

1. SDLC Selection Model

Selection development life cycle is a process utilized by the software industry for designing, developing and testing high quality software. The SDLC intent to produce a high-quality software that meets or outreaches customer expectancy, that meets completion within times and also less costing.

2. OBJECTIVES

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality softwares. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

1. SDLC is the acronym of Software Development Life Cycle.
2. It is also called as Software Development Process.
3. SDLC is a framework defining tasks performed at each step in the software development process.
4. ISO/IEC 12207 is an international standard for software life-cycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software.

3. INTRODUCTION

SDLC provides a well-structured flow of phases that help an organization to quickly produce high-quality software which is well-tested and ready for production use. The SDLC involves six phases as explained in the introduction. Popular SDLC models include the waterfall model, spiral model, and Agile model .

4. PROCEDURE

Our Software Development Life Cycle consists of the following stages –

Stage 1: 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. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

Stage 2: 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.

Stage 3: 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. This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product. 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). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

Stage 4: 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 developed.

Stage 5: 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.

Stage 6: 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. 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.

5. IMPLEMENTATION

Selecting a Software Development Life Cycle (SDLC) methodology is a challenging task for many organizations and software engineers. What tends to make it challenging is the fact that few organizations know what are the criteria to use in selecting a methodology to add value to the organization. Fewer still understand that a methodology might apply to more than one Life Cycle Model. Before considering a framework for selecting a given SDLC methodology, we need to define the different types and illustrate the advantages and disadvantages of those models

How to select the right SDLC:

Selecting the right SDLC is a process in itself that the organization can implement internally or consult for. There are some steps to get the right selection.

STEP 1: Learn the about SDLC Models SDLCs are the same in their usage. In order to select the right SDLC, you should have enough experience and be familiar with the SDLCs that will be chosen and understand them correctly. As described in the software development life cycle models article, models are similar to the tools that important to know each tool usage to know which context it can fit into.

STEP 2: Assess the needs of Stakeholders

We must study the business domain, stakeholders concerns and requirements, business priorities, our technical capability and ability, and technology constraints to be able to choose the right SDLC against their selection criteria.

STEP 3: Define the criteria

Some of the selection criteria or arguments that you may use to select an SDLC are:

- Is the SDLC suitable for the size of our team and their skills?
- Is the SDLC suitable for the selected technology we use for implementing the solution?
- Is the SDLC suitable for client and stakeholders concerns and priorities?
- Is the SDLC suitable for the geographical situation (distributed team)?
- Is the SDLC suitable for the size and complexity of our software?
- Is the SDLC suitable for the type of projects we do?
- Is the SDLC suitable for our software engineering capability?
- Is the SDLC suitable for project risk and quality insurance?

Step 4 :

Our group's analysis :

The screenshot shows an Excel spreadsheet titled "Group - 06.xlsx" with a dark theme. The spreadsheet compares five software development models (Waterfall, V-shaped, Iterative, Spiral, Agile, and Prototype) based on seven criteria. The criteria are: 1. Well known requirement, 2. Technological knowledge, 3. Efficiency, 4. Risk analysis, 5. User testing, 6. Dependence and security, and 7. Required time. The models are evaluated as "Yes" or "No" for each criterion. A summary row at the bottom shows the total count of "Yes" responses for each model: Waterfall (12), V-shaped (16), Iterative (9), Spiral (5), Agile (9), and Prototype (9).

Priority	Criteria	Waterfall	V-shaped	Iterative	Spiral	Agile	Prototype
1	Well known requirement	Yes	No	No	No	No	Yes
2	Technological knowledge	Yes	Yes	Yes	No	No	Yes
3	Efficiency	No	Yes	No	Yes	No	No
4	Risk analysis	Yes	Yes	Yes	No	No	No
5	User testing	Yes	No	No	Yes	Yes	Yes
6	Dependence and security	No	No	No	No	Yes	No
7	Required time	Yes	Yes	No	No	Yes	No
Total - 30	Overall	12	16	9	5	9	9

6. TEST RESULT

We think that The Agile Model will be the most suitable model to select here. Agile derived from agility and flexibility to adapt to the new changes easily .

7. ANALYSIS AND DISCUSSION

1. During the first step of the agile software development life cycle, the team scopes out and prioritizes projects.
2. For each concept, we define the business opportunity and determine the time and work it'll take to complete the project.
3. Once we identified the project, we worked with stakeholders to determine requirements.