Ch 9 - Evaluation Techniques

1. Role of evaluation: Assessing our designs and testing our systems, to ensure they behave as expected.
2. Ideally, evaluation phase should exist throughout the design life cycle.
3. Two ideal techniques: evaluation by designer and evaluation by users.
4. Goals:
    a. To assess the extent and accessibility of systems functionality.
    b. To assess user's experience of the interaction.
    c. To identify any problems with the system.
5. Evaluation through expert analysis.
    a. It can be expensive and difficult to have users perspective, early in the design cycle.
    b. The basic idea: to identify any violation in cognitive principles or ignorance in empirical results.
    c. 4 Approaches
        i. Cognitive Walkthrough
            1. Introduction to psychological theory into informal and subjective walkthrough technique.
            2. Similar to code-walk in SW Engg.
            3. In Cognitive walkthrough, there is sequence of steps that an interface will require an user to accomplish to perform any given task.
            4. Main focus: How easy it is to learn the system.
            5. To do this evaluators go through each task and provide a story.
            6. To do this we need:
                a. Prototype of the system
                b. Task description.
                c. complete written list of actions needed to complete the task.
                d. Indication of who the users are and the assumptions made about them.
            7. 4 questions need answers
                a. Is the effect of the action same as the users goal at that point?
                b. Will the users see that the action is available?
                c. Once the user have found the correct action, will they know it is the one they need?
                d. After the action is taken, will users understand the feedback they get?
            8. Document the above details for analysis.
        ii. Heuristic Evaluation
            1. It is a guideline to critique the decisions made.
            2. Hence considered as discount usability technique.
            3. At least 3 or 5 evaluators to critique and come with potential usability problems.

4. This amounts to approx. 75% of potential problems being discovered.
5. To aid the evaluators, heuristics are provided as guidelines.
6. They also rate the severity of the problem on a scale of 1-4, 4 being usability catastrophe.
7. Nielsen's ten Heuristics are:
    a. Visibility of the system status.
    b. Match between the system and real world.
    c. User control and freedom
    d. Consistency and standard
    e. Error prevention
    f. Recognition rather than recall
    g. Flexibility and efficiency of use
    h. Aesthetic and minimalistic design
    i. Help users recognize, diagnose and recover from errors
    j. Help and documentation

iii. Model based evaluation: Use of model for the purpose of evaluation like dialog models and design methodologies

iv. Using previous studies in evaluation: Using expert review

6. Evaluation through User Participation
    a. Styles of evaluation
        i. Laboratory Studies

        Users are taken away from their normal work environment and brought to controlled environment like a lab.

        Advantages: sophisticated audio/visual recording and analysis facilities, two way mirrors, instrumented computers... can be used.

        Disadvantages: Does not emulate the real world, altough sometimes that may be the only option.

        ii. Filed Studies

        The analysis in the users work environment

        Disadvantages: Observation is difficult as more noise, movement and interruptions.

        Advantages: represents the actual use of the system in real world

    b. Empirical Methods: Experimental evaluation
        1. Participants
            a. Must match the expected user population
            b. Familiar demography

        c. Testing with one user will yield at least a third of the problems and it is of little gain after using more than 5 users
    2. Variables
        a. Independent variables: like interface style, menu and options, etc. This can be manipulated
        b. Dependent variables: these are measuring standards based on independent variables.
    3. Hypotheses
        a. Is a prediction of the outcome of the experiment
        b. framed in terms of dependent and independent variables
        c. The aim is get the hypotheses right
    4. Experimental design
        a. Between subjects: each participant is assigned a different condition by manipulating the independent varibales
        b. Within subjects: each user performs under each different condition
c. Observational Techniques
    i. Think Aloud:
        1. Users are asked to talk through what he is doing as he is being observed
        2. How the system is actually used?
    ii. Cooperative evaluation
        1. variation from think aloud where the users are asked some questions like Why? and What-if?
d. Query Techniques: Ask the user
    i. Interviews
    ii. Questionnaires

Usability Heuristics

1. UI should be very simple because every potential feature is another possibility of getting it wrong or misunderstanding it.
2. ideally only present the info, the user needs and nothing more.
3. Similar tasks can be grouped together.
4. Do not over do colors in an interface. They should not be saturated or there should not be wide contrast.
5. Use colors only to highlight or categorize them as 8% male are colorblind.
6. Extraneous information will always confuse a novice user.
7. Must always use user centered language than the actual terminology, eg, 317 - British pounds.
8. Should not enforce naming convention.

9. There should be mapping between computer display and users conceptual model information
10. Metaphors may mislead users. E.g.. Typewriter metaphor can help users recognize backspace and return keys but will not give them any knowledge about replace option
11. Minimize Users memory load
    i. Computers can very easily store data, thus must not burden users with recalling data.
    ii. It is much easier for users to manipulate data than create it from scratch
    iii. The use of generic rules to govern the complex system.
12. Consistency
    i. If the user know that the same command always performs the same action, they will be more confident in using the system
    ii. The same information or commands must be located in same place in all the screens to preserve consistency.
13. Feedback
    i. System must continuously inform the users about what it is doing and how it is interpreting the users input
    ii. It should also provide positive feedback and partial feedback as mush as possible
    iii. It should not be abstract, but should be specific
14. Response time: Feedback is important when there is long response time
    i. If 0.1s, then no special feedback necessary.
    ii. If 1.0s, then no feedback necessary, but the user will feel that it is acting on the data
    iii. If 10s, then it is important to let the uses know about the system and if longer, than must allow users to perform other tasks.
15. System Failure: Informative feedback must be provided.
    i. Users must not feel trapped when using a system, hence there should be clear way of exiting or cancelling a task in every phase(Undo)
    ii. Shortcuts: Some frequently used operations can have shortcuts, so that user can spend less time performing repetitive tasks.
16. Good Error Message
    i. Critical for 2 reasons
    ii. represent situations where the user is in trouble and cannot achieve their goal
    iii. They present opportunities to learn about the system
    iv. Must follow 4 rules
        1. Must use clear language and avoid using codes. It must be understood without referring to the manuals.
        2. Must be precise rather than vague.
        3. It should help user solve the problem
        4. It should be polite, and must not explicitly blame the user
17. Prevent Error: Better than good error messages is to prevent them altogether

i. User errors can avoided by redesigning the way of representing information
ii. This can be helpful for frequently used operations
iii. Avoid modes
iv. If cannot be avoided, then there should be clear status indicators

18. Help and Documentation:
    i. Ideally there should be no need for documentation.
    ii. It cannot be avoided if there is more potential than the user can learn in one go.
    iii. It is helpful when the user wants to be an expert.
    iv. The fundamental truth is that users generally avoid reading manuals, rather they directly jump in using the system.
    v. Three stages for using documentation
        1. Searching
        2. Understanding
        3. Applying