

**SOFT3302/3602
Software Quality
Assurance**

Usability Testing

Michaelmas 2007 SOFT3302 Lecture 12 1

Usability

ISO 9126

- *A set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users.*

We need to create products that are

- *Easy to learn and use*
- *Satisfying to use*
- *Valued by users*

Michaelmas 2007 SOFT3302 Lecture 12 2

Why?

One company was awarded \$USD286 000 in regular damages and \$USD2 000 000 in punitive damages for a computer system that didn't perform as expected [Cha91]

A \$3 000 000 project failed because the users flatly refused to use the application because it was too difficult to learn and unusable [May99]

One software vendor lost money on a product because the cost of customer support exceeded their profit margin [May99]

A genuinely usable system will gain a leading edge in the marketplace

Michaelmas 2007 SOFT3302 Lecture 12 3

Designing for Usability

‘Test with real users early in development and frequently thereafter’

- *First Law of Usability Engineering [HFR]*
- *Q.v. agile development methodology*

Usability requirements should be part of the initial requirements definition

Key stakeholders should be involved in the design process early

Ongoing assessment of usability status provides timely feedback

Michaelmas 2007 SOFT3302 Lecture 12 4

Common Usability Principles

Provide feedback

- *Always show system status*

Provide undo/escape mechanism

Fit the workflow

- *Minimal effort should be needed*

Provide sensible defaults

Provide ubiquitous help

Be consistent

Michaelmas 2007 SOFT3302 Lecture 12 5

User-centred Design Techniques

Participatory design

Focus group research

Surveys

Design walk-throughs

Paper-and-pencil evaluations

Expert evaluations

Beta-testing

Formal usability testing

Michaelmas 2007 SOFT3302 Lecture 12 6

Usability Testing Methodology

- Develop problem statements
- Choose a representative user sample
- Model the actual work environment
- Observe the users' interaction with the model
- Collect quantitative and qualitative data
- Formulate recommendations for improvement

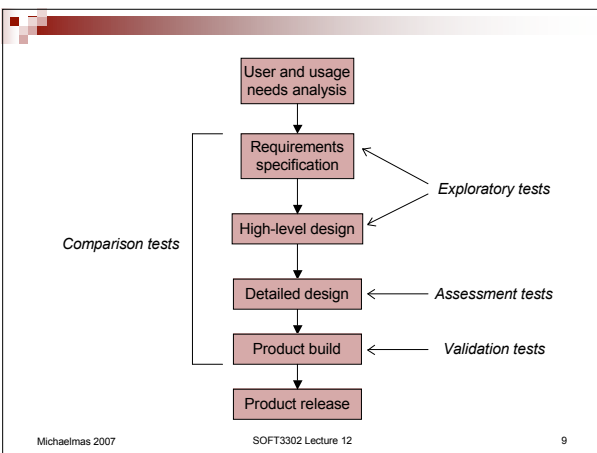
Michaelmas 2007 SOFT3302 Lecture 12 7

Usability Test Types

- Exploration
 - Collaborative, evaluate design
- Assessment
 - Evaluative of explorative tests, lower level
- Validation
 - Products usability
- Comparison
 - Compare two or more designs

<http://pages.cpsc.ucalgary.ca/~saul/wiki/pmwiki.php/CPSC681/BackgroundReadingList>

Michaelmas 2007 SOFT3302 Lecture 12 8



Basic User Profiles

Attributes of a generic user

- Age*
- Gender*
- Native tongue*
- Level of education*
- Computer literacy*
- Occupational experience*

Michaelmas 2007 SOFT3302 Lecture 12 10

Customising User Profiles

Customise profiles with attributes (skills, background, &c.) specific to the product

- Specific product experience*
- Rôle in relation to the product*
 - Operator, manager, &c.*

Identify the frequency distribution of the users, e.g.

- 75% have intermediate computer experience*
- 80% are male*

Michaelmas 2007 SOFT3302 Lecture 12 11

Additional User Characteristics

Learning style

- Read then do*
- Try then read*
- Learn by doing*

Left/right handedness

Attitudes to technology, type of product, &c.

Operating system experience

Types of interaction used

- GUI vs command-line*

Michaelmas 2007 SOFT3302 Lecture 12 12

Acquiring User Profile Data

- Product specifications
 - A good spec. will include a description of the intended audience including rôles
- Marketing/design studies
 - Market research studies; task analysis studies; ...
- Product manager
 - If no formal design is available a product manager may have preliminary reports or an idea/feeling/impression of the user base
- Help desk staff/manager
 - Help desk calls may indicate which users are having the most difficulty with an existing or similar product
- Directly from users
 - Surveys/calls to existing customers
 - Can be used to verify other available data

Michaelmas 2007 SOFT3302 Lecture 12 13

Rôles in Usability Testing

- Test monitor
- Data logger
- Timers
- Video recording operator
- Technical expert
- Test observers
- Other miscellaneous...

Michaelmas 2007 SOFT3302 Lecture 12 14

Qualities of a Good Test Monitor

- Understanding of usability engineering
- Quick learner
- Good rapport with participants
- Excellent memory
- Good listener
- Comfortable with ambiguity
- Flexible
- Long attention span
- Able to focus on 'the big picture'
- Good communicator
- Good organiser

Michaelmas 2007 SOFT3302 Lecture 12 15

Common Problems for Test Monitors

- Leading rather than enabling
- Too involved in data collection
- Too knowledgeable
- Too rigid with the test plan
- Inhospitable to participants
- Jumping to conclusions

Michaelmas 2007 SOFT3302 Lecture 12 16

Usability Test Plan

Essential process for clarifying the reasons for conducting the test

Defines the methods of conducting the test, the metrics to be used and what will be reported

Should include a schedule and costs for the test

- Often vulnerable to cut-backs due to poor scheduling or poor cost estimation

Michaelmas 2007 SOFT3302 Lecture 12 17

Test Plan Outline

Table of contents

- Purpose
- Problem statement/test objectives
- User profile
- Method (test design)
- Task list
- Test monitor rôle
- Test environment/equipment
- Evaluation measures
- Report contents and presentation
- Schedule
- Resource allocation and costings

Michaelmas 2007 SOFT3302 Lecture 12 18

Purpose

High level objectives are permissible, e.g.

- *Investigating the high number of service calls for a particular product*
- *New policy states that all products must be tested before release*

More specific goals are desirable, e.g.

- *Determining whether the documentation can compensate for known problems with the interface*
- *Identifying the nature of the problem with the control panel (which has been the subject of many complaints) and how it may be fixed*

Avoid vague or inappropriate goals, e.g.

- *Everyone is into usability testing nowadays*
- *Is there a need for product Y?*

Michaelmas 2007 SOFT3302 Lecture 12 19

**Problem Statement/
Test Objectives**

Beware vague test objectives, e.g.

- *Is the product usable?*
- *Is the product ready for release?*

Test objectives should produce measurable questions to which clear answers can be found, e.g.

- *What obstacles prevent completion of setup tasks?*
- *Can end users install the software using the quick setup guide?*
- *Is the response time a cause of user frustration or error?*
- *Are all buttons on the control panel able to be used correctly without documentation?*

Michaelmas 2007 SOFT3302 Lecture 12 20

Developing a User Profile

Characterise the target population
Identify each user category in terms of

- *Size of population*
- *Importance*

Assess the availability of representative users

- *Whence can users be recruited?*
- *How many are required?*
- *How many will the budget support?*

Michaelmas 2007 SOFT3302 Lecture 12 21

Divide Users into Categories

Users may be distinguished by

- Experience level: *novice, intermediate, expert*
- Organisation rôle: *operator, supervisor, manager, ...*
- Interface to the system: *MacOS, Gnome, KDE, Windows, ...*

Define criteria for each group, e.g.

- Novice: *< 6 months' experience*
- Intermediate: *6–24 months' experience*
- Expert: *> 2 years' experience*

Balance the number of participants according to the importance of their category, especially if numbers are limited

- *In a study with 8 participants, include 6 operators and 2 managers*

Always include a few 'least competent users'

- *Helps highlight important usability issues*
- *Particularly in the realm of assumed knowledge*

Michaelmas 2007 SOFT3302 Lecture 12 22

Test Design

An overview of the testing process covering from when the participant arrives until when they leave

Essential formalisation process

- *Reveals necessary equipment*
- *Identifies necessary rôles*

Supports testing by multiple test monitors

Constraints

- *Time*
- *Money*
- *Management backing*
- *Development team support*
- *Ability to acquire participants*

Michaelmas 2007 SOFT3302 Lecture 12 23

Test Design

Independent groups

Within-subject testing

Testing multiple product versions

Testing multiple user groups

Michaelmas 2007 SOFT3302 Lecture 12 24

Test Design

Use an adequate number of participants

- *A minimum of 10–12 participants per condition for formal tests*
- *4–5 participants still cover 80% of most major problems*
- *Less experienced testers require more participants to get the same coverage*

Consistency is essential

- *Use scripts for instruction; do not ad lib*
- *Use checklists to ensure all tasks are done*
- *Use the same test monitor for all sessions if possible*

Conduct a pilot test

- *Essential for debugging the experimental design*
- *Ensures all materials are in place*

Michaelmas 2007 SOFT3302 Lecture 12 25

Test Design

Keep tests simple

- *A small but reliable test is worth far more than a sophisticated but suspect test*
- *Test only one or two variables at a time*
- *Use simple tasks*

Confirm the characteristics of the participants

- *Use background questionnaires when testers arrive: they may not be the person you were expecting*

Note any problems with a test

- *Some events may invalidate the results of a test*
- *Others may be able to be taken into account*
- *Accurate notes of such events will be essential to this decision*

Michaelmas 2007 SOFT3302 Lecture 12 26

Developing the Task List

Give initial outline of the testing activity, include

- *A brief description of the task*
- *Required materials*
- *Machine/system state*
- *Specification of successful completion criteria*
- *Maximum time-limits for each task*

Aim to expose usability problems indirectly

- *Do not draw attention directly to the feature under test*
- *Set up a task which involves these features in their normal context of use*
- *Incorporate critical usability factors into the successful completion criteria*

Michaelmas 2007 SOFT3302 Lecture 12 27

Ranking Task Lists

Covering all test objectives adequately with a minimum number of tasks that are simple enough to be presented is difficult
Risk analysis is essential to the process
Risk factors include

- *Frequency: aim to cover the 75% of functions that are used most frequently before covering more obscure functions*
- *Criticality: identify which faults will result in support line class, data loss, damage to machines, loss of reputation, &c.*
- *Vulnerability: identify those tasks which you already suspect will have usability flaws*

In the 'real world', you may test functions in order of readiness and availability

Michaelmas 2007 SOFT3302 Lecture 12 28

Test Environment/Equipment

Not all equipment required for conducting the test task

- *Desk, PC, software (including version), peripherals, hardware, stationery, &c.*

Attempt to model the normal working environment as closely as possible; may involve

- *Lighting conditions*
- *Space restrictions, degree of clutter, &c.*
- *Noise conditions, e.g. ringing 'phones*

Note all equipment required for data collection, such as

- *Audio/video recording/monitoring equipment*
- *Data logging/screen capture equipment*
- *Observation forms and checklists*

Michaelmas 2007 SOFT3302 Lecture 12 29

Example: Microsoft Games Studios [Nei07]



Michaelmas 2007 SOFT3302 Lecture 12 30

Example: Microsoft Games Studios

The diagram shows three labs, Lab A, Lab B, and Lab C, arranged in a row. Each lab contains a grid of workstations. Each workstation is represented by a small icon of a computer monitor and keyboard. The labs are separated by low walls or partitions. The layout is symmetrical and organized.

Michaelmas 2007 SOFT3302 Lecture 12 31

Evaluation Measures

Measures must be valid and reliable
A valid measure correlates strongly with the objective being tested

- Words/minute is a valid measure of typing speed; keys pressed/minute is not*

A reliable measure will give the same results repeatedly under the same conditions

Measures may be quantitative or qualitative

- Rating scales are quantitative*
- Comments are qualitative*

Measures may be objective or subjective

- Time take to complete a task is objective*
- Perceived difficulty rating is subjective*

Michaelmas 2007 SOFT3302 Lecture 12 32

Performance Measures

- Time to complete each task
- Number/percentage of tasks completed correctly
- Number/percentage of tasks completed incorrectly
- Time required to access information in the manual/help system
- Time required to recover from errors
- Count of all errors
- Count of incorrect menu choices
- Count of incorrect icon selections
- Count of user manual accesses
- Count of visits to the index
- Count of visits to the table of contents

Michaelmas 2007 SOFT3302 Lecture 12 33

Preference Measures

Ratings and rationale concerning

- *Usefulness of the product*
- *How well the product matched expectations*
- *Appropriateness of product functions to the user's tasks*
- *Ease of use overall*
- *Ease of learning overall*
- *Ease of setup and installation*
- *Ease of accessibility*
- *Usefulness of the index, table of contents, &c.*

Preference and rationale for

- *One prototype vs another*
- *This product vs another*
- *One conceptual model vs another*

Michaelmas 2007 SOFT3302 Lecture 12 34

Choosing Suitable Metrics

Depending on the goal of the evaluation, particular aspects of usability may be the focus

- *Understandability*
 - Attributes of software that bear on the users' effort for recognising the logical concept and its applicability
- *Learnability*
 - Attributes of software that bear on the users' effort for learning its application (e.g. *operation control, input, output*)
- *Operability*
 - Attributes of software that bear on the users' effort for operation and operation control

Michaelmas 2007 SOFT3302 Lecture 12 35

Preparing Test Material

Test materials need to cover every aspect of the testing process

They include:

- *Screening questionnaire*
- *Orientation script*
- *Background questionnaire*
- *Data collection instruments*
- *Non-disclosure agreement, tape consent form*
- *Pre-test questionnaire*
- *Task scenarios*
- *Post-test questionnaire*

Michaelmas 2007 SOFT3302 Lecture 12 36

Screening Questionnaire

Based on the user-profile, but focuses on the distinctive characteristics relevant to the product

- E.g. *direct experience with a related product*

May be used on paper or over the 'phone

- Must be easy to administer*
- Place important disqualifying questions early in the order*

May also be administered over the web

- E.g. *for screening beta test users*

Michaelmas 2007 SOFT3302 Lecture 12 37

Screening Questionnaire: Example

Do you work with a computer on a regular basis or plan to purchase one within the next 3 months? **Yes**

How many printers have you previously purchased? **1**

Have you ever purchased a laser-printer or do you use one on a regular basis? **No**

When printing, how many different typefaces do you use at one time? **3 or less**

Of your total time spent using a computer, please indicate the percentage of time you spend using each application

- Word processing < 80%*
- Spreadsheet < 80%*
- Database <10%*
- Desktop-publishing <5%*

If you currently use a computer and printer, do you require a wide-carriage printer or one that can print multi-part forms? **No**

Michaelmas 2007 SOFT3302 Lecture 12 38

Orientation Script

Used to let a participant know what he is there for

Must be delivered *verbatim* to each participant so that

- Each participant receives the same information prior to the test*
- The test monitor is less likely to project his own feelings about the test onto the participant*
- There is consistency between different test monitors*

Michaelmas 2007 SOFT3302 Lecture 12 39

Orientation Script Format

Make introductions

- Do not indicate that anyone present has worked on the product as this may discourage honest opinion sharing

Offer refreshments

Explain why the participants are here

Describe the environment

- Point out various pieces of equipment such as one-way mirrors, video cameras, &c.

Explain what is expected of the participants

- Encourage them to perform as they normally would
- Do not make comments about 'how easy this will be'

Assure the participants that they are not being tested

- Don't be surprised if they don't believe this

Michaelmas 2007 SOFT3302 Lecture 12 40

Orientation Script Format

Explain any unusual requirements

- E.g. 'thinking out loud' protocol

Mention that it is okay to ask questions at any time

- But note that you may not always answer them

Give an opportunity to ask questions

- Ensure the participants have understood everything so far

Hand out any forms that need to be completed

- E.g. non-disclosure forms, tape consent forms

Michaelmas 2007 SOFT3302 Lecture 12 41

Background Questionnaire

An expanded form of the user profile

Include questions on experience with particular software packages or with particular tasks if that is relevant

Don't use open-ended questions

- Offer multiple-choice answers
- Use ranged values
 - Age: 18–29, 30–39, 40–49, 50+

Michaelmas 2007 SOFT3302 Lecture 12 42

Task Scenarios

Provide realistic scenarios

- *Include contextual information and motivations for each task*

Sequence task scenario actions in the order in which they would most likely be performed

- *If linear order is not important vary it to avoid bias*

Match task scenarios to the experience of the participants

Avoid using jargon or give-away wording

Include a substantial amount of work in each scenario

- *Set tasks out as a goal to be achieved*
 - *'Edit the document "Announcement" so that it is sent to your boss instead of the current recipient'*
- *Don't give a step-by-step description of how to achieve the goal*
 - *The point of the test is to see if the participant can work it out himself*

Michaelmas 2007 SOFT3302 Lecture 12 43

Task Scenario: Example

You have recently ordered a brand new printer over the 'phone and are waiting to receive it *via* courier at any time. Presently, the courier from Associated Couriers will arrive, drop off the product and have you sign for it. Once you sign for it we will begin our session. Please set up the printer in the same manner as you normally would at home. If you normally use instructions please do so today. If you normally do not use instructions until you get stuck, work that way today. Signal me when you reach the point at which you feel ready to print your first document. Any questions before we begin?

Michaelmas 2007 SOFT3302 Lecture 12 44

Data Collection

Review of metrics

Manual data collection

User generated data

On-line data collection

Fully-automated data-loggers

Michaelmas 2007 SOFT3302 Lecture 12 45

Performance Measures

- Time to complete each task
- Number/percentage of tasks completed correctly
- Number/percentage of tasks completed incorrectly
- Time required to access information in the manual/help system
- Time required to recover from errors
- Count of all errors
- Count of incorrect menu choices
- Count of incorrect icon selections
- Count of user manual accesses
- Count of visits to the index
- Count of visits to the table of contents

Michaelmas 2007 SOFT3302 Lecture 12 46

Example: Microsoft Games Studios

8000 gamers run through lab each year
Recently used for *Halo 3*

Why?

- Large amounts of money involved
- Particular interest with *Halo series and Xbox*

Analysed 3000 hours of gameplay across 600 participants [Tho07]

Michaelmas 2007 SOFT3302 Lecture 12 47

Preference Measures

Ratings and rationale concerning

- Usefulness of the product
- How well the product matched expectations
- Appropriateness of product functions to the user's tasks
- Ease of use overall
- Ease of learning overall
- Ease of setup and installation
- Ease of accessibility
- Usefulness of the index, table of contents, &c.

Preference and rationale for

- One prototype vs another
- This product vs another
- One conceptual model vs another

Michaelmas 2007 SOFT3302 Lecture 12 48

Manual Data Collection

Form based
Provide support for manual recording of counts, observations, user comments, &c.

Use codes to simplify recording

- B* begin task
- E* end task
- P* prompted by test monitor
- M* exceeded maximum time limit
- X* incorrect action
- R* reading manual

Use specialised forms for different recording tasks, e.g.

- Time recording
- Recording participant comments
- Observing participant actions

Michaelmas 2007 SOFT3302 Lecture 12 49

User-Generated Data

Can be integrated into the task itself

- Questionnaires that appears on-screen at the completion of a task

Can be done using paper-based forms

Most suited to subjective data collection

- Participants may be unaware/inaccurate regarding measures such as task duration

Beware of

- Biased wording
- Jargon

Michaelmas 2007 SOFT3302 Lecture 12 50

On-line Data Collection

Computer-assisted form of manual recording

- Event times are automatically recorded
- Event types are often entered using codes
- Observations may be typed in along with event entries

May be customised for the task-at-hand

Pen-/PDA-based solutions are particularly attractive

Benefits

- Improved accuracy for timing data
- Data is already entered for analysis

Costs

- Requires a high degree of concentration
- May be restrictive for qualitative note-taking, extensive observations, &c.

Michaelmas 2007 SOFT3302 Lecture 12 51

Fully Automated Data Loggers

Integrated with the software platform
Record numerous aspects of user behaviour in detail

- Keystrokes
- Mouse movements
- Menu accesses
- Screen accesses
- Total time spent on each screen or in each mode

No substitute for human observer

- Does not observe behaviours such as frustration, confusion, &c.

Analysis of keystroke data can be extremely time consuming

Michaelmas 2007 SOFT3302 Lecture 12 52

Pre-Test Questionnaire

Aimed at eliciting information about the product prior to using it

- Attitudes and first-impressions about ease-of-use
- Opinions about the utility or value of the product

Secondarily aimed at

- Establishing the participant's prerequisite knowledge
- Determining the participant's inclusion in one group or another

Michaelmas 2007 SOFT3302 Lecture 12 53

Example: Halo 3

Run through Microsoft Games Studios testing lab
Take snapshots of progress automatically as people play the game
Also records video footage of gameplay and links these to individual progress reports

- May be reviewed later to examine issues in detail

[Tho07]

Michaelmas 2007 SOFT3302 Lecture 12 54

Post-Test Questionnaire

Use problem statements as the basis for content

- *Answers to questions should help you move towards a design decision*

Administer questionnaires before participants forget their experience

- *Long tests (> 2hr) can be broken into modules with questionnaires interspersed*

Focus on eliciting non-observable data

- *Subjective preferences, opinions, &c.*

Design questions for eliciting simple, brief responses

- *Make use of multiple choice and response scales*
- *Don't use open-ended questions*

Pilot test the questionnaire

Michaelmas 2007 SOFT3302 Lecture 12 55

Questionnaire Design Techniques

Linkert scales

- *Strongly disagree | Disagree | Neither | Agree | Strongly Agree*

Semantic differentials

- *Simple.....3...2...1...0...1...2...3.....Complex*

Fill-in questions

- *I found the following aspects of the manual particularly easy to use (please list 0-3 aspects):*

Check-box questions

- *Please check the statement that mode closely approximates your feelings about the documentation*
 - I always use documentation
 - I use documentation only when I must
 - I never use documentation

Michaelmas 2007 SOFT3302 Lecture 12 56

Conducting the Test

Monitor the session impartially

Be aware of your voice and body language

Test each participant as an individual

Don't 'rescue' participants who are struggling

Ensure participants are finished before move to the next task

Keep the atmosphere 'light'

Interact appropriately

Assist the participants only as a last resort

Michaelmas 2007 SOFT3302 Lecture 12 57

Impartial Monitoring

Present the product neutrally

- *If you show an opinion of the product, it may affect the participants' performance*

Do not react to 'mistakes' or correct behaviour

- *You are an observer, do not pass judgement*
- *Participants will always assume that mistakes are their fault, do not compound this by reacting to their mistakes*

Encourage participants to focus on their own experience of the product rather than what others might hypothetically think

Michaelmas 2007 SOFT3302 Lecture 12 58

Body Language

Body language and tone of voice are subtle but powerful cues to our own attitudes that we ourselves may not notice

- *Moving closer can indicate acceptance*
- *Raising the pitch of one's voice indicates agreement*
- *Lowering the pitch, disagreement*

When observing a participant, be aware of

- *Grunting approval when they perform correctly*
- *Sighing with frustration or boredom*
- *Note taking*

Watch yourself on tape to learn about your behaviour

Michaelmas 2007 SOFT3302 Lecture 12 59

Participants Are Individuals

It is easy to be affected by the performance and comments of the last person you observed

Allow time in the schedule to 'clear your mind' between tests

If you are fatigued, you will miss important behaviours or fail to understand them

Michaelmas 2007 SOFT3302 Lecture 12 60

Don't Rescue Participants

Resist the temptation to help a participant who is stuck

Instead, help them to verbalise their thoughts and feelings

A struggling participant is one of you most valuable assets

- *They are revealing real problems with the product*

Michaelmas 2007 SOFT3302 Lecture 12 61

Ensure Participants Are Finished

Participants are often unsure if they have completed a task successfully

Allow time for them to reach their own decision

- *Do not move them on to the next task prematurely*
- *Beware of making notes as soon as they finish*

Instead, get participants to indicate that they have finished

Michaelmas 2007 SOFT3302 Lecture 12 62

Light Atmosphere

Participants will already be nervous

An overly serious atmosphere will put pressure on the participant and reduce their willingness to comment freely even further

Use humour appropriately

- *Help the participant to have fun, it helps them speak their mind*
- *Join in if they find something amusing about the product*
- *Do not laugh at them*

Michaelmas 2007 SOFT3302 Lecture 12 63

Interact Appropriately

In formal experiments interaction should be minimal
 In exploratory tests, interaction is often essential
 Appropriate interaction is difficult to master

- *Err on the side of minimal interaction*

When probing

- *Focus on the participants expectations not your own*
- *Do not imply the correct behaviour or answer*
- *Do not ask direct or loaded questions, e.g.*
 - How did you get to that screen?
 - Most people find this easy, how about you?
- *Better examples*
 - What are you thinking right now?
 - You seem surprised/puzzled/frustrated
 - Is everything going as expected?
 - How did that differ from what you expected?

Michaelmas 2007 SOFT3302 Lecture 12 64

Assistance as a Last Resort

When the participant is lost or confused

- *If you can wait until they pass the time limit, they are already unsuccessful and your assistance will not unduly affect the test*

When the participant is frustrated and may give up

- *Empathise with their frustration, it may extend their perseverance*

When the product is missing necessary functionality

- *E.g. an error message should appear but doesn't*

When the product malfunctions and needs repair

- *Remove the participant from the room if making the repair would reveal information they should not have*
- *Assure the participant that they have done nothing wrong*

Michaelmas 2007 SOFT3302 Lecture 12 65

How to Assist

Never blame participant for a problem

- *Mistakes are always the fault of the product*

Help the participant clarify their problem

- *Avoid telling him how to do something*

Provide gradually more revealing hints

- *Don't give the whole answer at once*

Be careful not to give hints that will affect future tasks

Michaelmas 2007 SOFT3302 Lecture 12 66

User-Centred Testing

Involving users during development and testing is central to producing truly user-centred software

User-centred testing provides a framework for that involvement

- Ad hoc input is unlikely to be useful

User-centred testing may be applied at various stages with various degrees of formality

Formal user-centred testing is costly and requires a greater degree of expertise to administer correctly

Exploratory/informal user-centred testing is still better than an *ad hoc* approach and may give the best return on investment when used strategically

Michaelmas 2007 SOFT3302 Lecture 12 67

Heuristic Evaluations

Performed by a group of expert evaluators (3–5)

Evaluators

- Assess the product against a list of accepted usability principles
- Initially work independently to avoid influencing each other
- Compare findings and produce a final report
- Perform at least two passes through the interface
 - Familiarisation
 - Detailed assessment of each element

Assistance may be required

- Especially if the product requires domain-specific knowledge

Michaelmas 2007 SOFT3302 Lecture 12 68

Pros and Cons of Heuristic Evaluations

Pros

- Very cost effective
- Can be done early in development
- Less effort/expertise required by internal staff

Cons

- Difficulty finding suitable expert evaluators
- Not as comprehensive as formal testing

Michaelmas 2007 SOFT3302 Lecture 12 69

Neilsen's Usability Heuristics

Visibility of system status

- *The system should always keep users informed about what is going on, through appropriate feedback within reasonable time*

Match between system and real world

- *The system should speak the user's language with words, phrases and concepts familiar to the user rather than system-oriented terms; follow real world conventions making information appear in a natural and logical order*

User control and freedom

- *Users often choose system functionality by mistake and will need a clearly marked 'emergency exit' to leave the unwanted state without having to go through an extended dialogue; support undo and redo*

Consistency and standards

- *Users should not have to wonder whether the different words, situations or actions mean the same things; follow platform conventions*

Michaelmas 2007 SOFT3302 Lecture 12 70

Usability Heuristics

Error prevention

- *Even better than good error messages is a careful design which prevents a problem from occurring in the first place*

Recognition rather than recall

- *Make objects, actions and options visible; the user should not have to remember information from one part of the dialogue to another; instructions for use of the system should be visible or easily retrievable whenever appropriate*

Flexibility and efficiency of use

- *Accelerators—unseen by the novice user—may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users; allow users to tailor frequent actions*

Michaelmas 2007 SOFT3302 Lecture 12 71

Usability Heuristics

Aesthetic and minimalist design

- *Dialogues should not contain information which is irrelevant or rarely needed; every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility*

Help users recognise, diagnose and recover from errors

- *Error messages should be expressed in plain language (no codes), precisely indicate the problem and constructively suggest a solution*

Help and documentation

- *Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation; any such information should be easy to search, focused on the user's task, list concrete steps to be carried out and not be too large.*

Michaelmas 2007 SOFT3302 Lecture 12 72

Strategies for Usability Testing

Must consider trade-off between cost and risk

User-centred testing can be high-cost but give high return

- *Best considered if usability risk is high and budget supports it*

Heuristic testing is lower cost but can be less reliable and/or give lower coverage

- *Best considered if*
 - Budget is limited
 - Access to representative users is limited
 - Size/risk level does not warrant extensive usability testing

Any usability testing is better than none

- *Usability reviews based on use-cases and conducted at strategic points in the SDLC provides a minimal option requiring low overheads and minimal expertise while still providing something more structured than ad hoc testing*

Michaelmas 2007 SOFT3302 Lecture 12 73

Formulating Recommendations

Compile and summarise data

Analyse data

Develop recommendations

Produce the final report

Michaelmas 2007 SOFT3302 Lecture 12 74

Compile and Summarise Data

Compile data at the end of each test/day

Summarise data using descriptive statistics

- *Mean, median, range and deviation for timing measures*
- *Percentage of successful task completions*
- *Distribution graphs for ranking or limited choice questions*
- *Categorisation of free-form questions*
 - *E.g. topic, positive, negative*
- *Group by product version or user group*

Michaelmas 2007 SOFT3302 Lecture 12 75

Analyse Data

Identify hot-spots

- *Tasks with the highest degree of failure*

Identify user errors and difficulties

Identify the sources of user errors


- *Focus on the product rather than the task*
- *Where the task is complex, look for the primary source of difficulty*

Rank by risk

Use qualitative data to disambiguate results

Michaelmas 2007 SOFT3302 Lecture 12 76

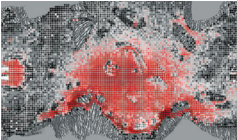
Example: Halo 3



In early tests, players wandered lost around the Jungle level: Colored dots showing player location at five-second intervals (each color is a new time stamp) were scattered randomly. So Bungie fixed the terrain to keep players from backtracking. Sure enough, the dots clustered by color, showing that players were moving smoothly through the map.

Michaelmas 2007 SOFT3302 Lecture 12 77

Example: Halo 3



In April, Bungie found a nagging problem with Valhalla, one of *Halo 3*'s multiplayer levels: Player deaths (represented in dark red on this 'heat map' of the level) were skewing toward the base on the left, indicating that forces invading from the right had a slight advantage. After reviewing this image, designers tweaked the terrain to give both armies an even chance.

Michaelmas 2007 SOFT3302 Lecture 12 78

Develop Recommendations

Focus on high-impact issues/solutions, e.g.

- *Departure from known usability principles*
 - Over programme logic reflects user tasks
 - User is able to navigate easily
 - Navigation rules are consistent
 - User orientation is maintained
 - ...
- *Problems arising from*
 - User expertise levels
 - User expectations, &c.

Ignore 'political considerations' in first draft
Provide short-term and long-term solutions

Michaelmas 2007 SOFT3302 Lecture 12 79

Final Report Format

Executive summary

- *Include major findings and recommendations*

Method

- *Include the setup, user profiles, data collection methods*

Results

- *Quantitative and qualitative results summaries*
- *Raw data can be included in appendices but not in main body*

Findings and recommendations

- *Divide into general and specific*
- *List findings and recommendations for specific item together*

Appendices

- *If including raw data, user profiles, &c. do not include participants names*

Michaelmas 2007 SOFT3302 Lecture 12 80

References

[EUS] European Usability Support Centres,
Loughborough University

- <http://www.lboro.ac.uk/eusc/>

[HFR] Human Factors Research Group,
University of Cork

- <http://www.ucc.ie/research/hfrg/>

[UMT] The Usability Methods Toolbox

- <http://jthom.best.vwh.net/usability/>

Michaelmas 2007 SOFT3302 Lecture 12 81

References

[Cha91] A. Chapanis, *The Business Case for Human Factors in Informatics*, in *Human Factors for Informatics Usability*, B. Shackel and S. Richardson, eds. Cambridge University Press, 1991.

[May99] D. Mayhew, *The Usability Engineering Lifecycle*, Morgan Kaufmann, San Francisco, 1999.

[Nie94] *Heuristic Evaluation in Usability Inspection Methods*, John Wiley & Sons, New York, 1994.

Michaelmas 2007 SOFT3302 Lecture 12 82

References

[Nei07] <http://www.useit.com/alertbox/multiple-user-testing.html>

[Rub94] J. Rubin, *Handbook of Usability Testing*, John Wiley & Sons, New York, 1994.

[Tho07] C. Thompson, *Halo 3: How Microsoft Labs Invented a New Science of Play*, Wired Magazine, 2007, 15:9, http://www.wired.com/gaming/virtualworlds/magazine/15-09/ff_halo

Material adapted from 2004 SOFT3103 course.

Michaelmas 2007 SOFT3302 Lecture 12 83
