Homework # 1

3-Tier Client-Server Project

For this project, please provide the following:

A) A single README file explaining how the project works.
B) A working project that can compile, build, and run. You can use programming languages such as C/C++, C#, Java, or Python. You will also use database and a web server.

Submit all your files in a single *.zip file to Canvas web site.

Question 1: [100 points]

Client-Server Application:

1- Implement a 3-Tier client-server application which consists of the following:
   a. Client: Using HTTP protocol to communicate to the server-side.
   b. Web Server that contains application server.
   c. Database server: Contains the SQL Data Base Management System (DBMS).
2- The server side resides on the Cloud.
3- Connection is through HTTP and TCP.
4- Client uses a GUI Interface through a web browser or other GUI.
5- Web Server receives and responds to requests from client using HTTP protocol. Example of web server are Apache, Internet Information Services (IIS), or Internet Server Application Program Interface (ISAPI). You can use HTML, XML, Javascript, or any other web-page technology.
6- The web server is a multi-threaded server. Server is to respond to requests from several clients.
7- Application server is implemented within the web server and performs functionalities such as creating web pages, querying database server, or transactions.
8- Due to multi-threads, use Mutex at the server side. Assuming you have a shared variable (or shared file), Use Mutex to protect against simultaneous access from multiple clients (or multiple threads).
9- Database server hosts the DBMS such as SQL, or MySQL.

You can implement the above application on one of the following Operating System:

1) Microsoft Windows.
2) Linux.
3) Azure Cloud Services or Amazon Web Service.

On any of the following commercial Clouds:

1) Microsoft Azure
2) Amazon Cloud
3) Google Cloud

The following are example applications:

EXAMPLE 1:

A 3-tiered client-server model with SQL database where the database resides on the DBMS server while the client is used to query, update, and insert new data into the database in a form of transaction. Application can be for students (university), employees (company), on-line air-line reservation system, retail-shop products and inventory, or a bank financial and customer information.

EXAMPLE 2:

1- File server, or Key/Value store.
   a. For a file server, it implements functionalities such as OPEN, READ, WRITE, SETATTRIB, DEL, MKDIR, RMDIR. The key result is that your servers should be able to handle requests from multiple clients to do concurrent operations (i.e., 5 READ, 5 WRITE, 5 DELETE).
   b. For a Key/Value store, you can use your client to pre-populate the server with enough data (i.e, Key-Value store with data and a set of keys). The composition of the data is up to you in terms of what you want to store there. Once the key-value store is populated, your client can do any operation (such as PUT(Key), GET(Key), DELETE(Key)).

EXAMPLE 3:

Commercial Clouds now supports Face-recognition API, Computer-Vision API, Speech and Voice-API. For example, build a client application that detects faces and send then to the Cloud for recognition, Identification, and storage. Amazon Alexa is one example that uses voice-recognition API.