

TUBES SOA
GO-BOX

SIDE 40 GAB



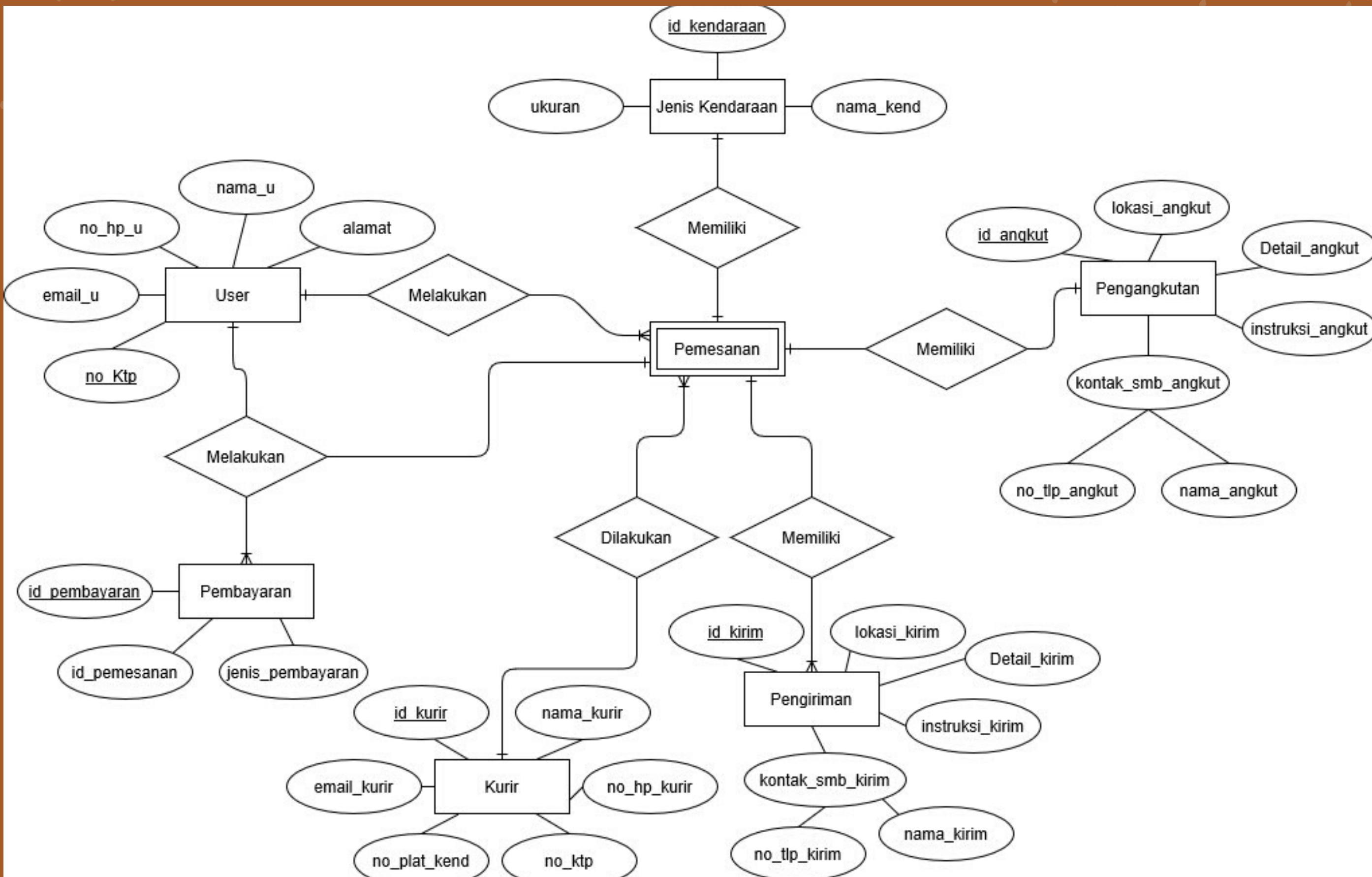
Ainun Abidin
1301164159

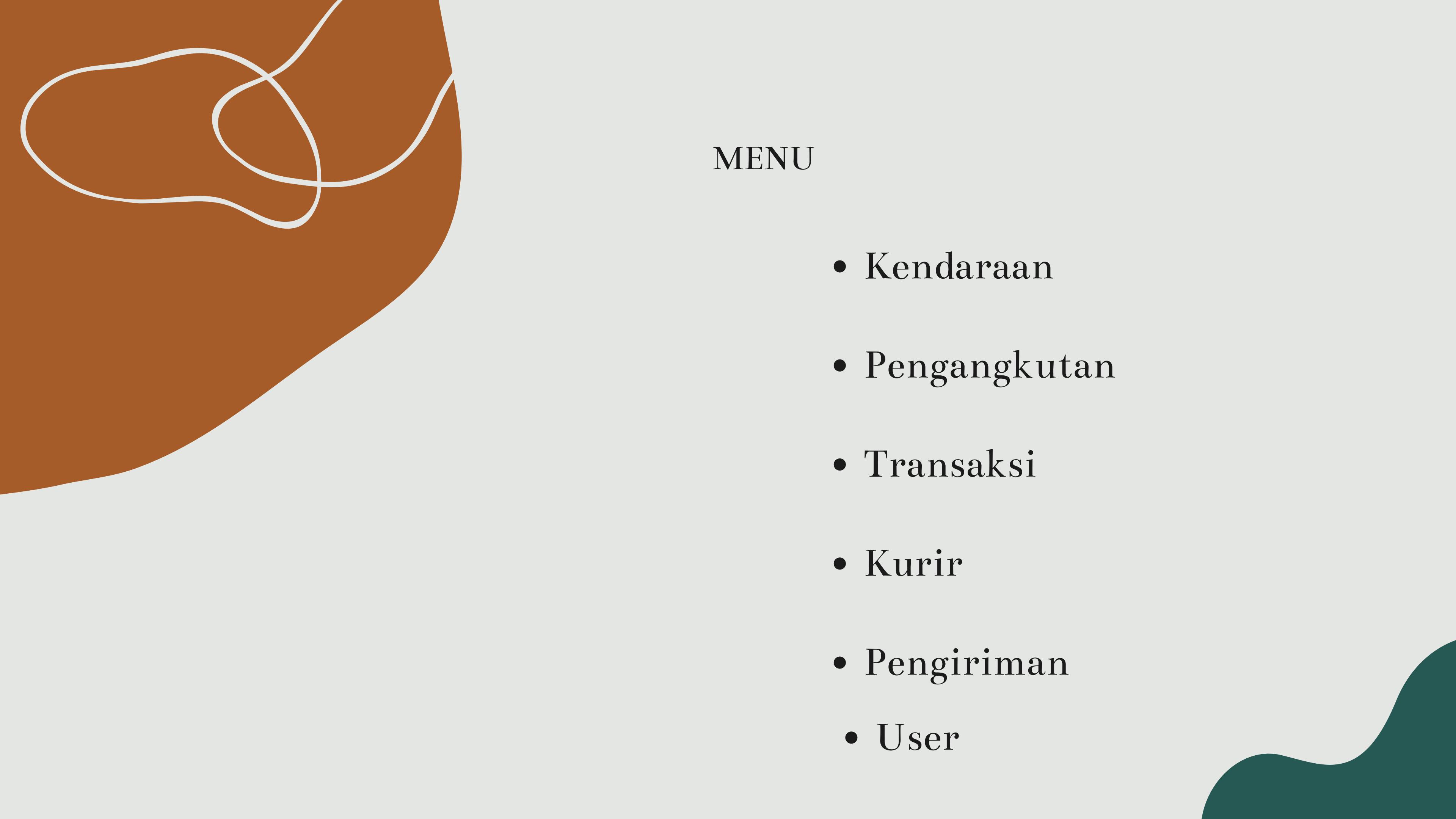
Febry Triyadi
1301162195

Nisrina
Nurhuda
1301164268

Rizki Nastiti
1301164231

ERD GO-BOX





MENU

- Kendaraan
- Pengangkutan
- Transaksi
- Kurir
- Pengiriman
- User

DynamoDB

AWS Services Resource Groups Febry Triyadi Oregon Support

DynamoDB Dashboard Tables Backups Reserved capacity Preferences

DAX Dashboard Clusters Subnet groups Parameter groups Events

Create table Delete table Filter by table name Choose a table group Actions Viewing 7 of 7 Tables

Name	Status	Partition key	Sort key	Indexes	Total read capacity
Kendaraan	Active	idkendaraan (String)	-	0	5
Kurir	Active	idkurir (String)	-	0	5
Pengangkutan	Active	idangkut (String)	-	0	5
Pengiriman	Active	idkirim (String)	-	0	5
Transaksi	Active	idtransaksi (String)	-	0	5
User	Active	iduser (String)	-	0	5
Users	Active	noktp (String)	-	0	5

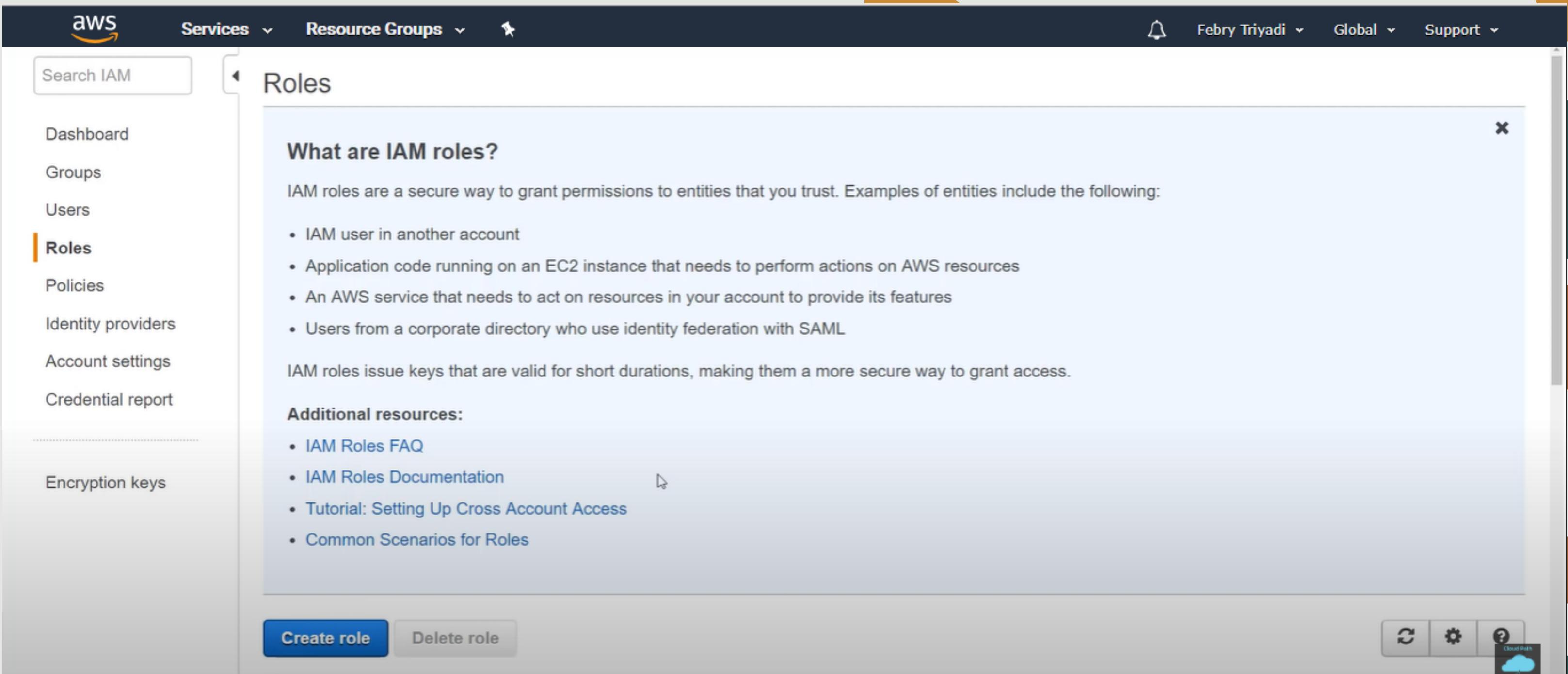
Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Item Table

The screenshot shows the AWS DynamoDB console interface. On the left, the navigation menu is visible with options like 'Create table', 'Delete table', 'Filter by table name', 'Choose a table', and 'Actions'. The 'Tables' section is selected, showing a list of tables including 'Kendaraan', 'Kurir', 'Pengangkutan', 'Pengiriman', 'Transaksi', 'User' (which is selected), and 'Users'. The main right panel displays the 'User' table's 'Items' tab. The table has columns: iduser, alamat, email, lokasi_sekarang, nama, and nohp. Two items are listed:

	iduser	alamat	email	lokasi_sekarang	nama	nohp
	01	sukabirus	silvi@gmail.com	bandung	silvi	085221991211
	1	jl.delima	triyadi@gmail.com	cirebon	Febry	0813213213

AWS IAM - Create Role



The screenshot shows the AWS IAM Roles page. The left sidebar has a 'Roles' section selected, indicated by an orange border. At the bottom of the sidebar are buttons for 'Create role' and 'Delete role'. The main content area contains information about IAM roles, a list of entities they can grant permissions to, and additional resources.

What are IAM roles?

IAM roles are a secure way to grant permissions to entities that you trust. Examples of entities include the following:

- IAM user in another account
- Application code running on an EC2 instance that needs to perform actions on AWS resources
- An AWS service that needs to act on resources in your account to provide its features
- Users from a corporate directory who use identity federation with SAML

IAM roles issue keys that are valid for short durations, making them a more secure way to grant access.

Additional resources:

- [IAM Roles FAQ](#)
- [IAM Roles Documentation](#)
- [Tutorial: Setting Up Cross Account Access](#)
- [Common Scenarios for Roles](#)



Services

Resource Groups



Febry Triyadi

Global

Support

Create role

1

2

3

4

Select type of trusted entity

**AWS service**

EC2, Lambda and others

**Another AWS account**

Belonging to you or 3rd party

**Web identity**

Cognito or any OpenID provider

**SAML 2.0 federation**

Your corporate directory

Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose a use case

Common use cases

[EC2](#)

Allows EC2 instances to call AWS services on your behalf.

[Lambda](#)

Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

[API Gateway](#)[CodeDeploy](#)[EMR](#)[KMS](#)[RoboMaker](#)[AWS Backup](#)[CodeGuru](#)[ElastiCache](#)[Kinesis](#)[S3](#)

* Required

[Cancel](#)[Next: Permissions](#)[Feedback](#)

English (US)

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

[Privacy Policy](#)[Terms of Use](#)

aws Services Resource Groups

Create role

1 2 3 4

▼ Attach permissions policies

Choose one or more policies to attach to your new role.

Create policy

Filter policies ▾ AWSLambdaBasicExecutionRole Showing 3 results

	Policy name ▾	Used as
<input checked="" type="checkbox"/>	AWSLambdaBasicExecutionRole	Permissions policy (3)
<input type="checkbox"/>	AWSLambdaBasicExecutionRole-363820b0-dd0c-43e8-8192-75d8853d8af8	Permissions policy (1)
<input type="checkbox"/>	AWSLambdaBasicExecutionRole-71950c2c-2e77-46ad-9aaf-c37209cc5576	Permissions policy (1)

* Required Cancel Previous Next: Tags

The screenshot shows the AWS IAM 'Create role' wizard, step 2: Attach permissions policies. The user is selecting policies to attach to a new Lambda execution role. A search bar filters results by 'AWSLambdaBasicExecutionRole'. One policy is selected: 'AWSLambdaBasicExecutionRole' (Permissions policy (3)). Other listed policies include 'AWSLambdaBasicExecutionRole-363820b0-dd0c-43e8-8192-75d8853d8af8' and 'AWSLambdaBasicExecutionRole-71950c2c-2e77-46ad-9aaf-c37209cc5576'.



Services ▾

Resource Groups ▾



Febry Triyadi ▾

Global ▾

Support ▾

Create role

1

2

3

4

Review

Provide the required information below and review this role before you create it.

Role name*

Tubesoa

Use alphanumeric and '+,-,@-_ characters. Maximum 64 characters.

Role description

Allows Lambda functions to call AWS services on your behalf.

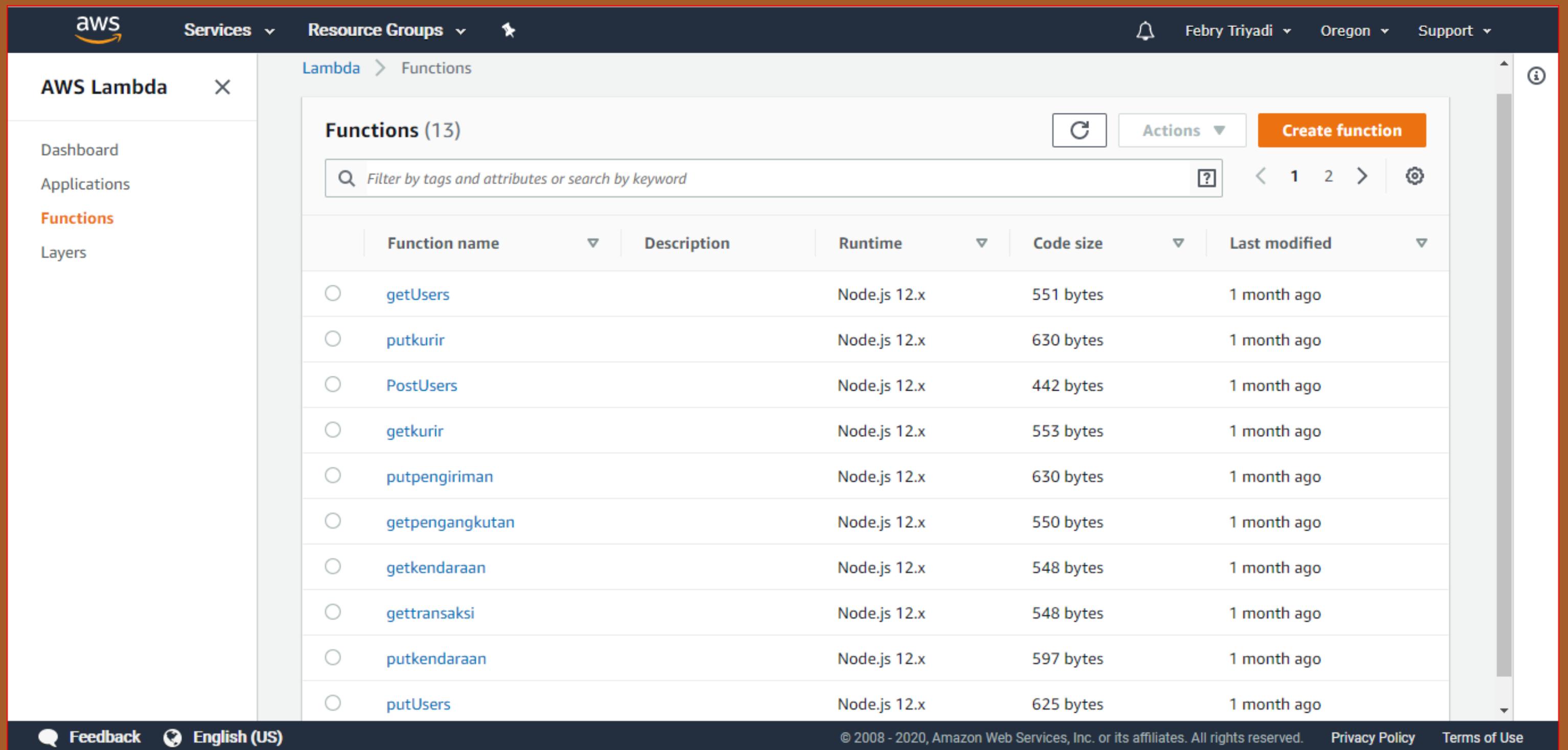
Maximum 1000 characters. Use alphanumeric and '+,-,@-_ characters.

Trusted entities AWS service: lambda.amazonaws.com**Policies**

AWSLambdaBasicExecutionRole

Permissions boundary Permissions boundary is not set*No tags were added.**** Required**[Cancel](#)[Previous](#)[Create role](#)

Lambda Functions



The screenshot shows the AWS Lambda Functions dashboard. On the left, a sidebar menu is open under the 'AWS Lambda' heading, with 'Functions' selected. The main area displays a table titled 'Functions (13)' containing 13 entries. Each entry includes a circular icon, a function name, a description, runtime, code size, and last modified date. All functions listed are in Node.js 12.x with varying code sizes and were last modified '1 month ago'. A 'Create function' button is located at the top right of the table header.

	Function name	Description	Runtime	Code size	Last modified
○	getUsers		Node.js 12.x	551 bytes	1 month ago
○	putkurir		Node.js 12.x	630 bytes	1 month ago
○	PostUsers		Node.js 12.x	442 bytes	1 month ago
○	getkurir		Node.js 12.x	553 bytes	1 month ago
○	putpengiriman		Node.js 12.x	630 bytes	1 month ago
○	getpengangkutan		Node.js 12.x	550 bytes	1 month ago
○	getkendaraan		Node.js 12.x	548 bytes	1 month ago
○	gettransaksi		Node.js 12.x	548 bytes	1 month ago
○	putkendaraan		Node.js 12.x	597 bytes	1 month ago
○	putUsers		Node.js 12.x	625 bytes	1 month ago

AWS Lambda X

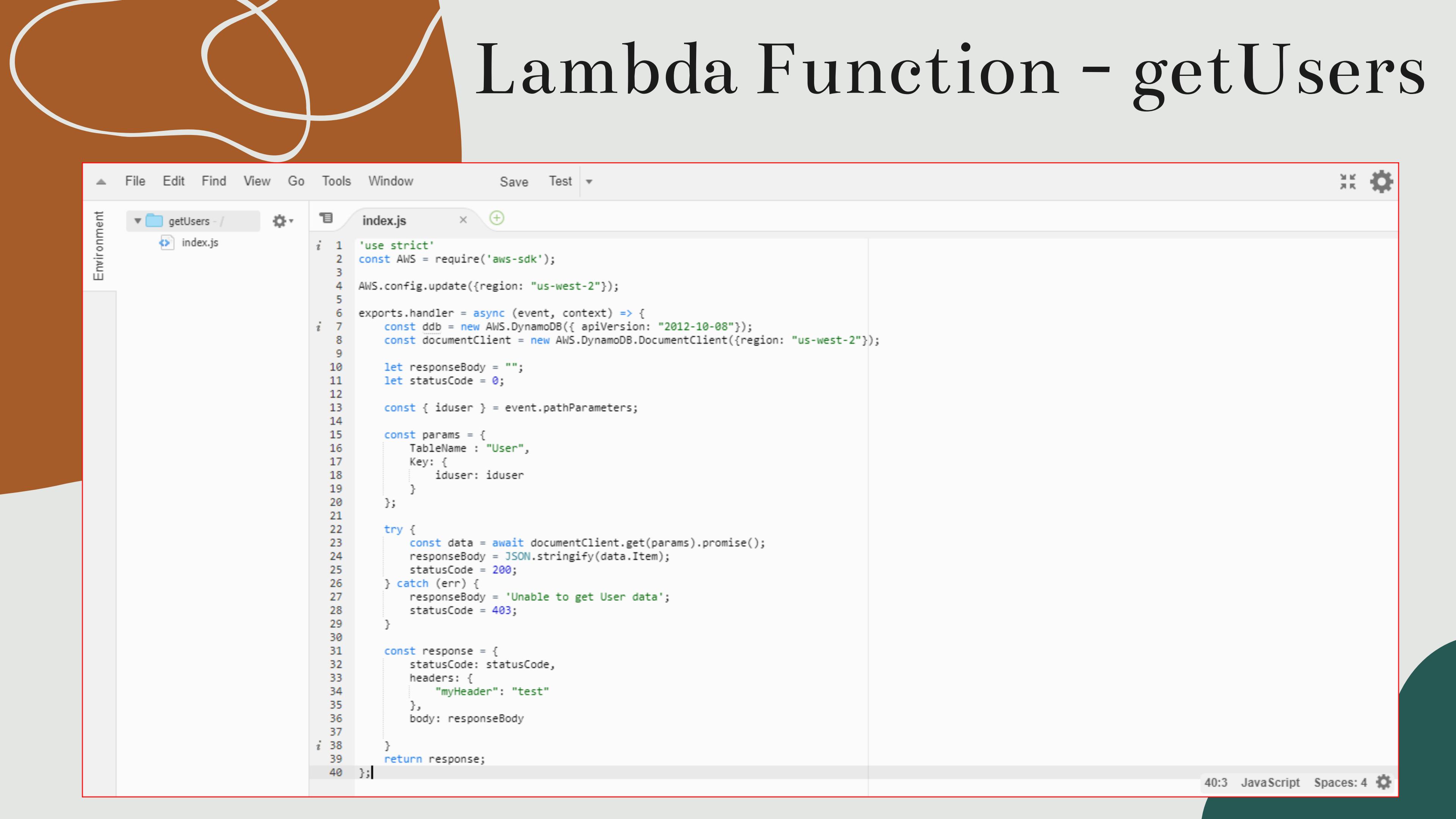
Services ▼ Resource Groups ▼ ★

Febry Triyadi ▼ Oregon ▼ Support ▼

Feedback English (US)

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Lambda Function - getUsers



A screenshot of a code editor showing an AWS Lambda function named "getUsers". The editor has a toolbar at the top with File, Edit, Find, View, Go, Tools, Window, Save, and Test. On the left, there's a sidebar titled "Environment" with a dropdown showing "getUsers - /" and a file list containing "index.js". The main area shows the "index.js" file content:

```
'use strict';
const AWS = require('aws-sdk');
AWS.config.update({region: "us-west-2"});

exports.handler = async (event, context) => {
    const ddb = new AWS.DynamoDB({ apiVersion: "2012-10-08" });
    const documentClient = new AWS.DynamoDB.DocumentClient({region: "us-west-2"});

    let responseBody = "";
    let statusCode = 0;

    const { iduser } = event.pathParameters;

    const params = {
        TableName : "User",
        Key: {
            iduser: iduser
        }
    };

    try {
        const data = await documentClient.get(params).promise();
        responseBody = JSON.stringify(data.Item);
        statusCode = 200;
    } catch (err) {
        responseBody = 'Unable to get User data';
        statusCode = 403;
    }

    const response = {
        statusCode: statusCode,
        headers: {
            "myHeader": "test"
        },
        body: responseBody
    };
    return response;
};
```

The status bar at the bottom right indicates "40:3 JavaScript Spaces: 4".

Lambda Function - PostUsers

The screenshot shows the AWS Lambda function configuration interface for a function named "PostUsers".

Function code (Info)

Code entry type: Edit code inline

Runtime: Node.js 12.x

Handler: index.handler

index.js

```
const ddb = new AWS.DynamoDB({apiVersion: "2012-08-10"});
const documentClient = new AWS.DynamoDB.DocumentClient({region: "us-west-2"});

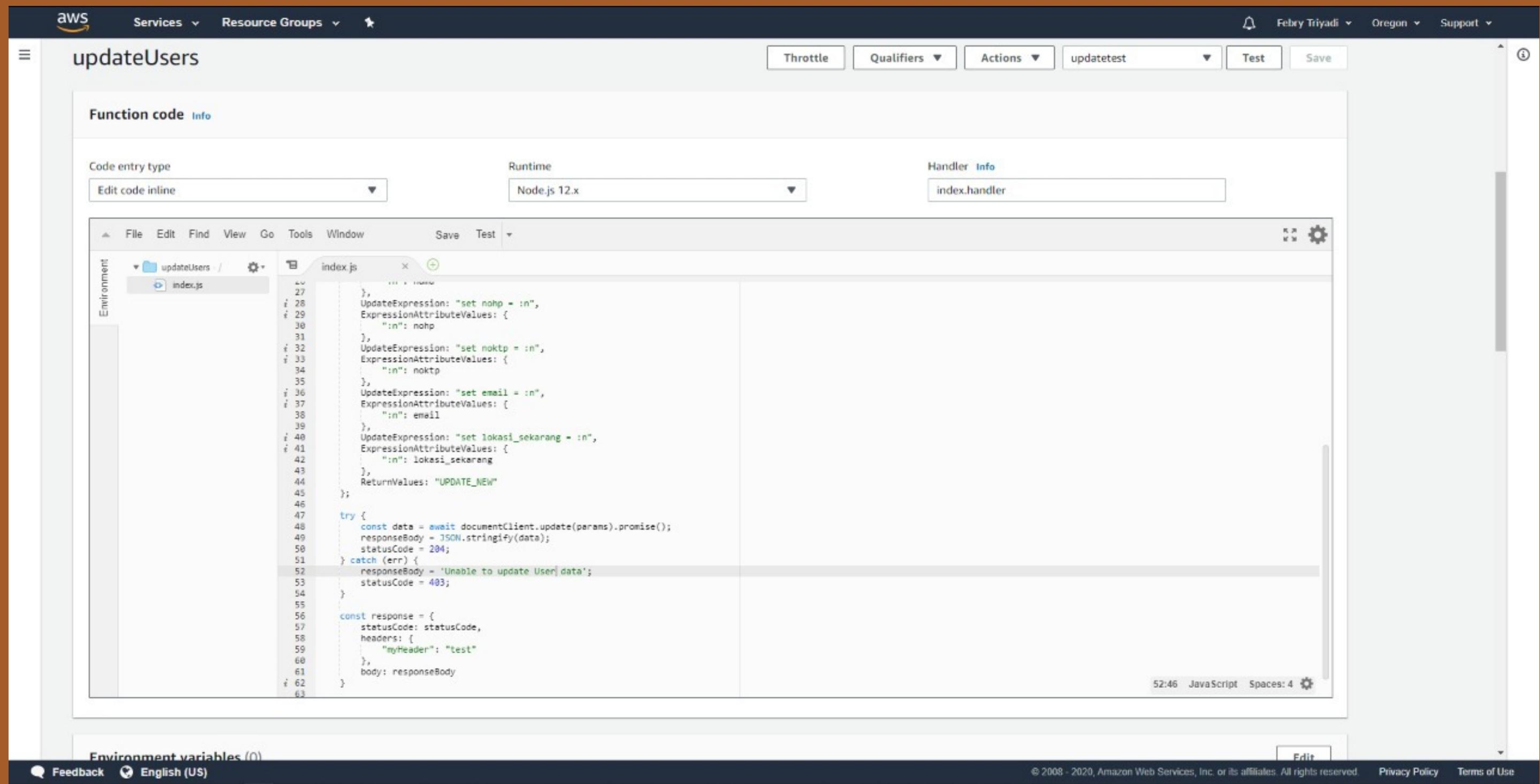
const params = {
    TableName: "Users",
    Item: {
        noktp: "1",
        nama_u: "Triyadi",
        no_hp_u: "08343434",
        email_u: "triyadi@gmail.com",
        alamat: "jl bangsat"
    }
}

try {
    const data = await documentClient.put(params).promise();
    console.log(data);
} catch (err) {
    console.log(err);
}
```

Environment

Feedback English (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Lambda Function - UpdateUsers



The screenshot shows the AWS Lambda function configuration interface for a function named "updateUsers".

Function code (Info)

Code entry type: Edit code inline

Runtime: Node.js 12.x

Handler: index.handler

Environment variables (0)

Feedback English (US)

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

```
index.js
  1  'use strict';
  2
  3  const AWS = require('aws-sdk');
  4  const documentClient = new AWS.DynamoDB.DocumentClient();
  5
  6  exports.handler = async (event) => {
  7      const params = event;
  8
  9      try {
 10          const data = await documentClient.update(params).promise();
 11          responseBody = JSON.stringify(data);
 12          statusCode = 204;
 13      } catch (err) {
 14          responseBody = 'Unable to update User data';
 15          statusCode = 403;
 16      }
 17
 18      const response = {
 19          statusCode,
 20          headers: [
 21              {"myHeader": "test"}
 22          ],
 23          body: responseBody
 24      }
 25
 26      return response;
 27  };
 28
 29  const nohp = ':n';
 30  const ExpressionAttributeValues = {
 31      ':n': nohp
 32  };
 33  const UpdateExpression = "set nohp = :n";
 34  const ExpressionAttributeValues = {
 35      ':n': noktp
 36  };
 37  const UpdateExpression = "set email = :n";
 38  const ExpressionAttributeValues = {
 39      ':n': email
 40  };
 41  const UpdateExpression = "set lokasi_sekarang = :n";
 42  const ExpressionAttributeValues = {
 43      ':n': lokasi_sekarang
 44  };
 45  const ReturnValues = "UPDATE_NEW";
 46
 47  try {
 48      const data = await documentClient.update(params).promise();
 49      responseBody = JSON.stringify(data);
 50      statusCode = 204;
 51  } catch (err) {
 52      responseBody = 'Unable to update User data';
 53      statusCode = 403;
 54  }
 55
 56  const response = {
 57      statusCode,
 58      headers: [
 59          {"myHeader": "test"}
 60      ],
 61      body: responseBody
 62  }
 63
```

API Gateway

The screenshot shows the AWS API Gateway console interface. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, a user icon for Febry Triyadi, and regions Oregon and Support. The main navigation on the left lists APIs, Custom Domain Names, and VPC Links. Below this, under the API: Tubes SOA section, the Resources tab is selected, showing a hierarchical list of resources and their methods:

- / (selected)
 - /kendaraan
 - POST
 - {idkendaraan} (selected)
 - GET
 - /kurir
 - POST
 - {idkurir} (selected)
 - GET
 - /pengangkutan
 - POST
 - {idangkut} (selected)
 - GET
 - /pengiriman
 - POST
 - {idkirim} (selected)
 - GET
 - /transaksi
 - POST
 - {idtransaksi} (selected)
 - GET
 - /user

The Methods section displays the message "No methods defined for the resource." at the bottom.

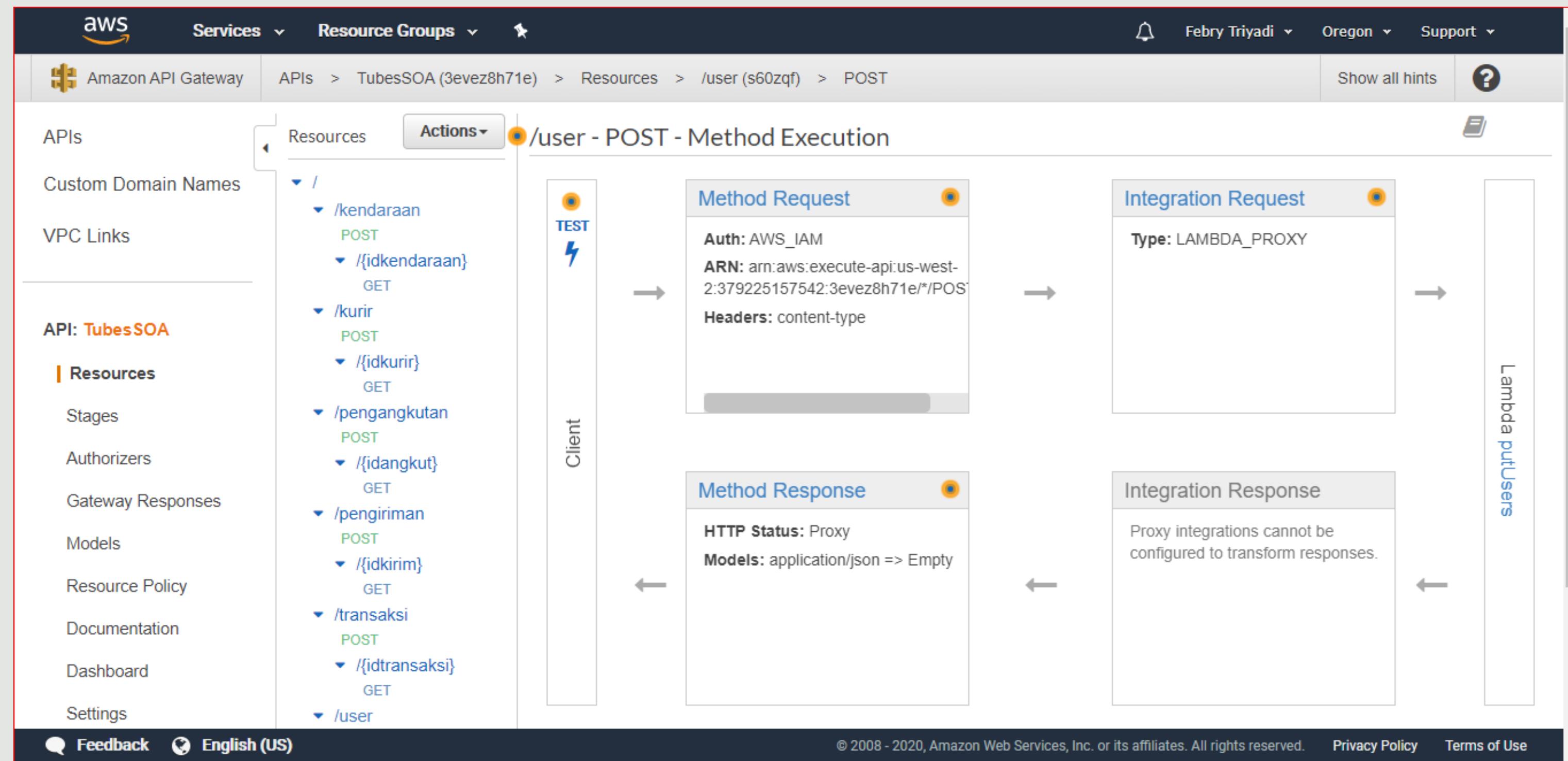
At the bottom of the page, there are links for Feedback, English (US), and footer links: © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved., Privacy Policy, and Terms of Use.

Models

The screenshot shows the AWS API Gateway interface for managing models. The top navigation bar includes the AWS logo, 'Services' dropdown, 'Resource Groups' dropdown, user profile 'Febry Triyadi', location 'Oregon', and support links. The left sidebar lists 'APIs', 'Custom Domain Names', 'VPC Links', and an expanded section for 'API: TubesSOA' containing 'Resources', 'Stages', 'Authorizers', 'Gateway Responses', 'Models' (which is selected), 'Resource Policy', 'Documentation', 'Dashboard', and 'Settings'. The main content area is titled 'Update Model' for the 'user' model. It displays the JSON schema:

```
1  {
2    "$schema": "http://json-schema.org/draft-04/schema#",
3    "title": "UsersInputModel",
4    "type": "object",
5    "properties": {
6      "iduser": { "type": "string" },
7      "alamat": { "type": "string" },
8      "nama": { "type": "string" },
9      "nohp": { "type": "string" },
10     "noktp": { "type": "string" },
11     "email": { "type": "string" },
12     "lokasi_sekarang": { "type": "string" }
13   }
14 }
```

Below the schema, there are fields for 'Model name' (set to 'user'), 'Content type' (set to 'application/json'), and 'Model description' (with a pencil icon). A large button labeled 'Delete Model' is located in the top right corner of the main content area.



Post User

API: TubesSOA

Resources

- Stages
- Authorizers
- Gateway Responses
- Models
- Resource Policy
- Documentation
- Dashboard
- Settings
- Usage Plans
- API Keys
- Client Certificates
- Settings

▼ /{idkendaraan}
 GET

▼ /kurir
 POST
 ▼ /{idkurir}
 GET

▼ /pengangkutan
 POST
 ▼ /{idangkut}
 GET

▼ /pengiriman
 POST
 ▼ /{idkirim}
 GET

▼ /transaksi
 POST
 ▼ /{idtransaksi}
 GET

▼ /user
 POST
 ▼ /{iduser}
 GET

No path parameters exist for this resource. You can define path parameters by using the syntax **{myPathParam}** in a resource path.

Query Strings

{user}
param1=value1¶m2=value2

Headers

{user}
content-type: application/json

Stage Variables

No stage variables exist for this method.

Client Certificate

No client certificates have been generated.

Request Body

```
1 * {  
2   "iduser": "5",  
3   "alamat": "Jl. Harapan nilai besar",  
4   "name": "Asep",  
5   "nohp": "081395939187",  
6   "noktp": "1301625192",  
7   "email": "asep123@gmail.com",  
8   "lokasi_sekarang": "Bandung"  
9 }
```

Test

Feedback English (US)

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Running Post User

The screenshot shows the AWS API Gateway Method Execution interface for a POST request to the '/user' endpoint. The left sidebar lists the API resources, and the main panel displays the method test results.

Request:

```
param1=value1&param2=value2
```

Response Headers:

```
{"myHeader": "test", "X-Amzn-Trace-Id": "Root=1-5eb18283-ecec839f70811c2287ad1712;Sampled=0"}
```

Logs:

```
Execution log for request f1041c4c-98e7-45bb-b2a4-c9fb8fdfc347
Tue May 05 15:13:07 UTC 2020 : Starting execution for request: f1041c4c-98e7-45bb-b2a4-c9fb8fdfc347
Tue May 05 15:13:07 UTC 2020 : HTTP Method: POST, Resource Path: /user
Tue May 05 15:13:07 UTC 2020 : Method request path: {}
Tue May 05 15:13:07 UTC 2020 : Method request query string: {}
Tue May 05 15:13:07 UTC 2020 : Method request headers: {content-type= application/json}
Tue May 05 15:13:07 UTC 2020 : Method request body before transformations: {
    "iduser": "5",
    "alamat": "Jl.Harapan milai besar",
    "nama": "Asep",
    "nohp": "081395939187",
    "noktp": "1301625192",
    "email": "asep123@gmail.com",
    "lokasi_sekarang": "Bandung"
}
Tue May 05 15:13:07 UTC 2020 : Request validation succeeded for content type application/json
Tue May 05 15:13:07 UTC 2020 : Endpoint request URI: https://lambda.us-west-2.amazonaws.com/2015-03-31/functions/arn:aws:lambda:us-west-2:379225157542:function:putUsers/invocations
Tue May 05 15:13:07 UTC 2020 : Endpoint request headers: {x-amzn-lambda-integration-tag=f1041c4c-98e7-45bb-b2a4-c9fb8fdfc347, Authorization=*****}
```

Request Body:

```
1+ {
2   "iduser": "5",
3   "alamat": "Jl.Harapan milai besar",
4   "nama": "Asep",
5   "nohp": "081395939187",
6   "noktp": "1301625192",
7   "email": "asep123@gmail.com",
8   "lokasi_sekarang": "Bandung"
9 }
```

Running Get User

The screenshot shows the AWS API Gateway Method Test interface. The URL is `/user/{iduser}` - GET. The path parameter `{iduser}` is set to `5`. The response status is `200`, latency is `1307 ms`, and the response body is:

```
{ "alamat": "Jl. Harapan nilai besar", "noktp": "1301625192", "nama": "Asep", "nohp": "081395939187", "iduser": "5", "email": "asep123@gmail.com", "lokasi_sekarang": "Bandung" }
```

The response headers include `{"myHeader": "test", "X-Amzn-Trace-Id": "Root-1-5eb182c7-f73a7fcfd3504ee22cc4803a; Sampled-0"}`.

The logs show the execution log for the request:

```
Execution log for request 1a8e9f55-8be3-4093-8ecb-c1256f52513e
Tue May 05 15:14:15 UTC 2020 : Starting execution for request: 1a8e9f55-8be3-4093-8ecb-c1256f52513e
Tue May 05 15:14:15 UTC 2020 : HTTP Method: GET, Resource Path: /user/5
Tue May 05 15:14:15 UTC 2020 : Method request path: {iduser=5}
Tue May 05 15:14:15 UTC 2020 : Method request query string: {}
Tue May 05 15:14:15 UTC 2020 : Method request headers: {content-type= application/json}
Tue May 05 15:14:15 UTC 2020 : Method request body before transformations:
Tue May 05 15:14:15 UTC 2020 : Endpoint request URI: https://lambda.us-west-2.amazonaws.com/2015-03-31/functions/arn:aws:lambda:us-west-2:379225157542:function:getUsers/invocations
Tue May 05 15:14:15 UTC 2020 : Endpoint request headers: {x-amzn-lambda-integration-tag=1a8e9f55-8be3-4093-8ecb-c1256f52513e, Authorization=*****}
*****
```

At the bottom, there are links for Feedback, English (US), Privacy Policy, and Terms of Use.

Running Update User

The screenshot shows the AWS Lambda API Gateway test interface for an 'Update User' operation. The left sidebar lists various stages, authorizers, gateway responses, models, resource policy, documentation, dashboard, settings, usage plans, API keys, client certificates, and settings. The 'Client Certificates' section is currently selected.

The main area displays the following details:

- Stages:** /pengangkutan (POST), /idangkut (GET), /pengiriman (PATCH, POST), /idkirim (GET), /transaksi (POST, GET), /user (POST, GET, PATCH).
- Authorizers:** param1=value1¶m2=value2
- Gateway Responses:** {iduser}
- Models:** content-type: application/json
- Resource Policy:** /{iduser}
- Documentation:** No stage variables exist for this method.
- Dashboard:** No client certificates have been generated.
- Settings:** Client Certificate
- Request Body:**

```
1 - {
2   "iduser": "5",
3   "alamat": "Jl. Delima no.10",
4   "nama": "Febry",
5   "nohp": "0813959123321",
6   "noktp": "1301162195",
7   "email": "triyadifebry@gmail.com",
8   "lokasi_sekarang": "Cirebon"
9 }
```
- Response Headers:** {"myHeader": "test", "X-Amzn-Trace-Id": "Root=1-5eb597a7-603409d35760e94f6fd0999e;Sampled=0"}
- Logs:**

```
Execution log for request b68dfddb-7729-4164-b81f-e5b6d4721ff5
Fri May 08 17:32:23 UTC 2020 : Starting execution for request: b68dfddb-7729-4164-b81f-e5b6d4721ff5
Fri May 08 17:32:23 UTC 2020 : HTTP Method: PATCH, Resource Path: /user/5
Fri May 08 17:32:23 UTC 2020 : Method request path: {iduser=5}
Fri May 08 17:32:23 UTC 2020 : Method request query string: {}
Fri May 08 17:32:23 UTC 2020 : Method request headers: {content-type= application/json}
Fri May 08 17:32:23 UTC 2020 : Method request body before transformations:
{
  "iduser": "5",
  "alamat": "Jl. Delima no.10",
  "nama": "Febry",
  "nohp": "0813959123321",
  "noktp": "1301162195",
  "email": "triyadifebry@gmail.com",
  "lokasi_sekarang": "Cirebon"
}
Fri May 08 17:32:23 UTC 2020 : Request validation succeeded for content type application/json
Fri May 08 17:32:23 UTC 2020 : Endpoint request URI: https://lambda.us-west-2.amazonaws.com/2015-03-31/functions/arn:aws:lambda:us-west-2:379225157542:function:updateUsers/invocations
Fri May 08 17:32:23 UTC 2020 : Endpoint request headers: {x-amzn-lambda-integration-tag=b68dfddb-7729-4164-b81f-e5b6d4721ff5, Authorization=*****}
*****
*****ed0321, X-Amz-Date=20200508T173223Z, x-amzn-apigateway-api-id=3evez8h71e, X-Amz-Source-Arn=arn:aws:execute-api:us-west-2:379225157542:3evez8h71e/test-invoke-stage/PATCH/user/{iduser}, Accept=application/json, User-Agent=AmazonAPIGateway_3evez8h71e, X-Amz-Security-Token=IQoJb3JpZ2luX2VjEKh//////////wEAxCVzLXd1c3QtMijHMEUCIQDjZ2cJhMbVJhnePhpg5tHUo1oEmv5WhdZBWZ+v3adIIgIgLixXY1tj99Wvl852AR1VQ5gb6Ur2xuDuOjWfIs43AgvQMIZv/////////ARACGgwxDkzNTEzMDk0MDciDOWjMe3iyU1p9kXdaCqRA5lwMEckZkF7UwTk9vQvcP6leFLCe9k14wGm81mr3CXVF7er/HJztITkn4unanSotmonBlg3Ubco2rQAOXRxAxRL5n07RWtgDUqgst2UGM56DgaENA40ubAwqIZfqamg [TRUNCATED]
Fri May 08 17:32:23 UTC 2020 : Endpoint request body after transformations: {"resource":"/user/{iduser}", "path":"/user/5", "httpMethod":"PATCH", "headers":{"content-type": "application/json"}, "multiValueHeaders":{"content-type": ["application/json"]}, "queryStringParameters": null, "multiValueQueryStringParameters": null, "pathParameters": {"iduser": "5"}, "stageVariables": null, "requestContext": {"resourceId": "jxxwqr", "resourcePath": "/user/{iduser}", "httpMethod": "PATCH", "extendedRequestId": "ROS1HIpEPHcFeiQ=", "requestTime": "08/May/2020:17:32:23 +0000", "path": "/user/{iduser}", "accountId": "379225157542", "protocol": "HTTP/1.1", "stage": "test-invoke-stage", "domainPrefix": "testPrefix", "requestTimeEpoch": "1588959143003", "requestId": "b68dfddb-7729-4164-b81f-e5b6d4721ff5", "identity": {"cognitoIdentityPoolId": null, "cognitoIdentityId": null, "apiKey": "test-invoke-api-key", "principalOrgId": null, "cognitoAuthenticationType": null, "userArn": "arn:aws:iam::379225157542:root", "apiKeyId": "test-invoke-ap"}
```

Stage Editor

Screenshot of the Amazon API Gateway Stage Editor interface.

The navigation bar shows: AWS, Services, Resource Groups, Feby Triyadi, Oregon, Support.

The breadcrumb path is: APIs > TubesSOA (3evez0h71e) > Stages > TubesSOA.

The left sidebar menu includes:

- APIs
- Custom Domain Names
- VPC Links
- API: TubesSOA**
 - Resources
 - Stages** (selected)
 - Authorizers
 - Gateway Responses
 - Models
 - Resource Policy
 - Documentation
 - Dashboard
 - Settings
- Usage Plans
- API Keys
- Client Certificates
- Settings

The main content area is titled "TubesSOA Stage Editor". It displays the "Invoke URL: https://3evez0h71e.execute-api.us-west-2.amazonaws.com/TubesSOA".

The "Export" tab is selected in the top navigation bar, showing options to export as Swagger (selected), OpenAPI 3, or both.

Three export icons are shown:

- Export as Swagger (represented by a curly brace icon)
- Export as Swagger + API Gateway Extensions (represented by a stylized hash icon)
- Export as Swagger + Postman Extensions (represented by a pen icon)

Below the export icons are JSON and YAML download buttons.

The footer includes: Feedback, English (US), © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved., Privacy Policy, Terms of Use.

Terima
Kasih