Usability Heuristics

There are 10 usability heuristics that you must be able to explain and give examples of in the exam. These heuristics are broad rules, not specific usability guidelines.

Visibility of system status – The system should provide appropriate feedback within reasonable time.

Match between system and the real world – The system should communicate to the user with words, phrases and familiar concepts to the user rather than system-oriented terms. Information should appear in a natural and logical order.

User control and freedom – Support undo and redo and have an emergency exit button if a user goes to an unintended location.

Consistency and standards – Users shouldn't have to worry whether different words, situations or actions mean the same thing. Don't be ambiguous.

Error Prevention – You should design your product carefully to avoid errors.

Recognition rather than recall – Make objects, actions and options visible so that the user doesn't have to remember information.

Flexibility and efficiency of use – Your system should cater to experienced and inexperienced users.

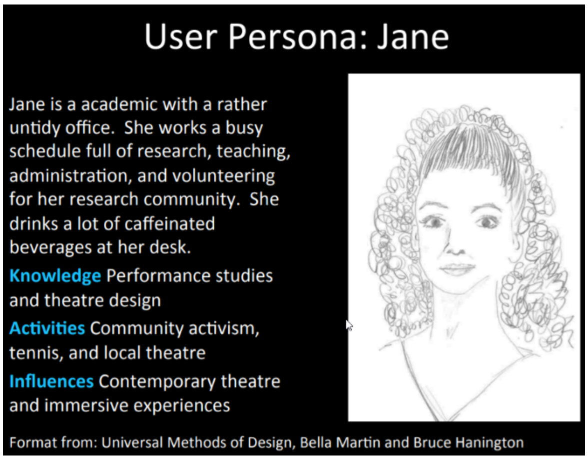
Aesthetic and minimalist design – Prevent the use of irrelevant data as it drowns out the relevant data.

Helpful error messages – Show errors in plain text with solutions so that users are able to understand what went wrong and how to fix it.

Help and documentation – It is good to keep documentation so that users can search for instructions if they are unsure on how to complete a task.

Personas

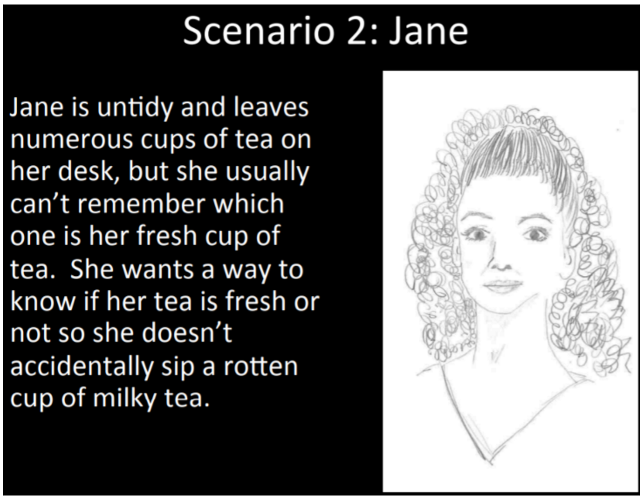
Personas can be used to understand your interface from the perspective of an average user. They are used to describe a certain type of user and their behaviours in a profile. Remember: you can't design for everyone. To sketch a persona, you need a **name**, **photo/sketch** and a **narrative** which can include relevant key features of their life, goals and behaviours.



Here is an example taken from Julie's lecture notes. She gives a **name** (Jane), a **sketch** of the person and tells us about her job and hobbies.

Scenarios

Scenarios are used to identify potential users, their common tasks and the context where interaction will occur. They also identify the users' goals and the steps needed (by user and the system) to achieve the goal. You can either create an entire storyboard with sketches to explain this, or just a simple list of bullet points outlining the scenario.



Here is another example from Julie's lecture. She uses Jane in a scenario to identify issues they might face in their daily life and how the product may be able to help.

Sketching Techniques

For some reason we must know sketching techniques (for computing :/):

Remember to use a pencil!

Types of lines – straight and curved lines are created using the elbow as a pivot or by using the 20 degrees angle grip.

Grip – The higher the angle of your grip, the darker the lines.

Independent and Dependent Variables

Independent Variable – what is manipulated

Dependent Variable – what is measured

Independent and Dependent Samples

Independent Samples – paired measures for a set of unrelated items

Dependent Samples – paired measures for a set of related items

Parametric and Non-Parametric Data

Each statistical test comes with several assumptions, and often one assumption is that the data is normally distributed.

In many cases, data from user studies is not normally distributed (68% of area under the curve is within 1 standard deviation)

Interaction

Interact – making a computer do what we want

Formally – driving a system into a state compatible with user intentions

Intention – what a user wants to do

State – what the system currently believes about what a user wants to do

Signals

Computers interact with humans by sensing the physical world. These signals come from the input device's sensors. These signals can be corrupted by noise and irrelevant observation.

Feedback

Control requires feedback. A system without feedback is uncontrollable and unusable.

The state that is fed back must offer opportunities for action – it must be possible to use the feedback to make useful things happen.

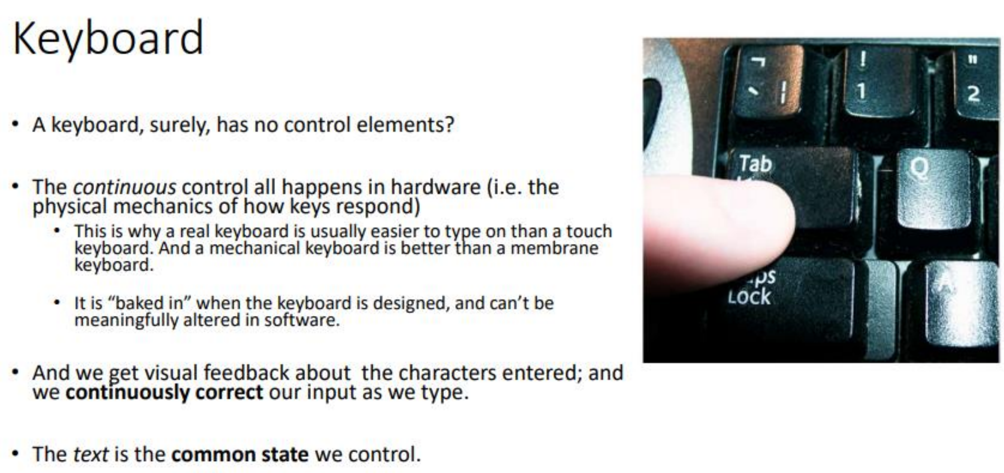
And it must represent an internal state of the system being controlled – a common "view on the world"

Mouse Control Example

A mouse:

- Uses a cursor to mediate control
- The cursor is the state which represents belief about what you are going to do
- Feedback is presented visually, using a spatial metaphor
- Displacement of the hand is related to displacement of the cursor
- But not 1:1!
- Measurement (signal) is of camera images of the surface below

Keyboard Example



Touchscreen

Continuous signals from an array of capacitive sensors.

Direct mapping: relation of finger to display determines targeting

Control is over visual finger position relative to display

Position recovered from sensor image

Compatibility of dynamics

For a human to control something, the internal (controlled) state needs to be compatible with their dynamics; that is the motion of their limbs and their perceptual capabilities.

Interaction is (almost) universally carried out by limb movement

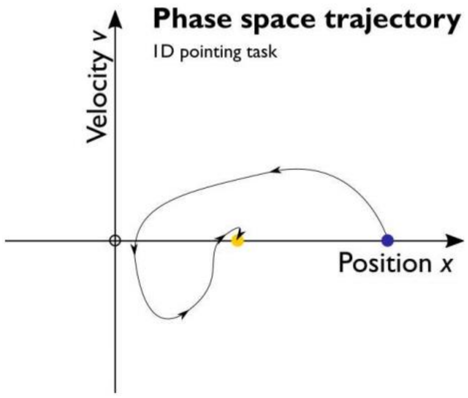
Bad examples: Imagine a mouse cursor that moved faster than your eye could track or a keyboard that updated so slowly it took a full minute to see what you had typed.

Smoothing

Smoothing eliminates noise from signals that users are physically incapable of smoothing out themselves. This can make a signal more useful for control at the expense of delay.

Phase space

Phase space just means representing both time derivatives (velocity) and original values (position) as points in space.



Thresholding

Most UI tasks are discrete actions: print a document, submit a form, delete a file, copy a slide.

But signals are continuous (this is true even of devices like keyboards)

We need to threshold continuous values to trigger actions. All interaction comes down to thresholding at some point.

Discrete actions

Now we can complete actions. For example, I can turn on my phone. Doing this changes the actions I can then do next. I can open my email app, then I have a new set of actions such as composing a new email. In each case, the signals are interpreted differently.

The thresholding is typically the same (e.g finger touch) but the consequences are different.

Finite State Machines

States – A state represents a continuing state of the world

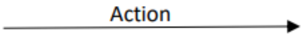
Action – An action or transition allows movement between states; an event.

The system moves from one state to another when an action occurs. The states persist over some period of time. The actions are assumed to be instantaneous and the state machine which specify what the next state will be for every valid action.

State Diagrams

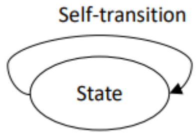
Ellipses are states

Arrows are actions



Every action for a given state must be drawn

An action can go from a state back to a state, this is called a *self-transition*

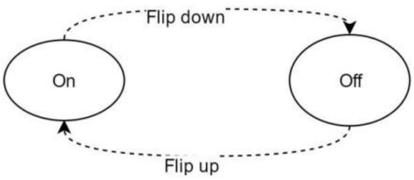


A state without any arrows leaving is a terminal state: you are stuck if you get there.

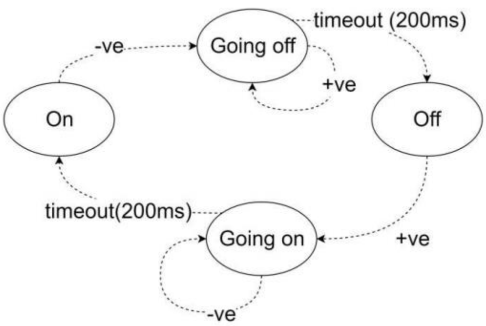
A state with no arrows arriving is unreachable: no action can take you there.

State Diagram Examples

A light: Two states, two actions

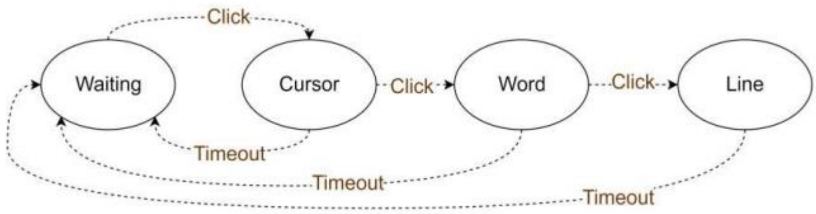


Debounced button [DEMO]



Timing

Timeouts are a common type of transition that occurs after a certain time has elapsed. The timeout is reset every time the state is entered.



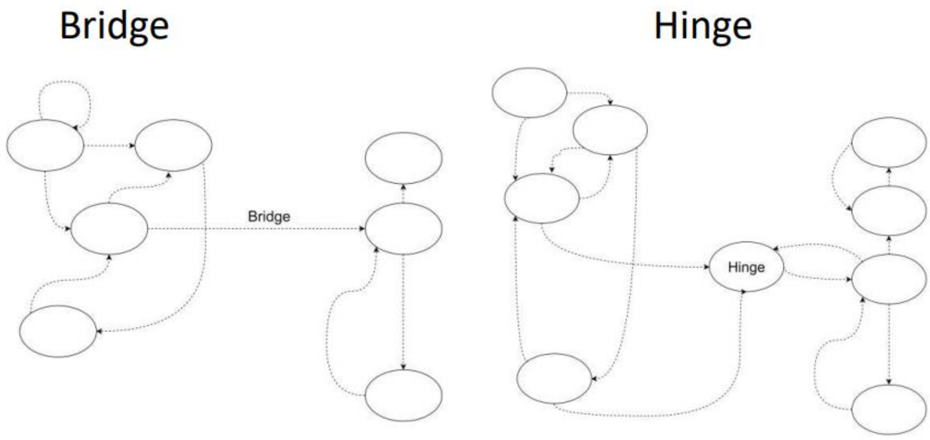
Railroading

States that cannot change an outcome are pointless

A state where the is only one incoming and one outgoing action is likely to either be a waiting action or pointless.

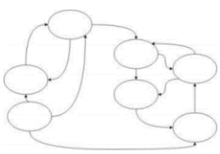
Bridge and Hinge

An event that links together two otherwise disconnected parts of an FSM is a bridge. A state that is the only connection between two disconnected parts is a hinge. Bridges and hinges are bad because they are bottlenecks. A user has to get to the bridge/hinge to "ford the river" into the other side of the graph. Imagine if you had to get to one specific web page to get the rest of the content of a site.

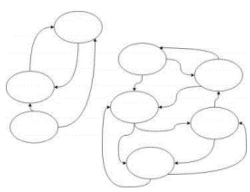


Traps and Disconnected States

A trap is a collection of states that have an incoming transition but no outgoing transitions. There is a way in, but no way out.



A state machine is disconnected if it has collections of states with no transitions between them. The options available depend on where the state machine started.



Fuzz Testing

Fuzz testing is a way of testing systems by rapidly giving them random inputs instead of user inputs. This can be seen as simulating the random movement of a user. From the output of fuzz testing, we can look at properties of the interaction:

- Will users get stuck in traps?
- Do users end up in one state way too often?
- Or are some states very hard to reach (require a very precise sequence of commands, like the Konami code)?

Note that this testing requires no users, runs very fast, and doesn't depend on specific interface design choices. We can apply it to state machines abstractly without defining how transitions will be made in practice.

States and Modes

A state precisely specifies the effect of all possible actions

A mode specifies the effect of some subset of actions e.g a watch has an *alarm set* mode where the buttons do a different thing than the *time set* mode.

Interfaces with modes can be thought of as being made up of multiple state machines, with transitions between them

Modes are "groupings of states". These can still represented as a single FSM, but we conceptually consider groups of states to be modes.

Usability Testing

This is defined as the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use. There are generally two types of usability tests:

Formative tests – finding and fixing usability problems by providing immediate feedback to improve learning

Summative tests – describing the usability of an application using metrics and evaluating what was learned

Most usability tests are formative.

There are two types of summative tests:

Benchmark – to describe how usable an application is relative to a set of benchmark goals. Benchmark tests provide input on what to fix in an interface and provide an essential baseline for comparison of post-design changes.

Comparative – involves more than one application. This can be a comparison of a current version with a prior version of the product. Different sets of users can work with different versions and the results can be compared.

Sample Sizes and Representativeness

Samples do not need to be large to use statistics and interpret quantitative data. You should plan on finding people who relate to the design rather than having a lot of users who are uninterested. There are concerns with small sample sizes are the sample isn't a true "representative" of the parent population. However, sample size and representativeness are different concepts. You can have a sample size of 5 represent a parent population. For example, if you want to test new hiking boots you would get a group of 5 hikers rather than 1000 tennis players.

Data Collection

Data can be collected in a moderated lab. The moderate will observe what the user does when attempting tasks. This can be very costly. Recently, remote moderated and unmoderated sessions have become more popular as users attempt tasks from their own computer at home while a moderator watched over screen-sharing software.

Data can be actively or passively collected from the user. For example, using automatic logging to collect data passively or prompts/questionnaires to gather data actively.

Completion Rates

A percentage of how many people completed the task successfully. 8 out of 10 is 80% completion rate or a 20% failure rate.

Usability Problems

If a user encounters a problem while attempting a task and it can be associated with the interface, it's a user interface problem (UI problem). These will usually have a severity level that takes the impact on the user into account.

Table 2.1 Example of a UI Problem Matrix

	User 1	User 2	User 3	User 4	User 5	User 6	Total	Proportion
Problem 1	X	X			X	X	4	0.67
Problem 2	X						1	0.167
Problem 3	X	X	X	X	X	X	6	1
Problem 4				X	X		2	0.33
Problem 5					X		1	0.167
Total	3	2	1	2	4	2	**14**	**p = 0.47**

Note: The X's represent users who encountered a problem. For example, user 4 encountered problems 3 and 4.

Measuring Task Time

Task completion time – the time of users who completed the task successfully

Time until failure – time on task until users give up or complete the task incorrectly

Total time on task – the total duration of time users spend on a task

Survey Data

Surveys are a great way to get attitudinal data from customers. They can be open ended, yes/no or rating between strongly agree and strongly disagree. From the results you can compute the mean and the standard deviation and use these to compare the results with other survey results of other products.

Table 2.2 Mapping of the Five Classic Likert Response Options to Numbers

This →	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Becomes This →	1	2	3	4	5

Requirements Gathering

Another key function of user research is to identify features and functions of a product. Behaviours can be put in a table and you can record which users exhibited each behaviour.

Types of Data

Two major types of measurements: quantitative and categorical.

Quantitative – data falls on a spectrum from continuous to discrete-binary (e.g task time, number of usability problems, rating-scale)

Categorical – things like gender, operating system, usability problem type

Discrete data have finite values. You can count them. Continuous data technically have an infinite number of steps. The number of usability problems would be discrete, as there are a finite number of usability problems.

You can tell the difference between discrete and continuous data because discrete data may have the phrase "number of… " in front of it. (e.g number of errors etc.)

You can convert categorical data into discrete quantitative data. You can give responses numerical values, for example, pass/fail can be 1/0 and a questionnaire can be values from 1 to 5 (Likert Scale).

Quantitative data can be broken down into: quantitative and categorical; continuous and discrete; nominal, ordinal, interval, ratio.

Populations and Samples

We rarely have access to the entire population of users, so we take a smaller portion of users and take their average statistics (means, standard deviations, medians, other summary values).

The most important thing in drawing conclusions from data is that the sample of users you measure represents the population. No amount of statistical manipulation can correct for making inferences about one population if you observe a sample from a different population. Ideally, you should select your sample randomly from the parent population. In practice, this can be very difficult due to (a) issues in establishing a truly random selection scheme or (b) problems getting the selected users to participate. It's always important to understand the potential biases in your data and how that limits your conclusions.

Measuring Central Tendency

Mean – Calculate the average. It's a way of summarizing the middle value.

Median – when data isn't symmetrical, like task times, the mean may be influenced by extreme data points. The median is the central value, which can produce a more accurate typical value.

Geometric Mean – defined as the center value. Transfer raw times to log times (using ln()), find the mean and then convert back to original scale (using exp())

Standard Deviation and Variance

This is a measure of the spread of data. This provides an estimate of the average difference of each value from the mean. If, however, you just subtract each value from the mean and take an average, you'll always get 0 because the differences cancel out. To avoid this problem, after subtracting each value from the mean, square the values then take the average of the squared values. This gets you the average squared difference from the mean, a statistic called the variance.

The Normal Distribution

Measures tend to look like a bell-shaped curve when graphed. Over the past century, researchers have found that the bulk of the values in a population cluster around the "head" of the bell-curve. In general, they've found that **68%** of values fall within one standard deviation of the mean, 95% fall within two standard deviations, and 99.7% fall within three standard deviations.

z-scores

For example, if you had a friend who had a height of 6 feet 10 inches, intuitively you'd think he was tall. In fact, you probably haven't met many people who are as tall as or taller than him. If we think of this person as one point in the population of adult men, we can get an idea about how unusual this height is. All we need to know is the mean and standard deviation of the population. Using the mean and standard deviation from the North American population (5 feet 10 inches and 3 inches, respectively), someone who is 6 feet 10 inches is 12 inches higher than the mean. By dividing this difference by the standard deviation of three, we get four, which tells us how many standard deviations this point is from the mean. The number of standard deviations is a unitless measure that has the same meaning regardless of the data set. In this case, our friend is four standard deviations above the mean. When used this way, the number of standard deviations also goes by the name z-score or normal score.

Area Under the Normal Curve

The total area under a curve adds up to 100%. It can be used like a pie chart to get slices. Since it is a curve, the area under the graph gets infinitely smaller towards the edges. If you have access to Excel, you can find the percent of area up to a point in the normal curve by using the function =NORMSDIST(z), where z is the number of standard deviations (the z-score). For example, a z-score of 1.28 provides the area from negative infinity to 1.28 standard deviations above the mean and accounts for about 90% of the area. We can use the area under the curve as a percentile rank to describe how typical or unusual a value is. A person who is four standard deviations above the mean would be in the =NORMSDIST(4) or 99.997 percentile in height. Most statistics books include a table of normal values. To find the percentile rank from the table, you find the z-score that's closest to yours and find the area.

Applying the Normal Curve to User Research Data

The examples so far have been mostly about height, weight, and IQ scores—metrics that nicely follow a normal distribution. In our experience, user researchers rarely use these metrics, more typically using measurements such as averages from rating scales and completion rates. Graphs of the distributions of these types of data are usually far from normal. For example, Figure A.8 shows 15 SUS scores from a usability test of the Budget.com website. It is hardly bell-shaped or even symmetrical. The average SUS score from this sample of 15 users is 80 with a standard deviation of 24. It's understandable to be a bit concerned about how much faith to put into this mean as a measure of central tendency because the data aren't symmetric. It is certainly even more of a concern about how we can use the normal curve to make inferences about this sort of data. It turns out this sample of 15 comes from a larger sample of 311 users, with all the values shown in Figure A.9. The mean SUS score of these data is 78. Again, the shape of this distribution makes you wonder if the normal curve is even relevant. However, if we take 1,000 random samples of 15 users from this large population of 311, then graph the 1,000 means, we get the graph shown in Figure A.10. Although the large sample of 311 SUS scores is not normal, the distribution of the random means shown in Figure A.10 does follow a normal distribution. The same principle applies if the population we draw randomly from is 311 or 311 million.

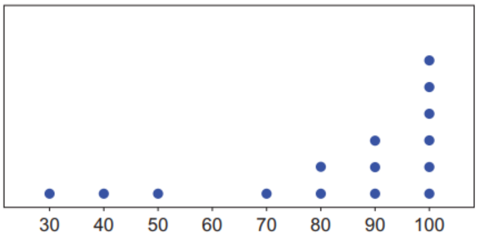


FIGURE A.8

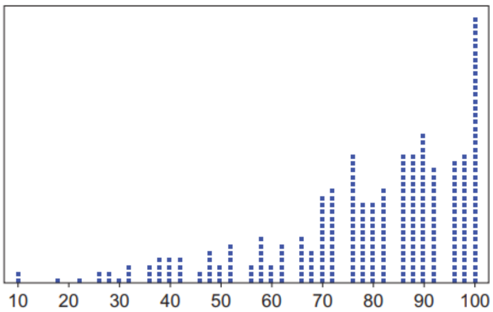


FIGURE A.9

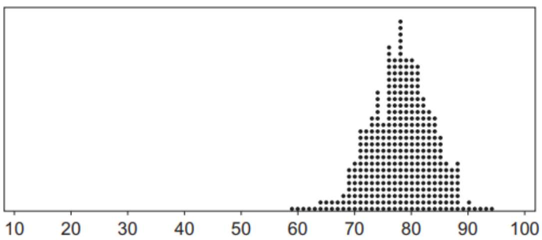


FIGURE A.10

Sources of Data

These include, but are not limited to: interviews, observations, videos, documents, drawings, diaries, group meetings, memoirs, newspapers, historical documents, and biographies. A researcher may combine interview with observation, then perhaps add documents or videos to interviews.

Interviews

There are three types of interviews: unstructured interviews, semi-structured interviews and structured interviews. All are used in qualitative research. Record and transcribe to complete analysis.

Observations

Observations are useful as people may not be aware or able to articulate what goes on during interaction between themselves and others.

Informed Consent

A researcher can never be certain why people agree to be research participants; all a researcher can do is ask potential participants if they are willing to participate in a study and then be sensitive to their nonverbal as well as verbal responses. If the person reading the consent form hesitates after hearing an explanation about the study, it may suggest that the participant is not certain. If, once given further explanation, the person is still hesitant, you are best to drop them from the research. Once a participant agrees, they should sign two consent forms: one copy for the researcher and another for the participant. Participants also have the right to withdraw at any point. All data should be confidential.

How to Run an Experiment

A standard lab experiment has these key stages:

- Provide information to participants about the goals of the experiment and what is involved
- Gather informed consent from participants
- Complete a series of tasks to collect data from participants
- Finish the experiment and provide de-briefing

Experimental Ethics

Participants must provide informed consent and experimenters must not be in a position of authority over participants. Participants understand they can withdraw at any time and are given the contact information of the experimenters.

User Experience

User experience encompasses all aspects of the end-user's interaction with the company, its services, and its products.

Both Usability and User Experience must come together to create immersive experiences.

Qualitative data

Involves interpretation by a researcher, thus making the researcher as much a part of the results as the participant and the data they provide. For example, interviews, observations, and other interpreted data sources.

Grounded Theory

Grounded Theory is a research method where theory is derived from data collection. Theoretical concepts are not used to guide data collection, and initial research questions can be very open ended.

Analysis is done through a three part process: Open, Axial, and Selective Coding.

Open Coding – you read through data and create labels for chunks of data that summarize what you see happening.

Axial Coding – consists of identifying relationships among the open codes.

Selective Coding – From the groups identified, give each a code which represents the key concept for each group. Use these codes to represent your results and re-code the transcript using these codes.

Example:

Open codes	Axial codes	Selective code
Wanting experiential learning; constantly learning; working in a good environment;pioneering social media and easily adapting to change; feeling entitled due to unique qualifications, as compared to previous generations; possessing the personal skills and characteristics needed; being groomed	Believing they are ready to be set loose on accounts	Wanting to make a difference
Craving immediate feedback and being motivated by feeling appreciated; detesting getting called out; receiving verbal encouragement and making observations	Seeking external validation	
Mind reading and expectations for a miracle worker;getting called out; not being heard	Silently blaming employers for failures	
Advocating a work-life balance; being cared for as a whole person; accommodating interests and preferences	Wanting a meaningful experience at work and outside of work	

Bias

When completing evaluations, researchers can introduce bias through the design of the questions.

Bias can be introduced by researching based on the questions they ask and how they phrase the question.

Do you think my mug design is really good?

Vs

What do you think of my mug design?

Ethnomethodology

In contrast to traditional sociology, which was concerned with large scale interaction, ethnomethodology is concerned with individual interactions. Talking is a central part of this.

Breaching Experiments in HCI

Often, we are developing new interactions that don’t have existing practice. This may include:

- Asking people to play in unusual settings
- Asking people to approach unfamiliar technologies
- Asking people to collaborate and share in new contexts