Chapter 7 :

Informatics Practices

Class XII (As per CBSE Board)

New Syllabus 2019-20

Software engineering-software processes and delivery models

Visit : python.mykvs.in for regular updates

Software engineering

Introduction :

Software engineering is an engineering branch where main focus is given over the development of software product using well defined scientific principal, procedures and methods.

SE is the process of analyzing user needs, designing, constructing, and testing end user applications that will satisfy the needs of the user through the use of software programming language(s).As compared to simple programming, software engineering is used for larger and more complex software systems, which are used as critical systems for business or for any organization.

Software engineering

Introduction :

The term software engineer is most often confused with programmer. Both work on vastly different disciplines.

Any programmer’s main task is to write source code that makes a program run, whereas a software engineer is responsible for designing, developing and implementing the software solutions.

Thus Software Engineering covers wider aspect and larger domain as compared to programming.

Programming is a component/part of software engineering.

Software engineering

Software processes :

A software process is the process of dividing software development work into different phases so that software design is improved and product/project is easily managed. It is also known as a software development process or software development life cycle.

It is a kind of structure imposed on the development of a software product. There is no ideal software process. It depends upon the product , the organization where it is being developed and the user for whom it is going to be developed.

Software engineering

Software processes :

Common fundamental activities of all software processes:

- Software specification - The functionality of the software and constraints on its operation are defined under this activity.
- Software design and implementation – Designing of software to meet the specification of software.
- Software validation - The software must be validated to meet the customer needs.
- Software evolution - The software must be developed in such manner that it can meet changing customer needs.

Software engineering

Software Process Models :

It is an abstract representation of a software process.

Common software process models are

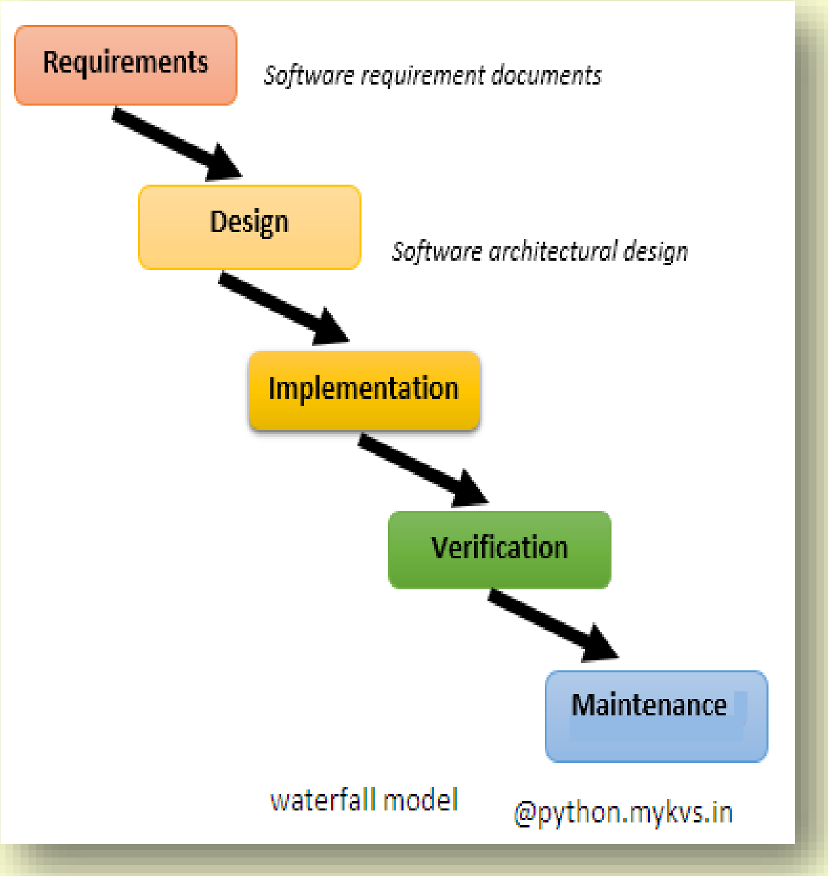
1. **Waterfall model**
2. **Evolutionary model**
3. **Component based model**

Software engineering

Software Process Models :

1. Waterfall model

It is also known as a linear-sequential life cycle model. It is easy to use. In a waterfall model , there are number of phases and each phase must be completed before the next phase begins and there must not be any overlapping in the phases.



Software engineering

Software Process Models :

1. Waterfall model

The sequential phases in Waterfall model are −

Requirement analysis − Requirement specification document must be prepared based on all possible requirement of software.

System Design − defining the overall system architecture based on the requirements recorded in requirement specification document.Most of software success depends on this phase.

Implementation − Based on system design small programs as units are developed and later on these are integrated to work like a software

Verification − Testing of the software to check whether it fulfills user requirements.

Maintenance − Some issues may arries at client environment to be resolved are as a part of maintenance.

Software engineering

Software Process Models :

1. Waterfall model

Situations to use/apply waterfall model

- Clear and fixed requirements.
- Stable problem definition.
- Technology is static.
- No ambiguous requirements.
- The project is small.

Software engineering

Software Process Models :

1. Waterfall model

Advantages of waterfall model

- ➢Easy to understand
- ➢Easy to arrange task
- ➢Clearly defined stages
- ➢Easy to manage
- ➢Well understood milestones

Disadvantages of waterfall model

- ➢ Poor model for long project
- ➢ Cannot fulfill changing requirement
- ➢ No working software is developed till last phase
- ➢ Dificult to measure the progress in phases

Software engineering

Software Process Models :

2. Evolutionary model

The evolutionary model is like "Incremental Waterfall Model" ,where development cycle is divided into smaller cycles , in this model users are able to get access to the product at the end of each cycle.

The users provide feedback on the product for planning to the development team for changing the product , plans or process.

These incremental cycles are typically of two or four weeks in duration and continue until the product is fully developed.

Software Process Models :

Advantages of Evolutionary Model

- Error reduction: because versions are tested at each incremental cycle
- User satisfaction: Users are given full chance of experimenting partially developed system.
- High quality: Quality is maintained due to thoroughly testing.
- Low risk: There is significant reduction of risk as a versions is implemented.
- Reduction Cost: It reduces cost by providing structured and disciplined steps.

Disadvantages of Evolutionary Model:-

- Multiple versions: Developer has to make table of different versions developed.
- Difficult to Divide software: It is difficult to "divide the software and the problems in several versions.
- Uncertain customer needs: A confused user has uncertainty over his requirements
- Time And Cost:As this model reduces "Time And Cost" but requirement is not gathered correctly,may later on effect over time and cost.

Software engineering

Software Process Models :

Following are the evolutionary process models.

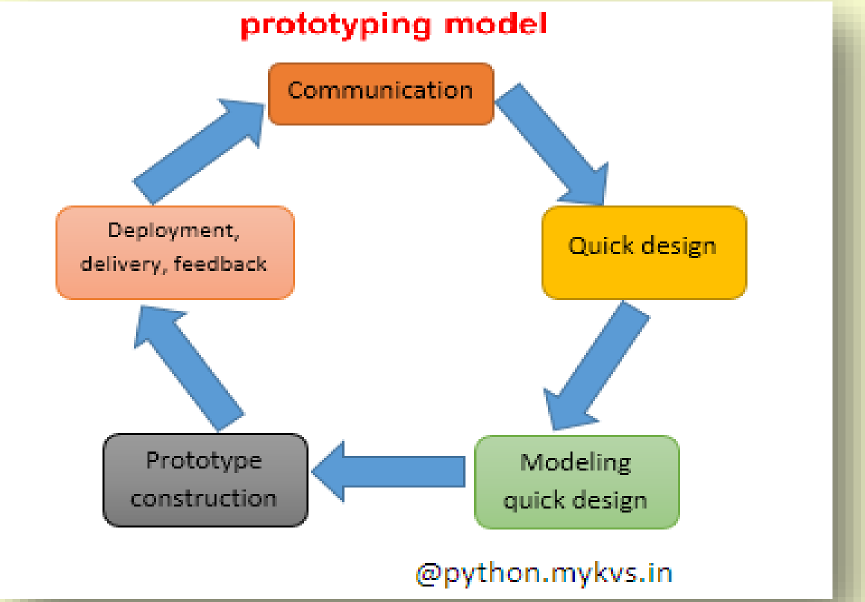
a. The prototyping model

b. The spiral model

c. Concurrent development model

a. The prototyping model

It is software working model of limited functionality. In this model, working programs are quickly produced.



Software engineering

Software Process Models :

a. The prototyping model

1.Communication- between developer and customer to discuss the overall objectives of the software.

2.Quick design- of known requirements

3.Modeling quick design- for clear idea about the development of software

4.Construction of prototype- to be evaluated by the customer itself.

5.Deployment, delivery, feedback- of developed software

Software Process Models :

a. Situations to use prototyping model

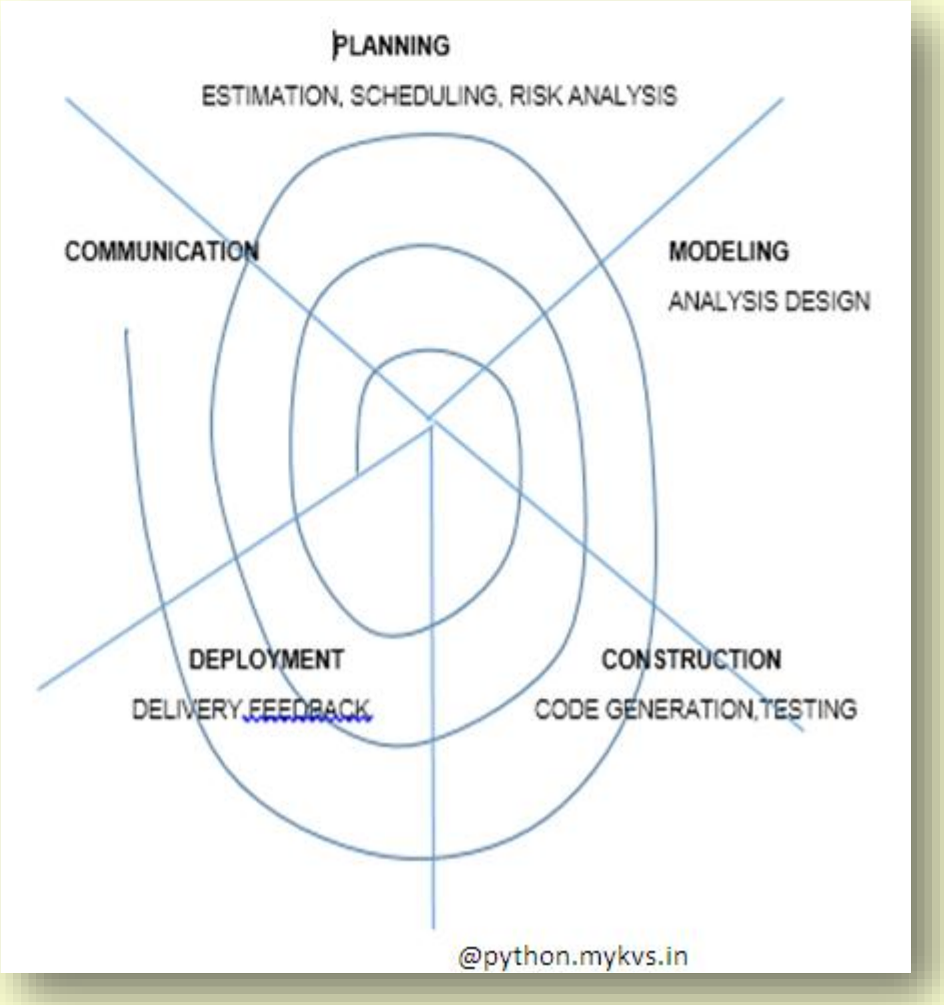
- ➢ when the desired system needs to have a lot of interaction with the end users.
- ➢ Typically, online systems, web interfaces where there are very high amount of interaction with end users, are best suited for Prototype model.
- ➢ In Prototype model the end users constantly work with the system and provide a feedback which is incorporated in the system.

Software engineering

Software Process Models :

b. The spiral model

It is a combination of prototype & sequential model or waterfall model. All activities are done in one iteration .Software development is done in the phases like planning, modeling, construction, deployment,communication in spiral/continuous manner.These phases are repeated till fully functional software is not developed.



Visit : python.mykvs.in for regular updates

Software Process Models :

b. Situations to use spiral model

- Large and high budget projects
- When risk assessment is very critical
- Requirements are not very clearly defined.
- Requirements are vague and even complex
- The organization does not have much experience with the domain.
- Ample time is available.

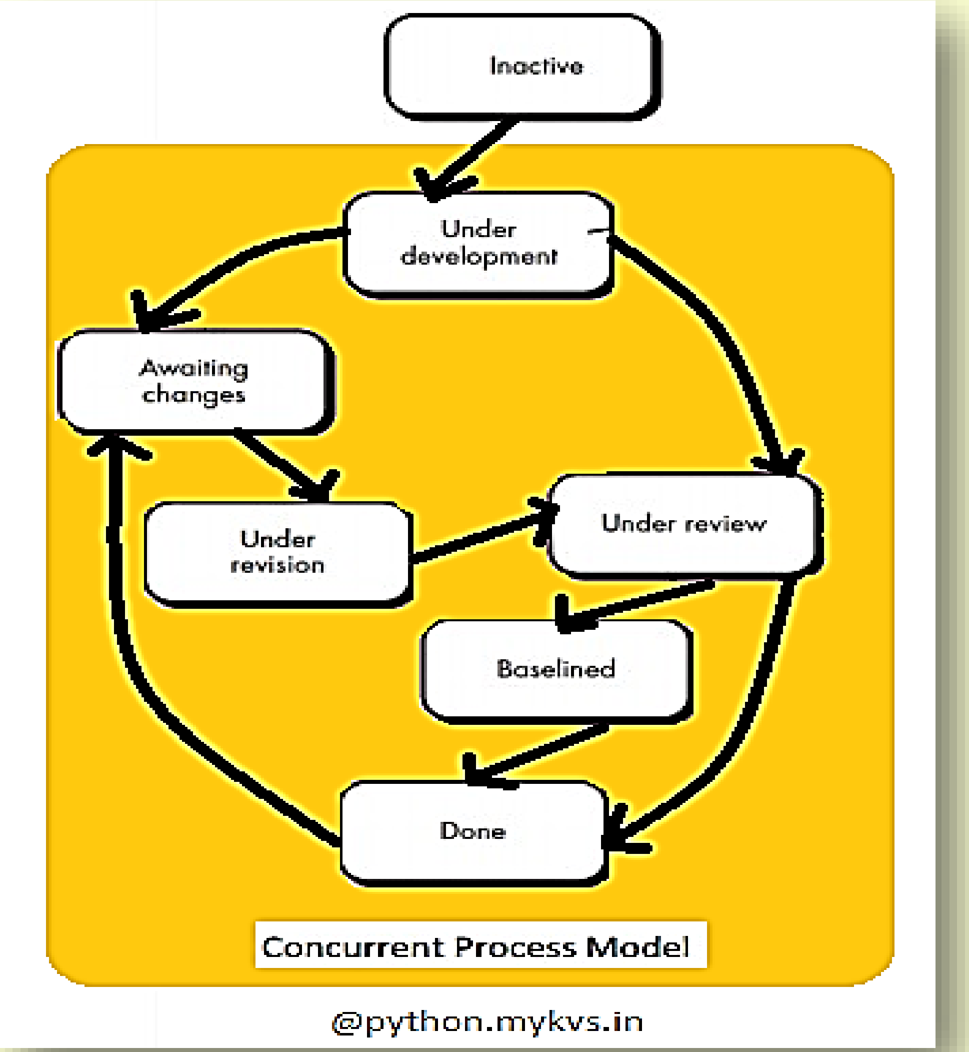
Software engineering

Software Process Models :

c. The Concurrent development model

This model defines a series of events that will trigger transition from state to state for each software engineering activities. During early stages of design, an inconsistency in the analysis model is procured. This will generate the event analysis model correction, which will trigger the analysis activity from the done state into the awaiting Changes State.

This model is suitable for the development of client/server applications.



Concurrent Process Model

Software Process Models :

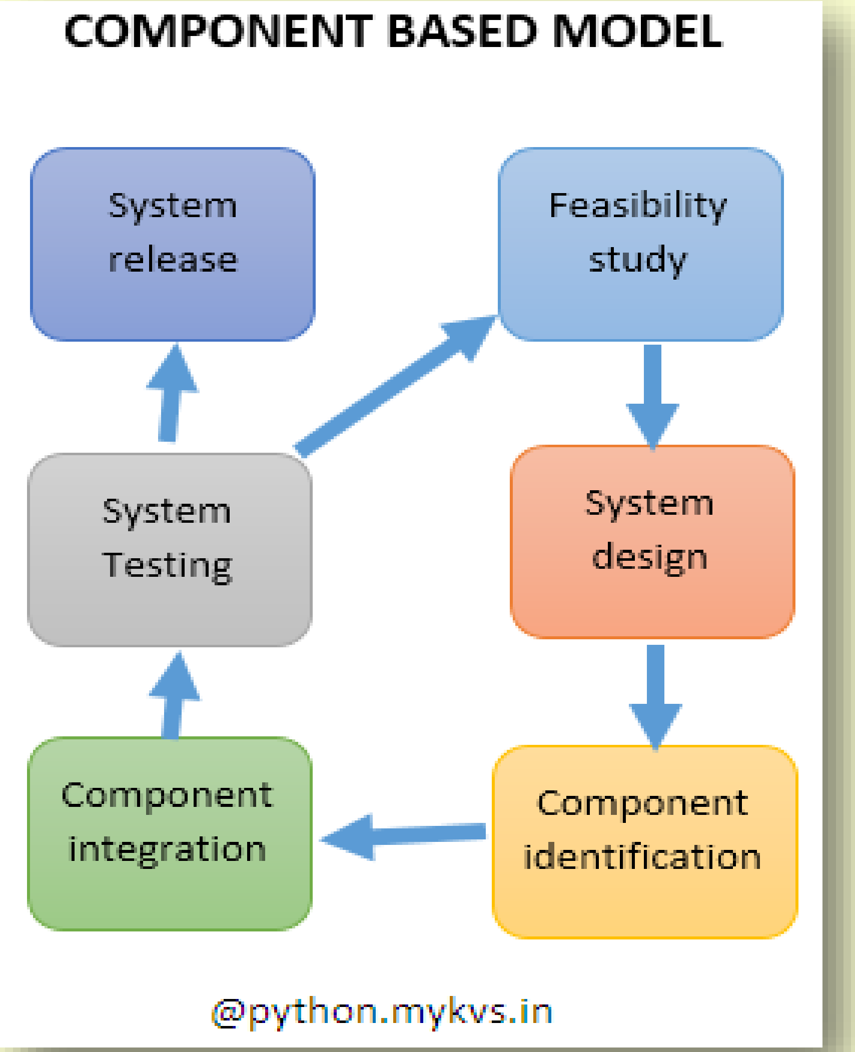
c. Situations to use Concurrent development model

- When incremental release to be determined by project team
- When immediate need of feedback after testing
- More features to be added later on in the project
- When there is need of continuous testing

Software engineering

Software Process Models :
3. Component Based Model

Software products are built using components, and these components are developed through a number of different sources, be written in several different programming languages. Software development gone through the steps like feasibility study, system design, component identification, component integration, system testing and system release.



Visit : python.mykvs.in for regular updates

Software Process Models :

3. Situations to use Component Based Model

- When the emphasize is given to buy components not to build components
- When the software structure is lower in coupling
- When the development process is evolutional and concurrent
- When the focus is assembling the software components not develope

Software engineering

Software Process Models :

3. Component Based Model

Advantages

- Reduce the cost & risk of software development.
- Reduce the amount of software to be developed.
- Faster delivery of software.

Disadvantages

- Requirement changes effect the software development.
- Control over the system evolution is lost

Software engineering

Delivery models :

Software Delivery is the process of getting a software product to market or to the client.

Software delivery includes the following actions:

- Client requirements finding
- Technology for software development and deployment
- Testing and validation of software for quality

A software delivery model is a term which is widely used in the IT industry. It is defined as a way of project delivery based on the location of client/user. The choice of a delivery model can affect the success of the entire project.

We will go through two types of delivery models

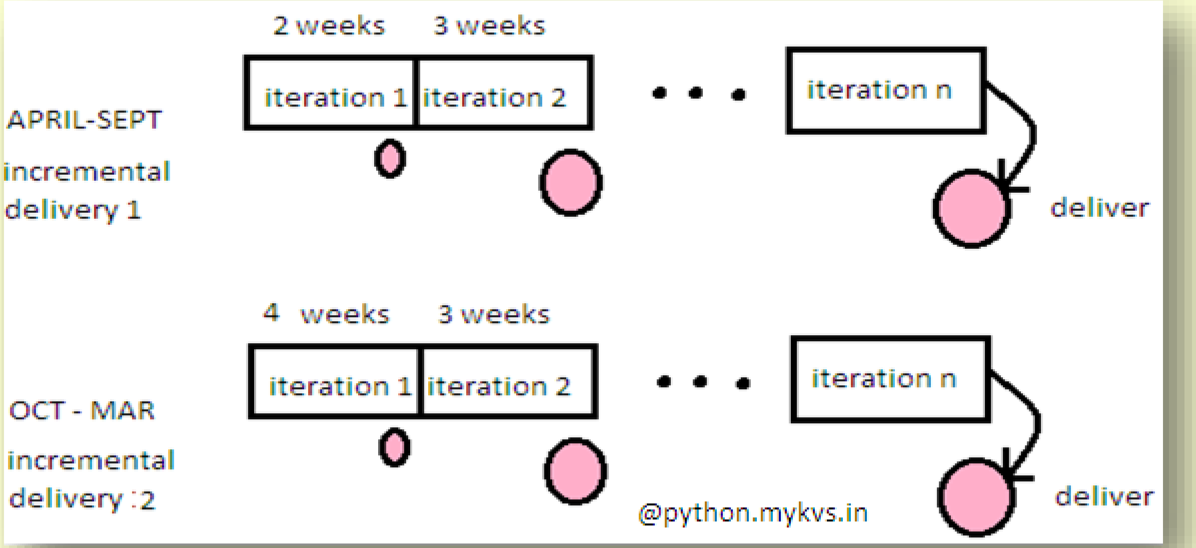
- **incremental delivery model**
- **spiral delivery model**

Software engineering

- **incremental delivery model**

Incremental delivery model refers to the practice of repeatedly delivering a system into client environment (or to the marketplace) in a series of expanding capabilities. Incremental deliveries are often between three and twelve months.

Incremental delivery model is often confused with iterative development method. A six-month delivery cycle could be composed of 10 short iterations. The results of each iteration are not delivered to the marketplace, but the results of an incremental delivery are delivered to the client.

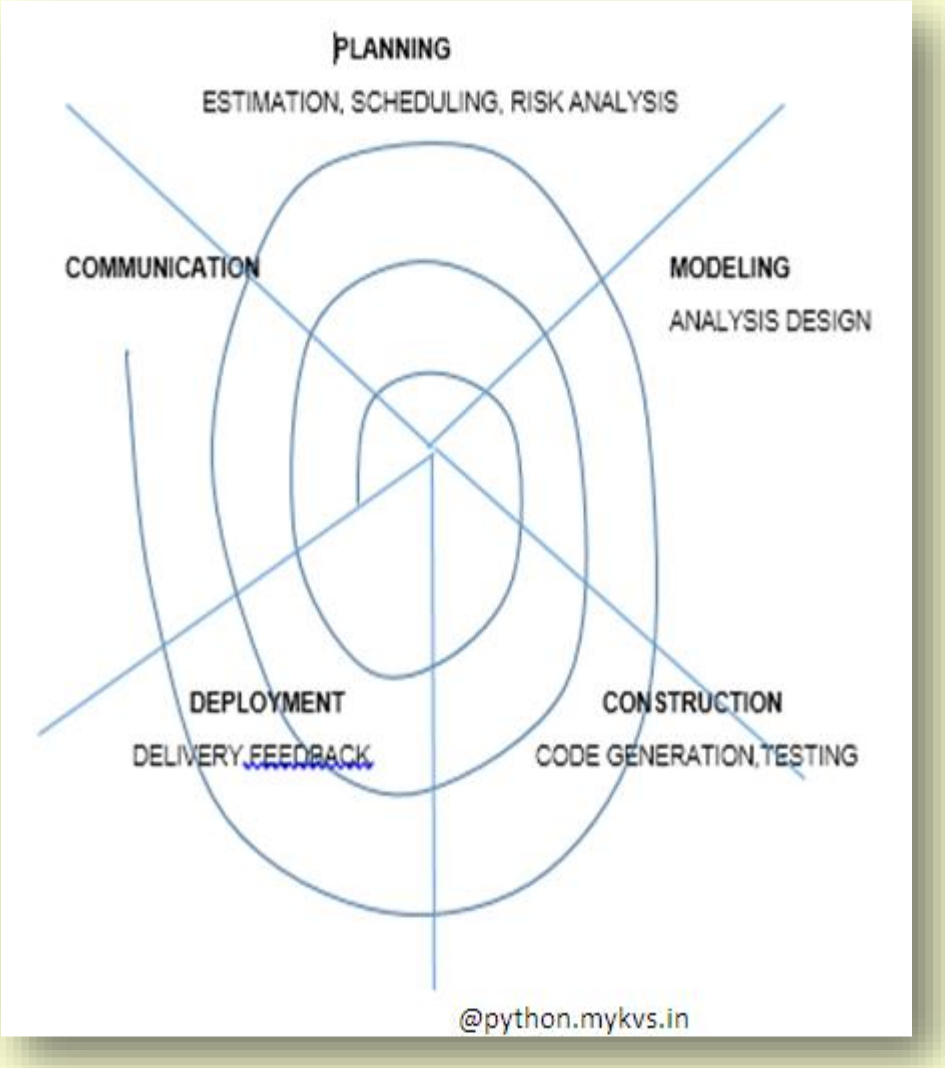


Software engineering

spiral delivery model

Spiral delivery model can be pretty costly to use and doesn’t work well for small projects. It’s a risk-driven model which means that the overall success of a project highly depends on the risks analysis phase. Risk analysis requires to review and analyze the project from time to time

Strong side of this model is that there’s a possibility to add some additional functionality at the last stages of software product development.Software is repeatedly delivered after each refinement.


PLANNING
ESTIMATION, SCHEDULING, RISK ANALYSIS
COMMUNICATION
MODELING
ANALYSIS DESIGN
DEPLOYMENT
DELIVERY,FEEDBACK
CONSTRUCTION
CODE GENERATION,TESTING
@python.mykvs.in