CSE6214 Software Engineering Fundamentals

Tutorial 8

Part A: Discussion
Topic (Lecture 8): WebApp and MobileApp Design

1. Discuss the need to design Web applications. When is it necessary to design them?

2. For the end-users, what are the important quality dimensions of Web applications?

3. Describe each level of the WebE Design Pyramid.

4. What are the important considerations in the development of mobile applications?

5. Discuss the typical mistakes in mobile application development and suggest some approaches to address them.

Part B: Tutorial Activity
Topic: Component-Level Design

Activity 1 – Cohesion

- Within the context of component-level design for object-oriented systems, cohesion implies that a component or class encapsulates only attributes and operations that are closely related to one another and to a class or component itself.
- Type and search “component cohesion” in the Internet.

Describe the following types of cohesion:

- Functional
- Layer
- Communicational

Activity 2 – Coupling

- Coupling is a qualitative measure of the degree to which classes are connected to one another.
- As classes (and components) become more interdependent, coupling increases.
- An important objective in component-level design is to keep coupling as low as possible.
- Software must communicate internally and externally.

- Therefore, coupling is a fact of life.
- However, designer should work to reduce coupling whenever possible and understand the ramifications of high coupling when it cannot be avoided.

Match the following coupling categories with its descriptions. You should select the best answer from Table 1.

Table 1 – List of Coupling Categories

Common coupling	Data coupling	Control coupling
Routine call coupling	Type use coupling	Couple coupling
Inclusion/Import coupling	Content coupling	External coupling
Stamp coupling	Unusual coupling	Good coupling

2a) ________________________

- Occurs when one component “surreptitiously modifies data that is internal to another component.
- This violates information hiding – a basic design concept.

2b) ________________________

- Occurs when a number of components all the make use of a global variable.
- Although this is sometimes necessary (e.g. for establishing default values that are applicable throughout an application), common coupling can lead to uncontrolled error propagation and unforeseen side effects when changes are made.

2c) ________________________

- Occurs when operation A () invokes operation B() and passes a control flag to B.
- The control flag the “directs” logical flow within B.
- The problem with this form of coupling is that an unrelated change in B can result in the necessity to change the meaning of the control flag that A passes.
- If this overlooked, an error will result.

2d) ________________________

- Occurs when **ClassB** is declared as a type of an argument of an operation of ClassA.
- Because ClassB is now a part of the definition of ClassA, modifying the system becomes more complex.

2e) ________________________

- Occurs when operations pass long strings of data arguments.

- The “bandwidth” of communication within classes and components grows and the complexity of the interfaces increases.
- Testing and maintenance are more difficult.

2f) ________________________

- Occurs when one operation invokes another.
- This level of coupling is common and is often necessary.
- However, it does increase the connectedness of a system.

2g) ________________________

- Occurs when component A uses a data type defined in component B.
- For example: this occurs whenever “a class declares an instance variable or a local variable as having another class for its type”
- If this type definition changes, every component that uses the definition must also change.

2h) ________________________

- Occurs when component A imports or includes a package or the content of component B

2i) ________________________

- Occurs when a component communicates or collaborates with infrastructure components (e.g.: operating systems functions, database capability, and telecommunications functions).
- Although this type of coupling is necessary, it should limit to a small number of components or classes within a system.

Part C: **Project**
Task: Interface, Component and Deployment Design

1. Complete the interface designs for all the interface objects and if necessary, add pseudocodes/flowcharts/activity diagrams to further explain the system.
2. From the software architecture, explain the components that will perform some processing such as calculations or data manipulations. You should specify the details of these processing using pseudocodes, flowcharts or activity diagrams.
3. Discuss the system configuration for a Web application. Identify the machines (e.g. client and server) for the system, and determine the artifacts and components in these machines. Draw the deployment diagram to illustrate the configuration.