



ISSN: 0975-766X
CODEN: IJPTFI
Research Article

Available Online through
www.ijptonline.com

PREDICTION OF LIFECYCLE MODELS FOR SOFTWARE

Mohamed Fahad M¹, Simama Hassan Mottur², Prasanna M³, Mohammed Fazeel K⁴

1,2,4 MCA, SITE School, VIT University, Vellore, Tamilnadu, India.

3 Associate Professor, SITE School, VIT University, Vellore, Tamilnadu, India.

Email: prasanna.m@vit.ac.in

Received on 25-10-2016

Accepted on 02-11-2016

Abstract

A Life cycle model for Software is an important aspect for development. Choosing a correct Life cycle model for Software can play a crucial role. If we are not selecting a proper life cycle model for Software it can lead to many drawbacks. Drawbacks such as wastage of time, wastage of Resources. So, selecting a correct Lifecycle model can make a huge impact on the development factors. This can be done by using the Prediction Methodologies. Based On the three Factors (1) Requirements (2) Development Team (3) Users.

If the requirements are more it will predict appropriate model for that. If the development team has less experience it will predict a particular model for that. If the users have less domain knowledge or less knowledge about technologies it will predict a separate model for that. By using prediction analysis, Prediction of lifecycle models can be done. This can improve the quality of the software. As well as time constraints. By using prediction of life cycle models proper planning can be done.

Keywords: Life cycle model, Prototype model, Agile Process model

1. Introduction

A life cycle model is a common method used for creating an application. There are many lifecycle models used for creation of softwarebased onthe requirements given by the user.

As well as development team, time constraints. Various lifecycle models include (i) Build and Fix Model (ii) Waterfall Model (iii) Prototype Model (iv) Iterative Enhance Model (v) Evolutionary Development Model (vi) Spiral Model (vii) Agile Process Model. This model is used according to the requirements, development team and users. These models are used in order to improve the quality of the software.

2. Problem Definition

As we know the importance of life cycle models and we also know why it is used but there is a lot of misconception of how and when it should be used.

So there needs to be a method or some solution for this problem. If we try to solve this problem it can be helpful for all the employees as well as students for completing their projects.

3. Survey

i) Doyou follow lifecycle models for creating an application or software?

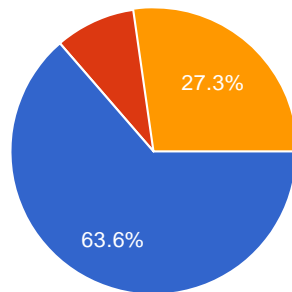


Figure 1: Survey result of first question

Yes	7	63.6%
No	1	9.1%
Maybe	3	27.3%

ii) On what basis do you follow life cycle model?

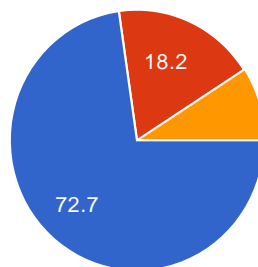


Figure 2: Survey result of second question.

Basedonrequirements	8	72.7%
Basedondevelopmentteam	2	18.2%
Basedonusers	1	9.1%

iii) Is it time consuming?

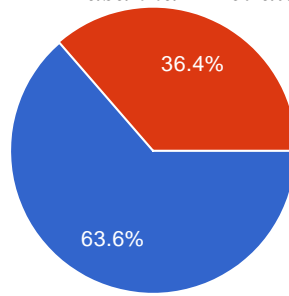


Figure 3: Survey result of third question.

Yes	7	63.6%
No	4	36.4%

iv) What method do you follow for creating a software?

- Water fall
- Rough Draft
- Check_List
- Rough Draft
- Flow Chart
- Java
- IDE
- Agile Model
- Windows
- No Method
- Depends

4. Solution

The solution for the above problem can be done using *prediction analysis*. By using prediction analysis, this problem can be solved and one can get the correct lifecycle model. This can improve the productivity of Software as well as can save human efforts.

5. Prediction analysis

Here a new method is defined in which a user has to enter some of the details in an application. Based on the answers for the given questions Analysis is done. And prediction is done according to it. Prediction Analysis is done on three modules.

6. Methodology:

- i) Based on Requirements

A set of Questions will be asked based on the Requirements. For example:

- Are requirements easily understandable and defined?
- Do we change requirements often?
- Can we define requirements early in the lifecycle?
- Requirements are indicating a Complex System to be built?

Based on the answers given by users. Prediction of Model is done.

ii) Based on Development Team

A set of Questions will be asked based on the Development Team. For example:

- Less Experience on Projects?
- Less Domain Knowledge?
- Less Experience on tools to be used?
- Availability of Training if Required?

Based on the answers, Prediction of Model is done.

iii) Based on Users

- User Involvement in all phases?
- Limited User Participation?
- User Have no previous experience in participation?
- Users are expert of Problem Domain?

Based on the answers Prediction of Model is done.

7. Architecture Diagram

The architecture diagram for the proposed methodology is shown in the following figure.



Figure 4: Architecture diagram

8. Snapshots

PREDICTION OF LIFE CYCLE MODELS FOR SOFTWARE

FILL THE FOLLOWING CATEGORIES

REQUIREMENTS
DEVELOPER
USERS

The diagram illustrates the Software Development Life Cycle (SDLC) as a circular process. It includes the following phases: Requirement Analysis (orange), Design (purple), Development (blue), Testing (green), and Deployment (red). The center of the cycle is labeled 'Software Development Life Cycle (SDLC)'.

REQUIREMENTS BASED QUESTIONS

ARE REQUIREMENTS EASILY UNDERSTANDABLE AND DEFINED ?
 YES NO

DO WE CHANGE REQUIREMENTS OFTEN ?
 YES NO

CAN WE DEFINE REQUIREMENTS EARLY IN THE LIFE CYCLE ?
 YES NO

REQUIREMENTS ARE INDICATING A COMPLEX SYSTEM TO BE BUILT ?
 YES NO

REQUIREMENTS BASED QUESTIONS

ARE REQUIREMENTS EASILY UNDERSTANDABLE AND DEFINED ?
 YES NO

DO WE CHANGE REQUIREMENTS OFTEN ?
 YES NO

CAN WE DEFINE REQUIREMENTS EARLY IN THE LIFE CYCLE ?
 YES NO

REQUIREMENTS ARE INDICATING A COMPLEX SYSTEM TO BE BUILT ?
 YES NO

Message

the predicted model for the selected option is:WATERFALL

OK

USER BASED QUESTIONS

USER INVOLVEMENT IN ALL PHASES ?
 YES NO

LIMITED USER PARTICIPATION ?
 YES NO

USER HAVE NO PREVIOUS EXPERIENCE OF PARTICIPATION ?
 YES NO

USER ARE EXPERTS OF PROBLEM DOMAIN ?
 YES NO

Message

the predicted model for the selected option is:SPIRAL

OK

9. Conclusion

Prediction of life cycle models can make a lot of impact in improving the quality of software, time, human effort, as well as resources. This can be very helpful for all the students and as well as in software organizations who are doing their Projects.

References

1. K. K Agarwal, 'Software Engineering', 5th Edition: New Age International (P) Limited; 2014.p 1-50.
2. Ian Sommerville, 'Software Engineering', 9th Edition:Pearson; 2015,p-1-100.
3. A Comparison between Five Models Of Software Engineering - Nabil Mohammed Ali Munassar1 and A. Govardhan- IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 5, September 2010.
4. A Comparative Analysis of Different types of Models in Software Development Life Cycle- Ms. Shikha maheshwari, Prof.Dinesh Ch. Jain. International Journal of Advanced Research in Computer Science and Software Engineering.