Module 1

1. **Explain the importance and benefits of good User Interface Design**

   Good UI design enhances the user experience, improves productivity and enables users to interact with technology seamlessly. As technology continues to develop, UI designers must keep up with the latest trends and technologies to create interfaces that meet the needs and preferences of users.

   Good UI design incorporates several elements that enhance the user experience.

   - **Consistency:** Consistency in design ensures that users can navigate through different screens and components of the interface easily without getting confused.
   - **Clarity:** Clarity in design ensures that the interface is easy to understand and navigate, even for first-time users.
   - **Responsiveness:** A responsive interface ensures that users can interact with technology in real-time, without experiencing lags or delays.

   OR

   - A well designed interface and screen is important to the users because that's the window for them to view the capabilities of the system. It is the vehicle through which many critical tasks are presented which has a direct impact on the organizations relation with the customers.
   - Benefits of a good design are:

i. Screens are less crowded
ii. Would be less time consuming, 25 percent less time
iii. Screen would be 20 percent more productive
iv. 25 percent fewer errors
v. Improve decision making time
vi. Training cost are lower
vii. The organization customers benefit because of improved services
viii. Condenses interfaces for smaller screens.

2. **Design Principles of User Interaction Design**

place can help designers understand the multifaceted and changing nature of the user experience.

1.6.3 Design Principles

GQ. Discuss about design principles .

- Design principles are used by interaction designers to aid their thinking when designing for the user experience. These are generalizable abstractions intended to orient designers towards thinking about different aspects of their designs. A well-known example is feedback: products should be designed to provide adequate feedback to the users to ensure they know what to do next in their tasks. Another one that has become increasingly important is findability
- . This refers to the degree to which a particular object is easy to discover or locate – be it navigating a website, moving through a building, or finding the delete image option on a digital camera.
- Design principles are derived from a mix of theory-based knowledge, experience, and common sense. They tend to be written in a prescriptive manner, suggesting to designers what to provide and what to avoid at the interface – if you like, the dos and don'ts of interaction design. More specifically, they are intended to help designers explain and improve their designs.
- However, they are not intended to specify how to design an actual interface, e.g. telling the designer how to design a particular icon or how to structure a web portal, but act more like triggers to designers, ensuring that they have provided certain features at an interface.
- A number of design principles have been promoted. The best known are concerned with how to determine what users should see and do when carrying out their tasks using an interactive product. Here we briefly describe the most common ones: visibility, feedback, constraints, consistency, and affordance.

Visibility

- The importance of visibility is exemplified by our contrasting examples at the beginning of the chapter. The voice mail system made the presence and number of waiting messages invisible, while the answer machine made both aspects highly visible. The more visible functions are, the more likely it is that users will be able to know what to do next.
- The controls for different operations are clearly visible, e.g. indicators, headlights, horn, hazard warning lights, indicating what can be done. The relationship between the way the controls have been positioned in the car and what they do makes it easy for the driver to find the appropriate control for the task at hand.

- In contrast, when functions are out of sight, it makes them more difficult to find and know how to use. For example, devices and environments that have become automated through the use of sensor technology (usually for hygiene and energy-saving reasons) – like faucets, elevators, and lights – can sometimes be more difficult for people to know how to control, especially how to activate or deactivate them.
- This can result in people getting caught out and frustrated. Highly visible controlling devices, like knobs, buttons, and switches, which are intuitive to use, have been replaced by invisible and ambiguous activating zones where people have to guess where to move their hands, bodies, or feet on, into, or in front of to make them work.

Feedback

- Related to the concept of visibility is feedback. This is best illustrated by an analogy to what everyday life would be like without it. Imagine trying to play a guitar, slice bread using a knife, or write using a pen if none of the actions produced any effect for several seconds.
- There would be an unbearable delay before the music was produced, the bread was cut, or the words appeared on the paper, making it almost impossible for the person to continue with the next strum, cut, or stroke.
- Feedback involves sending back information about what action has been done and what has been accomplished, allowing the person to continue with the activity. Various kinds of feedback are available for interaction design – audio, tactile, verbal, visual, and combinations of these.
- Deciding which combinations are appropriate for different kinds of activities and interactivities is central. Using feedback in the right way can also provide the necessary visibility for user interaction.

Constraints

- The design concept of constraining refers to determining ways of restricting the kinds of user interaction that can take place at a given moment. There are various ways this can be achieved. A common design practice in graphical user interfaces is to deactivate certain menu options by shading them gray, thereby restricting the user only to actions permissible at that stage of the activity (see Fig. 1.6.2).
- One of the advantages of this form of constraining is that it prevents the user from selecting incorrect options and thereby reduces the chance of making a mistake. The use of different kinds of graphical representations can also constrain a person's interpretation of a problem or information space.
- For example, flow chart diagrams show which objects are related to which, thereby constraining the way the information can be perceived. The physical design of a device can also constrain how it is used; for example, the external slots in a computer have been designed to only allow a cable or card to be inserted in a certain way. Sometimes, however, the physical constraint is ambiguous, as shown in Fig. 1.6.3.

- Where do you plug in the mouse and keyboard?

  Fig. 1.6.3 shows part of the back of a computer. There are two sets of connectors; the two on the right are for a mouse and a keyboard. They look identical and are physically constrained in the same way. How do you know which is which? Do the labels help?

Consistency

- This refers to designing interfaces to have similar operations and use similar elements for achieving similar tasks. In particular, a consistent interface is one that follows rules, such as using the same operation to select all objects.
- For example, a consistent operation is using the same input action to highlight any graphical object at the interface, such as always clicking the left mouse button. Inconsistent interfaces, on the other hand, allow exceptions to a rule. An example is where certain graphical objects (e.g. email messages presented in a table) can be highlighted only by using the right mouse button, while all other operations are highlighted using the left button. A problem with this kind of inconsistency is that it is quite arbitrary, making it difficult for users to remember and making the users more prone to mistakes.
- One of the benefits of consistent interfaces, therefore, is that they are easier to learn and use. Users have to learn only a single mode of operation that is applicable to all objects. This principle works well for simple interfaces with limited operations, such as a portable radio with a small number of operations mapped onto separate buttons. Here, all the user has to do is learn what each button represents and select accordingly.
- However, it can be more problematic to apply the concept of consistency to more complex interfaces, especially when many different operations need to be designed for. For example, consider how to design an interface for an application that offers hundreds of operations, e.g. a word-processing application.
- There is simply not enough space for a thousand buttons, each of which maps onto an individual operation. Even if there were, it would be extremely difficult and time-consuming for the user to search through them all to find the desired operation. A much more effective design solution is to create categories of commands that can be mapped into subsets of operations.

Affordance

- This is a term used to refer to an attribute of an object that allows people to know how to use it. For example, a mouse button invites pushing (in so doing activating clicking) by the way it is physically constrained in its plastic shell.
- At a simple level, to afford means 'to give a clue' (Norman, 1988). When the affordances of a physical object are perceptually obvious, it is easy to know how to interact with it. For example, a door handle affords pulling, a cup handle affords grasping, and a mouse button affords pushing. The term has since been much popularized in interaction design, being used to describe how interfaces should make it obvious as to what can be done at them.

- For example, graphical elements like buttons, icons, links, and scrollbars are talked about with respect to how to make it appear obvious how they should be used: icons should be designed to afford clicking, scrollbars to afford moving up and down, buttons to afford pushing.
- There are two kinds of affordance: perceived and real. Physical objects are said to have real affordances, like grasping, that are perceptually obvious and do not have to be learned. In contrast, user interfaces that are screen-based are virtual and do not have these kinds of real affordances.
- Using this distinction, he argues that it does not make sense to try to design for real affordances at the interface – except when designing physical devices, like control consoles, where affordances like pulling and pressing are helpful in guiding the user to know what to do.
- Alternatively, screen-based interfaces are better conceptualized as perceived affordances, which are essentially learned conventions.
- There are numerous websites and guidebooks that provide more exhaustive sets of design principles that we have just touched upon here, with specific examples for designing for the web, GUIs, and, more generally, interaction design. A well-known resource is Tog's First Principles of Interaction Design (asktog.com).

3. **Explain Inclusiveness and Accessibility in UI Desgin ? How one can achieve both ?**

a. Accessibility

i. It is the extent to which an interactive product is accessible by as many people as possible

ii. Focus is on people with disabilities; for instance, those using android OS provides a range of tools for those with disabilities or apple voiceover

iii. Accessibility can be achieved in two ways: first, through the inclusive design of technology, and second, through the design of assistive technology.

iv. When designing for accessibility, it is essential to understand the types of impairments that can lead to disability as they come in many forms.

v. They are often classified by the type of impairment

vi. For example:

1. Sensory impairment (such as loss of vision or hearing)
2. Physical impairment (having loss of functions to one or more parts of the body, for example,after a stroke or spinal cord injury)

3. Cognitive (for instance, learning impairment or loss of memory/cognitive function due to old age or a condition such as Alzheimer's disease)

b. Inclusiveness

i. Inclusiveness means being fair, open, and equal to everyone.
ii. Making products and services that accommodate the widest possible number of people is called inclusiveness.
iii. For example, smartphones designed for all and made available to everyone regardless of their disability, education, age, or income.

4. **Is User Interaction Design Important? Why?**

- a. User Interface Design is one of the reasons that your website will start to see an influx in traffic. Regardless of what anyone says, it should never be overlooked.
- One thing that takes people by surprise is how big of an impact even the smallest adjustments in UI Design can have.
- In addition to basic design changes like shapes of buttons and color schemes, chances are people won't stay on your site if it's difficult to interact with.
- When people have a good experience on your site, the conversion rates are higher and they tend to recommend more people about it. When people have a bad experience on your site, the chances of your product or company being rejected increases exponentially.
- Simply put, UI Design is important because people make snap judgments and when they visit your site, they'd rather spend 30 seconds opening a new site than meddling around on a difficult one.
- People want to be able to understand things easily , which is why most physical products can be figured out without digging through a 1,000-page manual.
- Simply put, User Interface Design is important because it can make or break your customer base . It creates fewer problems, increases user involvement, perfects functionality and creates a strong link between your customers and your website.

5. **Give the Characteristics of Web User Interface.**

a. Sophisticated Visual Presentation

i. It is what people see on the screen. The sophistication of a graphical system permits displaying lines, including drawings and icons. It also permits the displaying of a variety of character fonts, including different sizes and styles.

b. Pick-and-Click Interaction

i. To identify a proposed action is commonly referred to as pick, the signal to perform an action as click.
ii. Eg: mouse & keyboard

c. Restricted Set of Interface Options

i. The array of alternatives available to the user is what is presented on the screen nothing less, and nothing more.

d. Visualization

i. Visualization is a cognitive process that allows people to understand information that is difficult to perceive, because it is either too voluminous or too abstract.

e. Object Orientation

i. A graphical system consists of objects and actions. Objects are what people see on the screen as a single unit.

f. Properties or Attributes of Objects

i. Properties are the unique characteristics of an object. Properties help to describe an object and can be changed by users.

g. Actions

i. People take actions on objects. They manipulate objects in specific ways (commands) or modify the properties of objects (property or attribute specification).
ii. Example, Several words of text exist. The user then selects an action to apply to that object, such as the action BOLD. The selected words are made bold and will remain bold until selected and changed again.

h. Views

i. Views are ways of looking at an object's information. IBM describes four kinds of views: composed, contents, settings, and help.

i. Use of Recognition Memory
i. Continuous visibility of objects and actions encourages to eliminate “out of sight, out of mind” problem

j. Concurrent Performance of Functions
i. Graphic systems may do two or more things at one time. Multiple programs may run simultaneously.
ii. It may process background tasks (cooperative multitasking) or preemptive multitasking.
iii. Data may also be transferred between programs. It may be temporarily stored on a “clipboard” for later transfer or be automatically swapped between programs.

6. **Who Is Involved in Interaction Design for “A public kiosk providing information about the exhibits available in a science museum”**

   a. To develop the museum kiosk we would primarily require:
   i. Graphic and interaction designers

   ii. Museum curators
   iii. Educational advisers
   iv. Software engineers
   v. Software designers
   vi. Ergonomists

   b. Additionally we would also require:
   i. Visitors:
   1. The main stakeholder for the museum is its customers itself, and what exactly are their requirements is of prime importance
   ii. Clerks:
   1. The people who manage the museum and keep everything in order are also important as they know what is required to deal with large amounts or crowds.
   iii. Administrator
   1. The admin who created the blueprint of the museum and who is responsible for the various sections of the museum who know exactly how the navigate the crowd around the museum.
   iv. Historian

1. The historian will be the best person to know exactly what has to be displayed on the kiosk, and will also be able to classify the artifacts based on user preference.

7. **Explain any five Usablity goals with example.**

GQ: Is the product capable of allowing people to learn, carry out their work efficiently, access the information they need, or buy the goods they want?

- Efficiency refers to the way a product supports users in carrying out their tasks. The marble answering machine described at the beginning of this chapter was considered efficient in that it let the user carry out common tasks, e.g. listening to messages, through a minimal number of steps. In contrast, the voice mail system was considered inefficient because it required the user to carry out many steps and learn an arbitrary set of sequences for the same common task. This implies that an efficient way of supporting common tasks is to let the user use single button or key presses.
- An example of where this kind of efficiency mechanism has been employed effectively is in online shopping. Once users have entered all the necessary personal details in an online form to make a purchase, they can let the website save all their personal details. Then, if they want to make another purchase at that site, they don't have to re-enter all their personal details again. A highly successful mechanism patented by Amazon.com is the one-click option, which requires users only to click a single button when they want to make another purchase.

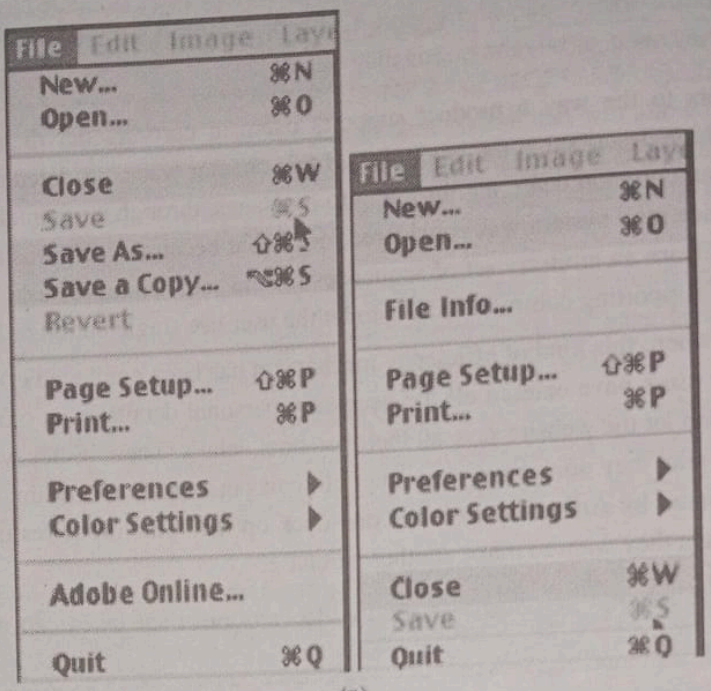
GQ. Once users have learned how to use a product to carry out their tasks, can they sustain a high level of productivity?

- Safety involves protecting the user from dangerous conditions and undesirable situations. In relation to the first ergonomic aspect, it refers to the external conditions where people work. For example, where there are hazardous conditions – such as X-ray machines or toxic chemicals – operators should be able to interact with and control computer-based systems remotely. The second aspect refers to helping any kind of user in any kind of situation avoid the dangers of carrying out unwanted actions accidentally. It also refers to the perceived fears users might have of the consequences of making errors and how this affects their behavior.
- To make interactive products safer in this sense involves

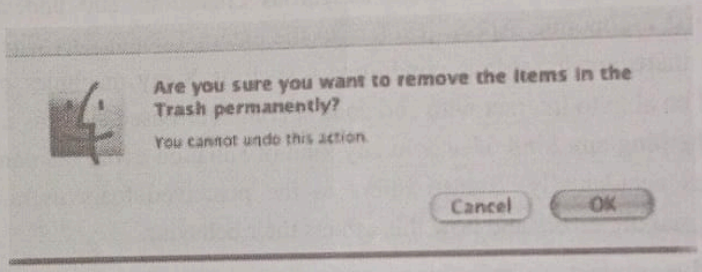
  (i) preventing the user from making serious errors by reducing the risk of wrong keys/buttons being mistakenly activated (an example is not placing the quit or delete-file command right next to the save command on a menu) and

  (ii) providing users with various means of recovery should they make errors. Safe interactive systems should engender confidence and allow the user the opportunity to explore the interface to try out new operations (see Fig. 1.6.1(a)). Other safety mechanisms include undo facilities and confirmatory dialog boxes that give users another chance to consider their intentions (a) well-known example is the appearance of a dialog box, after issuing the command to delete everything in the trashcan, saying:

  'Are you sure you want to remove all the items in the Trash permanently?'– see Fig. 1.6.1(b)).



(a)



(b)

(a) A safe and unsafe menu. Which is which and why? (b) A warning dialog box for Mac OS X

Fig. 1.6.1

GQ. What is the range of errors that are possible using the product and what measures are there to permit users to recover easily from them?

- Utility refers to the extent to which the product provides the right kind of functionality so that users can do what they need or want to do.
- An example of a product with high utility is an accounting software package that provides a powerful computational tool that accountants can use to work out tax returns.
- An example of a product with low utility is a software drawing tool that does not allow users to draw freehand but forces them to use a mouse to create their drawings, using only polygon shapes.

GQ. Does the product provide an appropriate set of functions that will enable users to carry out all their tasks in the way they want to do them?

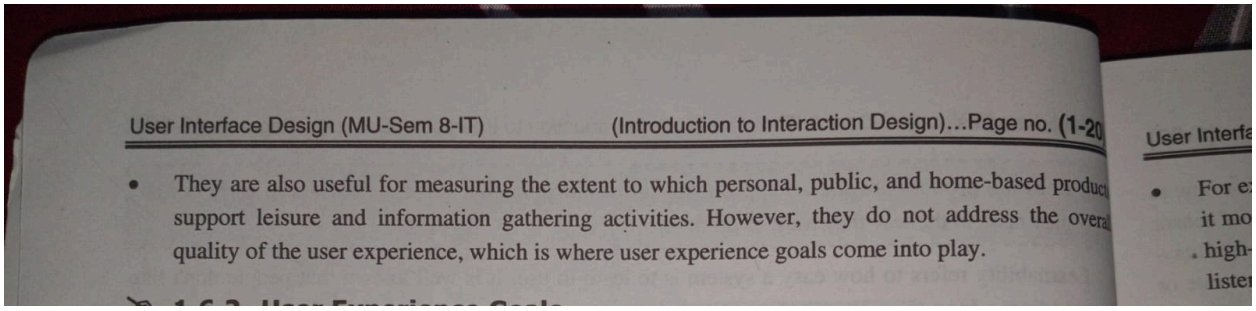
- Learnability refers to how easy a system is to learn to use. It is well known that people don't like spending a long time learning how to use a system. They want to get started straight away and become competent at carrying out tasks without too much effort. This is especially so for interactive products intended for everyday use (e.g. social media, email, GPS) and those used only infrequently (e.g. online tax forms). To a certain extent, people are prepared to spend longer learning more complex systems that provide a wider range of functionality, like web authoring tools.
- In these situations, pop-up tutorials can help by providing contextualized step-by-step material with hands-on exercises. A key concern is determining how much time users are prepared to spend learning a product. It seems a waste if a product provides a range of functionality that the majority of users are unable or not prepared to spend time learning how to use.

GQ. Is it possible for the user to work out how to use the product by exploring the interface and trying out certain actions? How hard will it be to learn the whole set of functions in this way?

- Memorability refers to how easy a product is to remember how to use, once learned. This is especially important for interactive products that are used infrequently. If users haven't used an operation for a few months or longer, they should be able to remember or at least rapidly be reminded how to use it. Users shouldn't have to keep relearning how to carry out tasks. Unfortunately, this tends to happen when the operations required to be learned are obscure, illogical, or poorly sequenced.
- Users need to be helped to remember how to do tasks. There are many ways of designing the interaction to support this. For example, users can be helped to remember the sequence of operations at different stages of a task through meaningful icons, command names, and menu options. Also, structuring options and icons so they are placed in relevant categories of options, e.g. placing all the drawing tools in the same place on the screen, can help the user remember where to look to find a particular tool at a given stage of a task.

GQ. What kinds of interface support have been provided to help users remember how to carry out tasks, especially for products and operations they use infrequently?

- As well as couching usability goals in terms of specific questions, they are turned into usability criteria. These are specific objectives that enable the usability of a product to be assessed in terms of how it can improve (or not) a user's performance.
- Examples of commonly used usability criteria are time to complete a task (efficiency), time to learn a task (learnability), and the number of errors made when carrying out a given task over time (memorability). These can provide quantitative indicators of the extent to which productivity has increased, or how work, training, or learning have been improved.



8. **Explain any five design principles to aid thinking when designing for product developement with good UX.**

Same as 2nd

9. **List atleast 4 good user experience based features of web/moible app**

1. Add Shortcuts for Advanced Users
Shortcuts are a great way of enabling users to accomplish repetitive work routines more quickly. People who regularly use interactive apps might intuitively try out shortcuts they expect to work, however you should still offer a legend or shortcut list to support learnability. The fact that shortcuts are usually an optional addition is what makes them so great, as new users aren't forced to learn them right away.

Keep in mind, that web apps run in browsers which also provide shortcuts. So you sometimes need to block default behaviors to make your own shortcuts function correctly. Besides that, make sure you stick to commonly used key combinations - best case scenario would be that users don't even have to learn your shortcuts as they're already familiar with them from other environments. Some basic example shortcuts your web app should support:

CTRL+C: copy
Ctrl+V: paste
DEL: delete

2. Forgive Mistakes with Undo-Redo
Making mistakes that aren't easily fixed leads to frustration, which is why fault tolerance is an important UX feature. Mistakes will inevitably happen since users have different background knowledge and generally don't know the app as good as the developers do. That's where undo-redo functionality can come in very handy, enabling the user to 'go back in time' to iron out wrong steps.

By now, undo-redo is probably even an expected feature, since you can find it in so many apps. For example, you can undo your typings in any input field of your browser by default. Offering such a mechanism makes the user more confident in his working steps and promotes creativity, due to the reduced fear of undoable mistakes.

We made SceneLab's undo-redo implementation open-source, so if you happen to develop your web app with Angular, check out ngrx-wieder.

Undo redo interaction in an online design editor

3. Make the UI Adjustable to Specific Use Cases
User interfaces don't have to be static, therefore consider providing variations that are supportive of different tasks. Not every user pursues the same objective or follows the same work routine, so making the UI dynamic let's the user control where the main focus should be.

Too much information can be distracting. Hiding parts of the interface makes users feel more comfortable and organized overall. Accordions, dropdowns or scrollable containers are common examples for blending out excessive information.

In our editor, users can hide the sidebars and fit the canvas to their screen to maximize the focus on the design area.

Adjusting the dynamic user interface of a webapp

4. Provide Visual Feedback for Loading, Success & Error States
You don't want users to be wondering what's going on 'under the hood' of your app. If possible, every interaction by a user should lead to some kind of (usually visual) feedback to reduce confusion. For example, loading indicators prevent multiple clicks which might lead to multiple server requests, eventually slowing down the workflow even more.

Most code out there isn't 100% perfect and admittedly, it's hard to predict every edge case that could break an app. However, good error-handling can cover most issues the user might run into. By exposing meaningful error messages to the user, they can react to them directly and possibly avoid future issues. This could save you several support requests or even lost customers.

Visual feedback is a rare case in which animations are actually helpful instead of distracting. To be relevant in the context, any message to the user should be shown as an immediate response to his actions. Movement or fade-ins will be noticed more easily and can also indicate that a process is still running (think of a spinning loading symbol).

By the way, you can use tools like Sentry to monitor frontend errors and track down potential bugs. We've even written a Sentry integration guide for Angular.

5. Implement Auto-Saving Mechanisms
For the most part, web applications rely on an active internet connection to function correctly. Accordingly, the app state feels more temporary than for desktop software, due to a higher risk of data loss. To counteract this, auto-saving is a valid strategy. It takes away responsibility from the users, allowing them not to constantly worry about saving their progress and instead focus on the actual work.

Embracing the ideas of the previous chapter, showing a 'saved' feedback can reinforce the user's trust in the saving mechanism to actually work.

If you want your app to be as independent of disconnections as possible, consider additionally caching information in the local storage of the browser or using a service worker.

Recommended: Implementing Auto-Saving with Angular

6. Incorporate Drag & Drop to Reduce Clicks
Drag & drop can be leveraged to add semantic meaning to an interaction. For example, you can directly assign a position when dropping elements or share objects between distinct contexts.

Here's a more specific case: Sometimes a user might have a file for upload already sorted out in his file explorer, but opening a file input dialog requires him to navigate to it again. A much easier workflow would be to simply drag the file into the browser window. Making the drop area as large a possible will make the whole process more convenient.

In SceneLab you can drag items from within the app to quickly position them, or drop your own images on predefined areas to quickly apply them to mockups with just one interaction.

Using drag & drop to place a smart phone and apply an image to the mockup
7. Use Cursors to Differentiate Interaction Types

It's a good practice to indicate which parts of the UI are interactive. This can be accomplished by showing anything other than the default cursor. In addition to that, different cursors are perfect to distinguish different types of interactions, here are some examples:

Pointer for toggles, links and click interactions in general
Move / Drag for global movement (like on google maps)
Text for editable text or input fields (the browser does this automatically)
Not-allowed to indicate that something is preventing the interaction (e.g. user permission)
On top of this, hover effects are suitable for emphasizing interactive areas even more.

Different cursor showing how to interact with specific elements

8. Disable Text-Selecting for User Interfaces
By default, text and image elements can be selected or highlighted by the user. Quite often, this happens by accident and is simply distracting, especially in apps that incorporate other drag interactions. This behavior can be turned off via the CSS-rule user-select: none;.

However, you should rather apply this to hand-picked elements instead the whole site. After all, highlighting text is still an expected feature for input fields or long text contents where the user might want to copy something.

9. Add a Custom Context Menu
In contrast to plain websites that only display content, web applications often include a lot more logic and functionality. By providing a custom context menu, you can hide functionality that doesn't have to be visible all the time. Additionally, it's a great opportunity to show different menus based on the clicked object (the context), which can make certain workflows much more convenient.

One thing to consider is that a context menu usually opens right next to where the right click happened. That means you have to prevent the menu from reaching out of the viewport when the user clicks close to the edge.

Custom context menu for a webapp
Custom context menu for a webapp

10. Embed a Support Chat

Even when you invest a lot of time into improving UX, it's unlikely that every user out there will be perfectly satisfied with your app. Offering a direct and hassle-free contact opportunity greatly increases your chances of getting valuable feedback from unhappy users. Think about it that way: If you had a bad experience using a website or app, would you want to spend even more time filling out a contact form, just to get a reply a few days later? The answer would probably be no, unless you really rely on the site to work and no alternatives exist.

Users might even have a better experience just by knowing that they could get quick support in case of any problems, making them more confident while using the app.

Embedded quick chat icon to get user feedback

10. **Write short note on usability goals and user experience goals.**

Same as 7th ans and

... where user experience goals come into play.

1.6.2 User Experience Goals

GQ. Explain desirable and undesirable aspects of the user experience

- A diversity of user experience goals has been articulated in interaction design, which cover a range of emotions and felt experiences.
- These include desirable and undesirable ones, as shown in Table 1.6.1.

Table 1.6.1 : Desirable and undesirable aspects of the user experience

Desirable aspects		
Satisfying	Helpful	Fun
Enjoyable	Motivating	Provocative
Engaging	Challenging	Surprising
Pleasurable	Enhancing sociability	Rewarding
Exciting	Supporting creativity	Emotionality fulfilling
Entertaining	Cognitively stimulating	
Undesirable aspects		
Boring	Unpleasant	
Frustrating	Patronizing	
Making one feel guilty	Making one feel stupid	
Annoying	Cutesy	
Childish	Gimmicky	

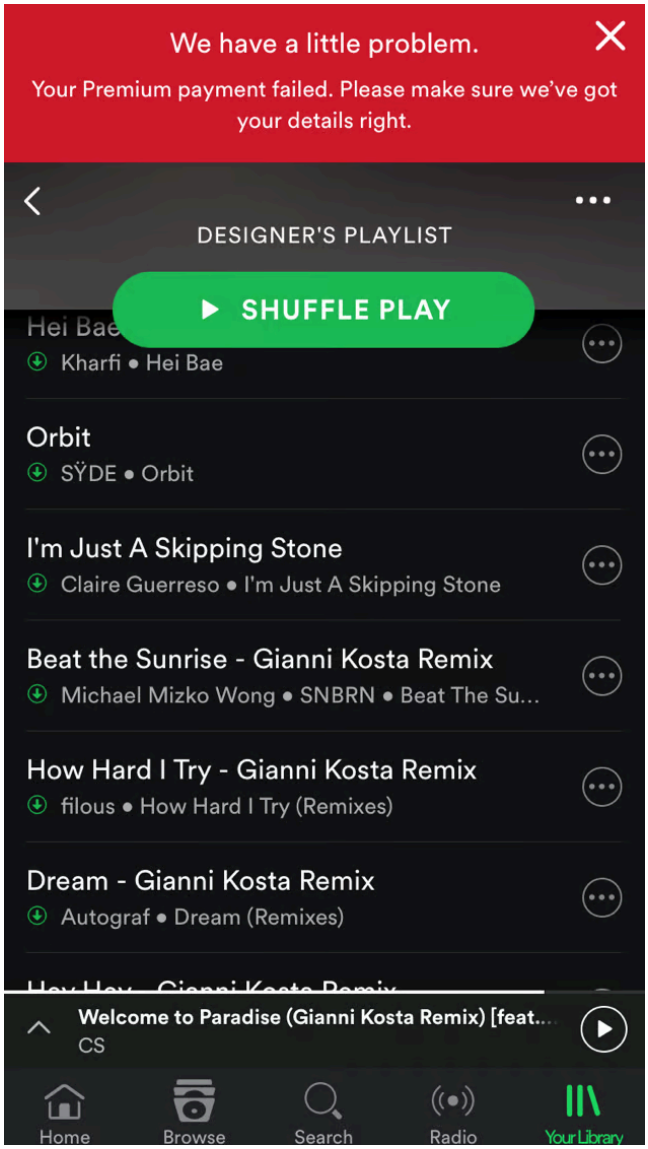
- Many of these are subjective qualities and are concerned with how a system feels to a user. They differ from the more objective usability goals in that they are concerned with how users experience an interactive product from their perspective, rather than assessing how useful or productive a system is from its own perspective. Whereas the terms used to describe usability goals comprise a small distinct set, many more terms are used to describe the multifaceted nature of the user experience.
- They also overlap with what they are referring to. In so doing, they offer subtly different options for expressing the way an experience varies for the same activity over time, technology, and place.

- For example, we may describe listening to music in the shower as highly pleasurable, but consider it more apt to describe listening to music in the car as enjoyable. Similarly, listening to music on a high-end powerful music system may invoke exciting and emotionally fulfilling feelings, while listening to it on an iPod Shuffle may be serendipitously enjoyable, especially not knowing what tune is next. The process of selecting terms that best convey a user's feelings, state of being, emotions, sensations, and so forth when using or interacting with a product at a given time and place can help designers understand the multifaceted and changing nature of the user experience.

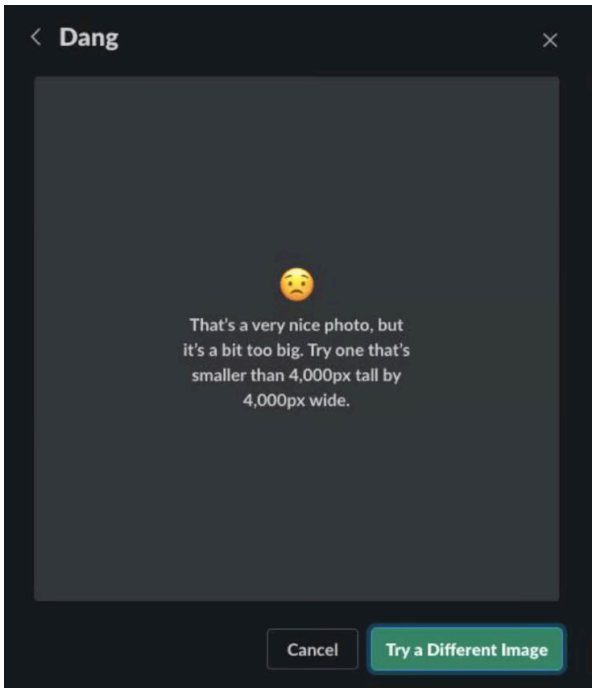
11. Explain Good Error Messages with examples.

1. Spotify's failed payment error message

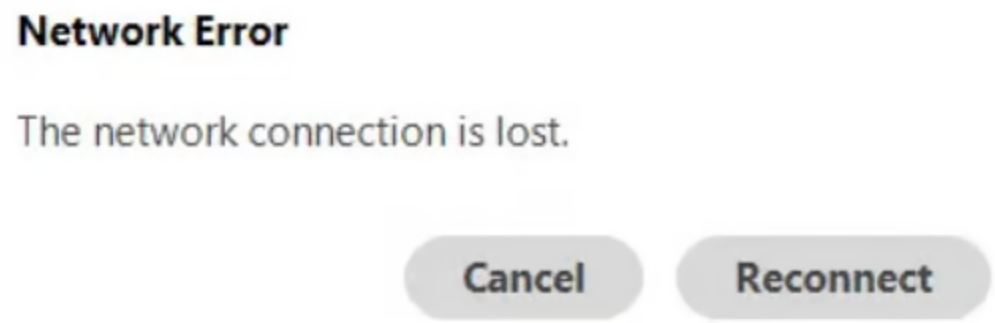
Spotify Error Message



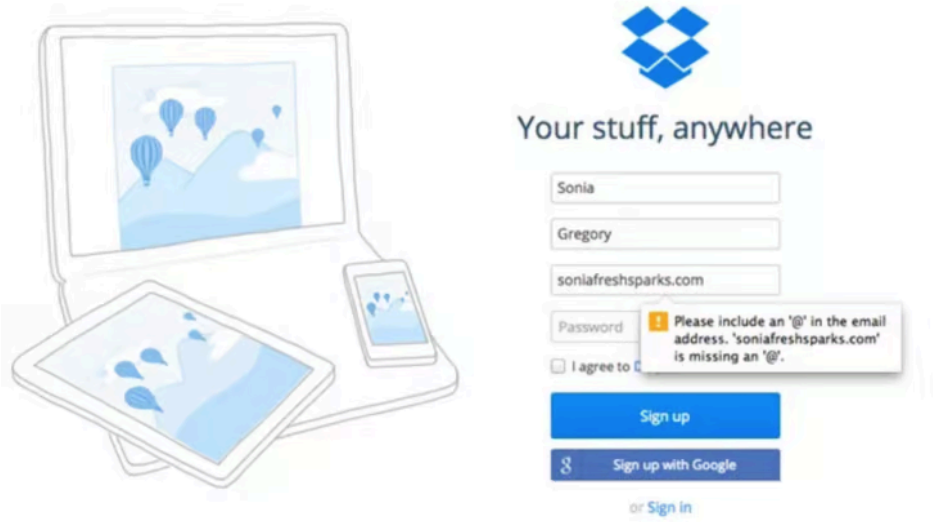
Why is Spotifiy's error message so effective?
It's immediately eye-catching due to the red banner
The language is empathetic rather than alarmist: "We have a little problem"
It explains the issue concisely without going into unnecessary detail: "Your payment has failed"
It provides clear instructions on what to do to fix the problem: "Please make sure we've got your details right"
It doesn't put the blame on the user: "… make sure we've got your details right"
2. Slack's "photo is too big" error message
Slack Error Message



Why is Slack's error message so effective?
The emoji is eye-catching, on-brand, empathetic, and communicates that something is wrong
The language is relatable and relevant to the specific issue: "That's a very nice photo"
The issue is clearly and succinctly described: "It's a bit too big"
Specific and clear instructions are provided alongside a button that complements the required action: "Try a Different Image"
3. Network error message
Network Error Message

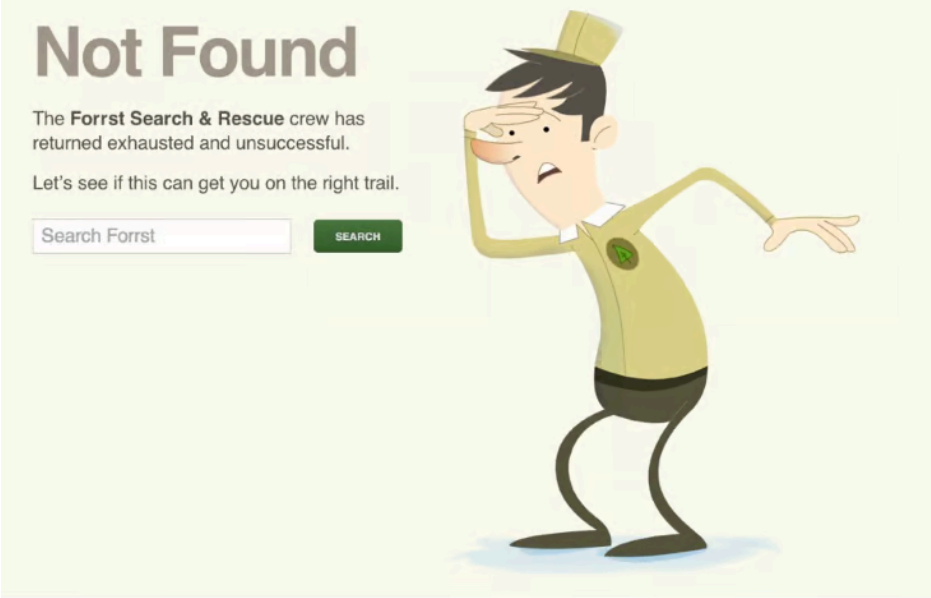


Why is this network error message so effective?
Although this is a technical issue, the user isn't confronted with technical or confusing language.
The description of the error is simple and clear: "The network connection is lost"
The user's next steps for solving the problem or canceling the attempt are clear from the text on the buttons
4. Dropbox's login error message
Dropbox Error Message



Why is Dropbox's error message so effective?
The use of a color that is not part of the brand's color palette — in this case orange — means the error message immediately stands out to the user

It gives precise instructions on how to fix the problem
It provides an example of a correct format to use so the user knows what their email address should look like
5. Forrst Search & Rescue's Not Found page
Forrst Error Message



Why is Forrst Search and Rescue's Not Found page so effective?
The on-page copy uses light-touch humor that is both positive and on-brand, providing some light relief in a frustrating moment
It offers users a clear solution that will get them back to where they need to be
Common mistakes to avoid when writing error messages
We've all seen them: those confusing, long, and jargon-heavy error messages that cause more trouble than they solve. Let's take a look at all th

12. What care will you take while designing an interface for blind person.

When designing interfaces for blind individuals, designers must prioritize accessibility by incorporating features such as screen readers, keyboard navigation, and descriptive alternative text for images. Additionally, ensuring a logical and intuitive layout and providing clear and concise textual descriptions are essential to facilitate effective interaction and navigation for blind users. While color blindness can be addressed through color selection, addressing peripheral vision loss or complete blindness typically requires the integration of assistive technologies tailored to the specific needs of blind individuals, such as screen readers or braille displays. Therefore, an inclusive design approach encompasses both consideration of color blindness and the implementation of assistive technologies to accommodate varying levels of visual impairment.

13. Explain principles of User-Centered Approach?

a. Early focus on users and tasks.

i. This means first understanding who the users will be by directly studying their cognitive, behavioral, anthropomorphic, and attitudinal characteristics.

ii. This requires observing users doing their normal tasks, studying the nature of those tasks, and then involving users in the design process.

b. Empirical measurement.

i. Early in development, the reactions and performance of intended users to printed scenarios, manuals, and so forth, are observed and measured.

ii. Later, users interact with simulations and prototypes, and their performance and reactions are observed, recorded, and analyzed.

c. Iterative design.

i. When problems are found in user testing, they are fixed, and then more tests and observations are carried out to see the effects of the fixes. This means that design and development are it

14. Importance of Involving Users in Design Process

a. Involving users in development is important because it's the best way to ensure that the end product is usable and that it indeed will be used.

b. In commercial projects, a role called the product owner is common. The product owner's job is to filter user and customer input to the development cycle and to prioritize requirements or features. This person is usually someone with business and technical knowledge, but not interaction design knowledge, and they are rarely (if ever) a direct user of the product.

c. The best way to ensure that developers gain a good understanding of users' goals, lead- ing to a more appropriate, more usable product, is to involve target users throughout development. However, two other aspects unrelated to functionality are equally as important if the product is to be usable and used: expectation management and ownership.

Expectation management

•Realistic expectations

•No surprises, no disappointments

•Timely training

•Communication, but no hype

–

Ownership

•Make the users active stakeholders

•More likely to forgive or accept problems

•Can make a big difference in acceptance and success of product

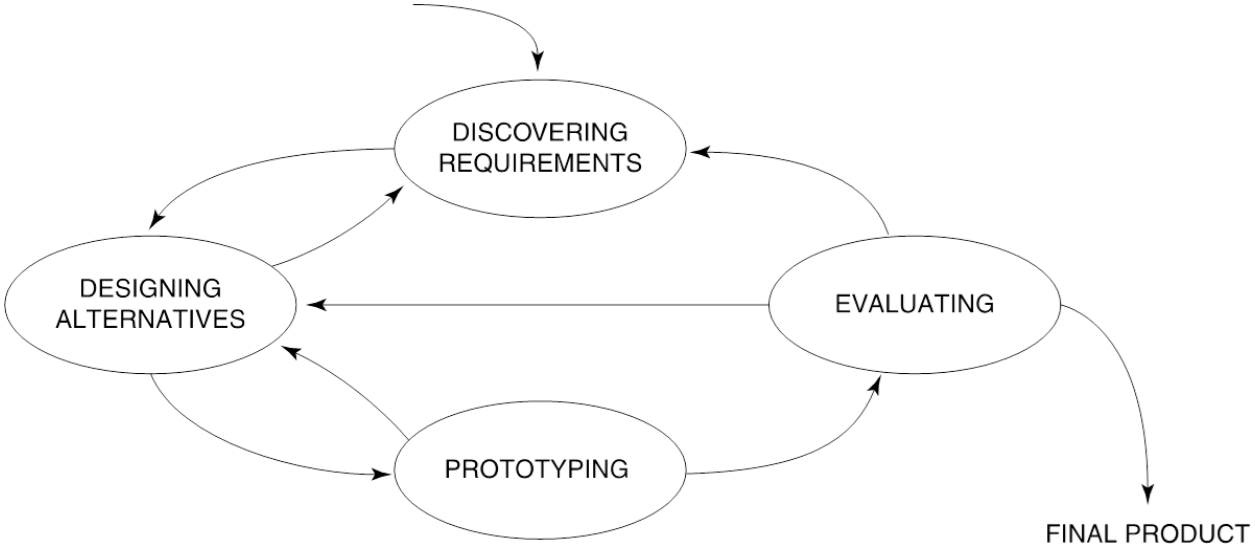
15. Elaborate the four basic activities of Interaction Design.

1.Discovering requirements

2.Designing alternatives

3.Prototyping alternative designs

4.Evaluating product and its user experience throughout



Discovering Requirements

This activity covers the left side of the double diamond of design, and it is focused on discovering something new about the world and defining what will be developed. In the case of interaction design, this includes understanding the target users and the support an interactive product could usefully provide. This understanding is gleaned through data gathering and analysis, which are discussed in Chapters 8–10. It forms the basis of the product's requirements and underpins subsequent design and development. The requirements activity is discussed further in Chapter 11.

Designing Alternatives

This is the core activity of designing and is part of the Develop phase of the double diamond: proposing ideas for meeting the requirements. For interaction design, this activity can be viewed as two subactivities: conceptual design and concrete design. Conceptual design involves producing the conceptual model for the product, and a conceptual model describes an abstraction outlining what people can do with a product and what concepts are needed to understand how to interact with it. Concrete design considers the detail of the product including the colors, sounds, and images to use, menu design, and icon design. Alternatives are considered at every point. Conceptual design is discussed in Chapter 3, and more design issues for specific interface types are in Chapter 7; more detail about how to design an interactive product is in Chapter 12.

Prototyping

Prototyping is also part of the Develop phase of the double diamond. Interaction design involves designing the behavior of interactive products as well as their look and feel. The most effective way for users to evaluate such designs is to interact with them, and this can be achieved through prototyping. This does not necessarily mean that a piece of software is required. There are different prototyping techniques, not all of which require a working piece of software. For example, paper-based prototypes are quick and cheap to build and are effective for identifying problems in the early stages of design, and through role-playing users can get a real sense of what it will be like to interact with the product. Prototyping is covered in Chapter 12.

Evaluating

Evaluating is also part of the Develop phase of the double diamond. It is the process of determining the usability and acceptability of the product or design measured in terms of a variety of usability and user-experience criteria. Evaluation does not replace activities concerned with quality assurance and testing to make sure that the final product is fit for its intended purpose, but it complements and enhances them. Chapters 14–16 cover evaluation.

The activities to discover requirements, design alternatives, build prototypes, and evaluate them are intertwined: alternatives are evaluated through the prototypes, and the results are fed back into further design or to identify alternative requirements.

16. **Explain steps invloved in “Google Design Sprint”**

Google Design Sprints (Adapted from Knapp et al. (2016))

Google Ventures has developed a structured approach to design that supports rapid ideation and testing of potential solutions to a design challenge. This is called the *Google Design Sprint*. A sprint is divided into five phases, and each phase is completed in a day. This means that in five days, you can go from a design challenge to a solution that has been tested with customers. As the authors say, "You won't finish with a complete, detailed, ready-to-ship product. But you will make rapid progress, and know for sure if you're headed in the right direction" (Knapp et al., 2016, p16–17). Teams are encouraged to iterate on the last two phases and to develop and re-test prototypes. If necessary, the first idea can be thrown away and the process started again at Phase 1. There is preparation to be done before the sprint begins. This preparation and the five phases are described next (see Figure 2.6).

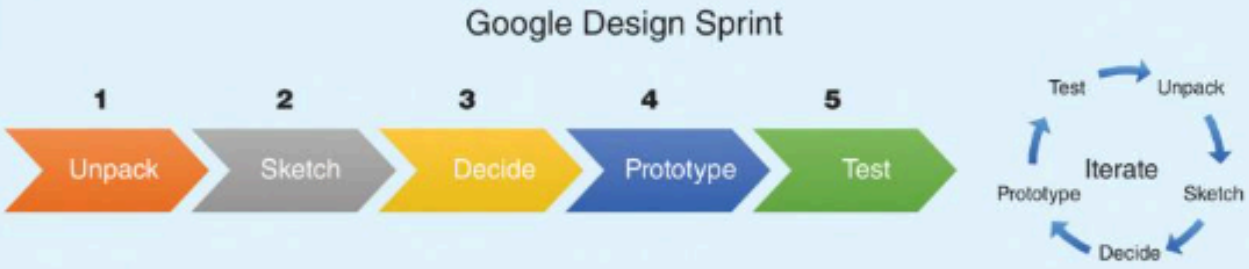


Figure 2.6 The five phases of the Google Design Sprint

Source: www.agilemarketing.net/google-design-sprints. Used courtesy of Agile Marketing

Setting the Stage
This time is used to choose the right design challenge, gather the right team, and organize the time and space to run the sprint (that is, full-time for everyone for five days). The sprint can help in high-stake challenges, when you're running out of time, or if you're just stuck. The team composition depends on the product, but it has about seven people including a decider (who chooses the design to show to the customer), customer expert, technical expert, and anyone who will bring a disruptive perspective.

Unpack
Day 1 focuses on making a map of the challenge and choosing a target, that is, a part of the challenge that can be achieved in a week.

Sketch Competing Solutions
Day 2 focuses on generating solutions, with an emphasis on sketching and individual creativity rather than group brainstorming.

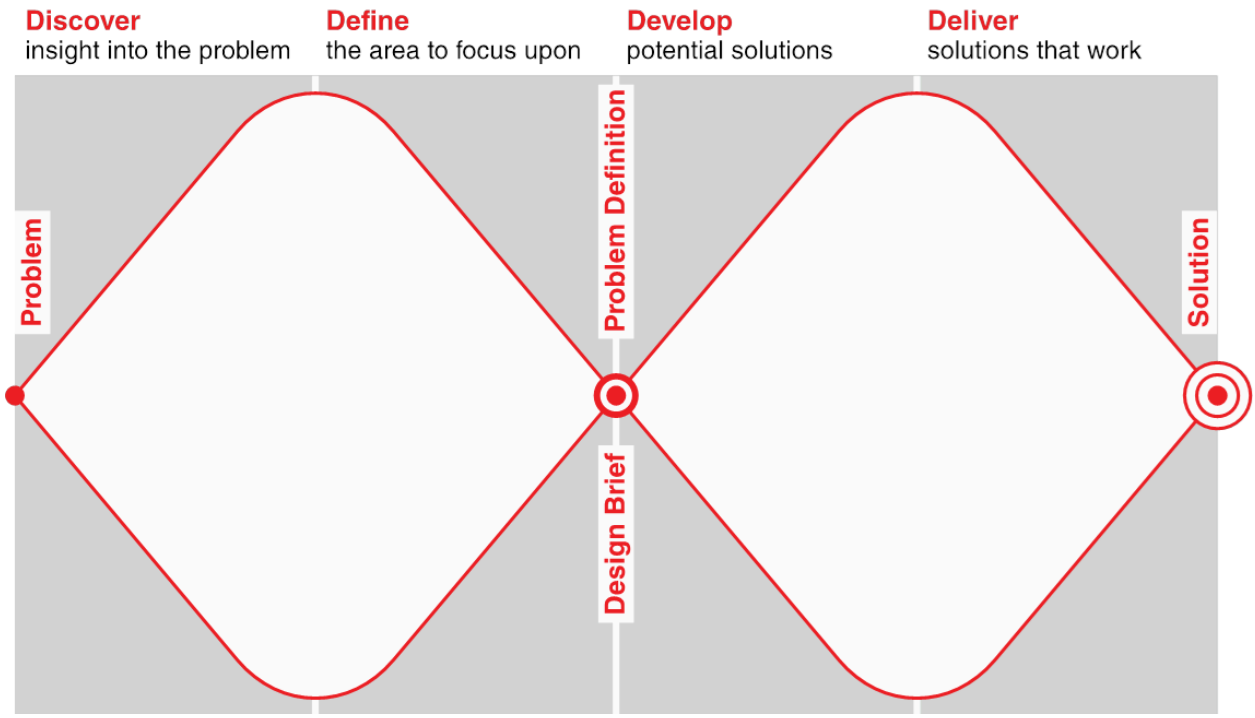
Decide on the Best
Day 3 focuses on critiquing the solutions generated on Day 1, choosing the one most likely to meet the sprint's challenge, and producing a storyboard. Whichever solution is chosen, the decider needs to support the design.

Build a Realistic Prototype
Day 4 focuses on turning the storyboard into a realistic prototype, that is, something on which customers can provide feedback. We discuss prototyping further in Chapter 12.

Test with Target Customers
Day 5 focuses on getting feedback from five customers and learning from their reactions.

The Google Design Sprint is a process for answering critical business questions through design, prototyping, and testing ideas with customers. Marta Rey-Babarro, who works at Google as a staff UX researcher and was the cofounder of Google's internal Sprint Academy, describes how they used a sprint to improve the experience of traveling for business.

17. Explain double diamond of design process in detail.



The Double Diamond is a design process model that was developed by the British Design Council. It is a visual representation of the divergent and convergent stages in the design process, illustrating how designers explore a problem space and then narrow down to a specific solution. The model is called "Double Diamond" because it consists of two diamond shapes, representing two main phases of the design process: divergent thinking and convergent thinking, each with its own diamond.

1. **Discover:** The first diamond represents the starting point of the design process, where the goal is to understand the problem space. This phase involves research, observation, and empathy to gain insights into the needs and challenges of the users. The key activities in this phase include:

- **Research:** Conducting user research, market analysis, and competitive analysis to gather relevant information.
- **Observation:** Observing users in their environment to understand their behaviors, preferences, and pain points.
- **Empathy:** Putting oneself in the shoes of the user to gain a deeper understanding of their needs and emotions.

2. **Define:** After gathering insights in the Discover phase, the next step is to define the problem statement. This phase involves synthesizing the

information gathered in the Discover phase to identify the key challenges and opportunities. The key activities in this phase include:

- **Synthesis:** Analyzing the research findings to identify patterns, themes, and insights.
- **Problem Definition:** Clearly defining the problem statement based on the insights gathered.
- **Stakeholder Alignment:** Ensuring alignment among stakeholders on the problem statement and project goals

.

3. **Develop:** The second diamond represents the development phase, where designers explore a wide range of solutions to address the defined problem. This phase involves brainstorming, ideation, and prototyping to generate and test ideas. The key activities in this phase include:

- **Ideation:** Generating a large number of ideas through brainstorming sessions and design workshops.
- **Prototyping:** Creating low-fidelity prototypes to quickly test and iterate on ideas.
- **User Testing:** Gathering feedback from users through prototype testing to validate ideas and make improvements.

4. **Deliver:** Once a promising solution has been identified and tested, the final step is to deliver the solution to the users. This phase involves refining the design, developing a detailed plan for implementation, and launching the product or service. The key activities in this phase include:

- **Refinement:** Iterating on the design based on user feedback and usability testing.
- **Implementation:** Developing a detailed plan for how the solution will be implemented and rolled out.

Launch: Introducing the solution to the market and monitoring its performance.
Overall, the Double Diamond model emphasizes the importance of both divergent and convergent thinking in the design process, highlighting the need to explore a wide range of possibilities before narrowing down to a specific solution. It is a flexible and iterative approach that encourages designers to continuously refine their ideas based on feedback and insights gathered throughout the process.

18. **Explain importance and drawback of involving users in design process?**

Same as 14 th and
For drawbacks

1. Biased Feedback: Users may provide feedback based on their personal preferences, which might not necessarily represent the broader user base. This can lead to design decisions that cater only to a specific subset of users, potentially neglecting the needs of others.

2. Limited Perspective: Users might not have a comprehensive understanding of the technical or business constraints that designers and developers face. This can result in unrealistic expectations or suggestions that are not feasible to implement within the given constraints.

3. Overreliance on User Feedback: Constantly seeking user input can lead to decision paralysis or a never-ending cycle of revisions. Designers may struggle to balance user feedback with their own expertise and vision, potentially hindering the progress of the project.

4. Lack of Innovation: Relying solely on user feedback may limit the creativity and innovation of the design team. Users may be unable to envision solutions that are radically different from what they are already familiar with, resulting in incremental improvements rather than breakthrough innovations.

5. Time and Resource Constraints: Involving users in the design process requires time and resources to conduct interviews, surveys, usability tests, etc. These activities can prolong the design phase and increase project costs, especially if there is a need for extensive iterations based on user feedback.

6. Difficulty in Managing Feedback: Gathering feedback from a large number of users can be challenging to manage and prioritize. Design teams may struggle to sift through contradictory feedback or distinguish between valuable insights and personal preferences.

sp

19. **Short note on Lifecycle model for interaction design.**

Same as 15th

20. What Is Involved in Interaction Design?

Interaction design has specific activities focused on discovering requirements for the product, designing something to fulfill those requirements, and producing prototypes that are then evaluated. In addition, interaction design focuses attention on users and their goals.

For example, the artifact's use and target domain are investigated by taking a user-centered approach to development, users' opinions and reactions to early designs are sought, and users are involved appropriately in the development process itself. This means that users' concerns direct the development rather than just technical concerns.

Design is also about trade-offs—about balancing conflicting requirements. One common form of trade-off when developing a system to offer advice, for example, is deciding how much choice will be given to the user and how much direction the system should offer. Often, the division will depend on the purpose of the system, for example, whether it is for playing music tracks or for controlling traffic flow. Getting the balance right requires experience, but it also requires the development and evaluation of alternative solutions.

Generating alternatives is a key principle in most design disciplines and one that is also central to interaction design. Linus Pauling, twice a Nobel Prize winner, once said, "The best way to get a good idea is to get lots of ideas." Generating lots of ideas is not necessarily hard, but choosing which of them to pursue is more difficult. For example, Tom Kelley (2016) describes seven secrets for successful brainstorms, including sharpening the focus (having a well-honed problem statement), having playful rules (to encourage ideas), and getting physical (using visual props).

Involving users and others in the design process means that the designs and potential solutions will need to be communicated to people other than the original designer. This requires the design to be captured and expressed in a form that allows review, revision, and improvement. There are many ways of doing this, one of the simplest being to produce a series of sketches. Other common approaches are to write a description in natural language, to draw a series of diagrams, and to build a prototype, that is, a limited version of the final product. A combination of these techniques is likely to be the most effective. When users are involved, capturing and expressing a design in a suitable format is especially important since they are unlikely to understand jargon or specialist notations. In fact, a form with which users can interact is most effective, so building prototypes is an extremely powerful approach.

Module 2

21. Explain interaction types in detail

2.4 INTERACTION TYPES

GQ. Elaborate Interaction types.

- Another way of conceptualizing the design space is in terms of the interaction types that will underlie the user experience. Essentially, these are the ways a person interacts with a product or application. We propose that there are four main types: instructing, conversing, manipulating, and exploring.
- Deciding upon which of these to use, and why, can help designers formulate a conceptual model before committing to a particular interface in which to implement them, e.g. speech-based, gesture-based, touch-based, menu based, and so on. Note that we are distinguishing here between interaction types (which we discuss in this section) and interface types. While cost and other product constraints will often dictate which interface style can be used for a given application, considering the interaction type that will best support a user experience can highlight the potential trade-offs, dilemmas, and pros and cons.
- Consider the following problem description: a company has been asked to design a computer-based system that will encourage autistic children to communicate and express themselves better. What type of interaction would be appropriate to use at the interface for this particular user group? It is known that autistic children find it difficult to express what they are feeling or thinking through talking and are more expressive when using their bodies and limbs.
- Clearly an interaction style based on talking would not be effective, but one that involves the children interacting with a system by moving in a physical and/or digital space would seem a more promising starting point.
- Below we describe in more detail each of the four types of interaction. It should be noted that they are not meant to be mutually exclusive (e.g. someone can interact with a system based on different kinds of activities); nor are they meant to be definitive.

1. **Instructing :** where users issue instructions to a system. This can be done in a number of ways, including: typing in commands, selecting options from menus in a windows environment or on a multitouch screen, speaking aloud commands, gesturing, pressing buttons, or using a combination of function keys.
2. **Conversing** : where users have a dialog with a system. Users can speak via an interface or type in questions to which the system replies via text or speech output.
3. **Manipulating** : where users interact with objects in a virtual or physical space by manipulating them (e.g. opening, holding, closing, placing). Users can hone their familiar knowledge of how to interact with objects.
4. **Exploring** : where users move through a virtual environment or a physical space. Virtual environments include 3D worlds, and augmented and virtual reality systems. They enable users to hone their familiar knowledge of physically moving around. Physical spaces that use sensor-based technologies include smart rooms and ambient environments, also enabling people to capitalize on familiarity.

1. **Instructing**

- Issuing commands and selecting options
- Where users instruct a system and tell it what to do
- For example: Tell the time, print a file, or save a file
- Very common conceptual model underlying a diversity of devices and systems
- For instance: Word processors, VCRs, and vending machines
- The main benefit is that instructing supports quick and efficient interaction
- Good for repetitive kinds of actions performed on multiple objects

2. **Conversing**

- Interacting with a system as if having a conversation
- Underlying model of having a conversation with another human
- Ranges from simple voice recognition menu-driven systems to more complex 'natural language' dialogs
- Examples include timetables, search engines, advice-giving systems, and help systems
- Also virtual agents, chatbots, toys, and pet robots designed to converse with you

3. **Manipulating**

- Interacting with objects in a virtual or physical space by manipulating them
- Involves dragging, selecting, opening, closing and zooming actions on virtual objects
- Exploit's users' knowledge of how they move and manipulate in the physical world
- Can involve actions using physical controllers (for example, Nintendo Wii) or air gestures (such as, Microsoft Kinect) to control the movements of an on-screen avatar
- Tagged physical objects (for instance, balls) that are manipulated in a physical world result in physical/digital events (such as animation)

4. **Exploring**

- Moving through a virtual environment or a physical space
- Involves moving through virtual or physical environments
- Users can explore aspects of a virtual 3D environment
- Physical environments can also be embedded with sensors that when detect the presence of someone will trigger digital or physical events to happen
- Many examples of virtual environments, including cities, parks, buildings, rooms, and datasets
- Enable users to fly over them and zoom in and out of different parts

5. **Responding**

- The system initiates the interaction and the user chooses whether to respond
- System takes the initiative to alert user to something that it "thinks" is of interest
- System does this by:
- Detecting the location and-or presence of someone in a vicinity and notifies them on their phone or watch,
- What it has learned from their repeated behaviors

- Examples:
- Alerts the user of a nearby coffee bar where some friends are meeting
- User's fitness tracker notifies them of a milestone reached
- Automatic system response without any requests made by the user

22. **What is Interface Metaphors? Give 2 examples of popular methaphor used in app developement in the last few years**
    - interface designed to be similar to a physical entity but also has own properties
    - For example, desktop metaphor, and web portals
    - Can be based on activity, object, or a combination of both
    - Exploit user's familiar knowledge, helping them to understand 'the unfamiliar'
    - Conjures up the essence of the unfamiliar activity, enabling users to leverage this to understand more aspects of the unfamiliar functionality
    - Conceptualizing what users are doing
    - For instance, surfing the Web
    - A conceptual model instantiated at the interface
    - For example, the desktop metaphor
    - Visualizing an operation
    - For instance, an icon of a shopping cart into which the user places items

23. **Explain different types of interfaces and its usage with appropriate example**

Sure, here are ten types of interfaces along with brief descriptions:

1. **Graphical User Interface (GUI)**: A GUI uses visual elements like icons, buttons, and windows to allow users to interact with software or devices. It provides an intuitive and user-friendly experience by representing complex tasks in a graphical manner.

2. **Command Line Interface (CLI)**: CLI allows users to interact with a computer system by typing commands into a text-based interface. It's preferred by power users and developers for its efficiency and precision in executing commands and system tasks.

3. **Voice User Interface (VUI)**: VUI enables users to interact with systems using spoken commands and natural language. It's commonly used in virtual assistants and smart devices, offering hands-free and accessible control over various functions.

4. **Touchscreen Interface**: Touchscreen interfaces allow users to interact with devices by directly touching the screen with fingers or stylus. They're prevalent in smartphones, tablets, and kiosks, providing intuitive and interactive user experiences.

5. **Gesture-Based Interface**: This interface recognizes hand movements, gestures, or body motions to control devices. It's used in gaming consoles, smartphones, and augmented reality systems, offering immersive and intuitive interactions.

6. **Augmented Reality (AR) Interface**: AR interfaces overlay digital content onto the real-world environment, typically through a device's camera. They're used in applications like navigation, gaming, and training, enhancing user experiences with contextual information.

7. **Virtual Reality (VR) Interface**: VR interfaces immerse users in simulated environments through headsets or goggles, enabling immersive and interactive experiences. They're used in gaming, simulations, and training for highly immersive user interactions.

8. **Natural Language Interface**: This interface allows users to interact with systems using natural language input, such as text or speech. It's used in chatbots, virtual assistants, and search engines, providing conversational interactions for various tasks.

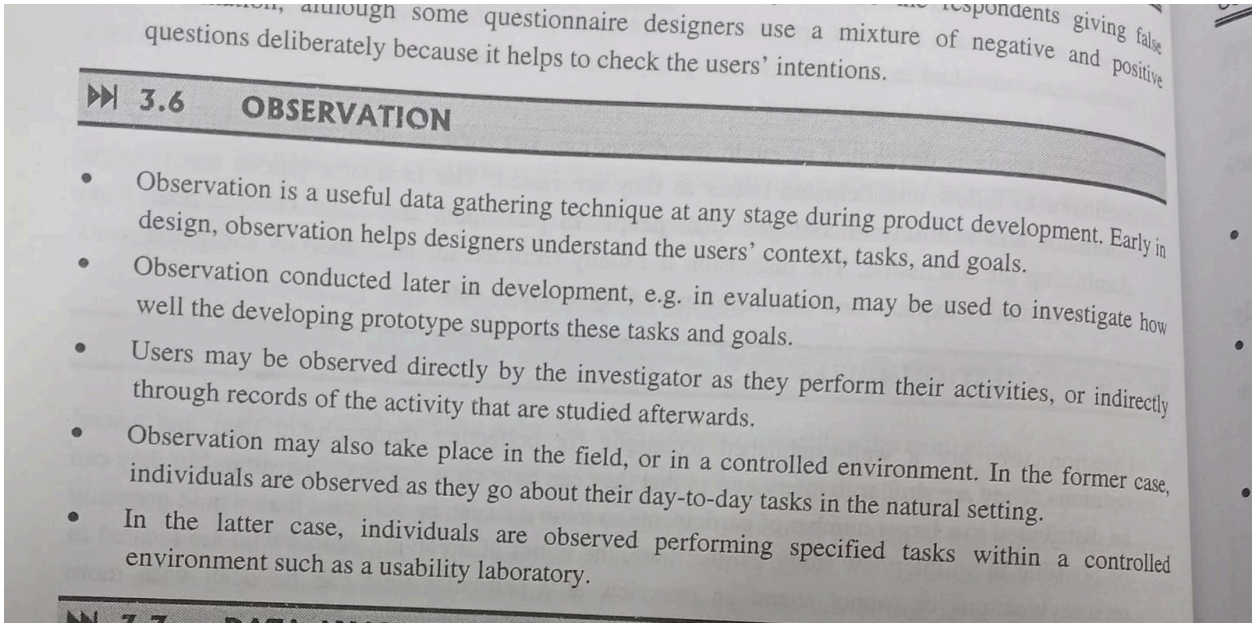
9. **Multi-Touch Interface**: Multi-touch interfaces enable users to interact with devices using multiple touch points simultaneously. They're used in smartphones, tablets, and interactive displays, allowing for gestures like pinch-to-zoom and swipe navigation.

10. **Biometric Interface**: Biometric interfaces utilize biometric data, such as fingerprints or facial scans, for user authentication and interaction. They're used in smartphones, security systems, and access control devices, providing secure and convenient user verification.

11.Virtual Reality (VR) Interface:
VR interfaces immerse users in a simulated environment, typically through a headset or goggles, allowing for immersive interactions.
Examples: VR gaming systems like Oculus Rift and HTC Vive provide users with immersive gaming experiences. VR training simulations allow users to practice skills in realistic virtual environments.

Module 3

24. Explain type and importance of ‘Observation’ in product development



although some questionnaire designers use a mixture of negative and positive questions deliberately because it helps to check the users' intentions.

respondents giving false

3.6 OBSERVATION

- Observation is a useful data gathering technique at any stage during product development. Early in design, observation helps designers understand the users' context, tasks, and goals.
- Observation conducted later in development, e.g. in evaluation, may be used to investigate how well the developing prototype supports these tasks and goals.
- Users may be observed directly by the investigator as they perform their activities, or indirectly through records of the activity that are studied afterwards.
- Observation may also take place in the field, or in a controlled environment. In the former case, individuals are observed as they go about their day-to-day tasks in the natural setting.
- In the latter case, individuals are observed performing specified tasks within a controlled environment such as a usability laboratory.

Observation is a fundamental aspect of product development, especially in user-centered design methodologies. It involves observing and studying users in their natural environments to understand their behaviors, needs, and pain points. Here's a breakdown of the type and importance of observation in product development:

Types of Observation:

1. **Contextual Observation**: This involves observing users in their natural environment while they interact with products or perform tasks related to the product's domain. Contextual observation helps designers gain insights into how users engage with products in real-life situations and identify unmet needs.

2. **User Testing**: User testing involves observing users as they interact with prototypes or early versions of the product. It helps designers evaluate usability, identify design flaws, and gather feedback for iterative improvements.

3. **Remote Observation**: In this type of observation, users are observed remotely using tools like screen recording software or remote usability testing platforms. Remote

observation allows designers to reach a wider audience and collect data from users in different locations.

Importance of Observation in Product Development:

1. **Understanding User Behavior**: Observation provides firsthand insights into how users behave, what they struggle with, and how they adapt to products in their environment. This understanding helps designers create products that better align with user needs and preferences.

2. **Identifying Pain Points**: By observing users in real-life contexts, designers can identify pain points and friction points in the user experience. This information is invaluable for improving product usability and enhancing user satisfaction.

3. **Validating Design Decisions**: Observation allows designers to validate design decisions by observing how users interact with prototypes or early versions of the product. It helps designers confirm whether their design solutions effectively address user needs and goals.

4. **Inspiring Innovation**: Observing users in their natural environments can inspire designers to think creatively and come up with innovative solutions to user problems. By understanding users' behaviors and challenges, designers can identify opportunities for innovation and differentiation in the market.

5. **Facilitating Iterative Design**: Observation is essential for the iterative design process, where designs are continually refined based on user feedback and observations. By observing users throughout the design cycle, designers can iteratively improve the product to better meet user needs and expectations.

In conclusion, observation plays a crucial role in product development by providing insights into user behavior, identifying pain points, validating design decisions, inspiring innovation, and facilitating iterative design. By incorporating observation into the design process, designers can create products that are truly user-centered and deliver superior user experiences.

25. **Explain Five key issues in Data Gathering process**

1. Setting goals
    - Decide how to analyze data once collected
2. Identifying participants
    - Decide from whom to gather data
    - How many participants are needed
3. Relationship with participants
    - Clear and professional
    - Informed consent when appropriate
4. Triangulation
    - Look at data from more than one perspective
    - Collect more than one type of data, for instance, qualitative data from experiments and qualitative data from interviews
5. Pilot studies
    - Small trial of main study

3.1 INTRODUCTION

This chapter presents some techniques for data gathering which are commonly used in interaction design activities. In particular, data gathering is a central part of establishing requirements, and of evaluation. Within the requirements activity, the purpose of data gathering is to collect sufficient, accurate, and relevant data so that a set of stable requirements can be produced; within evaluation, data gathering is needed in order to capture users' reactions and performance with a system or prototype. In this chapter we introduce three main techniques for gathering data: interviews, questionnaires, and observation.

3.2 DATA GATHERING SESSIONS : FIVE KEY ISSUES

GQ. Discuss about five key issues that required attention at a time of data gathering.

Data gathering sessions need to be planned and carried out carefully. Specific issues relating to the three data gathering techniques are discussed in the following sections, but first we **consider five key issues that require attention for any data gathering session to be successful: goal setting, identifying participants, the relationship between the data collector and the data provider, triangulation, and pilot studies.**

3.2.1 Setting Goals

- The main reason for gathering data at all is to glean information about something. For example, you might want to understand how technology fits into normal family life, or you might want to identify which of two icons representing 'send message' is easier to use, or you might want to find out whether the redesign you are planning for a hand-held meter reader is along the right lines. There are many different reasons for gathering data, and before beginning it is important to identify specific goals for the study.
- The goals that are set will influence the nature of the data gathering sessions, the data gathering techniques to be used, and also the analysis to be performed. Once the goals have been set, you can concentrate on what data to look for and what to do with it once it is gathered.
- The goals may be expressed more or less formally, e.g. using some structured or even mathematical format, or using a simple description such as the ones in the previous paragraph, but whatever the format they should be clear and concise. In interaction design it is more usual to express goals for data gathering informally.

3.2.2 Identifying Participants

- The goals you develop for your data gathering session will indicate the kind of people you want to gather data from. Those people who fit this profile are called the population. In some cases, the

people you need to gather data from may be clearly identifiable – maybe because there is a small group of users and you have access to each one.

- However, it is more likely that you will need to choose the participants to include in your data gathering, and this is called sampling. The situation where you have access to all members of your target population is called saturation sampling, but this is quite rare.
- Assuming that you will be choosing to involve a proportion of your population in data gathering, then you have two options: probability sampling or nonprobability sampling. In the former case, the most commonly used approaches are simple random sampling or stratified sampling; in the latter the most common are convenience sampling or volunteer panels. Random sampling can be achieved by using a random number generator or by choosing every nth person in a list. Stratified sampling relies on being able to divide the population into groups (e.g. classes in a secondary school), and then applying random sampling.
- Both convenience sampling and volunteer panels rely less on you choosing the participants and more on participants being prepared to take part. The term convenience sampling is used to describe a situation where the sample includes those who were available rather than those specifically selected.
- The crucial difference between probability and non-probability methods is that in the former you can apply statistical tests and generalize to the whole population, while in the latter such generalizations are not robust.
- Using statistics also requires having a sufficient number of participants. What exactly 'sufficient' means will depend on the type of data being collected and the kind of statistical tests that need to be applied. This can be a complex issue so if not confident with statistics, it is best to consult with a someone who knows about them.

3.2.3 Relationship with Participants

- One significant aspect of any data gathering is the relationship between the person (people) doing the gathering and the person (people) providing the data. Making sure that this relationship is clear and professional will help to clarify the nature of the study.
- One way in which this can be achieved is to ask participants to sign an informed consent form. The details of this form will vary, but it usually asks the participants to confirm that the purpose of the data gathering and how the data will be used have been explained to them and that they are happy to continue. It also often includes a statement that participants may withdraw at any time, and that in this case none of their data will be used in the study.
- It is common practice in many countries to use an informed consent form when running evaluation sessions, particularly where the participants are members of the public, or are volunteers in a research project. The informed consent form is intended to protect the interests of both the data gatherer and the data provider.

- The gatherer wants to know that the data she collects can be used in her analysis, presented to interested parties, and published in reports (as appropriate). The data provider wants reassurance that the information he gives will not be used for other purposes, or in any context that would be detrimental to him. For example, he wants to be sure that personal contact information and other personal details are not made public.
- This is especially true when people with disabilities or children are being interviewed. In the case of children, using an informed consent form reassures parents that their children will not be asked threatening, inappropriate, or embarrassing questions, or be asked to look at disturbing or violent images. In these cases, parents are asked to sign the form.
- However, this kind of consent is not generally required when collecting data for the requirements activity where a contract usually exists in some form between the data collector and the data provider.
- For example, consider the situation where a consultant is hired to gather data from a company in order to establish a set of requirements for a new interactive system to support timesheet entry. The employees of this company would be the users of the system, and the consultant would therefore expect to have access to the employees to gather data about the timesheet activity. In addition, the company would expect its employees to cooperate in this exercise. In this case, there is already a contract in place which covers the data gathering activity, and therefore an informed consent form is less likely to be required.
- As with most ethical issues, the important thing is to consider the situation carefully and make a judgment based on the specific circumstances. Increasingly, projects that involve collecting data from humans are being reviewed to ensure that participants' personal information is protected. Incentives for completing a questionnaire might be needed in some circumstances because there is no clear and direct advantage to the respondents, but in other circumstances, respondents may see it as part of their job to complete the questionnaire.
- For example, if the questionnaires form part of the requirements activity for a new mobile sales application to support sales executives, then it is likely that sales executives will complete a questionnaire about their job if they are told that the new device will impact their day-to-day lives. In this case, the motivation for providing the required information is clear. However, if you are collecting data to understand how appealing a new interactive website is for school children, different incentives would be appropriate. Here, the advantage for the individuals to complete a questionnaire is not so obvious.

3.2.4 Triangulation

Triangulation is a term used to refer to the investigation of a phenomenon from (at least) two different perspectives . Four types of triangulation have been defined :

(1) Triangulation of data means that data is drawn from different sources at different times, in different places, or from different people

(2) Investigator triangulation means that different researchers (observers, interviewers, etc.) have been used to collect and interpret the data.

(3) Triangulation of theories means the use of different theoretical frameworks through which to view the data or findings.

(4) Methodological triangulation means to employ different data gathering techniques.

The last of these is the most common form of triangulation. One application of triangulation (and again the most common) is to validate the results of some inquiry by pointing to similar results yielded through the use of different perspectives. However, validation through triangulation is difficult to achieve. Different data gathering methods result in different kinds of data, which may or may not be compatible. Using different theoretical frameworks may or may not result in complementary findings, but to achieve theoretical triangulation would require the theories to have similar philosophical underpinnings. Using more than one data gathering technique, and more than one data analysis approach, is good practice, but achieving true triangulation is rare.

3.2.5 Pilot Studies

- A pilot study is a small trial run of the main study. The aim is to make sure that the proposed method is viable before embarking on the real study. Data gathering participants can be (and usually are) very unpredictable, even when a lot of time and effort has been spent carefully planning the data gathering session.
- Plans should be tested by doing a pilot study before launching into the main study. For example, the equipment and instructions that are to be used can be checked, the questions for an interview or in a questionnaire can be tested for clarity, and an experimental procedure can be confirmed as viable. Potential problems can be identified in advance so that they can be corrected.
- Distributing 500 questionnaires and then being told that two of the questions were very confusing wastes time, annoys participants, and is an expensive error that could have been avoided by doing a pilot study. If it is difficult to find people to participate or if access to participants is limited, colleagues or peers can be asked to comment.
- Getting comments from peers is quick and inexpensive and can be a substitute for a pilot study. It is important to note that anyone involved in a pilot study cannot be involved in the main study. Why? Because they will know more about the study and this can distort the results.

26. Compare the outcome of different types of Interviews

GQ. Define Interview. Explain different types of Interview.

- Interviews can be thought of as a "conversation with a purpose". How like an ordinary conversation the interview can be depends on the type of interview method used.
- There are four main types of interviews: open-ended or unstructured, structured, semi-structured, and group interviews. The first three types are named according to how much control the interviewer imposes on the conversation by following a predetermined set of questions.
- The fourth involves a small group guided by a facilitator. The most appropriate approach to interviewing depends on the purpose of the interview, the questions to be addressed, and the stage in the lifecycle.
- For example, if the goal is to gain first impressions about how users react to a new design idea, such as an interactive sign, then an informal, open-ended interview is often the best approach. But if the goal is to get feedback about a particular design feature, such as the layout of a new web browser, then a structured interview or questionnaire is often better. This is because the goals and questions are more specific in the latter case.

3.4.1 Unstructured Interviews

- Open-ended or unstructured interviews are at one end of a spectrum of how much control the interviewer has over the interview process. They are exploratory and are more like conversations around a particular topic; they often go into considerable depth. Questions posed by the interviewer are open, meaning that there is no particular expectation about the format or content of answers.

- For example, the first question asked of all participants might be: 'What are the advantages of using a touch screen?' Here, the interviewee is free to answer as fully or as briefly as she wishes and both interviewer and interviewee can steer the interview. For example, often the interviewer will say: "Can you tell me a bit more about . . ." This is referred to as probing. Despite being unstructured and open, it is always advisable for the interviewer to have a plan of the main topics to be covered, so that she can make sure that all the topics of interest are included. Going into an interview without an agenda should not be confused with being open to hearing new ideas
- One of the skills necessary for conducting an unstructured interview is getting the balance right between making sure that answers to relevant questions are obtained, while at the same time being prepared to follow new lines of enquiry that were not anticipated.
- A benefit of unstructured interviews is that they generate rich data that is often interrelated and complex, i.e. data that gives a deep understanding of the topic. In addition, interviewees may mention issues that the interviewer has not considered.
- A lot of unstructured data is generated and the interviews will not be consistent across participants since each interview takes on its own format. Unstructured interviews can therefore be time consuming to analyze, although themes can often be identified across interviews using techniques from grounded theory and other approaches. These characteristics need to be taken into account when deciding which type of interview to choose.

3.4.2 Structured Interviews

- In structured interviews, the interviewer asks predetermined questions similar to those in a questionnaire, and the same questions are used with each participant so the study is standardized. The questions need to be short and clearly worded, and they are typically closed questions, which means that they require an answer from a predetermined set of alternatives (this may include an 'other' option, but ideally this would not be chosen very often).
- Closed questions work well if the range of possible answers is known, and when participants are in a rush. Structured interviews are only really useful when the goals are clearly understood and specific questions can be identified. Example questions for a structured interview might be:
  - Which of the following websites do you visit most frequently: amazon.com, google.com, msn.com?
  - How often do you visit this website: every day, once a week, once a month, less often than once a month?
  - Do you ever purchase anything online: yes/no? If your answer is yes, how often do you purchase things online: every day, once a week, once a month, less frequently than once a month?
- Questions in a structured interview should be worded exactly the same for each participant, and they should be asked in the same order.

3.4.3 Semi-structured Interviews

- Semi-structured interviews combine features of structured and unstructured interviews and use both closed and open questions. The interviewer has a basic script for guidance, so that the same topics are covered with each interviewee. The interviewer starts with preplanned questions and then probes the interviewee to say more until no new relevant information is forthcoming. For example:
- Which music websites do you visit most frequently? <Answer: mentions several but stresses that she prefers hottestmusic.com>
  - Why? <Answer: says that she likes the site layout>
  - Tell me more about the site layout <Silence, followed by an answer describing the site's layout>
  - Anything else that you like about the site? <Answer: describes the animations>
  - Thanks. Are there any other reasons for visiting this site so often that you haven't mentioned?
- It is important not to pre-empt an answer by phrasing a question to suggest that a particular answer is expected. For example, 'You seemed to like this use of color . . .' assumes that this is the case and will probably encourage the interviewee to answer that this is true so as not to offend the interviewer. Children are particularly prone to behave in this way.
- The body language of the interviewer, for example whether she is smiling, scowling, looking disapproving, etc., can have a strong influence on whether the interviewee will agree with a question, and the interviewee needs to have time to speak and not be moved on too quickly. Probes are a useful device for getting more information, especially neutral probes such as 'Do you want to tell me anything else?', and prompts which remind interviewees if they forget terms or names help to move the interview along. Semi-structured interviews are intended to be broadly replicable, so probing and prompting should aim to help the interview along without introducing bias.

3.4.4 Focus Groups

- Interviews are often conducted with one interviewer and one interviewee, but it is also common to interview people in groups. One form of group interview that is frequently used in marketing, political campaigning, and social sciences research is the focus group. Normally three to ten people are involved, and the discussion is led by a trained facilitator. Participants are selected to provide a representative sample of the target population.
- For example, in an evaluation of a university website, a group of administrators, faculty, and students may form three separate focus groups because they use the web for different purposes.
- In requirements activities it is quite common to hold a focus group in order to identify conflicts in terminology or expectations from different sections within one department or organization. The benefit of a focus group is that it allows diverse or sensitive issues to be raised that might otherwise be missed.

27. Explain the different formats of Question and Response during data gathering

•‘Yes’ and ‘No’ checkboxes

•Checkboxes that offer many options

•Rating scales

Likert scales

Semantic scales

3, 5, 7 or more points

•Open-ended responses

28. Give 2 examples of Quantitative Data Analysis methods

•Averages:

Mean: Add up values and divide by number of data points

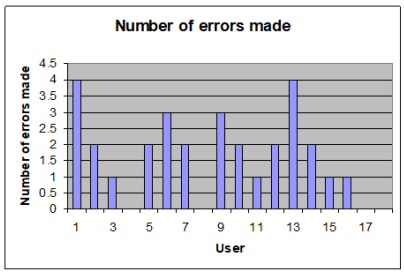
Median: Middle value of data when ranked

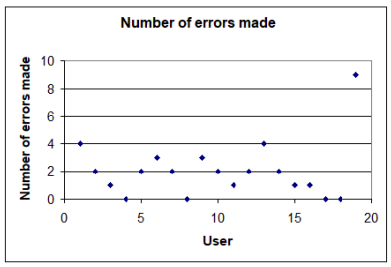
Mode: Figure that appears most often in the data

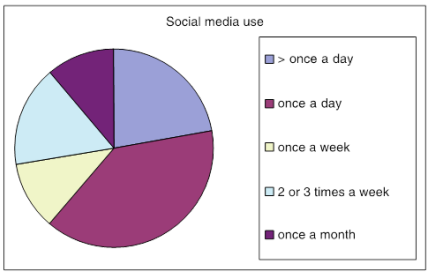
•Percentages

•Be careful not to mislead with numbers!

•Graphical representations give overview of data







29. Give **2 examples of Qualitative Data Analysis methods**

Looking for critical incidents

§Helps to focus in on key events

§Then analysis can proceed using specific techniques

•Identifying themes

§Emergent from data, dependent on observation framework if used

§Inductive analysis

•Categorizing data

§Categorization scheme pre-specified

§Deductive analysis

•In practice, combination of inductive and deductive

30. **Create the list of key requirements for designing a " Wearable interactive product to measure glucose levels for an individual with diabetes".**

Designing a wearable interactive product to measure glucose levels for an individual with diabetes involves considering several key requirements to ensure its effectiveness and usability. Here is a list of key requirements:

1. **Accurate Glucose Monitoring:** The device should provide accurate and reliable glucose measurements to ensure the user can effectively manage their diabetes.

2. **Non-Invasive or Minimally Invasive:** To improve user comfort and compliance, the device should be non-invasive or minimally invasive, such as using sensors that do not require frequent finger pricks.

3. **Continuous Monitoring:** The device should be capable of continuous or frequent monitoring of glucose levels to provide real-time data to the user.

4. **User-Friendly Interface:** The interface should be easy to use, with clear and intuitive controls for the user to interact with the device and view their glucose data.

5. **Compatibility with Mobile Devices:** The device should be compatible with mobile devices, allowing users to sync their glucose data to a mobile app for easier tracking and analysis.

6. **Long Battery Life:** To ensure continuous monitoring, the device should have a long battery life to minimize the need for frequent charging or battery replacement.

7. **Data Security and Privacy:** The device should have robust security measures in place to protect the user's glucose data from unauthorized access or breaches.

8. **Compact and Lightweight Design:** The device should be compact and lightweight, allowing users to wear it comfortably throughout the day without feeling bulky or obtrusive.

9. **Customizable Alerts and Notifications:** The device should allow users to set customizable alerts and notifications for high or low glucose levels to help them manage their condition more effectively.

10. **Integration with Health Records:** The device should be able to integrate with electronic health record systems to enable healthcare providers to access the user's glucose data for better management and monitoring.

11. **Cost-Effective:** The device should be cost-effective to ensure it is accessible to a wide range of users, including those with limited financial resources.

12. **Waterproof and Durable:** The device should be waterproof and durable to withstand daily wear and tear, including exposure to sweat and moisture.