SURVEY ON SOFTWARE DEVELOPMENT PROCESSING MODELS

Nancy Bindal¹, Aanchal Mehta²

¹Assistant Professor, Department of CSE, Maharishi Markandeshwar University, Mullana, Ambala, Haryana, India
²M.Tech Research Scholar, Department of CSE, MMU, Mullana, Ambala, Haryana, India

ABSTRACT:
Software development process is concerned with production of software. Software cycles a process used by a systems analyst to develop an information system, including software productivity, cost, and overall quality.

Keywords:
Software Engineering, Life cycles models

I INTRODUCTION

Software engineering is the process of creating new systems. Some methodologies use to develop systems. The concept generally refers to software product and information systems. In software engineering concept underpins many kinds of software development methodologies. These methodologies work on framework of system for planning and controlling software system. Software development process is concerned with production of software. Software cycle is a process used by a systems analyst to develop an information system, including software productivity, cost, and overall quality. Software cycles are representation of a software process. It presents some particular perspective. A software cycle is either prescriptive or descriptive. Software cycles should deal with system design, creation and maintenance. Software cycles work on strategies to produce a software system. A software life cycle models a standardized format for planning organizing, and running [1]

II SOFTWARE LIFE CYCLE MODELS

1. System development life cycle
2. Waterfall model.
3. Iterative model.
4. V-shaped model.
5. Spiral model.
6. Incremental Method
7. prototyping model
8. The Big Bang Model.
9. System development life cycle

III SDLC:

Any information system project follows a logical series of development phases these are known as system development life cycle. SDLC is process creating and altering systems.

Planning

The goal of planning is to costs, personnel requirements and quality assurance of the software developed. one must know what the problem is before it can be solved. The analyst first task is to prepare state next problem then reviews it with the users for accuracy.
Analysis

is a detail study of the various operation performed by a system & their relationships with inside & outside of the system during analysis data is collected on the available files ,decisions points transaction handled by present system .analysis gathers requirements for the system . This stage given detailed study of the needs of the organization.

Design

The term design describe of find system the process by which it is develop it refers to the technical specification that with be applied in implementing the candidate system it also include the construction of program & program testing.

Development

Development is concerned with the production of software. The covers a wide range of methodologies and tools created for the purpose. The general approaches for organizing a project into activities. This phase supported management of software development.

INTEGRATION AND TESTING

This phase is different unit are integrated .integration testing is performed after units testing. the verifying consistency and completeness of modules. Check the validating the performance of the system.

Implementation

it is concern with user trainer side preparation , file conversion when the candidate system is link to terminals or remote sides the telecommunication network & test of the network along with system are also included under implementation.

Maintenance

System definitely undergo changed it when delivered to the customer. Change could happen because of some unexpected input values into the system. Changes in the system could affect the software operations. Maintance is continued to bring the new system to standard.

Advantage

1. SDLC cycle end of each stage allowing maximum management control.
2. Progress of this system is measurable.
3. SDLC documentation produces the system requirements can be traced back to stated business
4. SDLC requires sign-off requirements.

Disadvantage

1. SDLC is Time consuming.
2. SDLC never uses never backward loop.
3. Software that follows the SDLC program is it encourages stiff implementation instead of creativity.
4. They gave large difficulty responding to changes [3]

IV WATERFALL MODEL

The waterfall model the common and classic life cycle model which introduced by Winston Royce. In Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially. It referred as a linear –sequential life cycle model. The project is right path whether or not to continue then discard software.

Following is a diagrammatic representation of different phases of waterfall model
The sequential phases in Waterfall model are:

**Requirement**

Software is always a large system work with requirements for all systems elements the software is essential when software interface with such elements are hardware people and resources the system is re-engineered and spruced up. The requirement team studies the methodologies of the system.

**Design**

In defined the design includes use age of existing components. They represent “how “phase programming languages, application, architecture, algorithms, structures are established. Design describes to implement the requirements. Design phase could be expansive to solve later stage of the software system.

**Implementation**

The code is generated, the program testing begins. This phase deals with issues of quality, performance, and debuggers. They use testing tools that are made own development operations implementation phase overlap testing phase act as a system whole.

**Verification**

All the units developed entire system tested for any faults and failures. Validating their operation and specifications software is distinguishing attribute of system indicating the degree of excellence. The product is verification in the customer environment and market.

**Maintenance**

There are some short issues are come client environment. To fix some issues patches are released. Also to en some better versions are released. Maintenance is done when changes into customer environment. Maintenance reaches with changing accommodate process. All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the define set of goals are achieved for previous phase and it is signed off, so the name “Waterfall Model”. [4]

**Advantages**

- This model is simple and easy to use.
- This model work on a series of stage.
- In this model phases are completed one at a time. Phases do not overlap.
- Waterfall model works well and understood smaller projects.

**Disadvantages**

- In this waterfall approach High amounts of risk and uncertainty.
- No working software is produced for complex problems.
- The model is difficult to change once are prepared
- Inflexible partitioning held in software during life cycle.[5]

**V ITERATIVE MODEL:**

Iterative model is any combination of both iterative design or iterative method .iterative process describe software process cycles of
activates of software. The key idea behind this cycles to solve problem through iterations. Each iterative identify some problem and solve these issues. This method is developing a system through repeated cycles and in smaller part of time. Iterative involves redesign and implementation to be simple, straightforward, and modular, supporting that stage or task added to project control list.

**Requirement**
In this phase software are analyzed. The result given requirements phase produces a complete and final specification of requirements. The specification is process establishing what services are required for software.

**Design & Development**
In design solution for software is designed which meets requirements they extension of the existing. Describes features and operations in detail, including screen layouts, rules, process diagrams, pseudo code and Documentation. The development phase refers to developed software generation. The first pass is initial development cycle. Life of product stops existing software evolve next generation by sequence of inception, elaboration, construction and transition phase develop a software.

**Implementation**
The software is implemented for solving problems. A phase, is software is evaluated, current requirements are reviewed, and changes additions to requirements propose. Decision made whether software will be discarded.

**Testing**
Even point will be reached when the requirements are complete and the software can be delivered the testing is done to software is ready according to the requirements of the users. [6]

**Advantages**
1. The model accurate reflects iterative nature of software’s on project with unclear requirements.
2. It is flexible as it incorporates advantage of waterfall and rapid prototyping methods,
3. Good project visibility in the project.
4. The model is a comprehensive and decreases risk in software.

**Disadvantages**
1. It needs technical expert to solve risk analysis.
2. The model is not understood by non–technical management.
3. In this model each phase is rigid with no overlaps.
4. This model is a complicated model that needs competent professional management. [7]

**VI V-SHAPED MODEL**

V- Model means Verification and Validation model. It called V-shaped represent like waterfall model, this cycle is a sequential path of execution of processes. Each phase must complete before next phase begins. Testing emphasize more than waterfall model. Testing procedure is developed in cycle prior any coding done .each phase preceding implementation. It modified U-shaped model.

**Diagram of V-model:**
The various phases of the V-model are as follows:

**Requirements**

BRS and SRS begin cycle model like waterfall model. In this model before development is started, a system test planning is created. The test plan focuses on meeting the functionality specified in the requirements gathering.

**The high-level design (HLD)**

Phase focuses a software architecture and design. It provides overview of solution, platform, software and service. Any plan is created this phase as well order to test pieces of the software systems ability to work together.

**The low-level design (LLD)**

Phase in which actual software components are designed. It defines the logic for each and every component of the software Class diagram with all methods and relation comes under LLD. Component tests are created in phase as well.

**Implementation:**

Implementation phase where all coding takes place. Once coding is complete, the path of execution continues up right side of the V where the test plans developed now put to use.

**Coding**

This is the bottom of V-Shape model. Module design is converted into code by developers.

**Testing**

In this phase different unit are integrated than overall system is tested .testing is performed all units of components.[8]

**Advantage**

1. It is very simple and easy to use.
2. Each phase this model have specific deliverables.
3. It works welled understood for small projects where requirements are easily handled.
4. high chances for success there than other models due to development of test plans during the cycle process.

**Disadvantage**

1. This model is very rigid like other models.
2. Flexibility and adjusting scope is difficult and much more expansive.
3. Model not provide clear path for problems that are found on the testing phase
4. The prototypes of software are produced .because developed started in implementation phase.[9]

**VII SPIRAL MODEL**

The spiral model is same to incremental model. This is a risk-driven process model generator for software’s. Based upon unique risk patterns of a given for software .In development process represented as spiral rather a sequence of activities
with backtracking each loop. In which no fixed phases for as specification and design loops in the process. A software repeatedly passes through phases in iterations called spirals in this model. This model combines adding features of prototyping and waterfall model and is advantages for large, complex, and expensive software’s.

Planning Phase

All objective are gathered during this phase. Special requirements are identified this phase. Each phase of quadrant come with identifying the goals for that process, it determines alternatives, they are possible accomplishing those goals.

Risk Analysis:

The process is undertaken identify risk and alternate solutions. A prototype is produced at end of risk analysis phase. If any risk found during risk analysis then alternate solutions are implemented. Development process of the software depends on remaining risks. Quadrant develops the final software while considering risks can be occurring. Risk analysis considers time and effort to be devoted to each phase activity such as planning, quality assurance, verification, and testing.

Engineering Phase

Software is developed in this phase, along with testing point at the end of phase. Hence this phase development and testing is done. A processing phase plans next step and planning for next prototype thus, comprises development plan, integration plan, and test plan.

Evaluation phase

This phase allows customer to evaluate output of the software to date before the processing continues to the next spiral. This model is each phases are completed by a review conducted by users. This includes review of all intermediate software, which developed during the phases. Plan for next evaluation and resources required for this phases. [10]

Advantages

- They produced high amount of risk analysis hence, avoidance of Risk enhanced.
- They Good for large and-critical software’s.
- Additional requirements are also added at later date.
- Strong approval and documentation control in this model.

Disadvantages

- They Can costly evaluation model to use.
- Risk analysis require highly specific expertise to solve major problems.
- Spiral model doesn’t specified for small software.
- software success is highly dependent on risk analysis phase.[11]

VIII INCREMENTAL MODEL

This model applies linear sequences staggered progress applied. In incremental model whole process is divided into various builds. Multiple development phases take place here, making the cycle a “multi-level” cycle. Cycles are divided up into smaller, more easily managed modules. Each module go through different phases such as requirements, design, implementation and testing phases. Software should break into separate sub phases. Each sub phases are released of the module add function to previous release. The
process continues till complete software is achieved. The software as finished when it satisfies all its requirements. This allows partial utilization of software and avoids a long time. This model helps ease traumatic effect introducing a completely new product all at once[12].

Requirements and analysis
Incremental model usually a short list of requirements that need to be defined and then quickly converted into processing model provides a Requirement Management module that allows create and track requirements. Whether Releases software increments can easily organized your requirements in the way that give works best for environment. And analysis the software modules.

Design
Due to requirement set and design cycle quickly build cases from the requirements so that increment component can easily delivered on time to the client when software design is prepared. allows make easily design That provides functional coverage for design module that you can quickly review As progress and new increments are created design phase is time to assess impact and risk new features link and create and improve development of model.

Implementation
Incremental methodology is generally test and debug in implementation phase. relatively smaller changes are made during each increment. This allows targeted and rigorous testing of each element within the overall software. Using the Incremental methodology even after first increment expectation product is ready. This means all phases must be tested and passed. This phase can link Test Cases to Requirements to establish coverage; however, can also use simple approach of testing and all Test Cases directly passed.

Testing
After each increment, regression testing be conducted. During this testing, faulty elements of software can be quickly identified few changes are see single increment. In testing against specific Test Cases , Test Case folders Test Suites can be used for specific new features. Test cases are executed against different Platforms or Configurations. Test cases can be copied quickly from one cycle to the next for testing process.

Advantages
- This model Generates working software quickly during software life cycle.
- This model is more flexible, less costly to change scope and requirements.
- It is easier to test and debug during a smaller changing process.
- This model easy to manage risk, they risky piece identified quickly.

Disadvantage
- This model Needs good planning, structure and design. Needs a clear and
complete processing of the whole system before broken down and built incrementally.

- In this model iteration processing is very rigid and they don’t overlap.
- This model arise problem in architecture in processing all requirements not gathered up.[13]

**IX PROTOTYPE MODEL**

The prototype model is developing a small prototype and followed by mini-waterfall process. First to gather requirements, then, first prototype is reviewed in subsequent loops. The team performs further requirements, design, implementations and review. The prototyping approach used requirement gathering in the analysis phase to capture the exact requirement of proposed system. After the requirements are frozen, the remaining phases of the development process needs to executed a complete development of software system. Prototyping is the process of quickly putting together a working software order to test various aspects of design, ideas or features and gathered user feedback. The prototyping serves to clarify models,

**Build prototype**

Information collect from design is modified to form first prototype, which represents the working model of the required software.

**Engineer product**

the requirements are completely, they accepts final prototype. The final system is evaluated thoroughly maintenance and for preventing large-scale failures and minimizing downtime.

**Refining prototype**

the user evaluates prototype and not satisfied, the current prototype is refined to requirements. a new prototype is developed with information provided by the user. The new prototype is evaluated same as previous prototype. This process continues until all the requirements specified by the user. user is satisfied with the new prototype, a final product is developed on the basis of final prototype.

**Customer evaluation**

the proposed system is presented by user for evaluation prototype to checks strengths and weaknesses such as something is to added or removed. Comments and suggestions are collected from the evolution and provide by the user.

**Advantages**

- This model is very useful and give good result for gathering requirements.
- This model Helps in reducing risks link with software.
- Errors are detected faster.
- Provides a model the process, enabling assessment and increasing user’s confidence.

**Disadvantages**

**Requirements**

A model begins with requirements and requirements of system are defined and accurate in details. The user is interviewee give order to know the requirements of the software.

**Quick design**

requirements are done, a quick design for the software is created. It is not a write design and includes only the important operation of system, which gives an idea of the user. A quick design helps in developing the model.
1. Leads implementing and repairing way of building software.

2. this methodology may increase complexity of the software may expand beyond plans.

3. Incomplete application may cause not used full software was designed
4. this model is very time consuming and expensive [14]

X BIG BANG MODEL

Big bang model focus all possible resources in software. This model is ideal for small projects with one or two developers working together. also useful for academic projects. The Big- Bang Model like a cosmological model it named after namely one which we put huge amounts of matter are put together, the outcome is either perfect software or not. The beauty of model is simple There is made with little planning, scheduling, or formal development process. All the effort of this model is spent developing and writing code. This model approach by small projects, since the developer does requirement analysis, writes code and develops software. Also developers need to understand requirements and develop the software, there is no formal testing phase this model, if testing occurs before release final product. The lack of a formal testing phase is cause for massive failure ratio this Model[15]

Advantage

- This is a simple and good learning model.
- Little and no planning required this model
- Easy to manage and easy to create in some time.
- Very few resources required and not heavy work load

Disadvantage

- Very High risk and uncertainty.
- Can move to very expensive if requirements are misunderstood
- Not a good model complex and object-oriented projects.
- Poor model are more required for long and ongoing projects.[16]

CONCLUSION

IN processing life cycle models Decide development Process They Work on model Strategies. In this paper various life cycles studied like SDLC, Waterfall model., Iterative model, V-shaped model, Spiral model, Incremental Method, prototyping model ,The Big Bang Model. We overview Different models methodologies and defined advantages and disadvantages.

REFERENCES

3. Software Development Life Cycle (SDLC), Power Point, – Powered by Google Docs


13. Dewar, Robert B. K.; Fisher Jr., Gerald A.; Schonberg, Edmond; Froelich, Robert; Bryant, Stephen; Goss, Clinton F.; Burke, Michael (November 1980). "The NYU Ada Translator and Interpreter"


AUTHOR'S BIBLIOGRAPHY

Nancy Bindal received his B.Tech. degree in Information Technology Engineering from JIET, Jind, Haryana in 2011 and M.Tech in CSE from R.N.C.E.T, Madlauda, Panipat, Haryana in 2014 , Currently working as Assistant Professor in MMEC, Mullana, Haryana.

Aanchal mehta had done B.TECH (CSE) From Shri sukhmani Institute of Engineering and Technology Derabassi and Pursuing M.TECH (CSE) From Maharishi Markandeshwar University Mullana in India.