Learning Journal Unit4

**a) Discuss the key points from GRASP patterns to assign responsibilities in designs.**

When designing a system, there are four design goals:

* Easy to modify
* Easy to understand
* Easy to divide labor
* Easy to reuse

The four points to achieve these goals, the nine principles of object responsibility assignment for object-oriented design are called GRASP (General Responsibility Assignment Software Patterns). (Aldrich and Garrod,2015)

Below are the main points for assigning responsibilities in the GRASP pattern:

**Information expert**

When assigning responsibility to an object, the principle is that, in principle, the necessary information should be assigned to the mourning p object. The information expert plays a role in assigning the task to the object best suited to perform it.

**Low Coupling**

Large dependencies between classes make modifications difficult to make and difficult to understand. Therefore, the principle is that modules should be as independent of other modules as possible.

**High Cohesion**

The principle is that all information necessary to execute a task should be concentrated in a single object.

**Controller**

When assigning responsibility for executing a task to an object that is not a user interface, it is necessary to pass information from the interface. The controller manages input and output events for each object and makes requests to the appropriate objects.

**Creator**

Objects must be created. Therefore, object B, which creates new object A, is created according to the following principle. (Aldrich and Garrod,2015)

* B aggregates A objects
* B contains A objects
* B records instances of A objects
* B closely uses A objects
* B has the initializing data for creating A objects

**Indirection**

To avoid tightly coupled objects and to achieve a loosely coupled design, intermediate objects should be prepared and assigned responsibilities.

**Polymorphism**

To design a class that ensures diversity, there should be a person to receive the different behaviors of the objects.

**Protected Variations**

The potential for change should be identified, and an interface should be provided that will work without problems even if changes are made.

**Pure Fabrication**

A class that is created to achieve loose coupling, high cohesion, and reusability, rather than a class that represents a concept that exists in the problem domain.

These GRASP patterns can be broadly categorized as follows:

**Object Dependency Principles**

* High Cohesion
* Low Coupling
* Indirection

**Principles related to role assignment**

* Information Expert
* Creator
* Pure Fabrication
* Controller

**Principles for Encapsulating Variability**

* Polymorphism
* Protected Variations

We believe it is important to protect the principles of object dependencies and role assignment when assigning responsibilities.

**b) Explain the concepts of Coupling and Cohesion with the help of a suitable example. Discuss the following items in your response:**

For basic system development, High Cohesion and Low Coupling are appropriate. Let us consider the reasons for each concept of Cohesion and Coupling.

Cohesion

Cohesion is a measure of how strongly the responsibilities of elements are related and converged. Therefore, a low Cohesion means that unrelated tasks are executed or that the tasks allocated to an object are excessive.

Coupling

Coupling is a measure of how dependent one element is on another. If an object is designed with a high coupling, it loses its modifiability and understandability, resulting in poorly maintainable code.

However, depending on the circumstances of the project, there may be patterns of design with low cohesion and tight coupling. It may also exist when there are only a very small number of personnel with specific domain knowledge and there are cases where this may be implemented to consolidate code and responsibilities.

Word-Count:542 (Excluding questions and references)

References

Aldrich, J.I., & Garrod, C. (2015). Assigning Responsibilities [Presentation slides]. In *Principles of Software construction: Objects, design and concurrency*. Institute for Software Research. https://www.cs.cmu.edu/~charlie/courses/15-214/2015-fall/slides/03b-assigning-responsibilities.pdf