

# **Building a full-stack Serverless Web application with React and AWS**

**Yinlin Chen**  
**ylchen@vt.edu**

**Digital Library Architect**  
**Virginia Tech Libraries**

# Agenda

- Introduction to AWS, Serverless, AWS Amplify, and React
- Create your first React application and setup AWS Amplify
- Setup access controls for your application
- Perform data mutations using AWS AppSync and GraphQL
- Introduction to multiple development environments
- Wrap-up and discussion

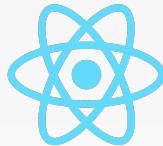
# Introduce Yourself

- Your name
- Your organization
- Your role
- Your experience level with AWS
- Your expectations

# Introduction

- **Amazon Web Service (AWS)**: <https://aws.amazon.com/>
- **Serverless**: Build and run applications and services without thinking about servers.
- **React**: A JavaScript library
- **AWS Amplify**: An open source **JavaScript** library provided by AWS that enables developers to build applications with cloud services on web or mobile platforms.

# Workshop Overview



React



AWS Amplify



AWS AppSync



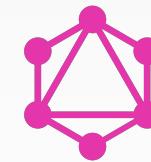
Node.js



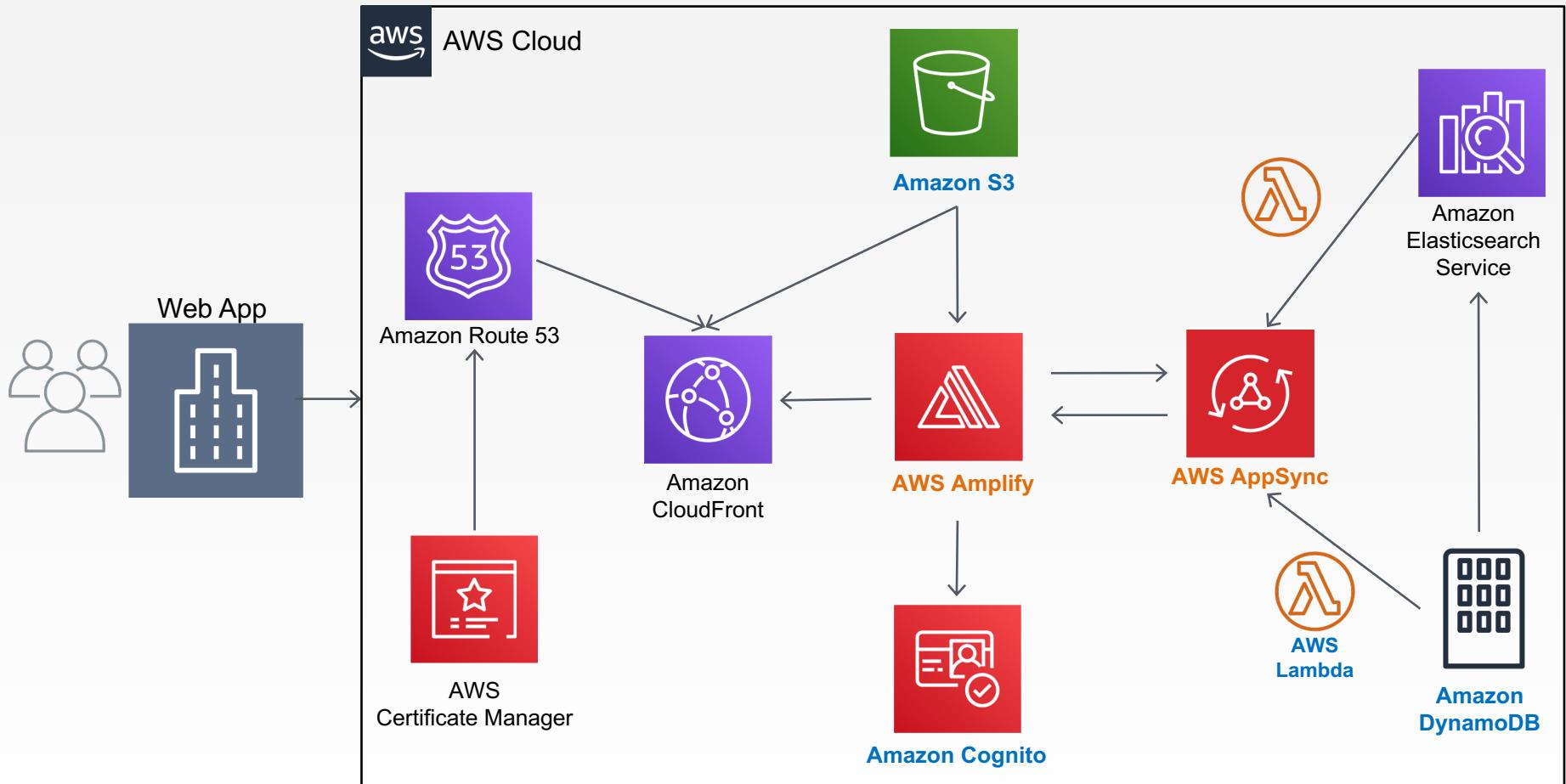
JavaScript



Web App



GraphQL



# AWS Amplify

Everything you need to develop & deploy  
cloud-powered mobile and web apps



## Develop

With the Amplify Framework

Build scalable cloud-powered apps



The Amplify Framework provides a command-line interface (CLI) and library for simplifying mobile and web development.

[GET STARTED](#)



## Deploy

With the Amplify Console

Build, deploy, and host modern web apps



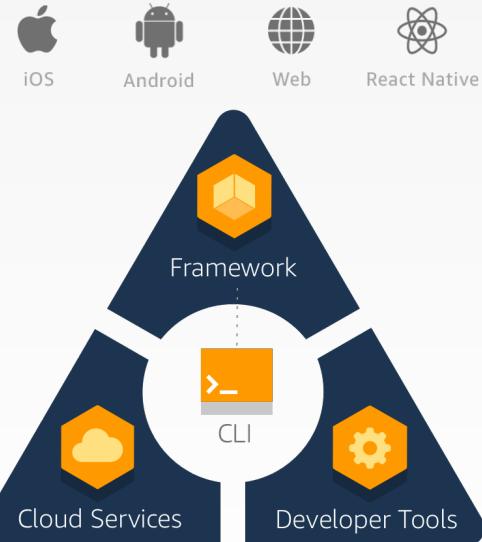
Connect your Git repository to continuously deploy your frontend and backend. Host it on a globally available CDN.

[GET STARTED](#)



# AWS Amplify Framework

- AWS Amplify: <https://aws-amplify.github.io/>
- Library: Connect to AWS services
  - Authentication, API, Storage, and etc.
- Toolchain: Command line interface
  - <https://github.com/aws-amplify/amplify-cli>
- UI Components
  - React or Angular, and Vue
  - [https://aws-amplify.github.io/media/ui\\_library](https://aws-amplify.github.io/media/ui_library)



# Workshop – Section 1

- Create your first React application
- AWS Amplify Setup
- GitHub Setup

# Amplify Command Overview

Command	Description
amplify configure	Configures a new AWS User Profile
amplify init	Initializes a new project
amplify status	Displays the state of local resources
amplify add <category>	Adds cloud features to your app
amplify update <category>	Updates existing cloud features in your app
amplify push	Provisions cloud resources with the latest local developments
amplify env <category>	Setup multiple environments
amplify delete	Delete resources tied to the project

# Workshop – Section 2

- Setup access controls for your application
- Add authentication
  - Email signup
- AWS Cognito
- Configure the React application
- Add E2E tests to app with Cypress
- Deploy the application
  - AWS Amplify Console

# AWS Cognito



## Amazon Cognito

Amazon Cognito offers user pools and identity pools. User pools are user directories that provide sign-up and sign-in options for your app users. Identity pools provide AWS credentials to grant your users access to other AWS services.

[Manage User Pools](#)[Manage Identity Pools](#)

# Cypress

The screenshot shows the Cypress application interface. At the top, there's a navigation bar with tabs for 'Tests' (selected), 'Runs', and 'Settings'. On the right of the navigation bar are buttons for 'Stop' (red) and 'Running Chrome 80' (green). Below the navigation bar is a search bar labeled 'Search...'. Underneath the search bar, there's a section titled 'INTEGRATION TESTS' containing a single test file named 'authenticator\_spec.js'. The main area of the interface is a code editor showing the test script. The script is titled 'Authenticator: Sign In:' and contains a single test case. The test starts with a 'VISIT /' command, followed by a series of 'GET' requests to input fields ('username-input', 'sign-in-password-input'), 'TYPE' commands ('code4lib', 'workshop'), a 'CLICK' command on a 'Sign In' button, another 'GET' request to a sign-out button, and finally a 'POST' request to '/'. The right side of the interface shows a browser window displaying the application's home page with the message 'Hello! This is an AWS Amplify application.' and a 'SIGN OUT' button.

```
Authenticator:  
  Sign In:  
    ✓ allows a user to signin  
      ✓ BEFORE EACH  
        ✓ VISIT /  
          ✓ TEST  
            1 GET [data-test="username-input"]  
            2 - TYPE code4lib  
            3 GET [data-test="sign-in-password-input"]  
            4 - TYPE workshop  
            5 GET [data-test="sign-in-sign-in-button"]  
            6 - CONTAINS Sign In  
            7 - CLICK  
            8 GET [data-test="sign-out-button"]  
            > XHR POST 200 /  
            9 - CONTAINS Sign Out
```

# Connect to Source Code

Amplify Console X

All apps

Documentation ?

Support ?

## Connect app

### Select a Git provider

GitHub 

BitBucket 

GitLab 

AWS CodeCommit 

Deploy without Git provider  
Host your app by manually uploading build artifacts from your local desktop, Amazon S3, or any public URL. 

Cancel Continue

# Add Repository Branch

Amplify Console X

All apps awesometodo

Documentation ↗

Support ↗

All apps > Connect app

Step 1 Add repository branch

Step 2 Configure build settings

Step 3 Review

## Add repository branch

### GitHub

 GitHub authorization was successful.

Repository service provider  GitHub

Recently updated repositories  
If you don't see your repository below, please push a commit and then click the refresh button.  
  

Branch  
Select a branch from your repository.  
 

Cancel  

# Build Settings

AWS Services Resource Groups ⚙

Yinlin Chen N. Virginia Support

Amplify Console X

All apps bookapp Connect branch

Step 1 Add repository branch

Step 2 Configure build settings

Step 3 Review

App settings

- General
- Domain management
- Email notifications
- Access control
- Access logs
- Rewrites and redirects

Documentation ↗

Support ↗

## Configure build settings

### App build and test settings

App name bookapp

Auto-detected frameworks

Frontend framework React

Backend framework Amplify

Testing framework Cypress

Backend deployments

Connect your backend to continuously deploy changes to both your frontend and backend

Deploy updates to backend resources with your frontend on every code commit

Select a backend environment

dev

Select an existing service role or create a new one so Amplify Console may access your resources.

amplifyconsole-backend-role

G

# Review

Amplify Console X

All apps > Connect app

Step 1 Add repository branch

Step 2 Configure build settings

Step 3 Review

## Review

Repository details	
Repository service	Branch
GitHub	master
Repository	Branch environment
yinlinchen/amplify-workshop	

App settings	
App name	Framework
amplify-workshop	Web
Build image	Build settings
Using default image	Auto-detected settings will be used
Environment variables	
None	

Cancel Previous Save and deploy

# Automatic Deployment

Amplify Console X

All apps > bookapp

## bookapp

The app homepage lists all deployed frontend and backend environments.

▶ Learn how to get the most out of Amplify Console 1 of 5 steps complete X

Frontend environments Backend environments

This tab lists all connected branches, select a branch to view build details. Connect branch

**master**  
Continuous deploys set up with **dev** backend ([Edit](#))

<https://master...amplifyapp.com>

Last deployment  
2/22/2020, 11:33:51 PM

Last commit  
update readme | 49d45d9 | [GitHub - master](#)

Previews  
Disabled

Provision Build Test Deploy Verify

# Workshop – Section 3

- Introduction to GraphQL
- Introduction to AWS AppSync
- Add a GraphQL API

# Introduction to GraphQL

- A query language for APIs
- Type system
- Operations
  - Queries
  - Mutations
  - Subscriptions

```
type BestSeller
{
    title: String          #from Book Database
    coverArtUrl: AWSURL   #from Object Store (S3)
    salesRank: Int          #from external web service
}

type Query {
    getBestSellers(week: Int): [BestSeller]
}
```

# Type System

- Type Book {
  - uuid: ID
  - title: String!
  - author: String
  - stars: Int
  - reviews: [Review]
  - inventory: Inventory}

# Query

- type BookQuery {  
    search(q: String): [Book]  
    getBook(uuid: ID): Book  
}
- search(q: “graphql”) {  
    uuid  
    title  
}
- getBook(uuid: “A123”) {  
    uuid  
    title  
    review {stars}  
}

# Mutation

- type BookMutation {  
    addStar(num: Int): Review  
}
- addStar(num: 1) {  
    stars  
}

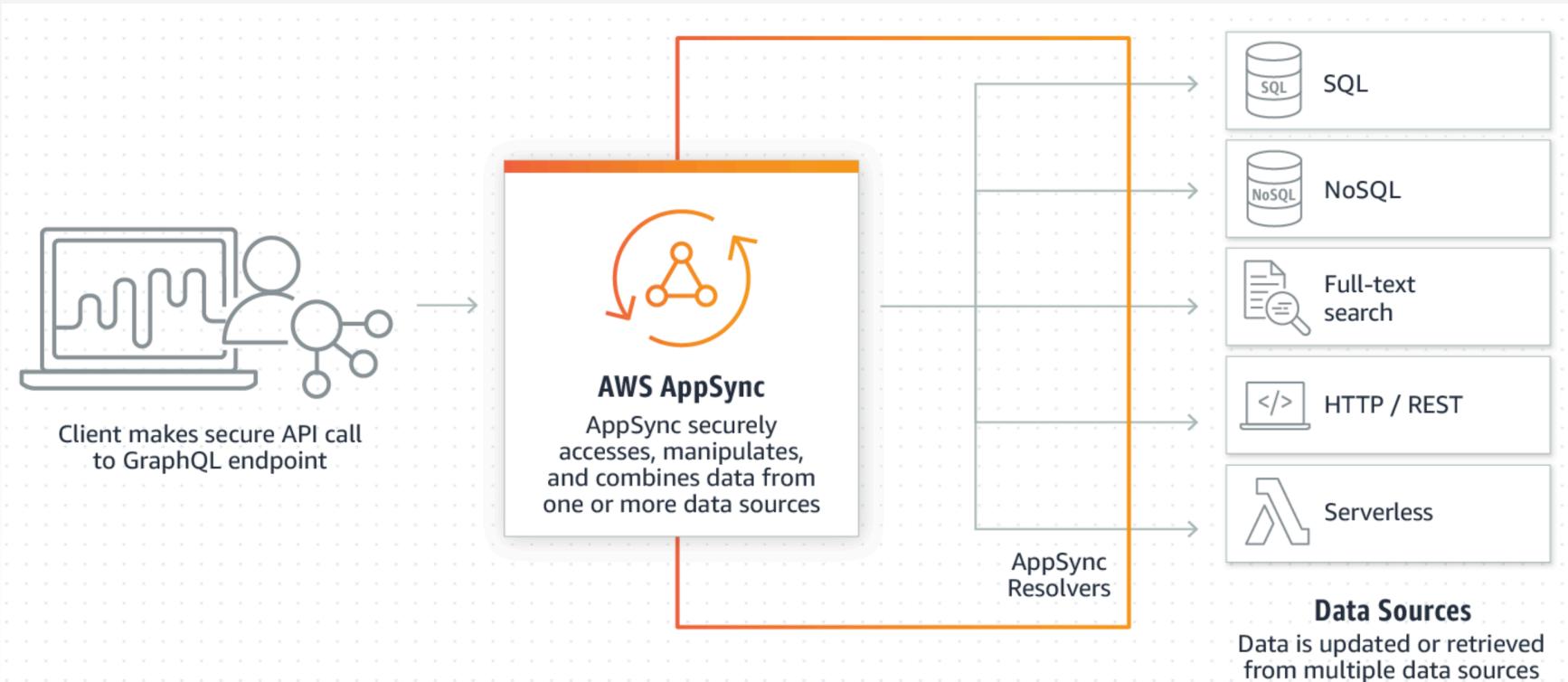
# Subscription

- Type BookSubscription {  
    onReview(uuid: ID): Review  
}
- onReview(uuid: “abcd”) {  
    uuid  
    id  
    stars  
}

# Schema

- schema {
  - query: BookQuery
  - mutation: BookMutation
  - subscription: BookSubscription}

# Introduction to AWS AppSync



# Set up GraphQL in Amplify

- amplify add api
- amplify status
- amplify push

# AWS AppSync Console

AWS AppSync X

APIs

**BookGraphQL-master**

Schema

Data Sources

Functions

**Queries**

Settings

AWS AppSync > BookGraphQL-master > Queries

## Queries

Write, validate, and test GraphQL queries. [Info](#)

Select the authorization provider to use for executing queries on this page:

API key ▾

▶

```
1 # Welcome!
2 #
3 # This is an in-browser tool for writing, validating, and
4 # testing GraphQL queries.
5 #
6 # An example query named "GetPost" might look like:
7 #
8 #   query GetPost {
9 #     singlePost(id: 123) {
10 #       id
11 #       title
12 #     }
13 #   }
14 #
15 # An example mutation named "PutPost" might look like:
16 #
17 #   mutation PutPost {
18 #     putPost(id: 123, title: "Hello, world!") {
```

# Workshop - Section 4

- Perform data mutations for your application
  - **C**reate
  - **R**ead
  - **U**pdate
  - **D**elete

# Update

<a href="#">Schema</a>	<b>Mutation</b>	<a href="#">Mutation</a>	<b>updateBook</b>	<a href="#">updateBook</a>	<b>UpdateBookInput</b>
<p><a href="#">Search Mutation...</a></p> <p>No Description</p> <p><b>FIELDS</b></p> <p><code>createBook(input: CreateBookInput!): Book</code></p> <p><code>updateBook(input: UpdateBookInput!): Book</code></p> <p><code>deleteBook(input: DeleteBookInput!): Book</code></p>		<p>No Description</p> <p><b>TYPE</b></p> <p><code>Book</code></p> <p><b>ARGUMENTS</b></p> <p><code>input: UpdateBookInput!</code></p>		<p><a href="#">Search UpdateBookInput...</a></p> <p>No Description</p> <p><b>FIELDS</b></p> <p><code>id: ID!</code></p> <p><code>bookId: ID</code></p> <p><code>name: String</code></p> <p><code>category: String</code></p> <p><code>description: String</code></p> <p><code>price: Float</code></p>	

# Delete

◀ Mutation

**deleteBook**

No Description

**TYPE**

Book

**ARGUMENTS**

input: DeleteBookInput!

◀ deleteBook

**DeleteBookInput**

🔍 Search DeleteBookInput...

No Description

**FIELDS**

id: ID

# More about API

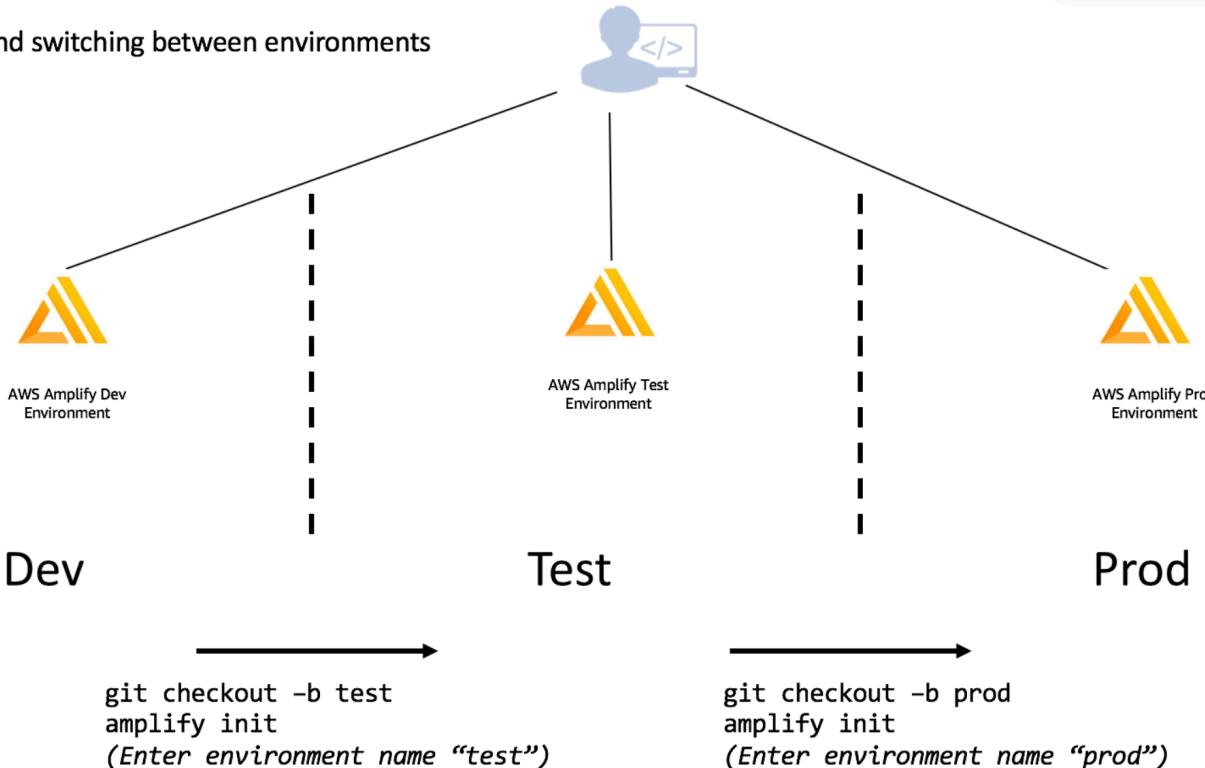
- Interact with REST API
- Add a Search API
  - ElasticSearch
  - @searchable

# Workshop – Section 5

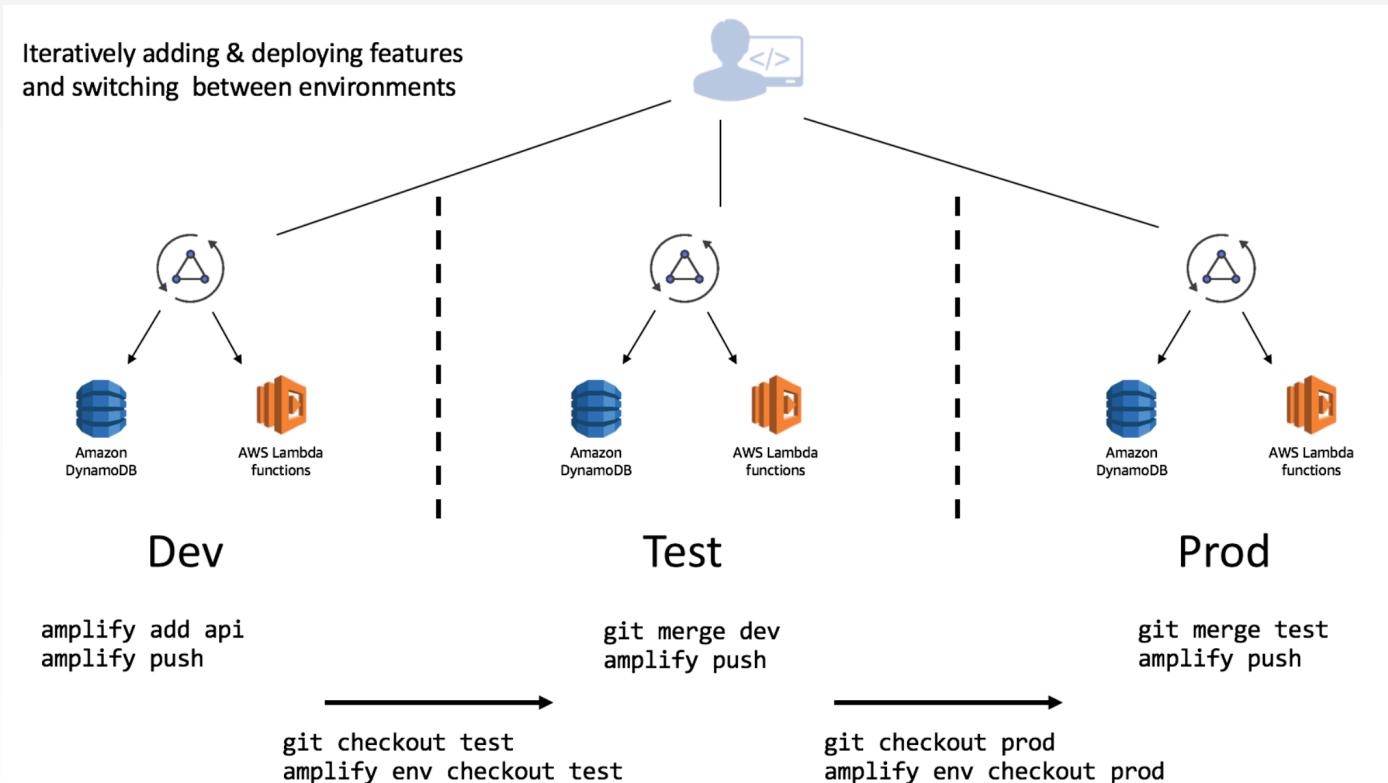
- Multiple development environments

# Switch between Environments

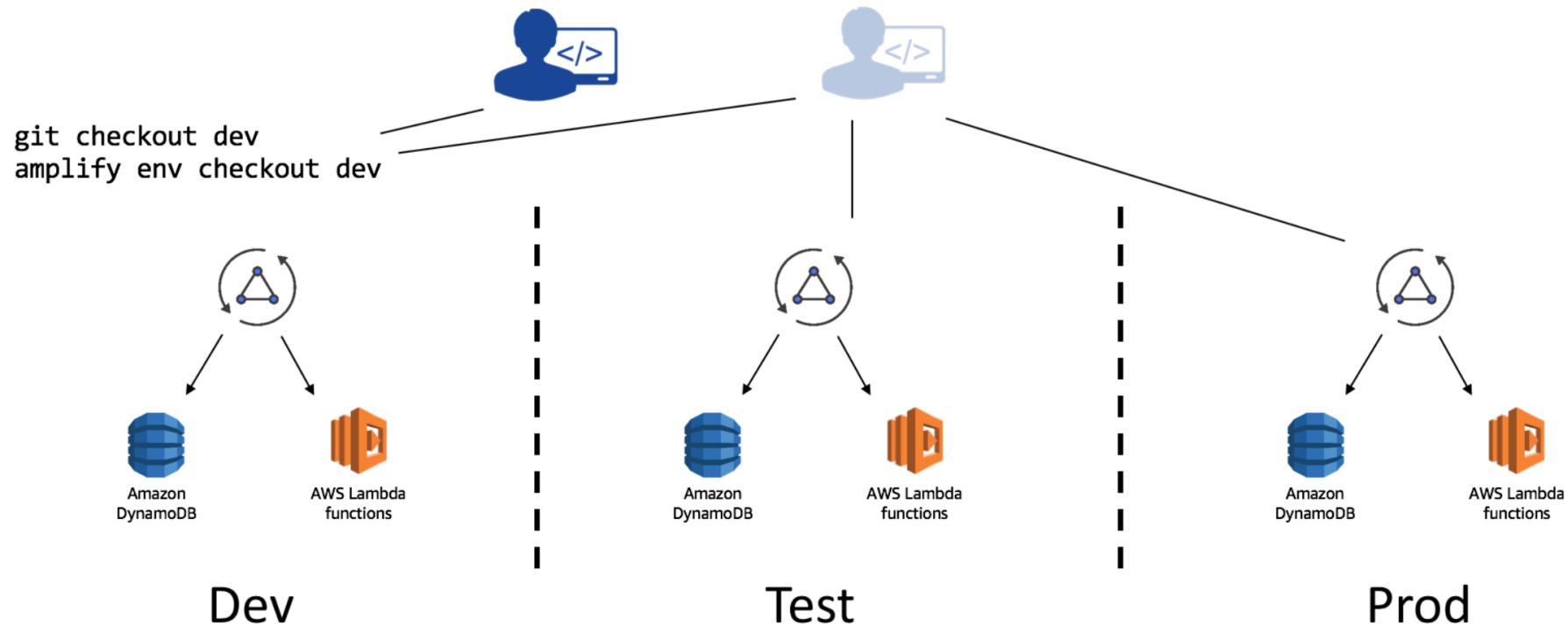
Creating and switching between environments



# Function add / update / test / merge



Multiple developers sharing environments in the same account



# Amplify env commands

Command	Description
amplify env add	Adds a new environment to your Amplify Project
amplify env list	Displays a list of all the environments in your Amplify project
amplify env remove	Removes an environment from the Amplify project
amplify env get –name	Displays the details of the environment specified in the command
amplify env pull	Pulls your environment with the current cloud environment. Use the restore flag to overwrite your local backend configs with that in the cloud
amplify env checkout	Moves your environment to the environment specified in the command. Use the restore flag to overwrite your local backend configs with the backend configs of the environment specified.

# Recap

- Create a React application
- Amplify setup
- Access control and data CRUD
- Application deployment
- More references:
  - <https://aws-amplify.github.io/docs/js/start>
  - <https://amplify.aws/community/>

# Q & A

# Thank You!



[ylchen@vt.edu](mailto:ylchen@vt.edu)



[www.linkedin.com/in/yinlin-chen](https://www.linkedin.com/in/yinlin-chen)