



## **Human Factors Engineering**

### **IDEA: Interface Design by Ecological Analysis**

*The Monterey Technologies, Inc.*

*User-centered analysis & design methodology*

---

The user is at the core of the design process, surrounded by the environment in which their work is performed. The design process is successful only if it considers *both* user and environment attributes; inadequate attention to one or the other leads to interface failure. We use an approach based in systems engineering where we observe and develop an understanding of activities at every stage of the process. We focus on learning the user's information requirements, task environment, capabilities, preferences, and level of satisfaction with the work. Our process identifies interface or process issues and problems, and provides us with an understanding of why the issues and problems occur. Only when there is a thorough understanding of why problems exist can reasonable solutions be developed. Our team is composed of a variety of experts in human behavior, technology subject matter experts, system and process designers, and individuals skilled in human performance improvement. This diversity provides us with the expertise to thoroughly understand human performance issues and design solutions to improve human performance and productivity.

The design process is called *IDEA*: Interface Design by Ecological Analysis. It is a structured set of steps to identify and document the task and user environment, identify and document existing process issues, identify likely areas for human performance improvement, and generate recommended courses of action to improve human performance. The steps in the process are as follows:

#### **1. Mission and Task Analysis**

- Define the objective
- Identify, from a top down perspective, what is to be accomplished, and any constraints that exist.
- Identify mission phases, segments, and tasks, and information required in each
- Identify and list mission critical tasks (those that result in mission success/failure)
- Identify 2 or 3 important "scenarios" in which these tasks are performed

#### **2. System Specification Documentation**

- Identify all existing systems associated with the mission to be accomplished.
  - Document information flowing into and out of each system associated with the mission.
  - Identify all required system interactions with supporting systems.
- 3. Conceptual Design: Mission and Task Analysis → Task Flows**
- For current system, document current ways of performing critical tasks (task flows, i.e. how the work is currently being accomplished)
  - Document mission triggers, and user's response to each, using current systems, in low and high levels of detail
  - For current system or process, estimate workload for mission critical tasks
  - Develop low detail "Blue Sky" (i.e. highly desirable) concept of how each critical task might be performed with new system
  - Develop high detail "Blue Sky" concept of how each critical task might be performed with new system
  - Compromise: obtain design team consensus on what is desirable, yet still feasible (within scope)
- 4. Design Prototyping**
- Develop a prototype of the interface or process concept.
  - User walks through design concept, providing usability feedback to designers.
- 5. Detailed Design**
- Use a rapid prototyping tool to model or prototype a new interface or process.
  - Walk through design with the users, incorporating scenarios identified during Mission and Task Analysis
  - Document user suggestions for change, rate for desirability vs. feasibility
  - Repeat (an iterative process to achieve optimum solution).
- 6. Documentation**
- Document all issues and problems identified in the user-centered design process.
  - Document recommendations for process changes to improve human performance.
  - Document recommendations for technology applications (not necessarily specific hardware or software) to improve processes.
  - Document the recommended approach or solution (user interface or process) such that developers doing the final implementation know how it should look, and respond.