iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XI **Month of publication:** November 2023

DOI: https://doi.org/10.22214/ijraset.2023.56554

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

Software Development Life Cycle (SDLC)

Rutuja S. Ghumatkar[1], Anup Date[2]
Department of Information Technology Dr .D.Y Patil University Pune, Maharashtra, India

Abstract: ***The software development life cycle (SDLC) process plays an important role in the development and delivery of a software product. This research article provides an in-depth analysis of various SDLC approaches and their unique capabilities, benefits, and challenges. By reviewing traditional and modern SDLC models,this article aims to provide insight to software developers, project managers, and project partners to help them make decisions. Information and development into the entire software development process.***

I. INTRODUCTION

SDLC contains a detailed plan for how the software will be developed, modified, maintained and updated. The software development life cycle model is a model that describes the entire list of how software will be developed, explaining how this is done at each stage. There are many types of models, such as the waterfall model, V-shaped model, evolutionary prototype model, spiral model, iterative and incremental model, and agile model. Therefore, to ensure the success of the project, the SDLC model will need to be selected based on the specific problems and needs of the project.

Software Development Life Cycle - SDLC is a method for completing the software development process on time and controlling software quality. The development life cycle provides a set of activities to be performed during development, often referred to as the software development life cycle. Software development is divided into tasks that enable all software development companies to easily manage their software products. The software development lifecycle model approaches the software development process step by step. If the process is strong, the end product will be strong and the project will be successful.

II. HISTORY OF SDLC

The job of "software developer" has been around since the first computers and their operators, to the days of ENIACs and vacuum cleaners. Software development practices and techniques have evolved over the years since the invention of the computer. This process is adapted to new technologies of modern thought in computer hardware, development tools and software development team organization and management. With this progress, new software development methods are emerging in private and public software development worldwide.

These methods vary widely in approach, yet they share a common goal: to develop software as cheaply, efficiently, and effectively as possible. The software process model is the representation of process it presents and the description of a process are as:

1) Specification
2) Design
3) Validation
4) Evolution

The software development life cycle is all about:

a) Understanding the problem. i.e. (problem domain)
b) Decide a plan for a solution. i.e. (solution domain)
c) Coding the planed solution
d) Test the actual program
e) Maintain the product

III. PHASES OF SDLC

1) *Planning:* The planning phase includes project analysis, feasibility studies, project decision making and software development plans.
2) *Requirements gathering:* The requirements gathering process involves collecting and documenting functional and non-functional requirements for software requirements. This involves gathering information about what the software is supposed to do, how it is supposed to work, and any other specific requirements.

©IJRASET: All Rights are Reserved

3) *Design:* The design phase consists of detailed design features of the software. This involves creating a design process, creating a detailed design for each product, and analyzing interactions between products.
4) *Implementation:* The implementation phase involves writing the software code and integrating various components into the system.
5) *Testing:* The testing phase will ensure that the software works as expected and meets the requirements. This includes functional testing, system testing, and validation.
6) *Deployment:* This term refers to installing the software in the target environment and ensuring that it works properly.

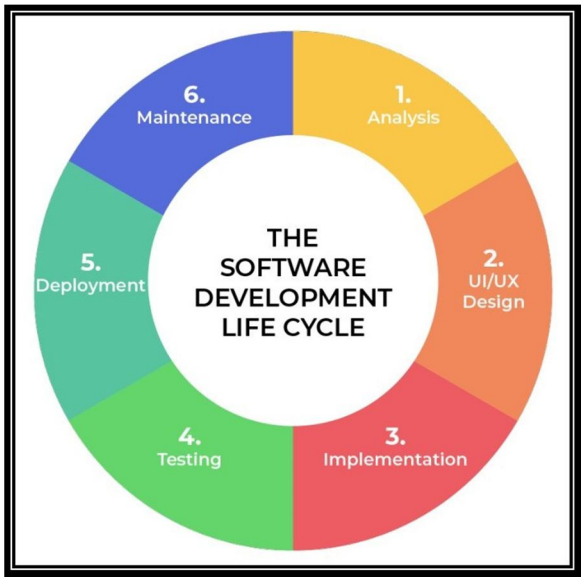


Fig 1: Phases of SDLC

IV. SDLC MODELS

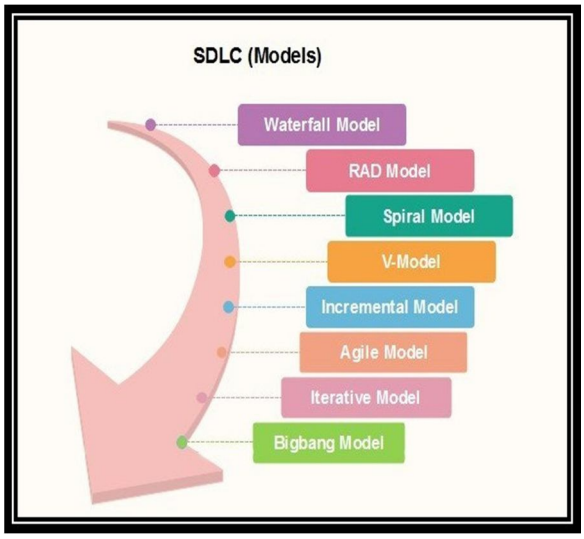


Fig 2: Models of SDLC

A. *Waterfall Model*

Waterfall is the most widely accepted SDLC model. In this way, the entire software development process is divided into several stages.

The waterfall model is a continuous software development model in which the development process is seen to flow continuously downward through requirements (waterfall-like) analysis, design, implementation, testing (verification), integration and integration steps. to manage.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XI Nov 2023- Available at www.ijraset.com

B. RAD Model

RAD or rapid development implementation process is the adoption of the waterfall model; The goal is to develop software in a short time. The RAD model is based on the idea that better systems can be developed in less time by using focus groups to write requirements.

1) Business Modeling
2) data Modeling
3) Process Modeling
4) Application Creation
5) Testing and Return

C. Spiral Models

Spiral Models are based on risk models. This SDLC model allows the team to choose waterfall, incremental, waterfall, etc. It helps in implementing elements of one or more models such as. Spiral technology is a combination of rapid design and integration in design and construction.Each loop in the spiral begins with defining the goal of the loop, possible variables to achieve the goal, and existing constraints.

D. V Model

In the SDLC model testing and development type, these steps are planned in parallel. So, on one side there is the verification phase, and on the other side there is the verification phase. V-Model is added from the coding stage.

E. Incremental Model

Incremental model is not a different model. There should be a series of waterfall loops. At the beginning of the project, requirements are divided into groups. The software for each team is developed according to the SDLC model. Repeat the SDLC process, adding additional features with each release until all requirements are met. In this way, each cycle works as a maintenance phase for the previous software version. Changes to the incremental model allow development to overlap. After that, the next cycle will start before the previous cycle has completed.

F. Agile Model

Agile methodology is a practice that supports the interaction between development and testing during the SDLC process of each project. In the agile method, the entire project is built in small increments. All of these designs are delivered in iterations, with each iteration taking one to three weeks.

According to various perspectives of the software giant, the characteristics of the software agility phase: will change. It is equally difficult to predict how key customers will change as the project progresses.

G. *Iterative Model*

It is a specific application of the software development life cycle that focuses on initialization, simple implementation, and then gradually increasing complexity and scope until the final commit is completed. Simply put, refactoring is a way to break the software development process of a large application into smaller parts.

H. Big Bang Model

The Big Bang Model focuses on all types of services in software development and coding with little or no planning. When emotions arise, we understand and use them.

This model is best for small projects where small groups work together. It is also useful for learning software development projects. This is the best model when demand is unknown or there is no final release date.

I. Prototype Model

The prototype model begins with gathering requirements. Developers and users meet and define the purpose of the software, set the rules, etc.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

V. BENEFITS OF SDLC

1) *Improved Project Management:* SDLC offers a well-defined roadmap for the entire project. This makes it easier to plan, track progress, allocate resources, and set clear milestones. Project managers can make informed decisions and maintain better control over the development process.
2) *Enhanced Quality Assurance:* SDLC incorporates rigorous testing and quality assurance processes throughout the development cycle. This ensures that software is thoroughly tested and validated, reducing the likelihood of defects and improving the final product's quality.
3) *Reduced Risk and Uncertainty:* By breaking the project into distinct phases with defined objectives and deliverables, SDLC helps identify and mitigate risks early in the development process. This minimizes the chances of unexpected issues causing project delays or budget overruns.
4) *Improved Communication and Collaboration:* SDLC encourages collaboration among different project stakeholders, including developers, testers, designers, and business analysts. Clear documentation and communication channels ensure that everyone understands their roles and responsibilities.
5) *Better Resource Allocation:* SDLC allows for efficient allocation of resources like personnel, time, and budget. This optimization ensures that resources are used effectively and minimizes wastage.
6) *Adaptability and Flexibility:* Some SDLC models, like Agile, emphasize adaptability to changing requirements. This flexibility enables development teams to respond to evolving customer needs and market dynamics.
7) *Documentation and Knowledge Preservation:* SDLC promotes comprehensive documentation of project specifications, design, code, and testing procedures. This documentation not only helps in troubleshooting and maintenance but also preserves institutional knowledge for future projects.
8) *Client and Stakeholder Satisfaction:* SDLC ensures that clients and stakeholders are involved in the development process and can provide feedback at various stages. This iterative feedback loop increases the likelihood of delivering a product that meets their expectations and requirements.

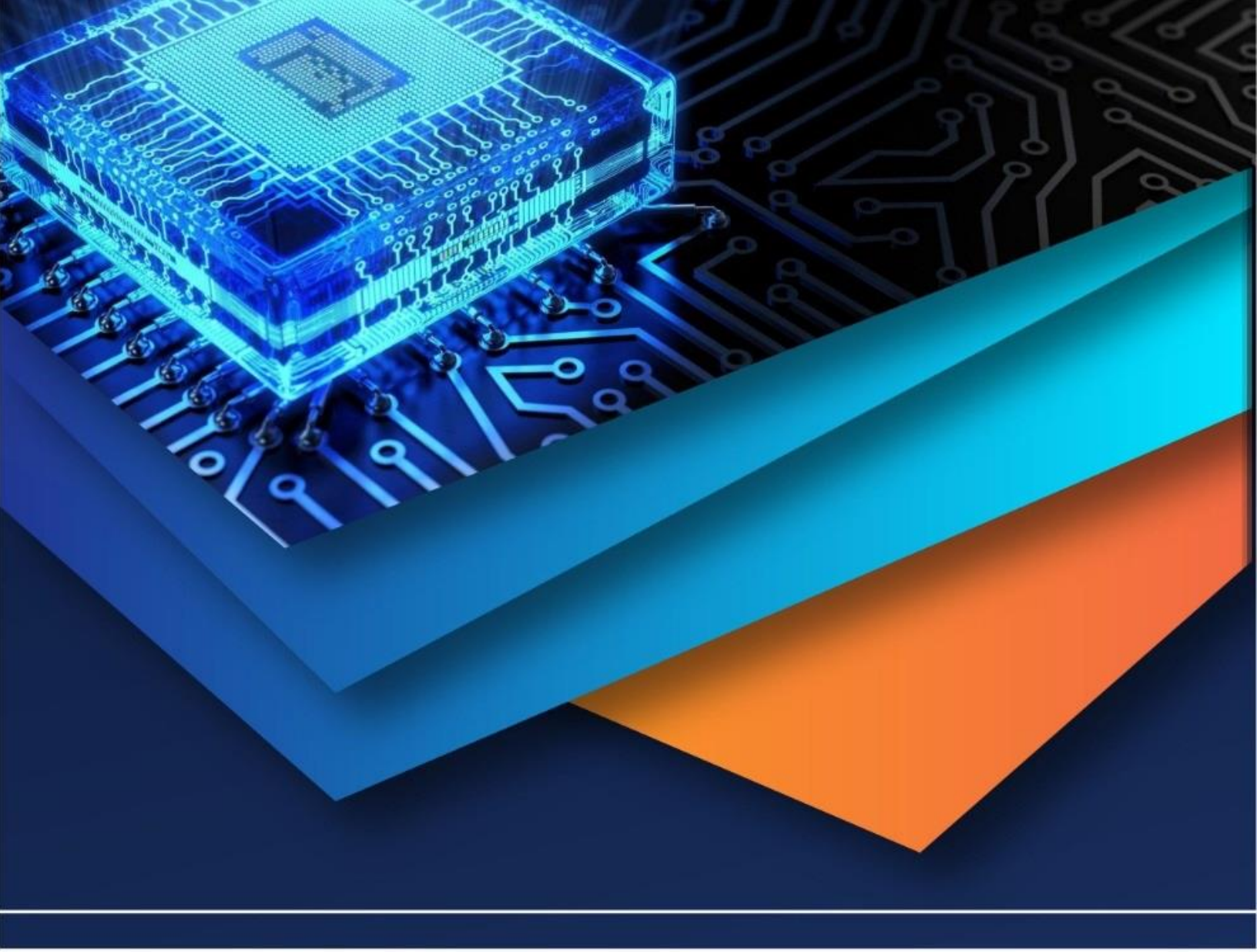
VI. CONCLUSION

This Chances of success in software development. It contains detailed plans that explain how to plan, build, and maintain a particular piece of software. All software starts as an idea and goes through stages until it becomes a development release version. The software development life cycle of an application or system continues with continuous updates and new features added until the day it is retired or replaced software development methods have been developed over the decades.

REFERENCES

[1] https://www.geeksforgeeks.org/software-engineering-prototyping-model/
[2] https://economictimes.indiatimes.com/definition/prototype-model
[3] https://searchcio.techtarget.com/definition/Prototyping-Model
[4] https://www.guru99.com/software-engineering-prototyping-model.html
[5] https://www.tutorialspoint.com/sdlc/sdlc_software_prototyping.htm
[6] https://www.w3schools.in/sdlc-tutorial/spiral-model/
[7] https://www.geeksforgeeks.org/software-engineering-spiral-model/
[8] https://medium.com/@jeevanesh/software-development-life-cycle-phases-and-models-dc7b7aec3726
[9] https://www.javatpoint.com/software-engineering-sdlc-models
[10] https://stackify.com/what-is-sdlc/

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)