Software Systems Design (2023) Architecture Assignment 2

(20 Points)

You are responsible to complete two individual tasks related to software quality attributes and architectural patterns. Each student has to independently complete the task described below and compile his/her report.

DEADLINE: 12:00am on (Monday) May 15, 2023

Task 1: Architecture Pattern Analysis (8 points)
Take *broker* and *peer-to-peer* patterns for analysis, discuss what possible impacts of these patterns on all the quality attributes from the lecture notes (including *availability, interoperability, modifiability, performance, security, testability, usability*) and from your own choices in Assignment 1, elaborate the rationale for the impacts with illustrative examples, and recommend the candidate tactics as design decisions to address the possible negative impacts of these patterns and the working condition associated with each tactic.

Task 2: Architecture Patterns vs. Quality Attributes & Tactics (12 points)
Pick TWO from the common *component-connector* architecture patterns (i.e. *pipe-filter, client-server, P2P, service-oriented, publish-subscribe, shared-data*) plus other TWO that was neither discussed in class nor in the lecture notes (e.g., *mediator* pattern). Analyze the relationships between these FOUR architecture patterns with your choice and the quality attributes (including *availability, interoperability, performance, security, testability, usability* introduced in lectures and more analyzed in Assignment 1) by identifying their possible impacts on different quality attributes. Use tables (matrix) with rows of Architectural Patterns for each chosen Quality Attribute to show the tactics applied in the pattern (with reference to an example for Modifiability as below), followed by the analysis of the possible benefits and penalties of the impacts on the quality attribute that may come with each pattern.

	Modifiability									
	Increase Cohesion		Reduce Coupling					Defer Binding Time		
Pattern	Increase Semantic Coherence	Abstract Common Services	Encapsulate	Use a Wrapper	Restrict Comm. Paths	Use an Intermediary	Raise the Abstraction Level	Use Runtime Registration	Use Startup-Time Binding	Use Runtime Binding
Layered	X	X	X		X	X	X			
Pipes and Filters	X		X		X	X			X	
Blackboard	X	X			X	X	X	X		X
Broker	X	X	X		X	X	X	X		
Model View Controller	X		X			X				X
Presentation Abstraction Control	X		X			X	X			
Microkernel	X	X	X		X	X				
Reflection	X		X							

(2 points for your own choice of patterns, 3 points per quality attribute)

Assignment Deliverable:

The assignment deliverables should be submitted individually. Each deliverable contains about no more than 12 pages (A4 size) plus a cover page.