NET307R

Customizing content delivery with Lambda@Edge

Tal Shalom

Sr. Solutions Architect, Edge
AWS

Samrat Karak

Sr. Product Manager, CloudFront
AWS

Lee Atkinson
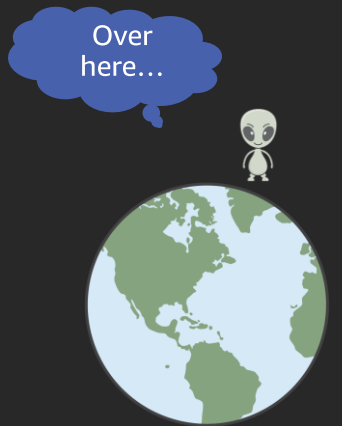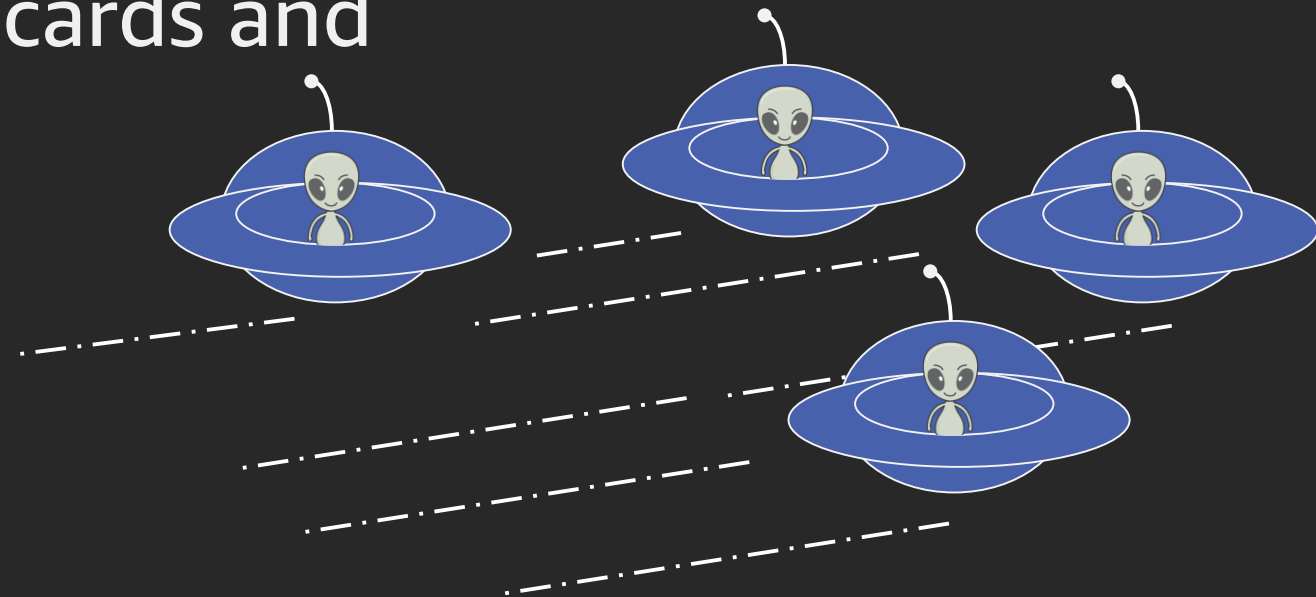
Principal Solutions Architect
AWS

AWS re:Invent

aws

Agenda

- Workshop Overview
- CloudFront and Lambda@Edge
- Event Engine
- Workshop Labs
- Wrap-up

Workshop Overview

AWS re: Invent

aws

Aliens have arrived

- Problem
  - Communication limitation
  - Use images, video, and short text
- Solution
  - Build educational website
  - Simple digital learning cards and

Over here…

Use the Following AWS Services

- **Amazon S3**
  - Website's static files

- **Amazon DynamoDB**
  - Dynamic Data

- **Amazon CloudFront**
  - Fast and secure content delivery

- **AWS Lambda@Edge**
  - Customize content on-the-fly
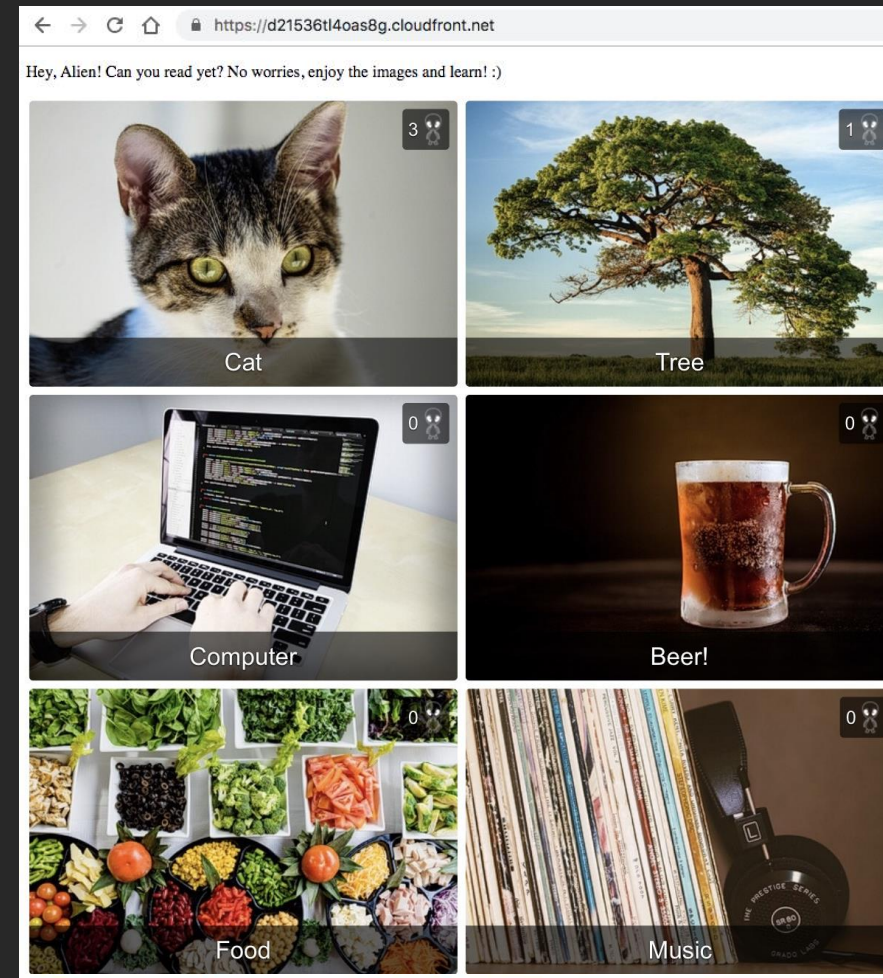
Amazon CloudFront Edge Locations

210 Edge Locations
(199 PoPs +11REC)
78 Cities, 37 Countries

North America (73)
Ashburn, VA (6)
Atlanta, GA (5)
Boston, MA (3)
Chicago, IL (7)
Dallas, TX (6)
Denver, CO(2)
Hayward, CA
Hillsboro, OR (2)
Houston, TX (4)
Jacksonville, FL
Los Angeles, CA (4)
Miami, FL (3)
Minneapolis, MN
Montreal, QC
New York, NY (3)
Newark, NJ (5)

Palo Alto, CA
Philadelphia, PA
Phoenix, AZ (2)
Salt Lake City, Utah
San Jose, CA(2)
Seattle, WA (4)
South Bend, IN
Toronto, ON (2)

REC: Virginia, Ohio, Oregon

Europe (50)
Amsterdam, The Netherlands (2)
Berlin, Germany (2)
Copenhagen, Denmark
Dublin, Ireland
Frankfurt, Germany (10)
Helsinki, Finland
London, England (9)
Madrid, Spain (2)
Manchester, England (2)
Marseille, France
Milan, Italy
Munich, Germany (2)
Palermo, Italy

Lisbon, Portugal
Oslo, Norway
Paris, France (5)
Prague, Czech Republic
Stockholm, Sweden (3)
Vienna, Austria
Warsaw, Poland
Zurich, Switzerland (2)
Brussels, Belgium

REC: Frankfurt, London

China(4)
Beijing;
Shenzhen
Shanghai;
Zhongwei

Middle East (4)
Dubai, UAE
Fujairah, UAE
Manama, Bahrain
Tel Aviv, Israel

Asia(48)
Bangalore, India (3)
Chennai, India (2)
Hong Kong, China (3)
Hyderabad, India (4)
Kuala Lumpur, Malaysia
Mumbai, India (2)
Manila, Philippines
New Delhi, India (5)
Osaka, Japan
Seoul, Korea (4)
Singapore (3)
Taipei, Taiwan (3)
Tokyo, Japan (16)
REC: Mumbai, Singapore,
Seoul, Tokyo

South America (6)
Sao Paulo, Brazil(2)
Rio de Janeiro, Brazil(3)
Santiago, Chile
Bogota, Colombia
Buenos Aires, Argentina

REC: Sao Paulo

Africa (2)
Johannesburg, South Africa
Cape Town, South Africa

Australia(4)
Melbourne
Perth
Sydney (2)

REC: Sydney

Edge
Locations

Multiple
Edge Locations

Regional
Edge Caches

https://aws.amazon.com/cloudfront/features/
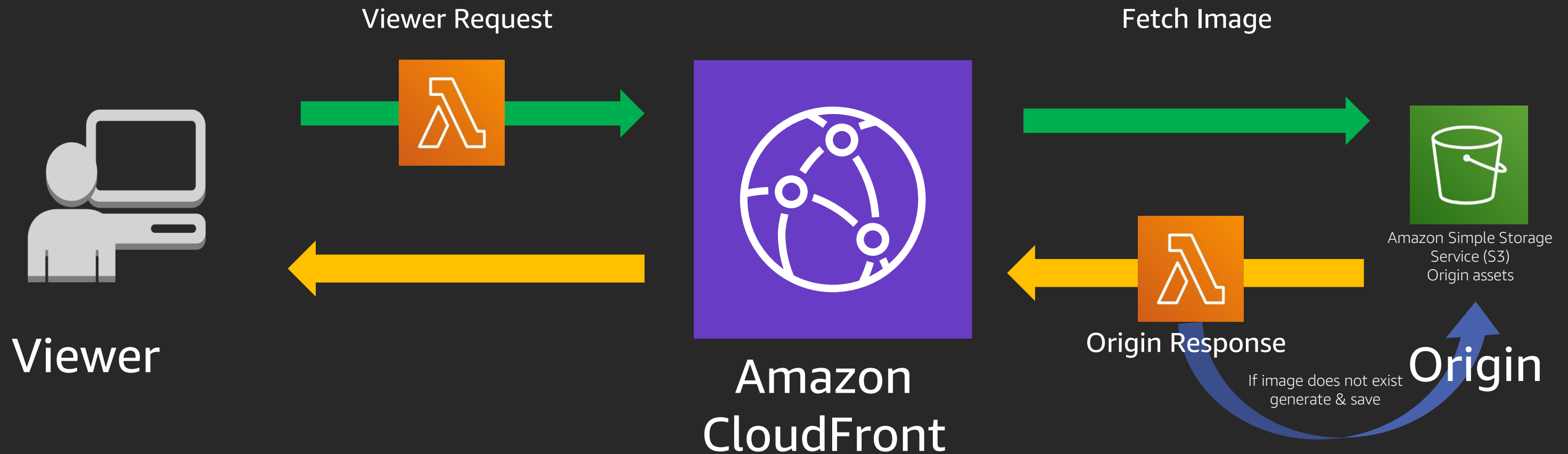
CloudFront triggers for Amazon Lambda@Edge
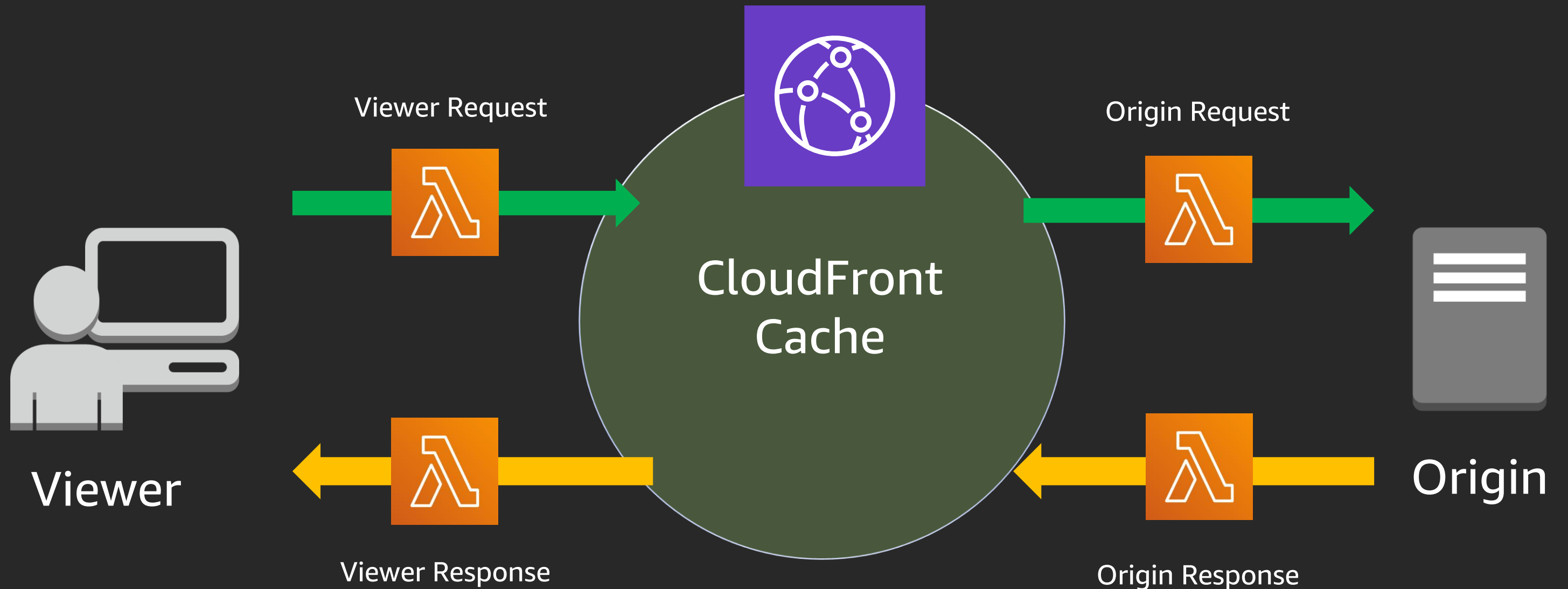
Lambda@Edge for flexibility in operations

- Adding headers (Cache-Control, HSTS…)
- A/B testing
- URL rewriting to origin
- Response manipulation from Origin
- 'Originless' logic like redirection

Example – Image Generation Workflow

Viewer Request

Fetch Image

Viewer

Amazon
CloudFront

Amazon Simple Storage
Service (S3)
Origin assets

Origin Response

If image does not exist
generate & save

Origin

Event Engine

https://dashboard.eventengine.run/

Who are you?

1. By using Event Engine for the relevant event, you agree to the <u>AWS Event Terms and Conditions</u> and the <u>AWS Acceptable Use Policy</u>. You acknowledge and agree that are using an AWS-owned account that you can only access for the duration of the relevant event. If you find residual resources or materials in the AWS-owned account, you will make us aware and cease use of the account. AWS reserves the right to terminate the account and delete the contents at any time.

2. You will not: (a) process or run any operation on any data other than test data sets or lab-approved materials by AWS, and (b) copy, import, export or otherwise create derivate works of materials provided by AWS, including but not limited to, data sets.

3. AWS is under no obligation to enable the transmission of your materials through [AWS Event Engine] and may, in its discretion, edit, block, refuse to post, or remove your materials at any time.

4. Your use of the [event engine] will comply with these terms and all applicable laws, and your access to [AWS Event Engine] will immediately and automatically terminate if you do not comply with any of these terms or conditions.

Team Hash (e.g. abcdef123456)

This is the 12 digit hash that was given to you or your team.

✓ Invalid Hash

Who are you?

1. By using Event Engine for the relevant event, you agree to the <u>AWS Event Terms and Conditions</u> and the <u>AWS Acceptable Use Policy</u>. You acknowledge and agree that are using an AWS-owned account that you can only access for the duration of the relevant event. If you find residual resources or materials in the AWS-owned account, you will make us aware and cease use of the account. AWS reserves the right to terminate the account and delete the contents at any time.
2. You will not: (a) process or run any operation on any data other than test data sets or lab-approved materials by AWS, and (b) copy, import, export or otherwise create derivate works of materials provided by AWS, including but not limited to, data sets.
3. AWS is under no obligation to enable the transmission of your materials through [AWS Event Engine] and may, in its discretion, edit, block, refuse to post, or remove your materials at any time.
4. Your use of the [event engine] will comply with these terms and all applicable laws, and your access to [AWS Event Engine] will immediately and automatically terminate if you do not comply with any of these terms or conditions.

This is the 12 digit hash that was given to you or your team.

✓ Proceed

User Dashboard

☁ Session

Set Team Name | AWS Console

Session: workshop test event

Team Name: **Not Approved**

Modules

ℹ **No Modules Available**

No modules have been enabled for this account yet.

Dashboard

Help   Logout

User Dashboard

☁ Session

Set Team Name   AWS Console

Session: workshop test event
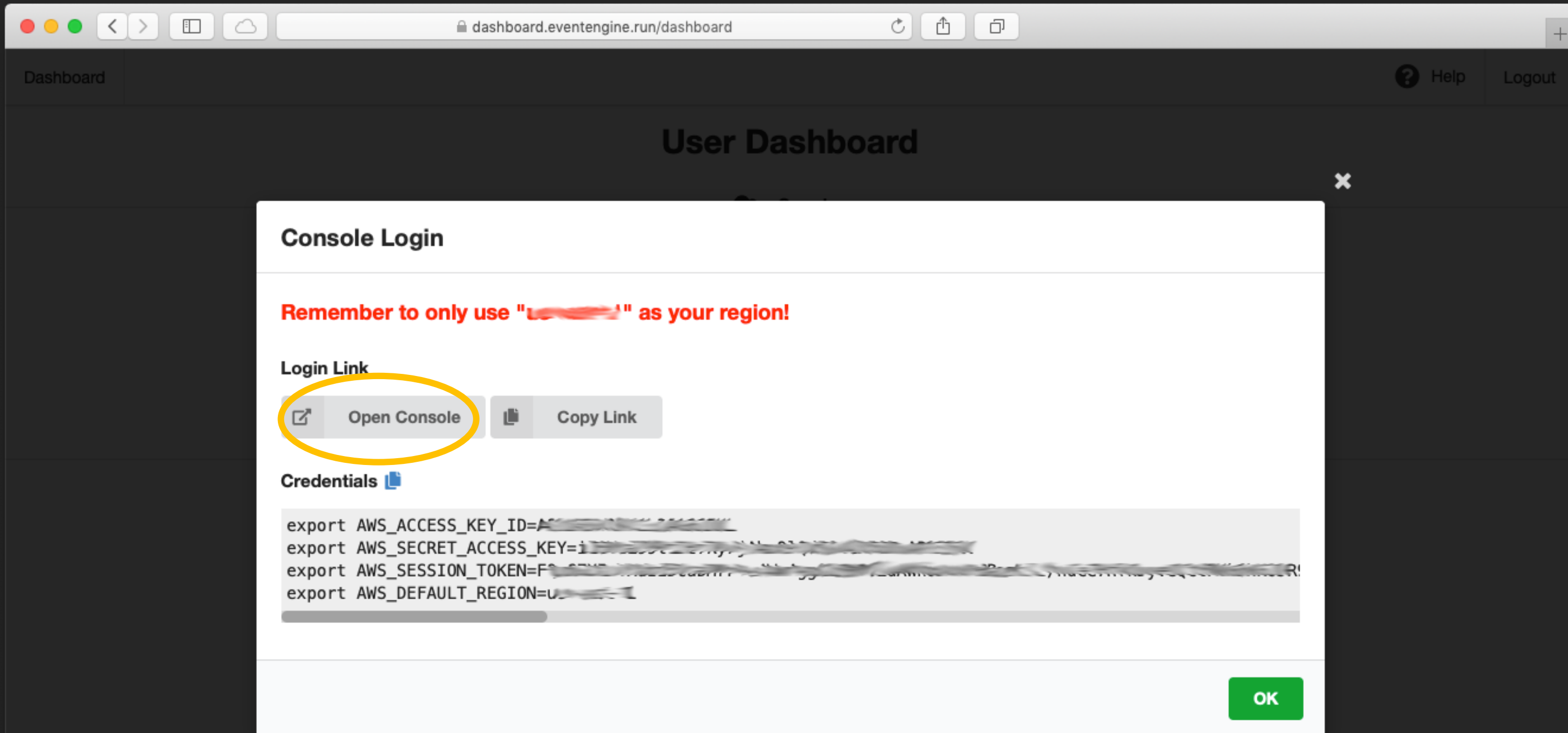Team Name: **Not Approved**

Modules

ℹ **No Modules Available**
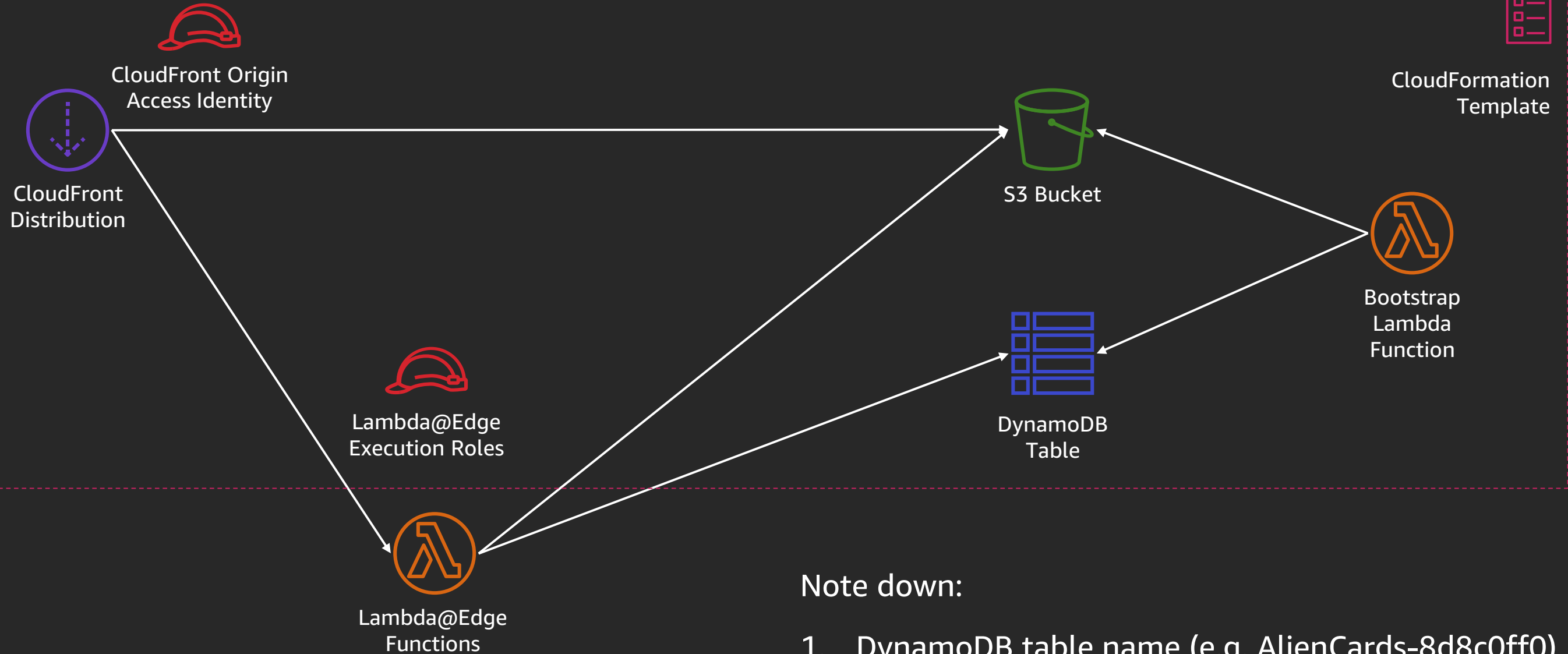
No modules have been enabled for this account yet.

Windows users: use set or Set-Credential instead of export

Labs

AWS re:Invent

aws

Labs

- Lab 0 – Launch the Stack
- Lab 1 – Security
- Lab 2 – Content Generation
- Lab 3 – Simple API
- Lab 4 – Pretty URLs
- Lab 5 – Customization

Lab 0 – Launch the Stack

CloudFront Origin
Access Identity

CloudFront
Distribution

CloudFormation
Template

S3 Bucket

Bootstrap
Lambda
Function

Lambda@Edge
Execution Roles

DynamoDB
Table

Lambda@Edge
Functions

Note down:

1. DynamoDB table name (e.g. AlienCards-8d8c0ff0)

2. CloudFront distribution domain name
   (e.g. d21536tl4oas8g.cloudfront.net)

Lab 1 - Security

- Check site security with Mozilla Observatory

  - https://observatory.mozilla.org/

- Improve site security:

  - Configure CloudFront for HTTP to HTTPS redirect

  - Add standard security headers using Lambda@Edge

    - `Strict-Transport-Security`

    - `Content-Security-Policy`

    - `x-xss-protection`

    - `x-content-type-options`

    - `x-frame-options`

  - Use 'Origin request' event so content is cacheable

Scan Summary

A+	Host:	d21536tl4oas8g.cloudfront.r
	Scan ID #:	9202683 (unlisted)
	Start Time:	November 19, 2018 1:30 AM
	Duration:	3 seconds
	Score:	105/100
	Tests Passed:	11/11

Lab 2 – Content Generation

- Use Lambda@Edge to generate card and home pages:
- Call CloudFront to get 'card' template
- Access 'card' data from DynamoDB
- Use 'Origin Request' event so content is cacheable

Cat

Lab 3 – Simple API

- Implement '+1' like feature

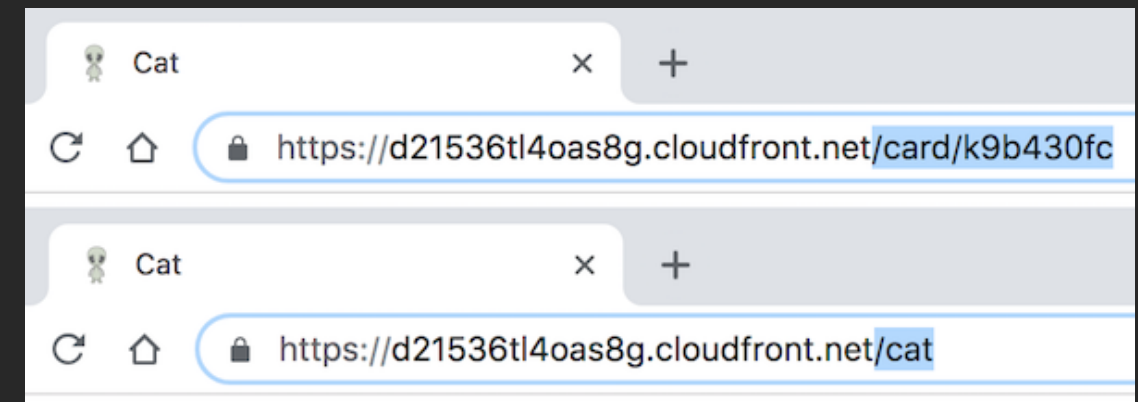- Use Lambda@Edge to accept POSTs

- Update a state in DynamoDB

Lab 4 – Pretty URLs

- Pretty URLs are easy to read and remember

  - Help with search engine ranking

  - Allow your viewers to use the descriptive links in social media

- Lambda@Edge rewrites the request's URL, for example:
  https://d123.cloudfront.net/card/da8398f4
  to
  https://d123.cloudfront.net/tree

Lab 5 – Customization

- Serve different CSS based on viewer device type

  - Different capabilities (e.g. touch vs mouse hover)

- Use CloudFront generated headers:

  - CloudFront-Is-Mobile-Viewer

  - CloudFront-Is-Desktop-Viewer
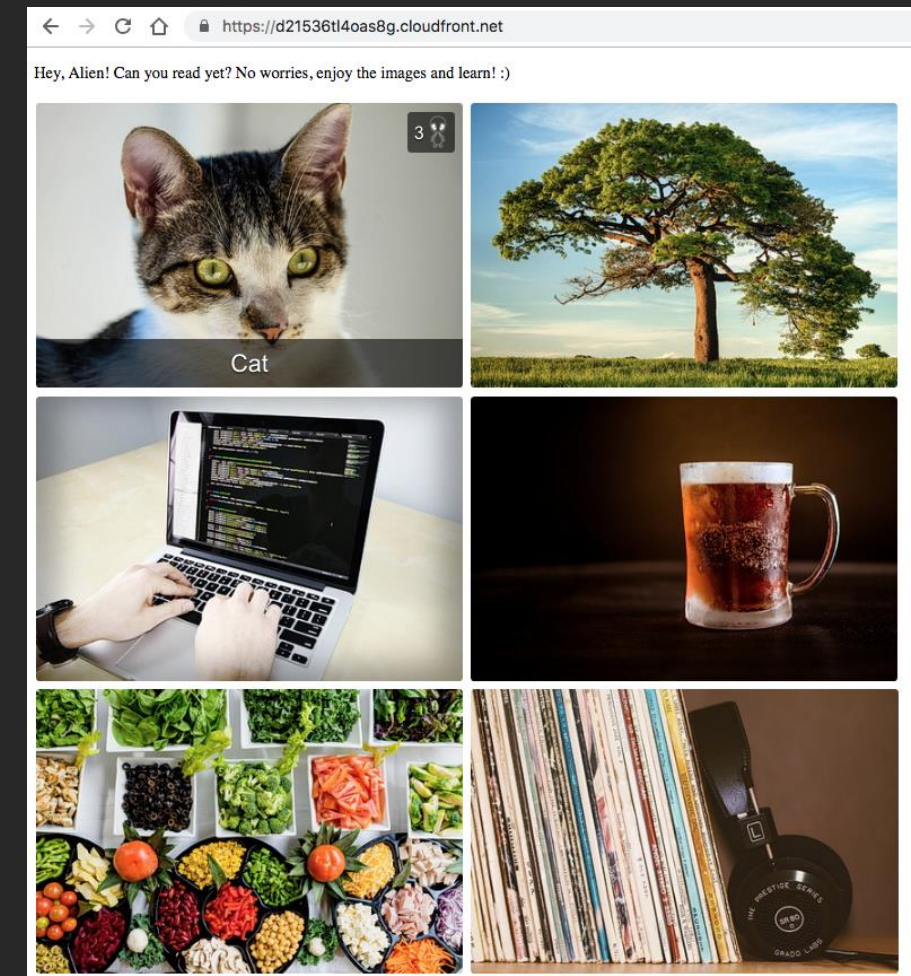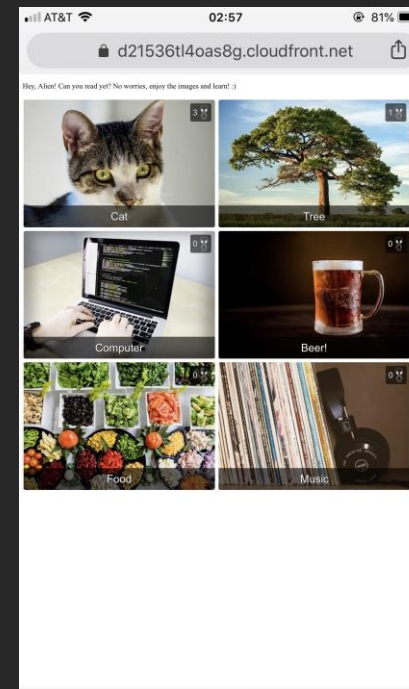
Labs

- Lab 0 – Launch the Stack – 10min
- Lab 1 – Security – 10min
- Lab 2 – Content Generation – 30min
- Lab 3 – Simple API – 20min
- Lab 4 – Pretty URLs – 30min
- Lab 5 – Customization - 20min

Wrap-up

AWS
re: Invent

Lambda@Edge Design Best Practices

- Choose the right trigger

- Performance – Memory size derive CPU allocation (can affect execution time)

- Be mindful of function scaling

- Optimize for cost

- Consider limits

- Additional resources:

    - Resizing Images with Amazon CloudFront & Lambda@Edge (https://aws.amazon.com/blogs/networking-and-content-delivery/resizing-images-with-amazon-cloudfront-lambdaedge-aws-cdn-blog/)

    - Authorization@Edge – How to Use Lambda@Edge and JSON Web Tokens to Enhance Web Application Security (https://aws.amazon.com/blogs/networking-and-content-delivery/authorizationedge-how-to-use-lambdaedge-and-json-web-tokens-to-enhance-web-application-security/)

    - Further reading and blogs (https://aws.amazon.com/blogs/networking-and-content-delivery/category/networking-content-delivery/lambdaedge/)

Please complete the session survey in the mobile app.

Thank you!

Tal Shalom
Sr. Solutions Architect, Edge
AWS

Samrat Karak
Sr. Product Manager, CloudFront
AWS

Lee Atkinson
Principal Solutions Architect
AWS

AWS re:Invent

aws