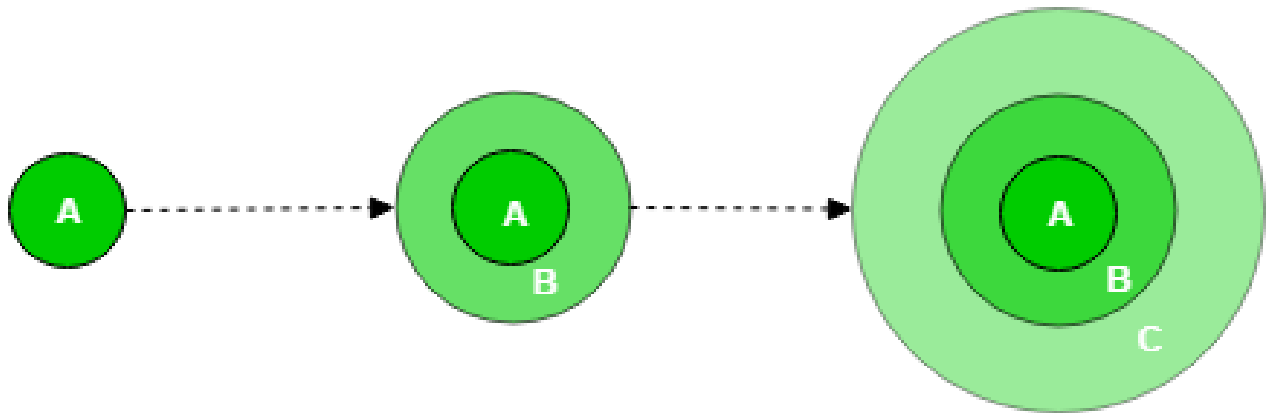# Incremental Model

The incremental model is advanced and popular life cycle model for software designing and development.

Incremental process model is also known as Successive version model.

First, a simple working system implementing only a few basic features is built and then that is delivered to the customer. Then thereafter many successive iterations/ versions are implemented and delivered to the customer until the desired system is released.

A, B, C are modules of Software Product that are incrementally developed and delivered.

Life cycle activities –
Requirements of Software are first broken down into several modules that can be incrementally constructed and delivered. At any time, the plan is made just for the next increment and not for any kind of long term plans. Therefore, it is easier to modify the version as per the need of the customer. Development Team first undertakes to develop core features (these do not need services from other features) of the system.

Once the core features are fully developed, then these are refined to increase levels of capabilities by adding new functions in Successive versions. Each incremental version is usually developed using an iterative waterfall model of development.

As each successive version of the software is constructed and delivered, now the feedback of the Customer is to be taken and these were then incorporated

in the next version. Each version of the software have more additional features over the previous ones.
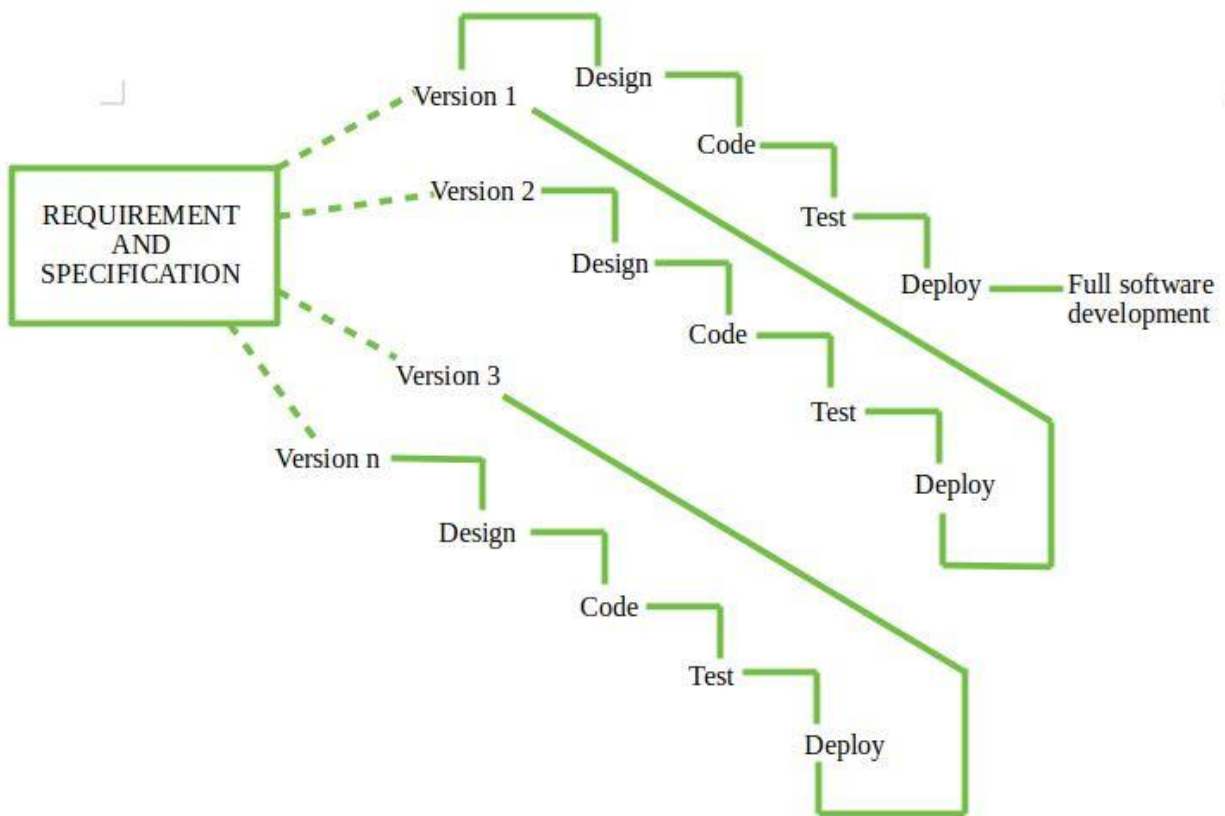
After Requirements gathering and specification, requirements are then spitted into several different versions starting with version-1(build 1), in each successive increment, next version is constructed and then deployed at the customer site. After the last version (version n)(build n), it is now deployed at the client site.

Types of Incremental model –
1. **Staged Delivery Model –** Construction of only one part of the project at a time.
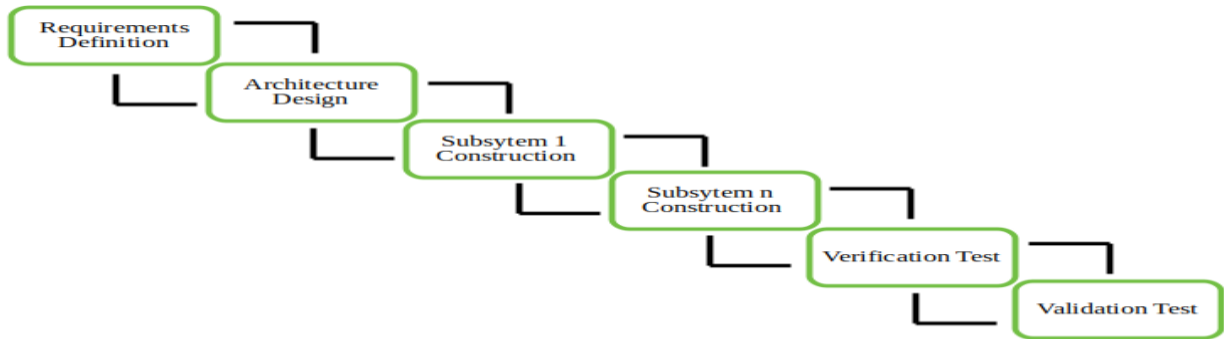
2. **Parallel Development Model –** Different subsystems are developed at the same time. It can decrease the calendar time needed for the development, i.e. TTM (Time to Market), if enough Resources are available.
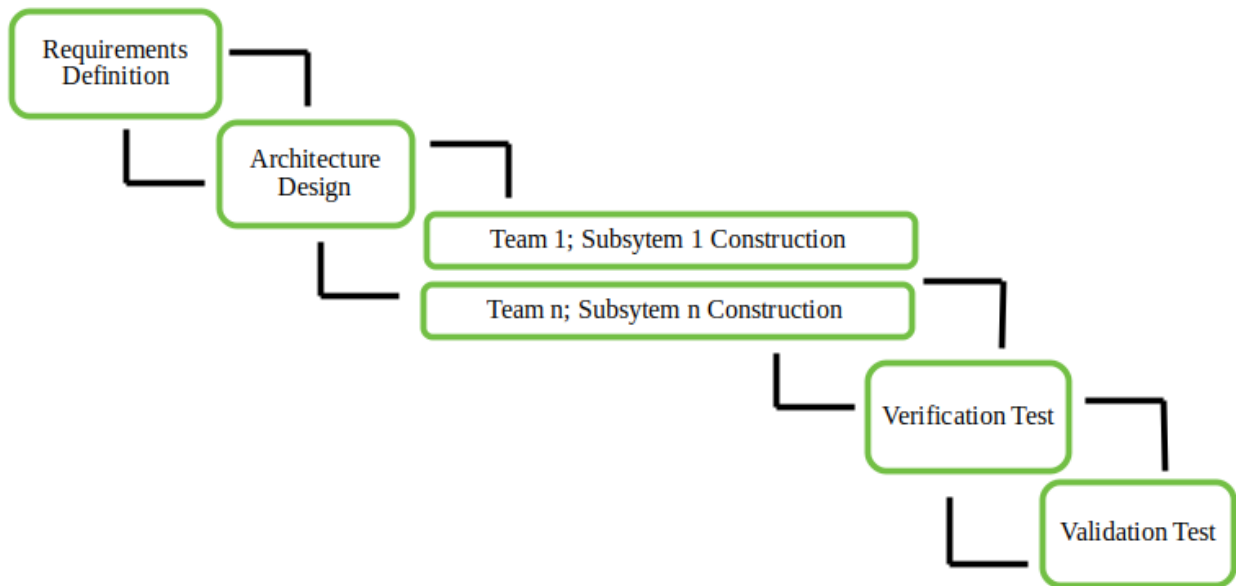
Advantages –
- Error Reduction (core modules are used by the customer from the beginning of the phase and then these are tested thoroughly)
- Uses divide and conquer for breakdown of tasks.
- Lowers initial delivery cost.
- Incremental Resource Deployment.

Disadvantages –

- Requires good planning and design.
- Total cost is not lower.
- Well defined module interfaces are required.

Applications:

1. Funding Schedule, Risk, Program Complexity, or need for early realization of benefits.
2. When Requirements are known up-front.
3. When Projects having lengthy developments schedules.
4. Projects with new Technology