**This document is a DRAFT and details will change based on feedback.**

**AWS Cloud Project Bootcamp**

*Organized by:* [*AWS Ontario Virtual User Group*](https://www.meetup.com/aws-ontario-virtual-user-group/)

**Course Scenario**

A startup company has decided to build their own micro-blogging platform and has hired you to be their first cloud engineer.

The company paid a web-development firm to translate their wireframe designs into a mock web-application for the purpose of demoing to raise capital.

After a successful round of funding, you [the cloud engineer] have been tasked with taking the mock web-application and making it production ready at scale.

The startup company consulted a fractional CTO to help choose some of the technical requirements to place the company on a good technical roadmap:

* The frontend application should be written in Javascript using React (functional components).
* The backend application should be written in Python using Flask.
* That an API specification be defined detailing the exact endpoints required.
* The web application:
  + That the web-application Sshall be deployed to AWS.
  + TThat the web-application takestake advantage of modern-applications cloud services.

The startup company has spent the majority of their funding on hiring you for the next 6 months (but mostly spent the money on marketing and buying a real cool domain) and so you also need to ensure you keep the cloud provider costs as low as possible.

Good Luck! 😂

**Learning Outcomes**

* To learn how to containerize a frontend and backend web application
* To learn how to work with multiple serverless containers
* To learn how to abstract API call with GraphQL
* To learn basic data modeling for NoSQL databases
* To learn how setup a CI/CD pipeline
* To learn how to use Infrastructure as Code (IaC)
* To learn how to offload background jobs to serverless functions

**Time Commitment**

Let’s detail the time a student will need to commit to fully experience this bootcamp over the 9 weeks:

|  |  |
| --- | --- |
| **Task** | **Estimated Time** |
| **Prerequisite Knowledge**  This is strongly recommended knowledge you will want to obtain prior to the course. The time commit can greatly vary so we’ll provide an average amount of time committed | 10 hours |
| **Prerequisite Technologies**  You need to register specific cloud service accounts. This needs to performed before the course starts | 1-2 hour |
| **Classroom time**  Each class is time-blocked for 2 hours.  Let’s also assume you might want to watch back the video | 2-3 hours per week  9 weeks = 18-27 hours |
| **Homework time**  Homework is not necessary to complete, you will be provided multiple challenges to perform on your own. Commit what time you want. | 2-10 hours per week  9 weeks = 18-90 hours |
| **Student Discussion**  In the discord and email we’ll send simple polls. We’ll have office hours where you can optionally attend. | 1 hour per week  9 weeks = 9 hours |
| **\*Per Month Commitment** | 28-69 hours |
| **Total Time Commitment** | 56-138 hours |

**Prerequisite Knowledge**

**Expectations**

The following should be completed before you attend the course.

Complete as many of the below steps as possible to make the bootcamp process smoother, that way you can spend less time worrying about your set up, and more time doing actual learning.

**[1] Cloud Knowledge**

The student is expected to have either obtained the knowledge of the Certified Cloud Practitioner or general cloud knowledge from another cloud service provider.

The student is expected to be comfortable navigating around the AWS Console eg.

* Changing regions
* Using the search bar to find a service

*If you do not have this knowledge, please watch:* [*AWS CLF-C01 (Youtube)*](https://www.youtube.com/watch?v=SOTamWNgDKc)

**[2] Programming Knowledge**

The student is expected to be able to perform basic programming tasks…

eg. functions, variables, loops, classes, conditionals, imports, some complex data types

The student is not expected to have deep knowledge of web-frameworks

The programming languages that will be used:

* Python 3
* Javascript ECMAScript 6 (ES6)
* YAML and JSON files

*If you do not have this knowledge or wish to prepare further, please watch:*

* [*Python for Beginners (Youtube)*](https://www.youtube.com/watch?v=eWRfhZUzrAc)
* [*Javascript Full Course for Beginners (Youtube)*](https://www.youtube.com/watch?v=PkZNo7MFNFg&t=8s)

**[3] CodeEditor Knowledge**

The student is expected to be comfortable working in the Visual Studio Code Editor (VSCode)  with common editor actions eg.

* Committing code to a linked git repository
* Working with indentation level
* Installing VSCode extensions
* Using find, search or replace
* Copying and pasting code
* Basic usage of the bash terminal

The student does not require Gitpod knowledge prior, but there is a linked free course if you want to increase your knowledge prior to the bootcamp.

*If you do not have this knowledge, please watch:*

* [*VSCode Crash Course (Youtube)*](https://www.youtube.com/watch?v=WPqXP_kLzpo)
* [*Gitpod Certification Course (Youtube)*](https://www.youtube.com/watch?v=XcjqapXfrhk)

**[4] Git and Online Version Control Systems (VCS) Knowledge**

The student is expected to be comfortable using git via the terminal and also through the VSCode [Source Control panel](https://code.visualstudio.com/docs/sourcecontrol/overview) to commit, push and pull code.

The student is expected to be comfortable forking a Github repository, and navigating their codebase within Github.

*If you do not have this knowledge, please watch:*

* [*Git and Github for Beginners Crash Course (Youtube)*](https://www.youtube.com/watch?v=RGOj5yH7evk)

**Prerequisite Technologies**

The student is required to register an account with the following online cloud services:

**[1]** [**Github Account**](https://github.com/)

Github will be used to store our application code.

We will be working from an existing minimal application web-application that is divided into a backend and frontend.

A student will be required to fork a copy of the following two repos:

* Backend Python Flask Application (TBA)
* Frontend Javascript React Application (TBA)

**[2]** [**Gitpod Account**](https://gitpod.io/)

Gitpod will be used as our Cloud Developer Environment (CDE).

We will be working in Gitpod to write code as we integrate cloud services into the provided web applications.

Gitpod has a generous free-tier which we’ll be using.

***Why not AWS Cloud9?***

The reason Gitpod was chosen over Cloud9 is to:

* Help reduce free-tier spend on AWS
* Gitpod has a VSCode experience
  + *(developers are generally more comfortable with VSCode)*
* Gitpod can be easily configured to stage the environment with needed development tools.

**[3]** [**AWS Account**](https://aws.amazon.com/)

Amazon Web Services (AWS) will be our Cloud Service Provider (CSPs)

AWS has a free-tier but requires a credit card to activate your account.

This is a Bring-Your-Own-Account (BYOA) cloud project bootcamp.

There is a riskis risk of spendingspend while we are performing this bootcamp.

You will find these potential costs under each respective week.

We will attempt to stay with free-tier services, and provide guidance to watch out for unexpected spend.

**[4]** [**Momento Account**](https://www.gomomento.com/)

Momento offers a serverless cache with a generous free tier.

To build in more scalability and keep our costs down, we will cache our DynamoDB requests using Momento Serverless Cache service.

Registering an Momento account happens through a CLI. So in order to register you must write some code in a bash terminal.

If you find this too challenging, just skip this step and we’ll perform registration on the DynamoDB day. If you can register ahead of time that will expedite the process.

Instructions on how to use Momento and Register an Account

* <https://www.freecodecamp.org/news/serverless-caching-for-your-web-applications/>
* <https://docs.momentohq.com/getting-started#obtain-an-auth-token>

**Project Course Outline**

**Week 0 — Bootcamp Overview and Introduction to Cloud Spend**

In the class we are going to layout the foundation for the entire bootcamp by:

* Discussing the format of the bootcamp
* Going over the business use-case of our project
* Looking at an architectural diagram of what we plan to build
  + Showcase how to use Lucid Charts to build architectures
  + Talk about [C4 Models](https://c4model.com/)
* Run through the cloud services will utilize
* Test that we can access our AWS accounts
* Looking at AWS free-tier and understand how to track spend in AWS
* Launch AWS CloudShell and looking at AWS CLI
* Generating AWS credentials

*Week 1 Possible Spend Considerations*

* You need a credit card to activate your AWS Account
* If your AWS account is older than a year, you will not be eligible for some free-tier services.

**Week 1 — Docker and App Containerization**

In this class, we are going to:

* Create a new Github repo
* Launch the repo within a Gitpod workspace
* Clone the frontend and backend repo
* Explore the codebases
* Ensure we can get the apps running locally
* Write a Dockerfile for each app
* Ensure we get the apps running via individual container
* Create a docker-compose file
* Ensure we can orchestrate multiple containers to run side by side
* Mount directories so we can make changes while we code

*Week 1 Spend Considerations*

* *Detail Github free-tier*
* *Detail Gitpod free-tier*

**Week 2 — NoSQL Databases**

In this class we are going to:

* Have a lecture about data modeling for NoSQL
* Launch DynamoDB local
* Seed our DynamoDB tables with data
* Write AWS SDK code for DynamoDB to query and scan for predefined endpoints
* Create a production DynamoDB table
* Update our backend app to use the production DynamoDB
* Add a caching layer using Momento Severless Cache

*Week 2 Possible Spend Considerations*

* *DynamoDB provisioned capacity*
* *CloudWatch Alarms generated by DynamoDB*

**Week 3 — Deploying Serverless Containers**

In this class we are going to:

* Create an ECR image repositories
* Push our container images to ECR
* Write a task definition file for Fargate
* Launch our Fargate services via CLI
* Test that our services individually work
* Play around with Fargate desired capacity
* How to push new updates to your code update Fargate running tasks
* Test that we have a Cross-origin Resource Sharing (CORS) issue

*Week 3 Spend Considerations*

* Elastic Container Repository spend
* Fargate Compute Spend
* CloudWatch Logs pricing
* Transfer charges (hidden costs)?

**Week 4 — Solving CORS with a Custom Domain and Load Balancing**

In this class we are going to:

* \*Create a hosted zone for Route53 to manage our domain
* Generate a public certificate via ACM
* Create an Application Load Balancer (ALB)
* Create ALB target group that points to our Fargate instances
* Update our application to handle CORS

\*We need a domain name, so you need to either purchase a domain name, or use a free domain name eg. freenom. Domain names take time to create and propogate, so you have mandatory homework or we’ll have an extra half-class just for domains between Week 3 and 4.

*Week 4 Possible Spend Considerations*

* *ALB free-tier and monthly cost of ALB*
* *Avoiding private ACM*

**Week 5 — Image Processing**

In this class we are going to:

* Test our javascript code to use Sharp and process a thumbnail
* Write a Lambda function
* Deploy our Lambda function
* Create an S3 Bucket
* Create a S3 Event trigger to process images upload to S3 and deposit them back in the bucket
* Implement basic file upload to S3 client side

*Week 4 Possible Spend Considerations*

* *Detail Lambda free-tier cost*
* *Detail S3 Bucket free-tier cost*

**Week 6 — CI/CD**

In this class we are going to:

* Write a buildspec.yml to build new images from our Github repository
* Test Codebuild is building and tagging our images correctly
* Write an appspec.yml with multiple lambda for steps
* Manually trigger a deploy with CodeDeploy to fargate
* Create a CodePipeline that will will trigger CodeBuild and CodeDeploy when code changes are push to our Github repository

*Week 5 Possible Spend Considerations*

* *Detail Codebuild costs*
* *Detail Codedeploy costs*
* *Detail CodePipeline cost*

**Week 7 — CloudFormation**

In this class we are going to:

* Write a CFN template for our Cluster and Load Balancer
* Write a CFN template for our CI/CD pipeline

> We might need an extra week because it can be a-lot of work to write CFN.

*Week 7 Possible Spend Considerations*

* *AWS CloudFormation is free but the resources it can provision are not. Review the overall cost of what we are provisioning*

**Week 8 — Modern APIs**

In this class we will use GraphQL and AppSync to modernize our API.

> bulleted list be better detailed when I figure this out 🙃

**Frequently Asked Questions**

**Can I attend this bootcamp if I’m not in Canada or Canadian?**Yes. It’s a Virtual User Group. All cloud services we will attempt to run, will be done in the Canada region.

Canada has the goal of welcoming half-million new immigrants per year by the year 2030.

So you may not be Canadian now, but you could be in the future. 🇨🇦

**When are live classes?**

Live Classes occurs on Saturdays at 1PM ET (Eastern Time)

**Where are live classes hosted?**  
Classes take place via private Zoom meetings that you join through the AWS Ontario Virtual User Group Meetup page.

There is a capacity limit of 100 students for the Zoom call

**How long is each class session?**

Each class is strictly set for 2 hours.  
There should be around 1 hour of work and another hour to adjust the pacing of the class session.

**Meetup.com won’t let me join the group, how do I participate?**  
DM Andrew Brown your email address either on LinkedIn and Twitter, and a link will be sent to you that way.

**What happens if I’m unable to attend a live class?**  
The Zoom classroom sessions are recorded.

They recordings are private and you must join TBA\_LOCATION to gain previous access

**When will previously recorded classes be up?**

Generally same day but could be as long as 48 hours after the last class

**I don’t have an AWS account, can I still participate?**

This is a Bring-Your-Own-Account (BYOA) project bootcamp.  
We cannot at this time provide an AWS Account for you.

**What are office hours?**

Office hours is a designated time where you can drop-in to ask any questions on a group call relating to the cloud project and technologies used.

**When is office hours?**

Office hours are every Wednesday at 1PM ET time.

**Where are office hours hosted?**

Office hours take place via a private Zoom meeting that you join through the AWS Ontario Virtual User Group Meetup page.

There is a capacity limit of 100 students for the Zoom call

**Where can we interact and discuss throughout the bootcamp?**

The AWS Ontario Virtual User Group has a discord.

Which can be found here: <https://discord.gg/bB7XVPvywA>

<https://discord.gg/bB7XVPvywA>

**What happens if I cannot keep pace with the instructional content?**

Classes are timeboxed for an exact amount of time.

Some students may not be able to keep up with the instruction.

In a live-setting there would be support instructors who could individually help students keep pace with class. In a zoom setting this is not possible to replicate this experience.

We check often if students are up to pace with instruction, if a student is consistently falling behind, they may need to switch from active participant to active listener, and then use the recording on their own time at their own pace to complete the instructional content.

Ideally we would want the same session more than once a week, but our capacity right now is one session per week.

**Can I offer to volunteer for the bootcamp in any way?**

For students, if you want to go beyond just participating you can:

* Organize study groups within the discord to review the same materials
* Create adjacent study notes and share them with your fellow students

If you have any ideas feel free to share them with your instructor.

If you’re looking to get “Volunteer” status for the purpose of your résumé this is not something we are offering currently.

**Is there homework?**

Yes, each week, there will be homework.

You can think of them as “stretch goals” that are optional that you can do.

I’ll present a few challenges.

**Will you review our homework?**

I would like to dedicate the first part of each class to let students to show us if they accomplished their homework or share their challenges with us so we can explore.

Office hours can also be utilized for this time to discuss homework challenges.

**Can I share the project that I work on publicly?**

You are encouraged to share your progress publicly.

I would recommend having a public repo, and showing history overtime.

You want to even write any notes you collect along the way in your repository.

**Can I use this project in my résumé?**

Yes, though I would suggest that you make it your own.

The purpose of homework is to help you go beyond the classroom time and so I hope this will help you think of ways you can do that.

***Phase 2 Cloud Project Bootcamp***

This section details how the bootcamp could be improved on the second iteration:

* Week X  — Planning for expansion
* Week X — Operations/Devops

**Contributors**

This section details contributors who helped with the bootcamp

* [**Kirk Kirkconnell**](https://twitter.com/NoSQLKnowHow) — Developer Advocate - Momento
  + Auditing bootcamp outline for grammatical spelling errors
  + Auditing bootcamp for cloud project outline completeness
  + Suggested outlining possible cloud spend
  + Suggested adding time participants will need to complete each week.
* [**Lou Bichard**](https://twitter.com/loujaybee) — OpenUpTheCloud
  + Audited project course outline’s overall composition
  + Clarified prerequisite expectations