

SOFTWARE ENGINEERING

The background features a gradient from dark blue at the top to light blue at the bottom. Overlaid on this are several wavy, horizontal bands of color. From top to bottom, these bands are: a thin dark blue line, a thick orange band, a thick light blue band, and a thick white band at the very bottom. The waves are smooth and flow from left to right.

UNIT I - INTRODUCTION:

- Software is more than just a program code.
- A program is an executable code, which serves some computational purpose.
- Software is the collection of computer programs, procedures rules and associated documentation and data.
- Software is an information transformer-producing, managing, modifying, displaying or transforming information that can simple as a single bit or a complex as a multimedia application.

Description of the Software

- A software is described by its capabilities. The capabilities relate to the functions it executes, the features it provides and the facilities it offers.

EXAMPLE

Software written for **Sales-order processing** would have different functions to process different types of sales order from different market segments .

- ✓ The features for example, would be to handle multi-currency computing, updating product, sales and Tax status.
- ✓ The facilities could be printing of sales orders, email to customers and reports to the store department to dispatch the goods.

Classes of Software

Software is classified into two classes:

- **Generic Software:**

is designed for broad customer market whose requirements are very common, fairly stable and well understood by the software engineer

- **Customized Software:**

is developed for a customer where domain, environment and requirements are being unique to that customer and cannot be satisfied by generic products

What is Good Software?

- Software has number of attributes which decide whether it is a good or bad .
- The definition of a good software changes with the person who evaluates it.
- The software is required by the customer, used by the end users of an organization and developed by software engineer
- Each one will evaluate the different attributes differently in order to decide whether the software is good

What are the attributes of good software?

The software should deliver the required functionality and performance to the user and should be **maintainable**, **dependable** and **usable**.

- **Maintainability**

- Software must evolve to meet changing needs

- **Dependability**

- Software must be trustworthy

- **Efficiency**

- Software should not make wasteful use of system resources

- **Usability**

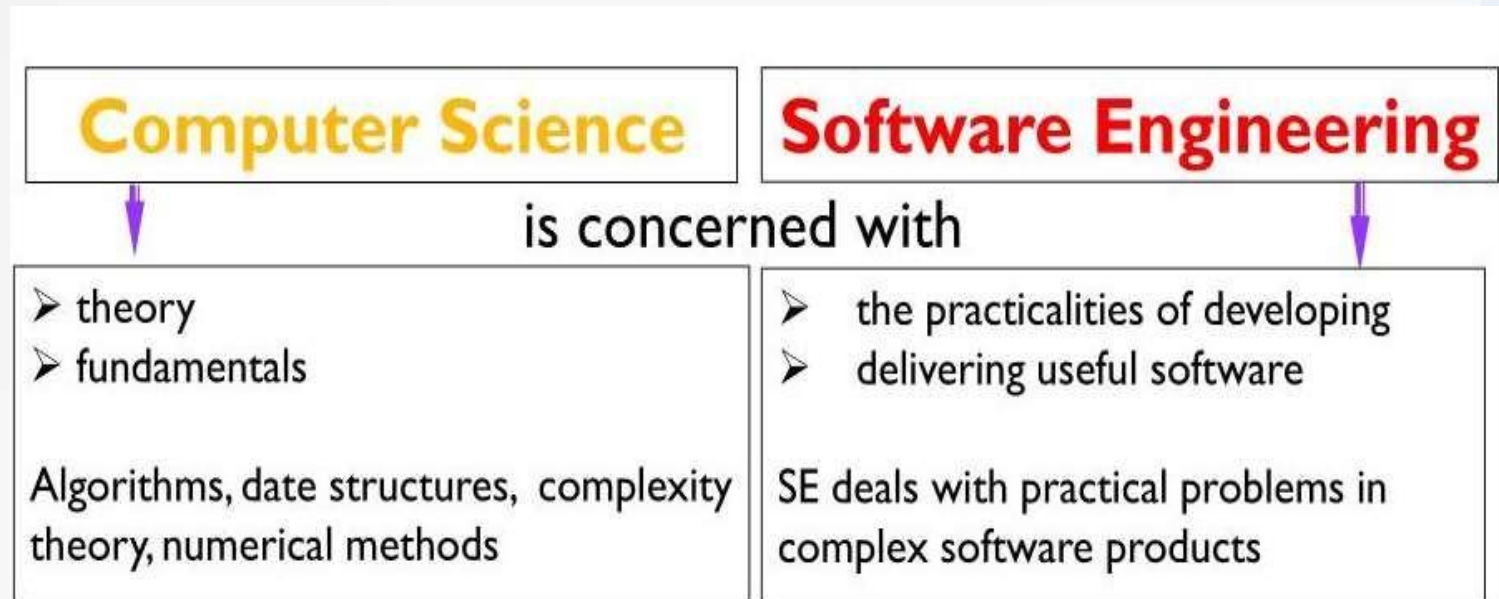
- Software must be usable by the users for which it was designed

Software Engineering: Definition

Software Engineering is a collection of techniques, methodologies and tools that help with the production of

- a high quality software system
- with a given budget
- before a given deadline

What is the difference between software engineering and computer science?



Computer science theories are currently insufficient to act as a complete underpinning for software engineering, BUT it is a foundation for practical aspects of software engineering

Software Engineering Paradigms:

Software Characteristics:

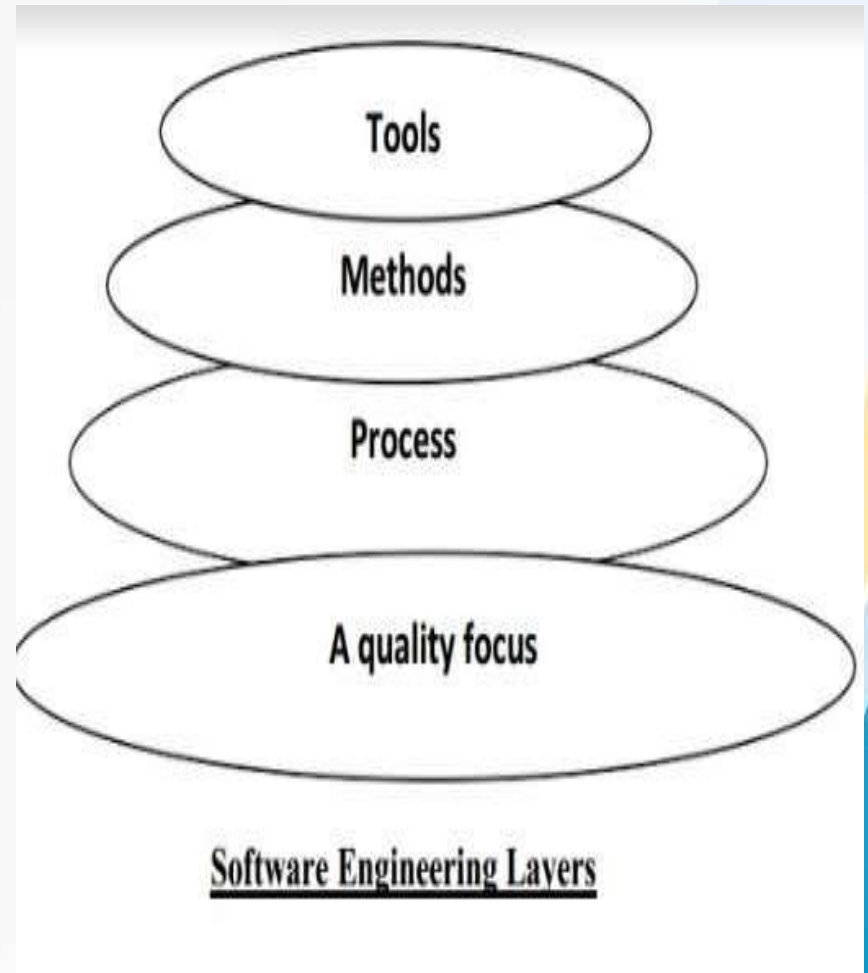
- Software is developed or engineered, it is not manufactured in the classical sence.
- Software doesn't "wear out".
- Although the industry is moving towards component based assembly, most software continues to be custom to built.

Software Applications Types:

- System Software.
- Real-time Software.
- Business Software.
- Engineering and Scientific Software.
- Embedded Software.
- Personal Computer Software.
- Web-based Software.
- Artificial Intelligence Software.

Software Engineering -A layered Technology:

- Application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software that is, the application of engineering software.



What are the five generic process framework activities?

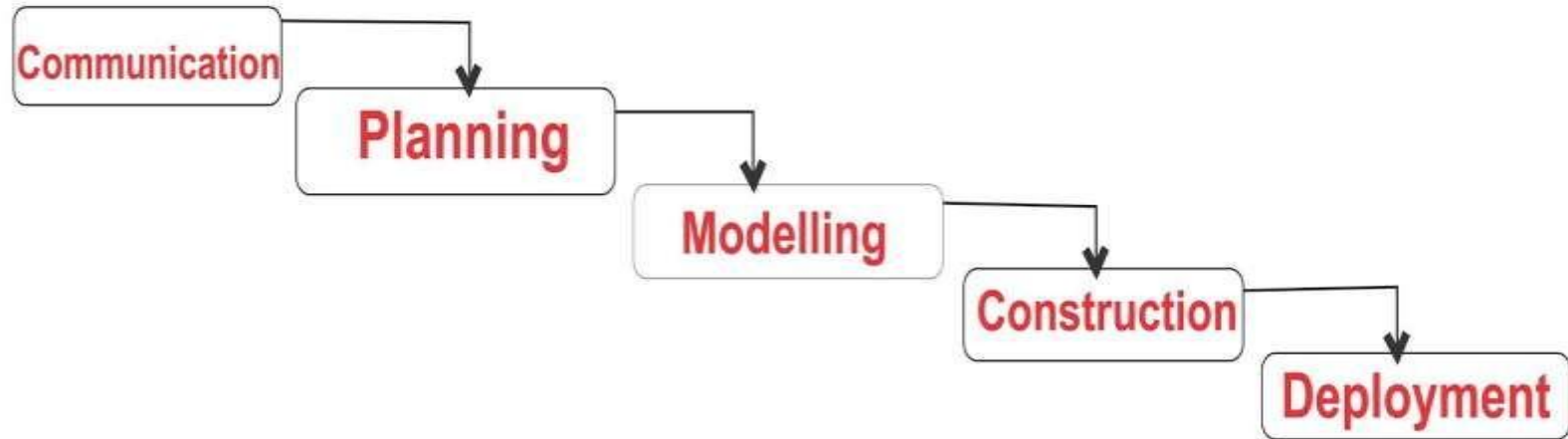
- The following generic process framework is applicable to the majority of software projects.
 - Communication.
 - Planning.
 - Modeling.
 - Construction.
 - Deployment.

Process Models:

- Every software engineering organization should describe a unique set of framework activities for the software process it adopts.
 - Waterfall Life Cycle Model.
 - Iterative Waterfall Life Cycle Model.
 - Prototyping Model.
 - Incremental Model.
 - Spiral Model.
 - RAD Model.

Waterfall Life Cycle Model.

- It is called classic life cycle or Linear model.
- Requirements are well defined and stable.
- It suggests a systematic, sequential approach to software development.
- It begins with customer specification of requirements and progresses.
 - Communication
 - Planning.
 - Modeling.
 - Construction and
 - Deployment.



WaterFall Model

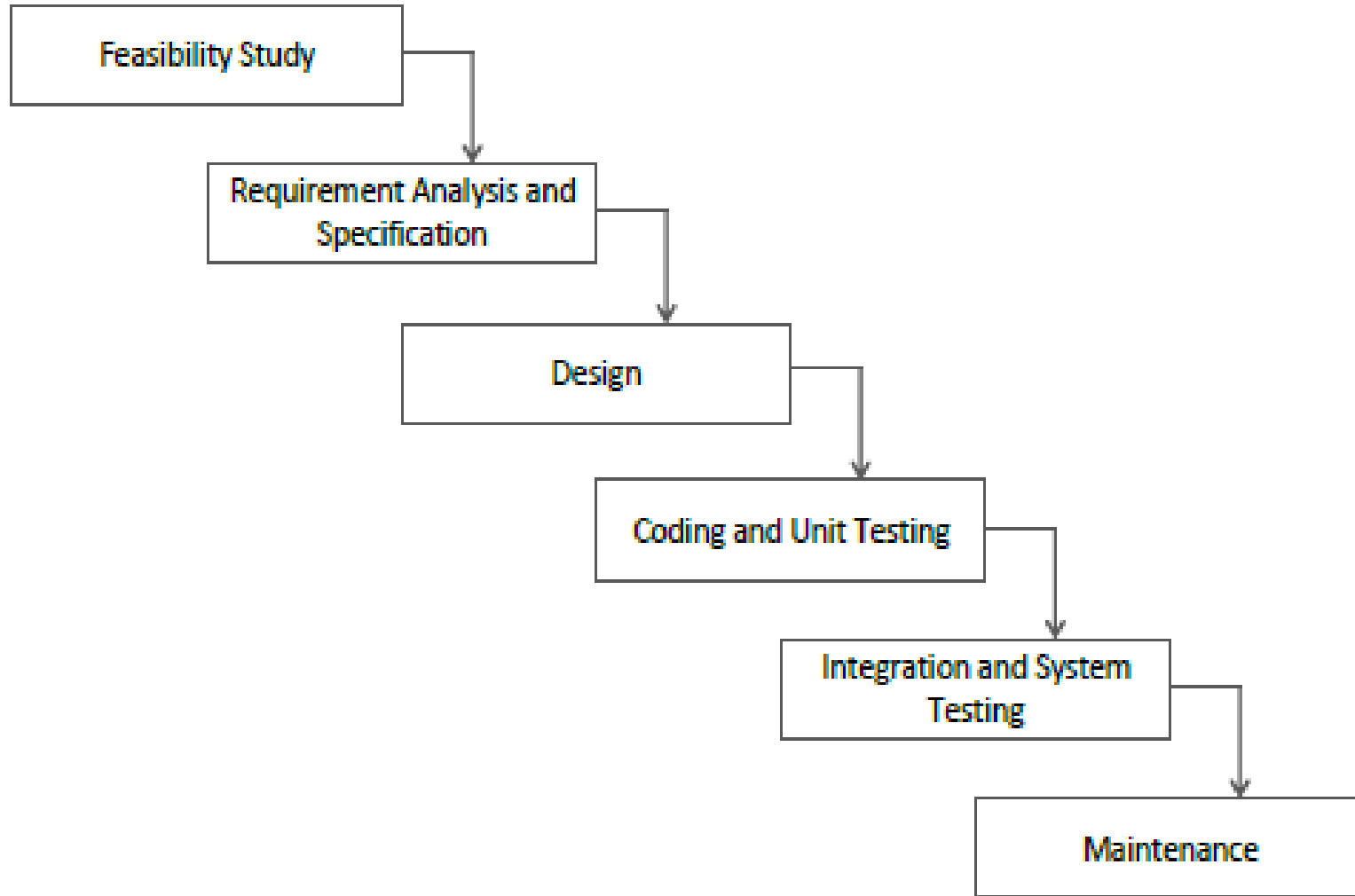
Advantages:

- Easy to understand.
- Each phase has well defined input and output.
- Helps project manager in proper planning of project.
- Provides a templates into which methods of analysis, design, code and support can be placed.

Disadvantages:

- One way street.
- It lack overlapping and interactions among phases.
- Model doesn't support delivery of system in pieces.

Phases of the Classical Waterfall Model:



Feasibility Study:

- It involves analysis of the problem and collection of all relevant information relating to the product.
- The collected data are analysed.
 - Requirements of the Customer.
 - Formulations of the different strategies for solving the problem.
 - Evaluation of different solution strategies.

Requirements Analysis and Specification:

- It is understand the exact requirements of the customer and to document them properly.
 - Requirements gathering and analysis.
 - Requirements specification.

Design:

- The design phase is to transform the requirements specified in the document into a structure that is suitable for implementation in some programming language.
 - Traditional Design Approach.
 - Object-Oriented Design Approach.

Coding and Unit Testing:

- The purpose of the coding and unit testing phase of software development is to translate the software design into source code.

Integration and System Testing:

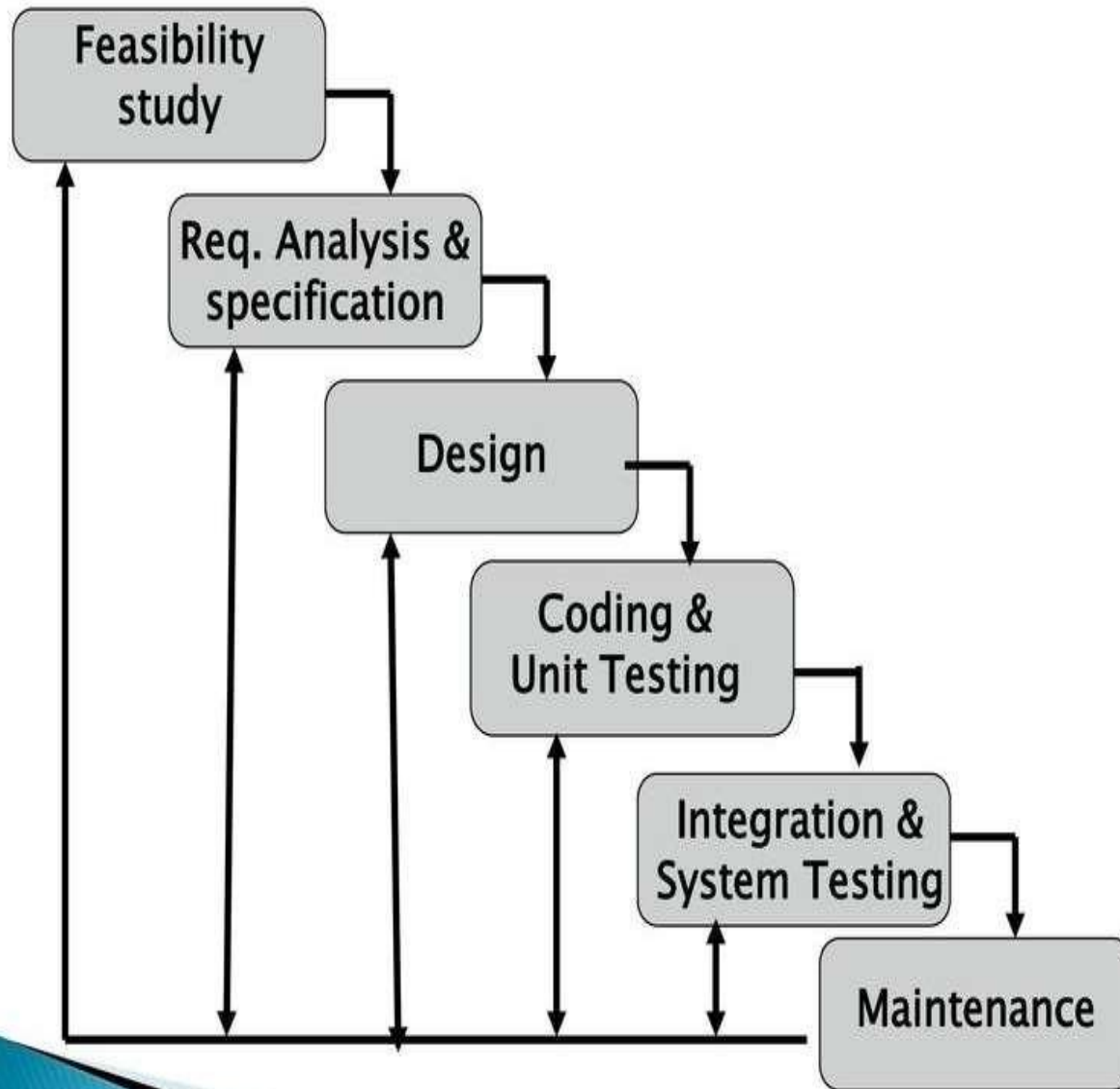
- 'Integration of different modules is coded and unit tested.'
 - Acceptance Testing.

Maintenance:

- Maintenance of a typical software products requires much more than the effort necessary to develop the product itself.

Iterative Waterfall life cycle model:

- The main changes is done by providing feedback paths from every phase to its preceding phase.



Prototype Model:

- Prototyping Model is a software development model in which prototype is built, tested, and reworked until an acceptable prototype is achieved.

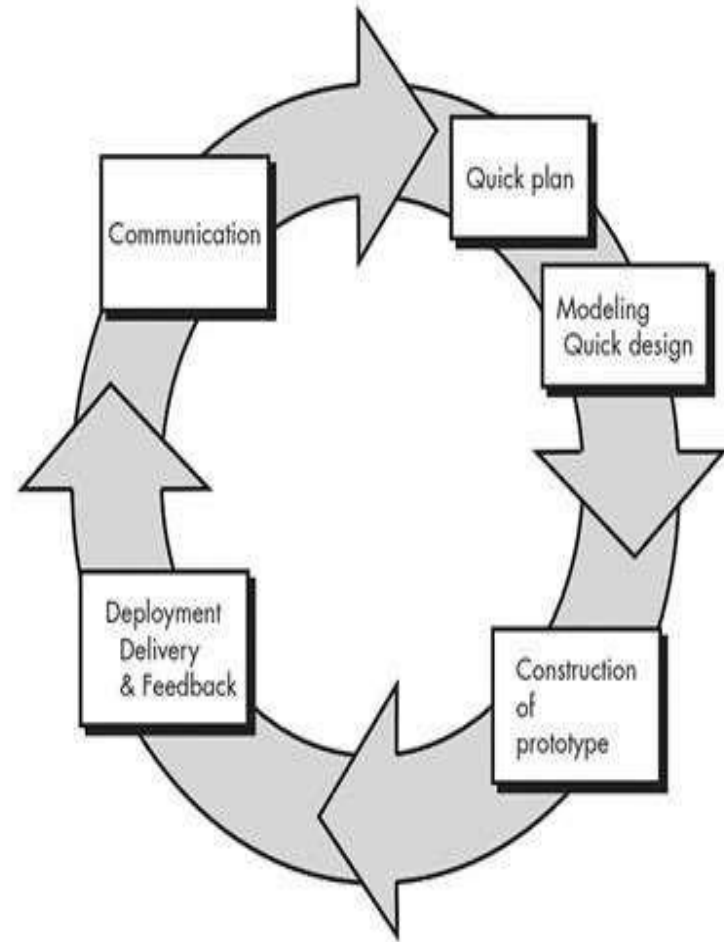


Figure: Prototype Model

Advantages:

- Clarity.
- Risk Identification.
- Good Environment.
- Take less time to complete.

Disadvantages:

- High cost.
- Slow process.
- Too many changes.

RAD Model:

- Rapid Application Development(RAD) is an incremental software model that a short development cycle.
- The RAD model is a “high-speed” of the waterfall model.
- The RAD process enables a development team to create a fully functional system within a very short time period.

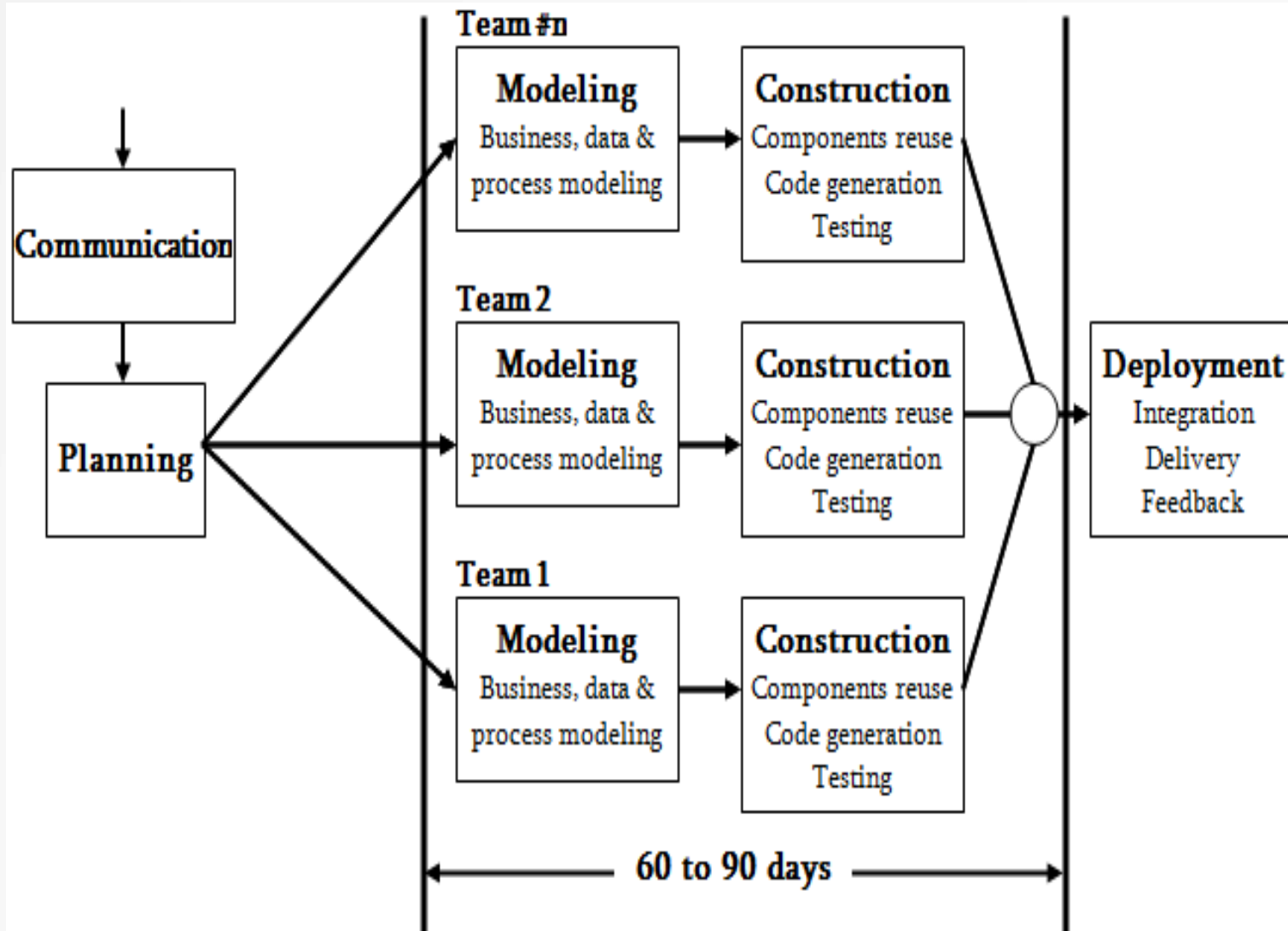


Figure : Flowchart of RAD model

Contents of RAD Packages:

- Graphical user development environment.
- Reusable Components.
- Code generator.
- Programming Language.

Advantages:

- Fast products.
- Efficient Documentation.
- Interaction with user.

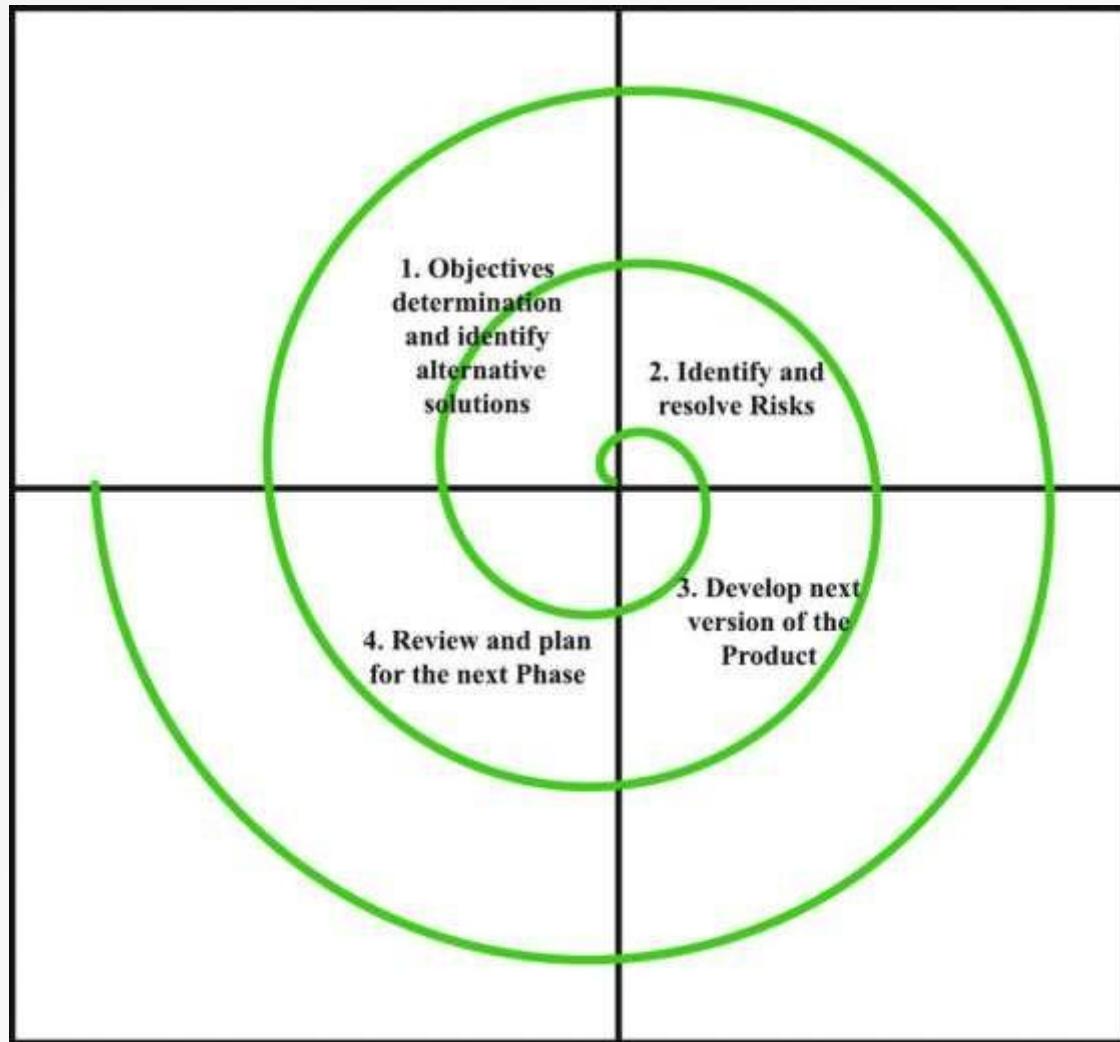
Disadvantages:

- User may not like fast activities.
- Not suitable for technical risks.

Spiral 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.
- The spiral model has four phases: Planning, Design, Construct and Evaluation.

Quadrants in spirial model :



Advantages :

- Risk Identification at early stage.
- Suitable for high projects.
- Flexibility for adding functionality.

Disadvantages:

- Costly.
- Risk dependent.
- Not suitable for smaller projects.
- Difficult to meeting budget.

Thanks

