

NAME: JASON TAN THONG SHEN

STUDENT ID: 20060534 (Perth ID) and 700034863 (Miri ID)

EMAIL: 20060534@STUDENT.CURTIN.EDU.AU and 700034863@STUDENT.CURTIN.EDU.MY

QUESTION 1

ANSWER:

i)

Major problem with normal synchronous calls:

Synchronous calls are blocking calls in which they do not return anything until the change has been fully applied or an error has been detected which will affect the speed and resource intensity. Blocking I/O operations are used by synchronous applications, where one operation must finish before the next may be started. A laborious process like a database query will block all other threads since synchronous activities take place one at a time. **If server calls back to large number of clients with normal calls, it will be slow and resource intensive.**

Synchronous calls can result in **slower receiving time** than asynchronous call. Given how synchronous programs manage numerous requests, this is to be anticipated. The subsequent threads in succession are also blocked when a thread is locked.

Enabling synchronous calls also **demand a substantial number of resources**. One thread may accommodate several asynchronous executions, whereas synchronous functions cannot. To handle more requests, you will require additional threads, which is usually too much to handle.

Why asynchronous and oneway calls do not suffer like synchronous calls?:

It is different for asynchronous calls. A polling URL is instantly delivered in response to an asynchronous call, while the request is still being processed. Using a **non-blocking input and output (I/O) protocol** is essential for asynchronous calls. This means that asynchronous call doesn't carry out activities in a hierarchical or sequential manner. Because of the ensuing **parallelization**, an asynchronous software may handle several requests concurrently and independently. The failure of one request has no bearing on subsequent requests. And even before the final task is finished, the application might switch to another one. **Loose coupling** is a term used to describe this type of benefit in software development. Decentralized processes are made possible by communication and/or code that is loosely linked.

One Way Calls are considered the easiest way of executing non-blocking remote calls. The only issue about this is it does not return anything back to the caller which means we wouldn't be able to get any information back after the function is called.

ii)

I will choose Asynchronous instead of oneway. The oneway approach does not return any data back to the caller whereas Asynchronous method can return values which would be beneficial for this newscasting scenario. It also uses non-blocking input and output (I/O) protocol, doesn't carry out activities in a hierarchical or sequential manner, and ensuing parallelization.

QUESTION 2

ANSWER:

The code should be placed in the business tier. Client tiers are never allowed to directly access the data tier without going through the business tier. Business tier handles the logic of the program, it decides which appropriate database to query data from for specific client. Business tier also handles the client authentication to make sure the client is the director of the company before allowing client to access the database data. Data tier's responsibility is to provide the business tier connection to send and retrieve data to the database. The business tier cannot directly access the database itself, it has to go through the database server of the specified database in the data tier first. In conclusion, any code that handles logic should be located in the business tier.