AWS DeepLens Workshop: Building a Computer Vision App

Jyothi Nookula - Senior Product Manager, Amazon Web Services

May 23rd 2018

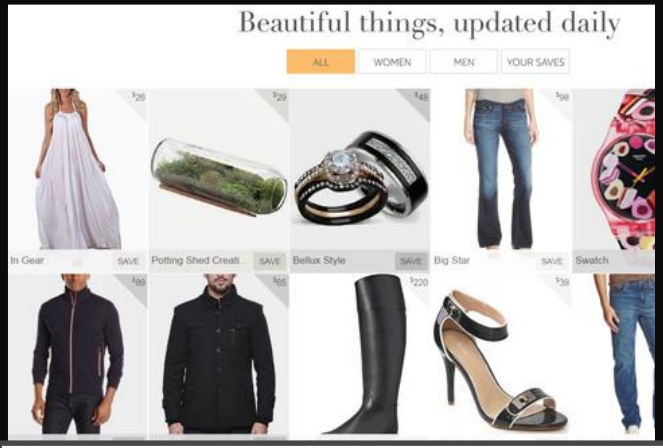
aws

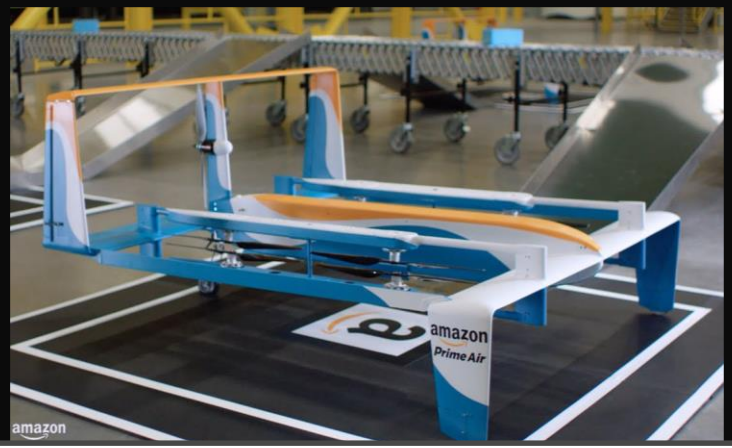
AWS DeepLens
is not a
video camera …

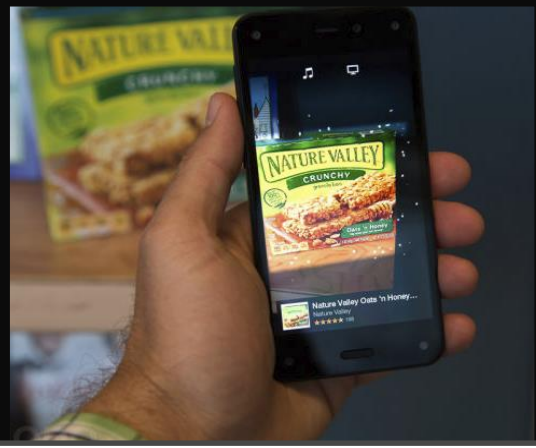
It's the
world's first
deep learning
enabled
developer kit

intel inside™

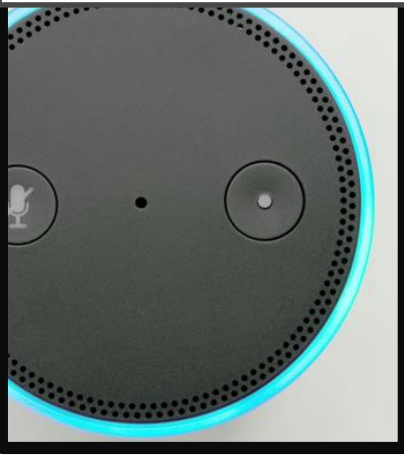
aws







Artificial Intelligence at Amazon



THE AMAZON MACHINE LEARNING STACK

APPLICATION SERVICES

Rekognition | Transcribe | Translate | Polly | Comprehend | Lex

PLATFORM SERVICES

Amazon SageMaker

FRAMEWORKS & INTERFACES

AWS Deep Learning AMIs

Caffe2 | CNTK | Apache MXNet | PyTorch | TensorFlow | Chainer | Keras | Gluon

EDUCATION

AWS DeepLens

aws

Get Started with Sample Projects

Artistic style transfer

Object detection

Face detection / recognition

Activity detection

Hot dog / not hot dog

Cat vs. dog

Add **custom functionality**

Or

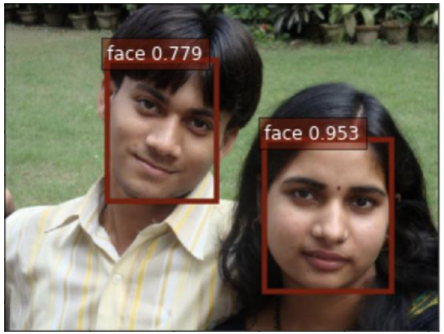
Create your **own project**

aws

Today We Will Cover

1. Machine learning overview

2. Training a model in Amazon SageMaker



3. Deploying a model to AWS DeepLens

4. Extending a project

AWS Lambda

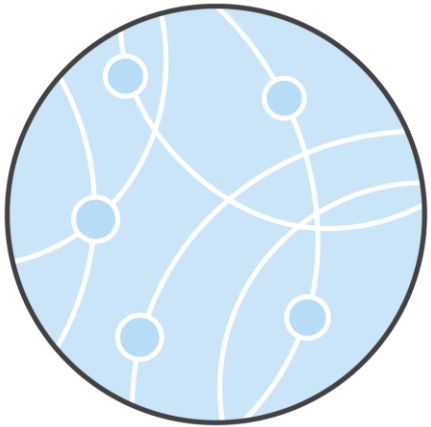
Amazon S3

Amazon Rekognition

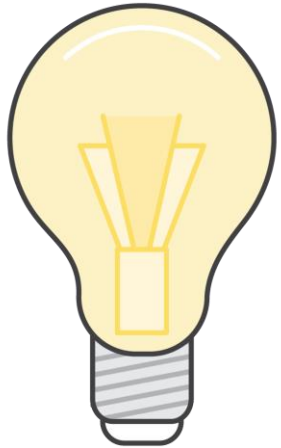
1. Machine Learning Overview

Overview of Deep Learning

Data



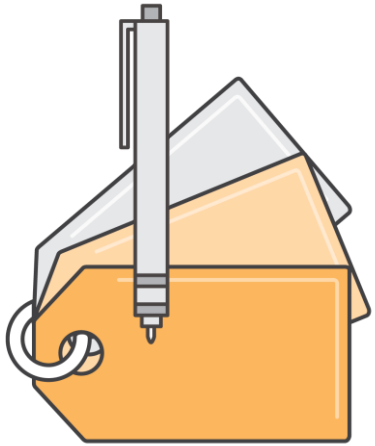
Model training



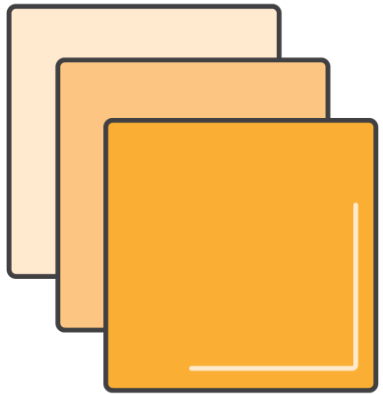
Inference

aws

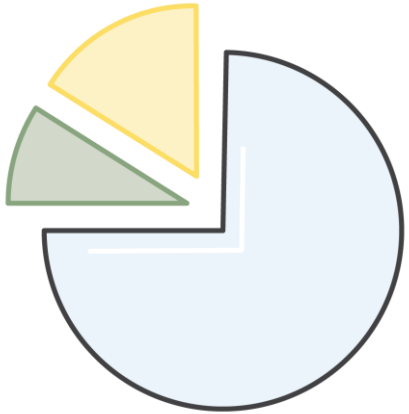
Data



Annotate



Preprocess

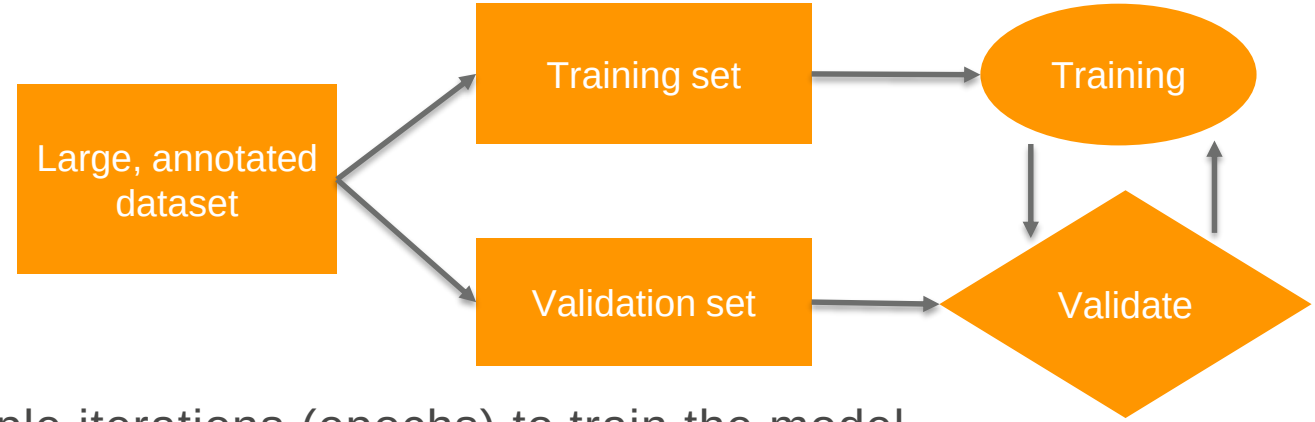


Data split

aws

Model Training

- Define model architecture
- Input the annotated and cleaned data into the model



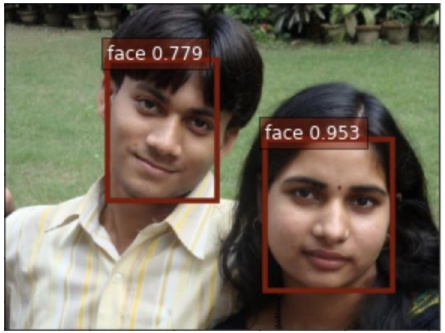
- Multiple iterations (epochs) to train the model
- Validate with held back dataset

Inference

It’s where the magic happens!

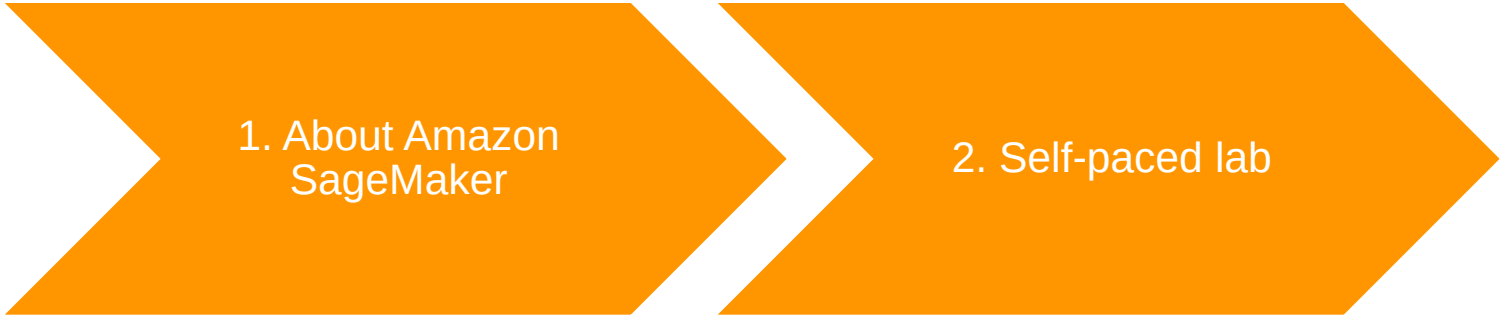
1. Preprocess the new data or image just like a training set.
2. Feed image back to the trained model to get a predicted output.

2. Training a Model in Amazon SageMaker

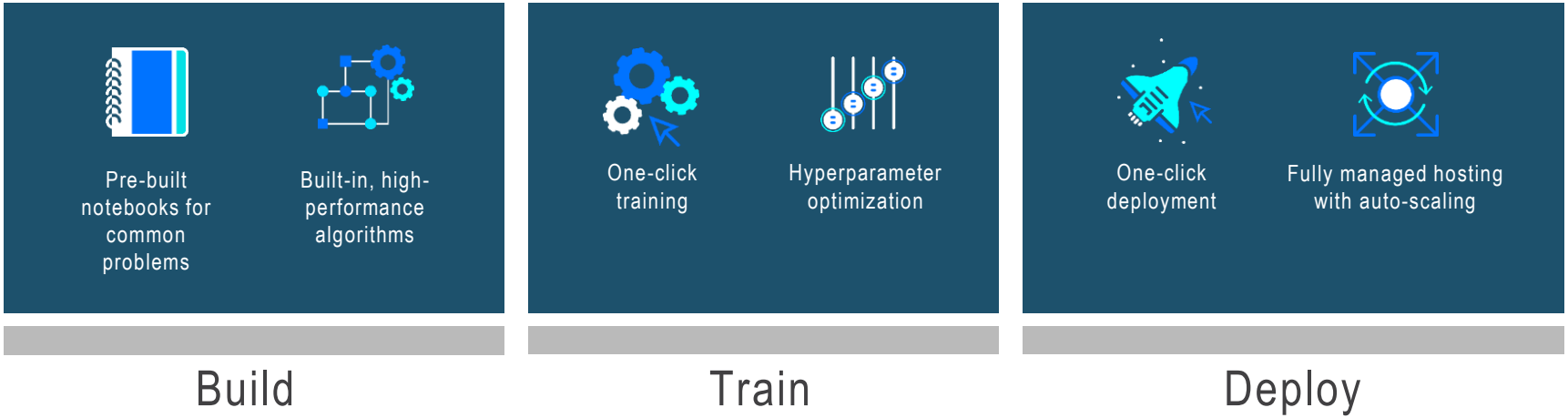


Lab #1: Training a Model in Amazon SageMaker

- Objective: You will learn how to build and train a face recognition model
- Time: 40 min.
- Steps:



Amazon SageMaker



Lab Details – Amazon SageMaker

The self-paced lab will include the following steps:

1. Import a prepared Jupyter Notebook into Amazon SageMaker.
2. The notebook walks you through building a face recognition model in Amazon SageMaker.
3. Create an S3 bucket, and export the updated model there.

aws

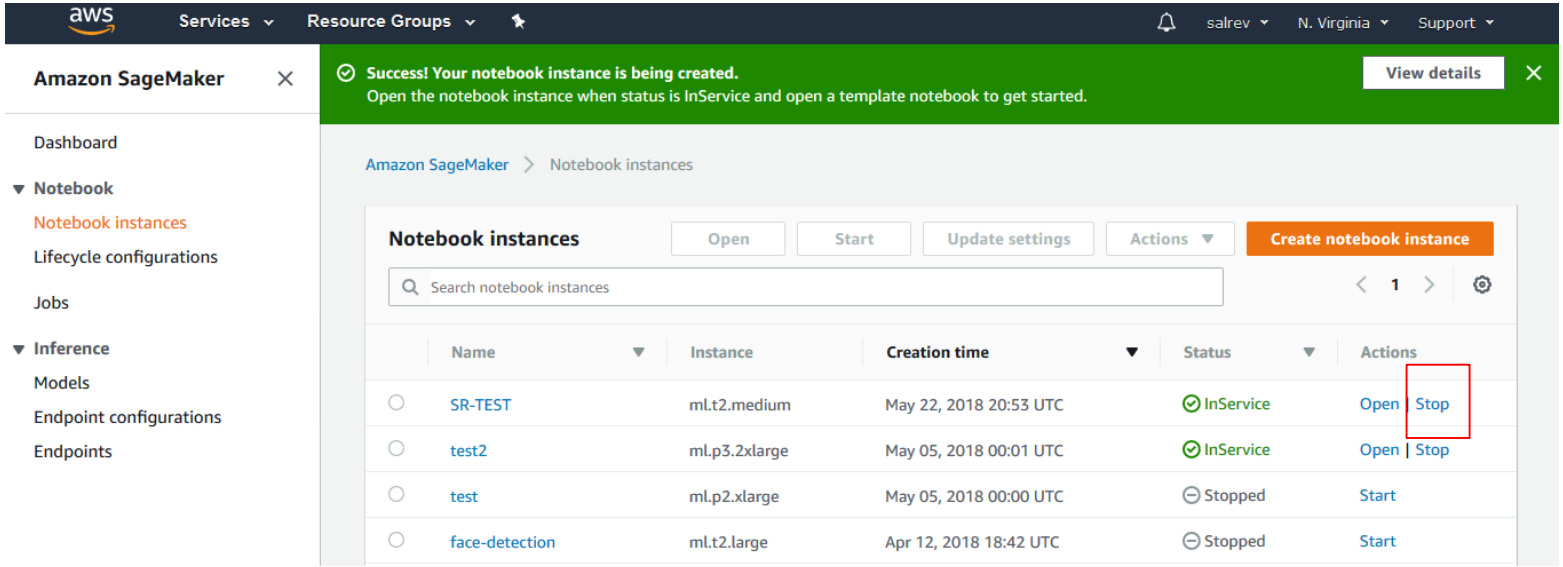
Self-Paced Lab – #1

1. Turn on DeepLens/ if it is turned on- access monitor and open Firefox
2. Find the instructions manual here: https://github.com/fibbonnaci/DeepLens-workshops
3. Access Hands-on Lab 1 tutorial

40 minutes

IMPORTANT NOTE!

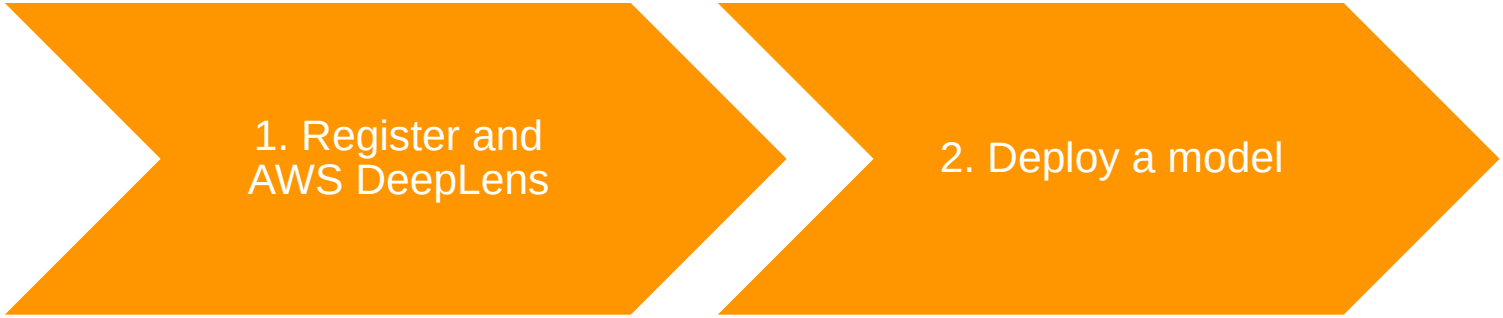
Please ensure you STOP the SageMaker notebook instance to avoid ongoing charges to your AWS account.



3. Deploying a Model to DeepLens

Lab #2: Deploying a Model to AWS DeepLens

- Objective: You will learn how to configure AWS DeepLens and deploy a model
- Time: 40 min.
- Steps:

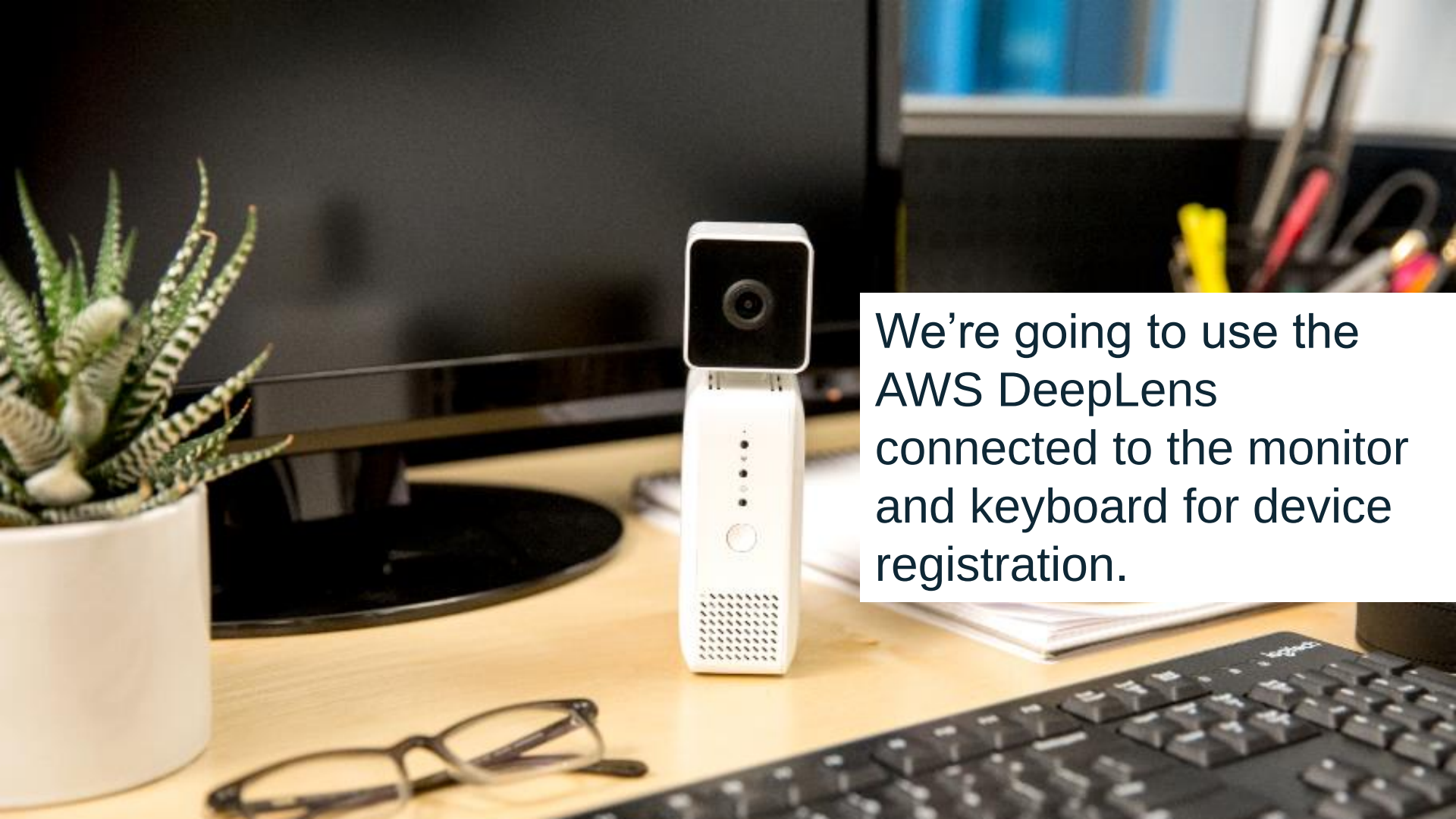


Follow Along Instructions

1. Find the instructions manual here: https://github.com/fibbonnaci/DeepLens-workshops

2. Access Hands-on Lab 2: Register your DeepLens and deploy to device

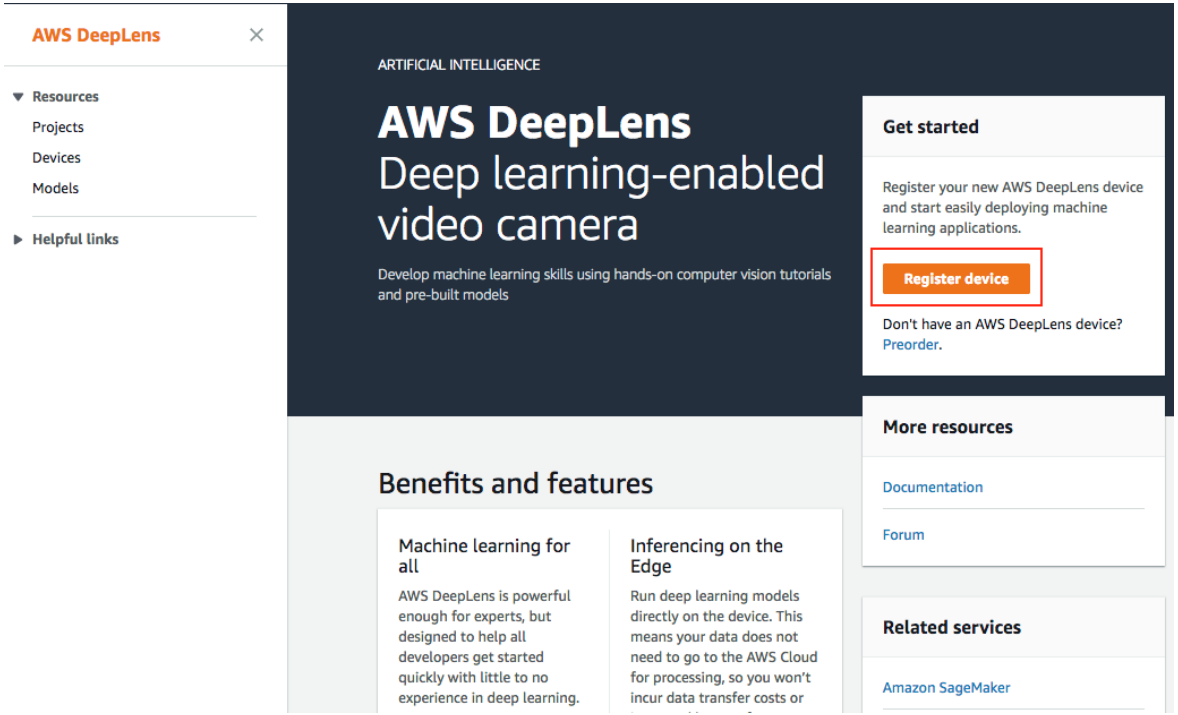
aws



We’re going to use the AWS DeepLens connected to the monitor and keyboard for device registration.

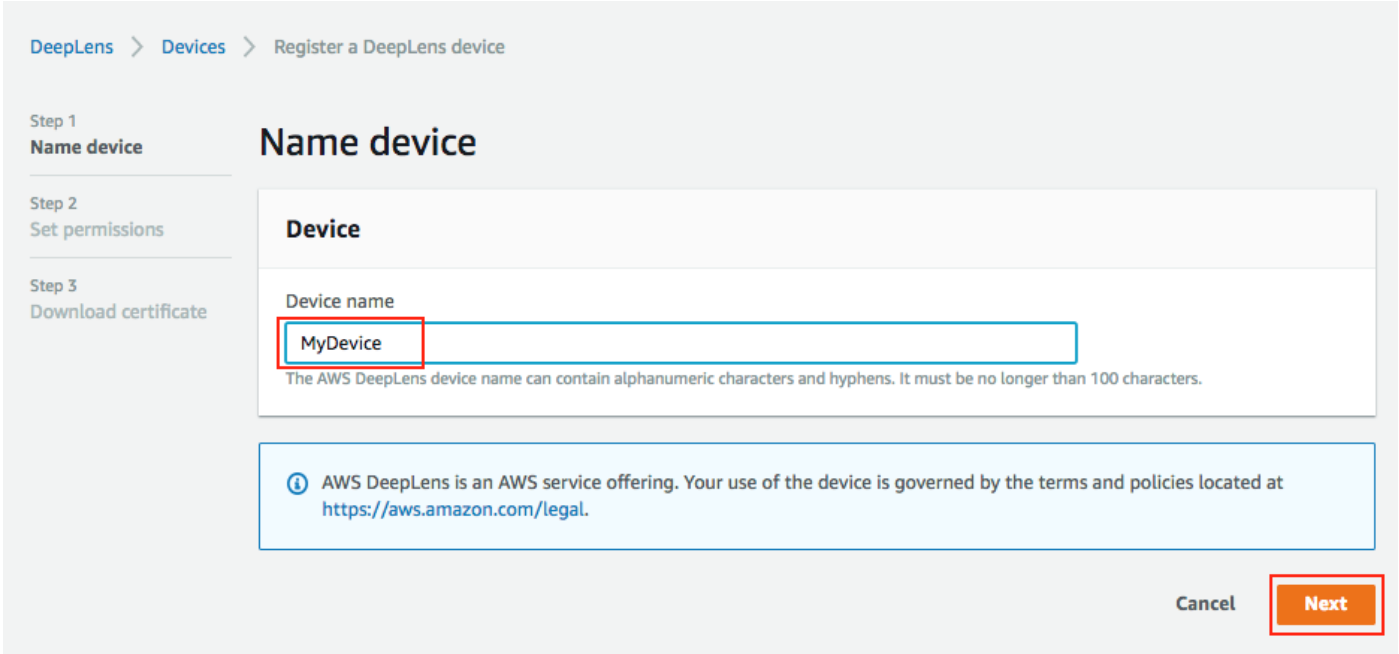
Register AWS DeepLens

1. Choose **Register device**.

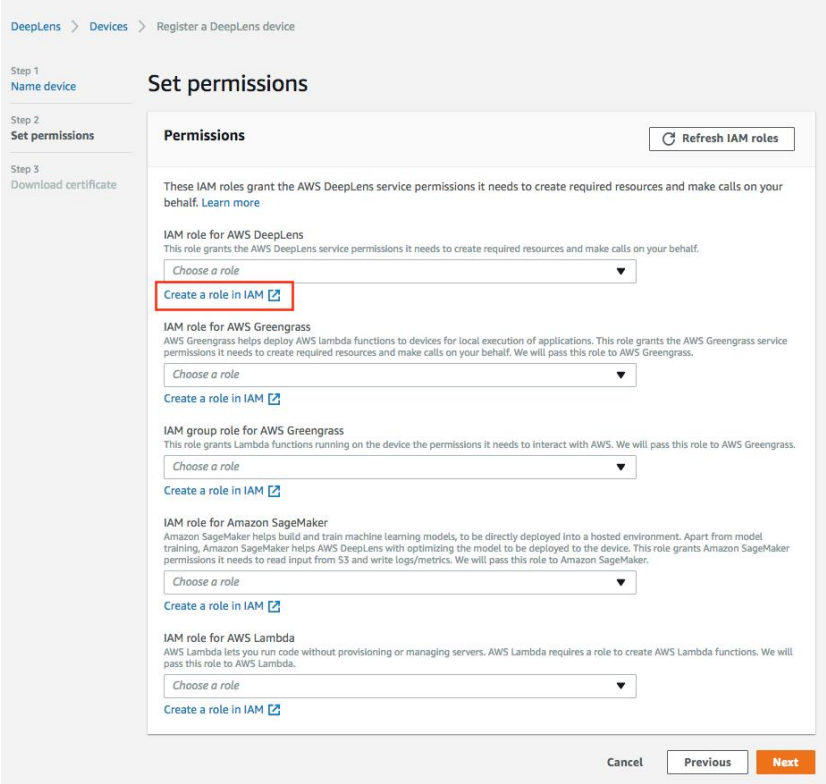


Register AWS DeepLens

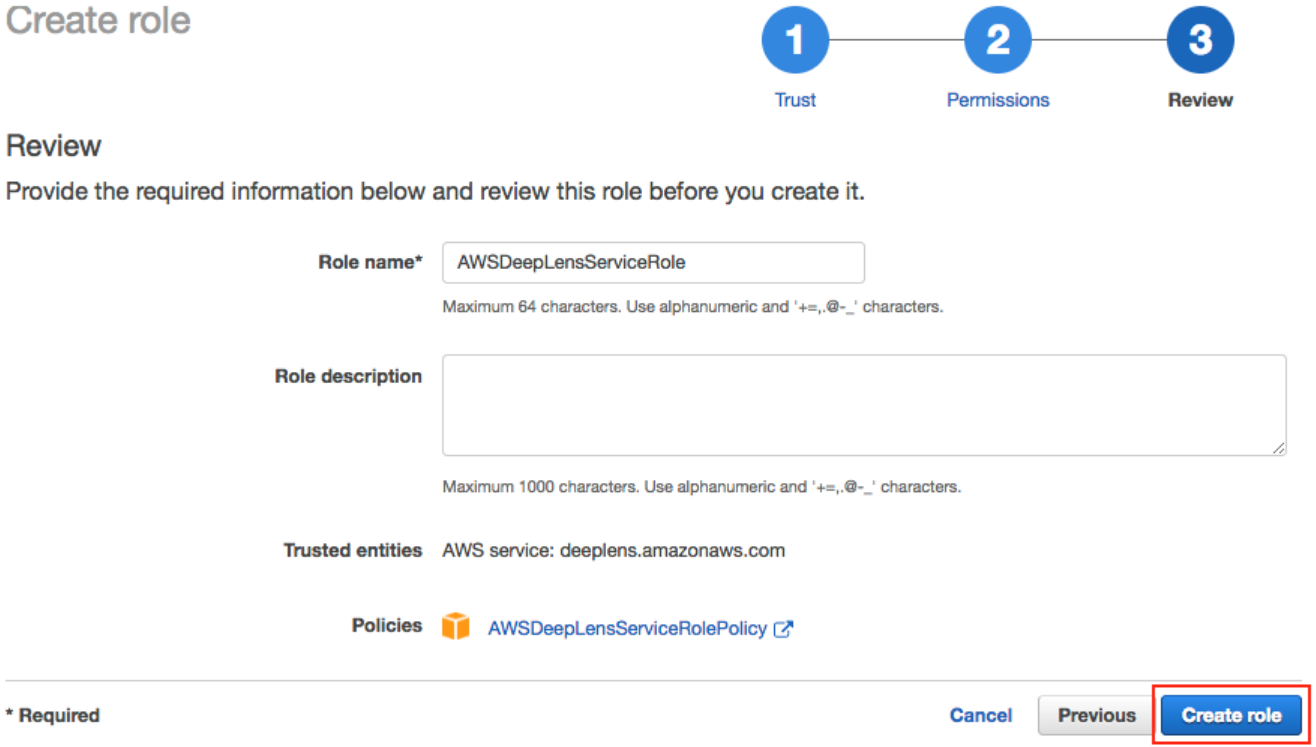
2. Provide a name for your device, for example, *yourname*-sf-AIDC
3. Choose **Next**.



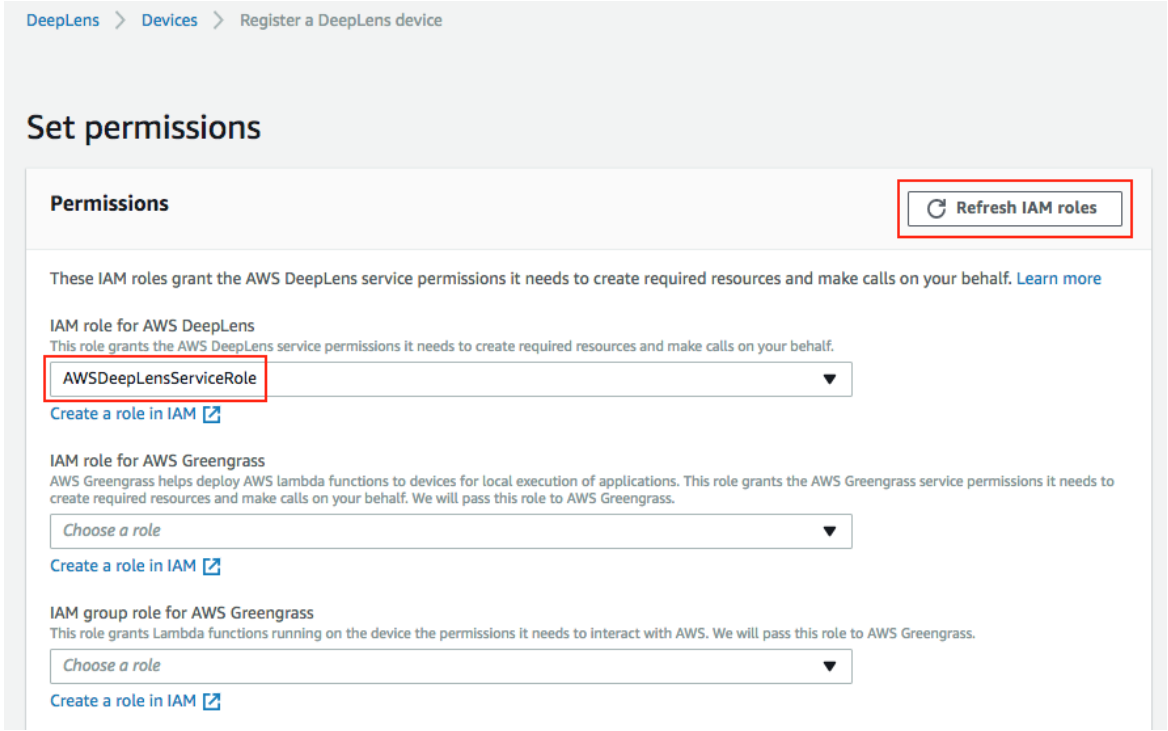
Manage Permissions



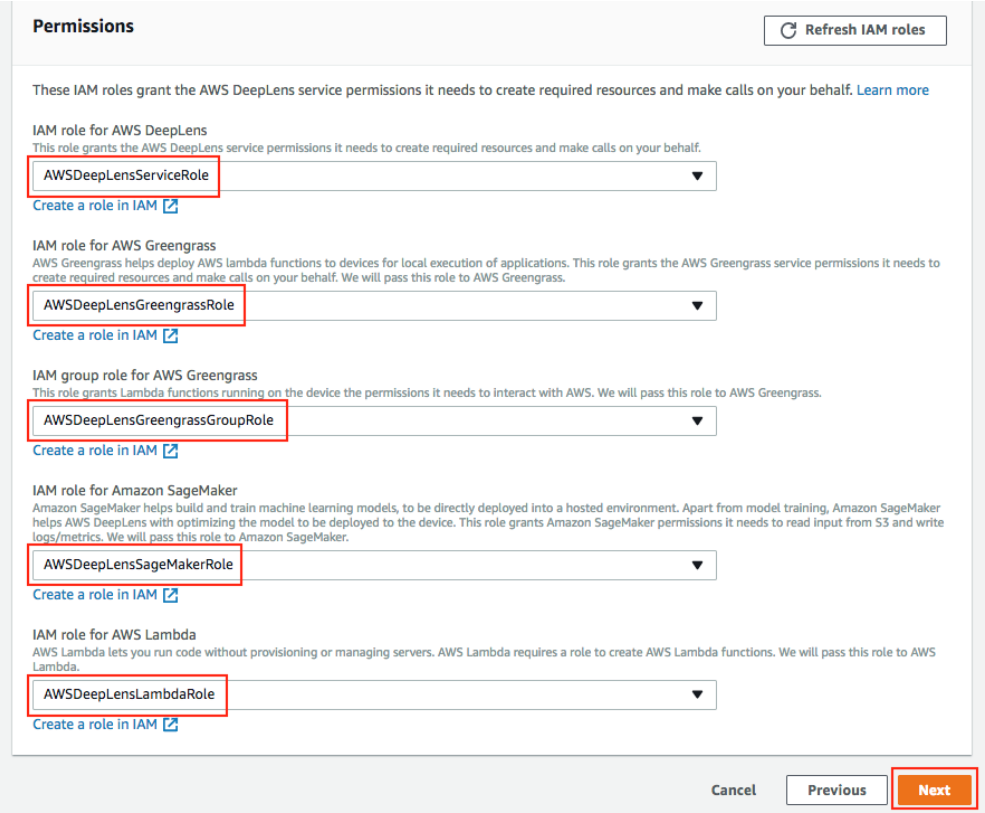
Manage Permissions



Manage Permissions

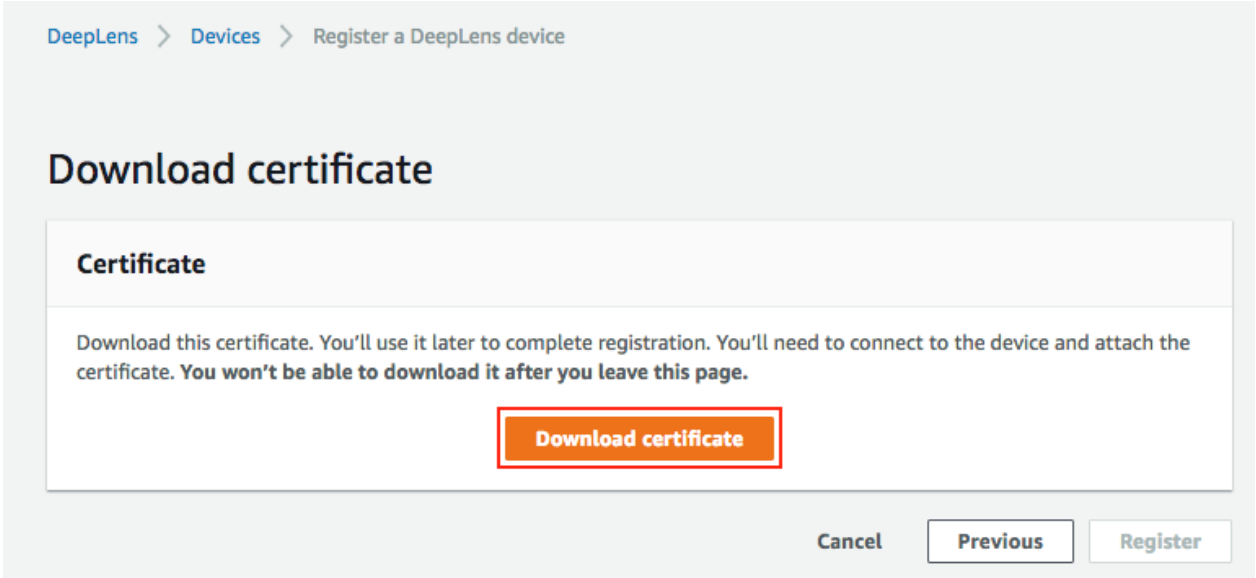


Manage Permissions



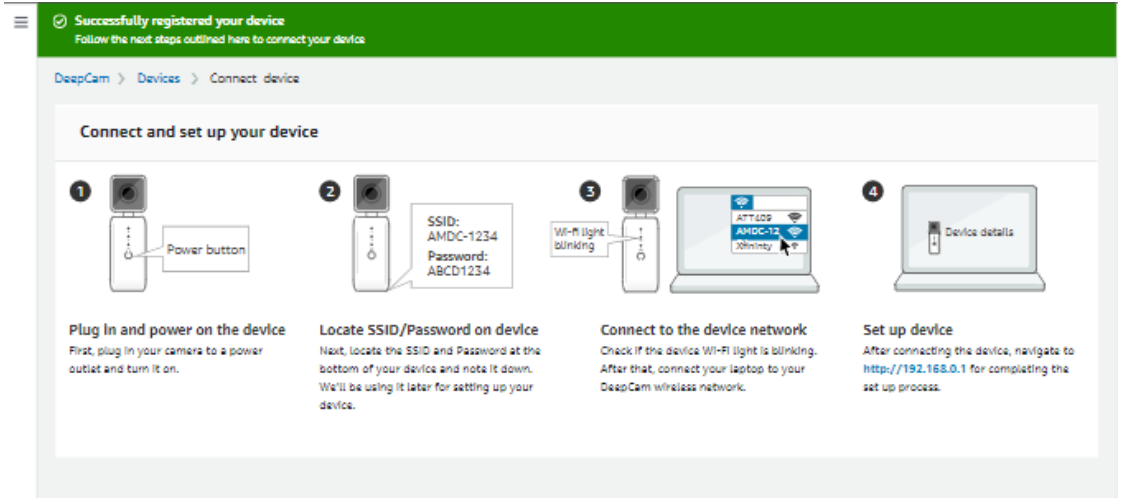
Download Certificate

1. Choose **Download certificate**.
2. Choose **Finish**.



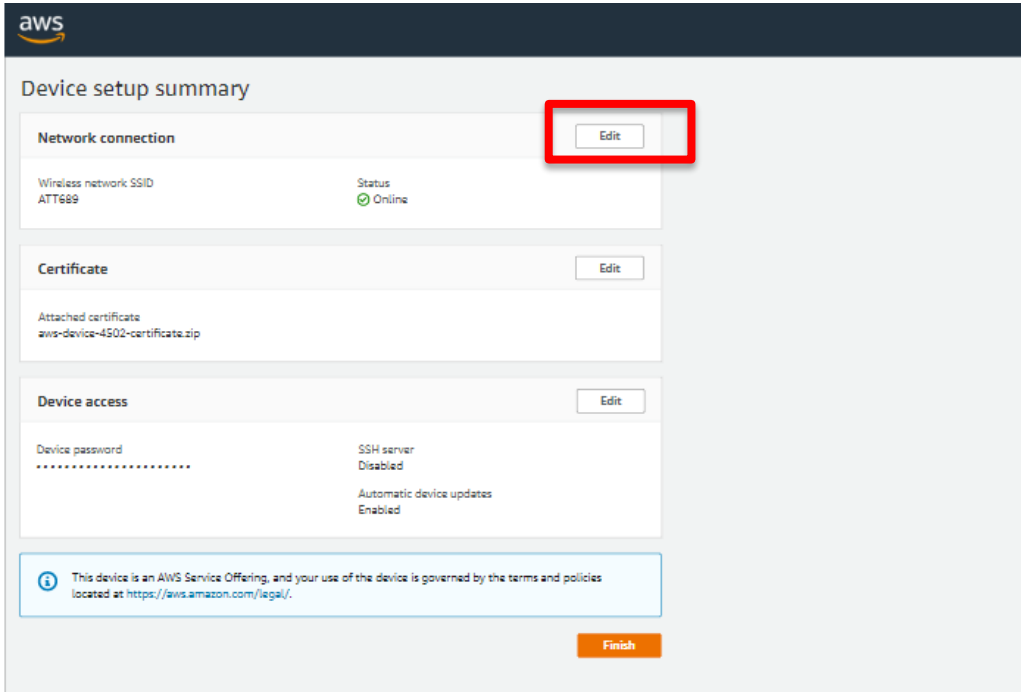
Configure AWS DeepLens

1. Find the reset pin in the back of the AWS DeepLens device.
2. Use the provided pin to reset the device. You should hear a click.
3. The middle LED (Wi-Fi) will be blinking.
4. Connect to 192.168.0.1.



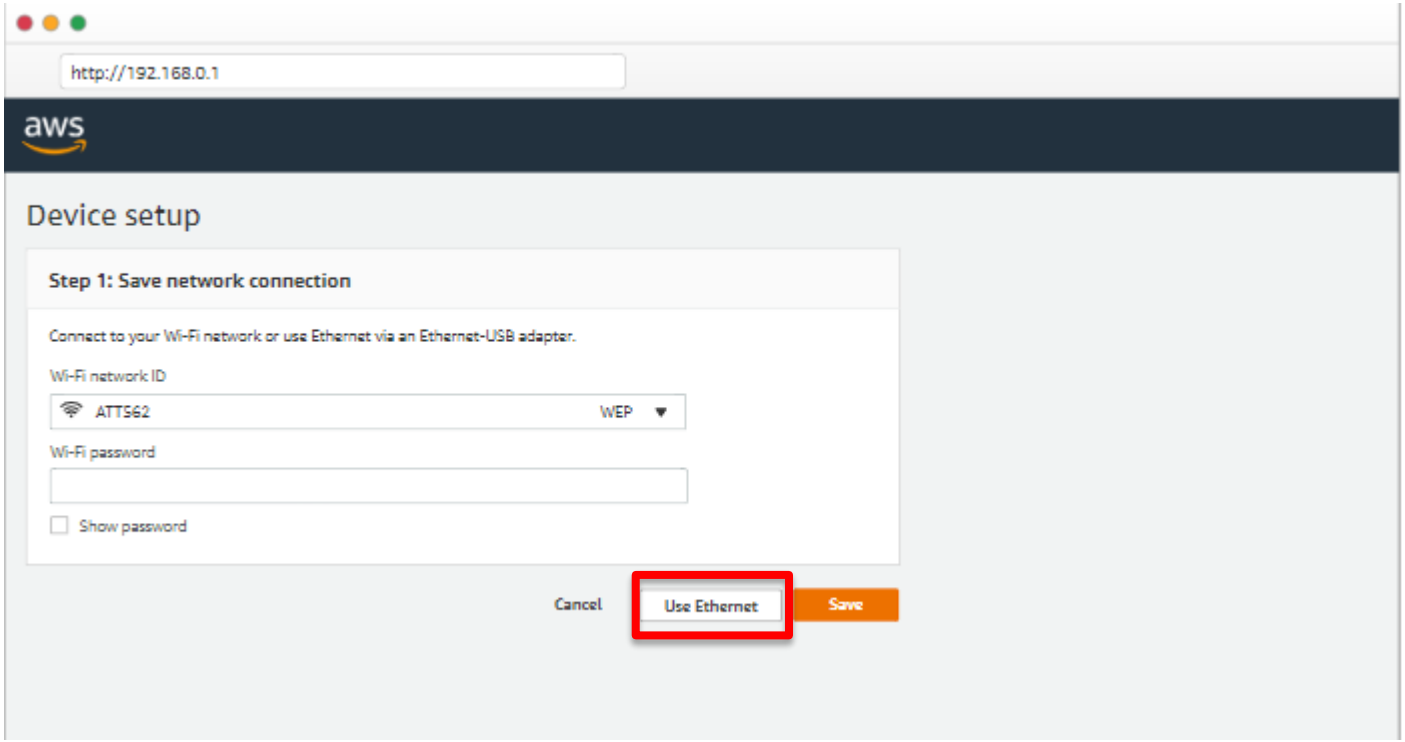
Configure AWS DeepLens

5. For **Network connection**, choose **Edit**.



Configure AWS DeepLens

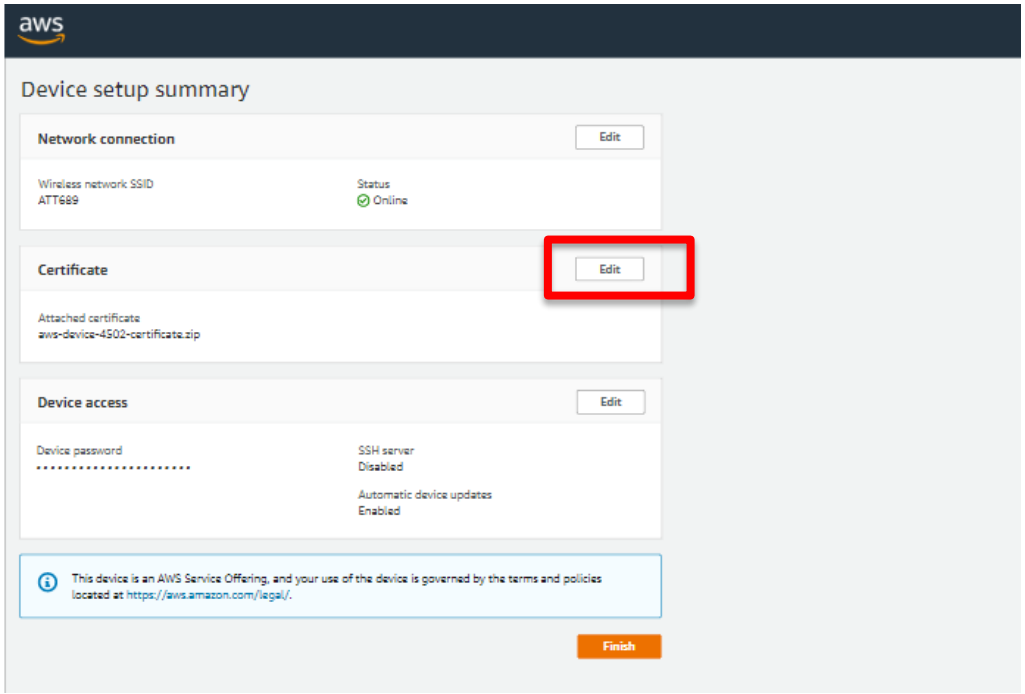
6. Choose **Use Ethernet**.



aws

Configure AWS DeepLens

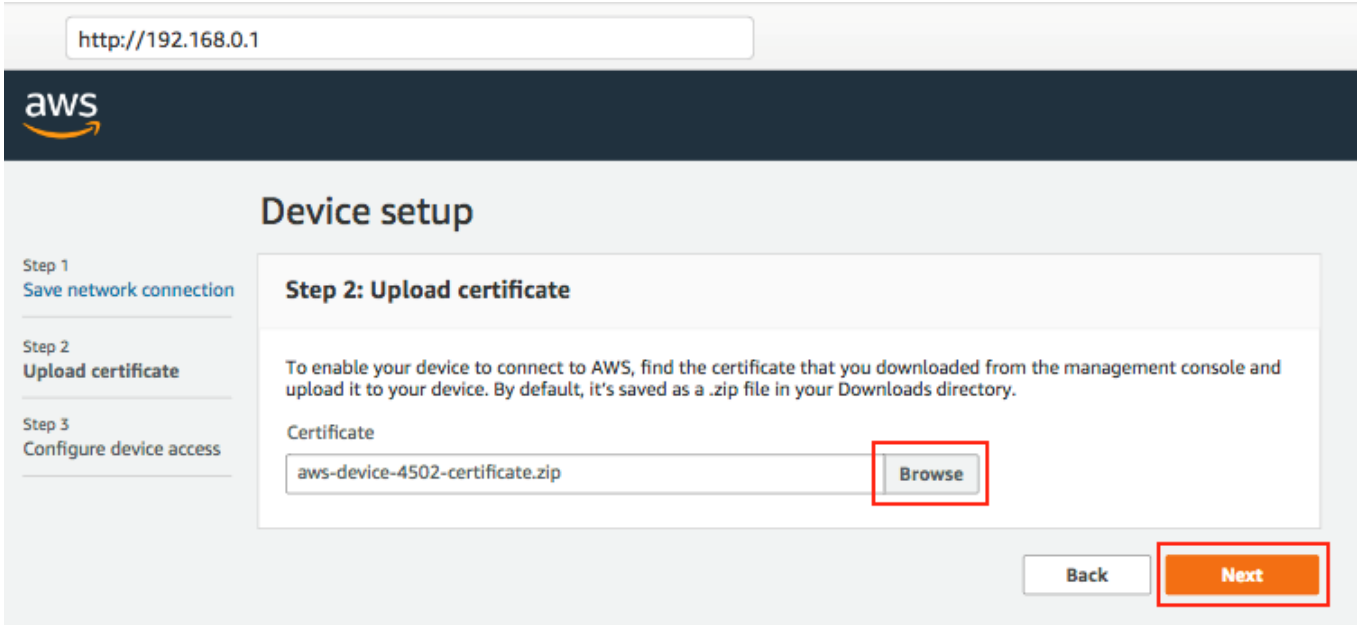
7. For **Certificate**, choose **Edit**.



Configure AWS DeepLens

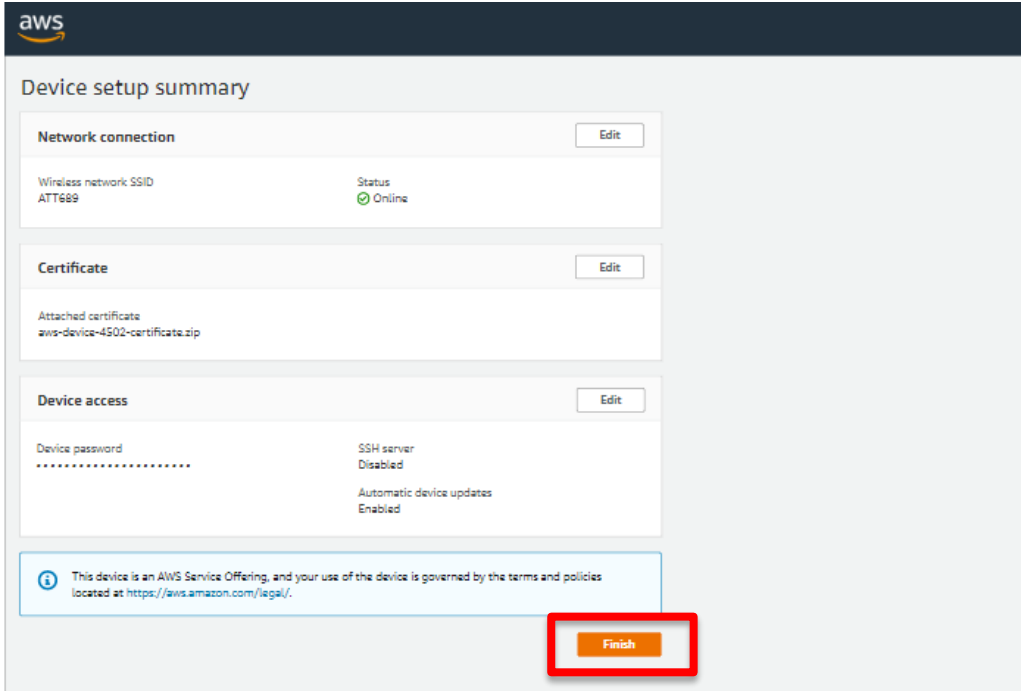
8. Upload the .zip file you downloaded during registration.

9. Choose **Next**.



Configure AWS DeepLens

10. Choose **Finish**.



Run this step before moving ahead

1. Open **Terminal**, and run this command:

```
sudo systemctl restart greengrassd.service --no-block
```

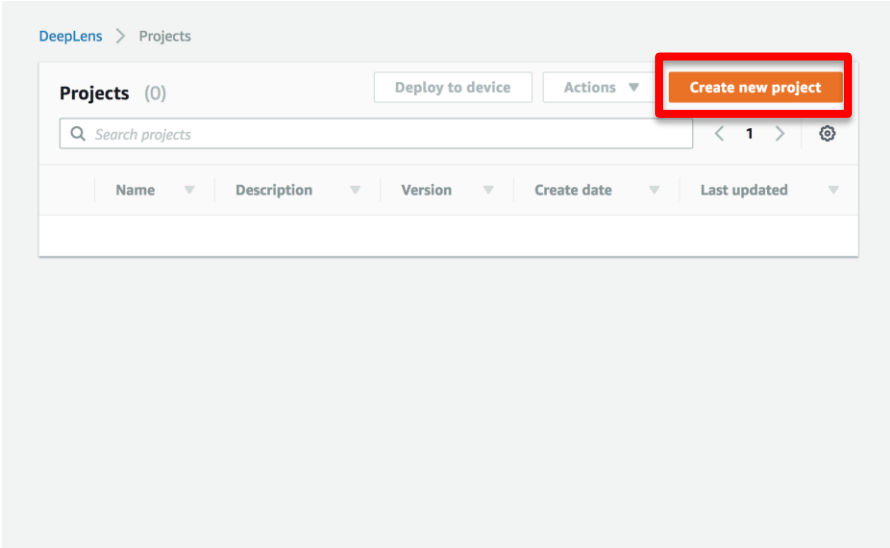
Now, It's Time to Create a Project

1. Log in to the AWS DeepLens console.

   https://console.aws.amazon.com/deeplens

   This is the AWS DeepLens console.

2. Choose **Create Project**.

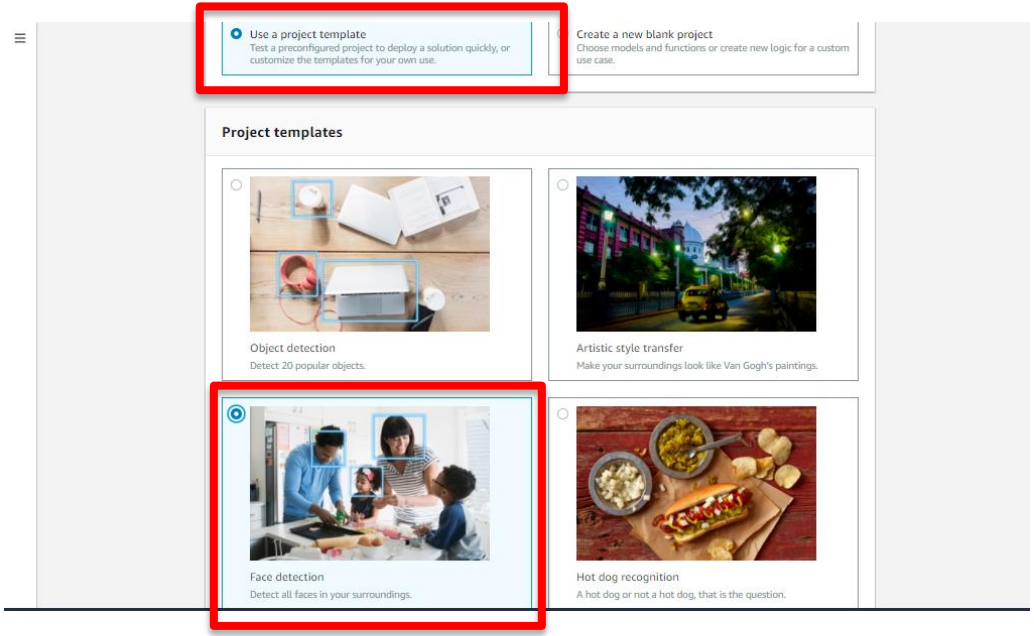


Use a Face Detection Sample

3. Select **Use a project template**.

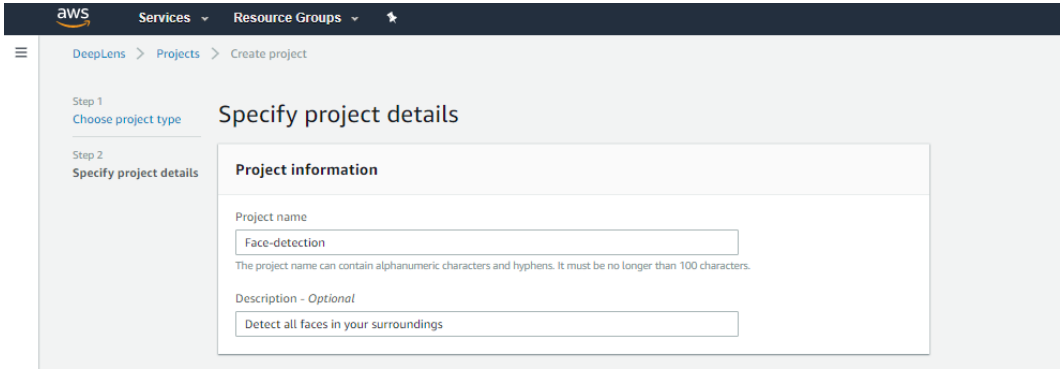
4. Select **Face detection** from sample project templates.

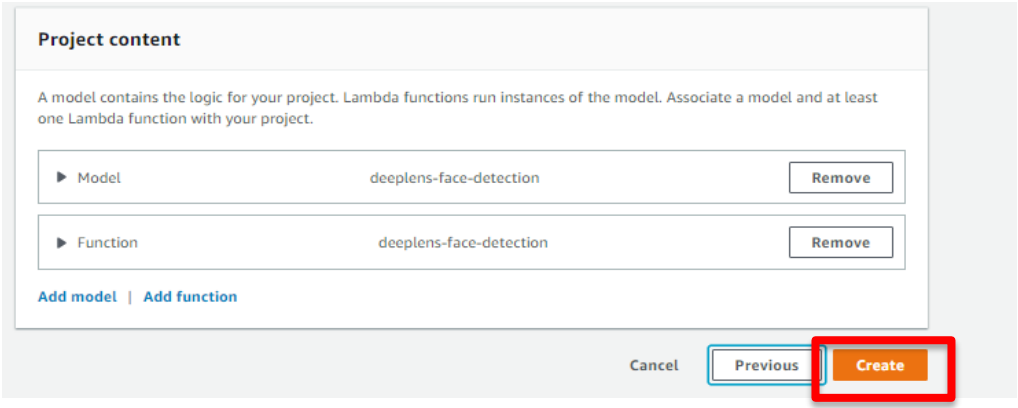
5. Choose **Next** at the bottom of screen.



Create a Project

6. Choose **Create**.



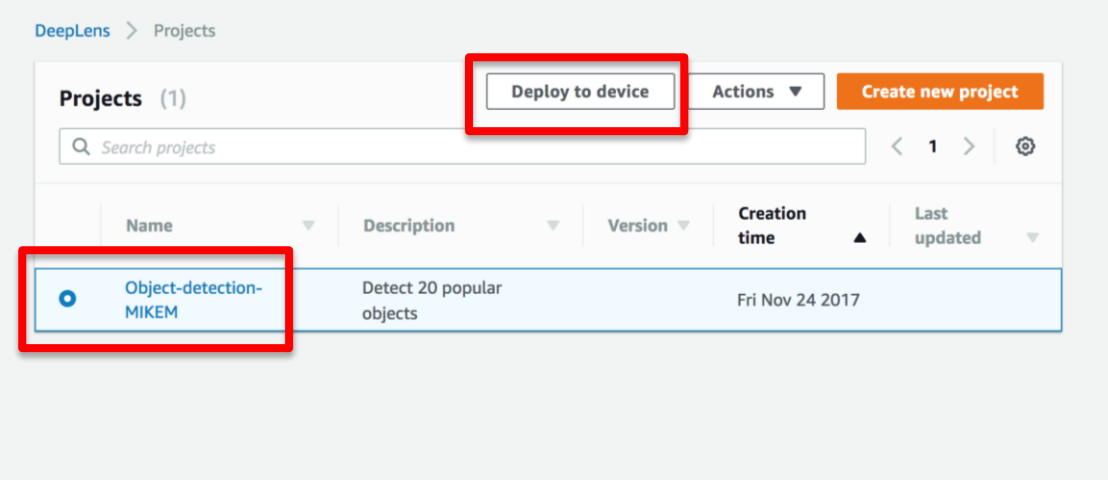


Deploy Project to the Device

7. Find your project in the list (the one you just named).

8. Choose the radio button.

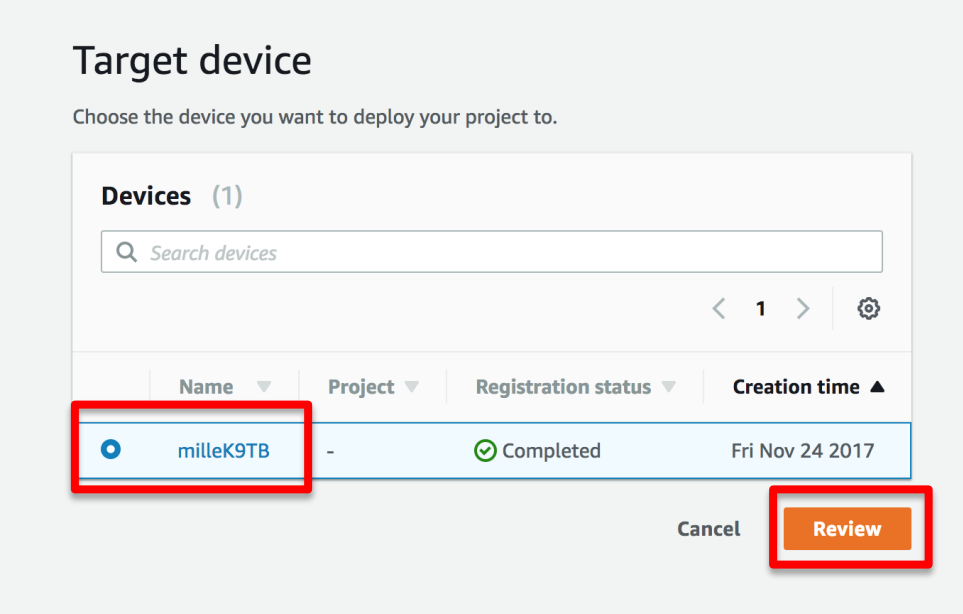
9. Choose **Deploy to device**.



Target Your Device

10. Select your device.

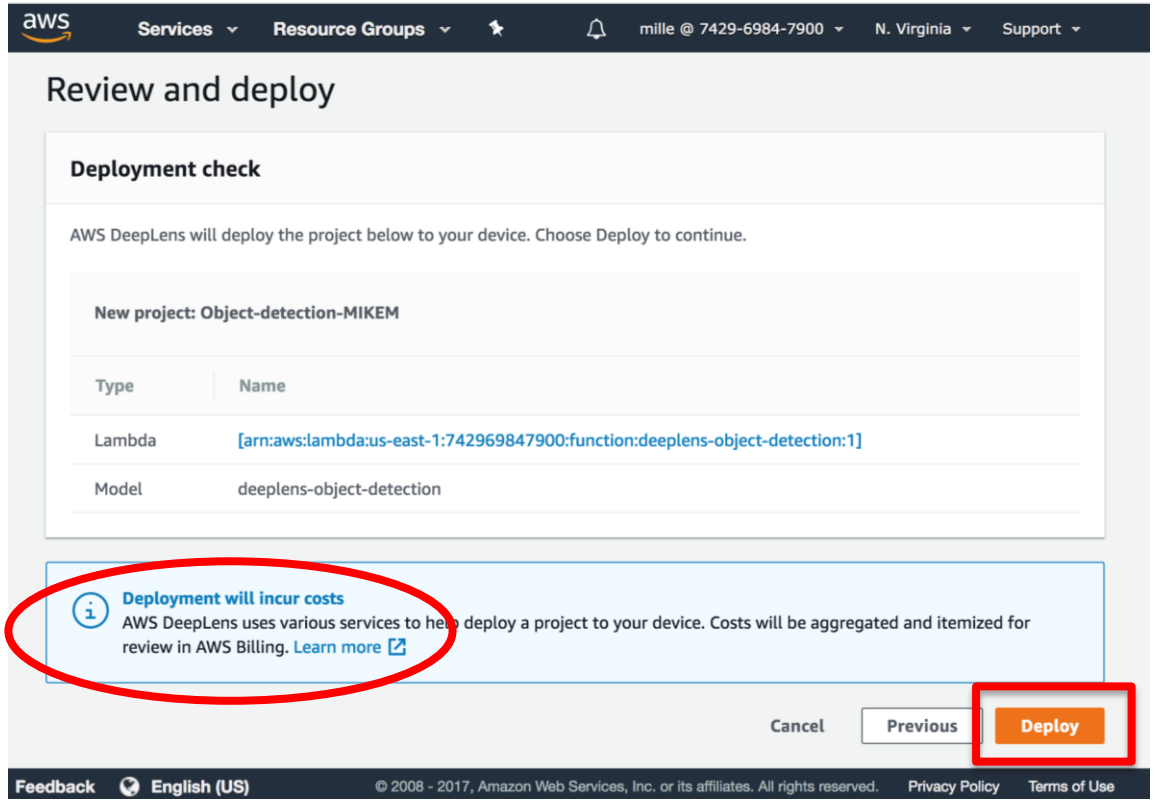
11. Choose **Review**.



Deploy!

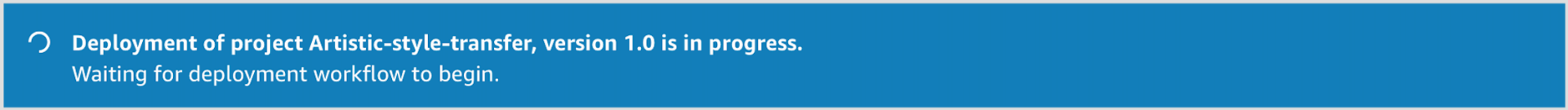
12. Choose **Deploy**.

A note on costs …

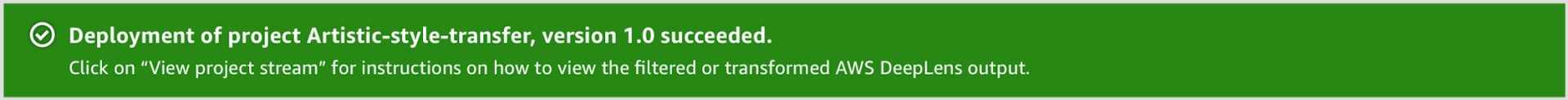


Wait for the Project to be Deployed

Blue banner = Deployment in progress



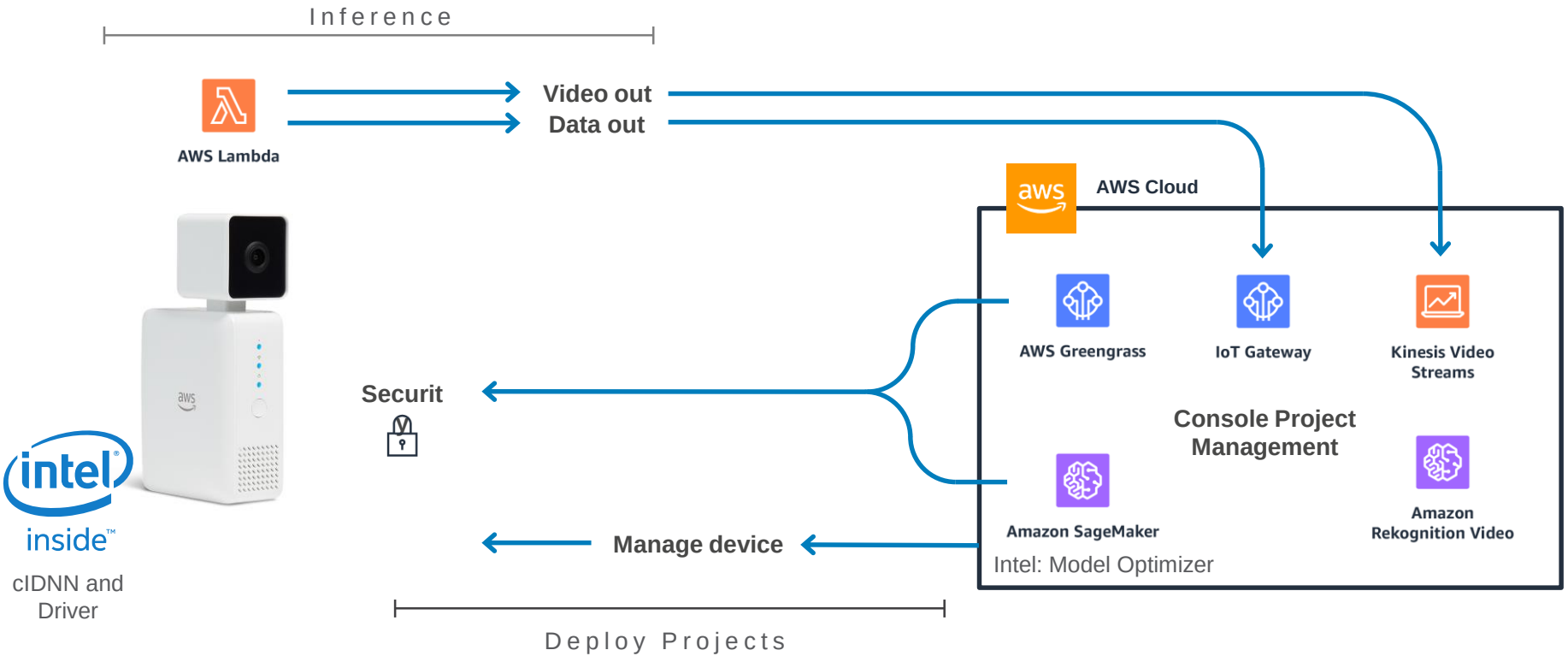
Green banner = Deployment successful



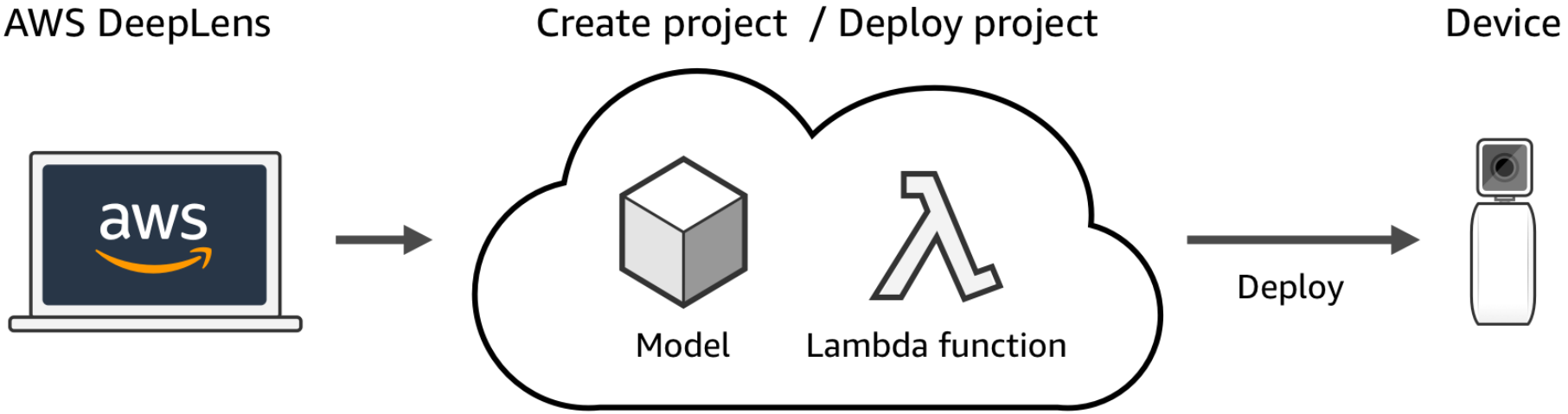
AWS DeepLens Specifications

- Intel Atom Processor
- Gen9 graphics
- Ubuntu OS- 16.04 LTS
- 100 GFLOPS performance
- Dual band Wi-Fi
- 8-GB RAM
- 16-GB storage (eMMC)
- 32-GB SD card
- 4 MP camera with MJPEG
- H.264 encoding at 1080p resolution
- 2 USB ports
- Micro HDMI
- Audio out
- AWS Greengrass preconfigured
- Intel clDNN Optimized for MXNet

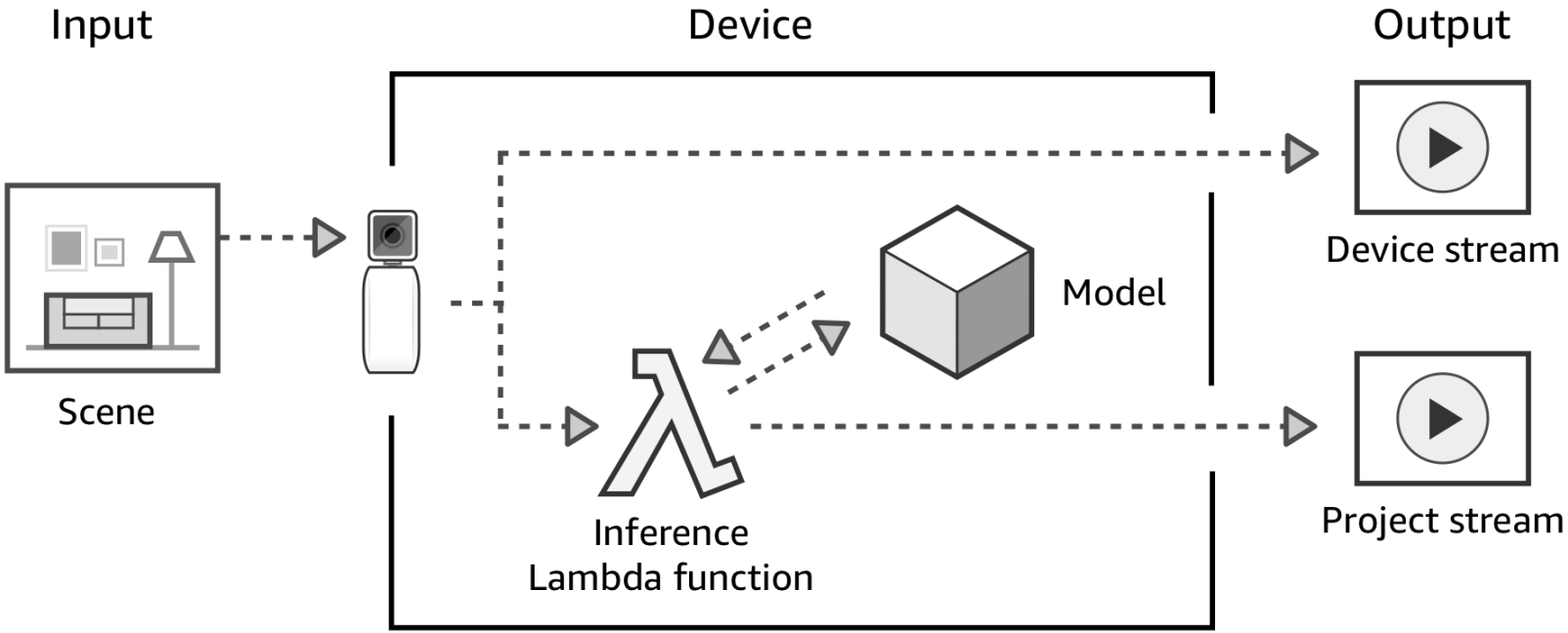
AWS DeepLens Architecture



Under the Covers – Console



Under the Covers – Device



Let’s go back to the console and view the output

13. Click **View project stream** for instructions.

Winners of the DeepLens Hackathon

First place

ReadToMe
Created by Alex Schultz

ReadToMe is a deep learning enabled application that is able to read books to kids. In this case, reading Green Eggs and Ham, by Dr. Seuss.

Second place

Dee
Created by Matthew Clark

Dee is a fun AWS DeepLens interactive device for children. The device asks children to answer questions by showing a picture of the answer.

Third place

SafeHaven
Created by Nathan Stone and Peter McLean

SafeHaven uses Alexa and AWS DeepLens to bring peace of mind for vulnerable people and their families.

View all 23 projects at: https://aws.amazon.com/deeplens/community-projects

4. Extending a Project: Audience Response Tracking with AWS

SUBMITTED TO

AWS DeepLens Challenge

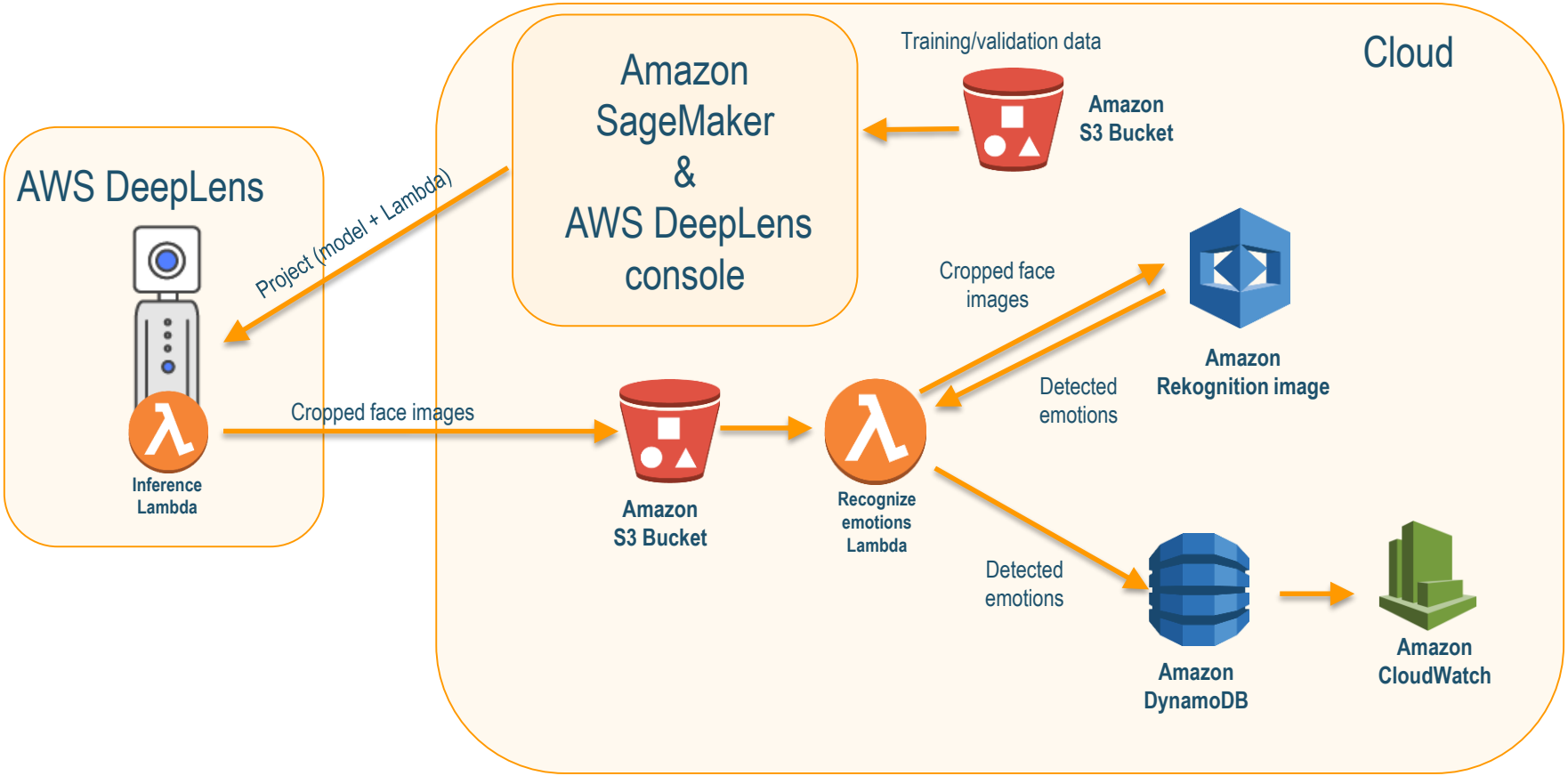
CREATED BY

Ricardo Mota

Jidesh Veeramachaneni

Report inappropriate content

Amazon statement regarding Amazon Rekognition: "our customers are really excited about how they can solve real world problems with Rekognition...as always, with all our services, we encourage our law enforcement customers to work with local government officials to develop acceptable use policies for facial recognition technologies that both protects the rights of citizens and enables law enforcement to do their job".



Now, let’s see it in action.

WHAT'S NEW WITH AWS DEEPLENS

May 24th:

1. Introducing AWS DeepLens Tech Talk

Tell your friends to sign up now:
https://aws.amazon.com/about-aws/events/

June 14th:

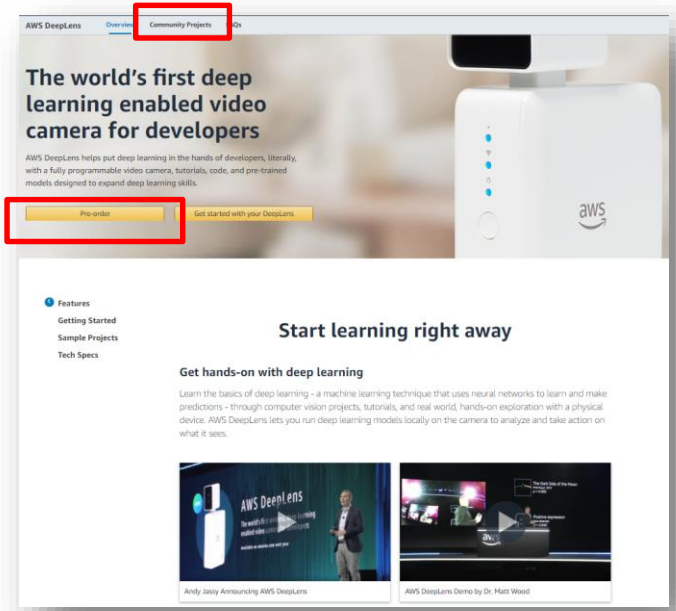
1.Shipping commences from Amazon.com plus product updates and new features!

aws

Thanks & Wrap-Up

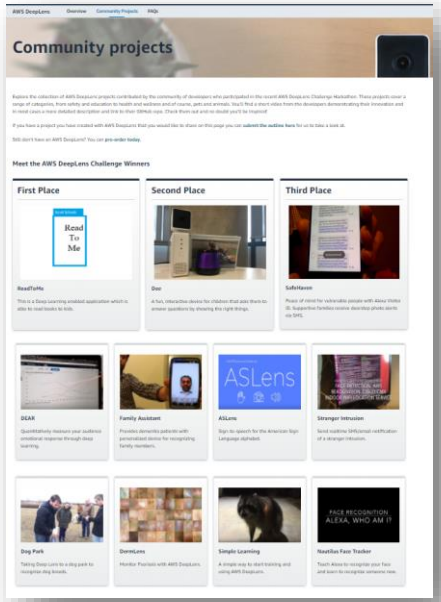
Order your device

aws.amazon.com/deeplens/



Learn more

aws.amazon.com/deeplens/community-projects



Request a workshop

Work with your AWS account management team to request a hands-on Amazon SageMaker & AWS DeepLens workshop


Thank you!