Key principles on user design

Design

These principles summarize best-practice for the design of usable systems. They are not 'hard-and-fast' and can be tailored to slut specific Interaction needs. 1 Design for the users and their tasks Interactive computer systems do not operate in isolation - they exist to support users to perform their work tasks. A successful system meets business objectives through being user-centered and task-oriented. Throughout development, always bear in mind the characteristics of the user population, their real-world tasks, and their working environment.

Be consistent To Meltzer the learning requirements for users, make the behavior of common interface elements and dialogue boxes as consistent as possible. This often means designing to be consistent with other existing components of the computer system. Although you may be able to design a new slick style of interaction, if it is inconsistent with the rest of the system, it will take users time and effort to learn it and get used to It. 3 Use simple and natural dialogue The dialogue between user and system should follow the natural sequence Implied by the task.

There should be no more information presented to the user than is necessary to complete the current task, because each item of irrelevant information adds complexity to the dialogue. All messages and Instructions should be in plain English and should use the vocabulary of the intended audience. Terminology should be defined so that the same term always has the same meaning. 4 Reduce unnecessary mental effort by the user users should be able to concentrate on their task without worrying about how the tool they are using - the computer.

The more complicated the interaction with the imputer, the more frustrated users become and the more distracted they are from their real task. If users have to invest too much mental effort in working out how to operate the computer, they will be less efficient and make more errors. This is particularly dangerous In safety-critical situations, or where a business relies heavily upon the outcome off task. Simplify frequent tasks as much as possible. Users should not have to remember Information from one part of the system In order to use another part.

Instructions for the use of the system should be visible or clearly attributable whenever appropriate. 5 Provide adequate feedback Users need to be confident that their actions have been successful. This is usually evident from a distinct change in appearance of the window, e. G. After completing a transaction. If completion will take more than a second or so, a progress or working Indicator should be displayed to give the user confidence that the computer Is soul operating. But avoid presenting the user with unnecessary or irrelevant diagnostic or status information about the internal state of the system.

Feedback should be provided at several levels of interaction. At a low-level, users should receive confirmation that they have operated a control successfully - for example, a button immediately Indicates when it has been operated by appearing momentarily pressed In. Users should also be Informed when a longer sequence of operations has been Provide enough relevant information for users to know where they are. This can usually be achieved by applying a meaningful and consistent mechanism for assigning titles to windows, and by using range and location indicators such as scroll bars or page numbers.

Other mechanisms, such as providing navigation map, overview, or history of areas visited (bread-crumbs), also should be considered. There should be clear and easy routes between the different windows users need to access for particular tasks. These should be provided in a form which is appropriate for the user at each stage of a task. Users often choose system functions by mistake and should be provided with a clearly marked " emergency exit" to leave the unwanted state without having to go through an extended dialogue, for example by providing an Undo facility, or a Cancel button in a modal window. 7 Let the user drive

The user should be able to select the information they need in a sequence which is convenient to support each individual task. The system should provide as few constraints as possible on how the task is achieved, but should provide simple ways to achieve frequently performed tasks. 8 Present information clearly Arrange on-screen information so as to increase the user's ability to discriminate between different items and groups of data, through the use of spacing, boxes, and visual coding techniques. However, don't overwhelm the user by providing more information or coding than is necessary to perform the task.

Where appropriate, present information in a consistent manner and location across different windows to enhance learn ability. 9 Be helpful So far as possible, systems should be self-explanatory so that they can be used with the minimum of help and documentation. The information on the window should be expressed in terms of the user's task. Always use tool tips for buttons labeled with an icon. The initial on-line help provided should relate to the specific context of interaction on the window. It should be task-oriented, listing the concrete steps to be carried out. 10 Reduce errors

Reduce errors by guiding the user down the correct path to accomplish their goal. Where appropriate to the task, users' responses should be constrained to avoid error (but not where this would limit the users' reasonable choice in how to perform their tasks). Similarly, the system should validate data entry as near to the point of input as possible. Error messages should be expressed in plain language avoiding the use of codes, indicates precisely the problem, and should suggest a solution. Most of all, error messages should be phrased so as to avoid blaming the user.