Software Engineering

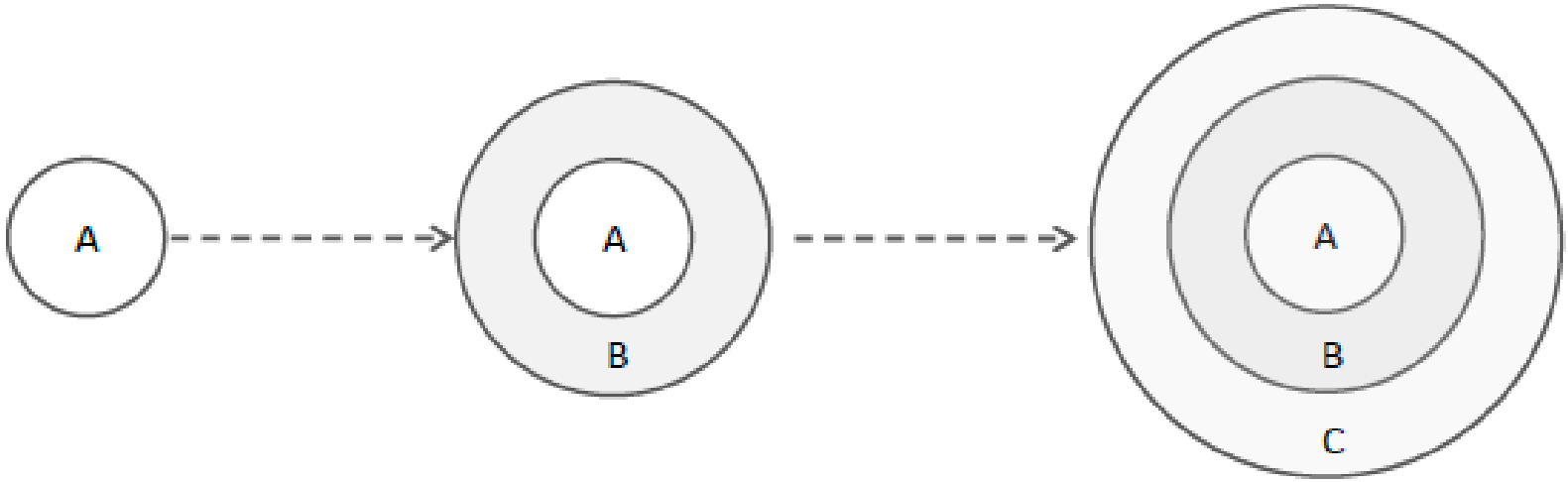
Prepared by: Neha Tripathi

Evolutionary Model

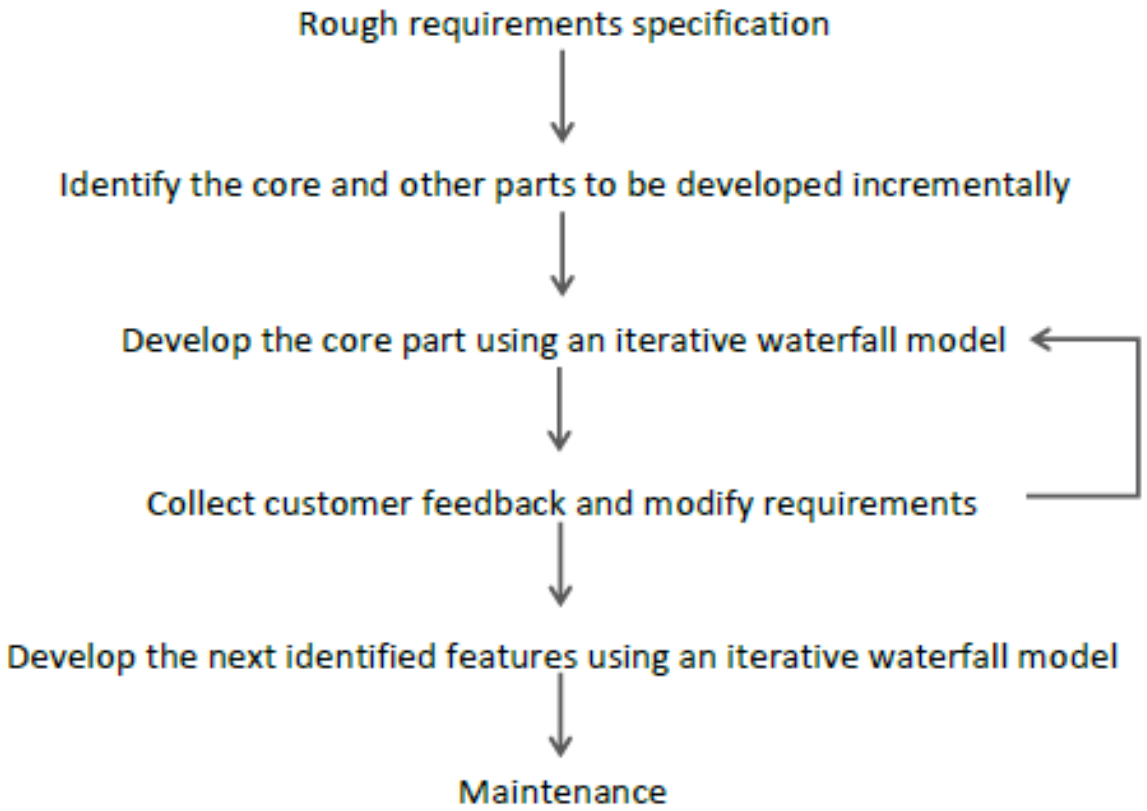
Evolutionary model

- **Evolutionary model** is also referred to as the **successive versions model** and sometimes as the **incremental model**.
- In Evolutionary model, the software requirement is first broken down into several modules (or functional units) that can be incrementally constructed and delivered.
- The development first develops the **core modules** of the system. The core modules are those that do not need services from the other modules.
- The initial product skeleton is refined into increasing levels of capability by adding new functionalities in successive versions.
- Each evolutionary model may be developed using an iterative waterfall model of development.
- The **evolutionary model** is shown in the figure. Each successive version/model of the product is a fully functioning software capable of performing more work than the previous versions/model.

Evolutionary Development of a Software Product



Evolutionary Model of Software Development



When to use the Evolutionary Model-Application?

- The **evolutionary model** is normally useful for very large products, where it is easier to find modules for incremental implementation.
- Often, **evolutionary model** is used when the customer prefers to receive the product in increments so that he can start using the different features as and when they are developed rather than waiting all the time for the full product to be developed and delivered.
- The evolutionary model is also very **useful in object-oriented software development** because all the development is divided into different units.

Advantages of Evolutionary Model

- **Large project:** Evolutionary model is normally useful for very large products.
- User gets a **chance to experiment with a partially developed software** much before the complete version of the system is released.
- Evolutionary model helps to accurately **elicit user requirements** during the delivery of different versions of the software.
- The core modules get tested thoroughly, thereby **reducing the chances of errors** in the core modules of the final products.
- Evolutionary model **avoids the need to commit large resources** in one go for development of the system.

Other benefits include:

- **Error reduction**: As the version is tested with customer which reduces the error thoroughly.
- **User satisfaction**: User gets satisfied and he gets the full chance of experimenting partially developed system.
- **Business benefit**: Successful use of this model can benefit not only business result but marketing and the internal operations as well.
- **High quality**: As you should get satisfied with every version, it produces the high quality product.
- **Low risk**: There is significant reduction of risk as a versions is implemented. This risk may be associated with : missing schedule deadline ,wrong feature sets , poor quality etc.
- **Reduction Cost**: Some design issues are cheaper to resolve through experimentation than through analysis. It reduces cost by providing structured and disciplined avenue for experimentation.

Disadvantages of Evolutionary Model

- **Difficult to divide the problem into several versions** that would be acceptable to the customer and which can be incrementally implemented and delivered.
- **Several version release**: Developer has to make table version which increases their Efforts.
- **Dividing software**: It is difficult to "divide the software and the problems in several versions that would be acceptable to the customer which can be implemented and delivered incrementally.
- **Uncertain nature of customer needs**: A confused user has uncertainty over his requirements, so giving him several version may change his requirement Rapidly.
- **Time And Cost**: As this model reduces "Time And Cost" but requirement is not gathered correctly. It will subsequently time, cost and efforts.
- **Confusion by several version**: An user might get "confused by several versions of the software. It will affect on the final product.

Thank You!