Based on National Curriculum of Pakistan 2022-23

Textbook of

Computer Science

Grade

12

National Curriculum Council
Ministry of Federal Education and Professional Training

National Book Foundation
as
Federal Textbook Board
Islamabad

Government Approval
Approved by the National Curriculum Council (NCC), Ministry of Federal Education and Professional Training, Islamabad vide letter No. F.No.1-1(2)TB/CS/NBF/NCC, dated 4th June, 2025

A Textbook of Computer Science for Grade 12
based on National Curriculum of Pakistan (NCP) 2022-23

Authors
Mr. Mohammad Sajjad Heder (Managing Author)
Ms. Lubna Kousar, Mr. Shah Islam Khan, Mr. Mohammad Khalid,
Dr. Saleem Iqbal, Dr. Faraz Ahsan, Dr. Mian Hamid Hassan (Co-Authors)

Contributors/Collaborators
Muhammad Zeeshan Lilla (Designer)

Supervision of Curriculum and Review Process
Dr. Tabassum Naz
Director, National Curriculum Council (NCC)
Ministry of Federal Education and Professional Training, Government of Pakistan, Islamabad

NCC Review Committee Members
Ms. Sadaf Zehra, Mr. Saghir Ahmed, Ms. Rozina Faheem
Desk Officer: Bushra Anwar, (Education Officer), National Curriculum Council (NCC)

Supervision of the FBISE Review Committee
Mr. Mirza Ali, Director (Test Development)

FBISE Review Committee Members
Ms. Rozina Faheem, Ms. Sadaf Zehra Kazmi, Ms. Sadia Mujtaba, Mr. Saghir Ahmed
Desk Officer: Syed Zulfiqar Shah, Deputy Secretary, Research and Acadamics

---

NBF Textbooks Development Supervision
Dr. Kamran Jahangir
Managing Director, National Book Foundation (NBF)

In-Charge, NBF Textbooks Development
Mansoor Ahmad, Deputy Director

Printed in Pakistan
First Edition: First Impression: June 2025 | Pages: 216 | Quantity: 80000

TEST EDITION

Price: PKR 300/-, **Code:** STE-733, **ISBN:** 978-969-37-1828-7
Printer: The Printman Printers & Publishers Peshawar

For details on additional publications from the National Book Foundation, please visit our website at www.nbf.org.pk
You can also reach us by phone at 051 9261125 or via email at books@nbf.org.pk
For feedback or corrections, kindly send your comments to 'nbftextbooks@gmail.com' and 'textbooks@snc.gov.pk'

Note
All illustrations, artwork, and images in this book are intended solely for educational and promotional purposes, benefiting the public interest.

PREFACE

This textbook has been developed by NBF according to the National Curriculum of Pakistan 2022- 2023. The aim of this textbook is to enhance learning abilities through inculcation of logical thinking in learners, and to develop higher order thinking processes by systematically building upon the foundation of learning from the previous grades. A key emphasis of the present textbook is on creating real life linkages of the concepts and methods introduced. This approach was devised with the intent of enabling students to solve daily life problems as they go up the learning curve and for them to fully grasp the conceptual basis.

After amalgamation of the efforts of experts and experienced authors, this book was reviewed and finalized after extensive reviews by professional educationists. Efforts were made to make the contents student friendly and to develop the concepts in interesting ways.

The National Book Foundation is always striving for improvement in the quality of its books. The present book features an improved design, better illustration and interesting activities relating to real life to make it attractive for young learners. However, there is always room for improvement and the suggestions and feedback of students, teachers and the community are most welcome for further enriching the subsequent editions of this book.

May Allah guide and help us (Ameen).

Dr. Kamran Jahangir

Managing Director

The Practical Importance of the Textbook in Everyday Life

The following is a unit-by-unit summary of what is presented in each unit with its practical importance in our everyday life.

UNIT 1 Computer Systems: This unit focuses on Human Computer Interaction (HCI). HCI is a field of study about designing and development of interactive systems that are efficient, effective and user-friendly while using digital devices. HCI plays an important role in interaction between the computer system and the user in providing ways through which the user can input data, receive feedback and navigate through the system.

UNIT 2 Computational Thinking & Algorithms: This unit is about developing algorithms which is the first step for writing computer programs. Learning how to develop algorithms holds immense significance for students as it enhances their ability to think critically and provide structured approach to computational problem-solving

UNIT 3 Programming Fundamentals: In this unit students learn programming fundamentals in Python. Python is a high level object-oriented programming language. This unit describes advanced programming constructs such as data structures, file handling and databases and how to implement complex algorithms in Python. In today's modern world, having computer programming skill is essential as it plays an important role in automation of tasks that we perform in our daily life.

UNIT 4 Data and Analysis: This unit focuses on data analysis. It contains material on Machine Learning (ML). ML is a powerful technology that is transforming many industries by making processes efficient and effective. It provides solutions to many problems related to various fields such as marketing and advertising, financial services, healthcare and self-driving cars. Material on data visualization is also explored which is representation of information in the form of chart, graph, maps, diagram, etc.

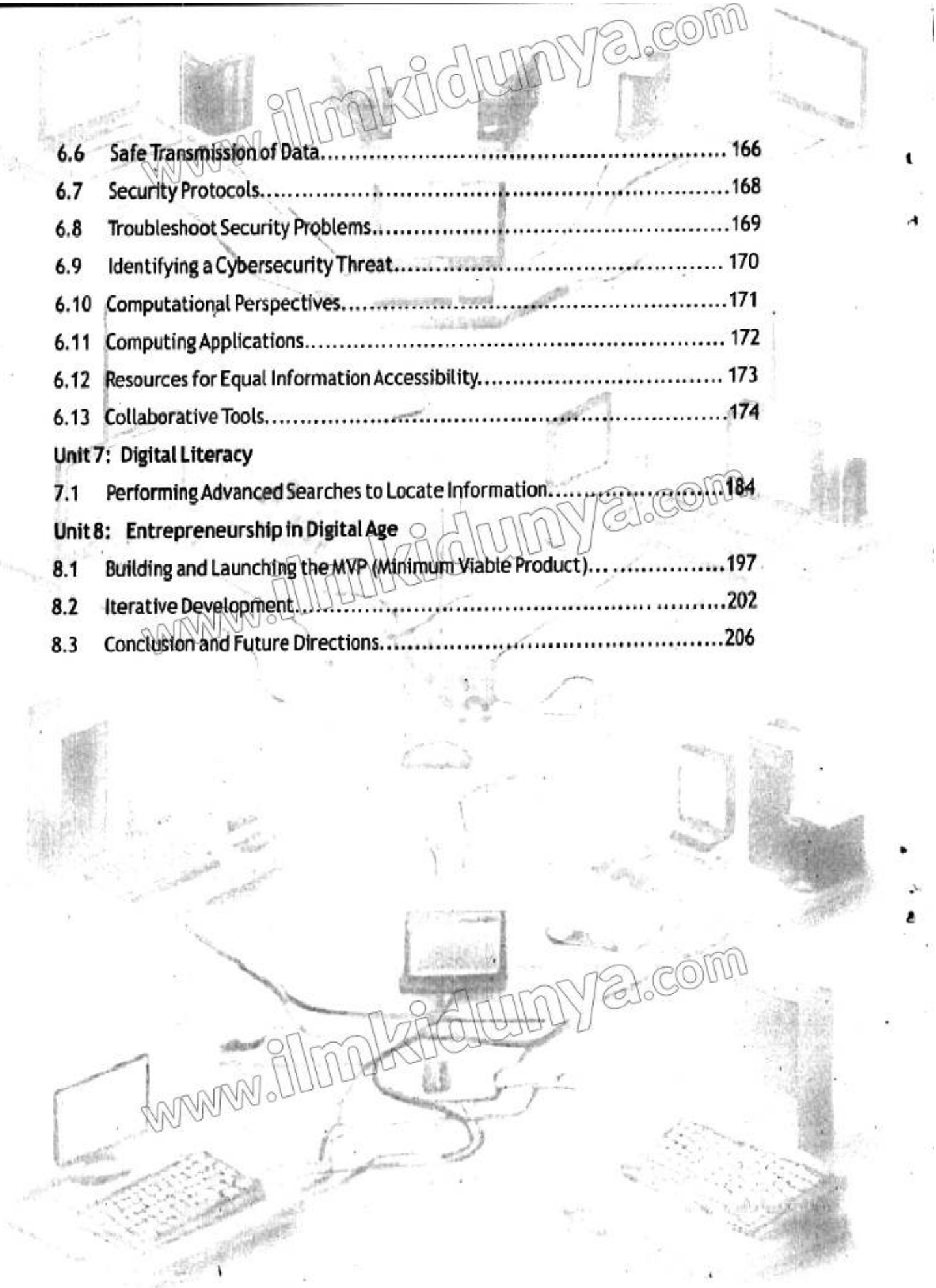
UNIT 5 Applications of Computer Science: This unit describes the applications of Internet of Things (IoT), Blockchain and Cloud Computing that are applicable in government and private sectors in Pakistan. Applications of these enhances our daily life by improving productivity, efficiency and security. Material on neural networks and deep learning is also presented that is used for development of complex systems. Data sharing and privacy conflicts are explored at the end of the unit to understand issues related with it.

UNIT 6 Impacts of Computing: This unit focuses on issues faced by people when collaborating on digital or online platforms. Today, much of our daily life revolves around the Internet and this exposes us to a wide range of online threats. This unit describes security measures and safe practices such as 2FA, biometric verification and secure ways for transmitting data, to mitigate online threats.

UNIT 7 Digital Literacy: This unit presents material that describes the creation of artefact that answers a research question, communicates results and conclusions through digital resources or tools. It plays an important role in strengthening research, analytical skills and fosters digital literacy.

UNIT 8 Entrepreneurship in Digital Age: This unit explores the creation of a Minimum Viable Product (MVP) for entrepreneurs and businesses. MVP provides low-risk testing ground before making huge investment into a product. It allows entrepreneurs and businesses to validate their business ideas, minimize costs and mitigate risks while launching a new product.

Table of Contents



1 Computer Systems

Learning Outcomes

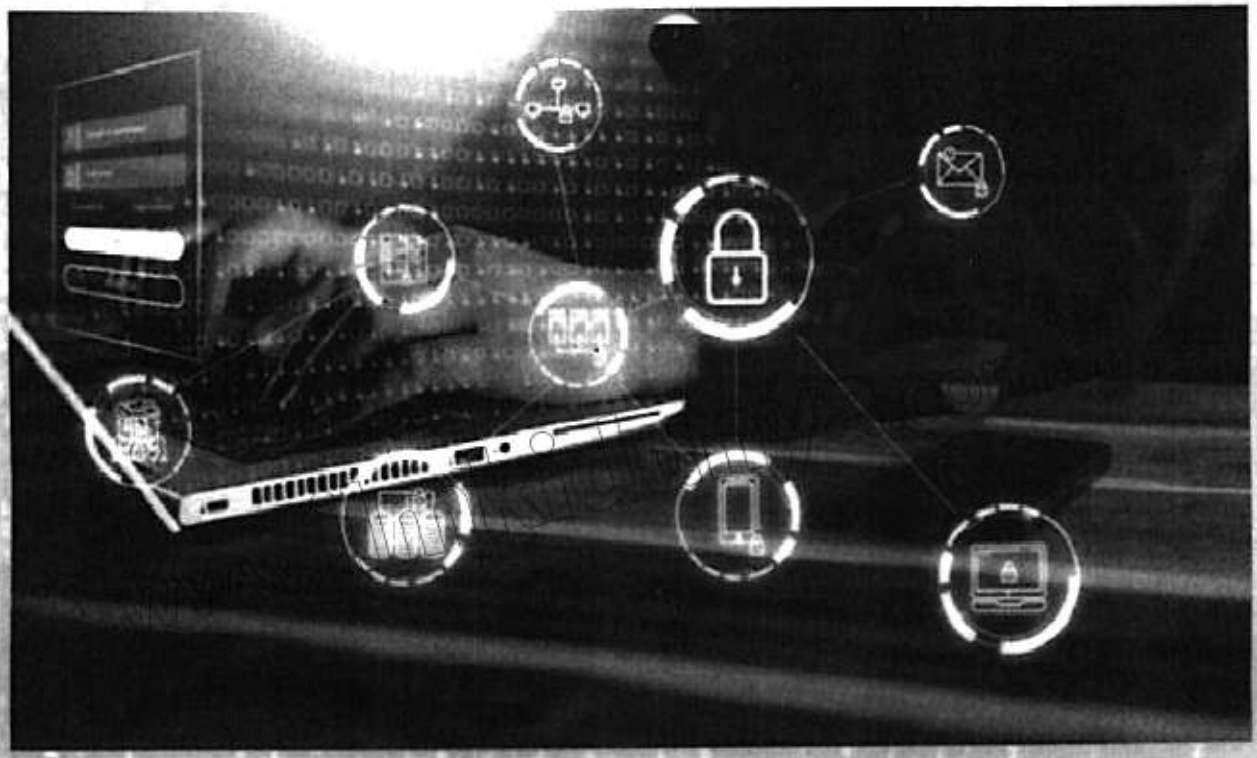
At the end of this unit students will be able to:

explain the usability, security and accessibility of devices, the systems they are integrated with.

explain human interaction with computer systems in terms of:

- Usability
- Common problems
- Methods for improvements
- Ethical, social, economic, and environmental implications

identify and explain tradeoffs between the usability and security of computing systems, recommend cybersecurity measures by considering different factors such as efficiency, cost, privacy, and ethics.



Introduction

In today's rapidly evolving digital world, technology plays a central role in our everyday lives. Whether we are using smartphones, accessing online services, or interacting with smart devices at home, the design and functionality of these technologies deeply impact our experiences. As we dive deeper into the world of technology, three key concepts emerge as critical to our interaction with these systems. These are **usability, security, and accessibility.**

- **Usability** ensures that devices and systems are easy to use and understand. Without it, even advanced technology can become frustrating and ineffective.
- **Security** protects our data and interactions in a world of evolving cyber threats. Strong measures are essential to safeguard sensitive information and prevent severe consequences.
- **Accessibility** guarantees that technology is available to everyone, including those with disabilities. As our world becomes increasingly digital, it is important that no one is left behind. Accessible design makes sure that all users, regardless of their physical abilities, can fully interact with and benefit from technology.

What will You Learn?

This unit explores the key principles of usability, security, and accessibility, highlighting their role in designing effective and inclusive technology. You will also learn about **Human-Computer Interaction (HCI)** and its focus on user needs. Additionally, the unit examines the **trade-offs between usability and security,** providing insights into how designers and engineers balance these often conflicting requirements to create systems that protect users without compromising their experience.

1.1 System/Device Usability, Security, and Accessibility

When devices and systems are designed, they should be easy to use (usability), safe and secure from threats (security), and accessible to everyone, including people with disabilities (accessibility). Understanding these aspects is crucial because they affect how we interact with technology and how technology affects our lives.

1.1.1 Device/System Usability

Usability is a key factor in the success of any device or system. It refers to how easily and efficiently users can interact with technology, directly affecting their satisfaction and productivity. By applying usability principles, designers can create user-friendly systems that meet user needs and provide a seamless experience.

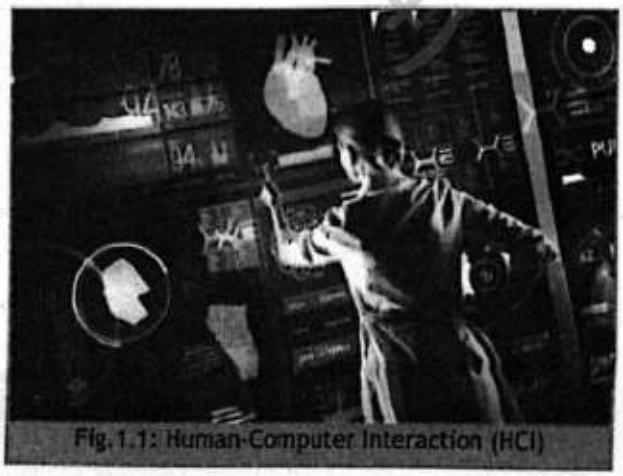


Fig.1.1: Human-Computer Interaction (HCI)

A well-designed system with high usability is easy to learn, simple to use, and aligned with user expectations. It enables users to achieve their

goals effectively, efficiently, and with minimal effort, ensuring a positive and satisfying experience. Fig.1.1 illustrates such a system.

Key Aspects of Usability:

The following are the key aspects of Usability.

- **Effectiveness:** The degree to which users can successfully complete tasks using the device or system.
- **Efficiency:** The speed and minimal effort with which tasks can be accomplished.
- **Satisfaction:** The user's sense of comfort and positive experience while interacting with the system.

Principles of Usability

Several key principles guide the design of user-friendly systems, ensuring they meet the needs and expectations of their users. These principles include:

- Simplicity
- Consistency
- Feedback
- Error Prevention
- Learnability
- Accessibility

1. Simplicity:

The design should be as straightforward as possible, removing unnecessary complexity and focusing on core functionalities.

Example: The Apple iPod (Fig.1.2), with its simple click-wheel interface, allowed users to navigate music/video quickly and easily, without the clutter of extraneous buttons.

Fig.1.2: Simplicity

Google's homepage (Fig.1.3) is iconic for its simplicity, concentrating on the primary function—searching—without overwhelming users with excessive information or options.

Fig.1.3: Google homepage

2. Consistency:

Interfaces should maintain a consistent design across different parts of the system, making it easier for users to predict how the system will behave and reducing the cognitive load required to use it.

Example: Microsoft Office uses similar menus and toolbars across Word, Excel, and PowerPoint, helping users transition smoothly between applications, as shown in Fig. 1.4.

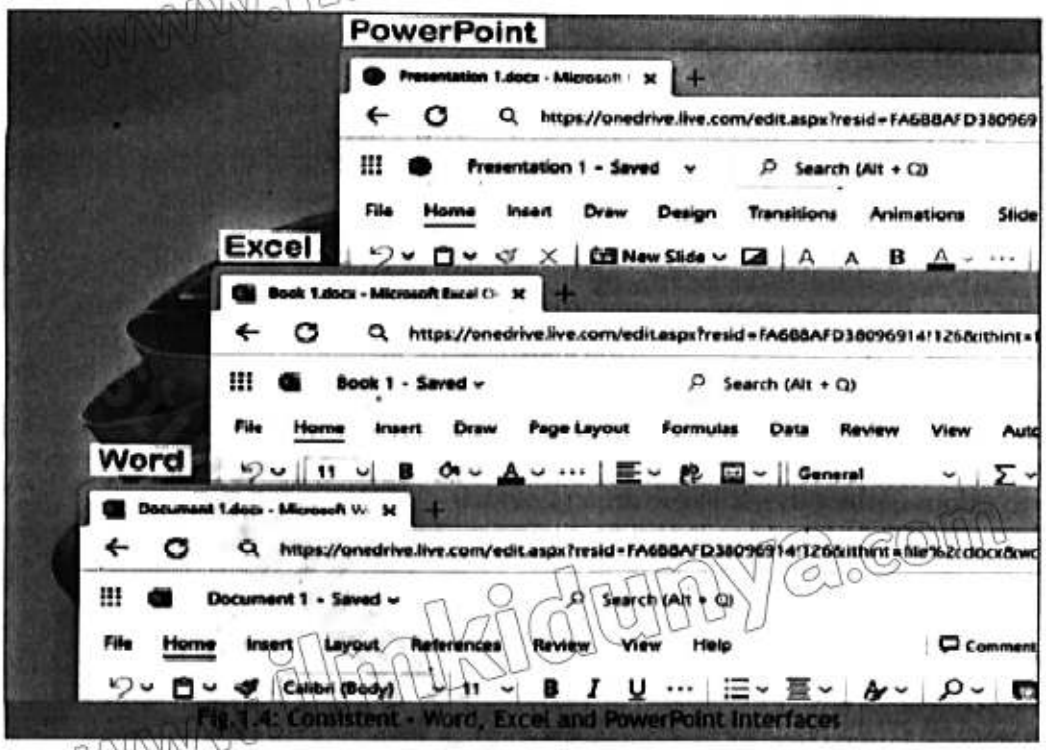


Fig. 1.4: Consistent - Word, Excel and PowerPoint Interfaces

3. Feedback:

The system should provide clear, immediate feedback to users regarding their actions, helping them understand if their inputs were successful and what the system is doing in response.

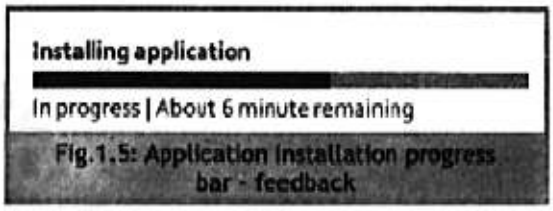


Fig. 1.5: Application installation progress bar - feedback

Example: E-commerce sites like Amazon provide immediate feedback when an item is added to the shopping cart, often with a pop-up or visual confirmation, reassuring the user that their action was successful. Another example is the progress bar in software installations, as shown in Fig. 1.5, which indicates the ongoing process and how much time is remaining.

4. Error Prevention:

The design should prevent errors before they occur and offer easy ways to correct them if they do.

Example: When you try to close a document without saving, a word processor like Microsoft Word prompts you to save your work, preventing accidental data loss. As shown in Fig. 1.6.

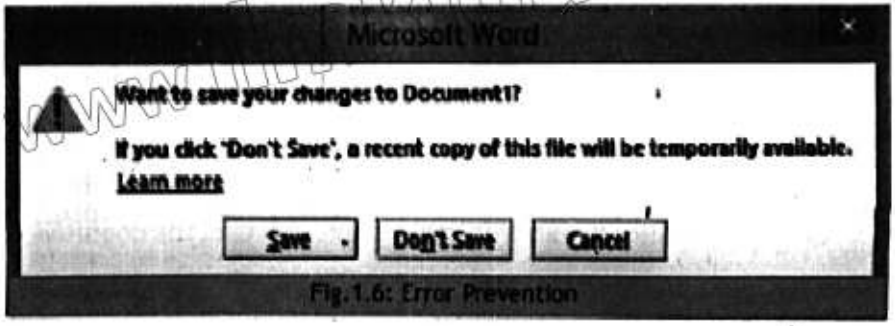


Fig. 1.6: Error Prevention

5. Learnability:

The system should be easy for new users to learn, allowing them to become proficient quickly, with minimal training or guidance.

Example: Mobile apps often include a tutorial or walkthrough for first-time users, helping them get started quickly.

6. Accessibility:

The system should be designed so that it can be easily used by everyone, regardless of their abilities. This includes making sure that people with disabilities can also use it comfortably. The goal is to ensure that the system is inclusive, meaning it works well for as many people as possible, no matter their physical or cognitive abilities.

Fig.1.7: Voice Over feature on Phones

Example1: Voice Over feature on Phones (Fig.1.7)

Voice Over feature allows visually impaired users to interact with their Phones by reading out text on the screen. Users can navigate through apps, read messages, and even use the camera with voice guidance, making the device accessible to those who are blind or have low vision.

Example2: TV Remote Controls (Device Accessibility)

Some TV remotes are designed with large buttons and high-contrast colors to assist users with vision or dexterity issues. These remotes are easier to see and press, making them more accessible to elderly users, as shown in Fig.1.8.

Fig.1.8: TV Remote Controls

1.1.2 Usability Testing

Usability testing is a key method used to evaluate how user-friendly and effective a system or device is. There are several types of usability testing, each with a specific focus and approach.

The following are some common types of usability testing methods.

1. Moderated Usability Testing

In this type of testing, a facilitator or moderator is present to guide users through tasks, ask questions, and observe behaviors in real-time. The moderator can also interact with users to clarify instructions or probe deeper into specific issues. Fig. 1.9 shows the whole process.

Example: A company developing a new software application conducts moderated usability testing by inviting participants to a lab environment. The moderator asks users to perform specific tasks, such as navigating through the app's features or completing a transaction. The moderator observes how easily users can accomplish these tasks and asks questions about their experience, helping to identify any usability issues.

Fig.1.9: Moderated Usability Testing

2. Unmoderated Usability Testing

In unmoderated usability testing, users complete tasks on their own without a moderator's direct involvement. This type of testing is often conducted remotely, allowing participants to use the system or device in their natural environment. The sessions are typically recorded for later analysis, as shown in Fig. 1.10.


Fig.1.10: Unmoderated Usability Testing

Example: An e-commerce website wants to test how easily customers can find and purchase products. The company conducts unmoderated usability testing by sending out tasks for users to complete at home, such as searching for a product, adding it to the cart, and completing the checkout process. The user interactions are recorded, and the company reviews the footage to identify any pain points or areas of confusion.

3. Qualitative Usability Testing

This approach focuses on understanding the user's experience, thoughts, and feelings while interacting with the system or device. The goal is to gather in-depth insights into user behavior, including what they find intuitive or challenging. As shown in Fig.1.11.


Fig.1.11: Qualitative Usability Testing

Example: A mobile app developer conducts qualitative usability testing by interviewing users after they use the app. Users are asked open-ended questions about their experience, such as what they liked or disliked and any challenges they encountered. This feedback helps the developer understand the user's perspective and make improvements to the app's design.

4. Quantitative Usability Testing


Fig.1.12: Quantitative Usability Testing

Quantitative usability testing collects numerical data to measure usability aspects like task completion time, error rates, and success rates. This method is useful for comparing different versions of a system or device or setting benchmarks for usability.

Example: A company tests two versions of a website's navigation menu. In a quantitative usability test, users are timed as they attempt to find specific information using each menu version. The company collects data on how long it takes users to complete the tasks and how often they make errors. This data is then analyzed to determine which menu is more efficient and user-friendly. As shown in Fig.1.12.

5. Remote Usability Testing

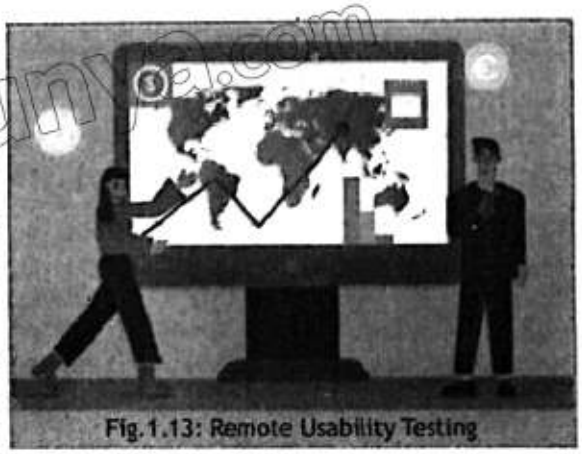


Fig.1.13: Remote Usability Testing

Remote usability testing allows users to test a system or device from their own location, using their own devices. This type of testing can be either moderated or unmoderated, and it is particularly useful for reaching a wider audience or observing how users interact with a product in their natural environment.

Example: A global software company wants to test a new feature with users in different countries. Remote usability testing is conducted, with participants using the software on their own computers while being observed through screen-sharing technology. The company gains insights into how the feature performs in different regions and environments without needing to bring users into a lab. As shown in Fig.1.13.

6. A/B Testing

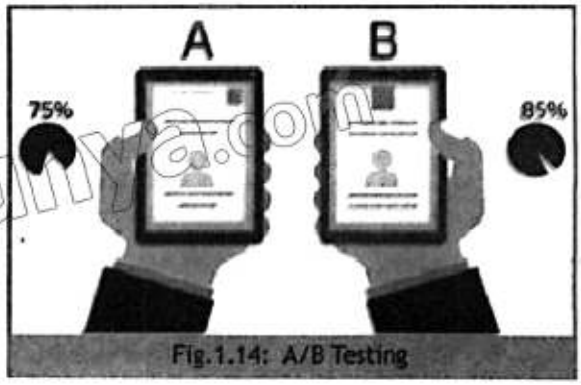


Fig.1.14: A/B Testing

A/B testing, also known as split testing, involves comparing two versions of a system or device to see which one performs better in terms of usability, as shown in Fig.1.14. This type of testing is commonly used in web design and online services.

Example: An online retailer wants to determine whether a new homepage layout will lead to more sales. They create two versions of the homepage and use A/B testing to randomly show each version to different users. By comparing the sales metrics and user engagement data between the two versions, the retailer can determine which layout is more effective.

1.1.3 Common Usability Issues and How to Avoid Them

Usability issues can significantly impact the effectiveness and user satisfaction of a system or device. When users encounter problems with usability, they may become frustrated, make errors, or even abandon the product altogether. Understanding common usability issues and knowing how to avoid them is crucial in designing user-friendly systems.

The following are some common usability issues, along with practical solutions and examples.

1. Complex Interfaces

Complex interfaces are those that overwhelm users with too much information, too many options, or overly complicated layouts. When users are faced with an interface that is difficult to understand or navigate, they are more likely to make mistakes, take longer to complete tasks, or feel frustrated.

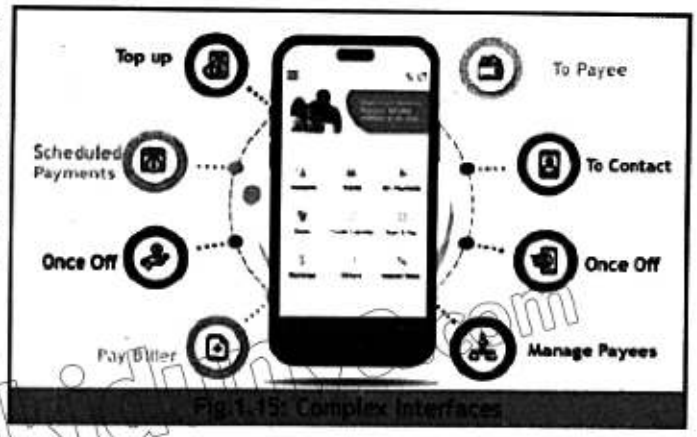


Fig.1.15: Complex Interfaces

Solution: To avoid complexity, focus on simplicity. Simplify the interface by prioritizing core functions and removing unnecessary features. The design should highlight the most important tasks and provide a clear and intuitive path for users to accomplish them.

Example: Example: Consider a mobile banking app (Fig.1.15) that initially has dozens of features crammed into a single screen. Users might find it difficult to locate basic functions like checking their balance or transferring funds. By simplifying the interface—perhaps by organizing features into easily accessible categories or hiding advanced options behind menus—the app becomes much easier to use. A good example is the early versions of the PayPal app, which went through multiple iterations to simplify its interface, eventually focusing on core actions like sending money and checking balances on the main screen.

2. Inconsistent Navigation

Inconsistent navigation occurs when the navigation structure or elements change from one section of the system to another. This inconsistency can confuse users, making it difficult for them to predict how to move around the system or find the information they need. As a result, users may feel lost or frustrated.

Solution: To maintain consistency, ensure that navigation menus, buttons, and other interface elements remain uniform across all parts of the system. Consistency in design allows users to develop a mental model of how the system works, making it easier for them to navigate.

Example: Imagine a website where the "Contact Us" button is located at the top of the page on some sections and at the bottom on others. Users might struggle to find the contact information depending on where they are on the site. A solution would be to place the "Contact Us" button in the same location on every page, such as in a fixed header or footer. Amazon's website, for

instance, keeps its navigation consistent across different pages, with the search bar and navigation menu always in the same place, helping users find what they need quickly.

3. Poor Error Messages

Fig.1.16: Poor Error Messages

Poor error messages are those that are unclear, overly technical, or provide little to no guidance on how to resolve the issue. When users encounter such error messages, they can become confused, frustrated, and unable to continue using the system effectively.

Solution: Write error messages that are clear, concise, and helpful. Good error messages should explain what went wrong in plain language and provide actionable steps that users can take to fix the problem.

Example: Consider an online form where a user enters an incorrect date format, and the system simply displays "Error 150: Invalid input.", as shown in Fig.1.16." This message is vague and doesn't help the user correct the mistake. Instead, the error message should read something like, "Please enter the date in the format MM/DD/YYYY." This clear guidance helps the user understand the issue and correct it immediately. Google's forms often provide such clear and helpful error messages, ensuring users can easily fix any mistakes.

4. Lack of Accessibility

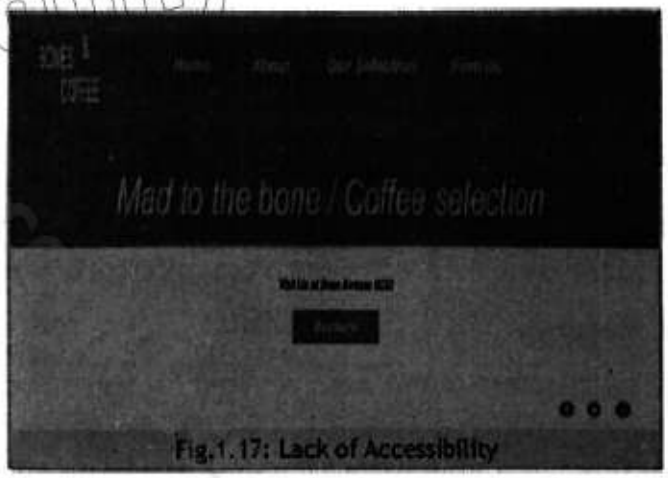


Fig.1.17: Lack of Accessibility

A lack of accessibility means that the system or device is not designed to accommodate users with disabilities, such as those who are visually impaired, hearing impaired, or have motor disabilities. Ignoring accessibility can exclude a significant portion of potential users and lead to a poor user experience for those with disabilities.

Solution: Implement accessibility features to make the system usable for people with a wide range of abilities. This includes adding screen reader support, keyboard navigation, alternative text for images, and options for adjusting text size and color contrast.

Example: Consider a website with small text and low contrast colors that are difficult for visually impaired users to read, as shown in Fig.1.17. By implementing accessibility features like text enlargement options, high-contrast color schemes, and ensuring the website is compatible with screen readers, the website becomes more accessible. Apple's website, for example, offers a high-contrast mode and supports screen readers, making it accessible to users with visual impairments.

1.1.4 Device/System Security

Security in the context of devices and systems refers to the measures and controls implemented to protect information and systems from unauthorized access, use, disclosure, disruption,

modification, or destruction. It aims to safeguard data and ensure that systems function as intended. The following are the key objectives of security.

Key Objectives:

Confidentiality: Ensuring that sensitive information is accessible only to authorized users.

Integrity: Protecting data from being altered or tampered with by unauthorized individuals.

Availability: Ensuring that systems and data are available for use when needed.

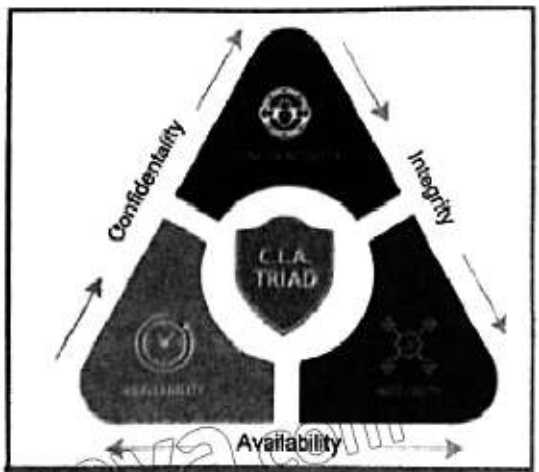


Fig.1.18: Principles of Security

Principles of Security:

The development and maintenance of secure systems rely on several foundational principles, commonly referred to as the CIA triad—Confidentiality, Integrity, and Availability, as shown in Fig.1.18. These principles serve as the cornerstone for designing systems that protect data and ensure that it remains safe and accessible under various conditions.

1. Confidentiality

Confidentiality involves protecting sensitive information from unauthorized access, ensuring that only those who have the appropriate permissions or clearance can view or use the data.

Example: Consider a company's database that stores customer credit card information. To keep this data confidential, the company might encrypt the credit card numbers before storing them in the database. This way, even if a hacker gains access to the database, they would not be able to read or misuse the credit card information without the decryption key.

2. Integrity

Integrity refers to the accuracy and consistency of data. It ensures that information remains unaltered during storage, transmission, or processing, except by those who are authorized to do so. This principle protects data from being tampered with or corrupted.

Example: When a user sends an important email, such as a contract, he/she might digitally sign it. A digital signature is a way to ensure that the content of the email has not been altered after it is sent. If the recipient receives the email with a valid signature, they can be confident that the content is exactly as intended and has not been tampered with.

3. Availability

Availability ensures that systems and data are accessible and usable when needed, even during or after a disruption. This principle involves designing systems to withstand and recover from failures or attacks quickly.

Example 1: To guarantee availability, a business might implement more than one servers and data storage. For instance, if one server fails, another can take over immediately, preventing stoppage and ensuring that users can continue to access the system without interruption.

Security Policies and Best Practices:

Implementing strong security policies and best practices is essential for protecting devices and systems.

Data Protection Policies: These are guidelines for handling, storing, and transmitting sensitive information to prevent unauthorized access or disclosure. For example a company policy requiring encryption of all sensitive customer data before transmission.

Regular Security Audits: Audits are conducting thorough assessments of systems and networks to identify vulnerabilities and ensure compliance with security standards. For example an organization performing quarterly security audits to check for any gaps in their defenses.

Employee Training and Awareness: Educating employees on security risks and best practices to prevent human error, which is a common cause of security breaches.

Incident Response Planning: Preparing a plan to quickly and effectively respond to security incidents to minimize damage.

Learning from Security Breaches

Case Study: The Target Data Breach of 2013

Overview

In December 2013, Target Corporation, a major U.S. retail chain, suffered a significant data breach that compromised the personal and financial information of millions of customers.

Key Details

- How it Happened: Hackers accessed Target's network through a third-party vendor, using stolen credentials.
- Data Compromised: Approximately 40 million credit and debit card accounts and personal information of 70 million customers were affected.
- Financial Impact: The breach cost Target an estimated $162 million in legal fees, settlements, and security upgrades.
- Reputation Damage: Target's reputation suffered, leading to a decline in customer trust and sales.
- Regulatory Response: Target paid $18.5 million in a settlement with 47 states and implemented stricter security measures.

Lessons Learned

- Strengthen Cybersecurity: Companies must ensure robust security measures, especially with third-party vendors.
- Employee Training: Regular training on recognizing cyber threats is essential.
- Incident Response Plan: A clear plan for responding to breaches can mitigate damage.
- Regular Audits: Conducting security audits helps identify and fix vulnerabilities.

Assignment

Research two additional data breaches. For each breach, include:

i. Background: Company and context
ii. Date of Breach
iii. Data Compromised
iv. Financial Losses
v. Reasons for the Breach
vi. Remedies Implemented

This exercise will help you understand the importance of cybersecurity in today's digital world.

1.1.5 Device/System Accessibility

Accessibility refers to the design of devices, systems, and services in a way that they can be used by people with a wide range of abilities and disabilities. It involves making technology usable by everyone, including those with visual, auditory, physical, cognitive, or other types of impairments.

Key Aspects of Accessibility:

The following are the key aspects of Accessibility.

- **Inclusivity:** Ensuring that technology is accessible to the widest possible audience.
- **Adaptability:** Allowing users to customize the way they interact with the system to suit their needs.
- **Compliance:** Adhering to accessibility standards and legal requirements to ensure that products are accessible.

Principles of Accessibility:

Accessibility is a fundamental aspect of design that ensures technology can be used by everyone, including individuals with disabilities. The principles of accessibility guide the creation of systems and interfaces that accommodate a wide range of needs. By adhering to these principles, designers can create products that are inclusive and usable by the broadest possible audience.

1. Perceivability

Perceivability means that all users must be able to perceive the information and user interface components, regardless of their sensory abilities. This principle ensures that content is accessible to users with various sensory impairments, such as those who are blind, deaf, or colorblind.

Examples1: Websites should provide text alternatives (alt text) for images. This allows screen readers to describe the content of the images to users who are visually impaired. For example an online store includes alt text for product images so that a blind user can hear a description of the item, such as "Red wool sweater with V-neck and ribbed cuffs," through their screen reader.

Examples2: Captions and Transcripts for Audio/Video: Providing captions for videos and transcripts for audio content ensures that users who are deaf or hard of hearing can access the information. For example a YouTube tutorial video on cooking includes captions so that users who are hearing impaired can follow along with the instructions. As shown in Fig.1.19.

Fig.1.19: Transcripts for

2. Operability

Operability refers to the ability of users to navigate and use the interface, regardless of their physical or motor abilities. This principle focuses on ensuring that all users, including those with disabilities, can interact with the system effectively.

Example1: Keyboard Navigation: Users should be able to navigate through the interface using

just a keyboard, without needing a mouse.


Fig.1.20: Voice Commands

Example2: Voice Commands: Implementing voice control features allows users to operate systems hands-free. For example a smart home system can be controlled via voice commands, enabling users with mobility impairments to turn on lights or adjust the thermostat without needing to physically interact with the controls. As shown in Fig.1.20.

3. Understandability

Understandability means that the design should make it easy for all users to comprehend the information and know how to operate the interface. This principle focuses on creating content and interactions that are clear and straightforward.

Fig.1.21: (www.fbr.gov.pk)

Example1: Simple Language: A government website (www.fbr.gov.pk) provides instructions for filing taxes in simple language, making the process easier to understand for all tax payers, including those with learning disabilities, as shown in Fig.1.21.

Example2: Error Prevention and Correction: An online form for registering an event highlights required fields and provides real-time feedback if a user forgets to fill out a section, along with instructions on how to correct the mistake.

4. Robustness

Robustness refers to the ability of content to be interpreted by a wide range of devices, including assistive technologies. This principle ensures that content remains accessible even as technologies evolve.

Fig.1.22: Cross-Browser Compatibility

Example1: Cross-Browser Compatibility: Ensuring that a website functions correctly across different web browsers is crucial for accessibility. For example an educational platform tests its website to ensure it works well on Chrome, Firefox, Safari, and Edge(As shown in Fig.1.22), so that students using different browsers have the same experience.

Example2: Support for Multiple Input Devices: A computer device is designed with multiple input options to cater to users with different preferences and needs. For example users can interact with the computer using voice commands for those who find it difficult to use a traditional mouse and keyboard. The touchscreen interface allows for gestures and touch interactions, providing an alternative for users who have difficulty with physical input devices.

1.1.6 Types of Disabilities and Accessibility Solutions

Designing technology that is inclusive and accessible to individuals with various disabilities involves understanding the unique challenges each type of disability presents and implementing tailored solutions.

The following are some accessibility solutions for different disabilities.

Visual Impairments:

Visual impairments encompass conditions such as blindness, low vision, and color blindness. Accessibility solutions for visual impairments aim to make content perceivable through alternative means.

Solutions and Examples:

Screen Readers: These tools convert text and other visual elements into spoken words or braille. Effective implementation requires semantic HTML and ARIA (Accessible Rich Internet Applications) roles. For example **JAWS** (Job Access with Speech) or **NVDA** (Non Visual Desktop Access) reads aloud content on a web page, such as "Welcome to our website. Today's date is September 1, 2024." This enables blind users to navigate and interact with web content, as shown in Fig.1.23.


Fig.1.23: Screen Readers

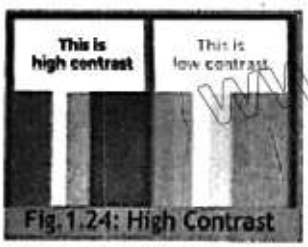


Fig.1.24: High Contrast

High Contrast Modes: Enhances readability by using contrasting colors to make text stand out against the background. For example Windows and macOS offer high contrast themes that switch text and background colors to improve visibility. Websites like BBC also provide high contrast modes for users with low vision. (Fig.1.24)

Zoom Functions: Allows users to magnify text and images on the screen to better accommodate low vision. For example Google Chrome and Mozilla Firefox browsers have built-in zoom features that enable users to increase the size of text and images, making them easier to read, as shown in Fig.1.25.

Fig.1.25: Zoom Functions

Hearing Impairments:

Hearing impairments can range from partial hearing loss to complete deafness. Accessibility solutions for hearing impairments focus on providing information through visual and textual means.

Solutions and Examples:

Closed Captions: Text that appears on screen to transcribe spoken dialogue, sound effects, and other audio cues in videos. For example YouTube provides closed captions for videos, enabling users who are deaf or hard of hearing to read the dialogue and sound effects as they watch, as shown in Fig.1.26.

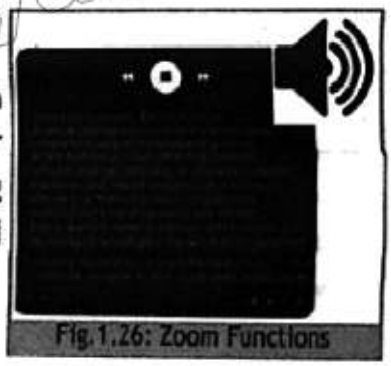


Fig.1.26: Zoom Functions

Transcripts: Written versions of audio content, which allow users to read the content instead of listening. For example TED Talks provides transcripts for each talk, allowing users to read the content in addition to or instead of watching the video.

Sign Language Interpretation: Includes sign language videos that interpret spoken content into sign language. For example Microsoft Teams and other video conferencing tools allow users to include sign language interpreters in video calls. Similarly, news channels like PTV News enhance accessibility by adding sign language interpreters through a side video, making news broadcasts understandable for deaf viewers, as shown in Fig. 1.27.

Fig.1.27: Sign Language Interpretation

Motor Impairments:

Motor impairments affect a person's ability to use traditional input devices like a mouse or keyboard. Accessibility solutions focus on providing alternative methods for interacting with devices.

Solutions and Examples:

Voice Recognition Software: Enables users to control their device and dictate text using voice commands. For example Dragon NaturallySpeaking allows users to navigate their computer, compose documents, and execute commands through voice recognition.

Alternative Input Devices: Includes specialized devices like adaptive keyboards, eye-tracking systems, and adaptive mouse.

On-Screen Keyboards: Virtual keyboards displayed on the screen that can be used with touchscreens or other input methods. For example On-Screen Keyboard provides a virtual keyboard that users can control using a mouse, touch screen, or other adaptive devices, as shown in Fig.1.28.

Fig.1.28: On-Screen Keyboards

1.2 Human-Computer Interaction (HCI)

Human-Computer Interaction (HCI) is a multidisciplinary field that studies how people interact with computers, digital devices, and software applications. The goal of HCI is to improve the interaction between users and systems by making these systems more user-friendly, efficient, and accessible. Good HCI can reduce user frustration, increase productivity, and ensure that technology meets the needs of its users.

Example: The design of a smartphone's touchscreen interface involves HCI principles to ensure that icons are large enough to be tapped easily, and that gestures (like swiping or pinching) are in-built.

1.2.1 Elements of Human-Computer Interaction

Human-Computer Interaction (HCI) is a complex and dynamic field that focuses on the design, evaluation, and implementation of interactive systems for human use. The effectiveness of any interactive system depends on several key elements that must be carefully considered during the design process. These elements include:

- Users
- Tasks
- Interface
- Environment

Users:

Users are the individuals who interact with a system. This involves studying their needs, abilities, limitations, and preferences. Users can vary widely in their skills, knowledge, experience, and expectations. Designers must consider the diversity of users, including factors such as age, cultural background, physical abilities, and technological proficiency.

Example: Designing a word processor for a professional writer involves understanding their need for robust editing tools, an organized workspace, and features that enhance their writing process, such as grammar checking, advanced formatting options, and distraction-free modes.

Tasks:

Tasks refer to the specific activities that users need to perform using the system. Identifying and understanding these tasks is essential for designing an interface that supports users in achieving their goals efficiently and effectively.

Example: On an e-commerce website, key tasks might include searching for products, filtering results, reading reviews, adding items to a shopping cart, and completing a purchase. Designers must ensure that each of these tasks is straightforward and easy to perform. As shown in Fig. 1.29.

Fig.1.29 Shopping Cart

Interface:

The Interface is the point of interaction between the user and the system. It is where the design of the system comes to life, allowing users to interact with it through various input and output mechanisms. The interface can take many forms, including graphical user interfaces (GUIs), voice command systems, gesture-based interfaces, and more.

Example: The layout of a video editing software's interface is crucial for user productivity. Tools like cut, paste, and trim should be easily accessible, and the timeline should be carefully designed to allow for precise edits. A well-designed interface might also include customizable workspaces, where users can arrange tools and panels according to their workflow preferences. As shown in Fig.1.30.

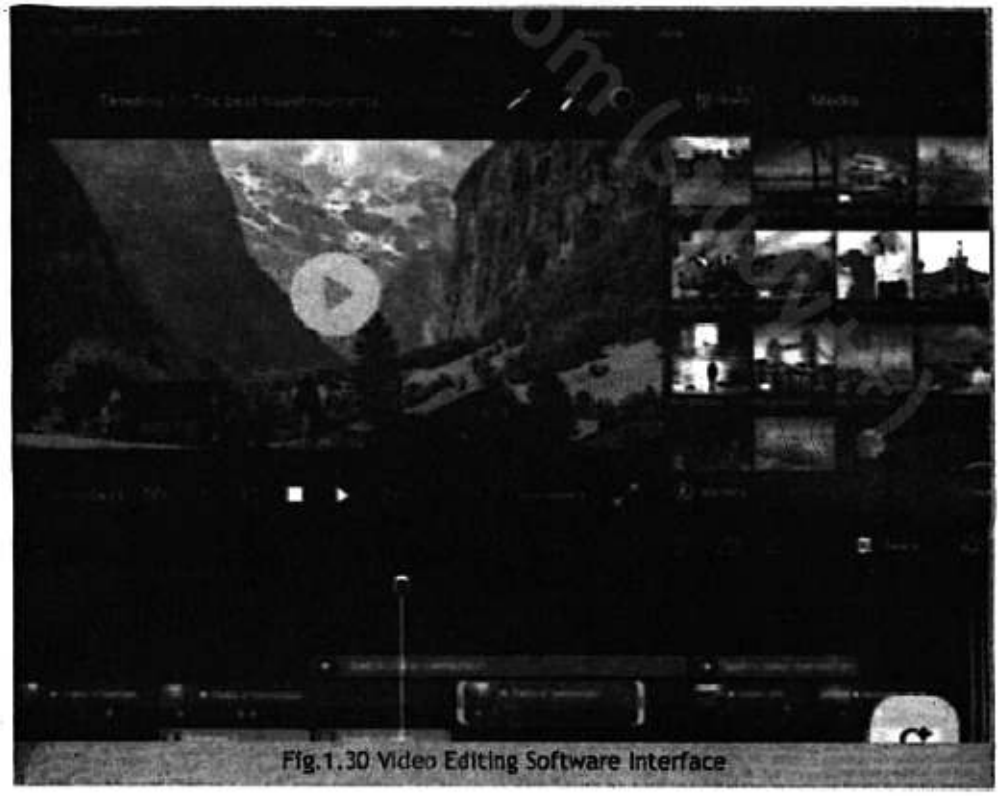


Fig.1.30 Video Editing Software Interface

Environment:

The Environment refers to the physical and social context in which the interaction between the user and the system takes place. The environment can significantly influence how a system is used, and designers must consider these factors when creating interfaces. The environment includes the physical surroundings (e.g., lighting, noise levels, and available space) and the social context (e.g., whether the system will be used individually or collaboratively).

Fig.1.31: A Public Kiosk Interface

Example: A public kiosk interface, such as those found in airports or shopping malls, must be designed for quick, efficient use in potentially noisy, crowded environments. The interface should be simple, with large buttons and clear instructions to accommodate users who may be in a hurry or distracted by their surroundings. Additionally, the kiosk might need to be designed for use by standing users, meaning the screen should be at a comfortable height and angle. As shown in Fig. 1.31.

1.2.2 Usability in Human-Computer Interaction

Usability is a fundamental concept in Human-Computer Interaction (HCI), and it plays a crucial role in determining how effective and enjoyable a system is for its users. Usability refers to the extent to which a system can be used by specific users to achieve specified goals with effectiveness. A system with high usability is one that is easy to use and learn.

Key Components of usability in HCI:

Usability can be broken down into several key components, each of which contributes to the overall user experience in HCI. These components include Learnability, Efficiency, Memorability, Error Handling, and Satisfaction.

Learnability:

Learnability refers to how easy it is for new users to accomplish basic tasks the first time they encounter the system. A highly learnable system allows users to quickly grasp how to use it without needing extensive guidance or prior experience. This is especially important for applications or devices intended for a broad audience, where users may have varying levels of technical expertise.

Fig.1.32: Online Mobile Banking App

Example: Consider a mobile banking app designed for a diverse user base. A new user should be able to quickly learn how to navigate the app to check their account balance, transfer funds, or view transaction history. The use of onboarding tutorials or interactive guides can further enhance learnability, ensuring that users feel confident and capable from their first interaction as shown in Fig.1.32.

Efficiency:

Efficiency pertains to how quickly and effectively users can perform tasks once they have learned how to interact with the system. An efficient system allows users to achieve their goals with minimal time and effort.

Example: A professional graphic designer using design software like Adobe Photoshop needs to perform tasks such as editing images, creating layouts, and applying filters swiftly. Efficiency in this context means providing shortcut keys, customizable toolbars, and responsive tools that allow the designer to work with speed and precision.

Memorability:

Memorability refers to how easily users can remember how to use a system after a period of not using it. A system with high memorability ensures that users can return to it after some time and still remember how to perform key tasks without needing to relearn the interface.

Example: Imagine a user who only occasionally uses an online form submission system, such as a tax filing website. After several months, when they return to the site, they should be able to easily recall how to navigate the system, fill out the necessary forms, and submit their information. This can be facilitated through consistent design patterns, clear labeling, and a logical flow of steps.

Error Handling:

Error Handling involves the system's ability to prevent errors, guide users in correcting them, and help users recover from mistakes. Good error handling is essential for maintaining user trust and ensuring a smooth interaction, even when things go wrong.

Fig.1.33: Online booking system

Example: An online booking system (Fig.1.33) for airline tickets should prompt users to correct mistakes, such as entering an invalid date or leaving a required field blank. Instead of allowing the user to proceed with faulty data, the system might highlight the problematic fields in red and provide an error message explaining what needs to be corrected. Additionally, providing helpful suggestions or autofill options can prevent common errors, such as typos in names or addresses.

Useful Links!

Explore HCI Concepts: " Interaction Design Foundation offers free courses on HCI and usability principles."

Accessibility Tools: "Check out WebAIM for resources on making web content accessible to people with disabilities."

1.2.3 Common Problems in Human-Computer Interaction

Human-Computer Interaction (HCI) is a critical field that focuses on improving the interaction between users and computer systems. While significant progress has been made in designing

more user-friendly and efficient systems, several common problems still persist. These issues can negatively impact user experience, leading to frustration, errors, and even exclusion of certain user groups. Addressing these challenges is essential for creating systems that are accessible, user-friendly, and efficient for all users.

Interface Issues:

Poorly designed interfaces are one of the most common problems in HCI. The interface is the primary means through which users interact with a system, and its design can make or break the user experience. Common issues with interfaces include untidy layouts that could lead users to unclear navigation paths that make it difficult to locate important features. These problems can lead to confusion, inefficiency, and increased errors.

Example: Consider an e-commerce website with a poorly designed navigation menu. If the menu is untidy with too many options, users may struggle to find the categories they are looking for. For instance, a user searching for electronics might have to examine through an extensive list of unrelated categories, leading to frustration.

Accessibility Challenges:

Accessibility in HCI refers to the design of systems that can be used by people with a wide range of abilities and disabilities. However, many systems are not designed with accessibility in mind, leading to challenges for users with visual, auditory, motor, or cognitive impairments. These challenges often arise because designers do not fully consider the needs of all potential users.

Example: A website that lacks alternative text (Alt text feature) for images is inaccessible to users with visual impairments who rely on screen readers. These users might miss out on important content or context provided by images, making their experience incomplete or frustrating. Similarly, a mobile app that relies solely on sound for notifications may be unusable by individuals with hearing impairments.

Performance Issues:

Performance issues, such as slow or unresponsive systems, can severely impact the usability of a system. Users expect systems to respond quickly to their inputs, and any delay can lead to frustration and a decrease in productivity. Performance problems can be caused by a variety of factors, including poor system optimization, excessive resource use, or network latency. When systems are slow or fail to load content efficiently, users may abandon tasks or look for alternative solutions.

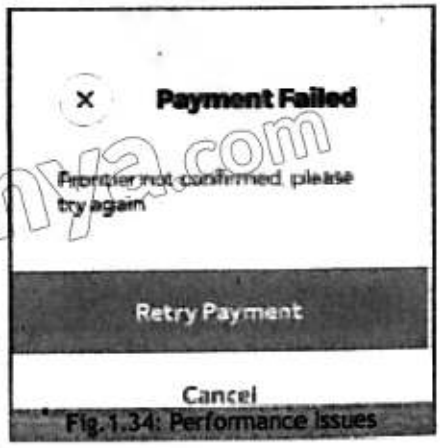


Fig.1.34: Performance Issues

Example: Imagine using an online banking app that takes an excessive amount of time to load account information or process transactions (Fig.1.34). Users may become impatient and concerned about the security of their transactions, leading them to avoid using the app in the future.

Poor Error Handling and Feedback:

Effective error handling and feedback are crucial components of a user-friendly system. When users encounter errors, they need clear, constructive feedback to understand what went wrong and how to fix it. Systems that provide poor or inadequate error messages can leave users confused and frustrated, leading to mistakes and inefficiency. Conversely, well-designed error handling helps users recover from mistakes quickly and with confidence, improving the overall user experience.

Example: Consider a form submission process on a website where the user is required to enter several fields of information. If the user leaves a required field blank or enters data in the wrong format, the system should provide an immediate, clear error message that explains what needs to be corrected. For instance, if an email field is left empty or filled incorrectly, the system should display a message like, **"Please enter a valid email address."** Without this guidance, users might not understand why their submission is not being accepted, leading to frustration and potentially abandoning the form altogether.

1.2.4. Methods for Improving Human-Computer Interaction

Improving Human-Computer Interaction (HCI) involves various strategies and methodologies aimed at creating systems that are more user-friendly, efficient, and complete. By addressing key aspects such as **user-centered design, accessibility, interface simplification, performance optimization, and error prevention,** developers can significantly enhance the users' experience.

User-Centered Design (UCD):

User-Centered Design (UCD) is a design philosophy that places the end users at the core of the design process. It is an iterative approach where user feedback is continuously gathered and integrated throughout the development cycle. This method ensures that the final product aligns closely with the users' needs, preferences, and limitations, resulting in a more effective and satisfying user experience.

Fig.1.35: User-Centered Design (UCD)

Example: Consider the development of a new educational software intended for use in schools. By involving teachers and students in the design process from the beginning, developers can gather valuable feedback on features such as interactive elements, user interfaces, and content organization. Early testing might reveal that students prefer interactive quizzes over static reading material, leading to adjustments that enhance engagement and learning outcomes. As shown in Fig.1.35.

Accessibility Enhancements:

Accessibility enhancements focus on making systems usable for individuals with a range of disabilities. This involves designing systems that are perceivable, practicable, understandable, and user-friendly, ensuring that people with,visual, auditory, motor, or cognitive impairments can effectively interact with the system.

Example: An educational platform that includes captions and transcripts for all video content

improves accessibility for students with hearing impairments. Additionally, incorporating keyboard navigation and screen reader compatibility ensures that users with motor impairments or visual disabilities can interact with the platform effectively.

Simplifying User Interfaces:

Simplifying user interfaces involves reducing unnecessary complexity and presenting information in a clear, concise manner. A well-designed interface should prioritize essential features and streamline interactions to minimize user effort and cognitive load.

Example: A streamlined email application that features a minimalist design with only the most commonly used tools (such as compose, inbox, and search) can significantly enhance user efficiency. By avoiding complex menus and excessive options, users can quickly access and manage their emails without unnecessary distractions.

Improving Performance

Optimizing system performance is essential for ensuring that applications run smoothly and respond promptly to user inputs. Performance issues such as slow load times or unresponsive interfaces can detract from the user experience and hinder productivity.

Example: Reducing the load time of a large e-commerce website by optimizing images, compressing files, and implementing efficient coding practices can greatly improve user experience.

Error Prevention and Recovery:

Effective error prevention and recovery mechanisms are crucial for minimizing the impact of user mistakes and system errors. Systems should be designed to prevent errors whenever possible and provide clear feedback and recovery options when errors do occur.

Example: A spreadsheet application that prompts users to save changes before closing the file helps prevent data loss. Additionally, an online form with real-time validation checks for required fields and provides immediate feedback when errors are detected can help users correct mistakes before submission.

1.2.5 Ethical, Social, Economic, and Environmental Implications in HCI

Human-Computer Interaction (HCI) is not just about creating systems that are user-friendly and efficient; it also encompasses broader considerations that affect society at large. Ethical, social, economic, and environmental implications play a crucial role in the development and deployment of any new system. Understanding these aspects is essential for designing systems that are not only effective but also responsible and sustainable.

Ethical Implications:

Ethical implications in HCI address how technology affects users' rights, privacy, and dignity. Ethical design ensures that technology is developed and used in ways that are respectful and fair, protecting users from harm and misuse.

- **Data Privacy:** One of the major ethical concerns is how user data is collected, stored, and

used. Developers must ensure that user data is handled with confidentiality and used only for its intended purpose. Users should have control over their data and be informed about how it is used.

Fig.1.36: Social media platforms

Example: Social media platforms such as, Instagram, Facebook, WhatsApp and Twitter (Fig.1.36), collect vast amounts of personal data from users. Ethical practices require these platforms to implement strong data protection measures, obtain explicit user consent before collecting data, and provide clear options for users to manage their privacy settings.

- **User Consent:** Users should be informed and provide consent before engaging with systems that collect or use their data. Consent should be clear and informed.

Example: Mobile apps that request access to personal information, such as location or contacts, should provide users with a clear explanation of why this access is necessary and obtain explicit consent before proceeding.

Social Implications:

Social implications refer to how technology influences communication, social interactions, and access to information. HCI can have both positive and negative effects on social structures and relationships.

For Your Info!

Social Media and Mental Health: "Research has shown that excessive use of social media can lead to anxiety and depression, particularly among young people. Balanced use is a key."

Example: Smartphones and social media enable instant messaging and video calls, fostering global communication. However, excessive screen time and reliance on digital interactions can affect face-to-face communication skills and lead to issues such as social isolation.

Economic Implications:

Technology impacts the economy by creating new opportunities and challenges. Understanding these implications helps in designing systems that support economic growth while addressing potential disruptions.

Example1: The rise of e-commerce platforms like Amazon (Fig.1.37) has generated jobs in tech, logistics, and customer service. These platforms have also led to the growth of related industries, such as digital marketing and online payment systems.

Fig.1.37: Amazon

Example2: The automation of manufacturing processes through robotics (Fig.1.38) can lead to increased production efficiency but may also result in job losses for assembly line workers. Training and reskilling programs can help mitigate these effects.


Fig.1.38: Manufacturing through robotic

Environmental Implications:

The environmental impact of technology involves considering energy consumption, resource use, and waste management. Sustainable practices in HCI are crucial for minimizing the ecological footprint of technology.

Example1: Companies like Google and Microsoft are investing in renewable energy to power their data centers, aiming to reduce their carbon footprint and promote sustainability.

Example2: Electronics manufacturers are implementing take-back programs and designing products with recyclable materials to reduce e-waste. Initiatives such as Apple's recycling program help ensure that old devices are properly recycled.

1.3 Trade-offs between Usability and Security in Computing Systems

When designing computing systems, achieving a balance between usability and security is crucial. Both factors are essential, but they often come into conflict. The following is a detailed look at the trade-offs between usability and security, along with recommendations for cybersecurity measures considering efficiency, cost, privacy, and ethics.

1.3.1 Trade-offs between Usability and Security

Usability focuses on how easy and intuitive a system is for users, while security aims to protect the system from unauthorized access and attacks. The trade-offs arise because enhancing security can sometimes degrade usability, and vice versa.

The following table represents the trade-offs between usability and security in computing systems, along with examples.

Aspect	Security	Usability	Example
Authentication Complexity	Enhances security by adding extra verification steps like multi-factor authentication (MFA), making unauthorized access much harder.	Reduces usability as MFA requires additional steps like entering codes from a phone, slowing down user access and causing potential frustration.	A user logging into a bank account needs to enter a password and then a code sent to their phone, increasing security but delaying access.
Access Controls	Protects sensitive data by restricting user permissions, ensuring only authorized individuals can access or modify information.	Hinders usability when users frequently encounter permission issues, requiring them to request access, which can slow down their work.	An employee needs frequent permission from IT to access certain files, delaying their ability to complete tasks efficiently.

Aspect	Security	Usability	Example
Encryption	Secures data by converting it into a format only accessible with a decryption key, protecting against unauthorized access even if data is intercepted.	Impacts usability as encryption can slow down data access and processing, and managing encryption keys can be complex and time-consuming.	Encrypting patient records keeps them secure but may slow down access, which can be challenging during emergencies.
Regular Updates and Patches	Maintains security by fixing vulnerabilities and protecting against new threats through regular updates.	Disrupts usability as updates may require system restarts or interrupt work, and they can sometimes introduce compatibility issues.	Frequent software updates may interrupt employees' work, requiring them to restart their systems, which can be inconvenient during busy times.

1.3.2 Recommendations for Cybersecurity measures

When designing and implementing cybersecurity measures, it is important to consider various factors to strike a balance between security and user experience. The key factors to consider are efficiency, cost, privacy, and ethics. Each of these plays a crucial role in determining how effective and sustainable the security solutions will be.

Efficiency: Efficiency in cybersecurity involves implementing solutions that protect systems without significantly hindering user productivity. The following computing solutions illustrate the principle of efficiency in cybersecurity measures.

- **Auto-Save and Backup Systems:**

Automatically saving work and backing up files regularly helps ensure that data is not lost in case of a system crash or cyberattack. This increases efficiency by reducing the risk of lost work and the need to redo tasks.

Example: Google Docs automatically saves documents in real-time to Google Drive, so users don't have to worry about losing their work if their computer shuts down unexpectedly.

- **Password Managers:**

Password manager stores and autofill strong, unique passwords for different websites, reducing the time and effort users spend managing multiple passwords while enhancing security.

Example: Using a password manager like LastPass or Bitwarden, users can automatically log into websites without needing to remember complex passwords, making the login process quicker and more secure.

Cost: Cost considerations are critical when determining the scope and scale of cybersecurity measures. The goal is to implement effective security within budget constraints. The following computing solutions illustrate the principle of cost in cybersecurity measures.

Helping Tip!

Choosing Security Tools: Consider open-source security tools as a cost-effective alternative to proprietary software, especially for small businesses or personal use.

Freemium Security Software:

Many security software providers offer free basic versions of their products, which can be sufficient for small businesses or individual users, allowing them to secure their systems without incurring high costs.

Example: Avast offers a free version of its antivirus software that provides essential protection against viruses and malware, which is often adequate for home users.

Cloud-Based Security Services:

Cloud-based security solutions often have lower upfront costs compared to on-premises hardware, as they eliminate the need for expensive infrastructure and maintenance.

Example: A small business might choose to use a cloud-based firewall service, such as Cloudflare, to protect their website from attacks, avoiding the cost of purchasing and maintaining their own hardware.

Open Source Solutions:

Open source tools and frameworks offer cost-effective alternatives to expensive proprietary software, providing robust security features without significant financial investment.

Example: The OWASP (Open Web Application Security Project) provides a variety of open-source tools that help developers identify and mitigate security vulnerabilities in web applications, reducing the need for expensive commercial solutions.

Privacy: Privacy protection is a foundation of cybersecurity, ensuring that users' personal data is handled responsibly and securely. The following computing solutions illustrate the principle of privacy in cybersecurity measures.

Private Browsing Modes:

Private browsing, or "incognito" mode, in web browsers prevents the storage of browsing history, cookies, and other data, helping protect user privacy on shared or public computers.

Example: When using a public computer at a library, users can enable incognito mode in Chrome or Firefox to ensure their browsing activity isn't stored, protecting their privacy.

Anonymization and Encryption:

These techniques protect user privacy by ensuring that even if data is accessed by unauthorized parties, it cannot be traced back to individuals or easily deciphered.

Example: Healthcare providers often encrypt patient records and anonymize data before sharing it for research purposes, ensuring that personal health information remains confidential.

Two-Factor Authentication (2FA):

2FA adds an extra layer of security by requiring users to provide two forms of identification (such as a password and a code sent to their phone), which helps protect their accounts from unauthorized access. As shown in Fig. 1.39.

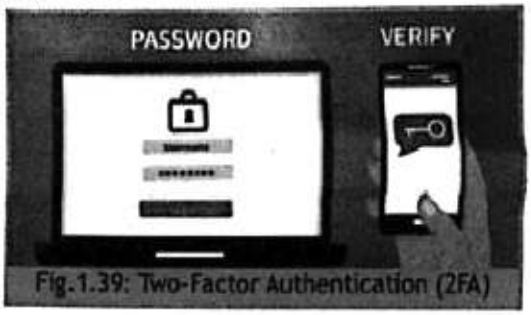


Fig.1.39: Two-Factor Authentication (2FA)

Example: When logging into their email, a user might enter their password and then receive a text message with a one-time code, ensuring that only they can access their account even if someone else knows their password.

Ethics: Ethical considerations in cybersecurity involve respecting user rights and maintaining trust through transparent and fair practices. The following computing solutions illustrate the principle of ethics in cybersecurity measures.

- **User-Friendly Privacy Settings:**

Designing privacy settings that are easy to understand and adjust ensures that all users, regardless of their technical knowledge, can control how their data is used.

Example: Facebook provides a simple privacy checkup tool that guides users through their privacy settings step by step, making it easy to adjust who can see their posts and personal information.

- **Accessibility Features for Disabled Users:**

Ensuring that security measures are accessible to users with disabilities is an ethical responsibility, helping to prevent exclusion from digital services.

Example: An online banking platform might offer voice recognition for account access, making it easier for users with physical disabilities to log in securely without needing to type a password as shown in Fig.1.40.

Fig.1.40: Online Banking-Voice Recognition

Useful Links

Cybersecurity and Usability Research: Usable Security and Privacy Group at Carnegie Mellon University explores ways to design systems that are both secure and user-friendly.

Summary

- **Usability:** Measures how effortlessly and effectively users can interact with a system.
- **Moderated Usability Testing:** Involves a facilitator guiding and observing users.
- **Unmoderated Usability Testing:** Users complete tasks independently, often remotely.
- **Qualitative Usability Testing:** Focuses on user experiences and feedback.
- **Quantitative Usability Testing:** Measures usability with numerical data.
- **Remote Usability Testing:** Testing conducted from the user's location.
- **A/B Testing:** Compares two versions of a system to evaluate performance.
- **Complex Interfaces:** Overwhelming users with too much information or complexity.
- **Inconsistent Navigation:** Variability in navigation elements across the system.
- **Poor Error Messages:** Unclear or unhelpful error messages.
- **Lack of Accessibility:** Failure to accommodate users with disabilities.
- **Security:** Measures to protect information and systems from unauthorized access and threats.
- **Malware:** Malicious software designed to harm systems.
- **Phishing:** Deceptive attempts to obtain sensitive information.
- **Man-in-the-Middle (MitM) Attacks:** Interception and potential alteration of communication.
- **Denial of Service (DoS) Attacks:** Overloading a system to make it unavailable.
- **Insider Threats:** Security risks from within an organization.
- **Accessibility:** Design to ensure usability by people with a range of abilities and disabilities.
- **Visual Impairments:** Solutions like screen readers, high contrast modes, and zoom functions.
- **Hearing Impairments:** Solutions like closed captions, transcripts, and sign language interpretation.
- **Motor Impairments:** Solutions like voice recognition software, alternative input devices, and on-screen keyboards.
- **Human-Computer Interaction (HCI):** A multidisciplinary field studying interactions between people and computers to make systems more user-friendly, efficient, and accessible.
- **Users:** Individuals interacting with a system; their needs, abilities, limitations, and preferences must be considered.

- **Tasks:** Specific activities users perform using the system.
- **Interface:** The point of interaction between the user and the system, including GUIs, voice commands, and gesture-based controls.
- **Environment:** The physical and social context of the interaction, such as lighting, noise, and whether the use is solitary or collaborative.
- **Interface Issues:** Problems with poorly designed interfaces, such as untidy layouts and unclear navigation.
- **Accessibility Challenges:** Issues in making systems usable for people with disabilities, including visual, auditory, motor, or cognitive impairments.
- **Performance Issues:** Problems such as slow or unresponsive systems affecting usability.
- **Poor Error Handling and Feedback:** Inadequate error messages and feedback leading to user frustration.
- **User-Centered Design (UCD):** A design approach focusing on users' needs and feedback throughout the development cycle.
- **Accessibility Enhancements:** Designing systems to be usable by people with various disabilities.
- **Simplifying User Interfaces:** Reducing complexity and presenting information clearly.
- **Improving Performance:** Optimizing system performance to ensure smooth operation.
- **Error Prevention and Recovery:** Mechanisms to prevent and recover from errors effectively.
- **Ethical Implications:** Concerns about user rights, privacy, and dignity.
- **Social Implications:** The impact of technology on communication and social interactions.
- **Economic Implications:** The effect of technology on the economy, including job creation and disruption.
- **Environmental Implications:** Considering energy consumption, resource use, and waste management in technology.
- **Authentication Complexity:** Balancing security measures like multi-factor authentication with user convenience.
- **Access Controls:** Restricting data access to protect security while considering the impact on user workflow.
- **Encryption:** Protecting data with encryption while managing the potential impact on performance.
- **Regular Updates and Patches:** Maintaining system security through updates while managing disruptions to usability.

Exercise

Select the best answer for the following Multiple-Choice Questions (MCQs).

1. A software product receives negative user reviews due to being difficult to use. Which of the following factors is least likely responsible?
   a. Inefficiency b. Ineffectiveness c. High satisfaction d. High complexity
2. A web form alerts the user when an email address is entered incorrectly before submission. Which usability principle does this best represent?
   a. User satisfaction b. Error prevention
   c. Performance optimization d. Feedback mechanism
3. If you want to observe users' emotional reactions and gather verbal feedback while they perform tasks, which usability testing method would be most suitable?
   a. Moderated Usability Testing b. A/B Testing
   c. Unmoderated Usability Testing d. Remote Usability Testing
4. Which practice best supports confidentiality in a hospital's patient records system?
   a. Ensuring the system is always online
   b. Allowing only authorized doctors to access patient data
   c. Letting patients edit their own files freely
   d. Encrypting data for public access
5. A user receives a seemingly legitimate email from their bank requesting login details. This is an example of:
   a. Phishing b. Malware c. Spyware d. Spoofing
6. A company wants to protect client data during transmission over the internet. Which security method should they apply?
   a) Firewall b. Encryption c. Access Control d. Antivirus Software
7. What is the main objective of Human-Computer Interaction (HCI)?
   a. System performance b. User interaction
   c. Software complexity d. Hardware costs
8. Which of the following is NOT considered a key element of HCI?
   a. Users b. Tasks c. Budget d. Interface
9. Which of the following is a common issue related to accessibility in HCI?
   a. Slow load times b. Inadequate error messages
   c. Lack of alternative text for images d. Complicated authentication processes
10. Which method involves placing the end users at the core of the design process in HCI?
    a. Performance Optimization b. Error Prevention
    c. User-Centered Design (UCD) d. Simplifying User Interfaces
11. What does the term 'interface' refer to in HCI?
    a. The physical device used b. Point where user interacts with system
    c. The software programming language d. The hardware configuration

12. Why is multi-factor authentication (MFA) often seen as a trade-off between security and usability?
    a. It's expensive to implement
    b. It improves system speed
    c. It increases security but can slow down user access
    d. It reduces the need for encryption
13. Which issue is caused by encryption?
    a. Processing speed b. Data access c. Key management d. Interface
14. How can auto-save improve cybersecurity efficiency?
    a. Recovery b. Encryption c. Complexity d. Management
15. Which ethical consideration involves transparency in data use?
    a. Efficiency b. Cost c. Privacy d. Accessibility

Give short answers to the following Short Response Questions (SRQs).

1. What are the key components of usability in HCI?
2. List and briefly describe the four key elements of HCI.
3. What is the difference between moderated and unmoderated usability testing?
4. What does 'error handling' mean in the context of HCI?
5. What is meant by 'User-Centered Design (UCD)' in HCI?
6. Give three benefits and two drawbacks of using password managers?
7. How does user feedback contribute to effective system design?
8. Provide an example of how user feedback led to a design change.
9. Give two considerations for designing accessible systems for users with disabilities.
10. Suggest three features that should be included in a new educational software application to enhance accessibility for users with disabilities.

Give long answers to the following Extended Response Questions (ERQs).

1. Describe how accessibility challenges impact the usability of a digital system and provide examples of improvements that can address these challenges.
2. What are the main elements involved in Human-Computer Interaction (HCI) design? Explain each element's role in creating effective interactive systems.
3. Explain the ethical considerations in Human-Computer Interaction (HCI) with examples. Why are these considerations important in the design and deployment of computing systems?
4. You are developing a new software tool for remote team collaboration. What usability and security features would you integrate to support efficient teamwork while protecting sensitive data?
5. You are tasked with designing an e-commerce website. The site must be both user-friendly and secure. List three features you would include to achieve this balance and explain why.
6. Discuss the trade-offs between usability and security in computing systems. Provide examples of how these trade-offs might manifest in real-world applications.

7. You are tasked with designing an e-commerce website. The site must be both user-friendly and secure. List *three features* you would include to achieve this balance and explain why.
8. How do, security measures like multi-factor authentication (MFA) and encryption impact usability, and how can designers balance these trade-offs?

Activity 1: Usability and Accessibility Design Challenge

Objective: To understand and apply principles of usability and accessibility in system design.

Group Project: Divide students into small groups and task them with designing a mock interface for a hypothetical app that must be user-friendly and accessible to people with disabilities.

Instructions:

- *Choose an* application (e.g., a fitness tracker, e-learning platform, or social media app).
- Apply usability principles (simplicity, consistency, feedback, error prevention, learnability) to the design.
- Incorporate accessibility features (e.g., voice guidance, large buttons, high-contrast modes).
- Create wireframes or mockups of the interface.
- Present the design to the class and explain how the features support both usability and accessibility.

Activity 2: Security Measures Evaluation

Objective: To analyze the impact of different security measures on system usability and functionality.

Individual Research and Presentation: Students research and present on different security measures and their potential impacts on usability.

Instructions:

- Choose one security measure (e.g., multi-factor authentication, encryption, or firewalls).
- Explain how this measure works and its purpose.
- Discuss how it might affect the usability of a system (e.g., increased steps for user login, potential for user confusion).
- Provide real-world examples of systems where this trade-off is evident.
- Present findings to the class and suggest ways to mitigate usability issues while maintaining strong security.

Activity 3: Design an Inclusive Interface

Objective: Students will design a user interface for a specific target audience considering different types of disabilities.

Instructions:

- **Identify User Needs:** Divide students into groups and assign each group a different disability (visual, auditory, motor).
- **Design Brief:** Each group will design an interface for a common application (e.g., a public information kiosk, a health tracking app) that caters to their assigned disability.
- **Design Elements:** Ensure designs include features like screen readers, high contrast modes, alternative input methods, etc.
- **Presentation:** Groups present their designs to the class, explaining how their interface meets accessibility standards.

Activity 4: Usability Testing Simulation

Objective: To practice and understand different usability testing methods.

Simulation Exercise: Conduct a simulated usability test of a given website or app, applying different testing methods.

Instructions:

1. **Preparation:** Select a website or app to test.
2. **Testing Methods:** Use different types of usability testing methods (moderated, unmoderated, qualitative, quantitative) to evaluate the system. For example:
   - **Moderated Test:** Have a classmate use the system while you observe and take notes.
   - **Unmoderated Test:** Have another student perform tasks on their own and record their interactions.
   - **Qualitative Test:** Conduct interviews with users to gather their feedback.
   - **Quantitative Test:** Measure task completion times and error rates.
3. **Analysis:** Compile the results from different tests and identify common usability issues.
4. **Reporting:** Prepare a report summarizing the findings and suggest improvements based on the testing.

Activity 5: Evaluating and Improving an Educational App's Interface

Suggested App: Taleemabad

- Taleemabad is a popular educational app aimed at teaching children in a fun and interactive way. It provides engaging content in subjects like mathematics, English, and Urdu through games and animated lessons.

Short Activity Overview:

Evaluate the **Taleemabad app's interface**, identify areas of improvement, and suggest user-centered changes that could enhance learning and usability.

Questions for Evaluation:

1. Is the app easy to navigate for both children and parents?
2. Is the app's design colorful and engaging for children?
3. Does the design support ease of use, or is it overwhelming?
4. Can children easily start and finish a lesson?
5. Is the app's flow intuitive for young users without adult assistance?

Suggestions for Improvement:

1. What would you suggest to make the app more user-friendly?
2. Propose 1-2 design changes that would improve the app's usability for young learners.