The Use and Adaptation of System Development Life Cycle in The Organization

Andrey S. Dragunov

Abstract: **This paper summarizes software development best practices and suggests how the System Development Life Cycle (SDLC) can be adapted for use in the organizations to increase the likelihood of successful development and delivery to end users. The key aspects and activities of the five stages of SDLC were discussed: planning, analysis, design, implementation and post-production. Key activities of development team and deliverables for each stage are described.**

Keywords— **System Development Life Cycle, software development, Plan-Driven methodologies, Agile methodologies.**

I. INTRODUCTION

The System (or Software) development is exited but the same time not intuitive process. Many software development teams are confused by relatively large projects and simply do not know where to start, when in fact the development process can be considered as a set of interrelated and dependent steps, with a full-functional system as a result. Another important issue that unfortunately, according to project management statistics about 70% of projects are fails [1]. There are number of reasons why project fails but undoubtedly the problem of understanding the developments process is on the top. No any tools or strong leadership can improve the situation in case of lack of understanding of how the development process should be organized. In this paper I summaries and clarify key steps that any project must come through.

II. THE SYSTEM DEVELOPMENT LIFE CYCLE

According to Alan Dennis [2] "The systems development life cycle (SDLC) is the process of determining how an information system can support business needs, by designing the system, building it, and delivering it to users".

Basically, it is easy to identify predominant five steps or phases in SDLC: Planning, Analysis, Design, Implementation and Post-Production.

1. Planning – the initial stage in which key business needs supposed to be identified and development methodologies are chosen. Originally, project plan supposed to be developed in planning phase but at this point due to lack of detailed business requirements only high-level plan may be developed. Later with deeper understanding of the systems problem domain and due to scope creep this high-level plan should be refined. Low-level planning should be done regularly depends on chosen development methodology.

2. Analysis – is the second, probably, the key phase is aimed on user (or business) requirements gathering and requirements elicitation. Both functional and nonfunctional requirements should be clearly understood at this moment. Also, functional, Structural and Behavioral models may be crated at this stage. At the end of Analysis phase System proposal must be prepared and approved by steering committee of organization.

3. Design – the third phase when business requirements supposed to be transformed in to form of system requirements. And architecture of the system including subsystems should be prepared.

4. The fourth stage – Implementation (or Development) phase should include writing the source code, testing and deployment or delivering to end-users.

5. Last fifth stage is Post Production phase includes maintenance. This is probably the longest phase in SDLC includes continues user training and improvements of the system.

Depends on the nature of the information system SDLC may be presented graphically in two forms as shown on Figure 1 and Figure 2.

Fig. 1. SDLC for systems with stable requirements. If business requirements changes dramatically new project should be kicked off.
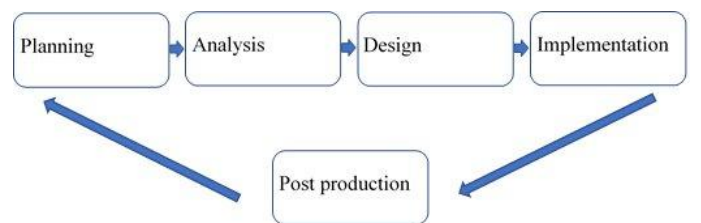

Fig. 2. SDLC for systems with unstable requirements. All changes in business requirements are considered just as a new iteration of SDLC.

Associate Professor, Inha University in Tashkent (IUT), Uzbekistan

The success or fails of project as a whole is based on first four phases: Planning. Analysis, Design and Implementation. Now let's pass through all steps and emphasize key moments on that attention should be paid.

III. PLANNING

All activities in SDLC are begins with System Request. System Request is formal or informal document that describes high-level business needs, business requirements and business values for proposed information system [3].

Generally, the ideas for new system may come from different sources most common variant is to use one of Business Project management strategies (BPA, BPI or BPI) [4].

There are number of ways may be chosen for System request acquisition. Three predominant are: "In-house development", "Packaged system" and "Rely on external vendor, outsourcing".

Based on System Request initial variant of high-level plan must be proposed.

Depends on the nature of the system one of Plan-Driven or Agile development methodologies may be chosen. But must common approach is to use a Plan-Driven methodology for high-level planning and Agile methodologies with a number of iterations (or sprints) for low-level plans as shown on Figure 3. As a Plan-Driven Waterfall or Phased development methodology may be chosen.
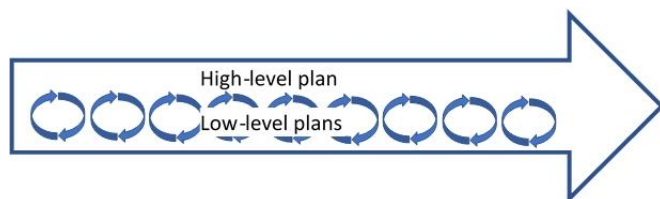

Fig 3 Plan-Driven methodology used for high-level planning and Agile methodologies for low-level planning.

One very important moment must be emphasized right now. I assume that Object Oriented Paradigm is considered as primary paradigm for project development and Iterative and Incremental approach is used. So, at this point of SDLC all plans are considered as very abstract plans that must be refined within better understanding of problem domain and moving through iterations. Low-level plans should be prepared based on Agile methodology chosen for low level planning, in case of SCRUM short term plans must be prepared for each sprint.

IV. ANALYSIS

Analysis phase may be considered as the most important phase of SDLC. During Analysis phase functional and nonfunctional requirements should be gathered and elicited. End-users must be involved in development process on all stages but exactly now trust and rapport must be built between development team and end-users.

Number of technics may be used for requirements gathering and elicitation:
• Interviews
• Document analysis
• Observations
• Joint Application Developments sessions
and many others.

Each of suggested technic has its own pros and cons and as common way is to use a combination of all technics for project.

The border between functional and non-functional requirements is very blurred the one possible way to distinguish functional and non-functional requirements is that functional requirements describes functionality of the system or information this system must operates then non-functional requirements describes characteristics of the system.

Functional requirements are divided in two categories: Process oriented and Information oriented.

Non-functional requirements are divided in four categories: Operational, Security, Performance and Cultural and Political.

At this point both functional and non-functional requirements are considered as initial business requirements and should be organized in form of requirements definition document, that is plain text document that contains both functional and non-functional requirements. Later all requirements supposed to be refined within better understanding of the problem domain.

Then initial business requirements are ready, the next step is to prepare functional, structural and if it is required behavioral models for proposed system.

Functional model shows functionality of the system, how system will operates with its environment. Functional models include Use-case diagrams, Activity diagrams and some others.

Structural model shows how underlined data should be organized. Predominant diagrams for structural model are class diagrams and Entity Relationships diagram.

For Class diagram development, textual analysis, brainstorming, common object lists and others technics may be used.

One of important activities at this stage is to balance Functional and Structural models. This may be organized with using walkthrough for example. At the end of Analysis phase System proposal should be created. The content of this document well described by IEEE organization [5]. Anyway, any format may be chosen. It should include System Request, Requirements definition, Functional model, Structural model and high-level Project Plan.

V. DESIGN

Next phase is aimed to increase the probability of successful implementation. In Design phase, Analysis Structural model should be evolved into Design Structural models. This may be done with using:

Factoring. Factoring is the process of separating out a module into a stand-alone module. The new module can be a new class or a new method.

Abstraction. Abstraction deals with the creation of a higher-level idea from a set of ideas.

Refinement. The refinement process is the opposite of the abstraction process.

Partitions and Collaborations. Based on all the factoring,

refining, and abstracting that can take place to the evolving system, the sheer size of the system representation can overload the user and the developer.

And some others.

Also, in Design phase architecture design and system specification should be prepared.

Architecture design should be based on both functional and non-functional requirements gathered in previous phase and includes the definition of a structured solutions to meet all requirements and next involves a set of decisions regarding software quality, maintainability, performance, and others.

System specification should include both hardware and software specifications.

VI. IMPLEMENTATION

Successful completion of the Design stage is a key factor in the success of the Implementation phase.

During Implementation phase key activities may be identified:

Translating the detailed requirements and design into system components

Testing the system using:
>   Unit tests
>   Integration tests
>   System tests
>   Acceptance tests

The user, with those responsible for quality assurance, validates that the functional requirements are met by the newly developed or modified system.

VII. POST PRODUCTION

Post Production phase is the longest phase in SDLC. It starts after system been developed, tested deployed and accepted by end-users. Somehow it relates to Continues Integration Continues Development philosophy but in assumption that major part of functionality is already done. So, in this case development team should concentrate on gathering feedbacks from end-user and improvements of the system. Some new requirements may be considered as new improvements and graphically Post Production Phase may looks like an endless series of short term SDLCs.

VIII. CONCLUSION

Five dependent and interrelated predominant phases (Planning, Analysis, Design Implementation and Post Production) are in the System Development Life Cycle. Key steps may be identified:

1. Develop/receive system request
2. Gather/elicit requirements
3. Prepare a plan Using both High level plan driven methodologies and low-level plan(s) with using agile methodologies
4. Understand functionality of to-be information system (functional modeling)
5. Develop Structure (Structure modeling)
6. Design and Implement the System
7. Maintenance

REFERENCES

[1] https://www.upwork.com/resources/project-failure-and-success-tips#:~:text=They're%20growing%20so%20quickly,are%20wasted%20and%20benefits%20unrealized.
[2] Alan Dennis "Systems analysis and design" / Alan Dennis, Barbara Haley Wixom, Roberta M. Roth.—Sixth edition. ISBN 978-1-118-89784-3
[3] Joseph Valacich "Modern Systems Analysis and Design" /Joseph Valacich, Joey George, Jeffrey Hoffer ISBN 0135172756
[4] https://businessprocessimprover.com/
[5] https://www.ieee.org/

Dragunov Andrey Sergeevich
−Associate Professor(Inha University in Tashkent, Uzbekistan)
PhD in physics and mathematics, Altay state technical university, Russia
MSc and BSc in Software engineering, Altay state technical university, Russia
TEACHING SUBJECTS
Internet Programming, Unix Programming, System Analysis, Computer Architecture, Application Programming in Java, Operating System