contexts.
2. Provide task-oriented and minimalist content.
3. Avoid irrelevant images or over-indexing topics.
4. Write content addressing user questions/problems.

- **Heuristics for Writing Help Content**:
    - Focus on tasks, not technology or functions.
    - Use minimalist designs and small file sizes.
    - Optimize for mobile users (e.g., reduced clicks/touches).
    - Collaborate with users during the process.

Additional Help Formats

- **Tutorials**:
    - Descriptive: Step-by-step guides.
    - In-App: Contextual and interactive.
    - Video/Animation: Visual demonstrations.
- **FAQs**:
    - Address common questions succinctly.