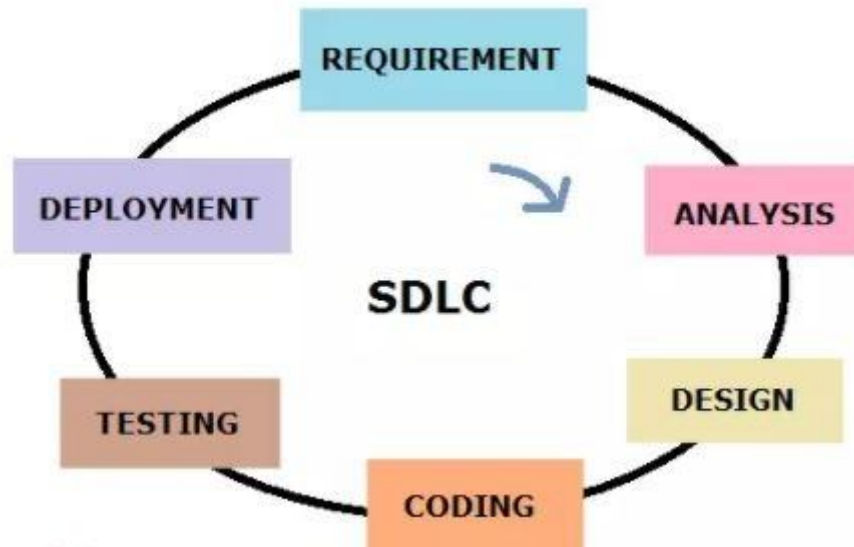


Case Study on SDLC Models

Introduction to Software Engineering:- Software Development Life Cycle, or SDLC is a process used to develop software. There are different stages or phases within the Software Development Life Cycle and in each phase, different activities take place.



SDLC Different Phases

1. Requirement and Analysis:

Software Development Life Cycle begins with Requirement Analysis phase, where the stakeholders discuss the requirements of the software that needs to be developed to achieve a goal.

Required Document is BRS (Business Requirement

Specification) SRS (Software Requirement Specification)

2. Design

The next stage of Software Development Life Cycle is the Design phase. During the design phase, developers and technical architects start the high-level design of the software and system to be able to deliver each requirement.

The technical details of the design is discussed with the stakeholders and various parameters such as risks, technologies to be used, capability of the team, project constraints, time and budget are reviewed and then the best design approach is selected for the product.

The selected architectural design, defines all the components that needs to be developed, communications with third party services, user flows and database communications as well as front-end representations and behaviour of each components.

The design is usually kept in the Design Specification Document (DSD)

SDLC Different Phases

3. Implementation :

After the requirements and design activity is completed, the next phase of the Software Development Life Cycle is the implementation or development of the software. In this phase, developers start coding according to the requirements and the design discussed in previous phases.

Database admins create the necessary data in the database, front-end developers create the necessary interfaces and GUI to interact with the back-end all based on guidelines and procedures defined by the company.

Developers also write unit tests for each component to test the new code that they have written, review each other's code, create builds and deploy software to an environment. This cycle of development is repeated until the requirements are met.

Documents required are Source Code Test Specification and Test cases

SDLC Different Phases

4. Testing :

Testing is the last phase of the Software Development Life Cycle before the software is delivered to customers. During testing, experienced testers start to test the system against the requirements.

The testers aim to find defects within the system as well as verifying whether the application behaves as expected and according to what was documented in the requirements analysis phase.

Documents required are Test Result

5. Deployment and Maintenance

Once the software has been fully tested and no [high priority issues](#) remain in the software, it is time to deploy to production where customers can use the system.

Once a version of the software is released to production, there is usually a maintenance team that look after any post-production issues.

If an issue is encountered in the production the development team is informed and depending on how severe the issue is, it might either require a hot-fix which is created and shipped in a short period of time or if not very severe, it can wait until the next version of the software.

Required Documents

are Check list for

Deployment Version

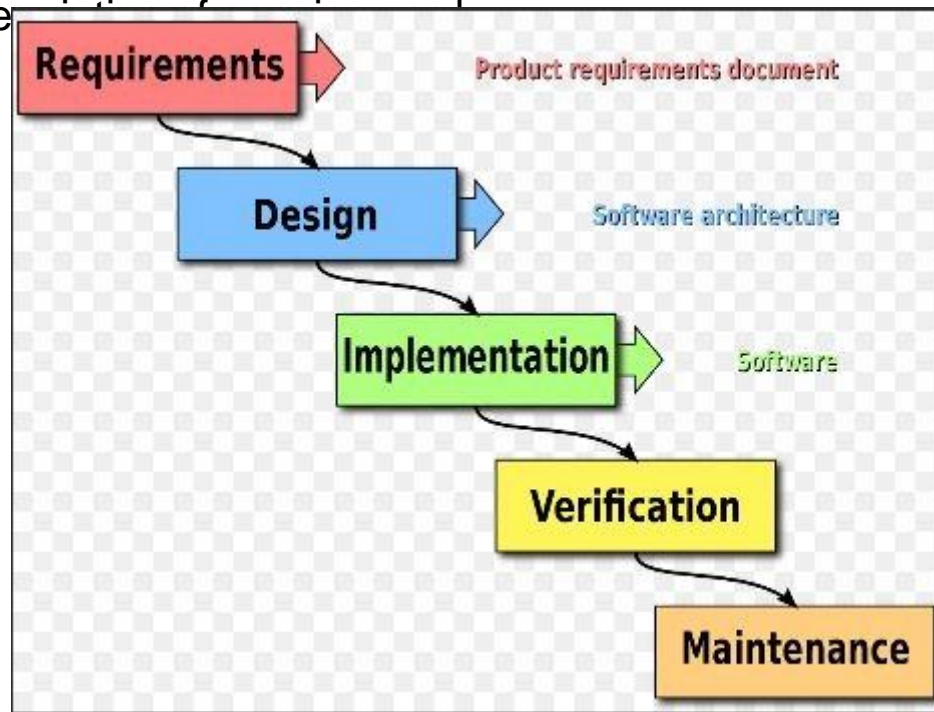
Management

All the respective documents to be revised as per maintenance enhancement

Software Development Life Cycle (SDLC) Models

Water Fall Model:- **Waterfall Model** illustrates the software development process in a linear sequential flow; hence it is also referred to as a **Linear-Sequential Life Cycle Model**.

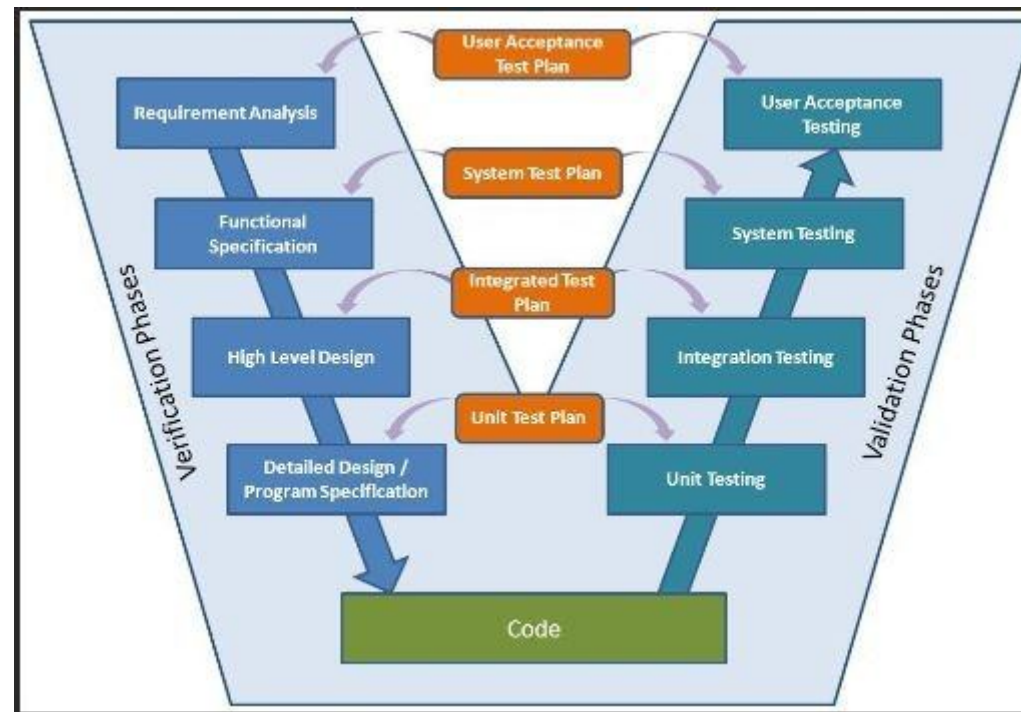
Example -> Online



Software Development Life Cycle (SDLC)

Models

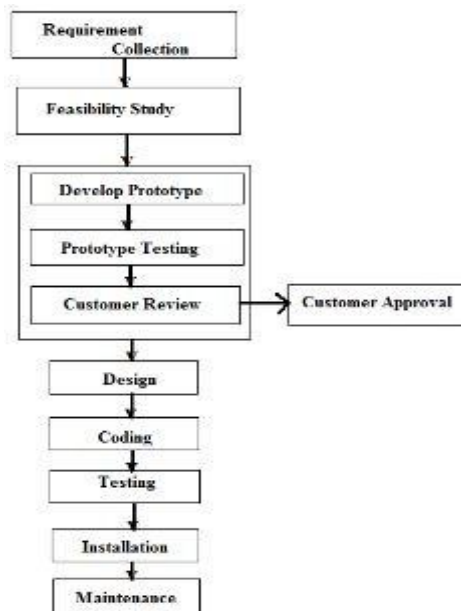
V Model :- The **V-Model** is **SDLC** model where execution of processes happens in a sequential manner in V-shape. *It is also known as **Verification and Validation Model**.*



Software Development Life Cycle (SDLC) Models

- Prototype Model:

Prototype Model is an early sample, model, or release of a product built to test a concept or process. It is a term used in a variety of contexts, including semantics, design, electronics, and software programming. **A prototype is designed to test and try a new design to enhance precision by system analysts and users.**



Software Development Life Cycle (SDLC) Models

- Incremental Model:

The **Incremental Model** is a method of **software development** where the product is *designed, implemented and tested* incrementally. Little more is added each time until the product is finished. It involves both development and maintenance.

This

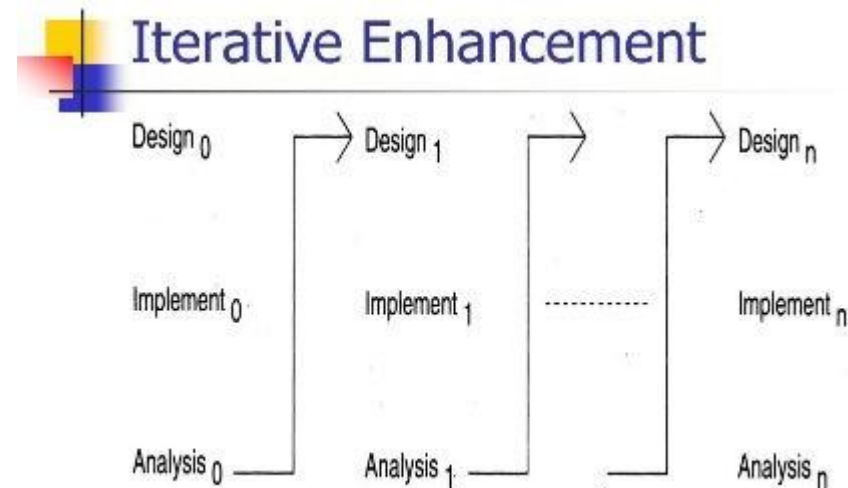
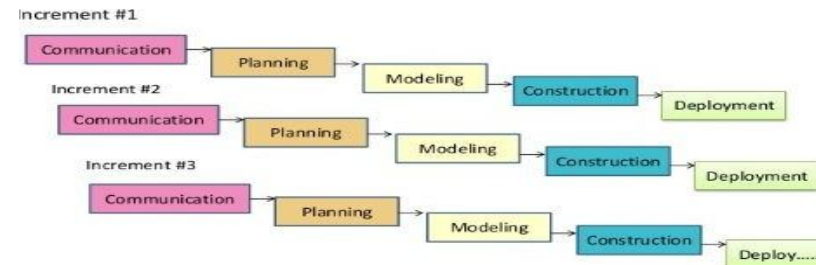
is also known as **Iterative Model**.

Example – Income tax return portal (e-return then assessment, refund module added)

- Iterative Enhancement Models

In the first step of iterative enhancement model, a simple initial implementation is done for a subset of the overall problem. This subset is the one that contains some of the key aspects of the problem which are easy to understand and implement, and which forms a useful and usable system. A project control list is created which contains, in an order, all the tasks that must be performed to obtain the final implementation. This project control list gives an idea of how far the project is at any given step from the final system.

Incremental Model (Diagram)

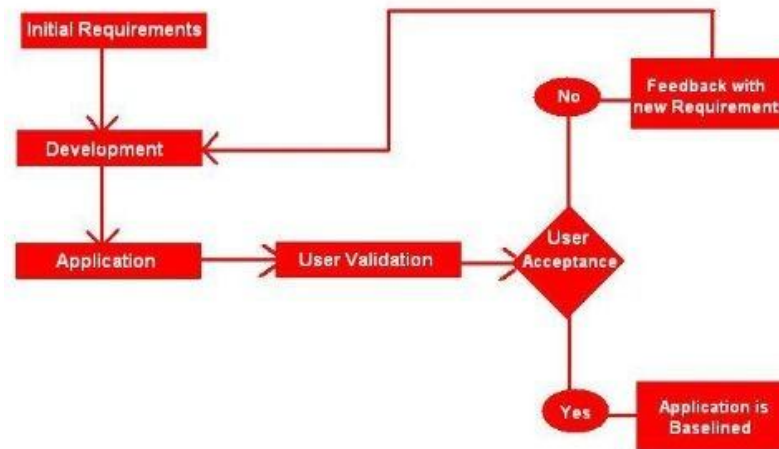


Software Development Life Cycle (SDLC) Models

- Evolutionary Development Models:

Evolutionary development of more appropriate for Development of a relatively small system. The problems of the existing system changes avoided by re-implementing the entire system whenever a significant change is needed. If modeling is used, not too expensive. Development of a system that has a relatively short.

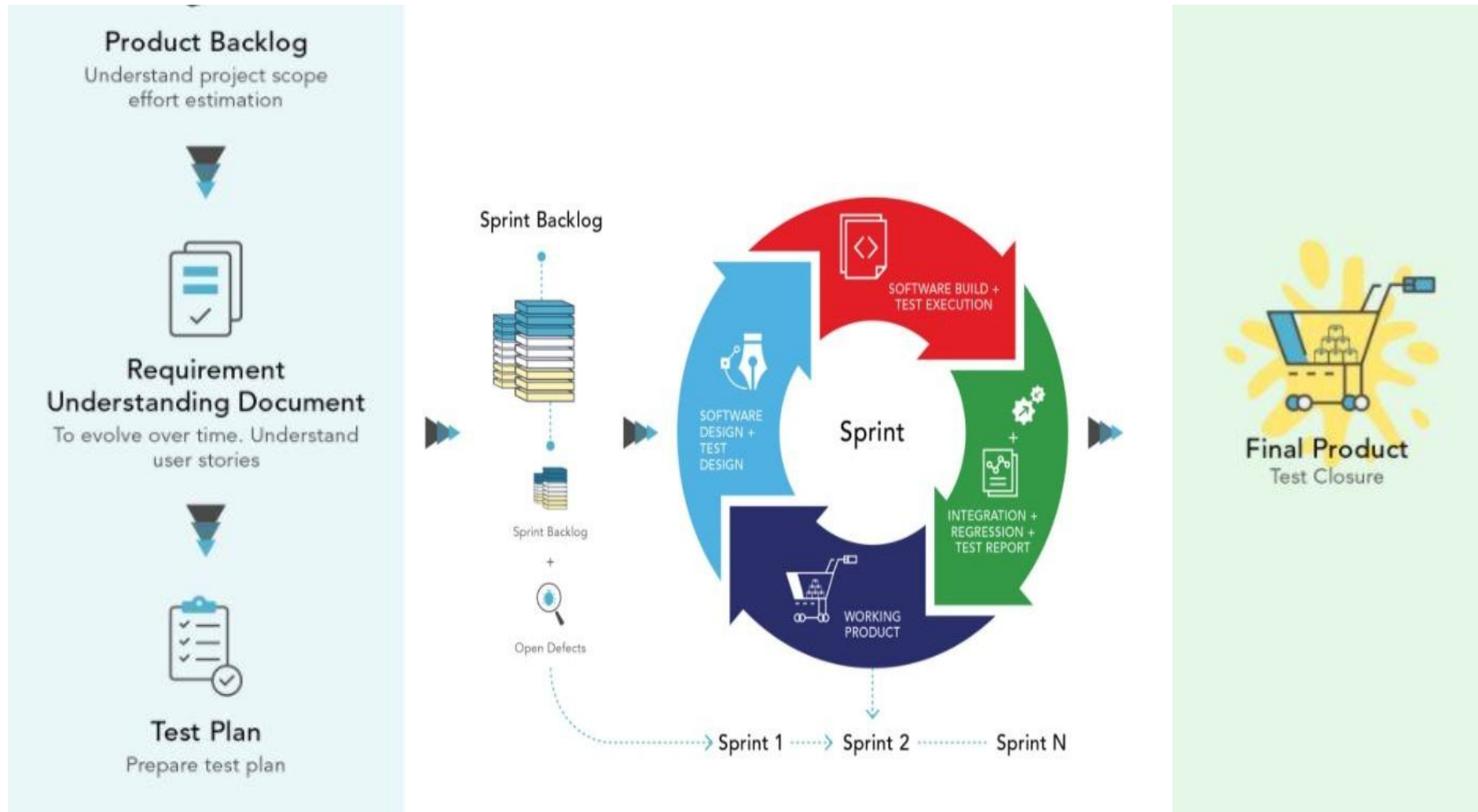
Evolutionary Model



Agile Model

- Agile Model
- Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.
- Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

Agile Technique- Sprint cycle

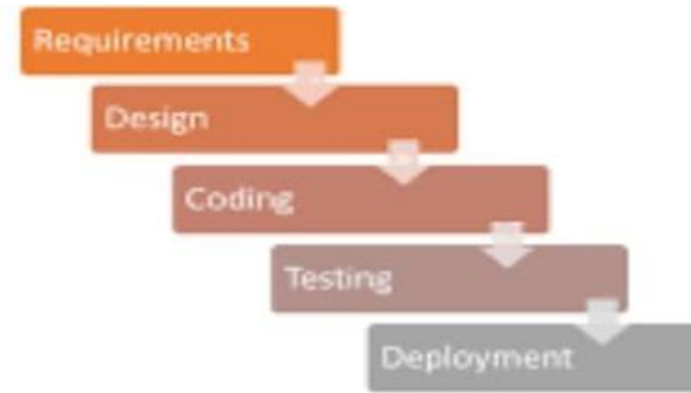


Difference between Agile and traditional Methodology

One main **difference between** the **traditional** and **agile methodologies** is the sequence of the phases in which the software development project is completed. The **traditional** method uses a linear **approach** where the stages of the software development process must be completed **in a sequential order**. The agile software development method uses an iterative and team-based approach

Traditional methodology

. Waterfall



Agile methodology

.Scrum....

.Crystal....

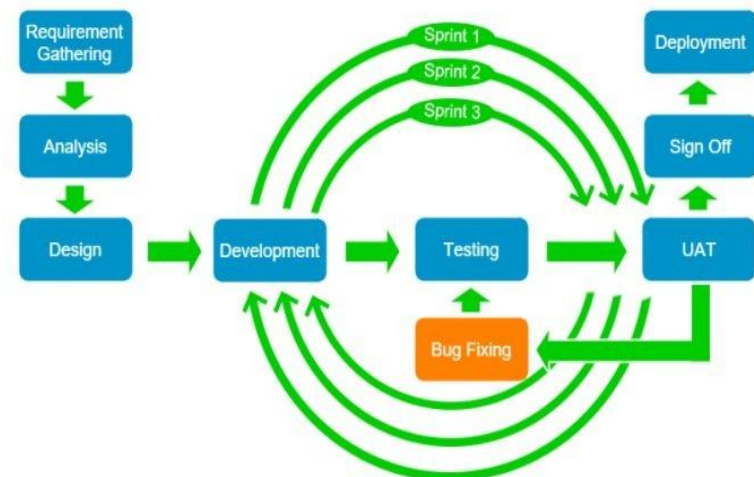
.FDD. ...

.DSDM. ...

.Lean Software Development.

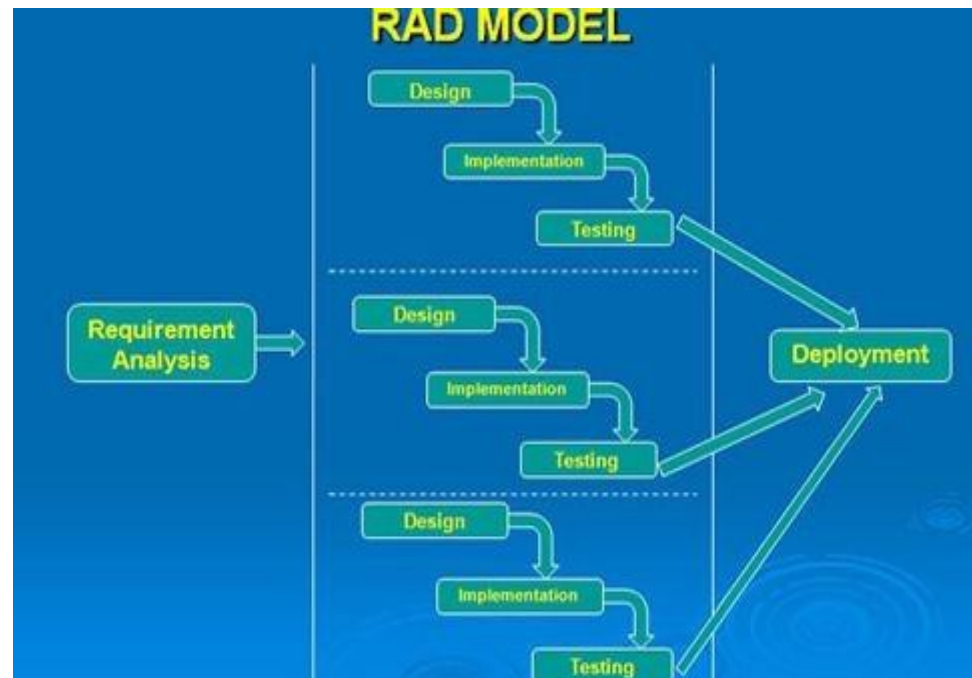
.XP

.xBreed



- **RAD Model:**

- *Rapid Application Development (RAD)* is a software development method that uses minimal planning in favor of rapid prototyping. The functional modules are developed simultaneously as prototypes and are integrated to make the complete product for faster product delivery.



• Big Bang Model

- The Big Bang model does not follow any particular process and customer also not sure about his requirement. There is only little formal development process. The entire effort is spent software developing and coding. Development teams are small and hence this model is usually followed for small projects

Part 1: Introduction to Software Engineering

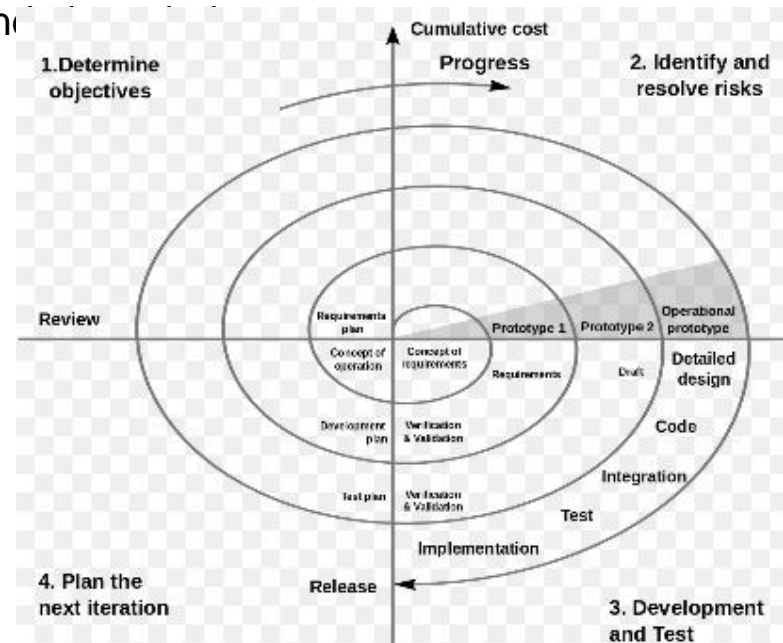


1. Big-Bang Model :

- The Approach
People+Money+Energy=Perfect Software or Nothing!
- Advantages
Simple, needs less planning, less formal development process
- Disadvantages
No formal testing activities, hard to manage and monitor, risky

• Spiral Model

The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around



Case Study for Uberization Project – In-plant Vehicle movement

Business Requirement:- Need to minimize the use of vehicle in the plant area. Usage of the vehicle is for movement of logistics and to give a critical support in un-planned shut down. Need online booking of vehicles as mobile app

Proposed solution -> A mobile app created and data stored on cloud. Technology used is J2ME and oracle and Google app used to know the location and shortest path between source and destination. OLA kind of app prepared for users along with phone booking facility

SDLC Model adapted:- RAD

Other applicable models -> Prototype , agile, Iterative Enhancement Models, spiral

How Development Done:- we have prepared a small app with limited scope only with booking facility for few user segment and for only one department.

Once successfully done then added other modules like cancel booking, booking in advance critical booking, bill generation, and then deployed the same for all location and all department

Case Study for Rack monitoring

Business Requirement:- Need to create a GPS tracker which can be plotted on the railways racks to maximize the utilization and monitor the same

Proposed solution -> Cloud based web application created and data stored on cloud. Technology used is Java Hibernate and oracle app server for monitoring also a GPS created separately by R & D Division.

SDLC Model adapted:-

Prototype Other applicable models -> RAD

How Development Done:- we have prepared a small web app and take only one GPS as a trail and monitor its activity to know the rack position between a particular range

Once successfully done then full project implemented

Case Study to print a online visitor diary after retirement

Business Requirement:- Need to create visitor diary for retired person and on submission of I-card and other belongings , system will auto issue a printed visitor diary to enter in the office premises

Proposed solution -> VB dot net based a web app has been created with oracle database with a two level approval and printing of diary

SDLC Model adapted:- waterfall

Other applicable models -> V Model, Evolutionary Development Models:

How Development Done:- we have prepared a small web app and deploy it on one department for one month and then deploy it for all department

Case Study for Online display of OEE for all Plants

Business Requirement:- Need to create dashboard which will display the OEE for all plants on monthly basis and trends for entire year and provide a facility to display the loop holes for reduced OEE(Overall equipment effectiveness)

Proposed solution -> we have prepared reports for one plant in Tableau desktop tool which was freeware and then apply the same on licensed version and dis it for all plants

SDLC Model adapted:- Agile scrum methodology Other applicable models -> RAD,

How Development Done:- we have prepared a small web app and deploy it on one department for one month and then deploy it for all department