

Assignment 3: Serverless Computing

Due Dec 18, 2022 by 12p.m. **Points** 35 **Submitting** a file upload

Available Nov 21, 2022 at 12a.m. - Dec 20, 2022 at 11:59p.m.

This assignment was locked Dec 20, 2022 at 11:59p.m..

Objective

This assignment will expose you to the following AWS technologies: Lambda functions, API Gateway, and DynamoDB (and more!).

Description

Use your creativity to develop a complete and "cool" web application that turns heads.

Requirements

Your application implementation should comply with the following requirements:


1. **Storage:** All persisted data should be stored on DynamoDB and S3. Use each one of them appropriately.
2. **Lambda:** You should use Lambda functions to develop and deploy your application.
3. **Background Process:** In addition to the main web application functionality, i.e., serving client requests, your application should also include a separate process (implemented as AWS Lambda function(s)) that runs in the background and does something useful for your web application (for example, doing some analytics on collected data, reacting to events, garbage collection, or something else).
4. **Deployment:** You may use Zappa or similar tools to deploy your application, but you must not use Zappa to deploy the background process, rather you should work directly with Lambda functions.
5. **Performance Results:** Run convincing performance experiments that track latency and throughput as one important control variable of your web application changes. You are free to choose which variable it is and how it changes as long as it is a relevant choice within the context of your web application. For example, you could show requests per unit of time, or number of functions spawned in order to handle concurrent requests, or something else of this nature that best demonstrates the performance of your application.
6. **Cost Model:** You should create a model for the total AWS costs incurred by deploying your application on AWS. Take all the AWS services you use (Lambda, API Gateway, S3, Network bandwidth, etc.) into account and predict the long-term deployment cost of your application based on the following parameters:

1. Average number of users per month that will be using your application,
2. Average number of invocations per month.

For this project you can use as many AWS services as you want. AWS has many services (such as image recognition, speech detection, geographical services, etc.) that your application could benefit from. Feel free to use them. The idea is to show modern applications can be built quickly by connecting already available services.

Report

A written report is required alongside the technical requirements of this assignment. The specification and contents of the report are as follows:

- One report per group.
- Group number, assignment number, each member's name, and student ID are required at the top of Page 1.
- Include page numbers on the bottom right corner of each page (including the first page).
- A page limit of 5 pages for the report (additional pages will result in mark deductions); here, less is more if you are able to succinctly describe everything.
- Good formatting & legibility (See [here](https://guides.lib.berkeley.edu/how-to-write-good-documentation)  <https://guides.lib.berkeley.edu/how-to-write-good-documentation>) for tips on how to write documentation).
- Include the following sections:
 - **A short description** of your application (something you would say if you wanted to advertise your application to the public).
 - Description on **how to use your application**.
 - The **architecture** of your application; more specifically, the functions you have defined and their interactions in addition to the background process.
 - The **cost model** for AWS costs. Also, predict these costs after six months for 10, 1000 and 1,000,000 users using your model. Discuss your assumptions about the user behavior, such as requests per day, etc.
 - **Two graphs** to show latency and/or throughput (as per requirement 5). Describe the **experimental set up** and provide **insightful discussion** of outcomes.

When you're ready to submit, see [Submission & Demo Instructions](https://q.utoronto.ca/courses/285775/pages/submission-and-demo-instructions) (<https://q.utoronto.ca/courses/285775/pages/submission-and-demo-instructions>).

Document Revisions

Here, we manually track major revisions to this document that occurred after the document was released.

Date	Revision
------	----------

Assignment 3 Rubric			
Criteria	Ratings		Pts
Originality/Creativity Application is different than A1 and A2. Application's purpose is clear and can be justified. Students should be able to explain the design/brainstorming process leading to their final application. For the demo, each group must prepare a short pitch (max 4 minutes) that highlights the key purpose of their system, what services it draws on, what the original idea is and what the implementation challenges were. For full marks, all group members must partake in the pitch.	10 pts Full Marks	0 pts No Marks	10 pts
Ease of Use and User Experience Students must be able to demonstrate how the application works, and what makes it a positive user experience. Application should serve a clear purpose to the user. Application interface should take user experience into consideration, and appropriate design decisions should be made with the user in mind. Students should be able to show that the application is intuitive to new users. For the demo, each group member must be able to provide more detail about the functionality of one feature of the application.	10 pts Full Marks	0 pts No Marks	10 pts
Background Process Background process is implemented using AWS services. Process is useful/serves a purpose for the application that can be justified. Groups must be able to explain the impacts and effects of the background process, and show it in action.	5 pts Full Marks	0 pts No Marks	5 pts
Technical Complexity Application should have approximately the same or more technical complexity as assignment 1 and 2. Students should be ready to explain each component, and what makes it technically challenging. Multiple AWS components should be used, as well as new AWS features that have not been used in A1 and A2. For the demo, the group must be able to explain the experimental set up of their performance measurements, recreate the performance graphs in their report, and provide meaningful discussion on the results.	10 pts Full Marks	0 pts No Marks	10 pts
Report See A3 report page for more details.	10 pts Full Marks	0 pts No Marks	10 pts
Total Points: 45			