Use of Artificial Intelligence in Software Development Life Cycle Requirements and its Model

VIPAN KUMARI[1], SANDEEP KULKARNI[2]

[1]F-84 UGF. Near Vibhuti Apartment, Sai Chowk Chhatarpur New Delhi 110074
[2]C/O G S Laxman,#4094/20,1st A main, B block, 2nd stage,Rajajinagar,Banglore-560021
[1,2]Himalayan University Arunachal Pradesh, Itanagar, India

---***---

Abstract:- *SDLC-It is the process that describe how to develop, design and maintain the software project ensuring that all the functional and user requirement, goal and objectives are met. Artificial Intelligence (AI) is the younger field in computer science ready to accept challenges. Software engineering (SE) is the dominating industrial field. So, automating SE is the most relevant challenge today. AI has the capacity to empower SE in that way. Here in this paper we present a state of the art literature review which reveals the past and present work done for automating Software Development Life Cycle (SDLC) using AI.SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. It aims to be the standard that defines all the tasks required for developing and maintaining software. It is process that describes how to develop, design, and maintain the software. This techniques improve the quality of the software project and over all the process of software development.*

Keywords: — Software develop1, Artificial Intelligence2, Software Code Generation3, Requirements Engineering4, SDLC, Software Design5, Software Estimation6, Software Testing7.

1. INTRODUCTION

Artificial intelligence and software engineering have developed separately. AI research techniques make it possible to perceive, reason and act. Research in software engineering is concerned with supporting engineers to developed better software in less period. Today several research directions of both disciplines come closer together and are beginning to build new research areas. Software agents play an important role as research objects in distributed AI (DAI) as well as in Agent Oriented Software Engineering(AOSE). Knowledge-based System (KBS) are being examine for Learning Software Organizations (LSO) as well as Knowledge Engineering (KE). Artificial Intelligence Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality software. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates. SDLC is the acronym of Software Development Life Cycle.

1.1 USE OF AI IN REQUIREMENT SPECIFICATION

The AI in the requirements engineering SDLC phase are in the following areas:

1. Use of AI in Software Planning and Requirement Analysis:-

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas. Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage.

2. Use of AI in Software Defining Requirements:-

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

3. Use of AI in Software Designing the Product Architecture:-

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification. A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any).

4. Use of AI in Software Building or Developing the Product:-

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle. Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being development.

5. Use of AI in Software Testing the Product:-

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

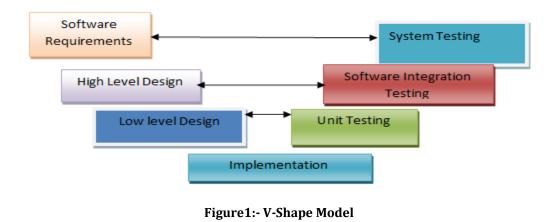### 6. Use of AI in Software design Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing).Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

2.1 Uses of Artificial Intelligence in SDLC Models:-

Following are the popular SDLC models followed in the industry :-

1. V-Model:-

The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape. It is also known as Verification and Validation model. The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. Under the V-Model, the corresponding testing phase of the development phase is planned in parallel. So, there are Verification phases on one side of the 'V' and Validation phases on the other side. The Coding Phase joins the two sides of the V-Model.

Figure1:- V-Shape Model

2.   Waterfall Model:-

The Waterfall Model is a linear sequential flow. In which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete. The waterfall approach is the earliest approach and most widely known that was used for software development.
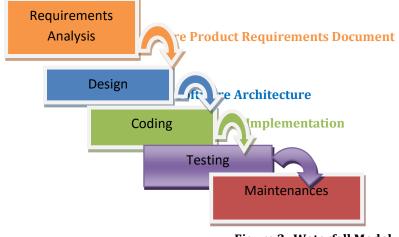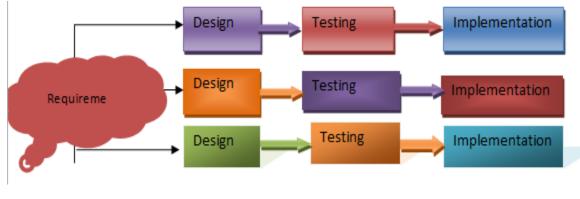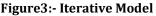

Figure 2:-Waterfall Model

3.   Iterative Model :-

It is developed to overcome the weakness of the waterfall model. It is start with an initial planning and end with development with the cyclic interaction in between. It can consist of mini V-shaped model. It used in shrink-wrap application and large system which built in small segments. It also, can be used in a system has separated component, examples:- ERP system. Which we can start with the budget module as a first iteration and then we can start with inventory module

Figure3:- Iterative Model

4.   Spiral model:-

Spiral model is one of the most important Software Development Life Cycle models, which provides support for **Risk Handling**. In its diagrammatic representation, it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. **Each loop of the spiral is called a Phase of the software development process.** The Spiral model is called as a Meta Model because it subsumes all the other SDLC models. For example, a single loop spiral actually represents the Iterative Waterfall Model. The spiral model incorporates the stepwise approach of the Classical Waterfall Model. The spiral model uses the approach of **Prototyping Model** by building a prototype at the start of each phase as a risk handling technique. Also, the spiral model can be considered as supporting the evolutionary model – the iterations along the spiral can be considered as evolutionary levels through which the complete system is built.
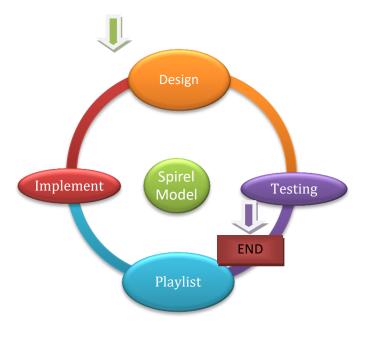
Figure 4:- Spiral Model

2. CONCLUSION

AI is at the centre of a new enterprise to build computational models of intelligence SDLC. The main assumption is that intelligence (human or otherwise) can be represented in terms of symbol structures and symbolic operations which can be programmed in a digital computer. There is much debate as to whether such an appropriately programmed computer would be a mind, or would merely simulate one, but AI researchers need not wait for the conclusion to that debate, nor for the hypothetical computer that could model all of human intelligence. Aspects of intelligent behavior, such as solving problems, making inferences, learning, and understanding language, have already been coded as computer programs, and within very limited domains, such as identifying diseases of soybean plants, AI programs can outperform human experts. In the last few years there has been a lot of interest in the use of anthologies for requirements and design. The development of domain anthologies is making it possible to encapsulate knowledge and rules governing a specific domain in one single resource.

References

[1] Artificial Intelligence using SDLC Author by Bhagyashree W. Sorte, 2 Pooja P. Joshi (2017)

[2] Panday S.K and Mustafa K. [2010] Recent Advances in SRE Research International journal of Computer Science.

[3] Kumar S. Arun and T. Arun Kumar (2011): 'Study the Impact of Requirements Management Characteristics in Global Software Development Projects: An Ontology Based Approach', International Journal of Software Engineering & Applications.

[4] Khatwani Charu (2012): 'Methodologies to Amalgamate Artificial Intelligence and Software Engineering', International Journal of Scientific & Engineering Research.

[5] Fatai Adesina Anifowose, Safiriyu Ibiyemi Eludiora,"Application of Artificial Intelligence in Network Intrusion Detection", SDLC, March (2012).