CS 310
Scalable Software Architectures
Fall 2024

Final Project (v1.0)

Assignment: Cloud-native app using AWS services
Submission: via Canvas (not Gradescope)
Policy: Individual or Teams (up to 3 members)
Complete By:
Milestone #01: Tuesday Nov. 19th @ 11:59pm CST
Milestone #02: Friday Dec. 6th @ 11:59pm CST (no late submissions)

Overview

The final project is a chance to work on a problem of interest to you, and explore features of AWS you want to learn more about. The project is intentionally open-ended, with the following restrictions:

1. *You must use AWS*
2. *You must have a client-side app of some kind to demo your work, but the client is not the main focus of the project --- this is not a project about client-side frameworks or user interface design.*
3. *You must have a server-side infrastructure, which represents the main focus of your project. The server-side must consist of a web server / service and at least one other component (lambda function, database, S3, etc.); the architecture can be serverless or serverful.*
4. *The server-side must perform at least 3 non-trivial, distinct operations that we haven't performed already. An example of a "trivial" operation would be the server retrieving data from a database and sending it back to the client; that's too easy. Another example is uploading data to the database or S3 --- we've already done that. You can have trivial operations in your web service, but you need at least 3 non-trivial operations. Examples? Compute something on the server (ML training), interact with another API (CTA / L tracking, ChatGPA, Spotify), or manipulate the data as you upload / download (e.g. compress, decompress, encrypt, decrypt, etc.).*

Here are some examples of final projects that are not valid:

a. Building a web site that returns web pages --- where's the computation?
b. Building a web service that pulls data from a database --- what's the computation?
c. Building a web service that performs basic mathematical computations --- too simple.

Coming up with an idea for a final project can be hard… Are there components of AWS that we didn't talk

about, or we talked about but didn't work with in the projects? A good example are the machine learning technologies in AWS, such as SageMaker. Or large-scale search technologies such as EMR (Elastic Map Reduce). We'll talk about how Twitter is built upon SQS (Simple Queueing Service), build a simple app around SQS and investigate the technology? Rebuild one of our projects using SQS?

Other ideas… You could rework project 04 using microservices instead of lambda functions, e.g. an authentication service, a compute service, an upload service, and a download service. Or you can work with the CTA online bus tracker API, a free service that provides real-time bus information, and build a web service around this that provides information about CTA buses supporting the Evanston campus. A similar online API exists for the L system.

Or you can extend project 02 with image functionality, e.g. image compression on upload, decompression on download, and resizing / rotating functions? Or extend project 03 with other PDF functionality, such as merging separate PDFs into one, deleting pages from a PDF, etc.

Milestone #01: project idea and team members (if any)

You need to decide two things by Tuesday November 19th: (1) what you plan to work on for your final project, and (2) are you working individually or in a team (max team size is 3). **Every** student in CS 310 then submits this information on Canvas under the assignment "Final Project: Proposal"; this will link to a google form. Read below before submitting…

If you plan to work as part of a team, get together with your team members and select a *team name* to uniquely identify your team --- keep in mind this name may be publicized, so don't embarrass yourselves (or NU) with an inappropriate name. ALL team members fill out the google form and enter the team name; only one team member needs to supply the project details.

You can change your project ideas, team members, etc. up until the deadline for milestone #1 (Tuesday November 19th). After this date, submissions are considered final and no changes are allowed unless confirmed by the staff.

If you do not submit something before the milestone #1 deadline, we will assume you are not submitting a final project and will score the final project a 0.

Start working on the final project as soon as possible, do not wait for feedback on your final project idea. We'll try to provide everyone with some feedback, but with a large class it takes a while and you don't need to wait for our approval to begin (you may need to adjust based on our feedback, but as long as you are meeting the requirements given on page 1, your project will be approved).

Milestone #2: submission of final project

Submissions will be collected via **Canvas** (not Gradescope). You will submit the following 3 items; if you are working in a team, please pick one team member to submit these items --- do not have all team members

submit otherwise we don't know which one to grade. Deliverables to submit:

1. A 5-10 minute video with an overview of the project followed by a demo. Every team member must participate in the video. Submit a link to your video Canvas under the assignment "Final Project: Video". You'll get an email confirming the submission, please check that email to confirm the link works; if the link doesn't work please submit again.
2. A "Project Description" document (2-3 pages is sufficient) overviewing the project, with 1 or more visuals showing the components and how they fit together (e.g. similar to the diagrams shown in class for projects 01 and 02). Submit a PDF on Canvas under the assignment "Final Project: Description".
3. The source code files, database schema (SQL) files, data files, Dockerfile, etc. for your project. Also create a "readme.txt" file that summarizes how to install your application; we don't need detailed instructions with screenshots, but some instructions on how we could setup and run your server and client if we wanted to. Put all this in a folder (along with your video), create .zip of the folder, and submit on Canvas under the assignment "Final Project: Artifacts"

That's it!