

Evaluating Program Designs

Textbook

Evaluating Program Designs



When using software, we assess its **readability** (how easy its code is for humans to understand) and **usability** (how easy and effective the program is for its users).

Readability

Readable code is crucial for software maintenance, collaboration among programmers, and efficient bug fixing. To evaluate readability, check for clear comments, meaningful variable and function names (e.g., `calculateTotal`), consistent formatting (indentation, spaces), modularity (code broken into specific functions), and overall simplicity. You can evaluate readability in programs from third parties (like open-source calculator apps), peers (classmates' code in group projects), and marketable programs (considering how a new developer would understand a company's codebase).

Usability

Usability focuses on the user experience (UX), ensuring a program is intuitive, efficient, and satisfying. Good usability leads to higher user adoption, task efficiency, and user satisfaction. To evaluate usability, consider learnability (ease for new users), efficiency (speed of task completion), memorability (ease of remembering how to use after a break), error prevention and recovery (how the program handles and recovers from errors), and overall user satisfaction. Examples for evaluation include popular marketable apps (Instagram, Google Maps), peer-created programs, and new online tools.

Efficiency and Effectiveness of Digital Tools or Resources

It's important to understand how **efficient** and **effective** digital tools are for real-world tasks.

Efficiency

Efficiency measures how well a tool performs a task with minimal waste of time, effort, or resources, helping you get things done quickly. Evaluate efficiency by looking at time savings, resource consumption (battery, data), automation capabilities, and processing speed. For instance, a word processor with auto-save and spell check is efficient for writing a research paper compared to manual methods. A search engine is highly efficient for finding information, and collaborative document editing software is efficient for group projects by enabling simultaneous work.

Effectiveness

Effectiveness measures how well a tool achieves its intended purpose or helps you accomplish your goals, ensuring you get the *right* things done. Evaluate effectiveness by assessing if the tool helps successfully complete the task, produces accurate and reliable results, enhances work quality, and genuinely solves the problem. For example, a language learning app is effective for learning a new language, a budgeting app for managing finances, and advanced presentation software for creating professional presentations, as they directly facilitate achieving these specific goals.

Key Takeaway: The best digital tools are both efficient (fast and easy to use) and highly effective (successfully achieve their purpose) for their specific tasks.

Exercise: Program Evaluation Challenge

This exercise will help you practice evaluating the readability and usability of a real program.

Task: Choose one of the following programs (or a similar program approved by your instructor) to evaluate:

1. **A simple open-source game:** (e.g., a basic Tic-Tac-Toe game, a simple platformer, or a classic arcade game clone available on GitHub or a similar platform).
2. **A basic utility program:** (e.g., a simple calculator, a unit converter, or a to-do list application, preferably one with publicly available code).
3. **A new web-based tool or app you haven't used before:** (e.g., a new online image editor, a collaborative brainstorming tool, or a simple data visualization website).

Instructions:

1. **Access the Program:** Obtain access to the program. If it's code, ensure you can view the source code. If it's a web tool or app, simply open it in your browser.
2. **Evaluate Readability (if applicable – for code-based programs):** Spend 15–20 minutes examining the program's code. Consider whether there are clear and helpful comments, or if they are missing or confusing. Assess if variable and function names are descriptive and easy to understand. Observe the formatting to see if the code is consistently indented and well-organized. Determine if the code is modular, meaning it's broken into logical, smaller pieces like functions or classes. Finally, consider if any part of the code seems unnecessarily complex. After your examination, write a short paragraph (5–7 sentences) summarizing your findings on the code's readability, providing specific examples from the code to support your points.
3. **Evaluate Usability:** Spend 15–20 minutes actively using the program as if you were a new user, attempting to complete a few common tasks. Evaluate its learnability: how easy was it to figure out how to use the program without instructions? Assess its efficiency: how quickly could you complete a task once you understood it, and did it require too many steps? Consider error handling: did you encounter any errors, and if so, was the error message clear and was it easy to recover? Finally, reflect on your satisfaction: did you enjoy using the program, and was the interface pleasant and intuitive? After your evaluation, write a short paragraph (5–7 sentences) summarizing your findings on the program's usability, providing specific examples of what worked well or what was challenging.

4. **Overall Critique:** Write a concluding paragraph (3–5 sentences) that offers an overall critique of the program. What are its biggest strengths and weaknesses in terms of both readability (if applicable) and usability? If you were to give advice to the developer, what would be your top suggestion for improvement?

Critical Thinking Questions

- When a software development team works on a major update, why would prioritizing the ease with which new programmers can understand the existing code be just as important as how smoothly the program runs for the end-user?
- Think about two different apps you use regularly: one you find incredibly helpful but sometimes frustrating to use, and another that is a joy to use but doesn't fully meet your needs. Which app do you consider more successful overall, and what does that tell you about the balance between a tool's ability to achieve its purpose and its ease of use?
- Imagine you're designing a new feature for a popular social media platform. What specific design choices related to user experience would you make to help users avoid common mistakes, and how would you ensure that if an error does occur, the user can easily fix it without becoming frustrated?

Questions (5)

1. A new programmer joins a team working on an existing calculator application. They need to add a new function for square roots. According to the passage, which aspect of the existing code would be most important for this new programmer to evaluate?

MULTIPLE CHOICE

Choose the correct answer:

- A. Its readability
- B. Its user satisfaction rating
- C. Its processing speed
- D. Its color scheme

2. A user is trying out a new online image editor. They find it difficult to locate the 'crop' tool, and the icons don't clearly indicate their functions. Which aspect of usability is primarily being challenged in this scenario?

MULTIPLE CHOICE

Choose the correct answer:

- A. Efficiency
- B. Memorability
- C. Error prevention
- D. Learnability

3. You are using a new online unit converter tool. You notice that it takes many clicks to perform a simple conversion, even after you know how to use it. Which aspect of the tool's efficiency is being negatively impacted?

MULTIPLE CHOICE

Choose the correct answer:

- A. Time savings
- B. Resource consumption
- C. Automation capabilities
- D. Processing speed

4. A language learning app helps users successfully achieve fluency in a new language. According to the passage, this app would be considered highly effective. What specifically defines a tool's effectiveness?

MULTIPLE CHOICE

Choose the correct answer:

- A. Its speed of task completion.
- B. How well it performs a task with minimal waste.
- C. How well it achieves its intended purpose or helps accomplish goals.
- D. Its ability to be used without any instructions.

5. A software development team is working on a major update. Why would prioritizing the ease with which new programmers can understand the existing code be just as important as how smoothly the program runs for the end-user?

MULTIPLE CHOICE

Choose the correct answer:

- A. User satisfaction is solely dependent on code readability.
- B. Only new programmers need to understand the code, not existing team members.
- C. Readable code ensures efficient maintenance, collaboration, and bug fixing for future development.
- D. Smooth program operation for the end-user is not a critical concern.

Answer Keys & Solutions

Questions

1. A new programmer joins a team working on an existing calculator application. They need to add a new function for square roots. According to the passage, which aspect of the existing code would be most important for this new programmer to evaluate?

MULTIPLE CHOICE

Correct Answer:

- | | |
|---------------------------------|-------------|
| A. Its readability | ✓ Correct |
| B. Its user satisfaction rating | ✗ Incorrect |
| C. Its processing speed | ✗ Incorrect |
| D. Its color scheme | ✗ Incorrect |

Explanation:

Think about what helps a human understand written code.

2. A user is trying out a new online image editor. They find it difficult to locate the 'crop' tool, and the icons don't clearly indicate their functions. Which aspect of usability is primarily being challenged in this scenario?

MULTIPLE CHOICE

Correct Answer:

- | | |
|---------------------|-------------|
| A. Efficiency | ✗ Incorrect |
| B. Memorability | ✗ Incorrect |
| C. Error prevention | ✗ Incorrect |
| D. Learnability | ✓ Correct |

Explanation:

Consider how easy it is for a new user to figure out how to use the program.

3. You are using a new online unit converter tool. You notice that it takes many clicks to perform a simple conversion, even after you know how to use it. Which aspect of the tool's efficiency is being negatively impacted?

MULTIPLE CHOICE

Correct Answer:

- | | |
|----------------------------|-------------|
| A. Time savings | ✓ Correct |
| B. Resource consumption | ✗ Incorrect |
| C. Automation capabilities | ✗ Incorrect |
| D. Processing speed | ✗ Incorrect |

Explanation:

Consider how quickly a task can be completed with minimal effort.

4. A language learning app helps users successfully achieve fluency in a new language. According to the passage, this app would be considered highly effective. What specifically defines a tool's effectiveness?

MULTIPLE CHOICE

Correct Answer:

- | | |
|---|-------------|
| A. Its speed of task completion. | ✗ Incorrect |
| B. How well it performs a task with minimal waste. | ✗ Incorrect |
| C. How well it achieves its intended purpose or helps accomplish goals. | ✓ Correct |
| D. Its ability to be used without any instructions. | ✗ Incorrect |

Explanation:

Focus on whether the tool actually helps you achieve your desired outcome.

5. A software development team is working on a major update. Why would prioritizing the ease with which new programmers can understand the existing code be just as important as how smoothly the program runs for the end-user?

MULTIPLE CHOICE

Correct Answer:

- A. User satisfaction is solely dependent on code readability. ✗ Incorrect
- B. Only new programmers need to understand the code, not existing team members. ✗ Incorrect
- C. Readable code ensures efficient maintenance, collaboration, and bug fixing for future development. ✓ Correct
- D. Smooth program operation for the end-user is not a critical concern. ✗ Incorrect

Explanation:

Think about the long-term implications of code quality for a development team.