Usability and accessibility Testing

Usability testing

- The testing that validates the ease of use, speed, and aesthetics of the product from the user's point of view is called usability testing.
- some of the characteristics of “usability testing” or “usability validation” are as follows:
  - Usability testing tests the product from the users’ point of view. It encompasses a range of techniques for identifying how users actually interact with and use the product.
  - Usability testing is for checking the product to see if it is easy to use for the various categories of users.
  - Usability testing is a process to identify discrepancies between the user interface of the product and the human user requirements, in terms of the pleasantness and aesthetics aspects.

Usability testing

From the above definition it is easy to conclude that Something that is easy for one user may not be easy for another user due to different types of users a product can have . Something what is considered fast (interms of say, response time) by one user may be slow for another user as the machines used by them and the expectations of speed can be different. Something that is considered beautiful by someone may look ugly to another. A view expressed by one user of the product may not be the view of another.

Usability testing

Throughout the industry, usability testing is gaining momentum as sensitivity towards usability in products is increasing and it is very difficult to sell a product that does not meet the usability requirements of the users. There are several standards (for example, accessibility guidelines), organizations, tools (for example, Microsoft Magnifier), and processes that try to remove the subjectivity and improve the objectivity of usability testing.

Usability testing

Usability testing is not only for product binaries or executables. It also applies to documentation and other deliverables that are shipped along with a product. The release media should also be verified for usability. Let us take an example of a typical AUTORUN script that automatically brings up product setup when the release media is inserted in the machine. Sometimes this script is written for a particular operating system version and may not get auto executed on a different OS version. Even though the user can bring up the setup by clicking on the setup executable manually, this extra click (and the fact that the product is not automatically installed) may be considered as an irritant by the person performing the installation.

Who performs Usability testing

Generally, the people best suited to perform usability testing are Typical representatives of the actual user segments who would be using the product, so that the typical user patterns can be captured, and People who are new to the product, so that they can start without any bias and be able to identify usability problems. A person who has used the product several times may not be able to see the usability problems in the product as he or she would have “got used” to the product's (potentially inappropriate) usability. Hence, a part of the team performing usability testing is selected from representatives outside the testing team. Inviting customer-facing teams (for example, customer support, product marketing) who know what the customers want and their expectations, will increase the effectiveness of usability testing.

Deliverables /Usability testing

A right approach for usability is to test every artifact that impacts users—such as product binaries, documentation, messages, media—covering usage patterns through both graphical and command user interfaces, as applicable.

Usability testing

Usability should not be confused with graphical user interface (GUI). Usability is also applicable to non-GUI interface such as command line interfaces (CLI). A large number of Unix/Linux users find CLIs more usable than GUIS. SQL command is another example of a CLI, and is found more usable by database users. Hence, usability should also consider CLI and other interfaces that are used by the users

WHEN TO DO USABILITY TESTING?

The most appropriate way of ensuring usability is by performing the usability testing in two phases. **First is design validation and the second is usability testing** done as a part of component and integration testing phases of a test cycle. When planning for testing, the usability requirements should be planned in parallel, upfront in the development cycle, similar to any other type of testing. Generally, however, usability is an ignored subject (or at least given less priority) and is not planned and executed from the beginning of the project. When there are two defects—one on functionality and other on usability—the functionality defect is usually given precedence. This approach is not correct as usability defects may demotivate users from using the software (even if it performs the desired function) and it may mean a huge financial loss to the product organization if users reject the product. Also, postponing usability testing in a testing cycle can prove to be very expensive as a large number of usability defects may end up as needing changes in design and needing fixes in more than one screen, affecting different code paths. All these situations can be avoided if usability testing is planned upfront.

WHEN TO DO USABILITY TESTING?

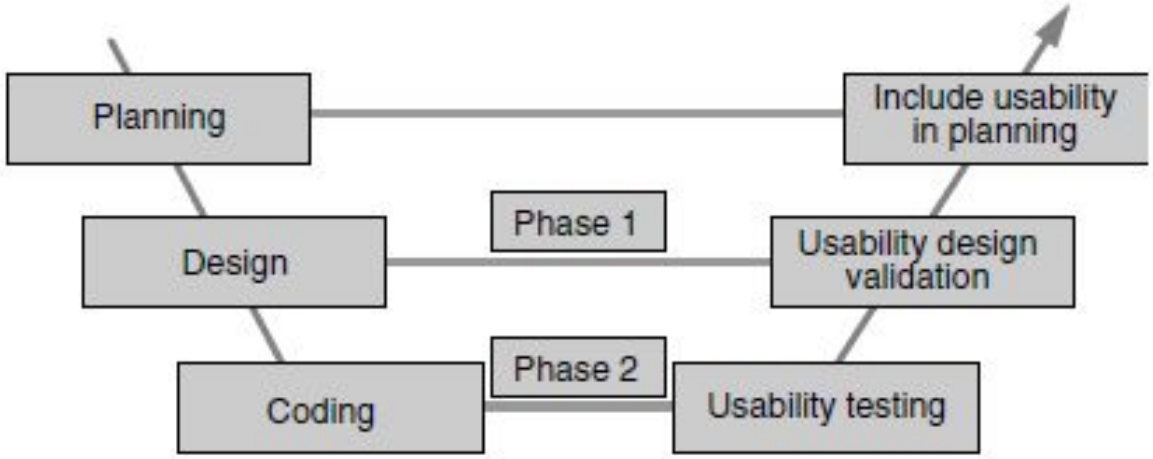

Figure 12.2 Phases and activities of usability testing.

WHEN TO DO USABILITY TESTING?

Usability design is verified through several means. Some of them are as follows:

- Style sheets : Style sheets are grouping of user interface design elements. Use of style sheets ensures consistency of design elements across several screens and testing the style sheet ensures that the basic usability design is tested. Style sheets also include frames, where each frames is considered as a separate screen by the user. Style sheets are reviewed to check whether they force font size, color scheme, and so on, which may affect usability.
- Screen prototypes : Screen prototype is another way to test usability design. The screens are designed as they will be shipped to the customers, but are not integrated with other modules of the product. Therefore, this user interface is tested independently without integrating with the functionality modules. This prototype will have other user interface functions simulated such as screen navigation, message display, and so on. The prototype gives an idea of how exactly the screens will look and function when the product is released. The test team and some real-life users test this prototype and their ideas for improvements are incorporated in the user interface. Once this prototype is completely tested, it is integrated with other modules of the product.

WHEN TO DO USABILITY TESTING?

- Paper designs Paper design explores the earliest opportunity to validate the usability design, much before the actual design and coding is done for the product. The design of the screen, layout, and menus are drawn up on paper and sent to users for feedback. The users visualize and relate the paper design with the operations and their sequence to get a feel for usage and provide feedback. Usage of style sheets requires further coding, prototypes need binaries and resources to verify, but paper designs do not require any other resources. Paper designs can be sent through email or as a printout and feedback can be collected.
- Layout design Style sheets ensure that a set of user interface elements are grouped and used repeatedly together. Layout helps in arranging different elements on the screen dynamically. It ensures arrangement of elements, spacing, size of fonts, pictures, justification, and so on, on the screen. This is another aspect that needs to be tested as part of usability design.

WHEN TO DO USABILITY TESTING?

- If an existing product is redesigned or enhanced, usability issues can be avoided by using the existing layout, as the user who is already familiar with the product will find it more usable. Making major usability changes to an existing product (for example, reordering the sequence of buttons on a screen) can end up confusing users and lead to user errors.
- **In the second phase**, tests are run to test the product for usability. Prior to performing the tests, some of the actual users are selected (who are new to the product and features) and they are asked to use the product. Feedback is obtained from them and the issues are resolved. Sometimes it could be difficult to get the real users of the product for usability testing. In such a case, the representatives of users can be selected from teams outside the product development and testing teams—for instance, from support, marketing, and sales teams. When to do usability also depends on the type of the product that is being developed.

QUALITY FACTORS FOR USABILITY

Some quality factors are very important when performing usability testing. As was explained earlier, usability is subjective and not all requirements for usability can be documented clearly. However focusing on some of the quality factors given below help in improving objectivity in usability testing are as follows.

Comprehensibility : The product should have simple and logical structure of features and documentation. They should be grouped on the basis of user scenarios and usage. The most frequent operations that are performed early in a scenario should be presented first, using the user interfaces. When features and components are grouped in a product, they should be based on user terminologies, not technology or implementation.

Consistency A product needs to be consistent with any applicable standards, platform look-and-feel, base infrastructure, and earlier versions of the same product. Also, if there are multiple products from the same company, it would be worthwhile to have some consistency in the look-and-feel of these multiple products. Following some standards for usability helps in meeting the consistency aspect of the usability.

WHEN TO DO USABILITY TESTING?

Navigation This helps in determining how easy it is to select the different operations of the product. An option that is buried very deep requires the user to travel to multiple screens or menu options to perform the operation. The number of mouse clicks, or menu navigations that is required to perform an operation should be minimized to improve usability. When users get stuck or get lost, there should be an easy option to abort or go back to the previous screen or to the main menu so that the user can try a different route.

Responsiveness How fast the product responds to the user request is another important aspect of usability. This should not be confused with performance testing. Screen navigations and visual displays should be almost immediate after the user selects an option or else it could give an impression to the user that there is no progress and cause him or her to keep trying the operation again. Whenever the product is processing some information, the visual display should indicate the progress and also the amount of time left so that the users can wait patiently till the operation is completed. Adequate dialogs and popups to guide the users also improve usability.

USABILITY TESTING : AESTHETICS TESTING

AESTHETICS TESTING : Another important aspect in usability is making the product “beautiful.” Performing aesthetics testing helps in improving usability further. This testing is important as many of the aesthetics related problems in the product from many organizations are ignored on the ground that they are not functional defects. All the aesthetic problems in the product are generally mapped to a defect classification called “Cosmetic,” which is of low priority. Having a separate cycle of testing focusing on aesthetics helps in setting up expectations and also in focusing on improving the look and feel of the user interfaces. Aesthetics is not in the external look alone. It is in all the aspects such as messages, screens, colors, and images. A pleasant look for menus, pleasing colors, nice icons, and so on can improve aesthetics.

Accessibility testing

Accessibility Testing is a subset of usability testing, and it is performed to ensure that the application being tested is usable by people with disabilities like hearing, color blindness, old age and other disadvantaged groups.

- People with disabilities use assistive technology which helps them in operating a software product.

Accessibility testing involves testing these alternative methods of using the product and testing the product along with accessibility tools. Accessibility is a subset of usability and should be included as part of usability test planning.

Verifying the product usability for physically challenged users is called accessibility testing.

Accessibility testing

Accessibility testing may be challenging for testers because they are unfamiliar with disabilities. It is better to work with disabled people who have specific needs to understand their challenges.

Accessibility to the product can be provided by two means :

Making use of accessibility features provided by the underlying infrastructure (for example, operating system), called basic accessibility, and

Providing accessibility in the product through standards and guidelines, called product accessibility.

Accessibility testing

Basic Accessibility: Basic accessibility is provided by the hardware and operating system. All the input and output devices of the computer and their accessibility options are categorized under basic accessibility.

Examples:

Keyboard accessibility

Screen accessibility

- **Speech Recognition Software -** It will convert the spoken word to text , which serves as input to the computer.
- **Screen reader software** - Used to read out the text that is displayed on the screen
- **Screen Magnification Software**- Used to enlarge the monitor and make reading easy for vision-impaired users.
- **Special keyboard** made for the users for easy typing who have motion control difficulties

Accessibility testing

Product Accessibility :A good understanding of the basic accessibility features is needed while providing accessibility to the product. A product should do everything possible to ensure that the basic accessibility features are utilized by it. For example, providing detailed text equivalent for multimedia files ensures the captions feature is utilized by the product.

Accessibility testing

Sample requirement #1: Text equivalents have to be provided for audio, video, and picture images.

Sample requirement #2: Documents and fields should be organized so that they can be read without requiring a particular resolution of the screen, and templates (known as style sheets).

Sample requirement #3: User interfaces should be designed so that all information conveyed with color is also available without color.

Sample requirement #4: Reduce flicker rate, speed of moving text; avoid flashes and blinking text.

Sample requirement #5: Reduce physical movement requirements for the users when designing the interface and allow adequate time for user responses.

Table 12.2 Sample list of usability and accessibility tools.

Name of the tool	Purpose
JAWS	For testing accessibility of the product with some assistive technologies.
HTML validator	To validate the HTML source file for usability and accessibility standards.
Style sheet validator	To validate the style sheets (templates) for usability standards set by W3C.
Magnifier	Accessibility tool for vision challenged (to enable them to enlarge the items displayed on screen
Narrator	Narrator is a tool that reads the information displayed on the screen and creates audio descriptions for vision-challenged users.
Soft keyboard	Soft keyboard enables the use of pointing devices to use the keyboard by displaying the keyboard template on the screen.

Accessibility testing

Following are the point's needs to be checked for application to be used by all users. This checklist is used for signing off accessibility testing.

- Whether an application provides keyboard equivalents for all mouse operations and windows?
- Whether instructions are provided as a part of user documentation or manual? Is it easy to understand and operate the application using the documentation?
- Whether tabs are ordered logically to ensure smooth navigation?
- Whether shortcut keys are provided for menus?
- Whether application supports all operating systems?
- Whether color of the application is flexible for all users?
- Whether images or icons are used appropriately, so it's easily understood by the end users?

Accessibility testing

- Whether an application has audio alerts?
- Whether user can adjust or disable flashing, rotating or moving displays?
- Check to ensure that color-coding is never used as the only means of conveying information or indicating an action
- Whether highlighting is viewable with inverted colors? Testing of color in the application by changing the contrast ratio
- Whether audio and video related content are properly heard by the disability people ? Test all multimedia pages with no speakers in websites
- Whether training is provided for users with disabilities that will enable them to become familiar with the software or application?