**OPTION 1: Programming Project**

**Cloud-based Online Catalog System**

**Project Description: (Marks: Maximum score = 100% of a project’s assigned scores)**

The aim of this project is to develop a simple cloud-based online catalog system. The catalog record contains product ID, name, amount in stock and supplier. For example

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Amount | Suppliers |
| 10001 | PCB Board | 100 | LexiCorp Co., Ltd. |
| 10002 | Fan Motor | 20 | LexiCorp Co., Ltd. |
| 10003 | LED Module | 50 | Pegatron Ltd. |

The catalog system must be implemented based on three-tier architecture (client/web – application – database). A server in each tier must be running on different virtual machines. See figure below for the illustration. These server may be hosted on a single cloud provider, but preferably be hosted on two different cloud providers.

Client/Web

Client/Web

respond request

App

App

respond query

Data Synchronization

Database

Database

Minimum functional requirements:

1. Client/Web app should be able to
   1. retrieve,
   2. add,
   3. modify or
   4. delete records stored in the catalog.
2. App-tier must support a failover system meaning if one App server is down/unavailable, a client app can still access data via another App server.
3. Concurrent access must be supported (An App server must support multiple client accesses).
4. Database-tier must achieve some forms of data synchronization. For example if one client app modifies a record on one database server while another client app adds a new record to another database server. Eventually, data (records) stored in both servers must be identical. Manual synchronization is not acceptable.
5. You may implement a Database tier with a basic text file (.txt, .csv, etc.) or any database server of your choice (e.g., SQL)

Sample test scenarios:

1. Two or clients access two different App servers at the same time. One client add few records while the other clients may delete or modify some records. After that two clients retrieve all the records. They should see the same records.
2. While two clients access two different App servers, adding few records. One App server will be suspended. Two clients add few more records. Then the suspended App server is resumed. Both clients retrieve the records. Both should get the same records shown.
3. Similar to the scenario discussed in test#2, but the suspension is to be done to one of the database server.
4. Manual modification to one of the database is to be executed. After that both databases are to be checked to ensure data synchronization.

**OPTION 2: Mini-research Project**

**Project Description: (Marks: Maximum score = 60% of a project’s assigned scores)**

**This research-based project shall include a comparison of the following cloud computing platforms.**

* [**Amazon’s Elastic Compute Cloud**](https://data-flair.training/blogs/aws-ec2-tutorial/)
* IBM Computing
* Microsoft’s Azure
* Google’s AppEngine

**Plus one of the following topics (group 2)**

1. Green Cloud Computing
2. Edge Computing
3. Cloud Cryptography
4. Cloud Analytics

Report’s evaluations include

1. Quality of contents (At least, but not limited to., Introduction, Architecture, Features, Services, Charging, Deployment Models, etc.
2. Related advanced technologies
3. Used case(s)
4. Do those four platforms offer service you choose to do a research in group 2? If not, provide a case study or application of the topic you choose. For example, if you choose to do a research on Green Cloud Computing and those four platforms provide this service, show a study case. If not, provide at least one case study of the selected topic.
5. Other things that are related to the four platforms and selected topic that you think will impress me.