Assignment 1

Student Name: Sahil Kaundal UID: 21BCS8197

Branch: CSE (Lateral Entry) Section/Group:20BCS-807\_B Semester: 4th Date of Performance: 18/03/2022

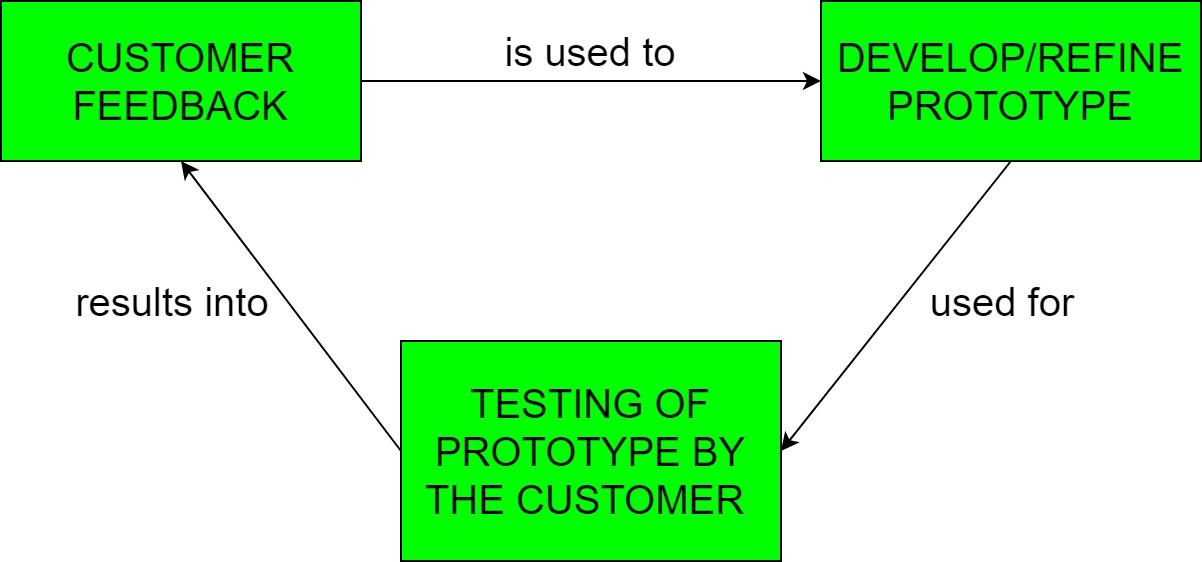
Subject Name: SE Subject Code: 20CSP-254

Q1.

Prototyping model affects the cost of project. Explain and justify, also illustrate the effect of designing a prototype on the overall cost of software project.

Answer:

The prototyping model is a systems development method in which a [prototype](https://www.techtarget.com/searcherp/definition/prototype) is built, tested and then reworked as necessary until an acceptable outcome is achieved from which the complete system or product can be developed. This model works best in scenarios where not all of the project requirements are known in detail ahead of time. It is an iterative, trial-and-error process that takes place between the developers and the users.



### Steps of the prototyping model

In most cases, the steps of the prototyping model are as follows:

1. The new system requirements are defined in as much detail as possible. This usually involves interviewing a number of users representing all the departments or aspects of the existing system.
2. A preliminary, simple design is created for the new system.
3. A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
4. The users thoroughly evaluate the first prototype and note its strengths and weaknesses, what needs to be added and what should to be removed. The developer collects and analyses the remarks from the users.
5. The first prototype is modified, based on the comments supplied by the users, and a second prototype of the new system is constructed.
6. The second prototype is evaluated in the same manner as was the first prototype.
7. The preceding steps are iterated as many times as necessary, until the users are satisfied that the prototype represents the final product desired.
8. The final system is constructed, based on the final prototype.
9. The final system is thoroughly evaluated and tested. Routine maintenance is carried out on a continuing basis to prevent large-scale failures and to minimize downtime.

Prototyping model affects the cost of project:

Prototyping may have some initial costs of developing, but it

reduces the overall budget by helping your product to be free of

the errors or glitches that could have occurred if the idea was

made from scratch without any prior user testing. Furthermore,

prototyping also helps to understand the intrinsic flaws,

shortcomings and drawbacks that can be improved during the

product development process. If the prototyping process is

ignored completely, it might result in the restructuring and

redesigning of the entire product after spending all your resources

on its development. So, the effect of designing a prototype on the

overall cost of a software project is to actually reduce the

additional costs of restructuring and reframing it after its full-

fledged development- which might cost a fortune.

Prototyping may have some initial costs of developing, but it

reduces the overall budget by helping your product to be free of

the errors or glitches that could have occurred if the idea was

made from scratch without any prior user testing. Furthermore,

prototyping also helps to understand the intrinsic flaws,

shortcomings and drawbacks that can be improved during the

product development process. If the prototyping process is

ignored completely, it might result in the restructuring and

redesigning of the entire product after spending all your resources

on its development. So, the effect of designing a prototype on the

overall cost of a software project is to actually reduce the

additional costs of restructuring and reframing it after its full-

fledged development- which might cost a fortune.

Prototyping may have some initial costs of developing, but it

reduces the overall budget by helping your product to be free of

the errors or glitches that could have occurred if the idea was

made from scratch without any prior user testing. Furthermore,

prototyping also helps to understand the intrinsic flaws,

shortcomings and drawbacks that can be improved during the

product development process. If the prototyping process is

ignored completely, it might result in the restructuring and

redesigning of the entire product after spending all your resources

on its development. So, the effect of designing a prototype on the

overall cost of a software project is to actually reduce the

additional costs of restructuring and reframing it after its full-

fledged development- which might cost a fortune.

Prototyping may have some initial costs of developing, but it

reduces the overall budget by helping your product to be free of

the errors or glitches that could have occurred if the idea was

made from scratch without any prior user testing. Furthermore,

prototyping also helps to understand the intrinsic flaws,

shortcomings and drawbacks that can be improved during the

product development process. If the prototyping process is

ignored completely, it might result in the restructuring and

redesigning of the entire product after spending all your resources

on its development. So, the effect of designing a prototype on the

overall cost of a software project is to actually reduce the

additional costs of restructuring and reframing it after its full-

fledged development- which might cost a fortune.

Prototyping may have some initial costs of developing, but it

Prototyping may have some initial costs of developing, but it

Prototyping may have some initial costs of developing, but it

Prototyping may have some initial costs of developing, but it

Prototyping may have some initial costs of developing, but it

Prototyping may have some initial costs of developing, but it

Explain and justify, also illustrate the effect of designing a

Explain and justify, also illustrate the effect of designing a

Explain and justify, also illustrate the effect of designing a

Explain and justify, also illustrate the effect of designing a

Prototyping may have some initial costs of developing, but it reduces the overall budget by helping your product to be free of the errors or glitches that could have occurred if the idea was made from scratch without any prior user testing. Furthermore, prototyping also helps to understand the intrinsic flaws, shortcomings and drawbacks that can be improved during the product development process. If the prototyping process is ignored completely, it might result in the restructuring and redesigning of the entire product after spending all your resources on its development. So, the effect of designing a prototype on the overall cost of a software project is to actually reduce the additional costs of restructuring and reframing it after its full-fledged development- which might cost a fortune.

The effect of designing a prototype on the overall cost of software project:

It depends on how you define “cost”. If you define cost in the traditional sense of the word - i.e. cost of labour, time and effort - then the cost will increase as your adding extra activities. Especially since these activities are mainly meant to learn and not to necessarily generate tangible out, things like prototyping will add additional cost.

However, if you define cost in terms of cost of delay or opportunity cost, then discovery through prototypes will reduce overall cost. The key about using prototypes - irrespective of level of fidelity - is the learning about potential risks:

1. **Customer risk**- Will the user buy this (or choose to use this)? Why (not).
2. **Usability risk** - Can the user figure out how to use this? What works, what doesn’t (and why).
3. **Implementation risk** - Can our engineer build this? Is it technically viable.
4. **Business risk** - Can our stakeholders support this? Is it viable from a business perspective?

The cost of not identifying and mitigating these risks early and often can be significant, if you think of cost in terms of missed opportunities and delays in speed to market.

Q2.

Justify the term: “Design is not coding and coding is not design”.

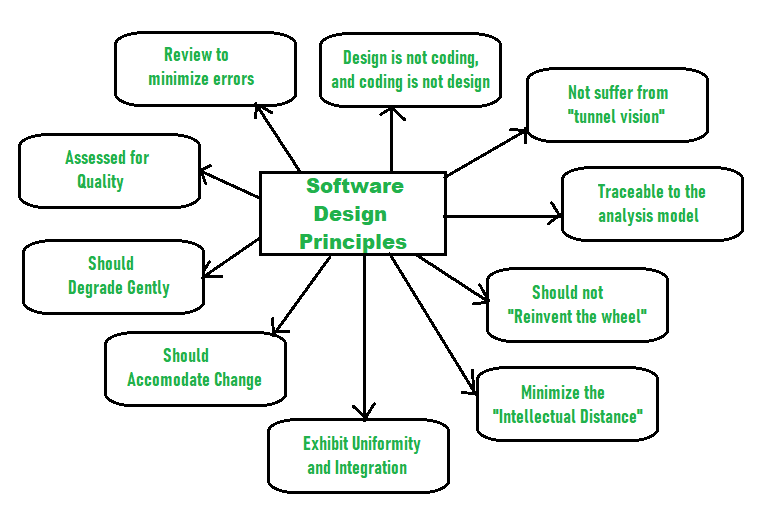
Answer:

Design means to draw or plan something to show the look, functions and working of it. Design means describing the logic of the program to solve any problem and coding is a type of language that is used for the implementation of a design.

Software Design is also a process to plan or convert the software requirements into a step that are needed to be carried out to develop a software system. There are several principles that are used to organize and arrange the structural components of Software design.

Software Designs in which these principles are applied affect the content and the working process of the software from the beginning.

These principles are stated below:



Principles Of Software Design:

1. Should not suffer from “Tunnel Vision” –  
   While designing the process, it should not suffer from “tunnel vision” which means that is should not only focus on completing or achieving the aim but on other effects also.
2. Traceable to analysis model –  
   The design process should be traceable to the analysis model which means it should satisfy all the requirements that software requires to develop a high-quality product.
3. Should not “Reinvent The Wheel” –  
   The design process should not reinvent the wheel that means it should not waste time or effort in creating things that already exist. Due to this, the overall development will get increased.
4. Minimize Intellectual distance –  
   The design process should reduce the gap between real-world problems and software solutions for that problem meaning it should simply minimize intellectual distance.
5. Exhibit uniformity and integration –  
   The design should display uniformity which means it should be uniform throughout the process without any change. Integration means it should mix or combine all parts of software i.e. subsystems into one system.
6. Accommodate change –  
   The software should be designed in such a way that it accommodates the change implying that the software should adjust to the change that is required to be done as per the user’s need.
7. Degrade gently –  
   The software should be designed in such a way that it degrades gracefully which means it should work properly even if an error occurs during the execution.
8. Assessed or quality –  
   The design should be assessed or evaluated for the quality meaning that during the evaluation, the quality of the design needs to be checked and focused on.
9. Review to discover errors –  
   The design should be reviewed which means that the overall evaluation should be done to check if there is any error present or if it can be minimized.
10. Design is not coding and coding is not design –  
    Design means describing the logic of the program to solve any problem and coding is a type of language that is used for the implementation of a design.

Software are programs used by computer. Coding is a language written using protocol of language to develop apps, software's, etc. Software's allows users or customers to interact with and is functional thing they can use. Coding is simply construction or design that is hidden aspect of software.

Designing a website with attributes and framing on right place is different from set of lines denoting the programming language with huge content which is basically the backend process. Hence coding is part of designing. It is not designing, in the same manner design is not coding.