

INFS803 Cloud Computing 2024 S1 Assignment

This assignment is group assignment (4-5 students each group) and worth 50% of the total marks. Each group can either choose Choice A (Design and Implement a Cloud Solution) or Choice B (Cloud Web Application Development and Deployment).

Proposal due: Week 6; full project delivery: Week 12

Note: Each member of the group should have a decent contribution to the project otherwise a percentage penalty will apply on individuals.

Note: Basic skills and tools to complete this assignment will be introduced in the lab session.

Option A: Design a Cloud Solution

Note: For Option A, you should use AWS as a cloud provider when you design your cloud solution.

The [Business Scenario](#) on page 3 states a business background of an education software company. You are expected to read the scenario carefully, analyze the requirements, and design a cloud solution. You can treat yourself as a cloud consultant and try to help a client migrate on-premises infrastructure to cloud. You need to write a cloud solution report for the company and let the client understand what solution they should use and why they should adopt this solution. Your cloud solution should be designed to address concerns and challenges set by the scenario. Your cloud solution also should be cost-effective without over-provisioning. Also, to check you are on the right track, you are expected to write a proposal of your idea of the design.

- **Task 1: Proposal** (one page limit): your proposal is expected to state your understanding of the scenario and initial/basic/draft design **in text**. The purpose of the proposal is to make sure you are on the right track.
- **Task 2: Architect Diagram:** there are some example diagrams you can refer to for example [Top 5 Architecture Blog Posts for Q2 2021](#). You might use [draw.io](#) to finish your diagram or download the official [AWS icon set](#) for building architecture diagrams.
- **Task 3: Solution Report:** Your report should comprehensively document your analysis of the company's requirements and present a detailed and valid cloud solution. This solution must enable our client to seamlessly transition their business to the cloud, ensuring cost-effectiveness without excessive resource allocation. For every component proposed, detailed specifications must be included. For example, if recommending an EC2 instance, specify the type (e.g., t2.large), including CPU, GPU, memory, operating system requirements, the IAM role it possess, and the applicable

security group rules. Merely suggesting the need for an EC2 instance and auto-scaling without these specifics will not suffice for a complete solution. Additionally, you are required to provide a cost estimation for the client (tools available at [AWS Pricing Calculator](#)). You are encouraged to make reasonable assumptions to fill any gaps in the scenario's description, ensuring your report aligns with the outlined marking scheme and structure.

- **Task 4: Presentation:** Group present the project results to the whole class. The presentation should be less than 6 minutes. All members should do the presentation. Video demo and/or slides can be used for presentation.

Hint: Below are some services you might use during your solution design, but you are not limited to these services. You also can refer to the [AWS Well-Architected best practice and whitepaper](#) when you design your solution.

- VPC's, subnets, AZ, NAT servers/gateways, Security Groups, NACLs, IAM
- EC2's, EFS, EBS, load balancer, autoscaling, S3, Database

Potential Report Structure:

- Introduction
- Architecture Solution
- Detail Solution for Each Component (Network, Web Tier, App Tier, Storage, Security Groups etc.)
- Cost Analyze
- How our solution achieves Elasticity, Security, Reliability, and Cost Optimization (no over-provisioning).
- Summary

What you should submission:

1. Proposal in PDF format
2. Solution Architect Diagram in PDF format
3. Analysis and Solution Report in PDF format

Please submit the proposal by Week 5 and then submit ALL the files using a single .zip file by Week 12.

Business Scenario

A start-up education software company has an existing knowledge sharing software application. The application manages student information and enables sharing of documents, books and videos to facilitate learning. The locations of the main users of this application are in Australia and New Zealand. Currently, this company uses on-premises IT infrastructure to host a Three-Tier application architecture while intends to use cloud solutions to support its rapid growth.

Current Production Environment

Web (Client) Tier:

- 4 physical servers (4 CPUs / 16-GB memory)
- Red Hat Enterprise Linux 7.5
- High Availability Proxy load balancer used to balance traffic between the web servers

Database Tier:

- 2 physical servers (8 CPUs / 32-GB memory / 5-TB storage)
- MySQL 5.7.22 database cluster
- DBAs access and manage the database, but no RDMBS or advanced configuration is required.

Application (Backend) Tier:

- 4 physical servers (8 CPUs / 32-GB memory / 5-TB storage)
- Red Hat Enterprise Linux 7.5
- High Availability Proxy load balancer used to balance traffic between app servers

The company expects the new cloud platform to host both test and production environment. The test environment is exclusively utilized by the developers for testing purposes, while the production environment is intended for user access to services. The production environment must at least match the current in-house environment and should also take into account potential future increases in demand. For the cloud solution, this education software company proposes the detailed requirements including:

Component requirements:

- **Database Tier:** Should be highly available.
- **Application Tier:** The architecture should be flexible and be able to handle peak traffic; Internet access is required for patching and updates without exposing the servers.
- **Web Tier:** The architecture should be flexible and be able to handle peak traffic; Application administrators need to be notified by email if there are over 50 “400 HTTP bad request” errors per minute in the application.

- **Storage:** User assets such as documents and pictures should be kept for five years. Note: these files are rarely requested after three months but occasional access is still required.
- **Network:** Expose a single point of access (DNS) to your application and High availability across zones (load balancer). The load balancers should support HTTP, HTTPS, TCP protocols. Services including Regions, VPCs, subnets, NAT, and Availability Zones provided by AWS should be considered.

Security requirements:

- Traffic Control: Web Tier should allow traffic only from Load Balancer; Application Tier should allow traffic only from Web Tier/Load Balancer; Database Tier should allow traffic only from Application Tier.
- Servers should not be directly accessible from the internet, and should consider how to avoid DDOS attack.
- Data should be protected, and disaster recovery plan should be considered.
- The applications in EC2 should access S3 and AWS RDS without exposing the credentials.

Marking Scheme

Task Type	Items		Max	Marks
A	1. Proposal (1 page max; marked based on writing quality and detail level)		5	
	2. Solution Architect Diagram		5	
	3. Analysis and Solution Report	Reasonable analysis, assumptions, and overall description	5	
		Architecture Solution	5	
		Detailed Solution for Each Component	12	
		Cost Analyze	5	
		Description on how the solution achieves Elasticity, Security, Reliability, and Cost Optimization	8	
	4. Quality of presentation (oral + slides)		5	
	<i>Total</i>		<i>50</i>	

Option B: Cloud Web Application Development and Deployment

Note: For Option B, you can use any cloud provider when you deploy your application. You also can use more than one cloud provider. Try to use a free tier when you deploy your application. Cloud providers such as Azure, AWS and GCP have free credit to new starters or students. You are expected to do some research and choose the suitable service.

For Option B, you are expected to **develop a web application and deploy the application in the cloud**. You are free to devolve any web application if it meets the following requirements.

Requirement:

- **Application Architecture:** Your web application should be a Three-Tier application, which has a backend layer, database layer and client layer. Two-Tier application (backend layer and client layer is acceptable but you will lose partial marks). You should let markers to access your backend layer without using client layer.
- **Decoupling Services:** Your web application should decouple clients and backend service by Restful web services. You are expected to develop at least functional 10 RESTful services such as /login, /register, /upload_file including at least 3 post method Restful API. You are expected to list all RESTful services and access URLs on documentation.
- **Deployment on cloud:** You are expected to deploy your client side and back end (Restful API) on cloud. You can choose [AWS EC2](#), [Azure Web App](#), or some similar services that can host your client side and backend side. Your client side and backend side should be accessible by markers. Markers can use public URLs to access your client side and backend side (Restful API) from the time of your submission until you receive notification that your assignment has been marked.
- **Cloud Database and Storage:** The database of your web application can be any cloud database (There are some free cloud databases or free tier database). There is an example for free use of NoSQL database [Firebase](#). You can choose SQL databases such as [AWS RDS](#), [Azure SQL Database](#). You are expected to use one of cloud database and you are not allowed to set up local database in VM such as MySQL installed in EC2. The assets for web applications such as pictures, files, etc. should store on cloud storage services such as AWS S3, Azure Blob Storage.
- **Languages/Framework:** You can use any languages/framework to develop your web application such as C#, PHP, .net, node.js.
- **Collaborate and Source Code Delivery:** You are expected to collaborate work within a Git repository and push your work to any cloud-based git service such as GitHub, GitLab, Azure Repos. You should not apply Git history-rewriting related

operations to your repo and markers would access your repo and see group members' contributions.

- **Bonus:** You will receive bonus marks if you integrate CI/CD into your deployment process. This means that you won't need to manually upload your artifacts to your server. Instead, any updates made to your GitHub repository will automatically trigger the deployment to your server.

Note: This is an alternative option for using serverless architectures to create Restful API without backend server (see Figure 1). You also can get full marks if you want to apply serverless architecture such as using AWS Lambda service. The cost for using serverless architecture is higher than a general virtual machine but it also has the free tier to use.

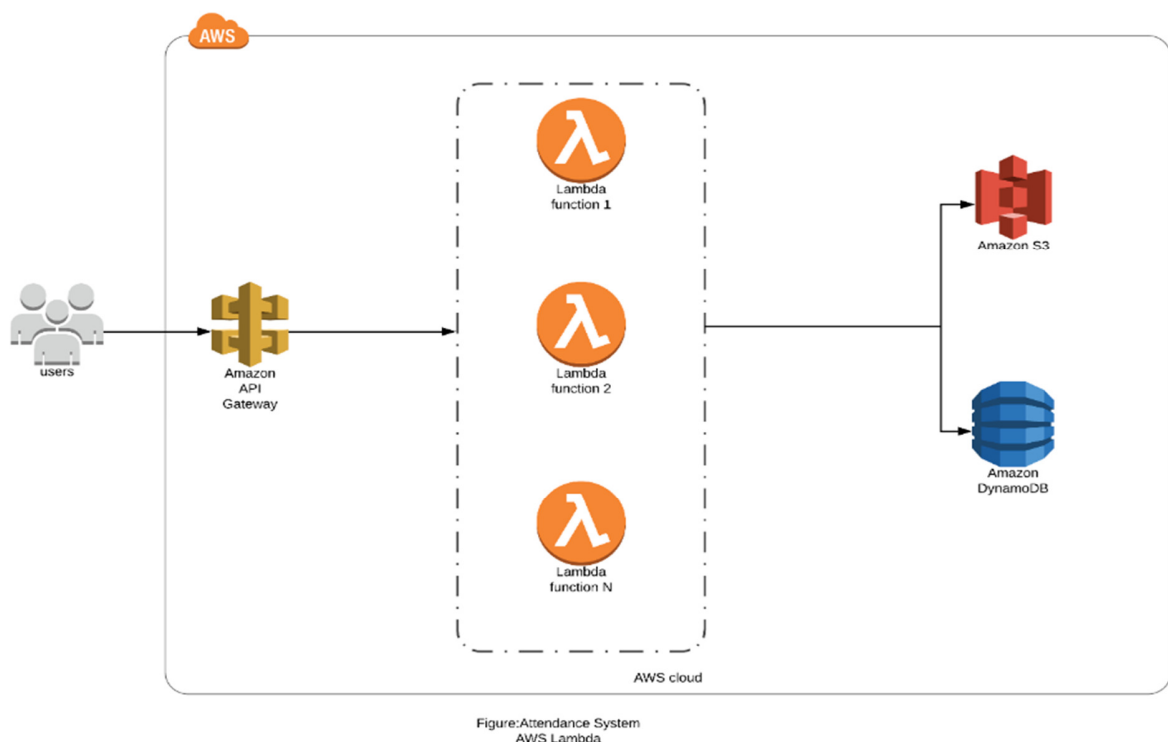


Figure 1: Serverless Architecture

You need to submit a one-page proposal. For proposal, you are expected to state the initial design of your web application (what functions the app have), which language/framework you will use to develop the web application and what cloud services you will use for database and storage and the cloud services you will deploy your application.

You are required to submit comprehensive documentation for your web application. This documentation should encompass several critical sections, including an introduction to your project, project planning, design, a work log, **details on cloud implementation**, a user manual, and a list of RESTful APIs detailing interactions with your backend without a web client. The primary aim of this assignment is to deepen your understanding of cloud services and their

practical implementation within your project. It is crucial to focus on the cloud services utilized and elaborate on how these services are implemented, providing evidence of cloud service usage such as cloud-based databases and storage solutions. Detailed explanations on the setup and utilization of these cloud services are necessary to avoid loss of marks. For instance, if your project is deployed on an EC2 instance, you must describe the process of setting up the EC2 instance and configuring the environment within it. Additionally, the first page of your documentation should prominently feature your Git Repository URL (containing the source code of your application) and the public URL of your web client.

Potential Report Structure:

- Introduction
- Project Planning
- Project Design
- Work Log
- **Details on cloud implementation**
- User Manual
- RESTful API List
- Summary

What you should submission (Option B):

- Proposal in PDF format
- Documentation in PDF format (Make sure your Git Repository URL and URL access to your web client can be found on the first page of your documentation)

Marking Scheme

Task Type	Items		Max	Marks
B	Proposal (1 page max; marked based on writing quality and detail level)		5	
	Size (the number of functions) and Quality		10	
	Implementation	Server-side code and deploy on cloud	4	
		Web client code and deploy on cloud	4	
		Cloud database deployment and cloud storage deployment	5	
		Collaborate work within a Git	2	
	Documentation		15	
	Quality of presentation (oral + slides)		5	