

# Cloud Developer Syllabus



---

## Contact Info

While going through the program, if you have questions about anything, you can reach us at [support@udacity.com](mailto:support@udacity.com). For help from Udacity Mentors and your peers visit the Udacity Classroom.

## Nanodegree Program Info

**Version:** 1.0.0

**Length of Program:** 105 Days\*

*\* This is a self-paced program and the length is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. Actual hours may vary.*

## Part 1: Welcome

## Part 2: Cloud Fundamentals

The cloud has become a key enabler for innovation with beneficial features like high availability, unlimited capacity, and on-demand scalability and elasticity. Learn the fundamentals of cloud computing while being introduced to compute power, security, storage, networking, messaging, and management services in the cloud. While learning the fundamentals, you will explore tools and services offered by Amazon Web Services (AWS) through interactive hands-on exercises. By the end of the course, you will have deployed your first website to AWS, and you will be prepared to continue your learning journey in the Cloud Developer nanodegree program.

### Project: Deploy Static Website on AWS

In this project, you will deploy a static website to AWS.

### Supporting Lessons

Lesson	Summary
<b>Cloud Computing</b>	Learn the basics of cloud computing including cloud deployment models, benefits, and popular options.
<b>Foundational &amp; Compute Service</b>	Learn why we need servers, compute power, and security.
<b>Storage &amp; Content Delivery</b>	Learn why we need storage and content delivery in the cloud.
<b>Security</b>	Learn the importance of security in the cloud.
<b>Networking &amp; Elasticity</b>	Learn the basics of networking and elasticity in the cloud.
<b>Messaging &amp; Containers</b>	Learn the basics of messaging and containers in the cloud.
<b>AWS Management</b>	Learn why we need logging, auditing, and resource management in the cloud.

## Part 3: Full Stack Apps on AWS

### Project: Udagram: Your Own Instagram on AWS

In this project, you will develop a cloud-based application for uploading and filtering images!

#### Supporting Lessons

Lesson	Summary
<b>Cloud Basics</b>	Learn key terminology and building blocks of a cloud system. Understand the design paradigm of modern cloud applications.
<b>Develop</b>	Implement a process so you write quality code, working alone or on teams. Unit and integration testing, a better way to git, and how to use packaged dependencies.
<b>Storing Data in the Cloud</b>	Set up and start using a cloud-based relational database for storing user data using AWS RDS. Implement a filestore for media like images using AWS S3.
<b>Building and Deploying</b>	Consume cloud data services (database and filestore) within your server application. Deploy your application using AWS Elastic Beanstalk.
<b>User Authentication and Security</b>	Cloud systems come with a new set of challenges to only allow authorized users access to sensitive information. Learn common mistakes and modern techniques for dealing with security.
<b>Scaling and Fixing</b>	Your work is never over. Cloud systems need to be maintained as dependencies are updated and there is more demand for your service. We explore tools and process to minimize growing pains

## Project: Optimize Your GitHub Profile

Other professionals are collaborating on GitHub and growing their network. Submit your profile to ensure your profile is on par with leaders in your field.

## Part 4: Monolith to Microservices at Scale

Microservices are becoming the default mode of developing and deploying applications at scale. The microservice architecture make it more easy to scale an application to large system and is a great enabler for continuous integration and delivery. Microservices architecture allows independent scaling, independent releases and deployments and independent development so that each service has its own codebase. In this course we will cover the best practices on how to develop and deploy microservices. You will learn topics such as different microservice architecture patterns, independent scaling, resiliency, service replication, service registration and discovery. By the end of this course, you should be able to design and build an application using an microservice architecture.

## Project: Refactor Udagram app into Microservices and Deploy

In this project, you will reuse their existing Udagram application and convert and extend into a microservice architecture.

## Supporting Lessons

Lesson	Summary
<b>Best Practices for Micro-Services Development</b>	In this lesson, you will learn about the best practices used in Microservices architecture so that we can later split our services.
<b>Independent Development</b>	In this lesson, you will split our services - "feed" and "user" into two separate services.
<b>Container</b>	In this lesson, you will run each of our services separately in an individual Docker container.
<b>Kubernetes</b>	In this lesson, you will run your application in a Kubernetes cluster. You will also learn about managing your containers on a large scale.
<b>Service Registration, Discovery &amp; Scaling</b>	In this lesson, you will learn to use ConfigMaps and Secret objects in Kubernetes deployment.
<b>Independent releases and deployments</b>	In this lesson, you will learn to use the Continuous Integration tool - Travis CI.

## Part 5: Develop & Deploy Serverless App

Serverless technologies have become very popular recently because they can increase the speed of development and drastically reduce the cost of running a cloud infrastructure. This course combines the theory of using serverless technologies with the practice of developing a complex serverless application and focuses on learning by doing. You will learn advanced serverless features such as implementing WebSockets and stream processing and learn about serverless best practices throughout the course.

### Project: Serverless Application

In this project, you will create a simple TODO application using AWS Lambda and Serverless framework.

## Supporting Lessons

## Lesson

## Summary

### Serverless Introduction

Introduction to the course, and Introduction to Serverless

### REST API

In this lesson, we will cover the development of REST API for a group of images, how to store data in DynamoDB and build a web app using React.

### Serverless Framework

In this lesson you will be introduced to the framework for Serverless and the various API we will be using.

### Event Processing

In this lesson you will learn about event processing using Serverless. We will focus on processing different event types with Lambda functions.

### Authentication

In this lesson, we cover authentication and authorization using Serverless. We will cover Authentication, OAuth, Auth0 protocols.

### Best Practices

In this lesson, we will cover some best practices with Serverless.

## Project: Improve Your LinkedIn Profile

Find your next job or connect with industry peers on LinkedIn. Ensure your profile attracts relevant leads that will grow your professional network.

## Part 6: Capstone

### Project: Capstone

In the capstone project, you'll build an application of your choice on AWS based on predefined criteria.



Udacity

Generated Sat Apr 18 12:56:50 PDT 2020