

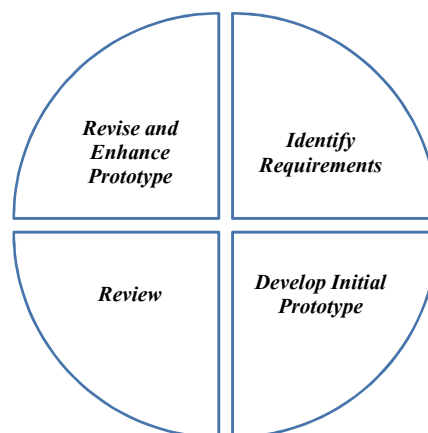
### ***PPM: Prototype Model***

- When a new car is being developed, **one or more prototypes will be individually built.** These prototypes are tested intensively before a production line is set up. It is possible to follow a similar approach with software development.
- **Prototyping** is the practice of building an early version of a system **which does not necessarily reflect all the features of the final system,** but rather those which are of interest.
- The purpose is to aid the **analysis and design stages** of a project by enabling users to see very early what the system will do due to Users seldom have a clear concise **understanding of their needs.**

## ***PPM: Prototype Model***

- *The process of prototyping involves the following steps:*
  1. **Identify basic requirements:** *Determine basic requirements including the input and output information desired.*
  2. **Develop Initial Prototype:** *The initial prototype is developed that includes only user interfaces.*
  3. **Review:** *The customers, including end-users, examine the prototype and provide feedback on additions requirements or changes.*
  4. **Revise and Enhance the Prototype:** *Using the feedback both the specifications and the prototype can be improved.*

## ***Prototype Process***

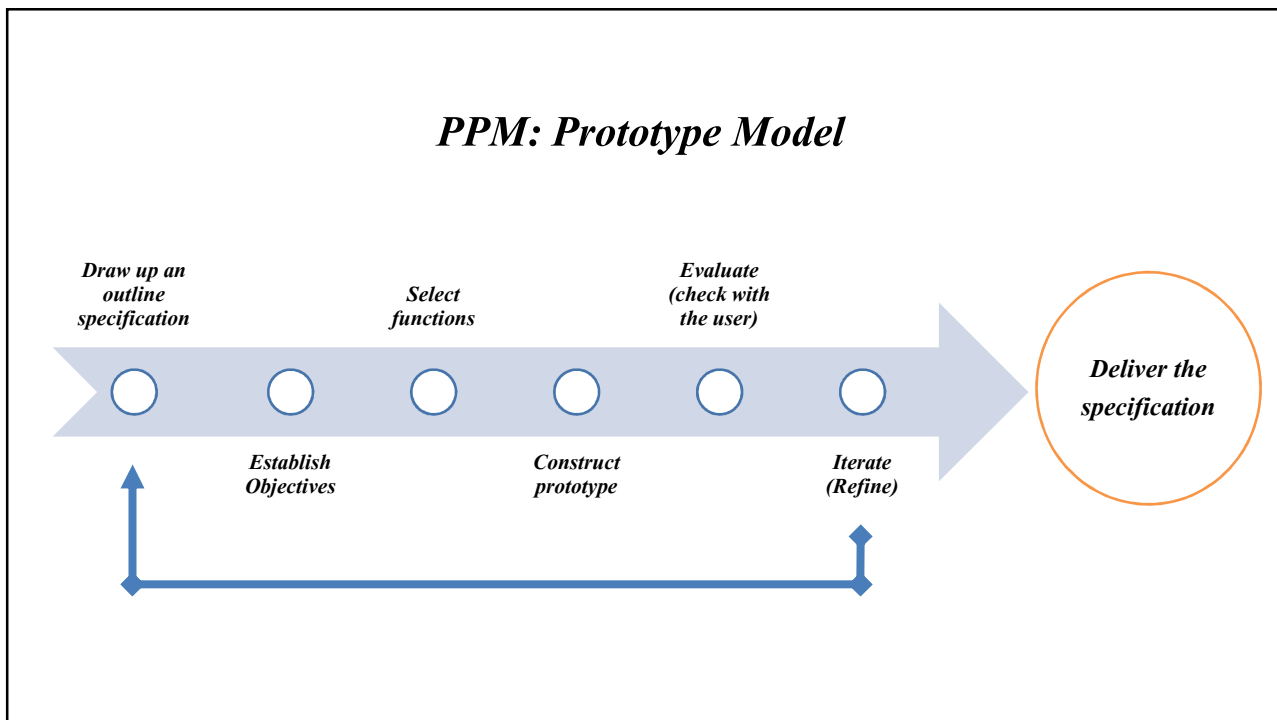


## ***PPM: Prototype Model***

- **The benefits of developing and demonstrating a prototype early in the software process are:**
  1. **Misunderstandings** between software developers and users may be identified missing facilities may be revealed.
  2. **To create an acceptable user interface.**
  3. **User training:** a prototype system can be used for training users before the final system has been delivered.
  4. **Difficult-to-use or confusing facilities may be identified,** and refined software developers may find incomplete and/or inconsistent requirements.

## ***PPM: Prototype Model***

- **Types Of Prototype Model:**
  1. **Throwaway:** Throwaway prototyping is sometimes called **rapid prototyping** and as the name suggests, a rapid prototype should cost very little and take very little time to develop.
- The starting point for throwaway prototyping is an outline specification for the software. **A throwaway prototype implements only those requirements that are poorly understood.**



### ***PPM: Prototype Model***

1. **Draw up an outline specification:** *The first step in throwaway prototyping is the creation of an initial, often partial, specification. This specification contains areas of uncertainty.*
2. **Establish Objectives:** *The objective may be to develop a system to prototype the user interface, to validate functional requirements, to explore uncertain new technologies or to demonstrate the feasibility of the application to management.*
3. **Select functions:** *The next stage is to decide what to put into and what to leave out of the prototype. This is determined by the objectives of the system.*

### ***PPM: Prototype Model***

4. **Construct prototype:** Speed and cost of construction of the prototype is crucial. Fast, low-cost construction is normally achieved by ignoring the normal quality requirements or the final product.
5. **Evaluate (check with the user):** The users use the prototype. This is more effective than watching a demonstration of the software. The prototype acts as an effective communication medium between the developer and customer.
6. **Iterate (Refine):** The prototype is rapidly modified, evaluation is carried out and the process repeated until the prototype meets the objectives (usually an agreed specification).

### ***PPM: Prototype Model***

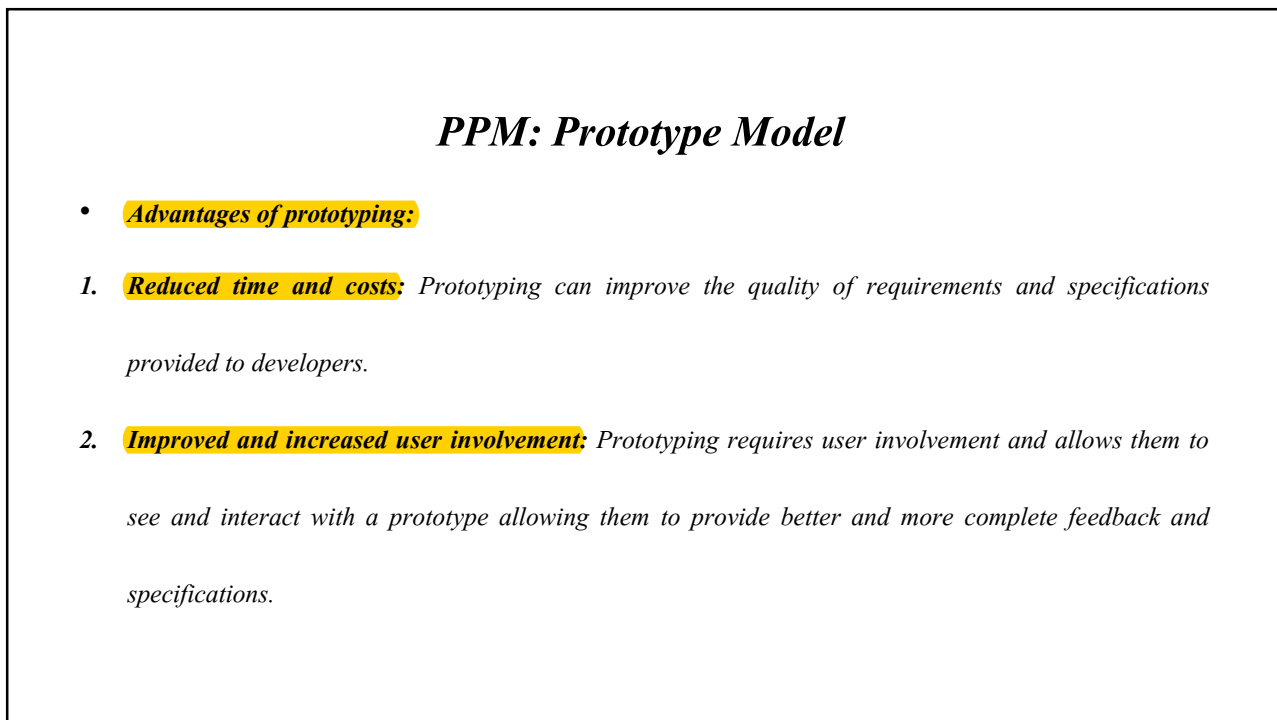
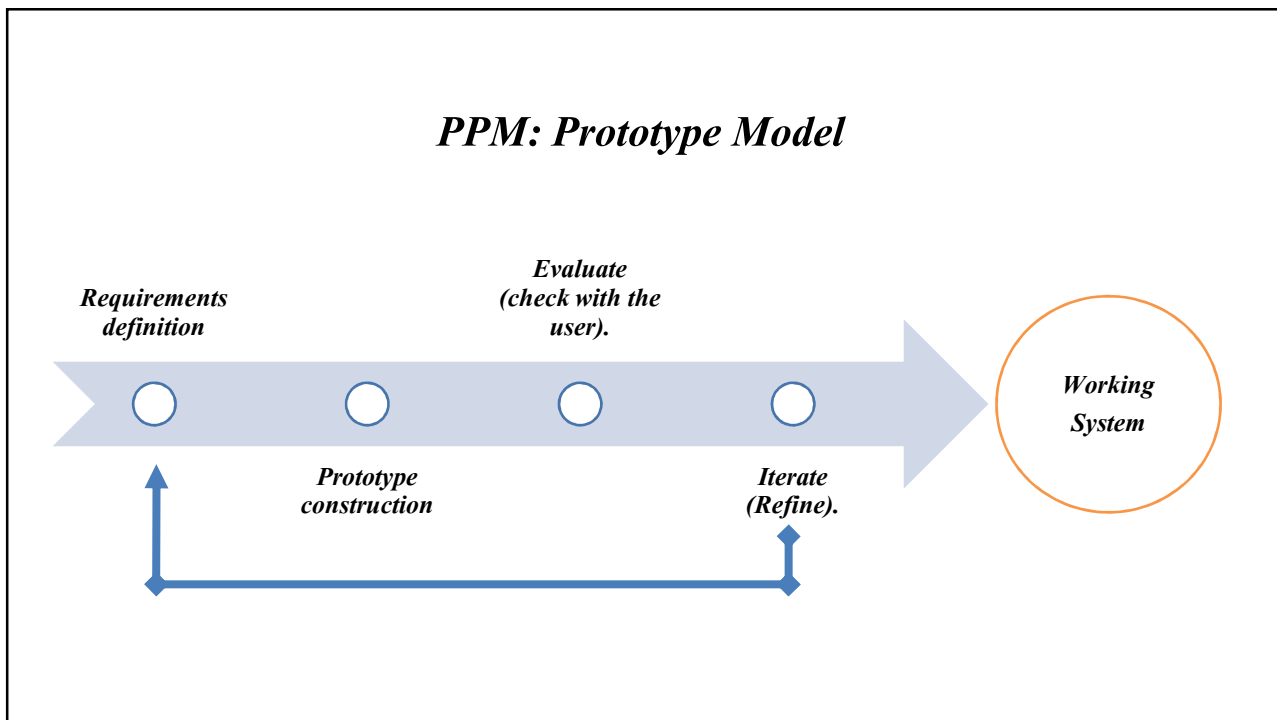
7. **Deliver the specification:** The product of the prototyping process is a specification that meets the users' requirements. Since the working prototype has been validated through interaction with the client.
  - **It is reasonable to expect that the resultant specification document will be correct.** When the requirements are clearly established, the prototype is thrown away.
  - **At this stage, a different software process model, such as the waterfall model,** is employed to develop the software.

## ***PPM: Prototype Model***

2. **Evolutionary Prototyping:** This type of prototyping is based on the idea of developing an initial implementation, exposing it to user comment and refining it through repeated stages until an adequate system has been developed. To be effective, evolutionary prototyping is carried out within a systematic framework.
1. **Requirements definition (initial specification):** Stage of thorough analysis is used to create an initial specification for the software.
2. **Prototype construction:** A prototype is built in a quality manner, including design, documentation, and thorough verification.

## ***PPM: Prototype Model***

3. **Evaluation (check with the user):** During evaluation, problems in the developer's perception of the customer requirements are uncovered. The prototypes are the communication medium that enables the developer and customer to communicate with each other.
4. **Iteration (refine the prototype):** Evaluation is carried out repeatedly until the prototype meets the objectives. The specification is updated with every iteration.



## ***PPM: Prototype Model***

- **Disadvantages of prototyping:** *Using, or perhaps misusing, prototyping can also have disadvantages.*
- 1. **Insufficient analysis:** *The focus on a limited prototype can distract developers from properly analyzing the complete project.*
- 2. **User confusion of prototype and finished system:** *Users can begin to think that a prototype, intended to be thrown away, is actually a final system that merely needs to be finished or polished.*

## ***PPM: Prototype Model***

- 3. **Excessive development time of the prototype:** *A key property to prototyping is the fact that it is supposed to be done quickly. If the developers lose sight of this fact, they very well may try to develop a prototype that is too complex.*
- 4. **Developer attachment to prototype:** *Developers can also become attached to prototypes they have spent a great deal of effort producing; this can lead to problems like attempting to convert a limited prototype into a final system when it does not have an appropriate underlying architecture.*

## ***Evolutionary Software Process Models***

- ***What Evolutionary means?*** This word is derived from the word “Evolution”. Evolution (noun): A regular process in which something changes into a different and usually much better form.
- Evolutionary models are iterative. They are characterized in a manner that enables software engineers to develop increasingly more complete versions of the software. There are several types of that model, these are:
  1. ***The Spiral Model***
  2. ***The Incremental Model***

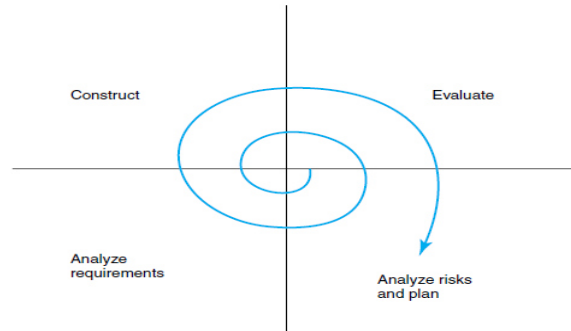
### ***ESPM: Spiral Model***

- The spiral model, originally proposed by Barry Boehm 1988. The spiral model is preferred for large, expensive and complicated projects. It combines the features of **Prototyping** and **Waterfall** models.
- The main feature of the spiral model is the recognition that there is often enormous uncertainty at many stages during a software development project. Many decisions are taken during software development, and for every decision there is a risk.
- Risk result in project problems such as schedule and cost overrun. The spiral model approach is therefore try to discover errors frequently – at each cycle. Then something can be done about them immediately. The distinctive feature of the spiral model is that it makes explicit the idea of risk.

## ***ESPM: Spiral Model***

- *The spiral model is shown in Figure. Progress is shown as a line that spirals out from near the centre of the diagram. As the diagram shows, each cycle consists of four steps:*

1. **Analyze risks and plan.**
2. **Analyze requirements.**
3. **Construct.**
4. **Evaluate.**



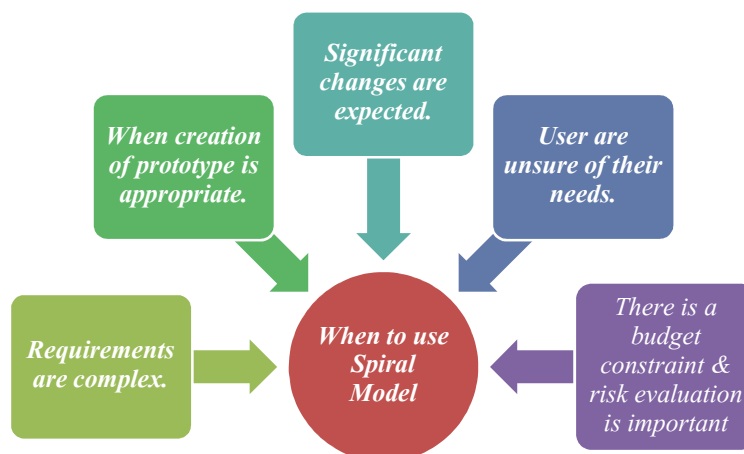
## ***ESPM: Spiral Model***

1. **Risk Analysis And Planning:** *This stage is the essential ingredient of the spiral model. It consists of:*
  1. *Establishing the objectives of the product of this stage (e.g. performance, functionality)*
  2. *Identifying the constraints that affect this stage (e.g. Cost, deadlines, interfaces with other software components). And Identifying the risks.*
  3. *Identifying the alternative ways of implementing this stage (buying it, reusing something else).*
  4. *Evaluating the alternative implementation schemes against the criteria set by the objectives and the constraints. And Deciding how to overcome the risk.*
  5. *Establishing deadlines for the next stage of project and deciding how many people will be involved.*

## *ESPM: Spiral Model*

2. **Analysis Of Requirements:** This consists of establishing the requirements for the next stage of the project.
3. **Construction:** This stage involve design, implementation, validation and verification, depending on the nature of the product.
4. **Evaluation:** Evaluation is used to establish whether the project is on track and whether all the participants are happy with the plans. This leads on to the next cycle of the project.

## *ESPM: Spiral Model*



## ***ESPM: Spiral Model***

- ***Advantages of the Spiral Model***

1. *Development can be divided into smaller parts and the risky parts can be developed earlier which helps in better risk management.*
2. *It is developmental and focuses on component reuse.*
3. *Requirements can be captured more accurately.*
4. *Low risk of making mistakes – waterfall.*
5. *Good customer support, management control, integration, decision making – evolutionary.*
6. *Allows extensive use of prototypes.*

## ***ESPM: Spiral Model***

- ***Disadvantages of the Spiral Model***

1. *Customers may feel insecure using this model.*
2. *Good risk-assessment skills are required.*
3. *Management is more complex.*
4. *End of the project may not be known early.*
5. *Not suitable for small or low risk projects and could be expensive for small projects.*
6. *Large number of intermediate stages requires excessive documentation.*

