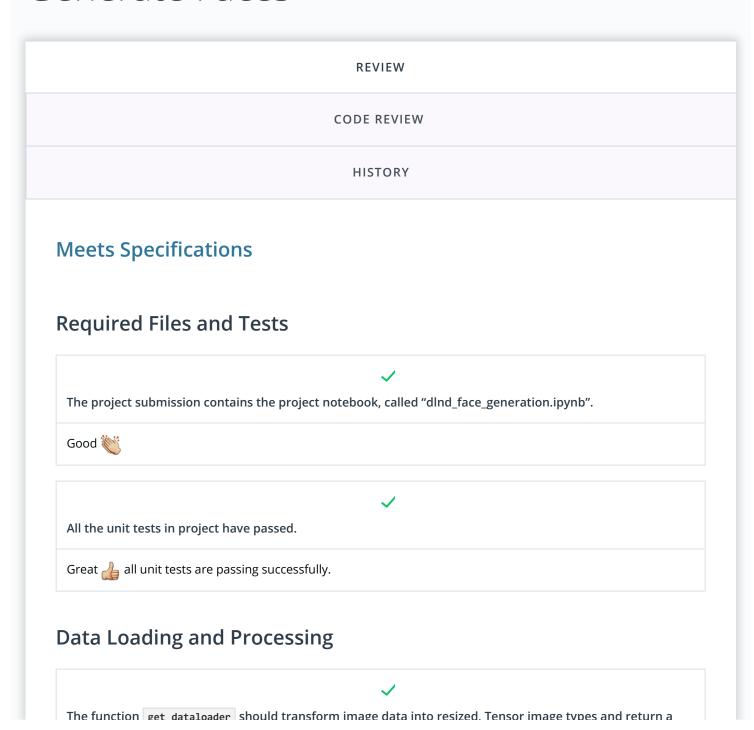


< Return to Classroom

Generate Faces



DataLoader that batches all the training data into an appropriate size.

Good you have used Image Folder wrapper. But also remember the DataLoader approach that used in the assignment of dog breed classification to understand the basics of the framework.

Pre-process the images by creating a scale function that scales images into a given pixel range. This function should be used later, in the training loop.

Great 👍

Build the Adversarial Networks



The Discriminator class is implemented correctly; it outputs one value that will determine whether an image is real or fake.

good 👌 you can also refer explaination from here



The Generator class is implemented correctly; it outputs an image of the same shape as the processed training data.

good 👌 you can also refer explaination from here



This function should initialize the weights of any convolutional or linear layer with weights taken from a normal distribution with a mean = 0 and standard deviation = 0.02.

Good 👑

```
if classname.find("Conv") != -1 or classname.find('Linear') != -1:
    torch.nn.init.normal_(m.weight.data, 0.0, 0.02)
elif classname.find("BatchNorm2d") != -1:
    torch.nn.init.normal_(m.weight.data, 1.0, 0.02)
    torch.nn.init.constant_(m.bias.data, 0.0)
```

