# Using the Serverless Framework with Node.js on AWS

## THE STATE OF SERVERLESS

**Fernando Medina Corey**
DATA ENGINEER

@fmc_sea    www.fernandomc.com

# Outline

Prerequisites

Why use a Serverless Framework?

Look at Several Frameworks

Benefits of the Serverless Framework

Concepts we'll be learning

What we will be building & new skills

# Suggested Prerequisites

AWS Account & AWS Basics

JavaScript Exposure - Node.js

Understand 'Serverless' – Introduction to AWS Lambda Course

Command Line Use

# "[The] Serverless Framework"

## The Serverless Framework

**The specific serverless framework we'll be using in this course. (serverless.com)**

## Serverless Frameworks

**A general term for any framework that focuses on the Functions as a Service or 'Serverless' paradigm.**
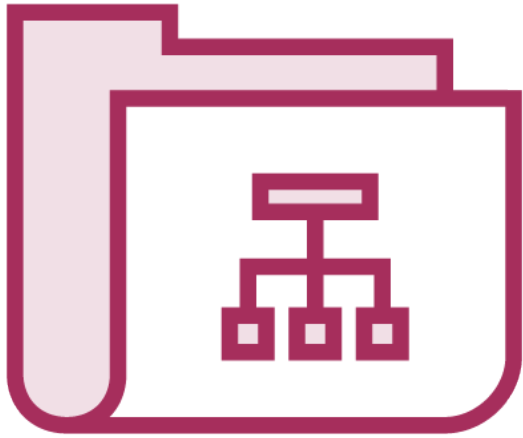
# Traditional vs. Serverless Architectures
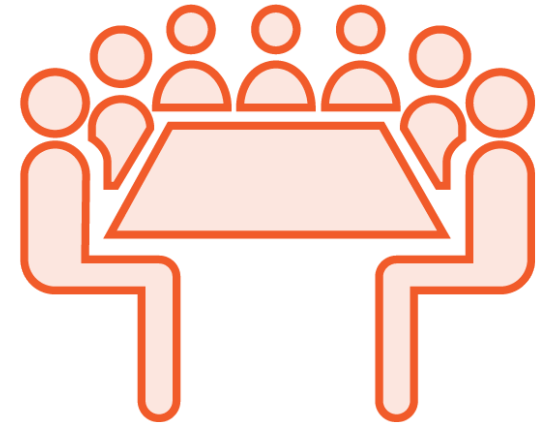
**Traditional Applications**

**Serverless Applications**

# Why Use Serverless Frameworks?

**Standardization and Organization**

**Tooling**

**Community**

# What Serverless Frameworks?



**Shep (Bustle.com)**

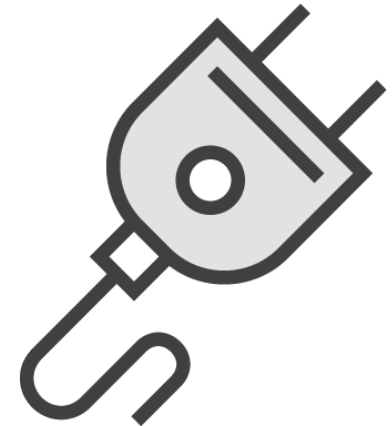**Sparta**

**Zappa**

# Some Criteria for Choosing Frameworks

**Core Functionality**

**Language Support**

**Infrastructure**

**Plug Ins**

# Why the Serverless Framework?

| Frameworks on GitHub | 👁 Watchers | ⭐ Stars | 🍴 Forks |
|---|---|---|---|
| Serverless | 👁 733 | ⭐ 17,952 | 🍴 1,753 |
| Zappa | 👁 186 | ⭐ 4,428 | 🍴 431 |
| Apex | 👁 193 | ⭐ 5,785 | 🍴 362 |
| Gordon | 👁 62 | ⭐ 1,878 | 🍴 124 |
| Sparta | 👁 9 | ⭐ 351 | 🍴 26 |
| Shep | 👁 24 | ⭐ 207 | 🍴 11 |

July
2017

# Why the Serverless Framework?

## Enterprises

- Coca-Cola
- Expedia
- Nordstrom

## Startups

- CrazyLister
- Ibotta

### Serverless Frameworks
### Github Stars - June 2017

**Series**
- Serverless
- Kubeless
- Funktion
- Fission
- Iron.Functions
- Openwhisk

Stars axis: 0, 5000, 10000, 15000

Time axis: 01/07/2015, 01/01/2016, 01/07/2016, 01/01/2017

**June 2017**

http://redmonk.com/jgovernor/2017/07/05/why-serverless-inc-is-crushing-it-right-now/
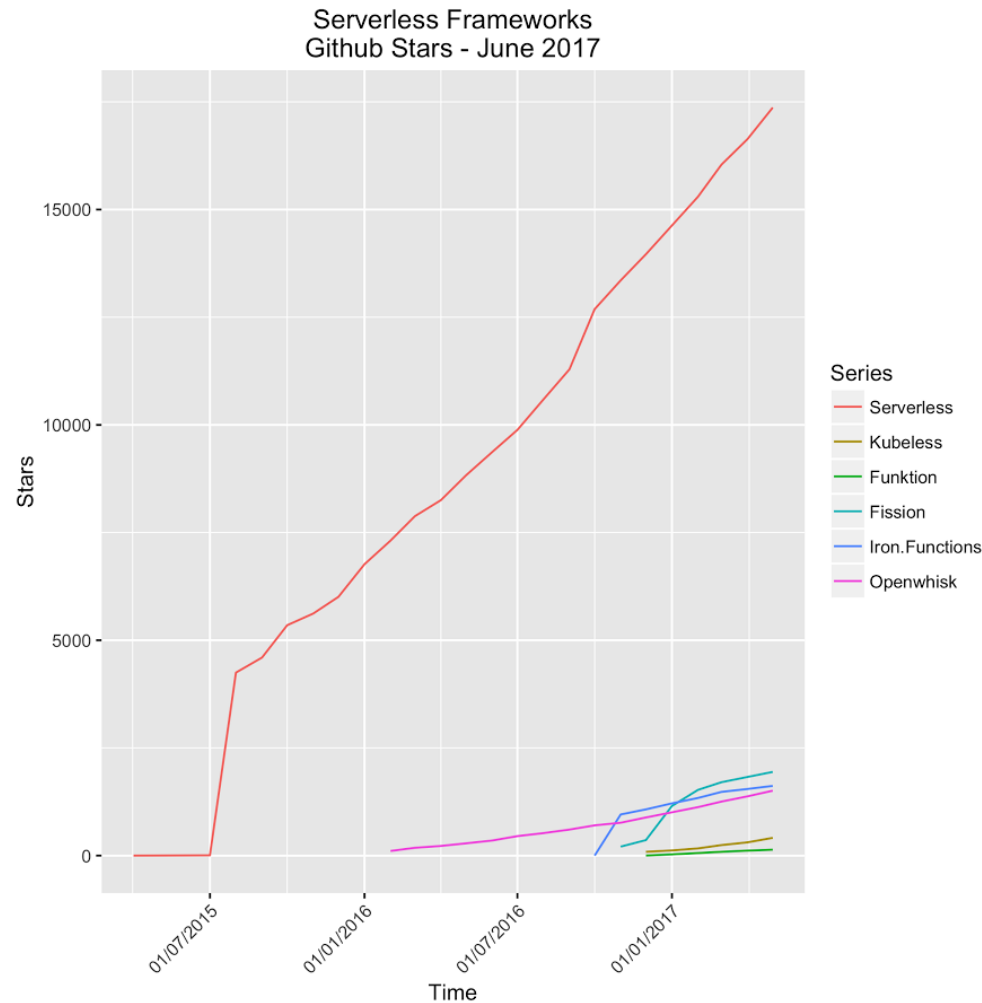
# Key Tooling



**Structured Development**

**Infrastructure Management**

**Reduce Provider Complexity**

**Infrastructure Independent**

# Infrastructure Providers

**Amazon Web Services**

**Google Cloud Platform**

**Microsoft Azure**

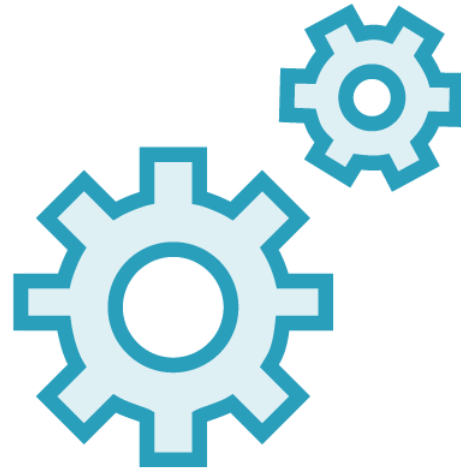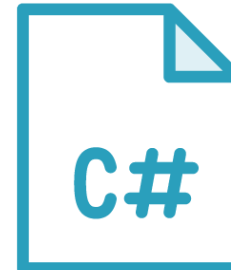**Open Whisk**

# Language Support



Node.js

Python

Java

PHP

C#

# Potential Serverless Drawbacks

**Obscures Backend Infrastructure**

**Permission Hungry**

**Compliance Challenges (e.g. HIPPA)**

**Opinionated Development Process**

# Course Plan & Concepts

# Module Concepts

**Up and Running with the Serverless Framework on AWS:**
*Installation, Application Model and Development*

**Deploying a Text Notification Website with a Serverless HTTP API:**
*Plugins, APIs and Third Party Integrations*

**Scaffolding and Deploying Serverless CRUD APIs:**
*CRUD APIs, Databases and Testing/Debugging*

**What is Multi-platform Serverless?:**
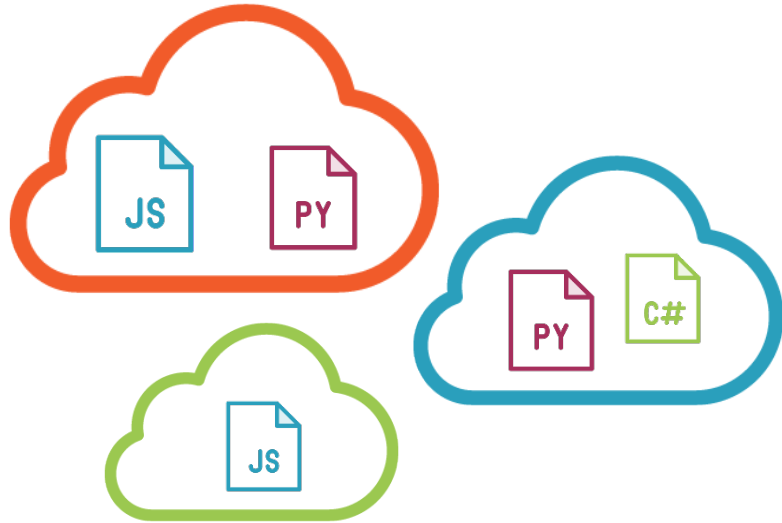*IaaS Risks, Multi-platform/Resilient Architectures, IaaS Agnostic Code*

**Course Scenario**

- Building services

- Technical components

- Internal Tools

# Language & Platform Focus



**Serverless Framework:**
**Polyglot & Infrastructure Agnostic**
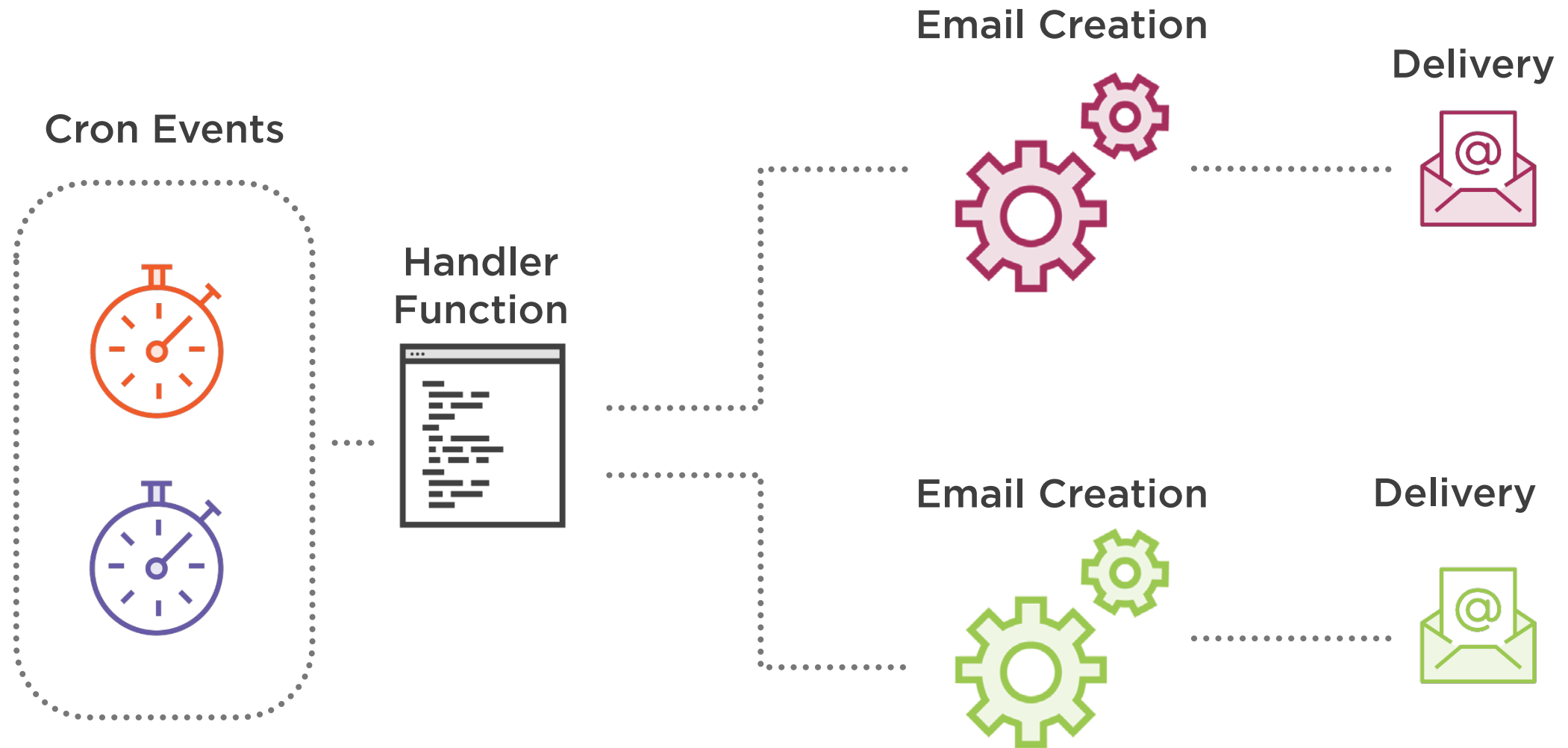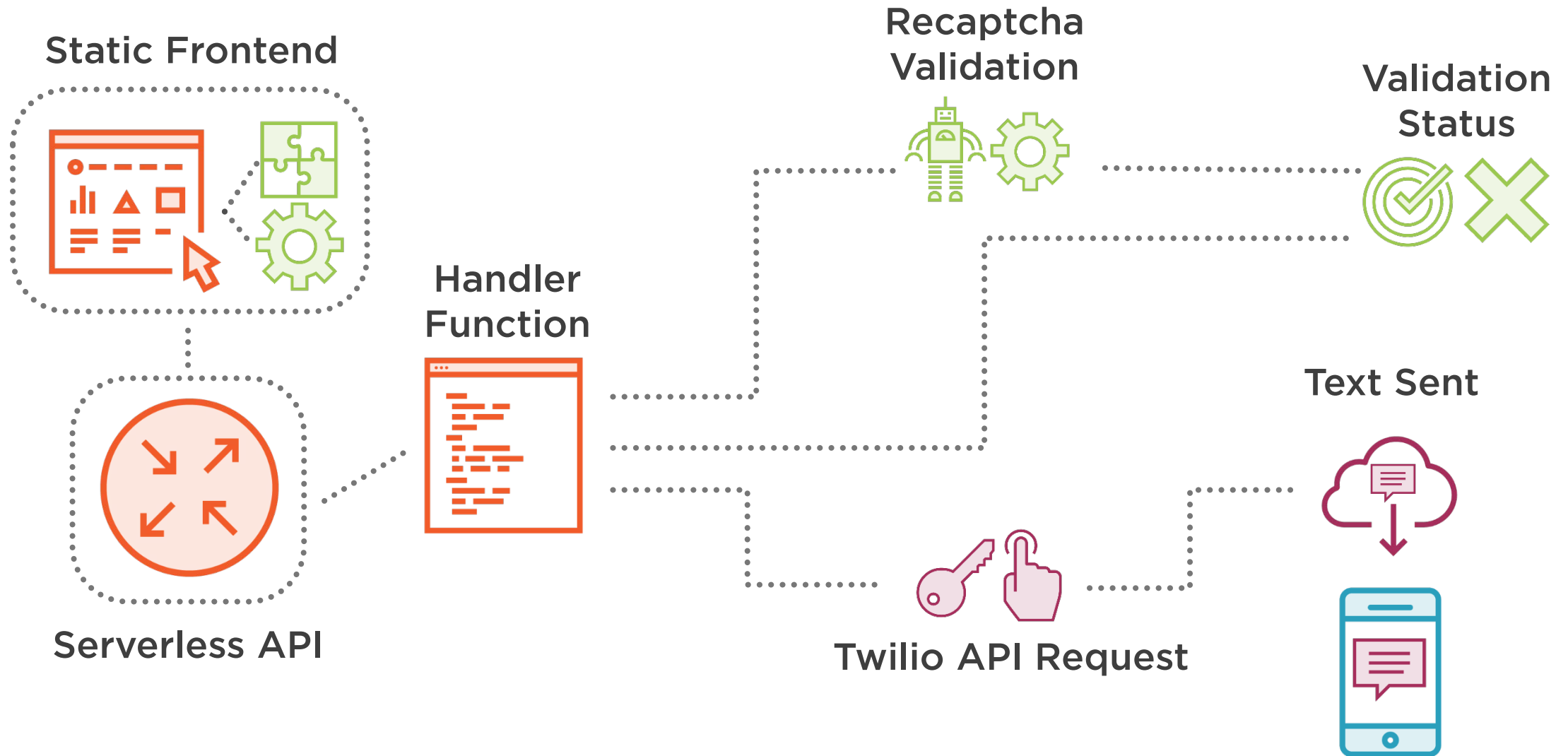
**Focus on Node.js & AWS**
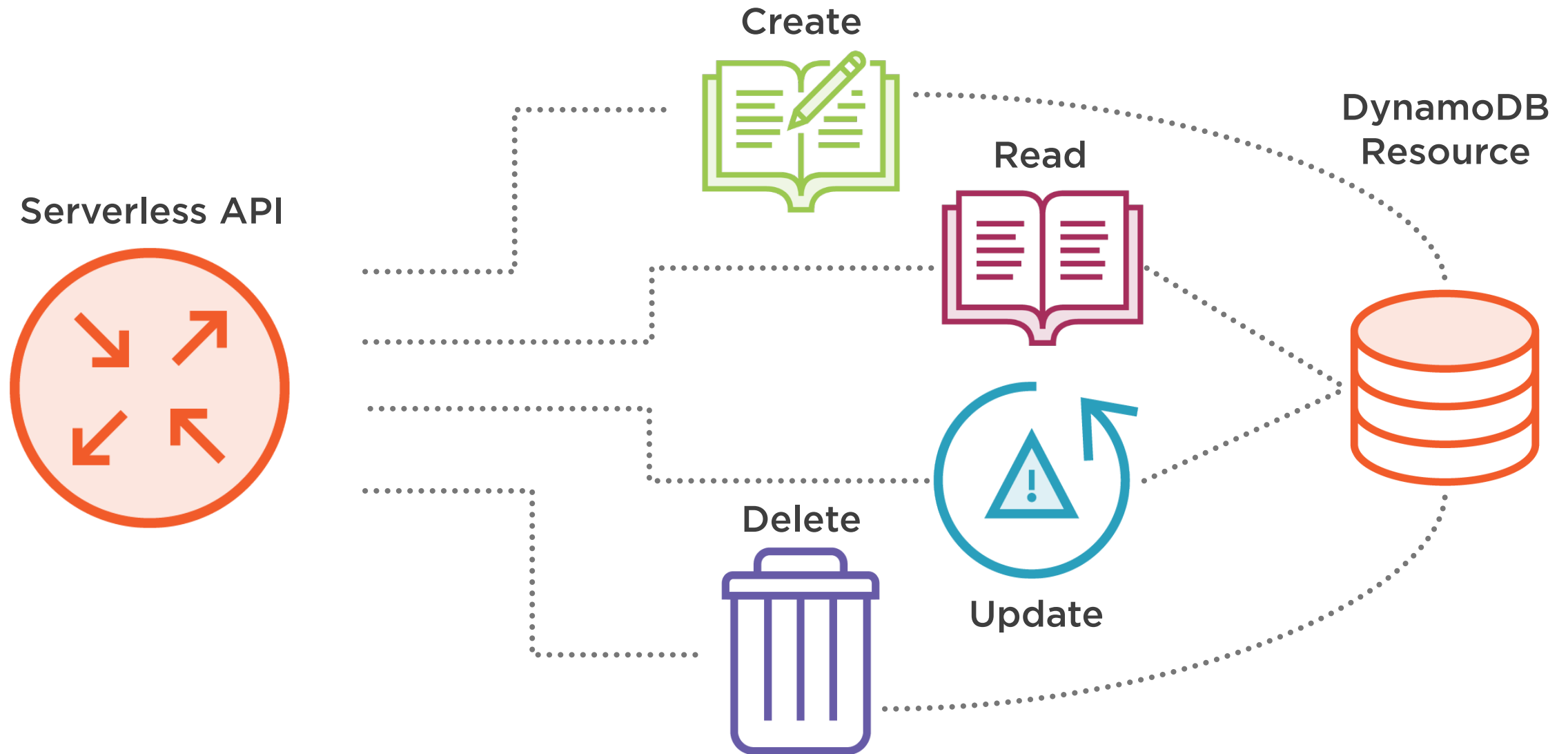
# Course Projects

Up and Running – Email Reminders

# Deploying Our Text Notification Website

**Static Frontend**

**Recaptcha Validation**

**Validation Status**

**Handler Function**

**Text Sent**

**Serverless API**

**Twilio API Request**

Scaffolding Serverless CRUD APIs

# Review

**Understand the Serverless landscape**

**Learned IaaS providers relationship to Serverless**

**Excited to build our new projects and skills**

**Coming Up**
- Serverless Framework Setup
- First Serverless Project