A Comprehensive Study on Software Development Lifecycle and Systematic Analysis of It Challenges

Mithilesh Pandey[1], Ms. Preeti Kuniyal[2],Mr. Govind Kumar[3]

1Assistant Professor, Department of Computer Science & Engineering, Shivalik College of Engineering, Dehradun
2Assistant Professor, College of Pharmacy, Shivalik, Dehradun

Mithilesh.pandey@sce.org.in

ABSTRACT:The Software Development Lifecycle (SDLC) was used for the software industry to comprehend, execute, and create software products that are of the highest caliber, reliability, efficiency, and timeliness. It is sometimes referred to as a software development process model. There are many SDLC major approaches from which to choose. In this paper, the author tried to define a few SDLC concepts in terms of their most practical uses. The desktop seems to be a need in today's culture given how often it is utilized in industries like banking, education, entertainment, etc. eases the burden of providing software support. Many companies create software solutions with these qualities because programs must deal with complicated, protracted, and repeated activities in a short period of time. Governments, banks, and businesses, for instance, will also be able to fulfill contractual obligations more rapidly. The future potential of SDLC is it can be used in the home cycle of automation and in internet of things also.

KEYWORDS:Design, Development, Model, SDLC (Software Development Life Cycle) Model, Software Engineering.

1. INTRODUCTION

The Software Development Life Cycle (SDLC) is a procedure for organizing, creating, and maintaining machinery and communication. There are many other SDLC models, such as the Flexibility in mind, which consists of five steps that must be finished in order to generate a technological platform, and the Sequence diagram, which is similarly considered as a process that occurs through a certain number of iterations. Additionally, the agile project management method somewhat puts development before the traditional, progressive building method. The seven steps of the process are now conception, goals, analysis, programming, deployment, validation, and assessment. In reality, many academics have studied the SDLC, and a number of models each with unique advantages and disadvantages have been proposed. Rapid application development (RAD), spiral, stepwise, rational unified process (RUP), agile approaches, and fast manufacturing are examples of effective SDLC techniques[1], [2].

Over the last ten years, one of the areas of software engineering that has received the most research has been software architecture. According to a number of academics, electoral decisions are the fundamental elements of software architecture. Web services technology has progressed into a decision-cantered method from with a's mandated. Framework as a Provider and software as a service's framework for sub-service administration and communication are both characterized as having architectural design as a differentiating feature[3]–[5].

Early in the change implementation phase, the architectural design is completed, taking into consideration the relationship between design processes and ultimate quality requirements. Additionally, such a spectacular design that performs the required tasks may satisfy the service objective at the analysis stage. Limitations it may outline the functional criteria for a security solution and helps in the detection and avoidance of security concerns throughout the development process.Figure 1 illustrates the different cycle of the SDLC structure.

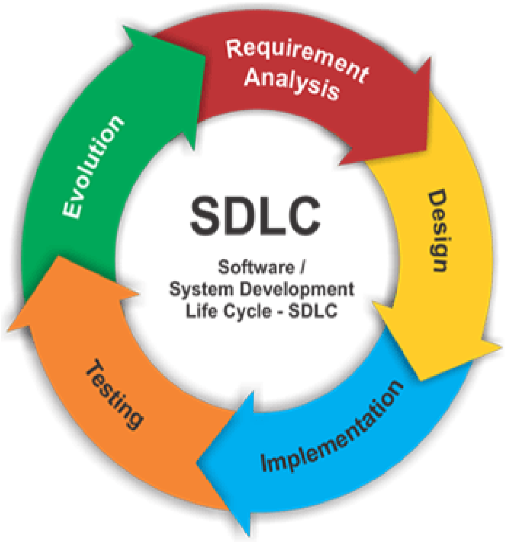

Figure 1: Illustrates the different cycle of the SDLC structure[6]**.**

To fend against cyber security dangers, designers and users must cooperate. The developers established a functional team to examine the sustainability of the system using agile methodologies and to provide the opinions of the stakeholders. Every perspective has the potential to provide a workable answer. Security configuration, particularly during the design and deployment phases, may become troublesome due to technological conflict between efficiency and security. SaaS architecture is a construction process that varies depending on the kind of company being created. The process of offering an online money transfer differs from offering a need since it has different rules, criteria, and goals[7], [8].

- *Stage 1: Planning and Analysis of Requirements*

The most crucial and basic step in SDLC is requirement analysis. It is carried out by senior members of the team with input from the client, the sales department, market surveys, and industry domain specialists. This data is then utilized to establish the main project strategy and undertake product feasibility studies in economic, operational, and technical aspects.The planning step also includes the preparation for quality assurance needs and the identification of project hazards. The technical feasibility study will describe the different technical techniques that may be used to effectively perform the project while minimizing risks.

- *Stage 2: Establishing Requirements*

Following the completion of the requirement study, the next stage is to explicitly describe and record the product needs and get approval from the client or market analysts. This is accomplished via the use of an SRS (Software Requirement Specification) document, which includes all of the product requirements to be defined and produced throughout the project life cycle.

- *Stage 3: Product Architecture Design*

SRS serves as a resource for product architects in developing the optimum architecture for the product under development. Typically, many design approaches for the product architecture are presented and documented in a DDS - Design Document Specification based on the requirements given in SRS.This DDS is evaluated by all essential stakeholders, and the optimal design strategy for the product is chosen based on many characteristics such as risk assessment, product resilience, architectural flexibility, price, and restrictions imposed.A design approach explicitly outlines all of the company's architecture components, as well as its communication and information flow representations with outside and third-party modules (if any). The internal structure of all modules of the proposed system must be explicitly documented in DDS down to the smallest detail.

- *Stage 4: Product Construction or Development*

The real production of the software begins at this level of the SDLC. During this step, the actual coding is produced in accordance with DDS. Code production may be conducted without very much difficulty if the architecture is thorough and structured.Engineers must adhere to their group's writing rules, and coding instruments such as processors, translators, debugging tools, and so on are used to create code. For coding, many high-level application programs including such C, C++, Python, Java, and PHP are utilised. The program code is selected based on the sort of software getting created.

- *Stage 5: Products Validation*

As in contemporary SDLC models, screening operations are mainly included in all SDLC stages, this phase is typically a combination of all phases. Nevertheless, this stage solely relates to the products testing stage, during which product faults are recorded, monitored, repaired, and tested regularly till the target is achieved the assessment methods stated in the SRS.

- *Stage 6: Market Adoption and Management*

Once the product has been rigorously checked and is fully operational, it is officially launched in the relevant market. Item rollout may occur in phases depending on the institution's marketing plan. The product may initially be introduced in a small market sector and evaluated in a real-world business setting (UAT- User acceptance testing).The products could then be launched as is or with proposed improvements in the targeted market niche depending on the comments. Following the manufacturer's release to the marketplace, it is maintained for the current client base.

On the other hand, there are a few factors that apply to all planning processes, such as the need that all services have a security type at different levels. The user is required to correctly, consistently, and securely authenticate both the functional and non-functional aspects of its service. In addition, the basic traits of every SDLC model that has been developed are the same. The majority of them comprise a set of steps or processes that teams of programmers must follow and complete in order to build networks and provide essential commodities. This paper will also examine the benefits and drawbacks of the Flow, Spirillum, and Cumulative models before quickly contrasting these other components.

The Project Lifecycle was one of the early SDLC implementations as well. A number of well-knownconcepts often employ one such perspective. One thing that sets this technique apart is its consecutive stages. It progresses through them. Development, coding, validation, and service come first. Additionally, it guarantees that comparable issues are found before production is done on the ground. This tactic is effective for missions when certification is a high concern since it contains all necessary information and instructions. The phases that make up its structure, in this case, do not really overlap, therefore the waterfall model starts and finishes one cycle before moving on to the next.

As systems go through the SDLC, people utilize the workplace to design, distribute, install, configure, manage, test, and operate them. Every classroom seems to be built with a specific purpose in mind and tied to various facets of the SDLC. The advantages of such ecosystems include the following: a scripting engine that allows businesses to operate independently after trying to merge their work with it as well as others for a week; a collaborative environment where incorporated work may be constructed as a unified system; Team members may test against their original requirements across a subscriber acceptance laboratory environment; primary monitoring of a console's integrating vectors across other source or surface and groundwater pipelines in a software quality assurance laboratory setting; and The item is now available for use by many users before solutions are finally announced[9]–[12].

2. DISCUSSION

Obligation: An obligation is a symbol of behavior that must be developed. Typically, customer-provided papers are used. Thus, it generates the technical specifications and functionality contracts between the client and the developer. Briefly said, requirements are acquired, analyzed, and then sufficient documentation is produced to help with the design phase Analysis all information gathered throughout the research process is analyzed, and a business strategy will also be created. It's all about the software project's development and critical thinking phases.

Computing during this phase the whole set of requirements will be converted to the production process. Testing the objective of this round is to really screen and validate the software suite that has been developed to meet the specified standards. Bugs and system faults are also found, fixed, and improved at this stage[13]–[16].

Deployment once the software has been deployed, it may need to be modified, improved, error-checked, and polished. Therefore, this approach is the first step in resolving such problems. The process of corkscrew testing involves fixing software. A penetration testing system that combines the benefits of side scrolls and bottom-up approaches while integrating elements like framework and idea in phases is a systems engineering system, which put it another way, refers to the practice of detecting, analyzing, and repairing flaws. Additionally, it focuses on minimizing the project plan and controlling risks. This may be accomplished by disassembling a network into smaller parts, allowing for easier alterations during the implementation plan as well as concerns about risk management and project continuity over the whole lifecycle. Every application team in this discussion begins with a modest set of criteria and works its way through each company process to achieve those goals. As a result, the functional structure may utilize its wireframes to identify and correct errors. As long as the software is not suited for design and construction, the firm should link back for any future requests. In this generation, everything has been altered, and the characteristics of the car are introduced.Figure 2 embellishes the different stages of the SDLC structure.

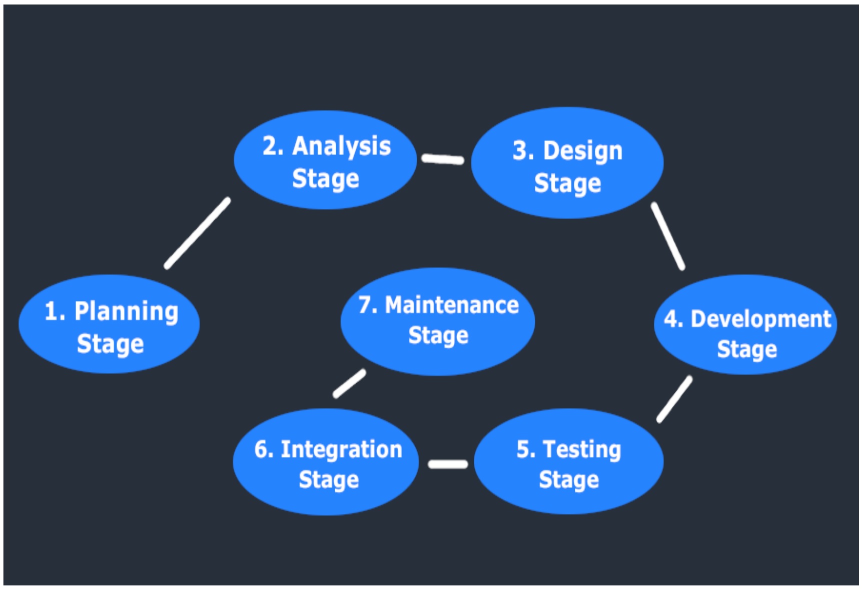

Figure 2: Embellishes the different stages of the SDLC structure[17]**.**

2.1.The following steps provide a quick description of the Spiral model's stages:

Development Scope during this phase, customers and integrators communicate often to analyze project requirements. In risk assessment during this stage, a system is recommended to manage risks and workable solutions. A design is created at the end of this stage. Development/Engineering this stage includes both the development and testing of software. Systematic Approach the client may assess the vendor's performance but at a point before moving on to the next round with the little. In this study's circumstances, the n-back paradigm is utilized to test working memory function, combining aspects of the backs paradigm. It is a frequently used technology in neuroimaging that is under progress. Programming that is both functional and non-functional incrementally is also produced by automated signal transduction mechanisms. Auxiliary configurations were never accessible at this time, despite the fact that the basic measurement matches the most crucial needs and constitutes the offering. A subset of a whole device's implementations might be sent using this manner. After then, more features are gradually added. Each movie recommendation system will thus add a new item to this list every time it is updated until all necessary qualities have been incorporated[18], [19].

3. CONCLUSION

A helpful method for producing high-quality software is the Software Development Lifecycle (SDLC). With the help of this application, software developers may be guided through the development process. Organizations may choose from a number of SDLC techniques, including waterfall, V-model, iterative, spiral, and agile models. A suitable SDLC model may be chosen based on the demands of the client and the corporate objectives. As long as they are aware of the demands of the client and follow a predetermined strategy, developers may profit from the SDLC in software development. According to our research, there aren't many approaches in use for developing systems based on customer requirements and project scope. Because of their many features or simply how well they understand the expectations of their target market, certain innovative designs catch on in society. Although the project life cycle, prototype model, and planning phase all have certain essential similarities, when it comes to creating systems, we each have specific benefits and disadvantages, thus each model seeks to address the drawbacks of the one before it. Future simulations will be added to this study to further it, since some of those model types may be simulated using a variety of techniques.

REFERENCES

[1] D. P. Pocobelli, J. Boehm, P. Bryan, J. Still, and J. Grau-Bové, "Pocobelli, D. P., Boehm, J., Bryan, P., Still, J., &amp; Grau-Bové, J. (2018). BIM for heritage science: a review. Heritage Science, 6(1), 30.," *Heritage Science*. 2018.

[2] T. Merel, "Virtual, Augmented and Mixed Reality are the 4th Wave of Digital Technology," *Digi-Capital*, 2016.

[3] The passive house resource, *Planning criteria for Passive Houses in New Zealand*. 2014.

[4] J. W. Choi, "Sneakerheads' Assessment of Sneaker Value and Behaviors throughout the Sneaker Ownership Cycle," 2017.

[5] V. J. Rideout, U. G. Foehr, and D. F. Roberts, *Generation M2: Media in the Lives of 8- to 19-Year-Olds*. 2010.

[6] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Futur. Gener. Comput. Syst.*, 2018, doi: 10.1016/j.future.2017.11.022.

[7] *et al.*, "Security Challenges and Solutions in theInternet of Things," *Nord. Balt. J. Inf. Commun. Technol.*, 2018, doi: 10.13052/nbjict1902-097x.2018.010.

[8] K. L. Lueth, "State of the IoT 2018: Number of IoT devices now at 7B – Market accelerating," *IoT Anal.*, 2018.

[9] F. Mohamed, J. Abdeslam, and E. B. Lahcen, "Towards new approach to enhance learning based on internet of things and virtual reality," 2018. doi: 10.1145/3230905.3230955.

[10] K. H. Chang, "Bluetooth: A viable solution for IoT?," *IEEE Wirel. Commun.*, 2014.

[11] Ș. Adrian and V. Diaconescu, “Hide’N’Seek: an Adaptive Peer-To-Peer Iot Botnet,” *Virusbulletin*, 2018.

[12] M. Gandhi and S. Yadav, “Energy efficiency through technology management: A literature review,” *International Journal of Energy, Environment and Economics*. 2017.

[13] T. L. Chavan, “Performance Analysis of PV Modules: A Review,” *Int. J. Trend Sci. Res. Dev.*, 2018, doi: 10.31142/ijtsrd15819.

[14] R. Aunkofer, “Connecting the World and Reinventing Customer Centricity,” *NIM Mark. Intell. Rev.*, 2018, doi: 10.2478/gfkmir-2018-0019.

[15] F. Haouari, R. Faraj, and J. M. Alja’Am, “Fog Computing Potentials, Applications, and Challenges,” 2018. doi: 10.1109/COMAPP.2018.8460182.

[16] R. Venkataramanan, V. Sridharan, D. Kadariya, A. Sheth, K. Thirunarayan, and M. Kalra, “KNOWLEDGE-ENABLED PERSONALIZED DASHBOARD FOR ASTHMA MANAGEMENT IN CHILDREN,” *Ann. Allergy, Asthma Immunol.*, 2018, doi: 10.1016/j.anai.2018.09.134.

[17] E. S. *et al.*, “Health management and pattern analysis of daily living activities of people with dementia using in-home sensors and machine learning techniques,” *PLoS One*, 2018.

[18] E. Pasqua, “LPWAN emerging as fastest growing IoT communication technology – 1.1 billion IoT connections expected by 2023, LoRa and NB-IoT the current market leaders,” *IoT Analytics*, 2018.

[19] T. Obi, “The 14TH WASEDA – IAC international digital government rankings 2018 report,” *Inf. Poility*, 2018.

[20] Panwar, K, Murthy, D, S, “*Analysis of thermal characteristics of the ball packed thermal regenerator*”, Procedia Engineering, 127, 1118-1125.

[21] Panwar, K, Murthy, D, S, “*Design and evaluation of pebble bed regenerator with small particles” Materials Today,* Proceeding, 3(10), 3784-3791.

[22] Bisht, N, Gope, P, C, Panwar, K, “ *Influence of crack offset distance on the interaction of multiple cracks on the same side in a rectangular plate*”, Frattura ed IntegritàStrutturale” 9 (32), 1-12.

[23] Panwar, K, Kesarwani, A, “Unsteady CFD Analysis of Regenerator”, International Journal of Scientific & Engineering Research, 7(12), 277-280.

[24] Singh, I., Bajpai, P. K., & Panwar, K. “*Advances in Materials Engineering and Manufacturing Processes*