|  |
| --- |
| Artificial Intelligence-Machine Learning (AI-ML) Enterprise Design Patter (EDP): Guidance to set up an AI Proof-of-Concept (POC) Lab environment in Microsoft (MS) Azure cloud |

**Introduction**

This document provides guidance / instructions for setting up the lab environment in the MS azure cloud for AI research in clinical applications for proof of concept studies. The Proof of Concept (POC) lab environment does not require formal VA funding and associated protocols such as getting a VIPR number. Once a POC has been established, researchers can conduct studies using actual VA data and information within the secure GovCloud environment

Researchers should first obtain access to the azure lab environment. Instructions on how to get access is provided in < document name > <location>

It is assumed that researchers have knowledge and experience in applying AI and ML techniques for clinical applications such as access to ML algorithms and relevant medical images in the Digital Imaging and Communications in Medicine (DICOM) format. The lab environment provides researchers the cloud infrastructure to conduct proof of concept studies. Key infrastructure components with the lab environment include the Blob storage (for the images), Graphics Processing Unit (GPU) resources required for an AI in clinical applications research.

Medical image classification project with the Custom Vision Online Interface.

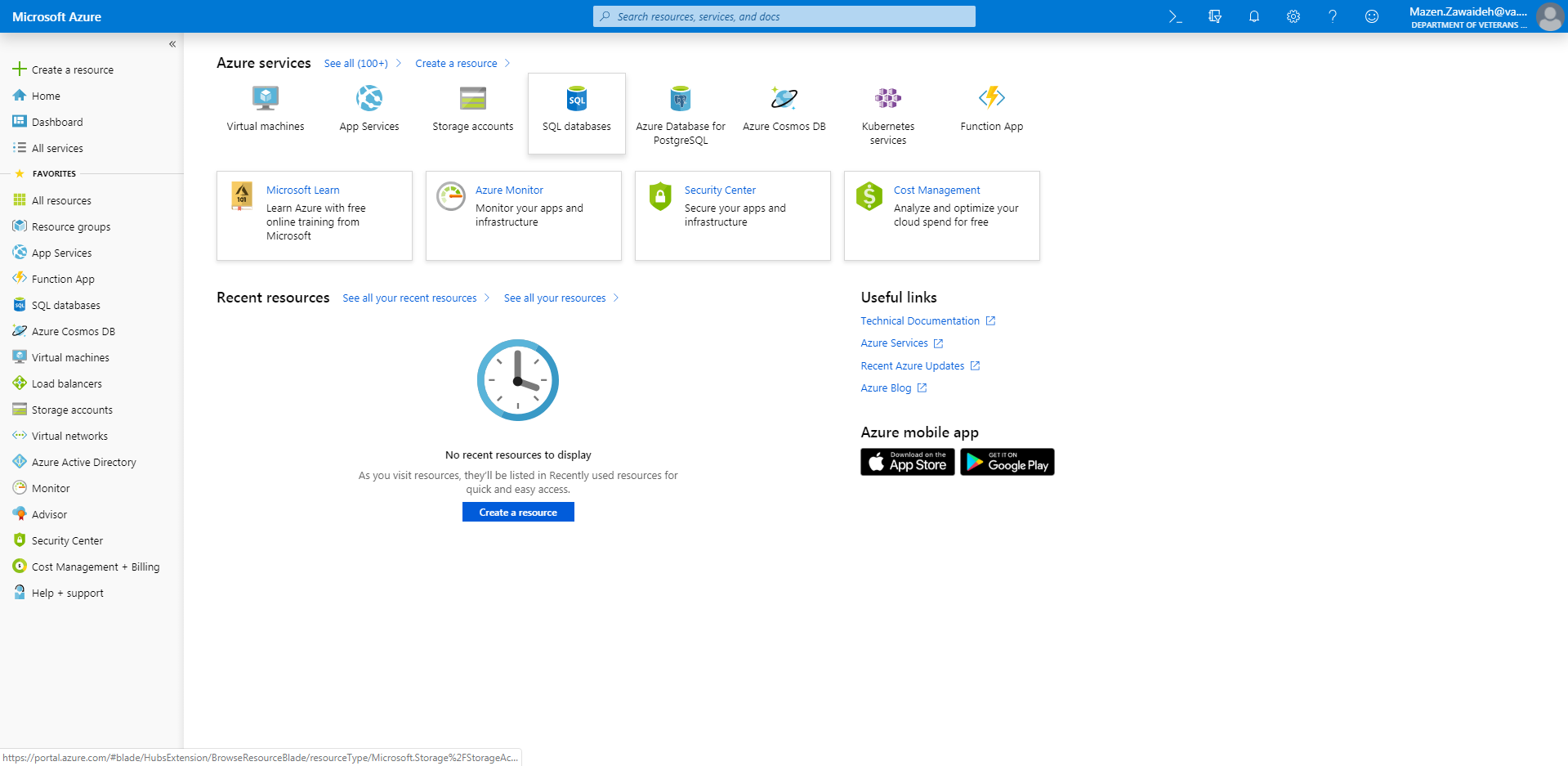
*Note: The lab environment is in a commercial space and outside of the VA network and hence researchers are advised not to use sensitive or confidential data such as Personally Identifiable Information (PII) and Protected Health Information (PHI)*

Guidance for setting up a lab environment using the Custom Vision Online Interface is provided below as a series of steps / workflow. The lab can be used for image classification studies.

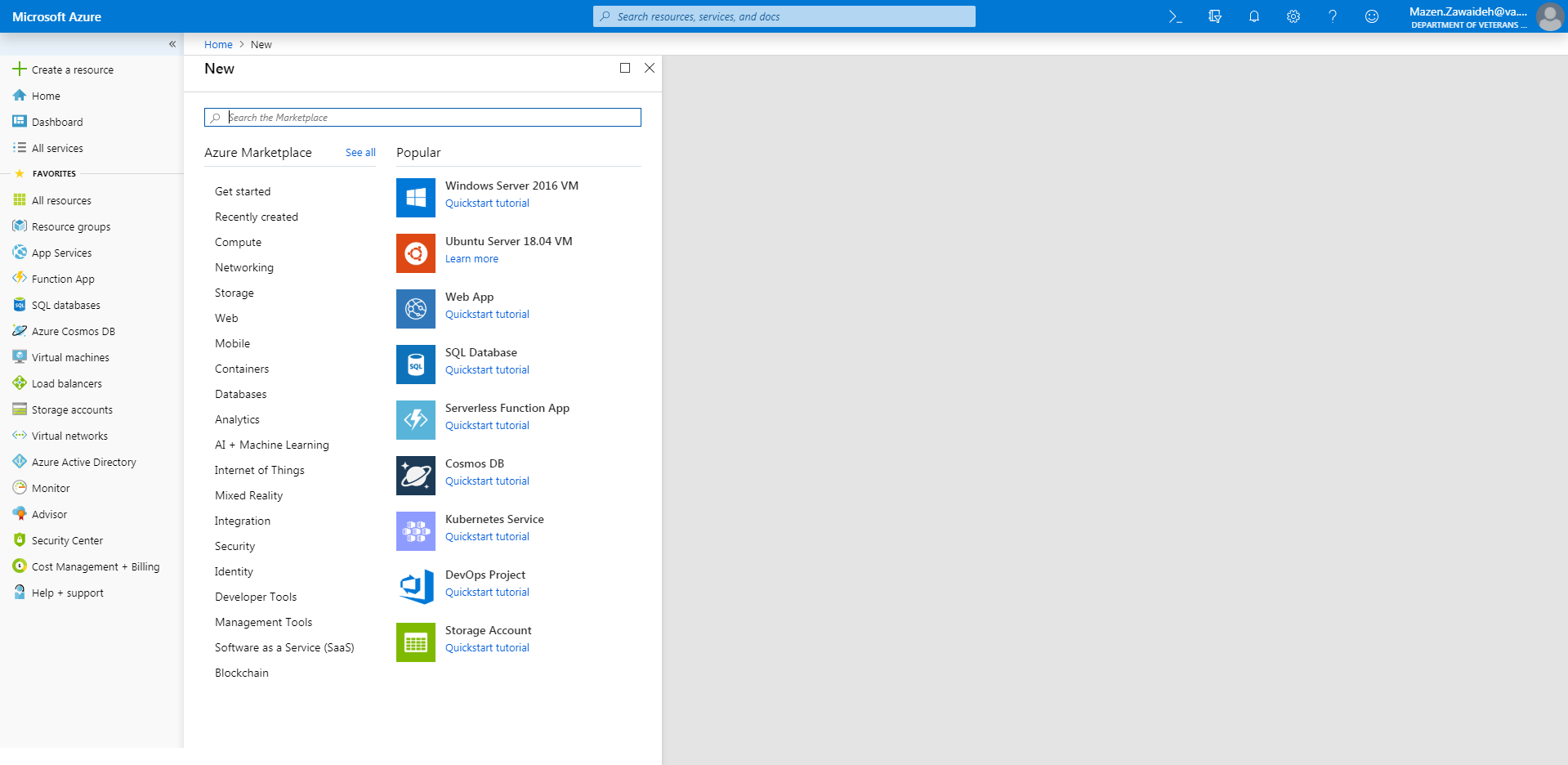
*For more experienced users who wish to perform this task programmatically, see the sample code at the official Custom Vision documentation (https://docs.microsoft.com/en-us/azure/cognitive-services/custom-vision-service/python-tutorial).*

After your Custom Vision instance is created, you can add tags, upload images, train the project, obtain the project's published prediction endpoint URL, and use the endpoint to programmatically test an image. Use this guide as a template for building your own application.

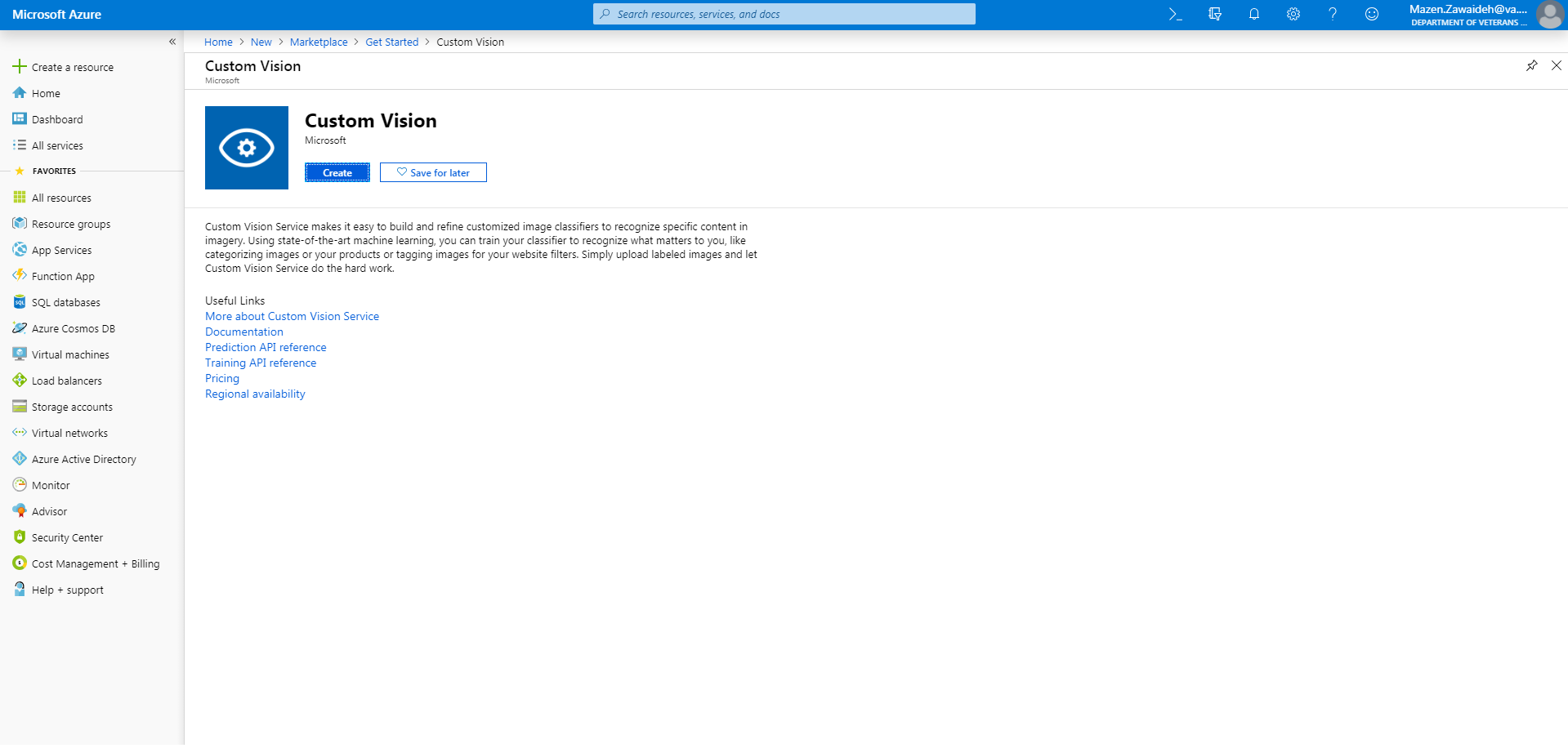
1. Provision the required resources in your Azure EDE account
   1. Sign into Portal



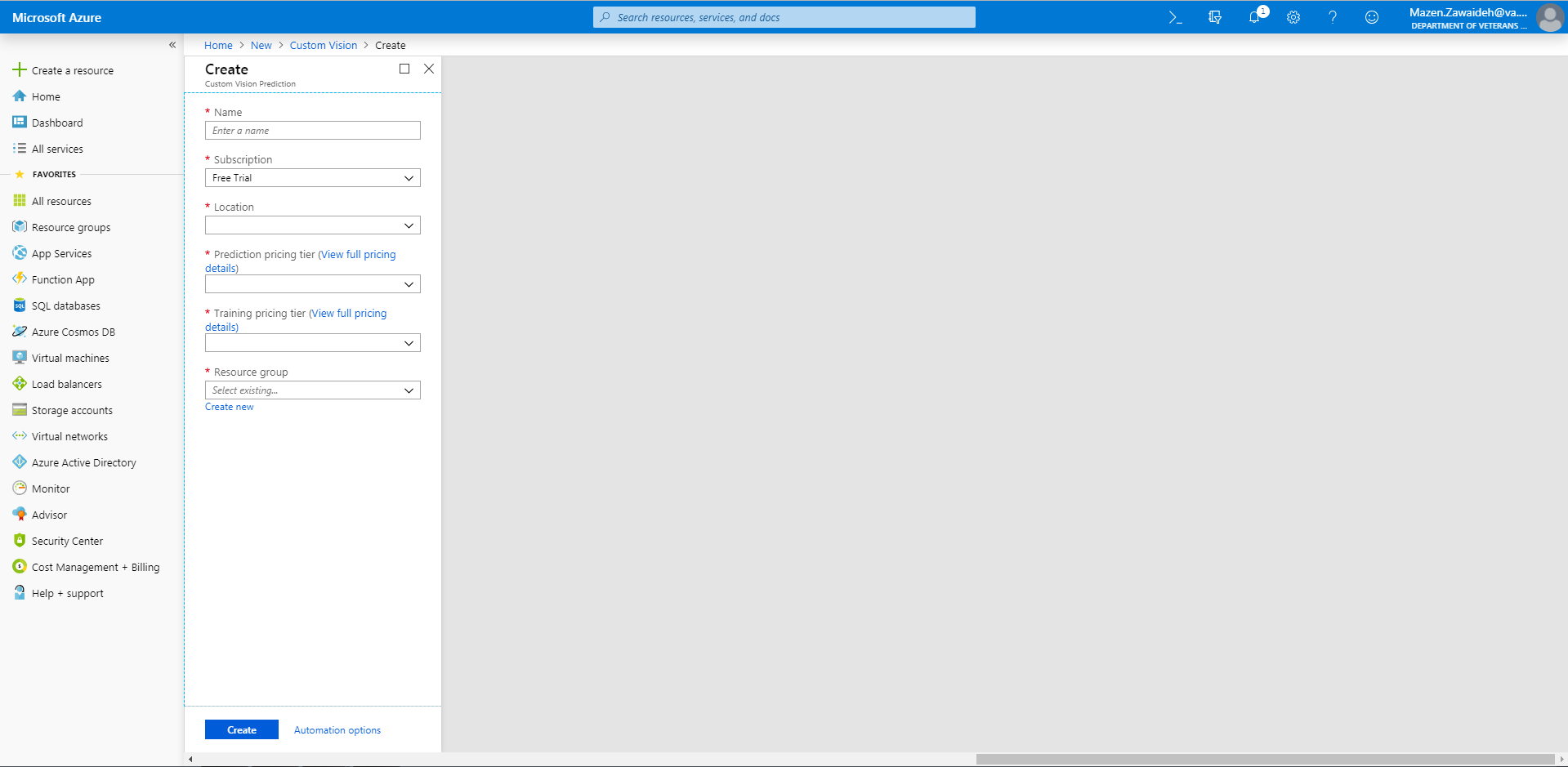
* 1. Click create a resource



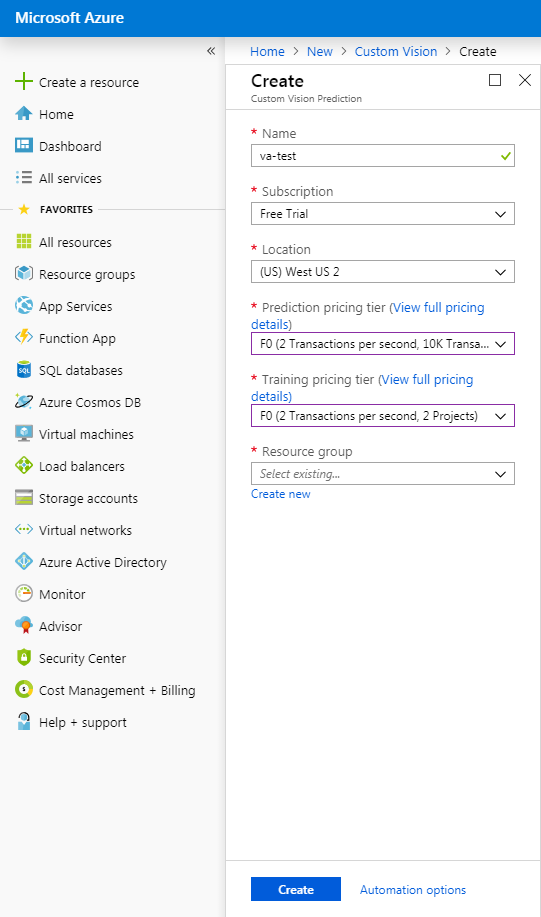
* 1. In the search box, search for “Custom Vision” , then select this.



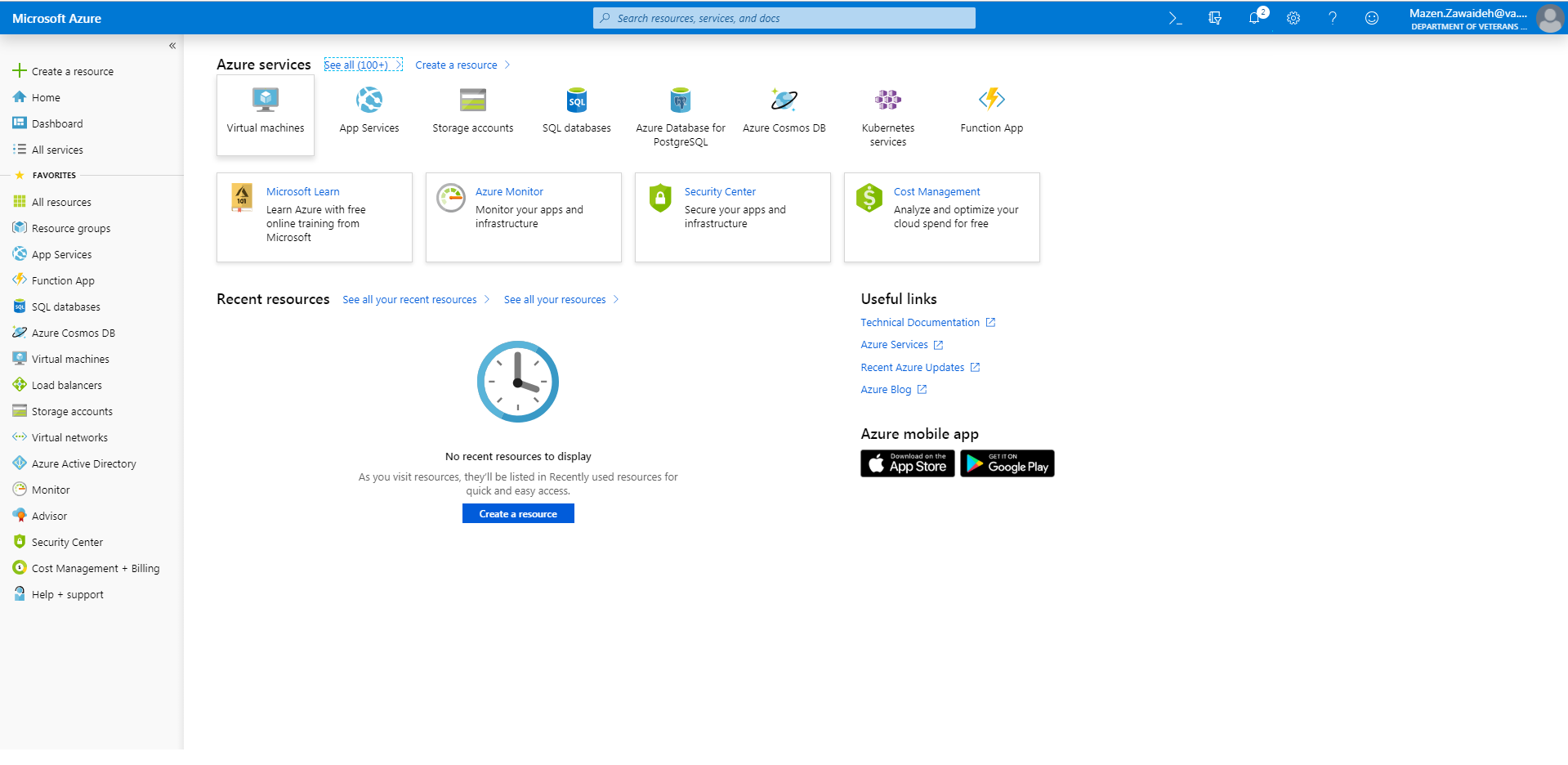
* 1. Click Create



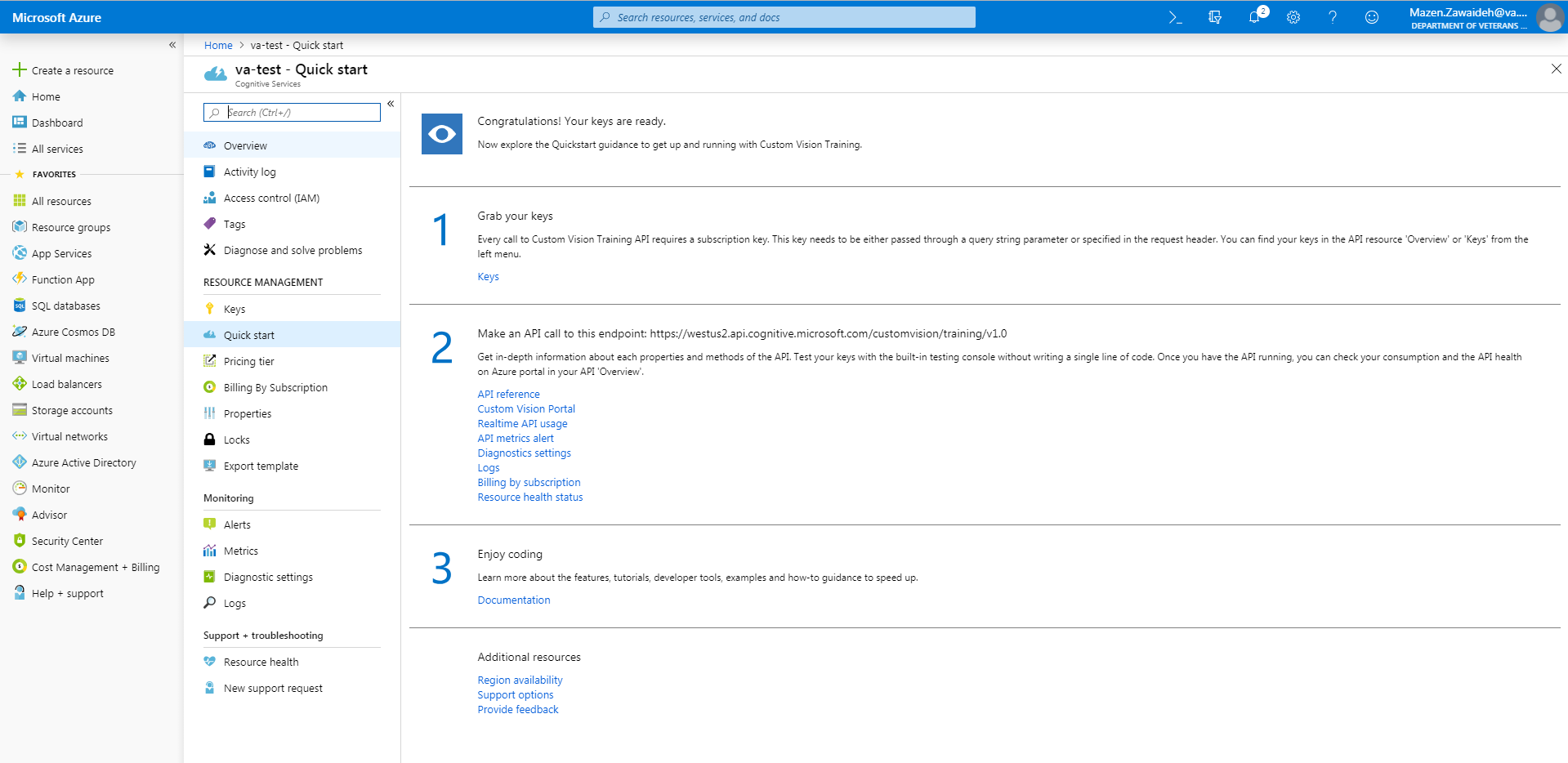
* 1. Follow the menu options to begin provisioning your instance of Custom Vision. For location, choose the region closest to you geographically. If you do not have an existing resource group you wish to use, click create new and name your resource group
  2. An example of the settings is shown below



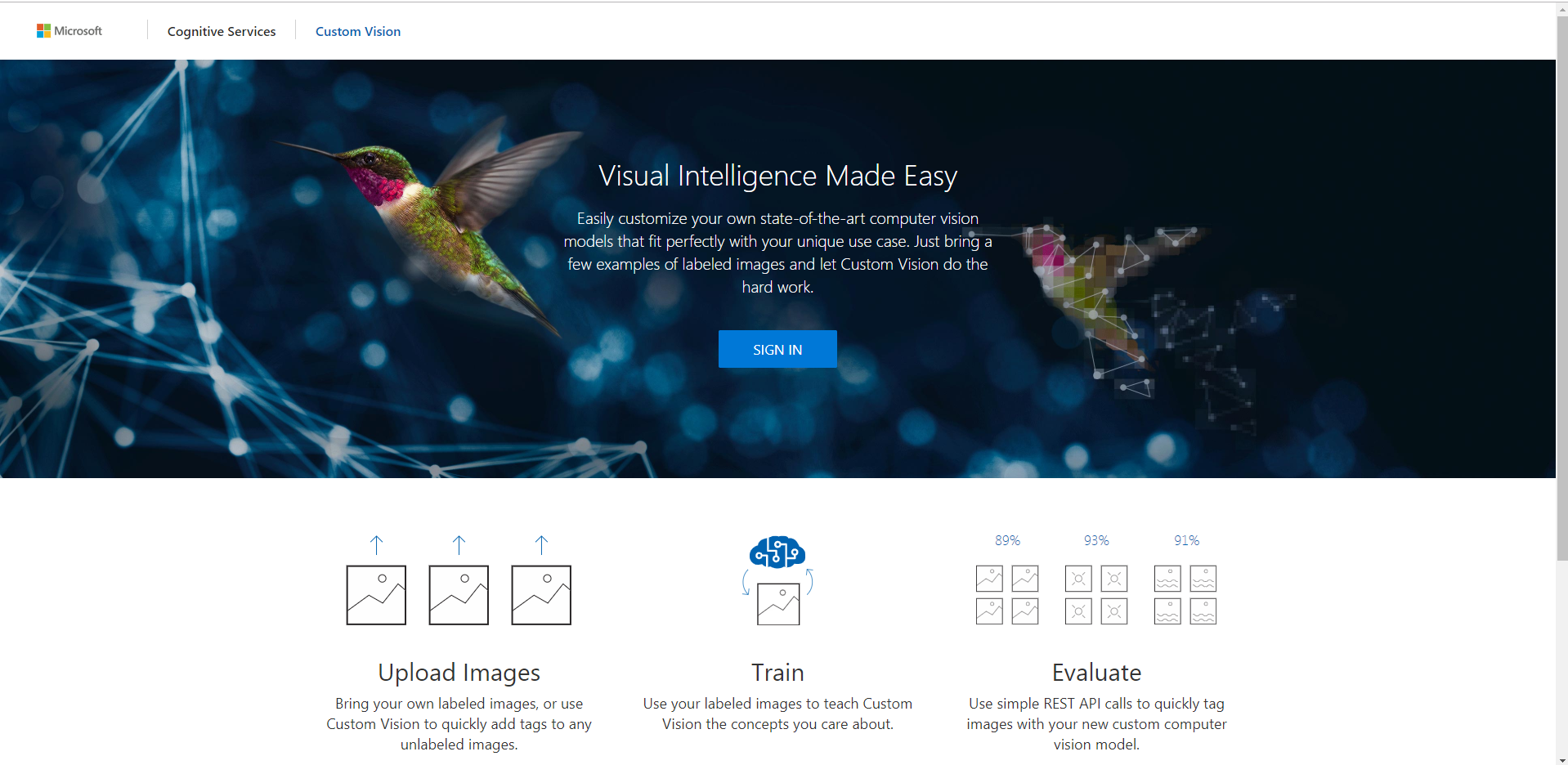
* 1. Once you click create, you are taken back to the portal main page. Note that there will be notifications underneath your bell icon, indicating that custom vision is being provisioned.



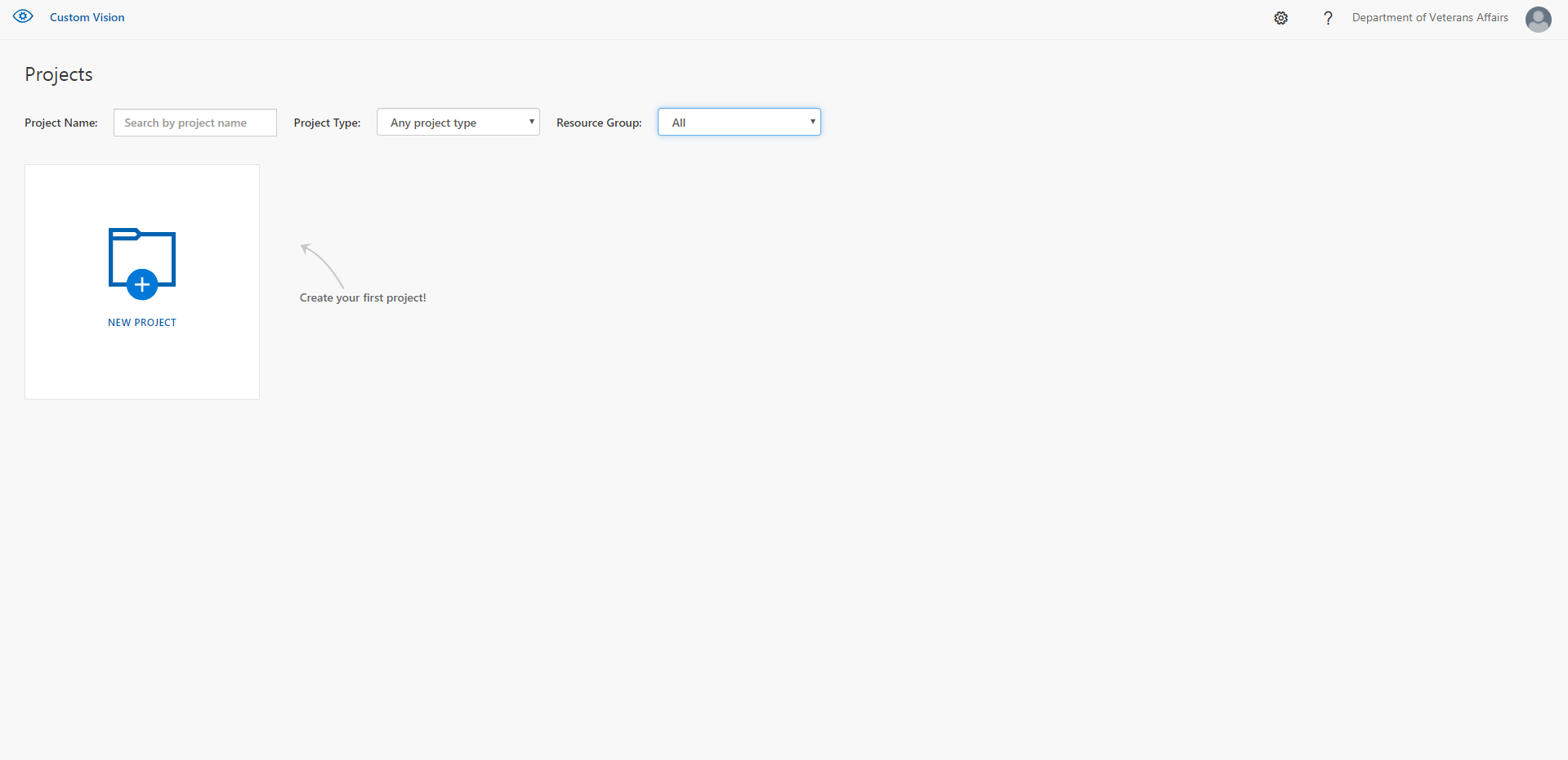
* 1. Once deployed, you can to navigate to Custom Vision. You should see the following screen.



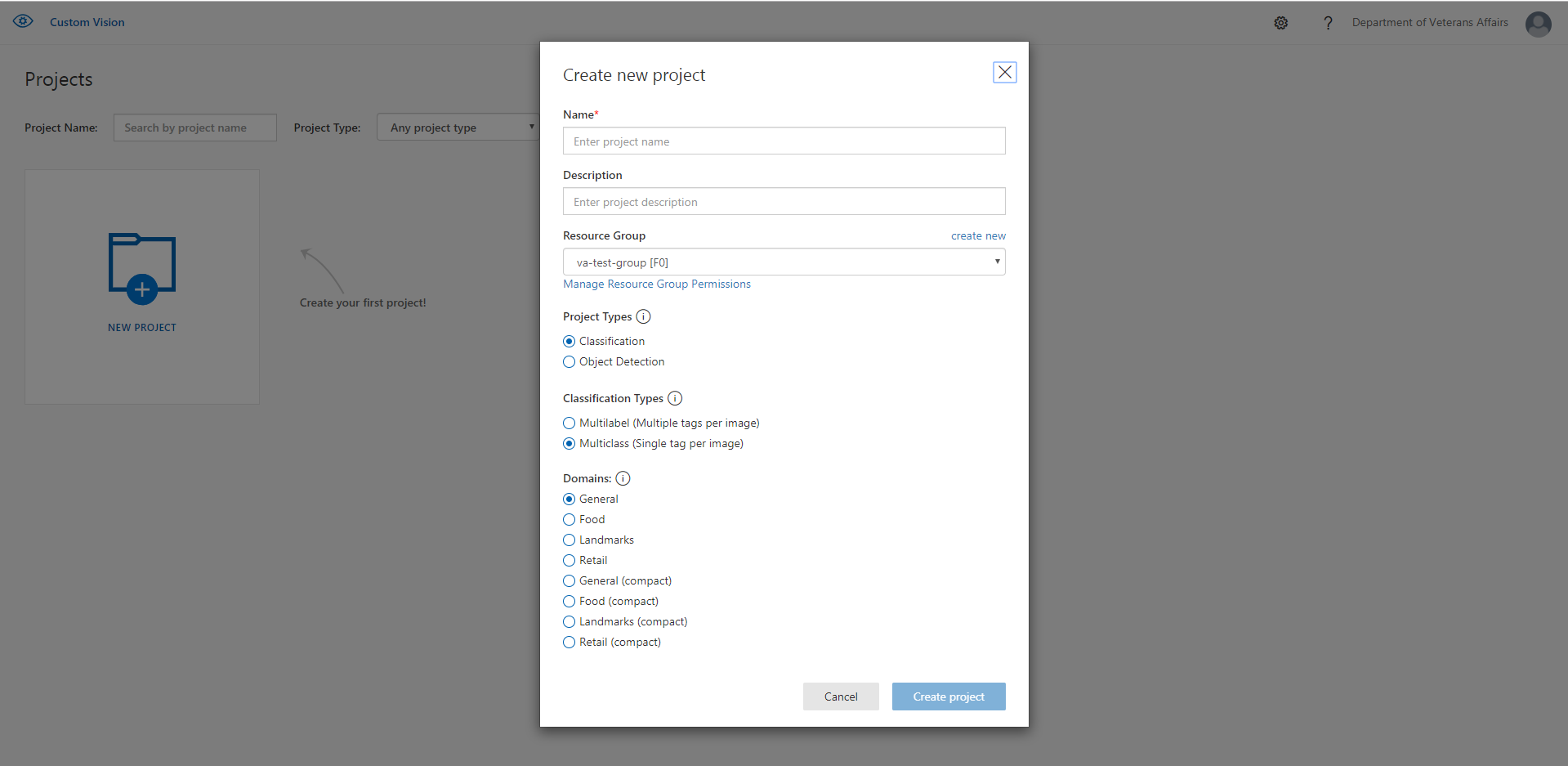
1. Creating Your First Sample Project
   1. Navigate to the Custom Vision online user interface at <https://www.customvision.ai/> and click Sign In. Use the same credentials you used to get into the Azure portal.



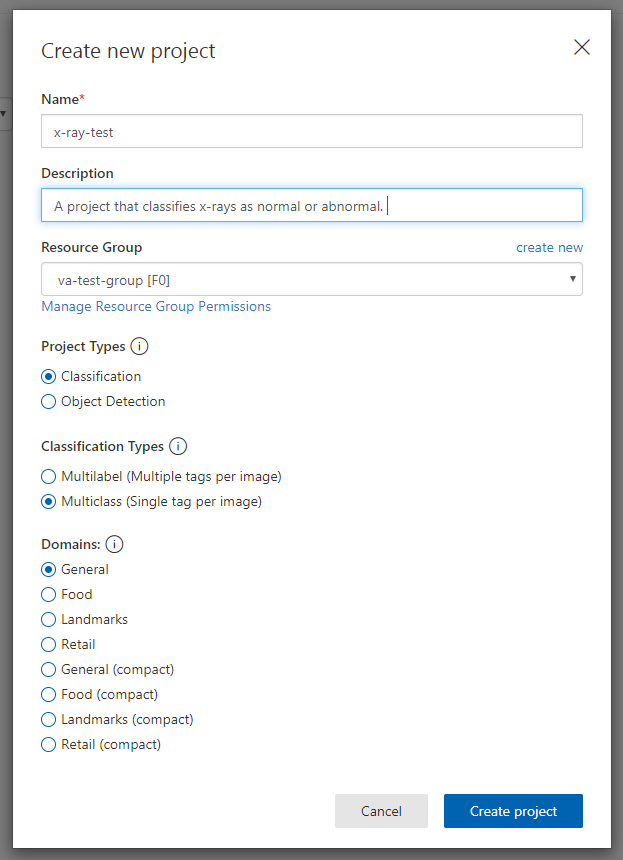
* 1. You will be presented with this interface



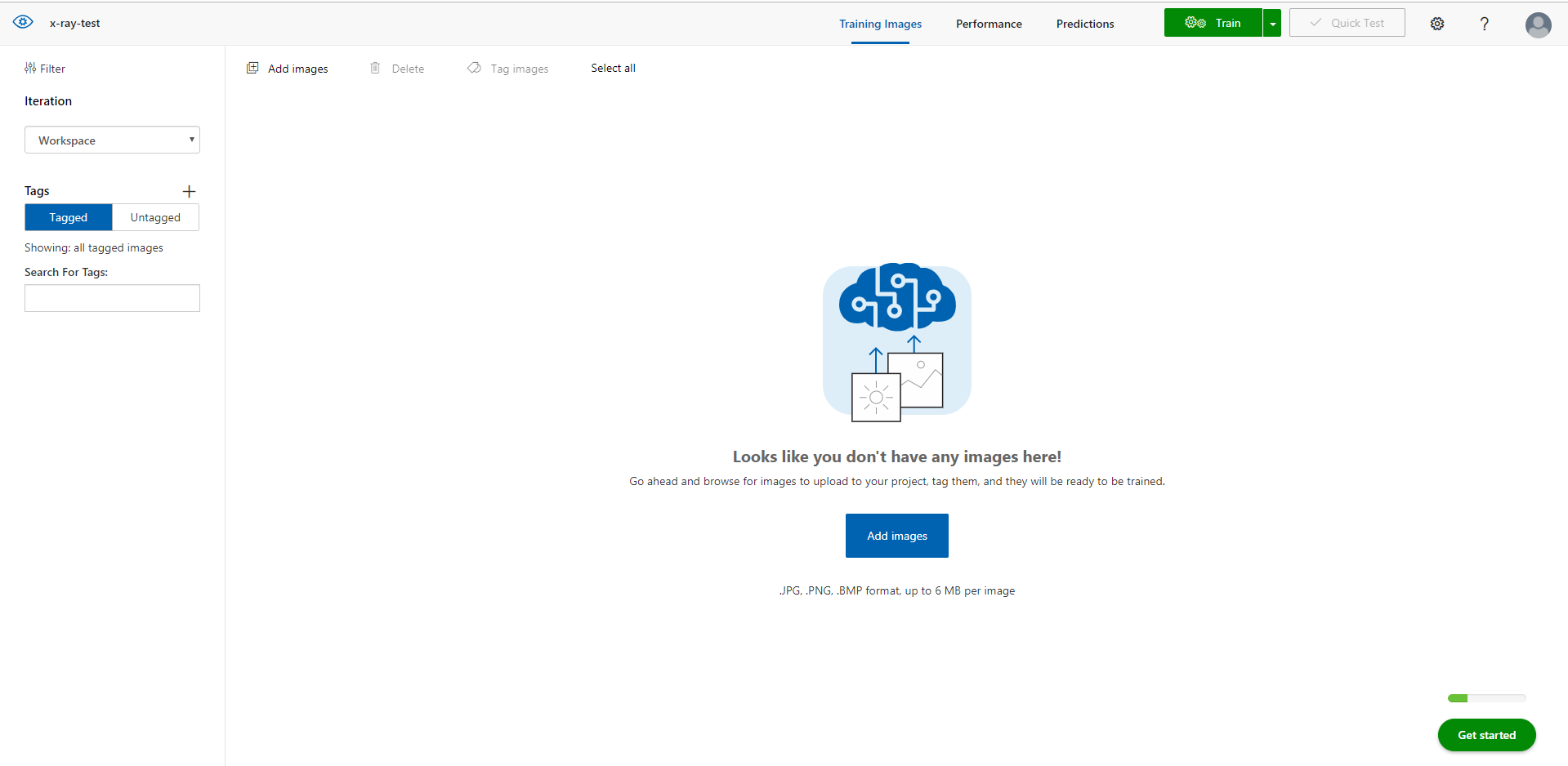
* 1. Let’s create your first project. To do so, begin by clicking the New Project icon.



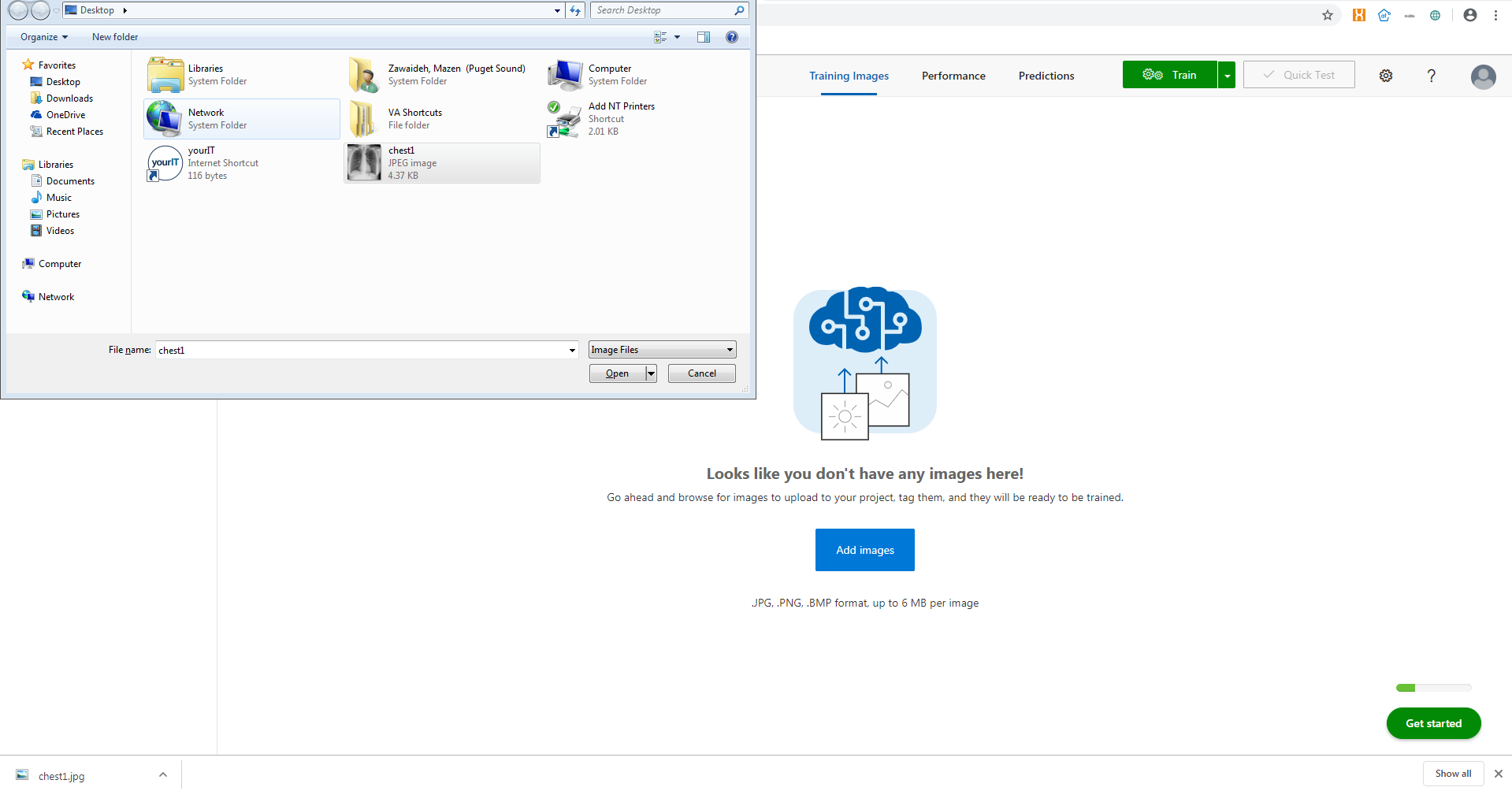
* 1. Begin filling out the required details. Make sure the resource group you choose is the same as the resource group that you used to provision your Custom Vision instance in your Azure portal.



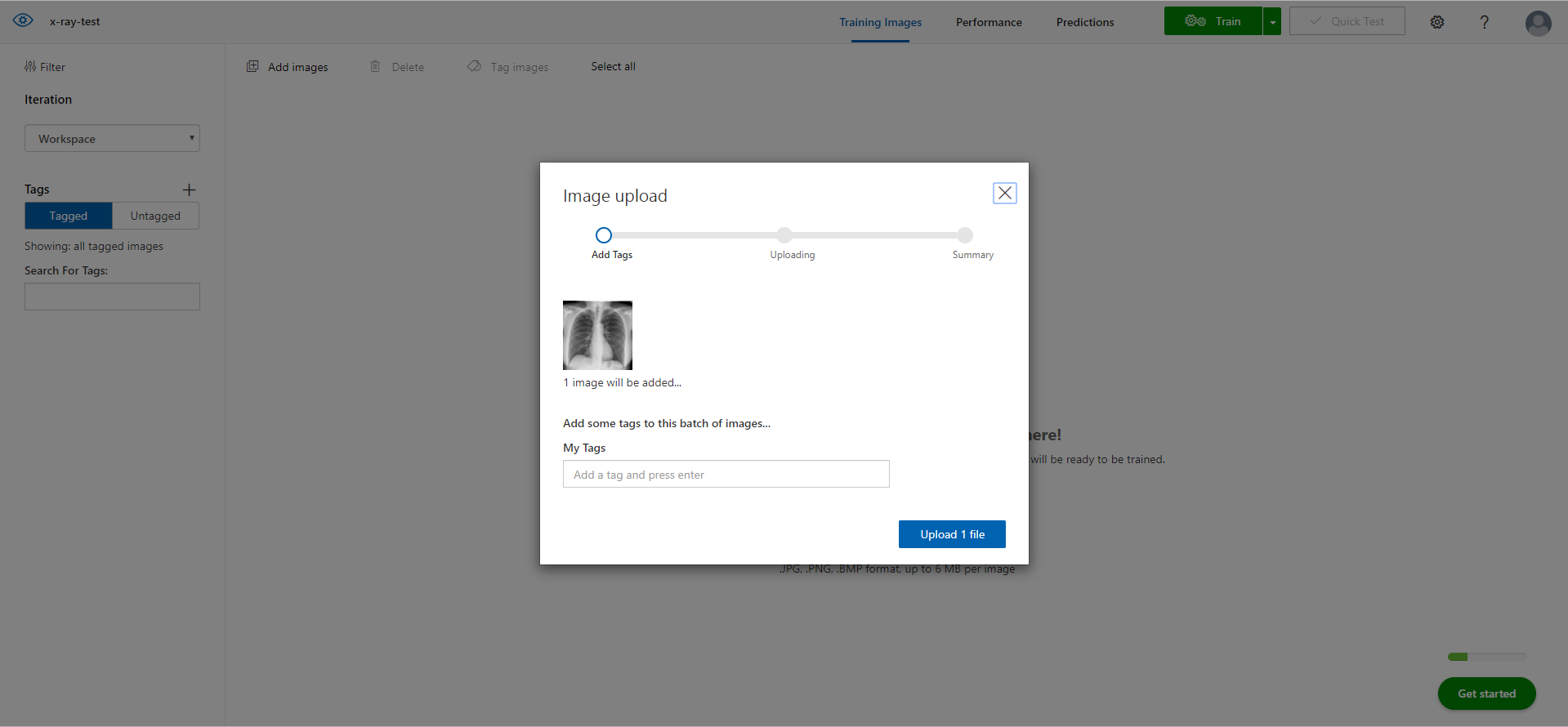
* 1. Filled out form for reference.



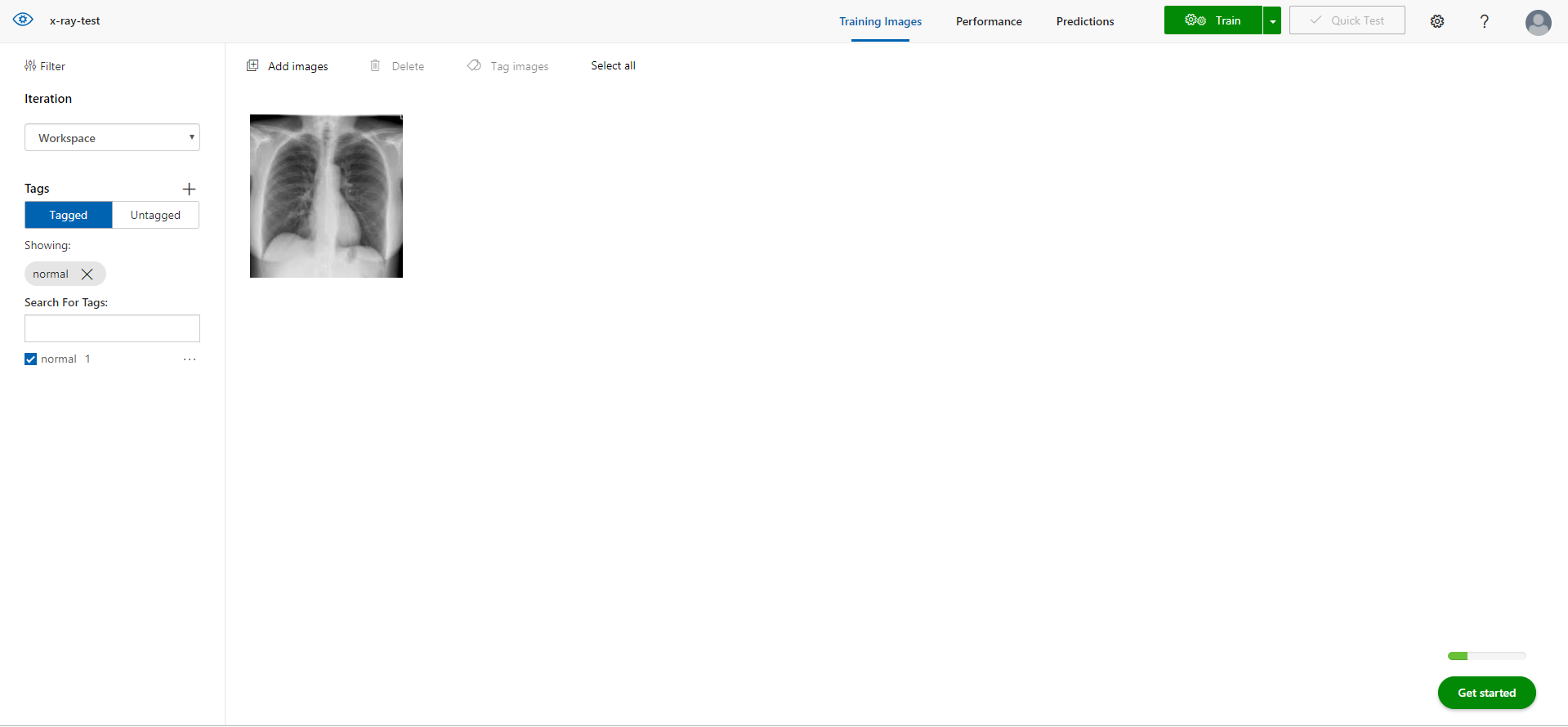
* 1. Once your project is created you will be presented with the following screen



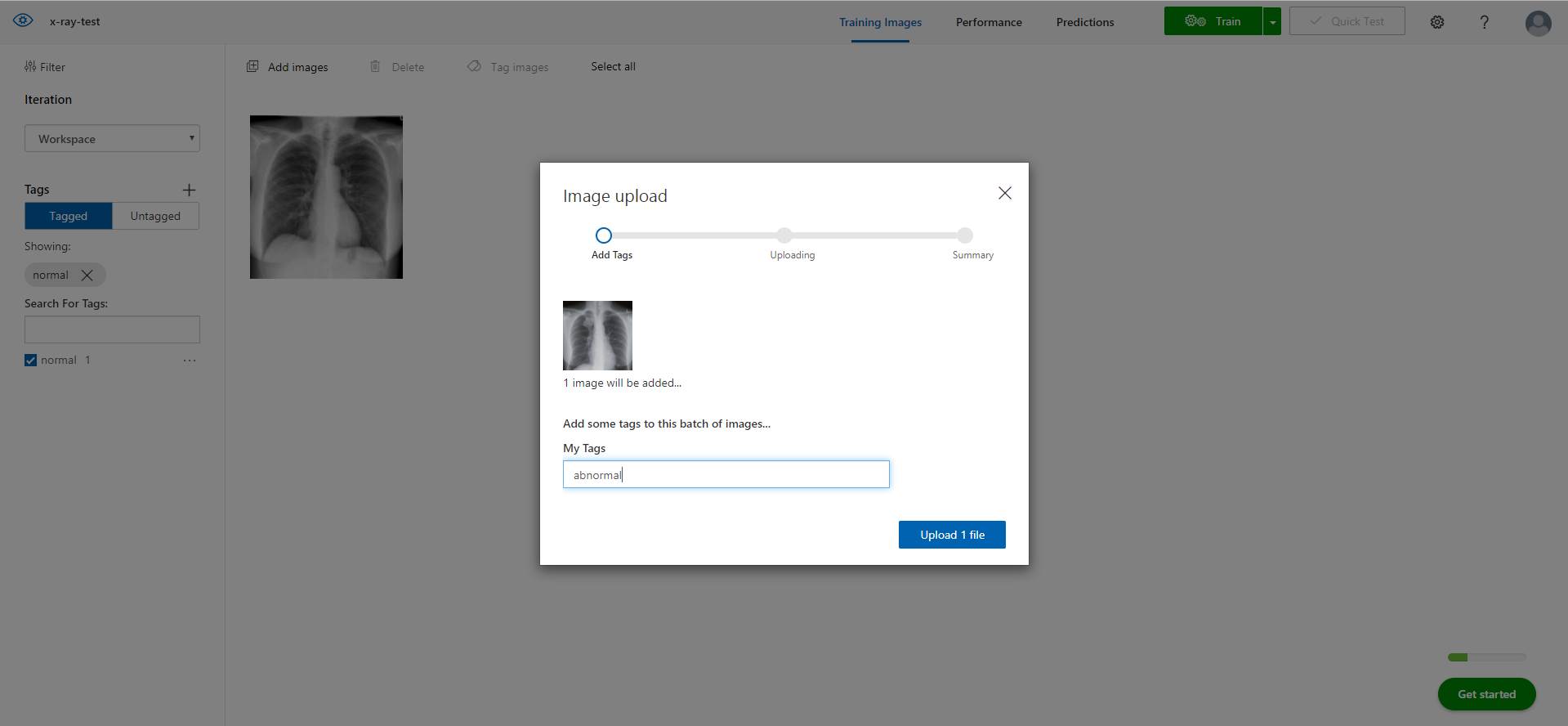
* 1. Let’s begin uploading images. In this case, we are uploading a normal chest xray image saved to the Desktop called chest1. It is assumed you have a set of medical images of interest. If not, the RSNA 2018 Pneumonia Detection Challenge dataset is a great place to begin (<https://www.kaggle.com/c/rsna-pneumonia-detection-challenge>). Please note that Custom Vision does not intake DICOM files. If your files are in the DICOM format, first convert them to JPEG or PNG using any software you prefer (For example, DCMTK <https://support.dcmtk.org/docs/dcmj2pnm.html>).



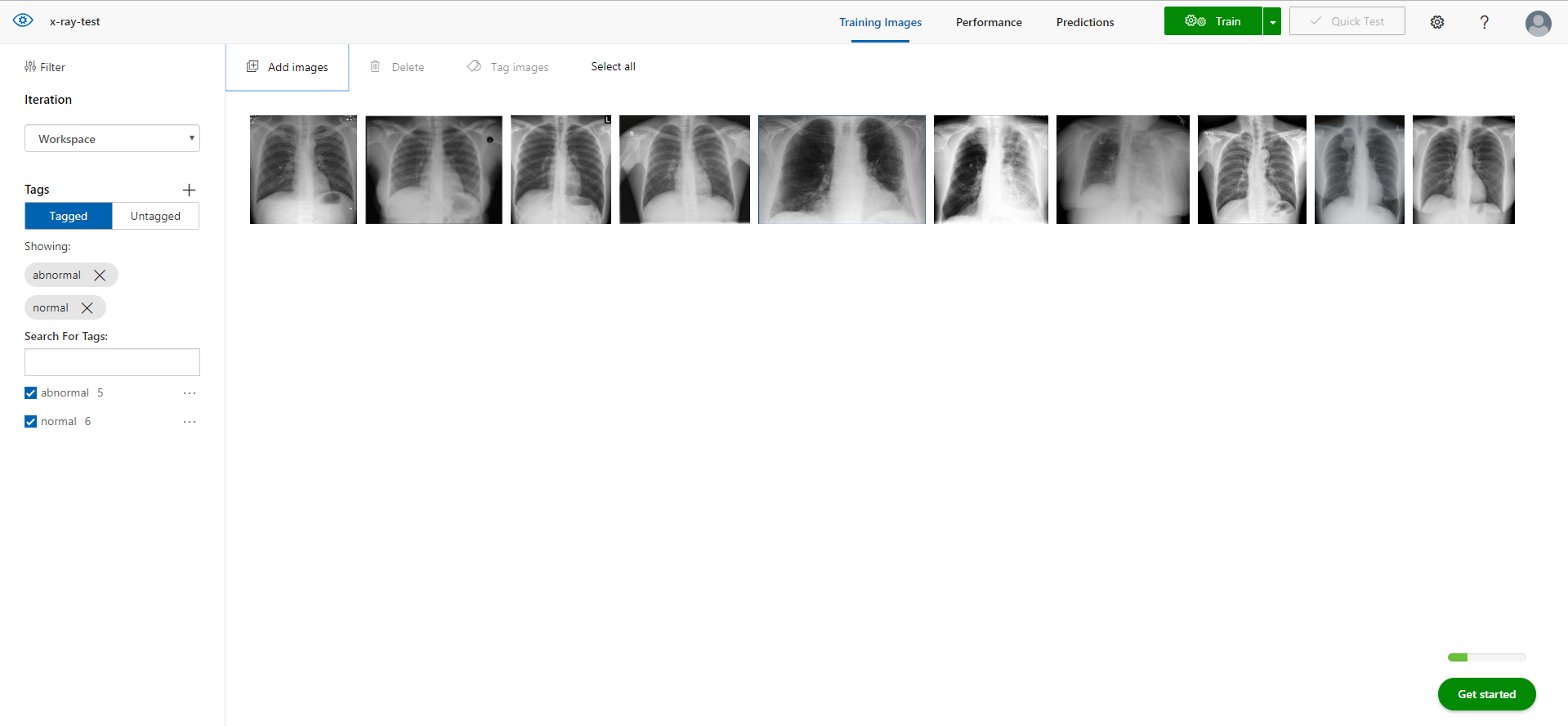
* 1. Once you click open, you are presented with this screen. Since we are building an image classifier to determine whether a chest x-ray is normal or abnormal, we will want to tag our image as either “normal” or “abnormal”. In this case, we will fill in “normal” under the box titled “My Tags”, press enter to make the tag available, then choose that tag for the uploaded image. Once that is done, press “Upload 1 file”.



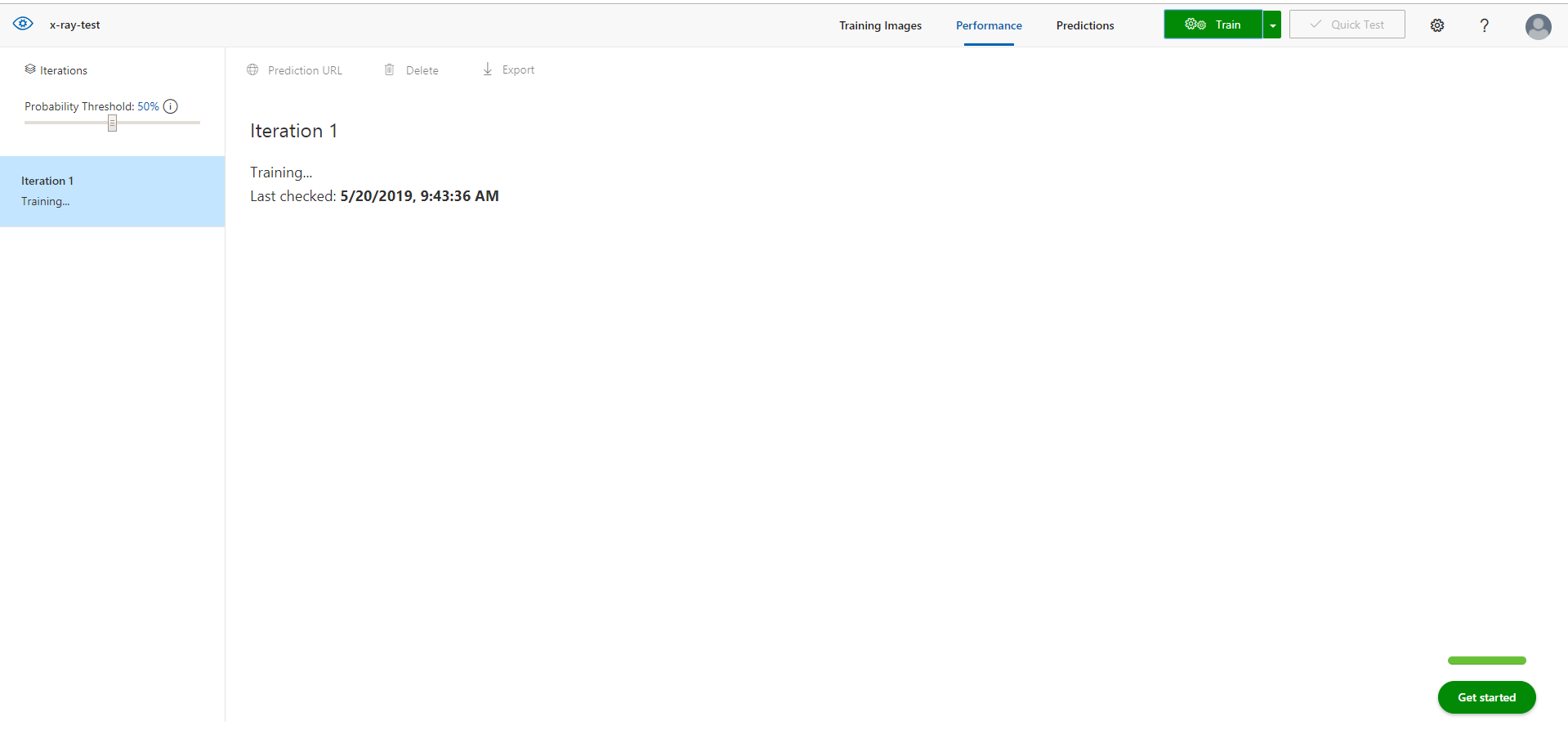
* 1. Once done, we are presented with screen. Let’s upload an abnormal example. To do so we follow the same process, but with a new image that will be assigned an abnormal tag.



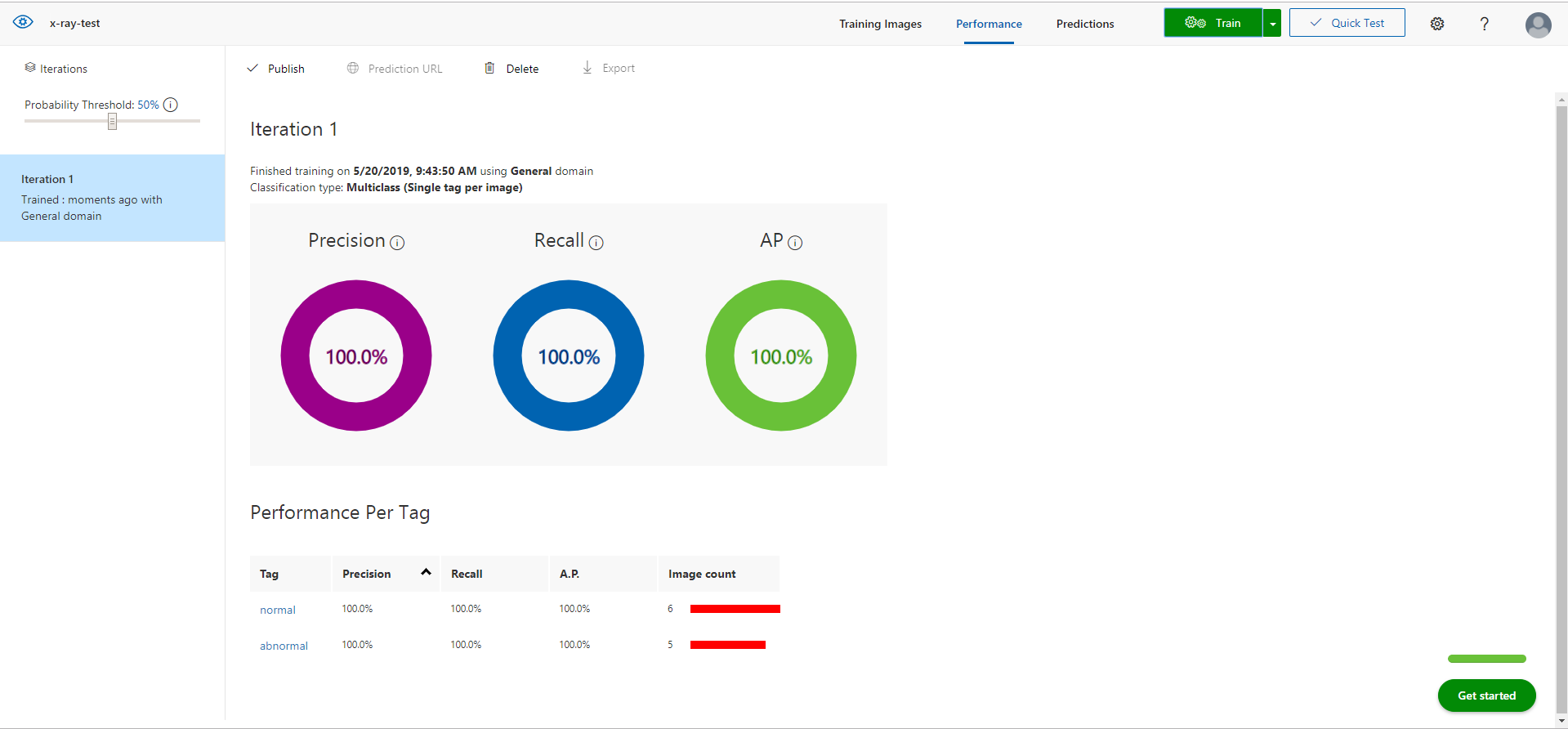
* 1. Example of uploading abnormal image. To train a model, custom vision requires at least 2 categories (aka classes) of images, and at least 5 images per category. Let’s repeat the process until we have enough data.



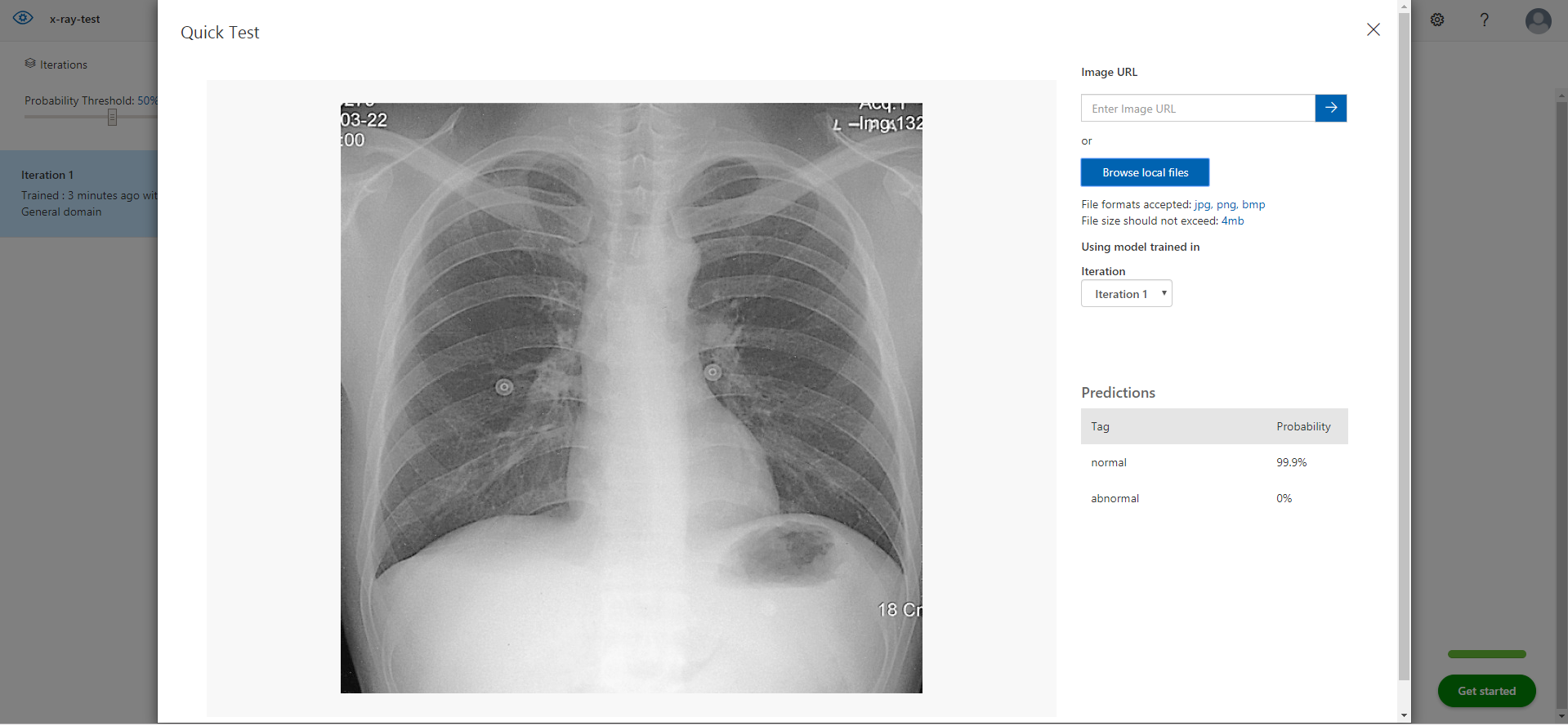
* 1. Now that we have our training images uploaded, we are ready to train our machine learning model. To do so, click the “Train” button on the top right corner. Please note, an actual project will use a dataset on the order of hundreds to thousands of images. This simply serves as an example.



* 1. Once you click train, the model training process begins, and you will be presented with this display.



* 1. Once training has completed, you will be notified.



* 1. Try your classifier out on a new image to see how it performs. To do so, click quick test and upload an image that was not part of the training set. How did the classifier do?
  2. Congratulations on finishing your first demo project using Custom Vision on EDE.