Sr No	Experiment List
1	To identify the traditional process models and Problem Definition of case study name
2	To Preparation of software requirement specification (SRS) Document in IEEE format.
3	To create Gantt chart/Time line chart for selected case study
4	To create structured data flow analysis. (DFD)
5	Use of metrics to estimate the cost.
6	To draw the class diagram for selected case study with any open source (Dia software).
7	To design test cases of selected case study.
8	To design test cases for white box testing. (Basisi path testing)
9	To prepare Risk Mitigation, Monitoring and Management Plan (RMMM).

Subject Incharge

Prof. Pankaj Deshmukh

Experiment No: 01

Aim: To identify the traditional process models and Problem Definition of case study name.

Objective:

After completion of this experiment you will be able to:

1. Understand different software process models.

2. Understand how to identify and analyse software process model for selected project, for solving any real world complex engineering application.

Outcome:

1. Ability to write and understand the detailed advantages and disadvantages of different software process models with examples.
2. Able to identify type of software process model for application.

Theory:

S/W Process Model:

Categories:

Linear Sequential

Incremental

Evolutionary

The various phases of Software Process model are:-

- Waterfall model
- RAD model
- Incremental model
- Evolutionary model

Prototyping

Spiral Model

Concurrent Development Model

1. Waterfall model

This is also called Classic Life Cycle Model (or) Linear Sequential Model (or)

Waterfall Method. This model has the following activities.

2. **System/Information Engineering and Modeling**

As software development is large process so work begins by establishing requirements for all system elements and then allocating some subset of these requirements to software. The view of this system is necessary when software must interface with other elements such as hardware, people and other resources.

3. **Software Requirement Analysis**

Software Requirement Analysis is also known as feasibility study. In this requirement analysis phase, the development team visits the customer and studies their system requirement. They examine the need for possible software automation in the given software system.

4. System analysis and design

In System Analysis and Design phase, the whole software development process, the overall software structure and its outlay are defined. In case of the client/server processing technology, the number of tiers required for the package architecture, the database design, the data structure design etc are all defined in this phase

5. Code Generation

In Code Generation phase, the design must be decoded into a machine-readable form. If the design of software product is done in a detailed manner, code generation can be achieved without much complication.

6. Testing

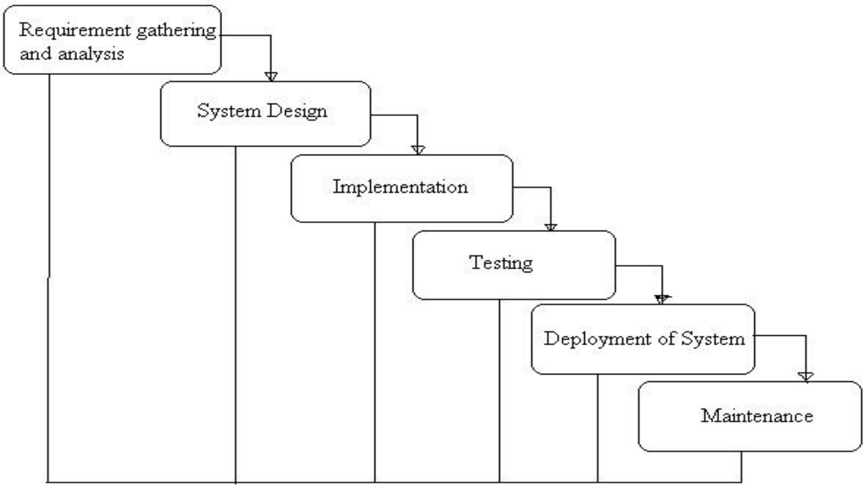
After code generation phase the software program testing begins. Different testing methods are available to detect the bugs that were committed during the previous phases. A number of testing tools and methods are already available for testing purpose.

7. Maintenance

Software will definitely go through change once when it is delivered to the customer. There are large numbers of reasons for the change. Change could happen due to some unpredicted input

values into the system. In addition to this the changes in the system directly have an effect

General Overview of "Waterfall Model"



software operations.

Advantages:

- The waterfall model is the oldest and most widely used model in the field of software development. There are certain advantages of the waterfall model, which causes it to be the most widely used model as yet. Some of them can be listed as under.

The amount of resources required to implement this model is minimal.

- One great advantage of the waterfall model is that documentation is produced at every stage of the waterfall model development. This makes the understanding of the product designing procedure simpler.
- After every major stage of software coding, testing is done to check the correct running of the code.

Disadvantages of Waterfall Model

There are some disadvantages of this widely accepted model too. Let us look at a few of them.

1. Many times, it happens that the client is not very clear of what he exactly wants from the software. Any changes that he mentions in between may cause a lot of confusion.
2. Small changes or errors that arise in the completed software may cause a lot of problem.
3. The greatest disadvantage of the waterfall model is that until the final stage of the development cycle is complete, a working model of the software does not lie in the hands of the client.

Examples:-

2. Rapid Application Development:

Rapid application development (RAD) is a software development methodology that uses minimal planning in favor of rapid prototyping. The "planning" of software developed using RAD is interleaved with writing the software itself. The lack of extensive pre-planning generally allows software to be written much faster, and makes it easier to change requirements.

Rapid application development is a software development methodology that involves methods like iterative development and software prototyping. According to Whitten (2004), it is a merger of various structured techniques, especially data-driven Information Engineering, with prototyping techniques to accelerate software systems development.

Four phases of RAD

Business Modeling: The information flow among business functions is defined by answering questions like what information drives the business process, what information is generated, who generates it, where does the information go, who process it and so on.

Data Modeling: The information collected from business modeling is refined into a set of data objects (entities) that are needed to support the business. The attributes (character of each entity) are identified and the relation between these data objects (entities) is defined.

Process Modeling: The data objects defined in the data modeling phase are transformed to achieve the information flow necessary to implement a business function. Processing descriptions are created for adding, modifying, deleting or retrieving a data object.

Application Generation: Automated tools are used to facilitate construction of the software; even they use the 4th GL techniques.

Testing and Turn over: Many of the programming components have already been tested since RAD emphasises reuse. This reduces overall testing time. But new components must be tested and all interfaces must be fully exercised.

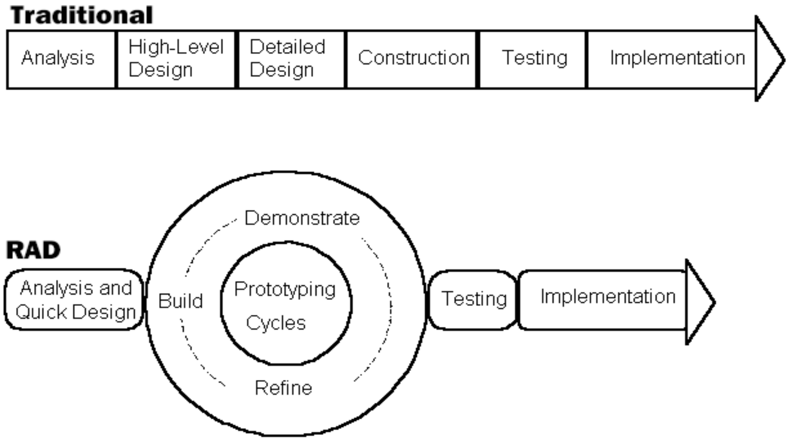


Fig. 1 Traditional v/s RAD Cycle

Advantages :

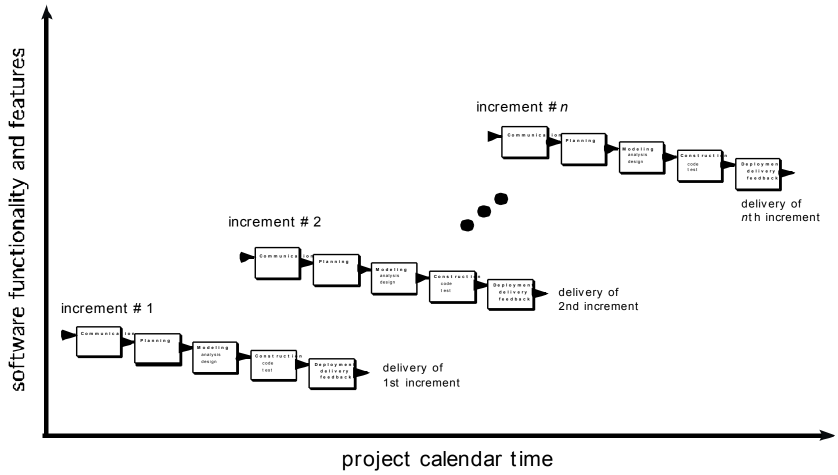
1. Flexible and adaptable to changes
2. Can handle large projects without a doubt
3. RAD realizes an overall reduction in project risk
4. RAD realizes an overall reduction in project risk.

Generally Rad incorporates short development cycles

Disadvantages :

1. Cant use for small projects
2. Requires more resources and money to implement RAD
3. All applications are not competible with RAD
4. RAD is not appropriate hen technical risks are high
5. Need both customer and developer commitments to complete a project. Otherwise RAD will fail

3. Incremental Model

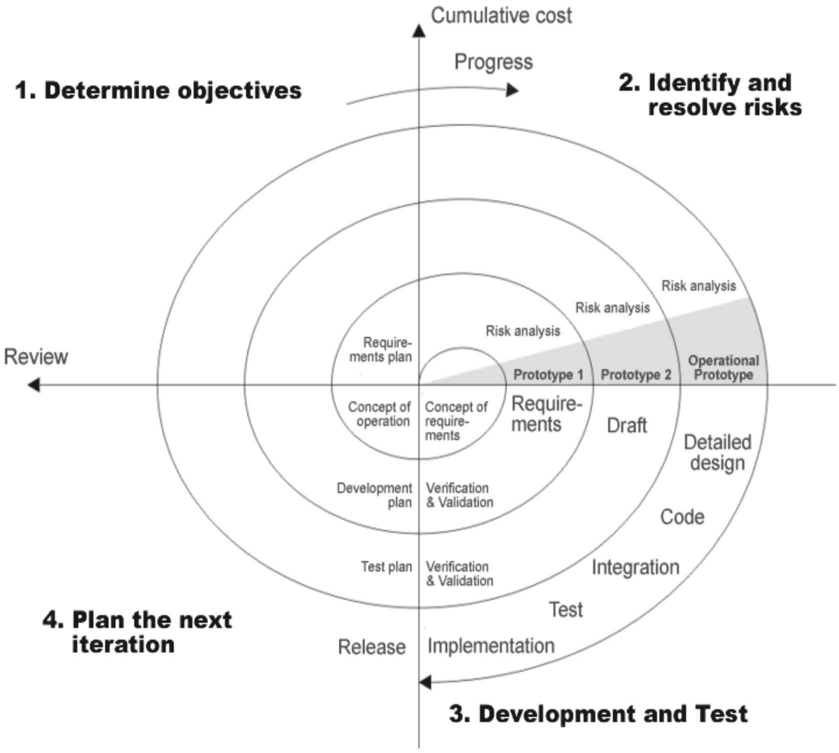

software functionality and features
increment # n
Communication
Planning
Modeling analysis design
Construction code test
Deployment delivery feedback
delivery of nth increment
increment # 2
delivery of 2nd increment
increment # 1
delivery of 1st increment
project calendar time

Evolutionary Models:-

3. Spiral Model

The spiral model is a software development process combining elements of both design and prototyping-in-stages, in an effort to combine advantages of top-down and bottom-up concepts. Also known as the spiral lifecycle model (or spiral development), it is a systems development method (SDM) used in information technology (IT). This model of development combines the features of the prototyping and the waterfall model. The spiral model is intended for large, expensive and complicated projects.

The spiral model combines the idea of iterative development (prototyping) with the systematic, controlled aspects of the waterfall model. It allows for incremental releases of the product, or incremental refinement through each time around the spiral. The spiral model also explicitly includes risk management within software development. Identifying major risks, both technical and managerial, and determining how to lessen the risk helps keep the software development process under control.



Advantages: :

1. Estimates (i.e. budget, schedule, etc.) become more realistic as work progresses
2. It is more able to cope with the (nearly inevitable) changes that software development.
3. Software engineers (who can get restless with protracted design processes) can get their hands in and start working on a project earlier

Disadvantages: :

1. Highly customized limiting re-usability.
2. Applied differently for each application.
3. Risk of not meeting budget or schedule.
4. Risk of not meeting budget or schedule.

S/w model used for ur project. Justify it.

- DSP-

  Write detailed statement of case study selected as Sentences explaining whole system in detail. Interaction between system entities
- List all entities present in DSP
- List all attributes of each entity
- List all functions of each entity
- List all interactions /relationships between all entities as statements

Conclusion: In this way software process models are discussed sucessfully.DSP of the selected case study is created as a document datewise. Identified & documented no. of entities attributes, operations and relationships.

Experiment No: 02

Aim: To Preparation of software requirement specification (SRS) Document in IEEE format.