Innovator Island – Day 2

Build a serverless web app for a theme park

James Beswick
Sr. Developer Advocate, AWS Serverless

© 2020, Amazon Web Services, Inc. or its Affiliates.

About me

- James Beswick
  - Email: jbeswick@amazon.com
  - Twitter: @jbesw
- Senior Developer Advocate – AWS Serverless
- Self-confessed serverless geek
- Software Developer
- Product Manager
- Previously:
  - Multiple start-up tech guy
  - Rackspace, USAA, Morgan Stanley, J P Morgan
  - Enjoys travel, coffee, comedy and theme parks…

Day 2 agenda

Realtime messaging

The tools and services

Coding and discussion

Full 5-day agenda at
http://bit.ly/islandworkshop

© 2020, Amazon Web Services, Inc. or its Affiliates.

Prerequisites

- AWS account – personal account with full admin access
- Cost:
  - Attending workshop is free BUT
  - Running the workshop is might incur small charges
- Workshop cannot be completed with a tablet
- Compatible with Chrome/Firefox (not Safari)
- Familiarity with the AWS Management Console
- You need everything you built from Day 1

Code repo and instructions at
http://bit.ly/innovatorisland

© 2020, Amazon Web Services, Inc. or its Affiliates.

This week's adventure

Day 1: Set up the app

- Deploy the frontend and backend applications

Day 2: Ride wait times

- Integrate your app with the ride systems so guests can see wait times

Day 3: Ride photos

- Build a photo-processing flow so guests can take selfies around the park

Day 4: Analyzing visitor stats

- Gain insights from all the wristband data collected from visitors

Day 5: Processing events

- Use an event bus to handle maintenance and ride malfunctions to improve operations

Code repo and instructions at
http://bit.ly/innovatorisland

© 2020, Amazon Web Services, Inc. or its Affiliates.

aws

Day 2: Ride wait times

Code repo and instructions at
http://bit.ly/innovatorisland

© 2020, Amazon Web Services, Inc. or its Affiliates.

Go to
http://bit.ly/innovatorisland

© 2020, Amazon Web Services, Inc. or its Affiliates.

Ways to interact between front-end and back-end

1. API Gateway – synchronous request and response.
2. API Gateway – WebSockets.
3. API Gateway – asynchronous request with polling.
4. API Gateway – asynchronous request with webhooks.
5. IoT Core – realtime pub/sub over MQTT.

© 2020, Amazon Web Services, Inc. or its Affiliates.

Publish/Subscribe

© 2020, Amazon Web Services, Inc. or its Affiliates.

Publish/Subscribe with AWS IoT Core

```
await iotdata.publish({
  topic: "theme-park-rides",
  qos: 0,
  payload: message
}).promise()
```

IoT topics

```
 // A message has arrived
mqttClient.on('message', (topic,
payload) => {
  if (topic === "theme-park-rides") {
    // Save/process ride times
  }
})
```

© 2020, Amazon Web Services, Inc. or its Affiliates.

Setting up the IoT Core resources in SAM

```yaml
  ##########################################
  # Resources for realtime messaging       #
  ##########################################

  ThemeParkRealtime:
      Type: "AWS::IoT::Thing"
      Properties:
          ThingName: "theme-park-realtime"
          AttributePayload:
              Attributes: {}

  UserPool:
    Type: "AWS::Cognito::UserPool"
    Properties:
      UserPoolName: ThemeParkUserPool
      MfaConfiguration: "OFF"
      Schema:
        - Name: email
          AttributeDataType: String
          Mutable: false
          Required: true

```

```yaml
  # Creates a User Pool Client to be used by the identity pool
  UserPoolClient:
    Type: "AWS::Cognito::UserPoolClient"
    Properties:
      ClientName: ThemeParkUserPoolClient
      GenerateSecret: false
      UserPoolId: !Ref UserPool

  # Creates a federated Identity pool
  IdentityPool:
    Type: "AWS::Cognito::IdentityPool"
    Properties:
      IdentityPoolName: ThemeParkIdentityPool
      AllowUnauthenticatedIdentities: true
      CognitoIdentityProviders:
        - ClientId: !Ref UserPoolClient
          ProviderName: !GetAtt UserPool.ProviderName
```

© 2020, Amazon Web Services, Inc. or its Affiliates.

Setting up the Cognito roles in SAM

```yaml
# Create a role for unauthorized access to AWS resources.
CognitoUnAuthorizedRole:
  Type: "AWS::IAM::Role"
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
      Statement:
        - Effect: "Allow"
          Principal:
            Federated: "cognito-identity.amazonaws.com"
          Action:
            - "sts:AssumeRoleWithWebIdentity"
          Condition:
            StringEquals:
              "cognito-identity.amazonaws.com:aud": !Ref IdentityPool
            "ForAnyValue:StringLike":
              "cognito-identity.amazonaws.com:amr": unauthenticated
    Policies:
      - PolicyName: "CognitoUnauthorizedPolicy"
        PolicyDocument:
          Version: "2012-10-17"
          Statement:
            - Effect: "Allow"
              Action:
                - "mobileanalytics:PutEvents"
                - "cognito-sync:*"
                - "iot:*"
              Resource: "*"
```

```yaml
# Create a role for authorized acces to AWS resources.
CognitoAuthorizedRole:
  Type: "AWS::IAM::Role"
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
      Statement:
        - Effect: "Allow"
          Principal:
            Federated: "cognito-identity.amazonaws.com"
          Action:
            - "sts:AssumeRoleWithWebIdentity"
          Condition:
            StringEquals:
              "cognito-identity.amazonaws.com:aud": !Ref IdentityPool
            "ForAnyValue:StringLike":
              "cognito-identity.amazonaws.com:amr": authenticated
    Policies:
      - PolicyName: "CognitoAuthorizedPolicy"
        PolicyDocument:
          Version: "2012-10-17"
          Statement:
            - Effect: "Allow"
              Action:
                - "mobileanalytics:PutEvents"
                - "cognito-sync:*"
                - "cognito-identity:*"
              Resource: "*"
```

© 2020, Amazon Web Services, Inc. or its Affiliates.

#InnovatorIsland

© 2020, Amazon Web Services, Inc. or its Affiliates.

Thank you - see you tomorrow!

James Beswick, AWS Serverless
jbeswick@amazon.com
@jbesw

© 2020, Amazon Web Services, Inc. or its Affiliates.