Office Solutions Development

Topic 12:

Evaluating Software Development

Scope and Coverage

▶ This topic will cover:

  ▶ The need to evaluate software development, the different types of evaluation techniques, software functionality, efficiency, reliability and usability.

  ▶ Successful user interaction, the identification of enhancements and potential improvements to software

Learning Outcomes - 1

▶ By the end of this topic, students will be able to:

  ▶ Explain the need for evaluation of software development

  ▶ Identify different types of evaluation techniques

  ▶ Evaluate the functionality of software

  ▶ Evaluate the efficiency of software

  ▶ Evaluate the reliability of software

  ▶ Evaluate the usability of software

Learning Outcomes - 2

- By the end of this topic, students will be able to:
  - Discuss how successful user interaction can be identified
  - Discuss how software enhancements can be identified
  - Discuss how potential improvements can be identified
  - Produce an evaluation checklist

Terminology

▶ Terminology will be explained in the lecture, tutorial and laboratory session. You should take notes!

▶ Ask questions if you do not understand.

Evaluation of Software Development

▶ Evaluation is required to check for:

  ▶ Functionality

  ▶ Efficiency

  ▶ Reliability

  ▶ Usability

▶ It helps to ensure that the software is developed to the highest standards.

▶ It can also help to improve productivity and enhance profitability.

Types of Evaluation Techniques

▶ Software can be evaluated using a number of types of techniques, such as:

    ▶ questionnaires

    ▶ observations

    ▶ Interviews

▶ Whatever method is used, if it is performed by end users, it can provide clear statements of any problems.

Evaluation Criteria

- The *functionality, efficiency, reliability and usability* of the software needs to be evaluated.

- There are three main criteria that need to be included when evaluating the above.

  - What information is provided by the software – is it reliable and user friendly?

  - How effective is *interaction* with the software – how efficient and usable is it?

  - Are the *technical* aspects of the software appropriate –is it fully functional and reliable?

Evaluating Functionality

- ▶ The main purpose of evaluating functionality is to ensure that the software performs the necessary commands as instructed.

- ▶ Functionality should have been tested thoroughly and should be free of errors before an end user evaluates it.

- ▶ An end user may, however, suggest enhancements even if the software is fully functional.

Evaluating Efficiency

- The main purpose of evaluating efficiency is to check whether the software performs all functions consistently.

- Are routine tasks automated?

- Does it allow commands to be undertaken quickly, with the minimum of actions/keystrokes?

- Are instructions to the user provided clearly?

- Does it operate quickly?

- Does it have the potential to help increase productivity?

Evaluating Reliability - 1

- The main purpose of evaluating reliability is to check that the software does what it is meant to do and does not perform any unexpected action.

- Do all the links work?

- Does it help the user recognise, diagnose and recover from an error?

Evaluating Reliability - 2

- Does it inform the user about the problem and suggest a solution in clear and unambiguous language?

- It can also evaluate how secure the software is.

Evaluating Usability - 1

▶ The main purpose of evaluating the usability of the software is to check that it can be interacted with easily.

  ▶ Is the language clear and understandable?

  ▶ Is help available and if so, is it easy to understand?

  ▶ Is it easy to navigate the software?

  ▶ Are directions easy to understand?

  ▶ Is information is accurate and up-to-date?

Evaluating Usability - 2

- Is the text is easy to read?
- Are the graphics relevant?
- Is the layout consistent?
- Is feedback, such as an error message, clearly presented?
- Is it easy to learn?

Identifying Successful User Interaction

▶ Successful user interaction is achieved when after evaluation:

▶ no weaknesses are identified.

▶ the results of whatever evaluation technique has been used are positive.

▶ there are no recommendations for improvements.

Identifying Software Enhancements and Improvements

▶ When user interaction has been evaluated as successful overall, it may be that the testing technique has returned comments that recommend how the software could be enhanced, e.g. a user may suggest a colour or font style change, which does not affect the functionality of the software feature or function.

▶ When improvements are suggested, this can mean that the software may not be not functioning as efficiently or reliably as it could.

Evaluation of Application Software – Some General Questions

▶ Does it perform all of the required functions?

▶ Can it be updated easily to accommodate changing user or business requirements?

▶ Does it have links to other software, e.g. accounting software linked to logistics and payroll software?

An Evaluation Checklist

► This should include questions on:

► information provided by the program

► user interaction with the software

► technical aspects of the software

► software strengths

► software weaknesses