

## **An approach of software maintenance**

*Paper contains a literature review about software maintenance standards and models. Analysing and determining difference of existing software maintenance models is represented. A new approach of software supporting by adding functionality of data visualization is proposed.*

### **Introduction. Software maintenance.**

Software maintenance is an integral part of a software life cycle. After the product is released, software maintenance is performed for a variety of purposes, such as improving the software in general, fixing problems or bugs, improving performance, and more [1]. As maintenance is often expensive, there is a reason to pursue further research to enhance productivity of maintenance activities.

Software maintenance is defined in a few standards: IEEE 1219, ISO/IEC 12207, ISO/IEC 14764. The standards ground the foundation knowledge about software maintenance. There are defined key terms, describes software maintenance categories and main actors, basic principles for organizing software maintenance processes and basic approaches of maintenance. All this thing helps to ground the foundation of designing a new software maintenance approach integrated to domain engineering fundamentals [4].

### **Software maintenance models.**

In order to fully analyse software maintenance process, it is also important to consider maintenance models, which provide specific recommendations for organizing of the software maintenance process [2]. In addition, it let to bring to light the specific drawback of organizing of the software maintenance process.

*Software maintenance model* is an abstract representation of the evolution of software to help analyze activities during software maintenance. There are several popular maintenance models: Quick-fix model, Boehm model, IEEE model, Iterative Enhancement model.

Quick-fix model where the maintenance process is a fire-fighting approach, which is the temporary custom software maintenance method, software problem should be solved as soon as possible, shouldn't analyze long-term effects on the implementation of changes [3].

Boehm model maintenance process is divided into management decision-making to achieve change, software delivery and evaluation of four stages, expressed as a closed loop to maintain the process by promoting the maintenance management decision-making process. From the functional point of view of production, the model reflects the economics of investment and the relationship between earnings.

IEEE model details the process of software maintenance activities, as standard, can be applied to all software maintenance process. However, different software, due to different characteristics, maintenance procedures will be different, IEEE is a large and comprehensive model specification for different software.

Iterative-enhancement is an evolutionary model proposed for development in environments where the complete set of requirements for a system was not fully understood or where the developer did not know how to build the full system.

**Comparative analysis of software maintenance models.**

During analysis of the software maintenance models, we found some characteristics by which these models can be compared. Comparison given in table 1. Characteristics are classified into three levels: low, medium, high or into two logic meaning: yes, no. Comparison characteristic are listed below.

Analyse long-term effects – how model analyse long-term effects of the implementation of changes.

Code analysing – the depth of code analysis that is provided by the given model.

Driven by economical aspect – how economic interest drive the software maintenance process.

Clarity of requirements – how clear or complete is set of requirements for the system, is developer know how to build the full system.

Time spending – show level of time spending to maintain software.

Money cost – show how much does software maintenance cost for a specific model.

Can be outsourcing – shows whether the maintenance process can be outsourced.

Documentation ability – shows the level of work on documentation.

*Table 1*

Software maintenance models comparison

Characteristic	Quick-fix model	Boehm model	IEEE model	Iterative enhancement model	New model
Analyse long-term effects	low	high	high	medium	medium
Code analyzing	low	low	medium	high	medium
Driven by economical aspect	no	yes	no	no	no
Clarity of requirements	medium	low	medium	low	high
Time spending	low	medium	medium	medium	medium
Money cost	low	medium	high	medium	medium
Can be outsourcing	no	yes	yes	yes	yes
Documentation ability	low	high	high	medium	high

**Proposed approach of software maintenance.**

Prototype of developed software maintenance model depicted on fig. 1.

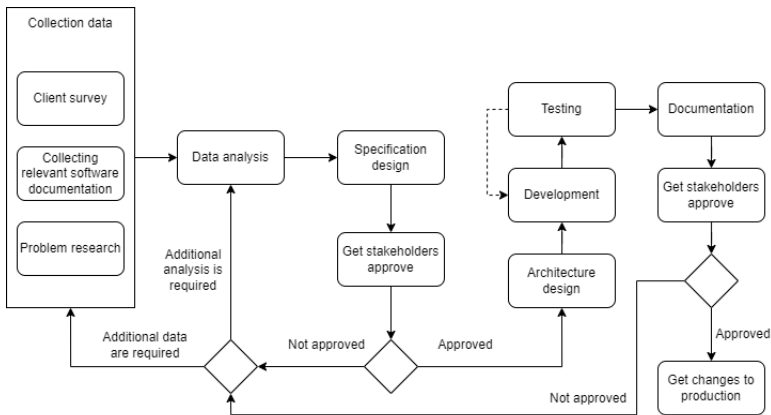


Fig. 1. Prototype of software maintenance model

The sequence of activities in the new approach can be described in the form of a list.

1. Determination of the functionality of the existing software that meets the objectives of the project.
  - 1.1. Analysis of documentation and other software development artefacts.
  - 1.2. Partial source code analysis.
2. Requirements analysis.
  - 2.1. Specification design.
    - 2.1.1. Trace requirements with artefacts [4].
  - 2.2. Scrum-meeting with the client.
3. Architecture design.
  - 3.1. Overcoming the risks of linking class diagrams of existing components with new ones.
  - 3.2. Choosing stack of technologies.
  - 3.3. Creation of architecture design.
4. Development.
  - 4.1. Functional development.
  - 4.2. Interface development.
5. Testing.
  - 5.1. Interaction testing of interfaces.
  - 5.2. Functional testing.
6. Documentation.
  - 6.1. UML diagrams.
  - 6.2. User documentation.
7. Get stakeholders approve.
8. Publish new additional functionality.

Points 2.1 and 2.2 may be in several iterations. And also points from 1 to 7 also can have more than one iteration depending on software maintenance results.

This approach provides better requirements analysis and its iterativeness provides additional flexibility in the process of maintenance.

### **Conclusion**

Software maintenance is one of the important process in software development lifecycle model. Software maintenance problems in the software development life cycle, consider that in later maintenance stages costs of the software and its complexity will be greater. Paper presents the review of software maintenance fundamentals and analysis of drawbacks and advantages of software maintenance models. Proposed software maintenance approach allows reducing cost of the maintenance process due to involvement of domain analysis activities into requirement analysis activities [4].

### **References**

1. Pabasara T., Hettige H. B. Maintain to gain fame: A Research review on Software Maintenance. 15th International Research Conference of KDU 2022. – Sri Lanka, July 2022. – 8 p.
2. "ISO/IEC/IEEE Approved Draft International Standard - Software Engineering - Software Life Cycle Processes - Maintenance," in ISO/IEC/IEEE P14764/FDIS, July 2021, no., vol., 13 Dec. 2021, 1-46 p.
3. Yongchang, R., Tao, X., Zhongjing, L., & Xiaoji, C.. Software Maintenance Process Model and Contrastive Analysis. 2018 International Conference on Information Management, Innovation Management and Industrial Engineering. – 2018, – P. 169-172.
4. Chebanyuk O. Palahin O. Markov Kr. Domain engineering approach of software requirements analysis Проблеми програмування. – 2020. – №2-3. – С. 164-172.