2. Cloud Services Exercises

Table of Contents

Overview

Task description

You will add additional functionality to your REST API application by using a variety of cloud services. You will also make your application stateless by using cloud persistence services to store data.

This assessment item is eligible for the 48 hour extension. You can read more information about this on the HiQ website (https://qutvirtual4.qut.edu.au/group/student/study/assignments/submitting/late-assignments-and-extensions) .

Unit Learning Outcomes assessed

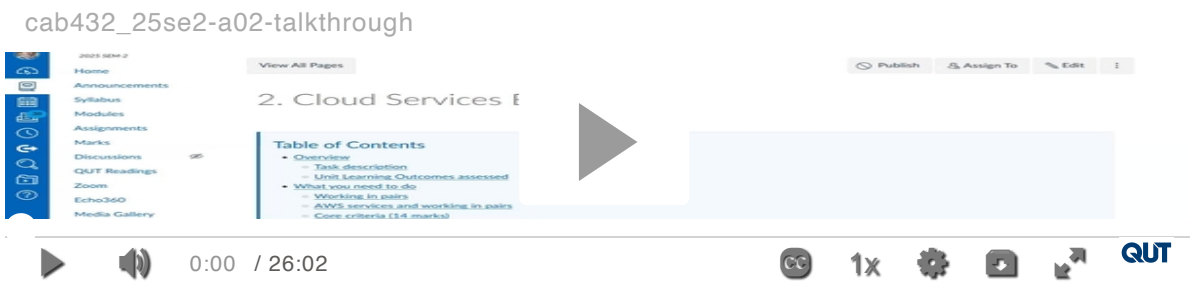
- ULO1 Discuss the elastic nature of cloud technologies and business models and their application in technical, commercial and sustainability contexts.
- ULO2 Analyse application design and implementation requirements to select suitable cloud infrastructure and software services from a range of XaaS offerings.
- ULO4 Design and implement scalable cloud applications using industry standard languages and APIs, deployed on a public cloud infrastructure and leveraging a range of cloud services.

Weighting	Group or Individual	How I will be assessed
30% of final grade	Individual or pairs	Grade out of 30

Talk-through video

In this video, Jake talks through the contents of this page.

Note: this video was recorded 26 August 2025. If there are changes after that date, the video won't capture the change.



Search Hide transcript

All right. This video is a talk through of Assignment two in c432. You can find this assignment through the module section on Canvas. This assignment is all about adding additional functionality to your existing rest API application that you built back in Assignment one. Here, you're going to use a variety of cloud services, particularly to

What you need to do

This assessment item builds on your REST API application project from Assessment 1. There you laid the foundations for this assessment by building the application logic around two kinds of data with different handling requirements. In this assessment you will move that data handling into cloud persistence services. You also incorporated application logic for handling multiple users. In this assessment you will make use of a cloud identity service for maintaining user identity information along with authentication and authorization.

You are free to add additional functionality to your application or to replace your application. Aside from where those might interact with the criteria that we specify below, such changes won't affect your grade for this assessment item. However, if your project didn't meet the core requirements in Assessment 1 then some such changes may be required to accommodate the development required for this assessment.

The marking rubric is available on the **CAB432 A02 - Cloud services exercises (Video) (https://canvas.qut.edu.au/courses/20367/assignments/202268)** submission portal.

Working in pairs

For this assessment item, along with Assessment 3, you have the *option* of working in pairs. The complexity of your project will continue to increase; working with a partner will help to compensate for that additional complexity.

- You are free to work on your own if that is your preference. We will apply the marking criteria the same for individuals and pairs.
- You can choose to work from either partner's Assessment 1 as the starting point, you can combine both partners' projects in some way, or you can start from scratch.
- If one partner obtains a formal extension for this assessment item then it will automatically apply to the other partner.
- You are free to change your partner for assessment 3, so you are not "locked in" to your partner.
- We have created the Find partners for assessments 2 and 3 channel on Teams to help you find a partner. We suggest that you ensure that you know:
  - what grade your partner is aiming for
  - how and when your partner can meet to work on the assessment
  - what skills your partner has that can complement your own

Please sign up your pair on Canvas (but not if working alone)

- Go to the *People* tab on Canvas on the sidebar
- Find a group like *A2-Pairs N* that has no students in it
- Sign up for the group
- Get your partner to sign up to the same group

If you accidentally sign up for the wrong group then email cab432@qut.edu.au and ask to be removed from the group. **Please provide the group name/number**.

Please don't sign up for a group if you are working alone.

AWS services and working in pairs

In the CAB432 AWS account some services are tied to you username and special steps need to be taken to give your partner access. See **AWS working with a partner on AWS services (https://canvas.qut.edu.au/courses/20367/pages/working-with-a-partner-on-aws-services)** for more information.

Core criteria (14 marks)

These criteria relate to functionality that you will extend in assessment 3. You should consider these your top priorities.

Although there are no criteria related to deployment, it is expected that you will deploy your application on EC2 in order to access the various AWS services required by the other criteria.

Note that in this assessment the core criteria represent a smaller portion of the total grade than in the first assessment. This is because we expect that there will be a wide variety of projects, and not all of the cloud services we have studied so far will be appropriate for all projects. Hence the additional criteria, where you have the choice to attempt criteria that are most appropriate to your project, make up a larger portion than in the previous assessment.

Data Persistence Services (6 marks)

Your application makes relevant use of two distinct cloud data persistence services from separate categories below:

- Object storage (S3)
- NoSQL databases (DynamoDB)
- SQL databases (RDS)
- Block and file storage (EBS, EFS)

Each category is worth 3 marks.

If you made appropriate choices for you two kinds of data in assessment 1 then these should naturally map onto two of these services. For example, a video transcoding app would probably use S3 for storing video files and either DynamoDB or RDS for metadata related to the videos.

Please note the following:

- With S3 you should not use public buckets for client access. Instead either use pre-signed URLs or handle requests for objects through your server.
- We have a relatively small cap on the number of RDS instances. Please **tag your RDS instances** with key `purpose` set to `assessment-2` and `qut-username` set to your full QUT username like `n1234567@qut.edu.au`. Teaching staff may need to delete old or improperly tagged RDS instances to make room for students using it for their assessment.

Authentication with Cognito (3 marks)

You will need to make relevant use of AWS Cognito for user identity management and authentication and integrate it with user functionality that you created in assessment 1.

You must implement the following features:

- User registration, including submission of a username, email, and password
- Email-based confirmation of registration
- User login using a username and password, returning a JWT upon successful authentication

Statelessness (3 marks)

In the next assessment you will implement horizontal scaling for your application. In preparation, you need to make your application stateless, with no exclusive data storage other than in cloud persistence services:

- All persistent data for your application is held in the cloud persistence services you have set up.
- Your application will tolerate the loss of any persistent connections (eg. websocket connection)
- Your application state (i.e. the state of the data in the persistence services) will remain consistent if your application is stopped at any moment. Basically, your application will function correctly if restarted with a fresh container/EC2 instance.
- If state is used (eg. a persistent connection for progress reporting) then the application will gracefully handle the loss of such state, or you have a convincing argument that a stateful design is required (eg. strict low latency or real-time requirements).

DNS with Route53 (2 marks)

In this assessment, you will configure a DNS record for a subdomain of `cab432.com` using a CNAME that points to the EC2 instance hosting your application. This setup prepares for Assessment 3, where you will use cloud services to add TLS to your public-facing server (i.e. implement HTTPS). That will require a valid TLS certificate linked to a specific domain name, which is why the subdomain configuration is necessary now. Later, you will update the DNS record to point to a service that provides TLS.

Additional criteria (16 marks)

These should have lower priority than the core criteria above. We have provided ten options but you do not need to attempt all of them. Not all of the additional criteria are weighted the same. Keep in mind that we will stop marking once we have considered enough additional criteria to account for 16 marks, regardless of whether you have earned the full 16 marks. There is also an open-ended *additional* that requires approval by the unit coordinator.

You cannot achieve more than 16 marks from these tasks. We will mark only those that you explicitly tell us to consider. You should choose the most appropriate for your application and those you will achieve the best outcome for.

More details are given in the marking rubric. Be sure you are completing the tasks in such a way that the marking rubric is satisfied.

It is not expected that you can respond to all additional criteria as several of them depend on the details of your application.

Parameter store (2 marks)

This criterion is about appropriately using Parameter store for storing relevant data for your application. For example,

- Application URL (often required by the front end for accessing your app's API)
- External API URL or other information

Secrets manager (2 marks)

This criterion is about appropriately using Secrets manager for storing relevant data for your application. For example,

- External API access keys
- Database credentials

In-memory caching (3 marks)

This criteria is about your appropriate use of in-memory caching for database queries or external APIs using memcached on Elasticache.

You should have a convincing reason that the data you are caching will be accessed frequently. This does not have to be true *now* but it should be true in an imagined wide-scale deployment of your application.

Infrastructure as Code (3 marks)

For this criterion you should aim to deploy all AWS services via IaC mechanisms. That includes infrastructure as code technologies for deployment of cloud services supporting core and additional criteria. We will **not** assess IaC use for deploying services related to assessment 1.

You can use Terraform, AWS CDK, or CloudFormation. For other technologies, please ask the teaching team.

Since using Docker compose for deploying multiple containers and IaC for EC2 were evaluated in assessment 1, this criterion only applies to services beyond these two cases. You can still use Docker compose if you like, but it will not count towards this criterion.

Identity management: MFA (2 marks)

For this criterion, you should make appropriate and non-trivial use of additional Cognito functionality: multi-factor authentication.

If you want to use other Cognito functionality, please discuss with the teaching team, as not everything will be possible in our AWS environment.

Identity management: federated identities (2 marks)

For this criterion, you should make appropriate and non-trivial use of additional Cognito functionality: federated identities, eg. Google, Facebook, etc.

If you want to use other Cognito functionality, please discuss with the teaching team, as not everything will be possible in our AWS environment.

Identity management: user groups (2 marks)

For this criterion, you should make appropriate and non-trivial use of additional Cognito functionality: user groups for organising permissions, eg. an "Admin" group that has additional permissions in your application.

If you want to use other Cognito functionality, please discuss with the teaching team, as not everything will be possible in our AWS environment.

Additional persistence service (3 marks)

This criteria can gain you marks for incorporating a third and distinct type of data persistence service from the category list in the **Persistence services** section, above.

There must be a compelling reason why this additional service is required/beneficial for your application; your application must take advantage of functionality that is not available in the other two services and is appropriate for the data that you are storing.

S3 Pre-signed URLs (2 marks)

This criteria is about using S3 pre-signed URLs for direct client upload and download.

Where a client needs to send or receive an object stored in S3, this is done by passing a pre-signed URL to the client which then up/downloads the object directly from S3.

Graceful handling of persistent connections (2 marks)

If your application uses persistent connections, such as server-side-events or websockets, in an appropriate way (eg. to allow for push style notifications or progress reporting rather than less efficient polling) then you need to address how this stateful aspect of your application impacts on the overall stateless design.

- Your application should gracefully handle the loss of persistent connections. Such a loss may be due to an instance of your server being shut down as the application scales in.
- For full marks, your application should show minimal to no degradation in functionality, for example by the client detecting the lost connection and re-establishing the connection (assuming that there is an instance of your server to serve the connection). Note that this means that whichever instance of your server serves the connection it will need to have access to whatever information is required to send to the client (eg. progress information)
- Part marks will be awarded for graceful degradation of functionality that has some effect but does not impact on the basic functionality of the application (eg. progress reporting stops and an error is reported, but the application otherwise functions correctly)

Upon request (3-6 marks)

This additional criteria exists twice in the rubric. You may not attempt it all, or once, or twice. Each are worth three marks. However, each attempt requires approval from a coordinator.

This criteria gives you the opportunity to gain marks for other functionality or aspects that demonstrate a high level of achievement in the project. This excludes things assessed in later assessment items. Be sure to **first** speak to a coordinator if there is something specific that you'd like to do and get additional credit for.

Anti-criteria

These are things that you will need to do later. You might not want to do these now as we will have particular criteria for how they will be implemented in later assessment items. There is no penalty for these other than the additional work that you might make for yourself. **You can ignore these if you like.**

- Multiple server components (eg. microservices)
- Lambdas, container orchestration
- Autoscaling, load balancers
- Cloud services for asynchronous communication
- Edge caching

Technologies and special permission

We will follow the same guidelines as for Assessment 1. The following technologies do not require special permission:

- Technologies that you already have permission for from assessment 1
- All AWS services listed in AWS services available. Note that some of these may not be available until they are enabled by DBS.
- Docker compose, Terraform

Submission

Details can be found at **2.1 Submission Process (https://canvas.qut.edu.au/courses/20367/pages/2-dot-1-submission-process)**

Feedback

Under normal circumstances, you will receive marks for each criterion via a Canvas rubric within 10-15 working days of submission. Click on Marks to see your results. Usually the reason for each choice of mark is self-evident, the marker will include some written feedback about your performance. You should use this feedback to strengthen your performance in the next assessment item.

Moderation

All staff who are assessing your work meet to discuss and compare their judgements before marks or grades are finalised.