Assignment-1

60004200107

Q1. Elaborate the task set for creating component level design in OO concepts

- component level design is an important step in OO software development. It involves creating detailed design for individual components of a software system based on requirement & specification. Task set involved in creating a component level design:

1) Identify all design classes that correspond to the problem domain. Using requirement and architecture model, each analysis class and architectural component is elaborated.

2) Identify all design classes that correspond to the infrastructure domain: These classes are not described in requirement model and are often missing from the architecture model. As we noted it includes GUI components, operating system components and object and data management components.

3) Elaborate all design classes that are not acquired as reusable components. Elaboration requires that all interfaces, attributes, and operations necessary to implement the class as described in detail. Design heuristics must be considered as this task is conducted.

(a) specify message details when classes or components collaborate.

The requirements model makes up of a collaboration diagram to show how analysis classes collaborate with one another. As component level ~~des~~ design proceeds, it is sometimes useful to show the details of these collaborations. Although this is optional it can be used as a precursor to specification

(b) identify appropriate interfaces for each component. In essence operations defined for the design class are categorized int one or more abstract classes. Every operation within the abstract classes.

(c) Ellaborate attributes and define data types and data structures required to implement them. If an attribute appears repeatedly across a number of design classes and it is a relatively complex structure.

(d) Describe processing flow within each operation in details. This may be accomplished using programming languages, pseudocode, or with a UML activity diagram.

4) Describe persistent data sources (databases and files) and identify classes to manage them. Database & files normally transcend the design description of an individual component. In most cases,

these persistent data structures.

5) Develop and elaborate behavioural representations for a class or component:

Some state diagrams were used as part of the requirements model to represent the externally observable behaviour of the system. During component level design it is sometimes necessary.

6) Elaborate deployment diagrams to provide additional implementation detail.

Deployment diagrams are used as part of architecture design and are represented in descriptor format. Deployment diagrams can be elaborated to represent the location of key packages of components. However, components are not represented individually.

7) Refactor every component level design representation and always alternatives.

It is an iterative process, first component model will not be complete, consistent or accurate. They do not suffer from tunnel vision. Develop alternatives and consider each carefully

Q2 Explain golden rules of user Interface design

The following three rules:

(i) Place user in control

(ii) Reduce user's memory load.

(iii) Make Interface consistent.

(i) Place user in control

- Define interaction mode in such a way that does not force a user into unnecessary undesired actions.
- Provide flexible interaction
- Allow user interaction to be interruptible and undoable.
- Streamline interaction as skill levels advance and allow the interaction to be customized.

(ii) Reduce user's memory load.

- Reduce demand on short term memory
- Establish meaningful defaults.
- Define shortcuts that are intuitive.
- The visual layout of interface should be based on real world metaphor

(iii) make interface consistent

- Allow user to put the current task into meaningful context.
- Maintain consistency across a family of applications
- If past interactive models have created user expectations, do not make changes unless there is a compelling reason to do so.